

Machine Learning Using Python

Tan Kwan Chong

Senior Data Scientist, SparkBeyond

Workshop Objectives

- Collect data from a variety of sources
- Explore large data sets
- Clean and "munge" the data to prepare it for analysis
- Apply machine learning algorithms to gain insight from the data
- Visualize the results of your analysis
- Build your own library and Python scripts

Self Introduction



Booz | Allen | Hamilton
strategy and technology consultants

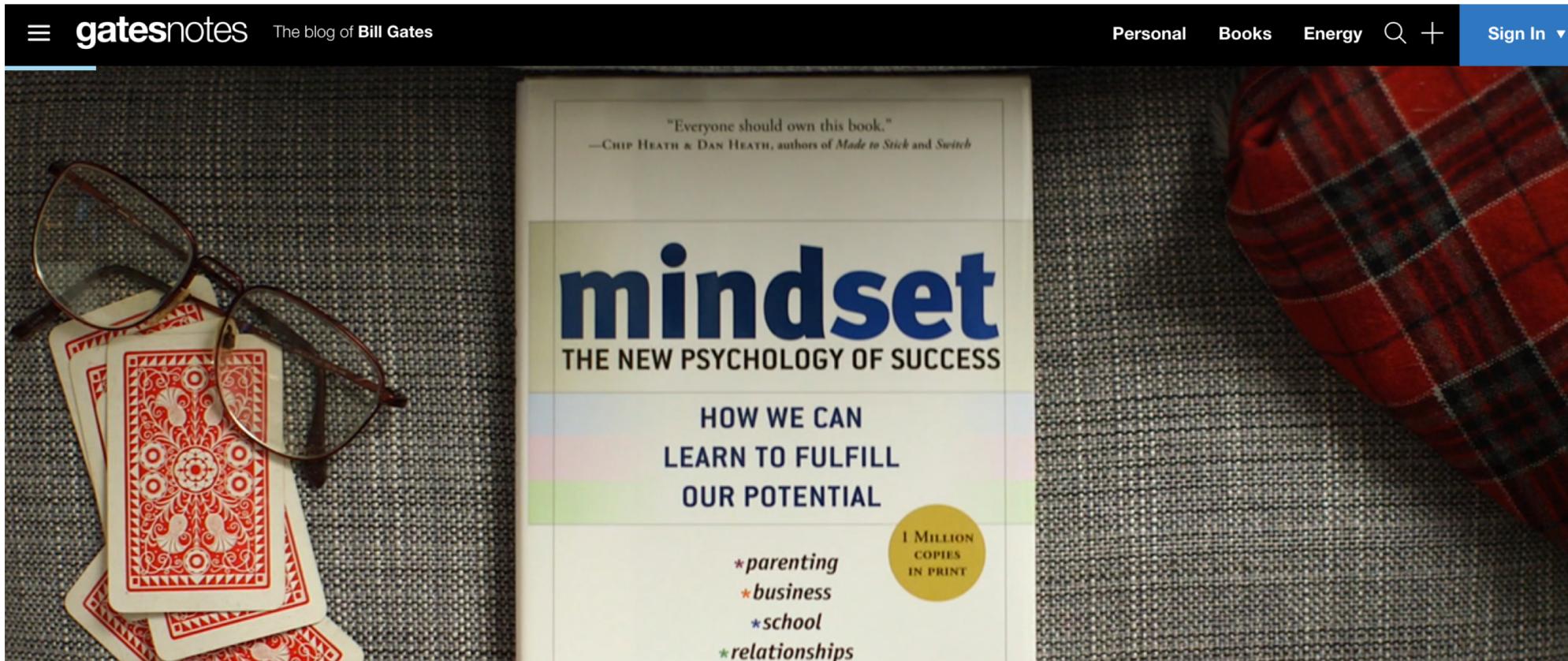


Self Introduction

Hello
my name is

- Name
- Background
- What do you want to gain out of this course?
- Experience in programming and data analytics?

Growth vs Fixed Mindset



Mindset Over Matter

1789 SHARES

97 SHARES

What You Believe Affects What You Achieve

Growth vs Fixed Mindset

10 Growth Mindset Statements

What can I say to myself?

INSTEAD OF: TRY THINKING:



1 What am I missing?
2 I'm on the right track.
3 I'll use some of the strategies we've learned.
4 This may take some time and effort.
5 I can always improve so I'll keep trying.
6 I'm going to train my brain in Math.
7 Mistakes help me to learn better.
8 I'm going to figure out how she does it.
9 Is it really my best work?
10 Good thing the alphabet has 25 more letters!

I'm not good at this.
I'm awesome at this.
I give up.
This is too hard.
I can't make this any better.
I just can't do Math.
I made a mistake.
She's so smart. I will never be that smart.
It's good enough.
Plan "A" didn't work.

Fixed Mindset –
Intelligence and talents are fixed at birth, capabilities derive from DNA and destiny

Growth Mindset –
Basic qualities including intelligence can be strengthened like muscles with practice and perseverance

Day 1 - Developing the Fundamentals

- Introduction to Machine Learning (AM)
 - What is machine learning?
 - Installation and update of tools
- Exploring and using Data Sets (PM)
 - Learn the steps to pre-process a data set and prepare it for machine learning algorithms
 - Introduction to machine learning algorithms – linear & logistic regression

Day 2 - Diving into Machine Learning

- Supervised vs Unsupervised Learning (AM)
 - Decision trees, random forests, K-nearest neighbors
 - K-means, hierarchical clustering
- Model Evaluation (PM)
 - Feature engineering and model selection
 - Model evaluation metrics
 - Overfitting and bias-variance trade-off
 - Cross validation

Introduction to Machine Learning

Background and Context

1 Data is increasingly cheap and ubiquitous



2 New technologies are emerging to organize and make sense of this avalanche of data

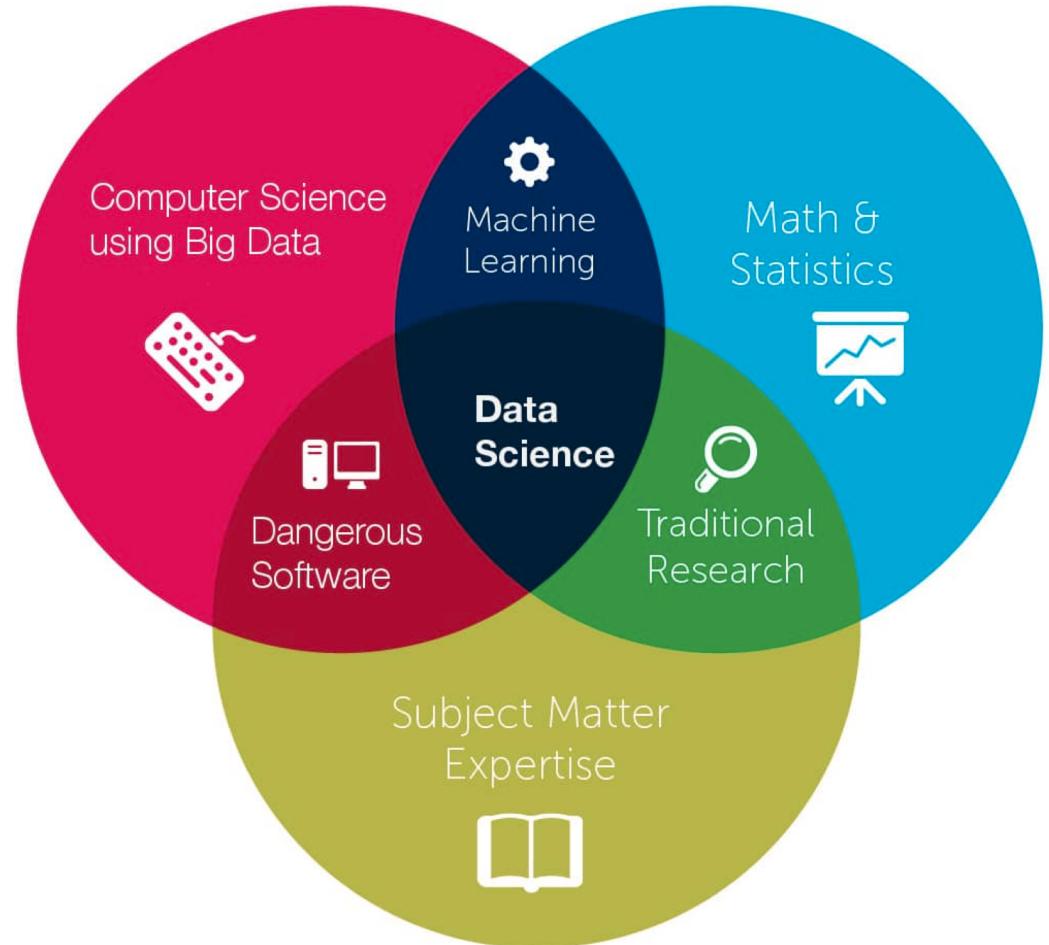


3 Organizations need to harness internal and external data effectively to attract and retain their customers

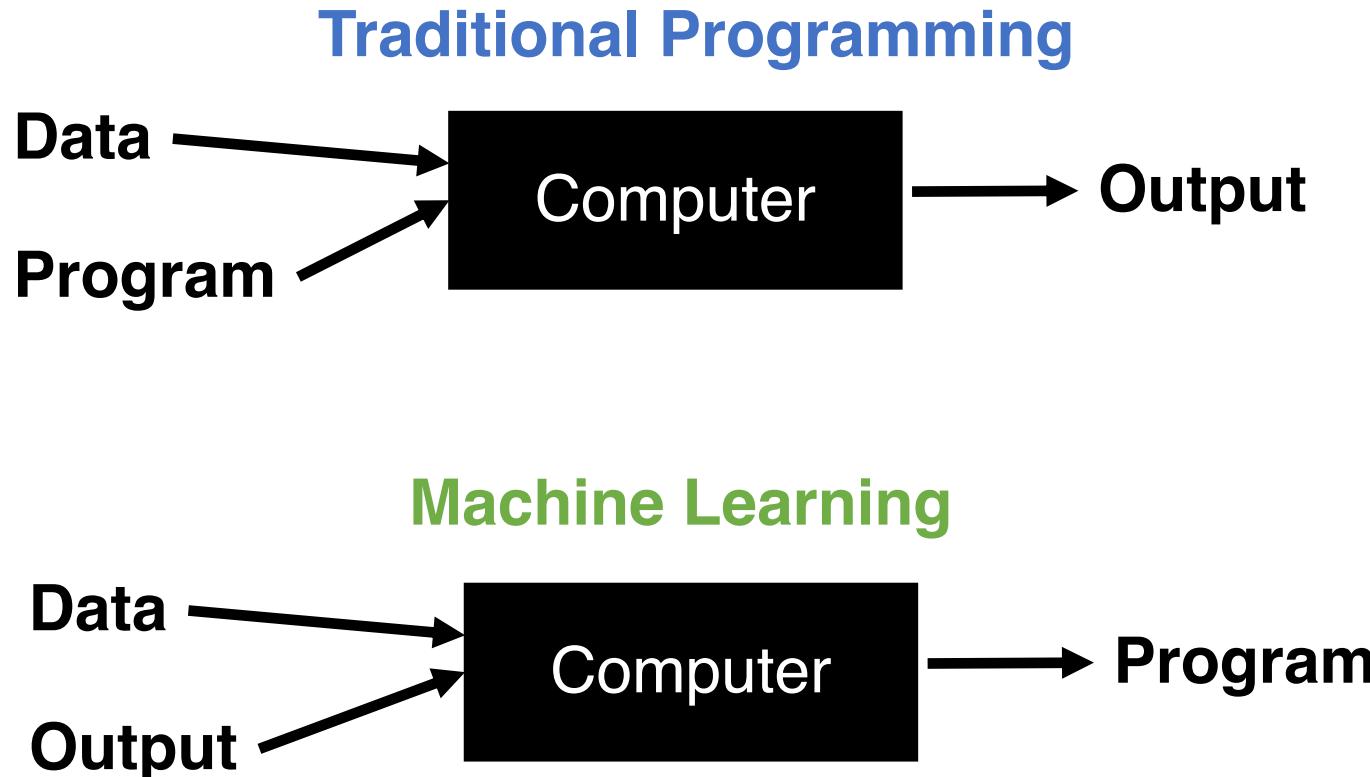


What is Data Science?

- A set of tools and techniques for data
- Interdisciplinary problem-solving
- Application of scientific techniques to practical problems
- The scientific approach to knowledge extraction from data



What is Machine Learning?



Machine learning is a field of computer science that often uses statistical techniques to give computers the ability to “learn” (i.e., progressively improve performance on a specific task) with data, without being explicitly programmed

Who uses Machine Learning?

NETFLIX

amazon.com®



Google

Grab

❖ **FiveThirtyEight**

 **DBS**



GOVTECH
SINGAPORE

LAZADA
Effortless Shopping

Singapore Examples – Is this Machine Learning?



datagovsg [Follow](#)

Official Medium account for <https://data.gov.sg>, Singapore's open data portal.

