ST445 Managing and Visualizing Data

Introduction to Data

Week 1 Lecture, MT 2020 - Chengchun Shi

What is Data?

"Data is a set of values of subjects with respect to qualitative or quantitative variables. " -– Wikipedia

---

summarized in the form of

- vector or matrix
- tensor (high-order matrix)
- image, or text

Data vs Information

Data

- raw, unorganized facts that need to be processed
- unusable until it is organized

Information

- created when data is processed, organized, structured
- needs to be situated in an appropriate *context* in order to become useful

Information Theory

The information content of a message depends on its probability:

$$I(x) = -\log_2 p(x)$$

- Two independent events with $p(x, y) = p(x)p(y)$ will have information $I(x, y) = I(x) + I(y)$

  which is the sum of the information of the individual events.

- In transmitting a message modelled as a random variable the average amount of information received is:
$$H[X] = -\sum_x p(x)\log_2 p(x) = \sum_x p(x)I(x)$$

- The quantity H[X] is called *entropy*.
- A measure of information in a single random variable.

Information Theory

- Joint entropy:

$$H(X,Y) = -\sum_{x,y} p(x,y)\log_2 p(x,y)$$

- Conditional entropy:

$$H(X|Y) = -\sum_{x,y} p(x,y)\log_2 p(x|y) = H(X,Y) - H(Y)$$

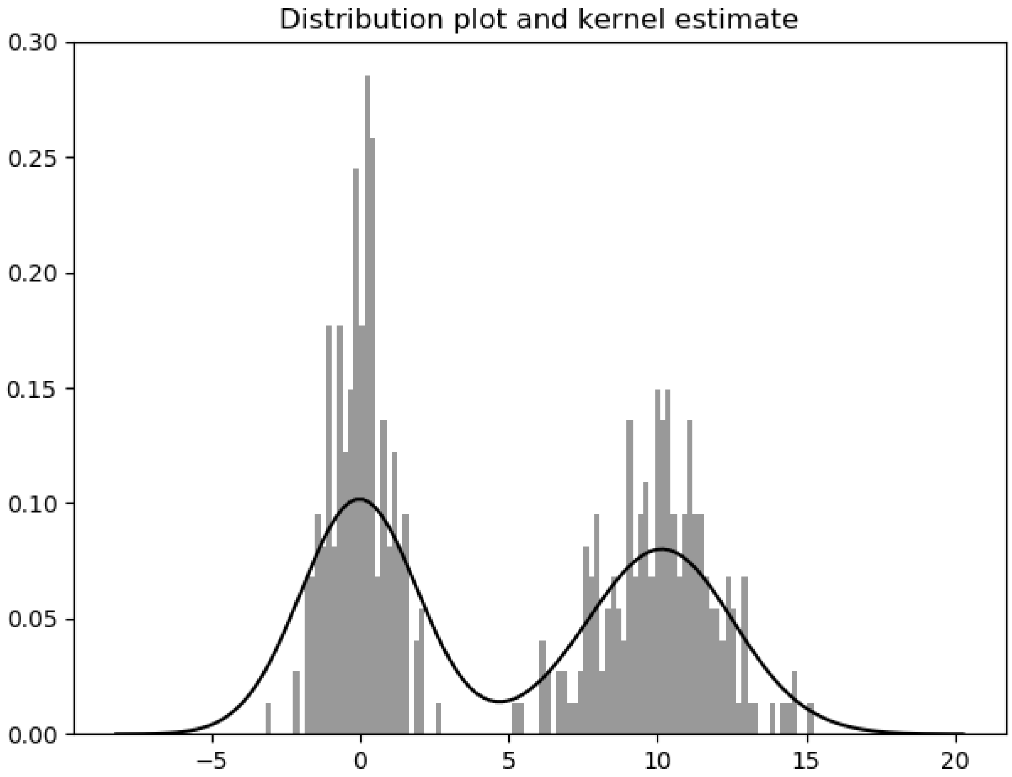
- Mutual information:

$$I(X|Y) = H(X) - H(X|Y) = H(X) + H(Y) - H(X|Y)$$

- Equals zero when X and Y are independent.
- Expect to see more on probabilstic models later in the course!