Dec 1, 2016 · 8 min read

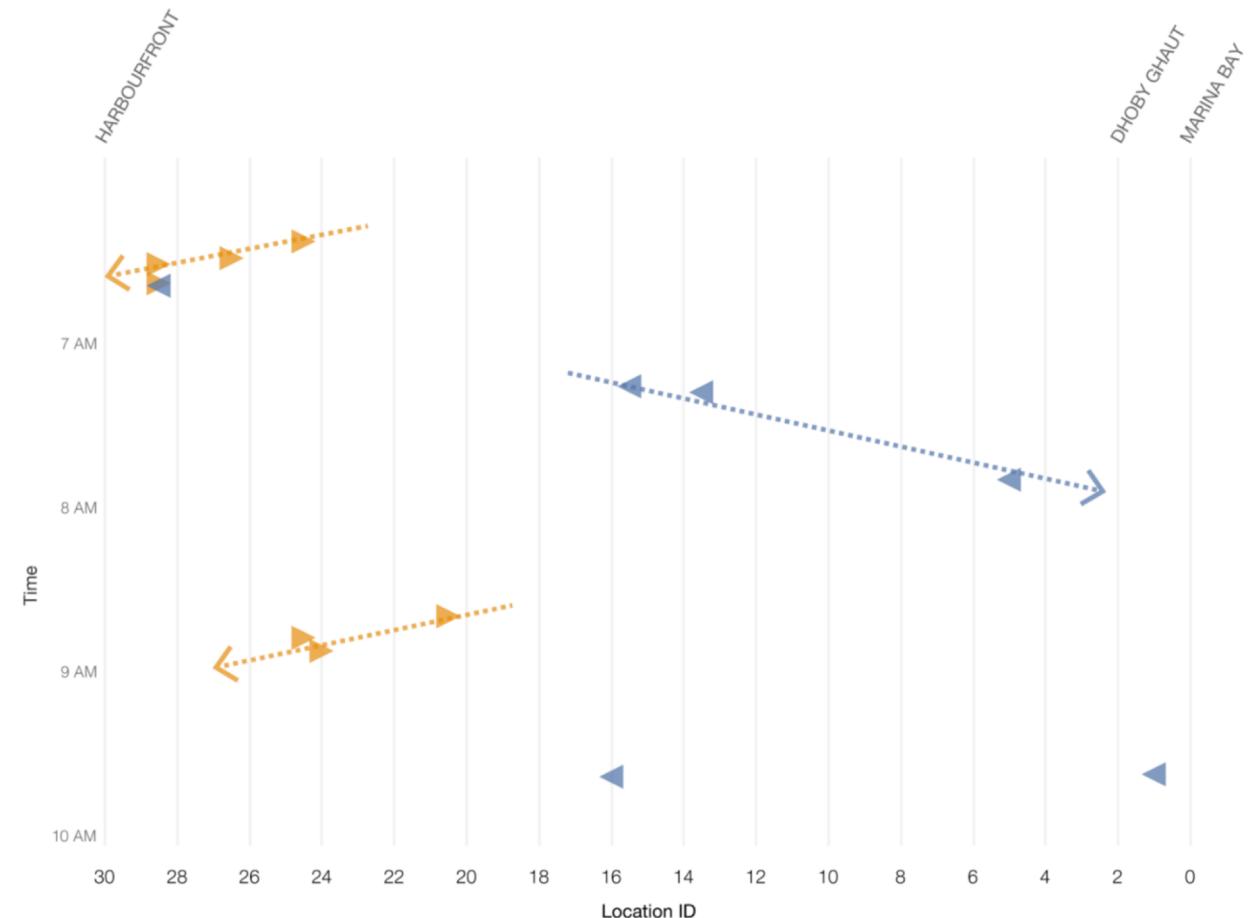
How the Circle Line rogue train was caught with data

Text: Daniel Sim | Analysis: Lee Shangqian, Daniel Sim & Clarence Ng

Singapore's MRT Circle Line was hit by a spate of mysterious disruptions in recent months, causing much confusion and distress to thousands of commuters.



Rogue Train: A Big Data Story



<https://blog.data.gov.sg/how-we-caught-the-circle-line-rogue-train-with-data-79405c86ab6a>

Singapore Examples – Is this Machine Learning?

Yellow taxis have fewer accidents than blue ones, says study



Researchers from NUS and the Chinese University of Hong Kong say that yellow taxis are more noticeable than blue taxis in both daylight and under street lighting. PHOTO: BLOOMBERG NEWS

© PUBLISHED MAR 8, 2017, 5:00 AM SGT | UPDATED MAR 8, 2017, 11:01 AM



Tay Hong Yi

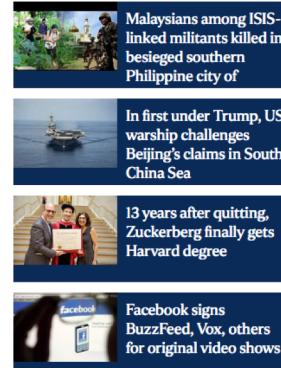
The next time you hail a taxi, take note of its colour, for it might mean a safer ride.

Researchers at the National University of Singapore (NUS) found that taxis painted yellow, a colour that stands out, were involved in significantly fewer traffic accidents than taxis painted blue.

Their results were based on analysing three years' worth of detailed taxi, driver and accident data from a large fleet of over 4,000 yellow taxis and 12,500 blue taxis locally.



ST VIDEOS



Recommended by Outbrain

SPONSORED CONTENT

Yellow taxis have fewer accidents than blue taxis because yellow is more visible than blue

Teck-Hua Ho^{a,b,1}, Juin Kuan Chong^c, and Xiaoyu Xia^d

^aOffice of the Deputy President (Research & Technology), National University of Singapore, Singapore 119077; ^bHaas School of Business, University of California, Berkeley, CA 94720; ^cNational University of Singapore Business School, National University of Singapore, Singapore 119245; and ^dDepartment of Decision Sciences and Managerial Economics, Chinese University of Hong Kong Business School, Chinese University of Hong Kong, Shatin, NT, Hong Kong

Edited by George A. Akerlof, University of California, Berkeley, CA, and approved January 31, 2017 (received for review August 3, 2016)

Is there a link between the color of a taxi and how many accidents it has? An analysis of 36 mo of detailed taxi, driver, and accident data (comprising millions of data points) from the largest taxi company in Singapore suggests that there is an explicit link. Yellow taxis had 6.1 fewer accidents per 1,000 taxis per month than blue taxis, a 9% reduction in accident probability. We rule out driver difference as an explanatory variable and empirically show that because yellow taxis are more noticeable than blue taxis—especially when in front of another vehicle, and in street lighting—other drivers can better avoid hitting them, directly reducing the accident rate. This finding can play a significant role when choosing colors for public transportation and may save lives as well as millions of dollars.

car color | road safety | data science | transportation science | sensory perception

Accidents involving public transport are common and cause significant economic losses as well as loss of human life. Applying statistical analysis to a unique and comprehensive dataset we establish that a change in color can avert a significant number of taxi accidents, leading to a reduction in economic losses. Specifically, analysis of a complete set of accident records from the largest taxi operator in Singapore, which uses yellow and blue taxis, shows that yellow is safer than blue because yellow is more noticeable, with the result that potential accidents are avoided by other drivers' timely responses.

Yellow has been a popular color for taxis since 1907, when the Chicago Yellow Cab Company chose the color based on a survey conducted at the University of Chicago. The survey showed that yellow was the most noticeable color, which would make it easy for potential passengers to spot a yellow taxi in the sea of mass-

demographic characteristics. These two datasets include millions of observations on the company's drivers and taxis, and accidents involving these taxis. The data from both datasets have been anonymized and are available in Datasets S1–S6.

The company uses yellow or blue for all its regular taxis (approximate colors are shown in Fig. 1).[†] The colors are the remnants of a 2002 merger that took place between two taxi companies, one of which used yellow and the other, blue. The company owns ~16,700 taxis in a ratio of one yellow to three blue (1y:3b), which translates to 4,175 yellow taxis and 12,525 blue ones. These account for 60% of the ~27,800 taxis in Singapore.[‡]

To control for the difference in the number of taxis used by the company (1y:3b), we calculated a normalized accident rate using the average number of accidents that occurred per 1,000 taxis

Significance

This paper examines the phenomenon that yellow taxis have fewer accidents than blue taxis. Statistical analysis of a unique and comprehensive dataset suggests that the higher visibility of the color yellow makes it easier for other drivers to avoid getting into accidents with yellow taxis, leading to a lower accident rate. This suggests that color visibility should play a major role in determining the colors used for public transport vehicles.

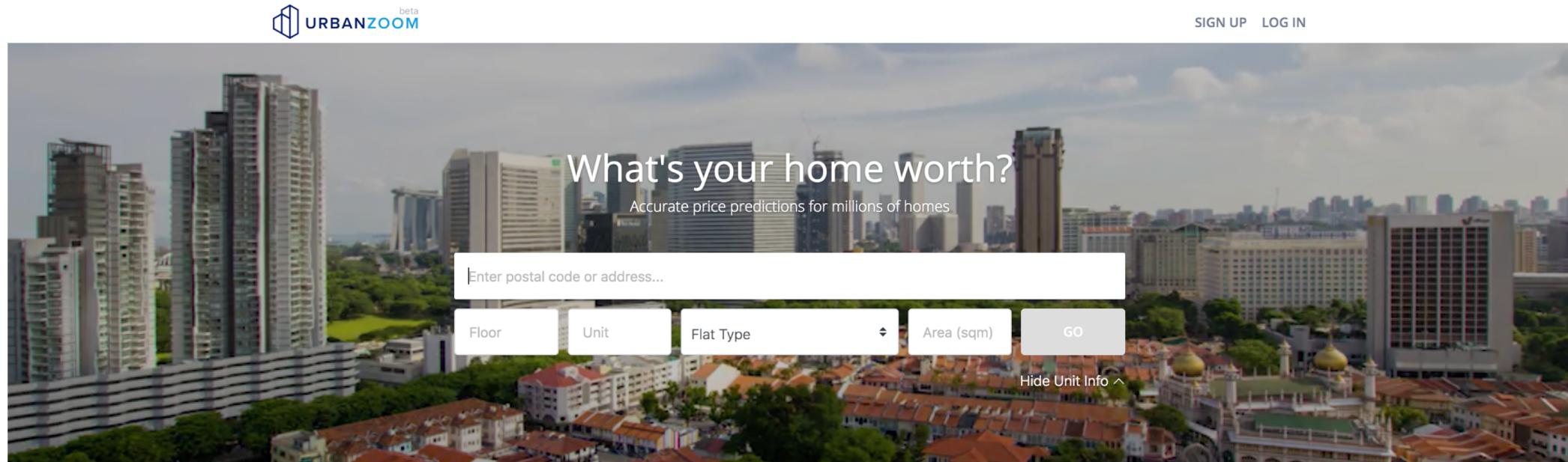
Author contributions: T.-H.H. and J.K.C. designed research; T.-H.H. and J.K.C. performed research; T.-H.H., J.K.C., and X.X. contributed new reagents/analytic tools; T.-H.H., J.K.C., and X.X. analyzed data; and T.-H.H., J.K.C., and X.X. wrote the paper.

The authors declare no conflict of interest.

This article is a PNAS Direct Submission.

Freely available online through the PNAS open access option.

Singapore Examples – Is this Machine Learning?



LATEST PREDICTIONS



Common Questions asked in Machine Learning

How much? How many?

- What will the stock price of Microsoft be next week?
- What will my fourth quarter sales be?
- How many kilowatts of electricity will be demanded between 7-8pm?
- How many new Twitter followers will I get next week?

Regression

- Predict a continuous outcome
- Linear Regression
 - K-Nearest Neighbors
 - Regression Trees

Common Questions asked in Machine Learning

Is this A, B, or C?	Classification
• Is this a fraudulent transaction?	Predict a discrete outcome
• Is this an image of a man, a cat, or a dog?	• Logistic Regression
• Will this customer click on the advertisement?	• K-Nearest Neighbors
• Is the sentiment of the review positive or negative?	• Classification Trees

Common Questions asked in Machine Learning

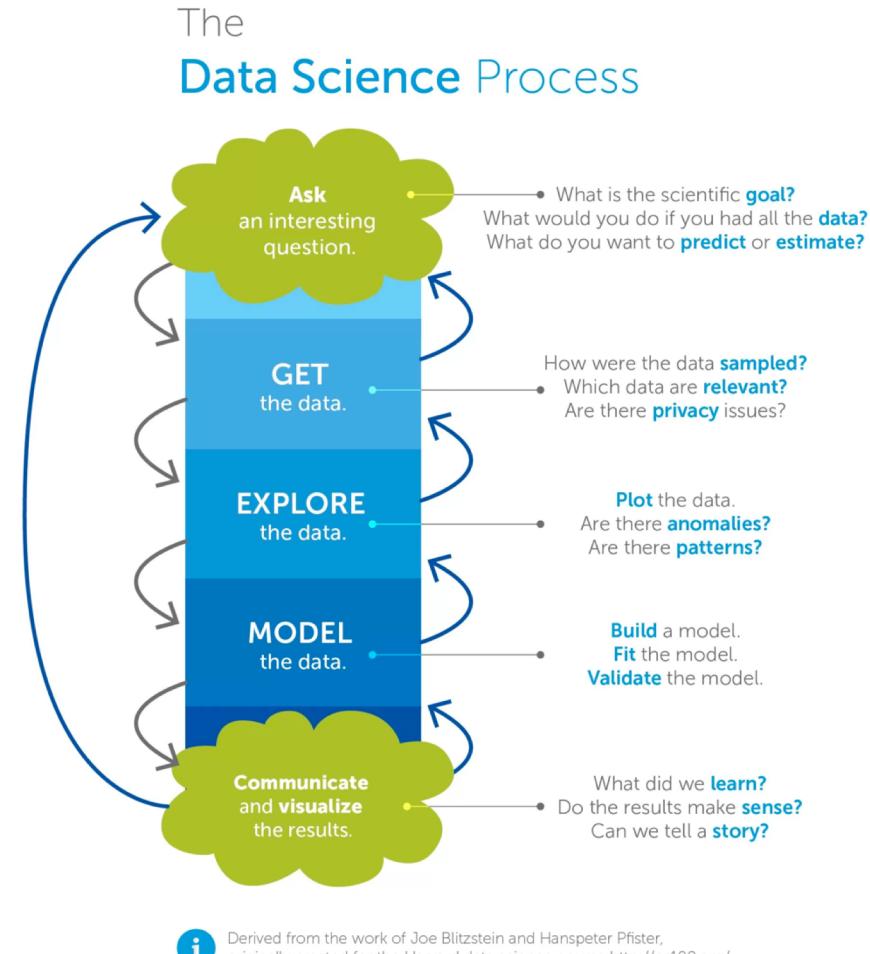
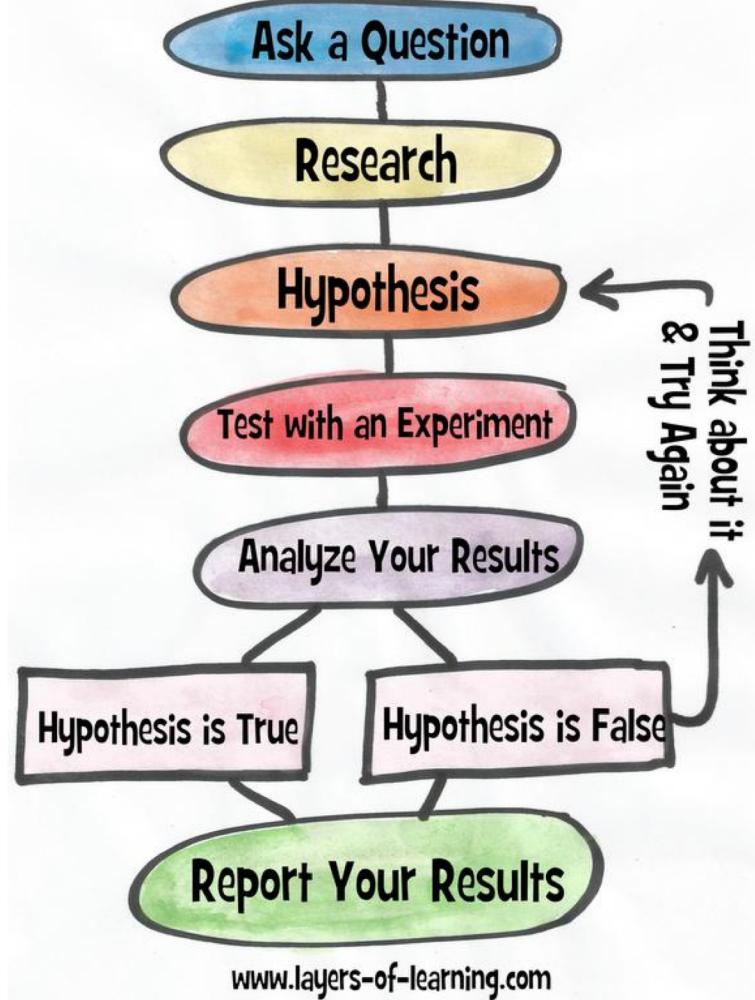
How is this data organized?

- What are the different types of coffee drinkers?
- Which viewers like the same kind of movies?
- Are there common clusters of cable channels that customers tend to purchase together?
- What is a natural way to break these documents into five topics?

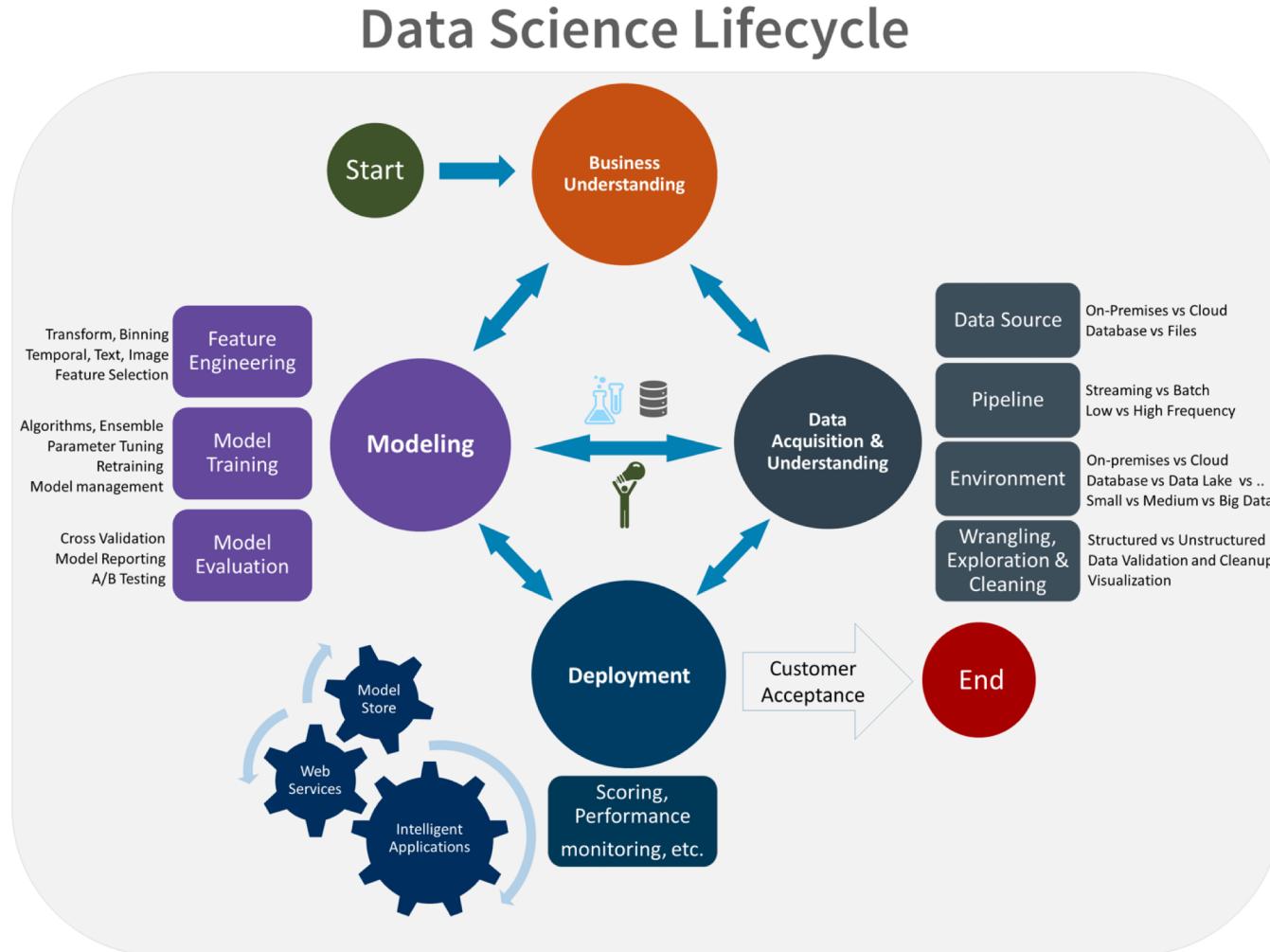
Clustering

- Segment data into different groups
- K-Means Clustering
 - Hierarchical Clustering

Data Science / Machine Learning Process



Data Science Process Lifecycle



Ask an Interesting (Impactful) Question

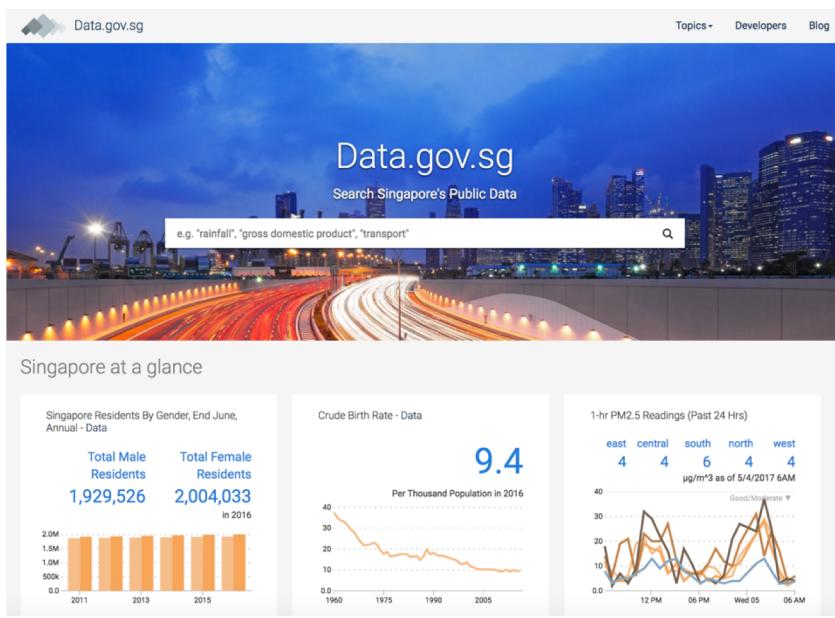
- Begin by identifying the objectives and desired outcomes of the analysis
- Typical scenarios:
 - Given dataset(s) and tasked to explore and uncover insights
 - Specific business problem and need to source the relevant data

Example Problem Statement: What will the expected resale value of my HDB flat be in 10 years time?



Get the Data

- Internal data – spreadsheets, csv files, databases, system logs
- External data – csv files, API requests, web scraping

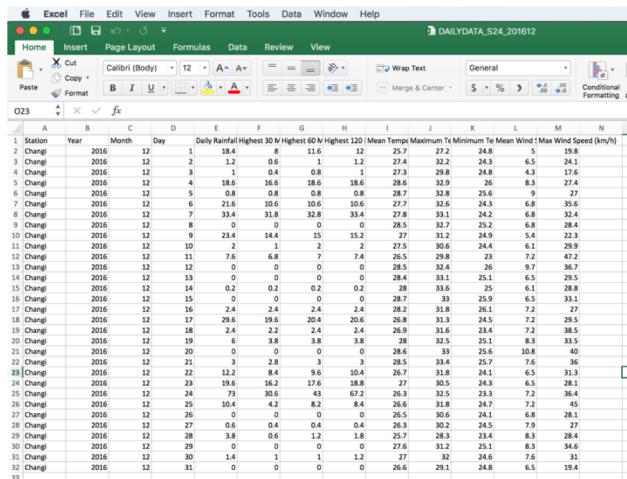


The screenshot shows the "Welcome to Kaggle Datasets" page. The header says "The best place to discover and seamlessly analyze open data". It features three main sections: "Discover" (with a magnifying glass icon), "Explore" (with a lab flask icon), and "Create a dataset" (with a circular icon). Below these are "Learn More" and "New Dataset" buttons. The main content area shows a list of 510 featured datasets, including "European Soccer Database" (25k+ matches, players & teams attributes for European Professional Football) and "IMDB 5000 Movie Dataset" (5000+ movie data scraped from IMDB website).

The screenshot shows the "facebook for developers" website with the "Graph API" tab selected. The left sidebar includes links for "All Docs", "Graph API", "Overview", "Using the Graph API", "Reference", "Common Scenarios", "Other APIs", "Webhooks", "Advanced", and "Changelog". The main content area is titled "The Graph API" and describes it as the primary way for apps to read and write to the Facebook social graph. It includes sections for "Overview", "Using the Graph API", and "API Reference". The "API Reference" section provides details on nodes, edges, and fields. The "Graph API and SDKs" section covers iOS, Android, JavaScript, and PHP SDKs.

Get the Data

- Structured data – well formatted, data schema defined, easy to parse
- Unstructured data – may require additional processing or programming tools to extract relevant metadata e.g. sentiments, word count, entities
- Semi-structured data – may require use of regex or programming logic to extract relevant fields e.g. logs, JSON, XML



Station	Year	Month	Day	Daily Rainfall	Highest 30 N	Highest 60 N	Highest 120	Mean Temp	Maximum Tx	Minimum Tx	Mean Wind	Max Wind Speed (km/h)
1 Changi	2016	12	1	18.4	8	11.6	12	25.7	27.2	24.8	5	19.8
2 Changi	2016	12	2	0.2	0.6	1.2	1.2	27.4	32.2	24.3	6.5	24.1
3 Changi	2016	12	3	1	0.4	0.8	1	27.3	29.8	24.4	4.9	27.4
4 Changi	2016	12	4	18.6	16.6	18.6	18.6	28.6	32.9	26	8.3	27.4
5 Changi	2016	12	5	0.8	0.8	0.8	0.8	28.7	32.8	25.6	9	27
7 Changi	2016	12	6	21.6	10.6	10.6	10.6	27.7	32.6	24.3	6.8	35.6
8 Changi	2016	12	7	31.4	10.5	32.8	32.8	27.3	33.1	24.2	6.9	33.4
9 Changi	2016	12	8	0	0	0	0	28.5	32.7	25.2	6.8	28.4
10 Changi	2016	12	9	23.4	14.4	15	15.2	27	31.2	24.9	5.4	22.3
11 Changi	2016	12	10	2	1	2	2	27.5	30.6	24.4	6.1	29.9
12 Changi	2016	12	11	7.6	6.8	7	7	26.5	29.8	23	4.7	33.5
13 Changi	2016	12	12	0	0	0	0	28.5	32.4	26	9.7	36.7
14 Changi	2016	12	13	0	0	0	0	28.4	33.1	25.1	6.5	29.5
15 Changi	2016	12	14	0.2	0.2	0.2	0.2	28	32	25	6.1	28.8
16 Changi	2016	12	15	0	0	0	0	28.7	33	25.9	6.5	33.1
17 Changi	2016	12	16	2.4	2.4	2.4	2.4	28.2	31.8	26.1	7.2	27
18 Changi	2016	12	17	29.6	19.6	20.4	20.6	26.8	31.9	24.5	7.2	29.5
19 Changi	2016	12	18	2.4	2.4	2.4	2.4	26.9	31.6	23.8	7.2	38.5
20 Changi	2016	12	19	6	3.8	3.8	3.8	28.5	32.5	25.1	7.8	33.5
21 Changi	2016	12	20	0	0	0	0	28.6	33	25.6	10.8	40
22 Changi	2016	12	21	3	2.8	3	3	28.5	33.4	25.7	7.6	36
23 Changi	2016	12	22	12.2	8.4	9.6	10.4	26.7	31.8	24.1	6.5	31.3
24 Changi	2016	12	23	33.8	18.1	21.5	23.8	27	30.5	24.3	28.1	33.3
25 Changi	2016	12	24	73	30.6	43	67.2	26.3	32.5	23.3	7.2	36.4
26 Changi	2016	12	25	10.4	4.2	8.2	8.4	26.6	31.8	24.7	7.2	45
27 Changi	2016	12	26	0	0	0	0	26.5	30.6	24.1	6.8	28.1
28 Changi	2016	12	27	0.8	0.4	0.4	0.4	25.5	30.2	24.5	4.9	37
29 Changi	2016	12	28	3.8	0.6	1.2	1.8	25.7	28.3	23.4	8.3	28.4
30 Changi	2016	12	29	0	0	0	0	27.6	31.2	25.1	8.3	34.6
31 Changi	2016	12	30	1.4	1	1	1.2	27	32	24.6	7.6	31
32 Changi	2016	12	31	0	0	0	0	26.6	29.1	24.0	6.5	19.4

Structured



ORAL ANSWERS TO QUESTIONS
FEATURES FOR MERGED SKILLSFUTURE AND JOBS BANK PORTALS

1 Ms Sun Xueling asked the Minister for Education (Higher Education and Skills) (a) whether the merger of the SkillsFuture and Jobs Bank portals will incorporate a review function for training providers to ensure outcome-based training provisions; (b) whether the merged portal can include internship opportunities for young Singaporeans; (c) whether the merged portal can include personality assessments that gauge the fit of the user to jobs as opposed to just skills; and (d) whether jobs listed in the Jobs Bank are regularly updated to ensure relevance.

The Parliamentary Secretary to the Ministers for Education (Assoc Prof Dr Muhammad Faishal Ibrahim) (for the Minister for Education (Higher Education and Skills)) : Madam, the Individual Learning Portfolio (ILP) is designed to be a one-stop online portal which empowers individuals to make informed learning and career choices. There will be platforms for individuals to review and provide feedback on the training programmes available on the ILP, and information on training outcomes will be published on the portal.

```
893252015.307 14 <client-ip> TCP_HIT/200 227 GET http://images.go.net.com/metacrawler/images/transparent.gif - NONE/- image/gif
893252015.312 23 <client-ip> TCP_HIT/200 4170 GET http://images.go.net.com/metacrawler/images/header.gif - NONE/- image/gif
893252015.318 38 <client-ip> TCP_HIT/200 400 GET http://images.go.net.com/metacrawler/images/bp2.gif - NONE/- image/gif
893252015.629 800 <client-ip> TCP_REFRESH_HIT/200 303 872 GET http://www.metacrawler.com/text/html
893252015.728 355 <client-ip> TCP_HIT/200 5691 GET http://images.go.net.com/metacrawler/images/market2.gif - NONE/- image/gif
893252016.129 465 <client-ip> TCP_HIT/200 219 GET http://images.go.net.com/metacrawler/templates/tips/.../images/pixel.gif - NONE/- image/gif
893252016.479 757 <client-ip> TCP_REFRESH_HIT/200 2106 GET http://images.go.net.com/metacrawler/templates/tips/.../images/ultimate.jpg - DIRECT/images.go.net.com/image/jpeg
```

```
Pretty Raw Preview JSON ▾
1 <odoato_metadata>: "http://datamall2.mytransport.sg/todatodataService$metadatalCorParkAvailability",
2 <value>: [
3   {
4     "CorParkID": 1,
5     "Area": "Marina",
6     "Neighborhood": "Sentosa City",
7     "Latitude": 1.2937,
8     "Longitude": 103.85718,
9     "Lots": 1468
10   },
11   {
12     "CorParkID": 2,
13     "Area": "Marina",
14     "Neighborhood": "Sentosa Square",
15     "Development": "The Sentosa Square",
16     "Latitude": 1.29315,
17     "Longitude": 103.85728,
18     "Lots": 1789
19   },
20   {
21     "CorParkID": 4,
22     "Area": "Marina",
23     "Development": "The Esplanade",
24     "Latitude": 1.29081,
```

Unstructured

Semi-structured

Get the Data

Data.gov.sg

Topics ▾ Developers Blog

Resale Flat Prices (Based on Registration Date), From March 2012 Onwards

Display 10 records Search: Filter

Resale Flat Prices (Based on Approval Date), 2000 - Feb 2012

Resale Flat Prices (Based on Approval Date), 1990 - 1999

Month	Town	Flat Type	Block	Street Name	Storey Range	Floor Area (Sqm)	Flat Model	Lease Commence Date	Resale Price (\$\$)
2017-03	WOODLANDS	5 ROOM	742	WOODLANDS CIRCLE	10 TO 12	122	Improved	1997	428,000
2017-03	WOODLANDS	5 ROOM	749	WOODLANDS CIRCLE	07 TO 09	122	Improved	1998	398,888
2017-03	WOODLANDS	5 ROOM	787C	WOODLANDS CRES	01 TO 03	123	Improved	1997	380,000
2017-03	WOODLANDS	5 ROOM	502A	WOODLANDS DR 14	07 TO 09	123	Improved	1998	443,000
2017-03	WOODLANDS	5 ROOM	526	WOODLANDS DR 14	10 TO 12	126	Premium Apartment	2000	430,000
2017-03	WOODLANDS	5 ROOM	503	WOODLANDS DR 14	01 TO 03	122	Improved	1998	400,000
2017-03	WOODLANDS	5 ROOM	504	WOODLANDS DR 10 TO 12	122	Improved	1998	418,000	

Showing 1 to 10 of 192512 records (Only last 2000 records shown)

« 1 2 3 4 5 ... 200 »

Resale Flat Prices (Based on Registration Date), From March 2012 Onwards Embed View

Resale Flat Prices

Managed by Housing and Development Board

Resale transacted prices. Prior to March 2012, data is based on date of approval for the resale transactions. For March 2012 onwards, the data is based on date of registration for the resale transactions.

Download

Explore the Data

- Review the data dictionary, understand what columns are present
- Check for missing values, data quality issues, outliers and anomalies

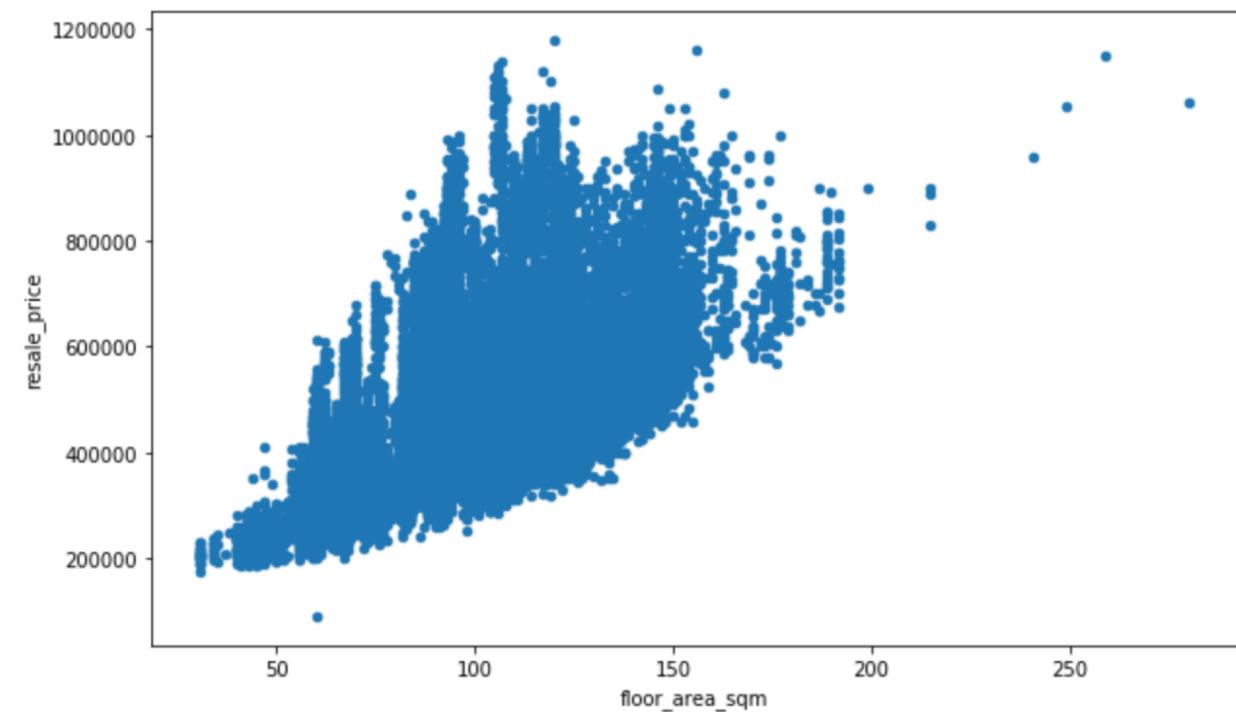
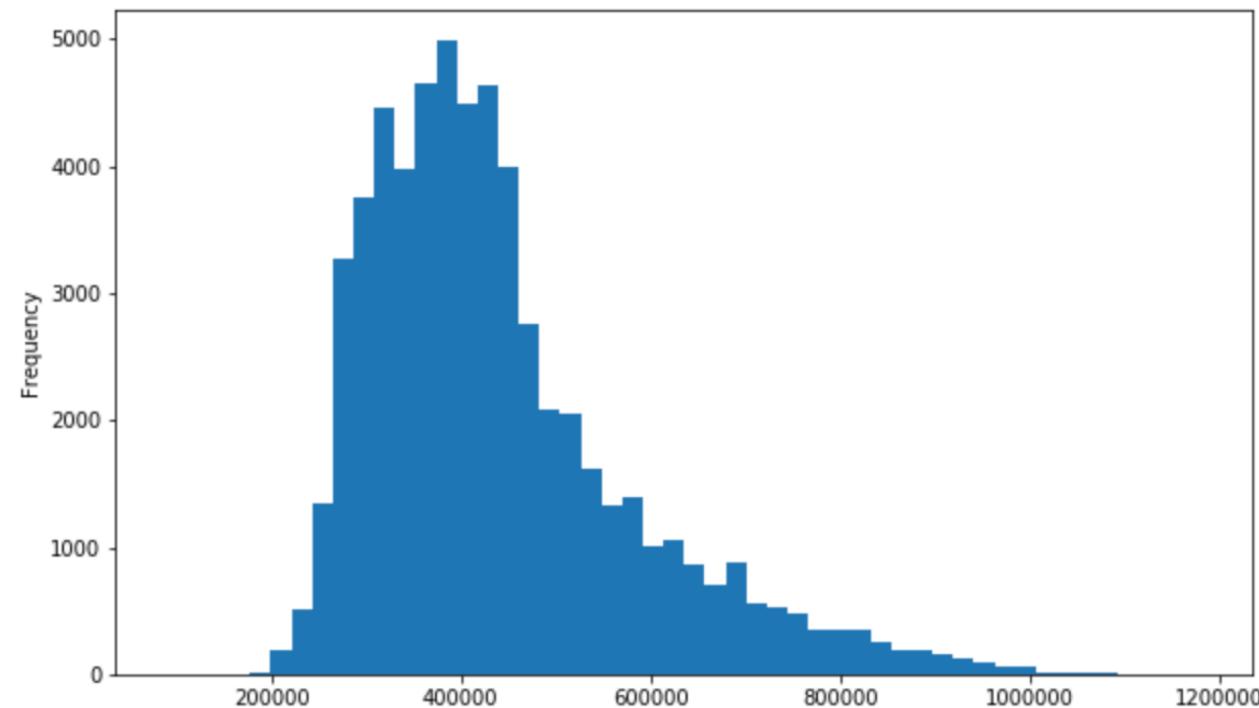
```
# Metadata for Resale Flat Prices
Identifier: '7a339d20-3c57-4b11-a695-9348adfd7614'
Name: 'resale-flat-prices'
Title: 'Resale Flat Prices'
Description:
- 'Resale transacted prices.'
- 'Prior to March 2012, data is based on date of approval for the resale transactions.'
- 'For March 2012 onwards, the data is based on date of registration for the resale transactions.'
Topics:
- 'Infrastructure'
Keywords:
- 'Cost of Living'
- 'HDB'
- 'Housing'
- 'Property'
- 'Public Housing'
- 'Resale Flats'
Publisher:
Name: 'Housing and Development Board'
Admin 1:
Name: 'Michelle Tay'
Department: 'CDG'
Email: 'Michelle_MB_TAY@hdb.gov.sg'
Sources:
- 'Housing and Development Board'
License: 'https://data.gov.sg/open-data-licence'
Frequency: 'Monthly'
Coverage: '1990-01-01 to 2017-04-30'
Last Updated: '2017-05-15T07:03:31.090642'
Resources:
- Identifier: '83b2fc37-ce8c-4df4-968b-370fd818138b'
Title: 'Resale Flat Prices (Based on Registration Date), From March 2012 Onwards'
Url: 'https://storage.data.gov.sg/resale-flat-prices/resources/resale-flat-prices-based-on-regis'
Format: 'CSV'
Coverage: '2012-03-01 to 2017-04-30'
Last Updated: '2017-05-15T07:03:30.565947'
Schema:
- Name: 'month'
Title: 'Month'
Type: 'datetime'
Sub Type: 'month'
Format: 'YYYY-MM'
```

resale_prices.head()											
month	town	flat_type	block	street_name	storey_range	floor_area_sqm	flat_model	lease_commence_date	remaining_lease	resale_price	
0	2015-01	ANG MO KIO	3 ROOM	174	ANG MO KIO AVE 4	07 TO 09	60.0	Improved	1986	70	255000.0
1	2015-01	ANG MO KIO	3 ROOM	541	ANG MO KIO AVE 10	01 TO 03	68.0	New Generation	1981	65	275000.0
2	2015-01	ANG MO KIO	3 ROOM	163	ANG MO KIO AVE 4	01 TO 03	69.0	New Generation	1980	64	285000.0
3	2015-01	ANG MO KIO	3 ROOM	446	ANG MO KIO AVE 10	01 TO 03	68.0	New Generation	1979	63	290000.0
4	2015-01	ANG MO KIO	3 ROOM	557	ANG MO KIO AVE 10	07 TO 09	68.0	New Generation	1980	64	290000.0

resale_prices.isnull().values.any()										
False										
resale_prices.describe()										
count	59809.000000		59809.000000	59809.000000	5.980900e+04					
mean	97.319194		1991.607200		74.017890	4.394339e+05				
std	24.166368		11.283492		11.274598	1.415803e+05				
min	31.000000		1966.000000		47.000000	9.050000e+04				
25%	75.000000		1984.000000		66.000000	3.400000e+05				
50%	96.000000		1989.000000		72.000000	4.100000e+05				
75%	111.000000		2001.000000		83.000000	5.000000e+05				
max	280.000000		2015.000000		97.000000	1.180000e+06				

Explore the Data

- Plot and visualize the data to understand distributions, trends and patterns



Model the Data

- Parse data into formats that are usable by the machine learning algorithm
- Create dummies (one hot encoding) for categorical values
- Perform feature engineering on the data

```
resale_prices = resale_prices.rename(columns={'month': 'year-month'})  
resale_prices['year'] = resale_prices['year-month'].apply(lambda x: int(x.split("-")[0]))  
resale_prices['month'] = resale_prices['year-month'].apply(lambda x: int(x.split("-")[1]))  
resale_prices['lower_storey_range'] = resale_prices['storey_range'].apply(lambda x: int(x.split()[0]))  
resale_prices['upper_storey_range'] = resale_prices['storey_range'].apply(lambda x: int(x.split()[2]))
```

```
df_flat_type = pd.get_dummies(resale_prices['flat_type'])  
resale_prices = pd.concat([resale_prices, df_flat_type], axis=1)
```

Model the Data

- Predicting a continuous variable -> regression model
- Target variable -> resale_price
- Iteratively add, remove, or modify input variables, validate errors

```
from sklearn import linear_model
from sklearn.metrics import mean_squared_error

factors = ["floor_area_sqm", "upper_storey_range", "remaining_lease", "1 ROOM", "2 ROOM", "3 ROOM",
           "4 ROOM", "5 ROOM", "EXECUTIVE", "MULTI-GENERATION"]

reg = linear_model.LinearRegression()
reg.fit(resale_train[factors], resale_train["resale_price"])

train_rmse = mean_squared_error(resale_train["resale_price"],
                                 reg.predict(resale_train[factors]))**0.5
test_rmse = mean_squared_error(resale_test["resale_price"],
                                reg.predict(resale_test[factors]))**0.5
print("Train RMSE: {}".format(train_rmse))
print("Test RMSE: {}".format(test_rmse))

Train RMSE: 91428.75090177791
Test RMSE: 102976.87624708698
```

Communicate the Results

- We have developed a linear regression model to predict the resale price of a HDB flat based on factors including floor_area_sqm, remaining_lease, and flat_type.
- The training root mean squared error of the model is \$90k and the testing root mean squared error is \$100k
- The predicted flat price increases by \$2.5k per sqm and \$318 per year of remaining lease

Environment Setup

Getting Help

Google and Stack Overflow are your friend

sort pandas dataframe column value

All Videos Images News More Settings Tools

About 99,700 results (0.56 seconds)

pandas.DataFrame.sort_values — pandas 0.20.1 documentation
https://pandas.pydata.org/pandas-docs/stable/.../pandas.DataFrame.sort_values.html
Sort by the values along either axis ... Specify list for multiple sort orders. ... For DataFrames, this option is only applied when sorting on a single column or label.

pandas.DataFrame.sort — pandas 0.18.1 documentation
<https://pandas.pydata.org/pandas-docs/version/0.18.1/.../pandas.DataFrame.sort.html>
DataFrame.sort(columns=None, axis=0, ascending=True, inplace=False, ... Sort DataFrame either by labels (along either axis) or by the values in column(s) ...

How to sort pandas data frame using values from several columns?
<https://stackoverflow.com/.../how-to-sort-pandas-data-frame-using-values-from-sever...>
Jul 12, 2013 - I have the following data frame: df = pandas.DataFrame([{'c1':3 ... Your code works for me. >>> import pandas >>> df = pandas.DataFrame([{'c1':3 ...

sorting - python, sort descending dataframe with pandas - Stack ...
<https://stackoverflow.com/questions/.../python-sort-descending-dataframe-with-pandas>
Jul 28, 2014 - [False], being a nonempty list, is not the same as False . You should write: ... Sort pandas DataFrame with function over column values.

python - Sort Pandas DataFrame by value - Stack Overflow
<https://stackoverflow.com/questions/37287938/sort-pandas-dataframe-by-value>
May 17, 2016 - If I'm understanding you correctly, you're trying to sort that df by 'retweets'? use: ... I Know this question has a lot of answers, for example: How to sort pandas data frame using values from several columns? I tried the solutions ...

stack overflow Questions Jobs Documentation Tags Users Search... Log In Sign Up

python, sort descending dataframe with pandas Ask Question

I'm trying to sort a dataframe by descending. I put 'False' in the ascending argument, but my order is still ascending.
My code is:

```
from pandas import DataFrame
import pandas as pd

d = {'one':[2,3,1,4,5],
     'two':[5,4,3,2,1],
     'letter':['a','a','b','b','c']}
df = DataFrame(d)

test = df.sort(['one'], ascending=[False])
```

but the output is

	letter	one	two
2	b	1	3
0	a	2	5
1	a	3	4
3	b	4	2
4	c	5	1

python sorting pandas

share improve this question asked Jul 28 '14 at 5:25 user3636476 364 ● 1 ● 3 ● 14

1 Your code actually gives the desired results on pandas version 0.14.1, so you may want to upgrade if possible. – Marius Jul 28 '14 at 6:12

Jobs near you

Expert node.js Developer for Singapore Zuhlike Engineering Ltd ♀ Singapore javascript node.js

Work on SaaS products sold to the world's Fortune 500 Lucep Pte Ltd ♀ Singapore \$36K - \$60K javascript html5

Full-Stack Engineer (Web) Grab ♀ Singapore RELOCATION javascript reactjs

Software Engineer, Backend (Incentives) Grab ♀ Singapore RELOCATION python javascript

More jobs near Singapore...

Getting Help

Check the documentation

The screenshot shows a web browser displaying the pandas 0.20.1 documentation at https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.sort_values.html. The page title is "pandas 0.20.1 documentation » API Reference » pandas.DataFrame »". The main content area is titled "pandas.DataFrame.sort_values". It includes the function signature `DataFrame.sort_values(by, axis=0, ascending=True, inplace=False, kind='quicksort', na_position='last')`, a note that it was "Sort by the values along either axis", and a "New in version 0.17.0." message. Below this, there are sections for "Parameters" and "Returns". The "Parameters" section lists `by`, `axis`, `ascending`, `inplace`, `kind`, and `na_position` with their descriptions. The "Returns" section states that the function returns a `sorted_obj : DataFrame`.

Secure https://pandas.pydata.org/pandas-docs/stable/generated/pandas.DataFrame.sort_values.html

pandas 0.20.1 documentation » API Reference » pandas.DataFrame »

Table Of Contents

- What's New
- Installation
- Contributing to pandas
- Package overview
- 10 Minutes to pandas
- Tutorials
- Cookbook
- Intro to Data Structures
- Essential Basic Functionality
- Working with Text Data
- Options and Settings
- Indexing and Selecting Data
- MultIndex / Advanced Indexing
- Computational tools
- Working with missing data
- Group By: split-apply-combine
- Merge, join, and concatenate
- Reshaping and Pivot Tables
- Time Series / Date functionality
- Time Deltas
- Categorical Data
- Visualization
- Styling
- IO Tools (Text, CSV, HDF5, ...)
- Remote Data Access
- Enhancing Performance
- Sparse data structures
- Frequently Asked Questions (FAQ)
- rpy2 / R interface
- pandas Ecosystem
- Comparison with R / R libraries

pandas.DataFrame.sort_values

`DataFrame.sort_values(by, axis=0, ascending=True, inplace=False, kind='quicksort', na_position='last')` [source]

Sort by the values along either axis

New in version 0.17.0.

Parameters:

- by** : str or list of str
Name or list of names which refer to the axis items.
- axis** : {0 or 'index', 1 or 'columns'}, default 0
Axis to direct sorting
- ascending** : bool or list of bool, default True
Sort ascending vs. descending. Specify list for multiple sort orders. If this is a list of bools, must match the length of the by.
- inplace** : bool, default False
if True, perform operation in-place
- kind** : {'quicksort', 'mergesort', 'heapsort'}, default 'quicksort'
Choice of sorting algorithm. See also `ndarray.np.sort` for more information. `mergesort` is the only stable algorithm. For DataFrames, this option is only applied when sorting on a single column or label.
- na_position** : {'first', 'last'}, default 'last'
`first` puts NaNs at the beginning, `last` puts NaNs at the end

Returns: sorted_obj : DataFrame

Getting Help

Review the error logs

```
resale_prices = pd.read_cs('data/resale-flat-prices-based-on-registration-date-from-march-2012-onwards.csv')

-----
AttributeError                               Traceback (most recent call last)
<ipython-input-19-de8346ce4883> in <module>()
----> 1 resale_prices = pd.read_cs('data/resale-flat-prices-based-on-registration-date-from-march-2012-onwards.csv')

AttributeError: 'module' object has no attribute 'read_cs'
```

```
resale_price.head()

-----
NameError                               Traceback (most recent call last)
<ipython-input-21-e451aa5364ef> in <module>()
----> 1 resale_price.head()

NameError: name 'resale_price' is not defined
```

Getting Help

- If you can't resolve the problem yourself within half an hour, reach out for assistance via our class Slack channel or on Stack Overflow
- Provide details about the steps you have taken, code you are running and errors to facilitate troubleshooting

Development Environment Setup

- Brief intro of tools
- Environment setup
 - Create a Github account <https://github.com/>
 - Install Github Desktop <https://desktop.github.com/>
 - Install Anaconda and Python 3.6 <https://www.anaconda.com/download/>
- Jupyter Notebook test

Exploring and using Datasets

Getting and Importing Data

- Determine if we have the “right” dataset for our problem
- Questions to ask about the data:
 - What type of data is it? Cross-sectional or longitudinal?
 - How was the data collected?
 - Is there much missing data?
 - Was the data collection instrument validated and reliable?
 - Is the dataset aggregated?
 - Do we need pre-aggregated data?

Understanding Your Data

- You need to understand what you are working with
- To better understand your data
 - Create or review the data dictionary
 - Perform exploratory surface analysis
 - Describe data structure and information being collected
 - Explore variables and data types

Data Dictionaries and Documentation

- Data dictionaries help judge the quality of the data
- They also help understand how it's coded
 - Does gender = 1 mean female or male?
 - Is the currency dollars or euros?
- Data dictionaries help identify any requirements, assumptions, and constraints of the data
- They make it easier to share data

Data Dictionary Examples

VARIABLE DESCRIPTIONS:

survival	Survival (0 = No; 1 = Yes)
pclass	Passenger Class (1 = 1st; 2 = 2nd; 3 = 3rd)
name	Name
sex	Sex
age	Age

sibsp	Number of Siblings/Spouses Aboard
parch	Number of Parents/Children Aboard
ticket	Ticket Number
fare	Passenger Fare
cabin	Cabin

embarked	Port of Embarkation (C = Cherbourg; Q = Queenstown; S = Southampton)
----------	---

SPECIAL NOTES:
Pclass is a proxy for socio-economic status (SES)
1st ~ Upper; 2nd ~ Middle; 3rd ~ Lower

Age is in Years; Fractional if Age less than One (1)
If the Age is Estimated, it is in the form xx.5

With respect to the family relation variables (i.e. sibsp and parch)
some relations were ignored. The following are the definitions used
for sibsp and parch.

Sibling: Brother, Sister, Stepbrother, or Stepsister of Passenger Aboard
Titanic
Spouse: Husband or Wife of Passenger Aboard Titanic (Mistresses and Fiances
Ignored)
Parent: Mother or Father of Passenger Aboard Titanic
Child: Son, Daughter, Stepson, or Stepdaughter of Passenger Aboard Titanic

Other family relatives excluded from this study include cousins,
nephews/nieces, aunts/uncles, and in-laws. Some children travelled
only with a nanny, therefore parch=0 for them. As well, some
travelled with very close friends or neighbors in a village, however,
the definitions do not support such relations.

Metadata for Government Procurement

Identifier: '085dd6c3-387b-4661-ab26-02e422ad1286'
Name: 'government-procurement'
Title: 'Government Procurement'
Description: 'This dataset lists all open tenders put out by government agencies since 2015.'
Topics:

- 'Finance'

Keywords:

- 'GEBIZ'
- 'procurement'

Publisher:

- Name: 'Ministry of Finance'

Admin 1:

- Name: 'Charles Tan'
- Department: 'Performance & Resource Management'
- Email: 'Charles_TAN@mof.gov.sg'

Sources:

- 'Ministry of Finance'

License: '<https://data.gov.sg/open-data-licence>'
Frequency: 'Monthly'
Coverage: '2015-01-02 to 2017-01-31'
Last Updated: '2017-02-16T09:09:51.650638'
Resources:

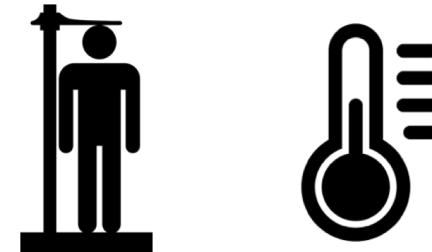
- Identifier: 'b9d8d509-5cb6-45dc-bb46-9508e670e3c2'
Title: 'Government Procurement via GeBIZ'
Url: '<https://storage.data.gov.sg/government-procurement/resources/government-procurement-via-gebiz-2016-11-22T11-45-37Z.csv>'
Format: 'CSV'
Coverage: '2015-01-02 to 2017-01-31'
Last Updated: '2016-11-22T11:45:37.105036'

Url: '<https://storage.data.gov.sg/government-procurement/resources/government-procurement-via-gebiz-2016-11-22T11-45-37Z.csv>'
Format: 'CSV'
Coverage: '2015-01-02 to 2017-01-31'
Last Updated: '2016-11-22T11:45:37.105036'
Schema:

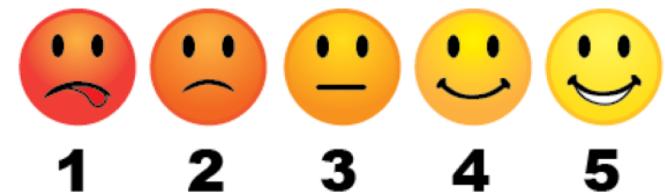
- Name: 'tender_no.'
Title: 'Tender No.'Type: 'text'
Sub Type: 'general'
- Name: 'agency'
Title: 'Agency'
Type: 'text'
Sub Type: 'general'
- Name: 'tender_description'
Title: 'Tender Description'
Type: 'text'
Sub Type: 'general'
- Name: 'award_date'
Title: 'Award Date'
Type: 'datetime'
Sub Type: 'date'
Format: 'YYYY-MM-DD'
- Name: 'tender_detail_status'
Title: 'Tender Detail Status'
Type: 'text'
Sub Type: 'general'

Variable Types

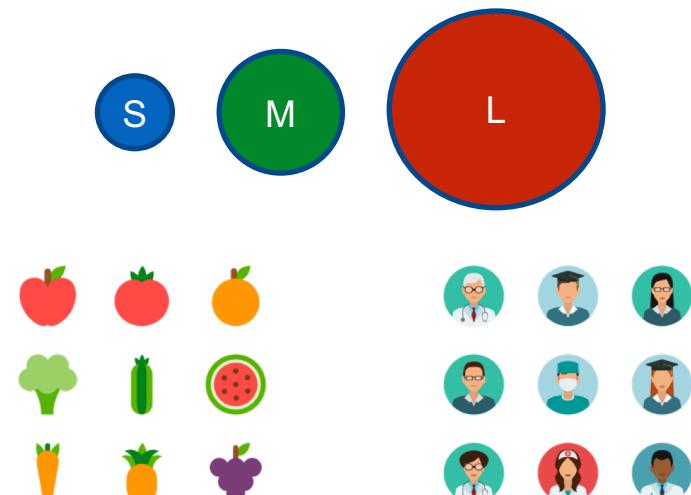
Numeric / Quantitative – values measured on a numerical scale e.g. height, income, temperature



Ordinal – values can be compared and ordered e.g. size (small, medium, large), attitudes (disagree, neutral, agree)



Categorical / Nominal – labels are not ordered e.g. nationality, occupation, movie genres



Class / Dummy Variables

- Let's say we have the categorical variable area, which takes on one of the following values: **rural**, **suburban**, and **urban**.
- We need to represent these numerically for a model. So how do we code them?
- How about **0=rural**, **1=suburban**, and **2=urban**?

Class / Dummy Variables

- But this implies an ordered relationship - is urban twice suburban? That doesn't make sense.
- However, we can represent this information by converting the one area variable into two new variables:

area_urban and **area_suburban**.

Class / Dummy Variables

- We'll draw out how categorical variables can be represented without implying order.
- First, let's choose a reference category. This will be our "base" category.
- It's often good to choose the category with the largest sample size and a criteria that will help model interpretation. If we are testing for a disease, the reference category would be people without the disease.

Class / Dummy Variables

- Step 1: Select a reference category. We'll choose rural as our reference category.
- Step 2: Convert the values rural, suburban, and urban into a numeric representation that does not imply order.
- Step 3: Create two new variables: **area_urban** and **area_suburban**.

Class / Dummy Variables

- Why do we need only two dummy variables?

rural	urban	suburban
-------	-------	----------

- We can derive all of the possible values from these two. If an area isn't urban or suburban, we know it must be rural.
- In general, if you have a categorical feature with k categories, you need to create k-1 dummy variable to represent all of the information.

Class / Dummy Variables

- Let's see our dummy variables.

	area_urban	area_suburban
rural	0	0
suburban	0	1
urban	1	0

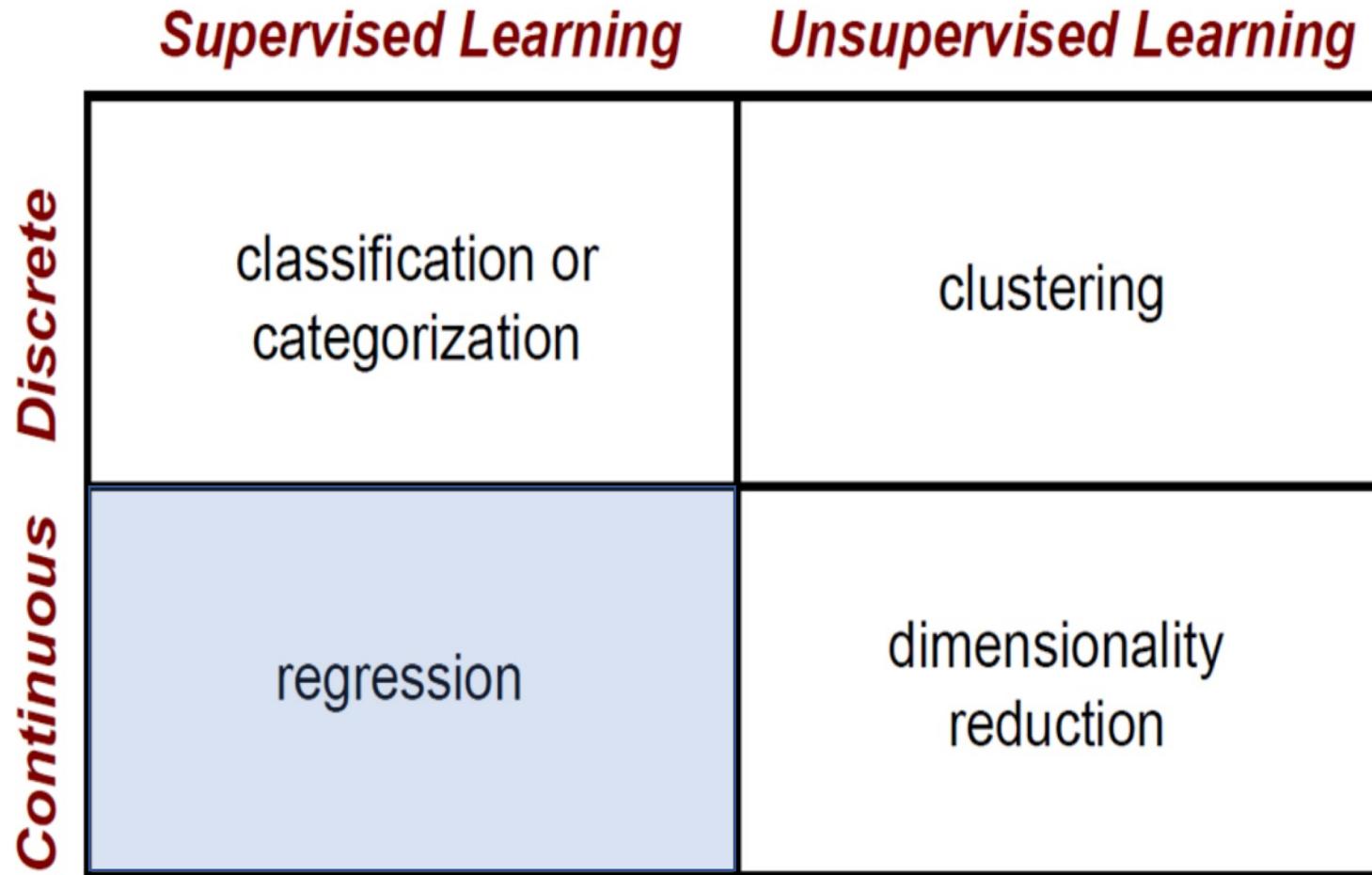
- As mentioned before, if we know `area_urban=0` and `area_suburban=0`, then the area must be rural.

Numpy and Pandas Introduction

- What are Numpy and Pandas? Python packages
- Pandas is built on Numpy
- Numpy uses arrays (lists) to do basic math and slice and index data
- Pandas uses a data structure called a Dataframe
- Dataframes are similar to Excel tables; they contain rows and columns
- With these packages, you can select pieces of data, do basic operations, calculate summary statistics

Machine Learning Algorithms

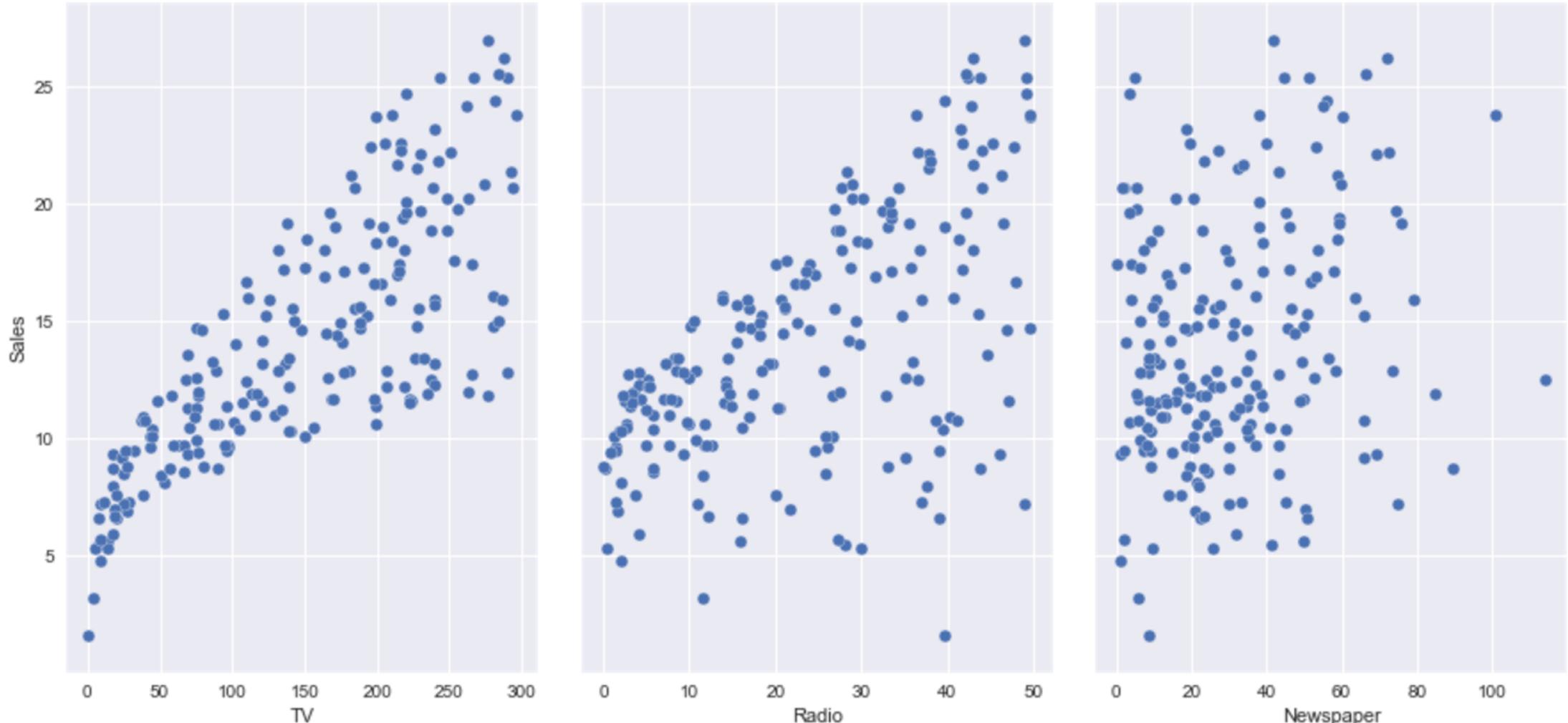
Machine Learning Categories



Linear Regression

- Linear regression is a simple approach to supervised learning. It assumes that the dependence of Y on X_1, X_2, \dots, X_p is linear
- True regression functions however are never linear
- Despite these limitations, linear regression is still very useful conceptually and practically
- “Essentially, all models are wrong, but some are useful” – George Box
- “The only way to find out what will happen when a complex system is disturbed is to disturb the system, not merely to observe it passively” – Fred Mosteller and John Tukey

Linear Regression



Advertising dataset: Each observation represents one market

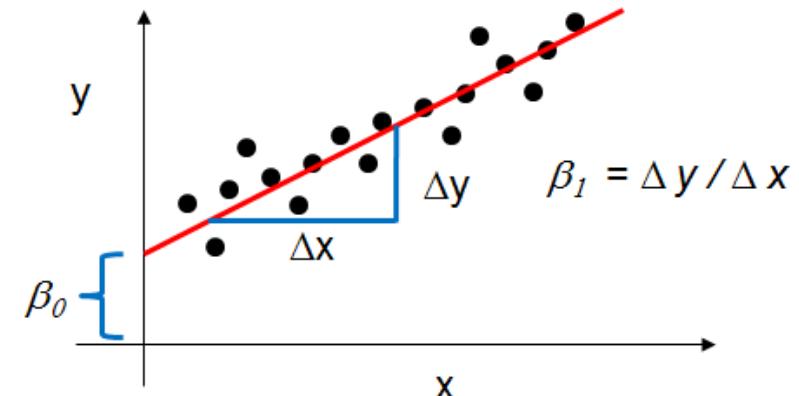
Linear Regression

Questions we might ask about the data:

- Is there a relationship between advertising and sales?
- How strong is the relationship?
- Which specific advertising types contribute to sales?
- What is the effect of each advertising type on sales?
- Given advertising spending in a particular market, can sales be predicted?

Linear Regression

- A simple linear model assumes the relationship: $Y = \beta_0 + \beta_1 X + \varepsilon$
- β_0 and β_1 are two unknown constants that represent the intercept and slope or also referred to as coefficients or parameters
- ε represents the error term
- Given estimates $\hat{\beta}_0$ and $\hat{\beta}_1$ for the model coefficients, we can predict future values using: $\hat{y} = \hat{\beta}_0 + \hat{\beta}_1 x + \varepsilon$

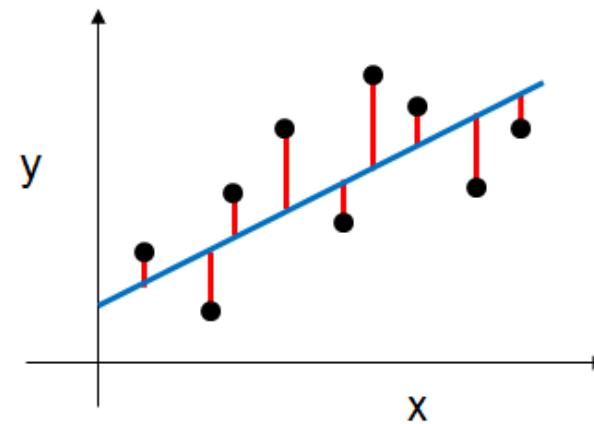


Linear Regression

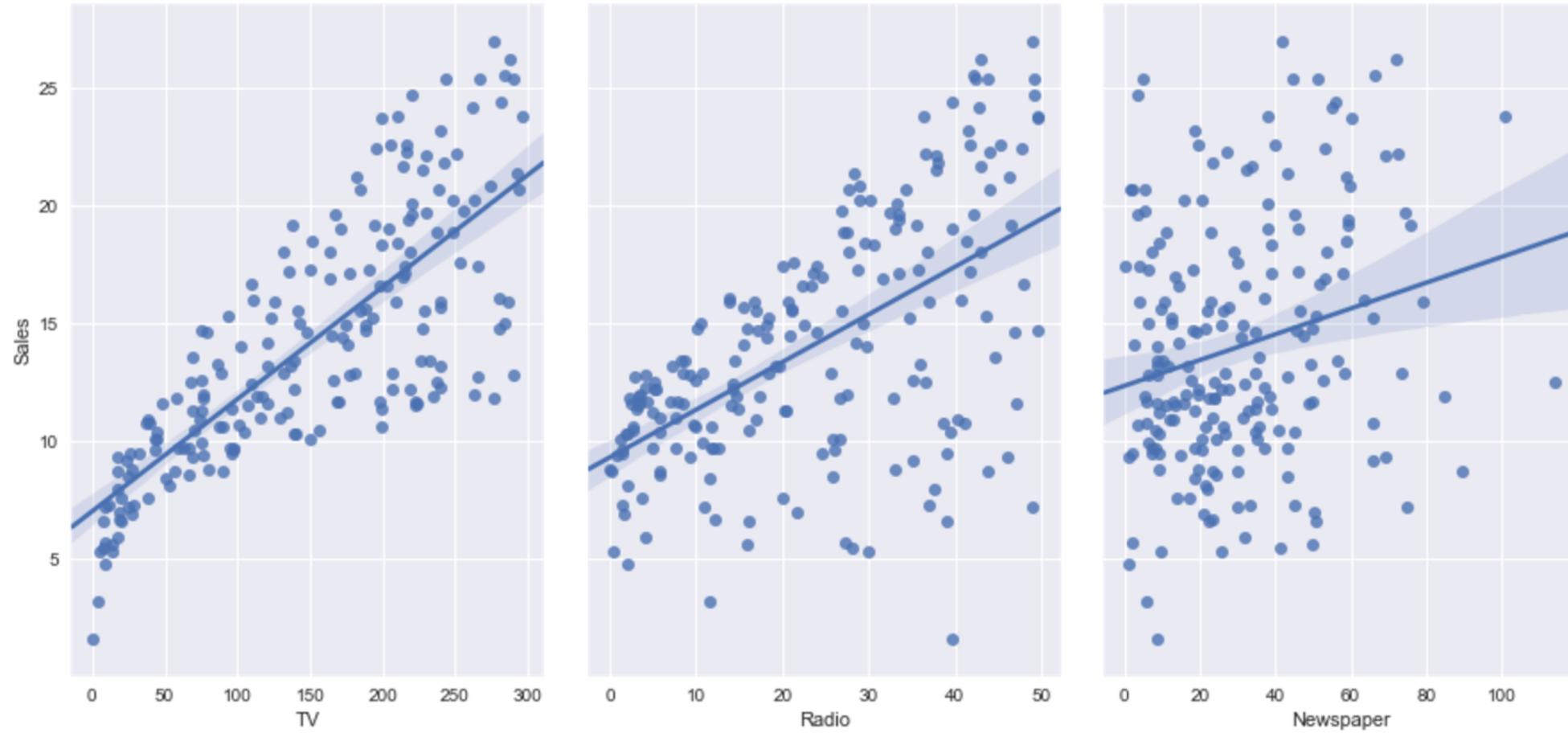
- Let $\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i + \varepsilon$ be the prediction for Y based on the *i*th value of X. Then $e_i = y_i - \hat{y}_i$ represents the *i*th residual
- We define the residual sum of squares (RSS) as: $RSS = e_1^2 + e_2^2 + \dots e_n^2$
- The least squares approach chooses $\hat{\beta}_0$ and $\hat{\beta}_1$ to minimize the RSS
- The minimizing values can be shown to be:

$$\hat{\beta}_1 = \frac{\sum_{i=1}^n (x_i - \bar{x})(y_i - \bar{y})}{\sum_{i=1}^n (x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$



Linear Regression Single Predictor



Least squares regression fits for Sales against TV, Radio and Newspaper

Multiple Linear Regression

- A multiple linear model assumes the relationship:

$$Y = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_p X_p + \varepsilon$$

- We interpret β_j as the average effect on Y due to a one unit increase in X_j while holding all other predictors constant
- In the advertising model, this equation becomes:

$$sales = \beta_0 + \beta_1 * TV + \beta_2 * Radio + \beta_3 * Newspaper + \varepsilon$$

- Similarly, we estimate $\beta_0, \beta_1, \dots, \beta_p$ as values that minimize the sum of square residuals

Multiple Linear Regression

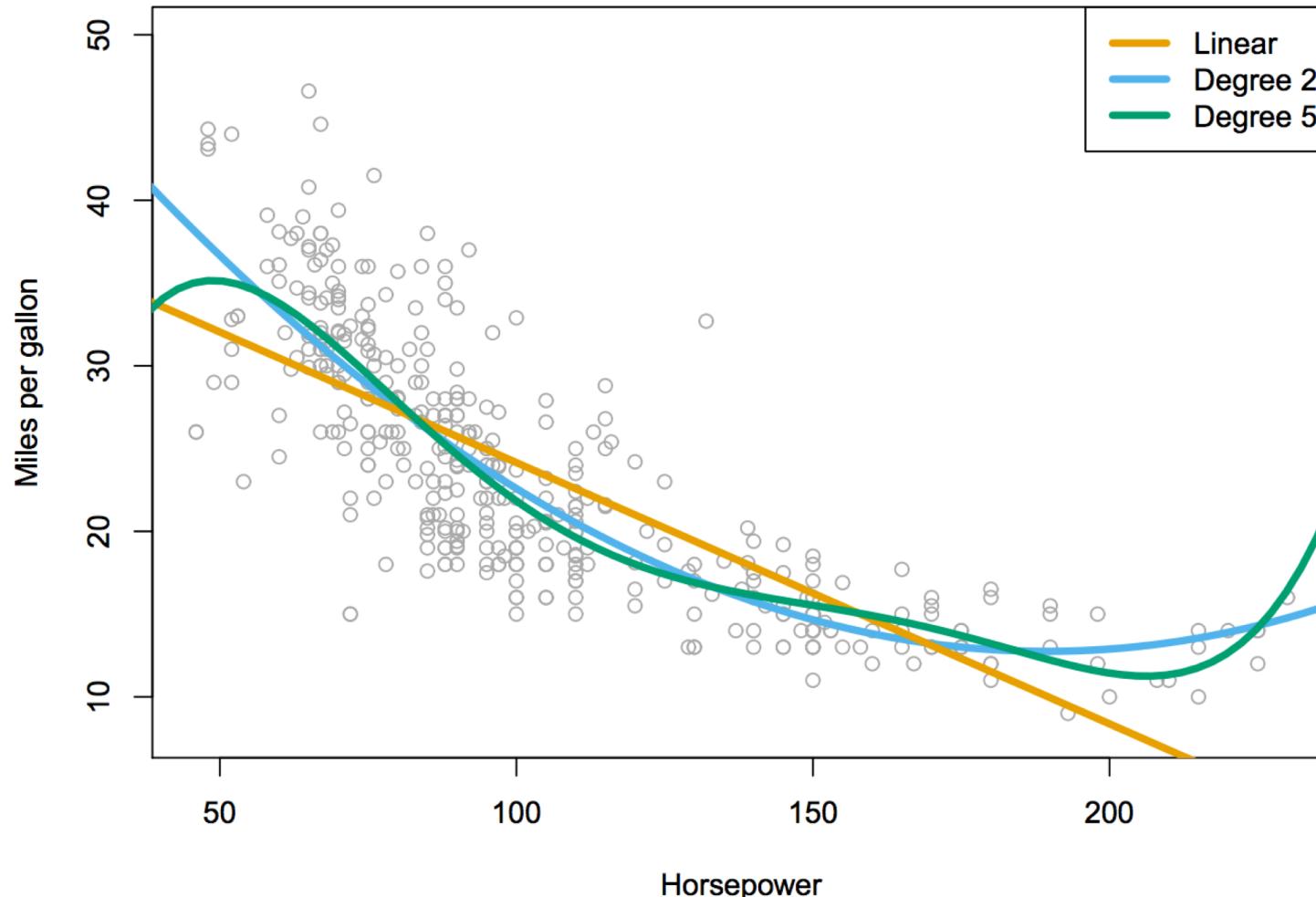
- Simple linear regression with one variable can explain some variance, but using multiple variables can be much more powerful.
- We want our multiple variables to be mostly independent to avoid multicollinearity.
- Multicollinearity, when two or more variables in a regression are highly correlated, can cause problems with the model.

Model Extensions: Interactions

- Suppose that spending money on radio advertising actually increases the effectiveness of TV advertising, thus the slope term for TV should increase as Radio increases
- In this scenario, given a fixed budget, spending half on Radio and half on TV may increase sales more than allocating the entire amount to either
- In business, this is known as the synergy effect, and in statistics it is referred to as an interaction effect
- Model takes the form:

$$sales = \beta_0 + \beta_1 * TV + \beta_2 * Radio + \beta_3 * (Radio * TV) + \varepsilon$$

Model Extensions: Polynomials



The figure suggests that including polynomial terms may provide a better fit

Linear Regression Analysis in Sklearn

- Sklearn defines models as objects (in the OOP sense).

You can use the following principles:

- All sklearn modeling classes are based on the base estimator. This means all models take a similar form.
- All estimators take a matrix X , either sparse or dense.
- Supervised estimators also take a vector y (the response).
- Estimators can be customized through setting the appropriate parameters.

Classes and Objects

- **Classes** are an abstraction for a complex set of ideas, e.g. human.
- Specific **instances** of classes can be created as **objects**.
 - `john_smith = human()`
- Objects have **properties**. These are attributes or other information.
 - `john_smith.age`
 - `john_smith.gender`
- Objects have **methods**. These are procedures associated with a class/object.
 - `john_smith.breathe()`
 - `john_smith.walk()`

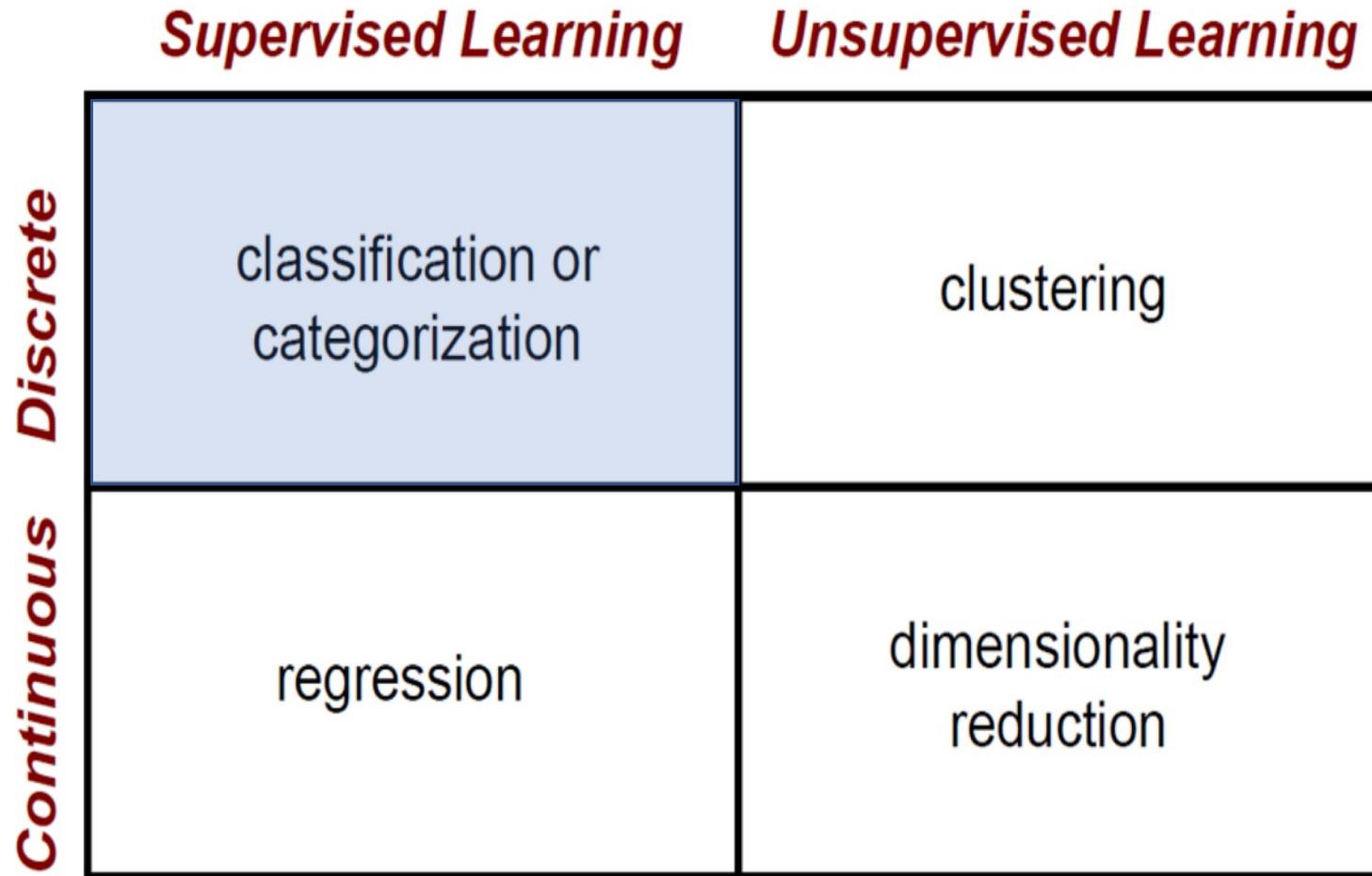
Linear Regression Analysis in Sklearn

- General format for sklearn model classes and methods

```
# generate an instance of an estimator class
estimator = base_models.AnySKLearnObject()
# fit your data
estimator.fit(X, y)
# score it with the default scoring method (recommended to use the metrics module in the future)
estimator.score(X, y)
# predict a new set of data
estimator.predict(new_X)
# transform a new X if changes were made to the original X while fitting
estimator.transform(new_X)
```

- LinearRegression() doesn't have a transform function
- With this information, we can build a simple process for linear regression.

Machine Learning Categories

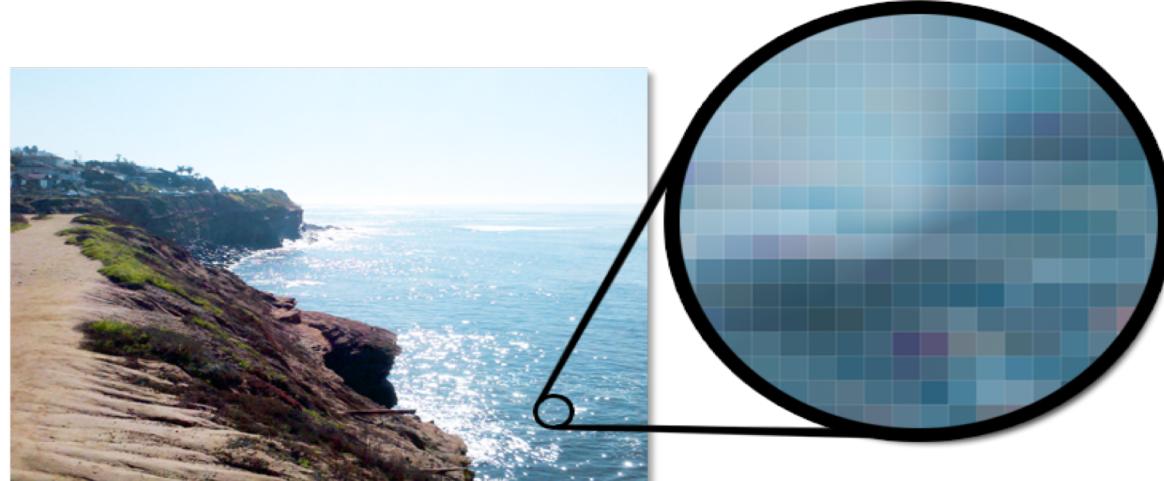


What is Classification?

- Classification is a machine learning problem for solving a set value given the knowledge we have about that value.
- Many classification problems are trying to predict binary values.
- For example, we may be using patient data (medical history) to predict whether the patient is a smoker or not.

What is Classification?

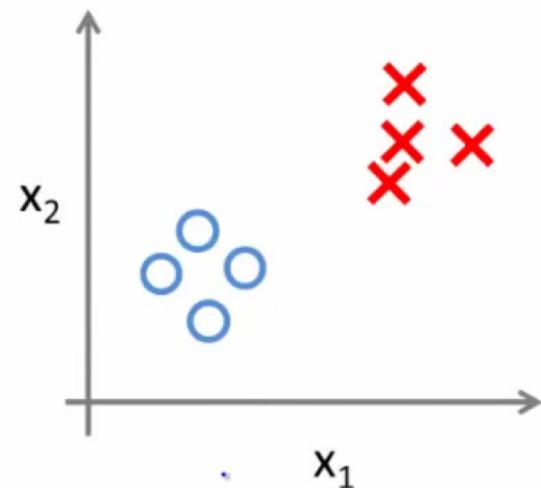
- Some problems don't appear to be binary at first glance. However, you can boil down the response to a boolean (true/false) value.
- What if you are predicting whether an image pixel will be red or blue?
- We don't need to predict that a pixel is blue, just that it is not red.
- This is similar to the concept of dummy variables.



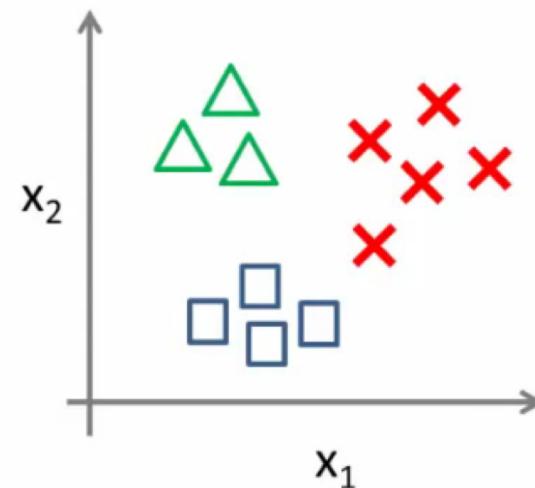
What is Classification?

- Binary classification is the simplest form of classification.
- However, classification problems can have multiple class labels.
- Instead of predicting whether the pixel is red or blue, you could predict whether the pixel is red, blue, or green.

Binary classification:



Multi-class classification:



What is a Class Label?

- A **class label** is a representation of what we are trying to predict: our target.
- Examples of class labels from before are:

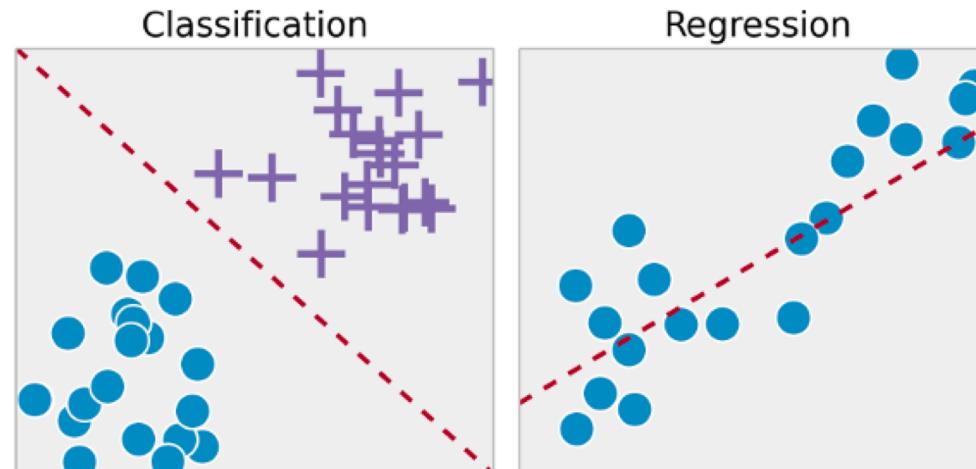
Data Problem	Class Labels
Patient data problem	is smoker, is not smoker
pixel color	red, blue, green

Determining Regression or Classification

- One of the easiest ways to determine if a problem is regression or classification is to determine if our target variable can be ordered mathematically.
- For example, if predicting company revenue, \$100MM is greater than \$90MM. This is a *regression* problem because the target can be ordered.
- However, if predicting pixel color, red is not inherently greater than blue. Therefore, this is a *classification* problem.

Logistic Regression

- Logistic regression is a linear approach to solving a classification problem
- That is, we can use a linear model, similar to Linear regression, in order to solve if an item belongs or does not belong to a class label
- Regression results can have a value range from $-\infty$ to ∞
- Classification is used when predicted values (i.e. class labels) are not greater than or less than each other



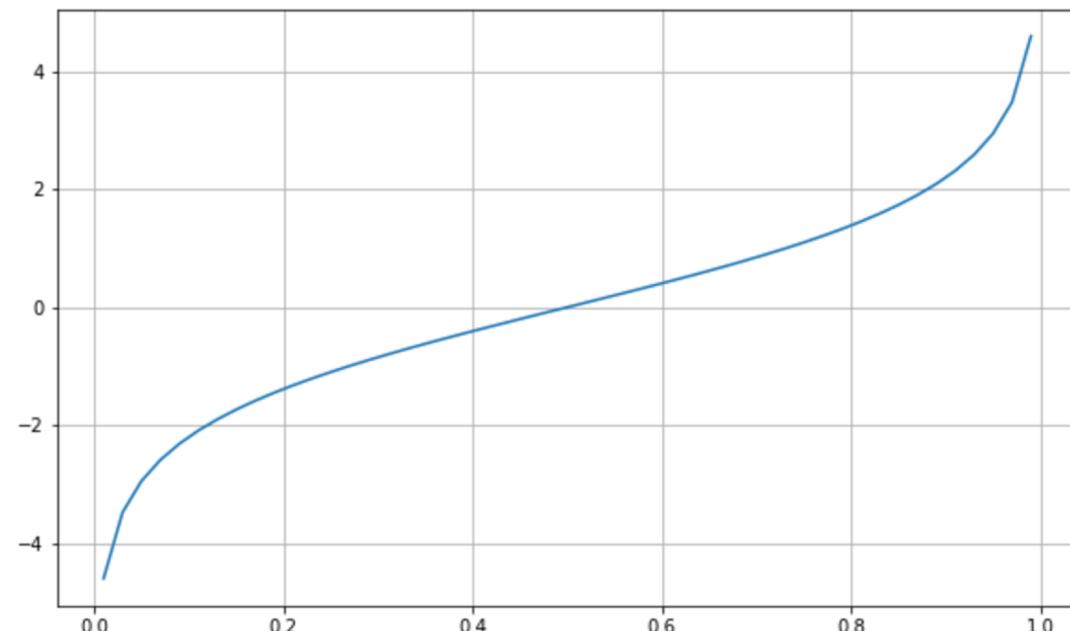
Logistic Regression

- The problem with using the linear regression equation to represent classification probabilities is predictions can be > 1 and < 0
- To avoid this problem, we use a function that gives outputs between 0 and 1 for all values of X

$$\text{Log Odds: } \ln\left(\frac{p(X)}{1-p(X)}\right) = \beta_0 + \beta_1 X$$

$$\text{Odds: } \frac{p(X)}{1-p(X)} = e^{\beta_0 + \beta_1 X}$$

$$\text{Probability: } p(X) = \frac{e^{\beta_0 + \beta_1 X}}{1+e^{\beta_0 + \beta_1 X}}$$



Odds and Log-Odds

- The value within the natural log, $\frac{p}{1-p}$ represents the *odds*. Taking the natural log of odds generates *log odds*.
- Odds for an event – 5:2 reflects the event happening 5 times and not happening 2 times i.e. probability = 5/7
- The odds multiply by e^{β_1} for every 1-unit increase in x.

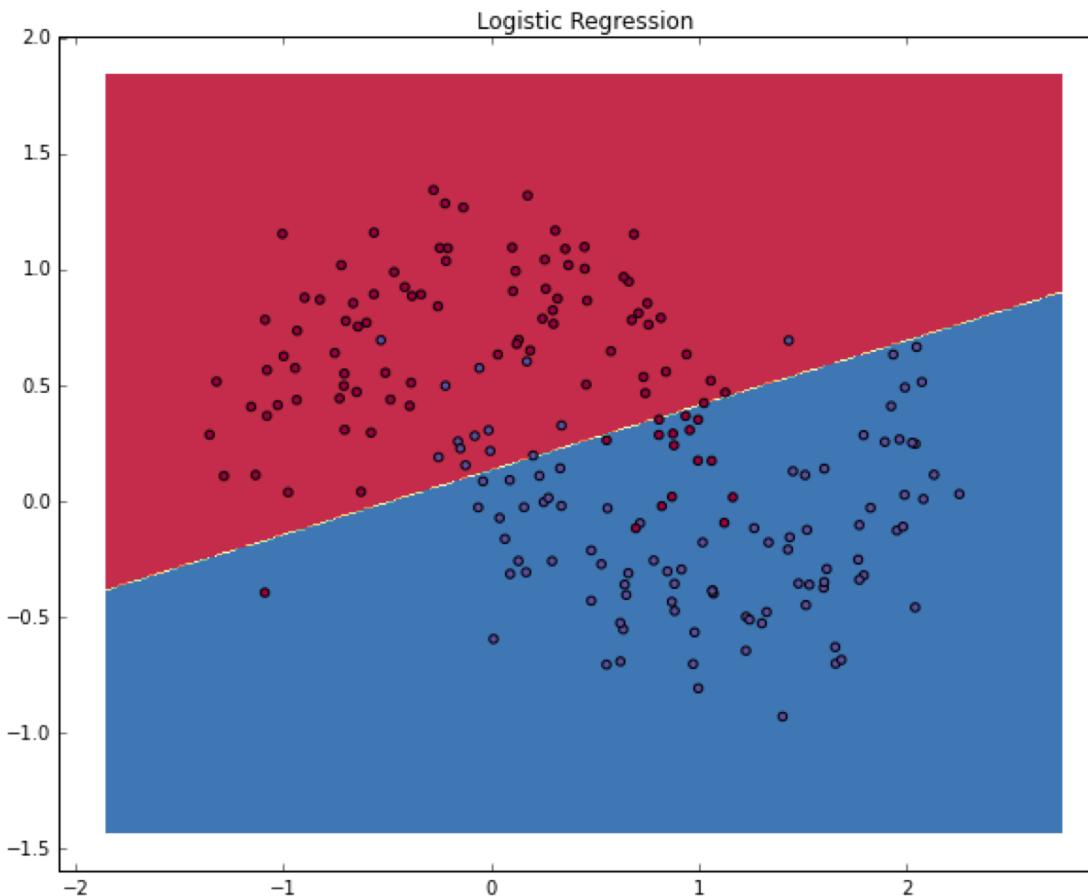
Odds and Log-Odds

- The coefficients are unknown and must be estimated based on the available training data
- Although we could use (non-linear) least squares to fit the model, the more general method of maximum likelihood is preferred since it has better statistical properties
- The basic intuition is that we seek estimates for the coefficients such that the predicted probability corresponds as closely to the actual observed value:

$$l(\beta_0, \beta_1) = \prod_{i:y_i=1} p(x_i) \prod_{i:y_i=0} (1 - p(x_i))$$

Odds and Log-Odds

With these coefficients, we get our overall probability: the logistic regression draws a linear decision line which divides the classes.



Classification Metrics

- Metrics for regression do **not** apply to classification.
- We could measure the distance between the probability of a given class and an item being in that class. Guessing 0.6 for a 1 is a 0.4 error.
- But this overcomplicates our goal: understanding binary classification, whether something is black or white, right or wrong.
- To do this, we'll measure “correctness” or “incorrectness”.

Classification Metrics

- We'll use two primary metrics: *accuracy* and *misclassification rate*.
- **Accuracy** is the number of *correct* predictions out of all predictions in the sample. This is a value we want to *maximize*.
- **Misclassification rate** is the number of incorrect predictions out of all predictions in the sample. This is a value we want to *minimize*.
- These two metrics are directly opposite of each other.
- $1 - \text{misclassification rate} = \text{accuracy}$