Visualising Distributions

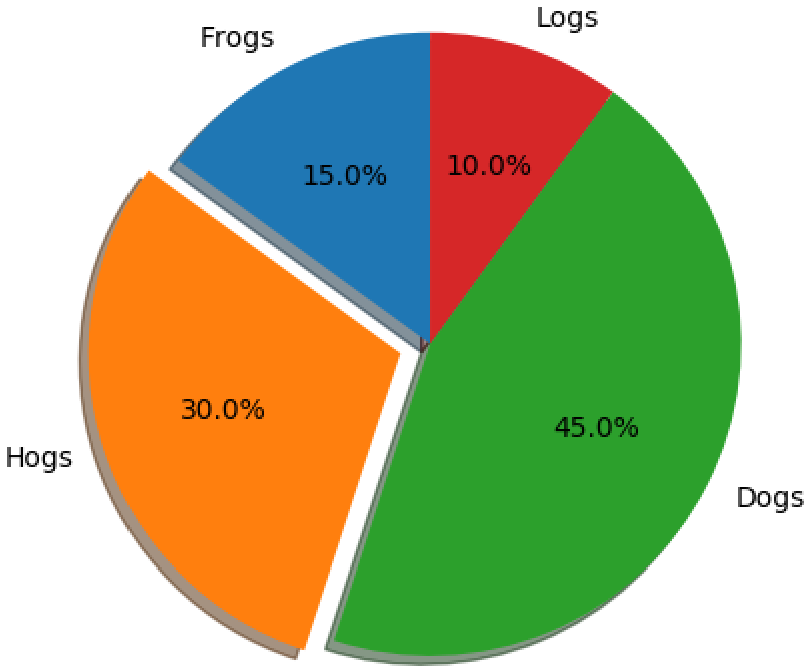
This is not a statistics course but the powerful tools to visualise distributions can helpyou understand your data.



See the code for this plot on Page 281 of "Python for Data Science" by Wes McKinney

Simplest can be best...

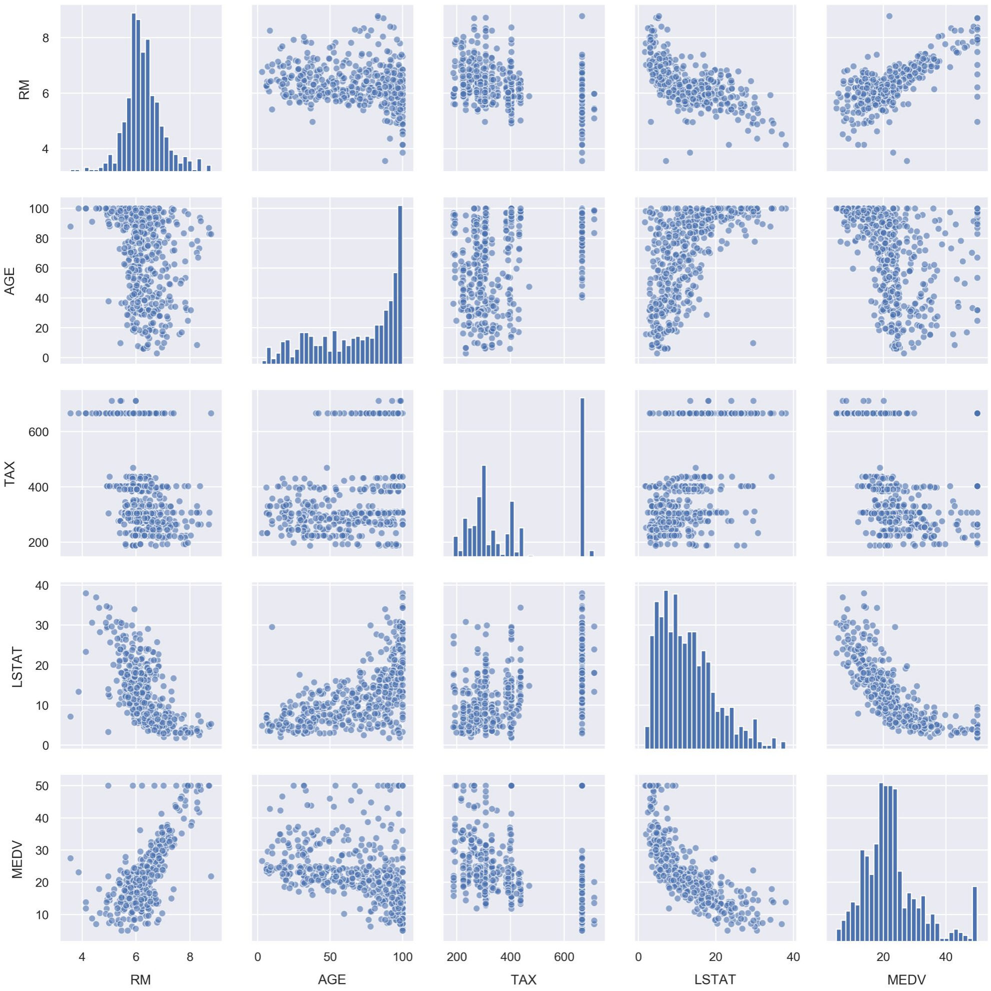
use **matplotlib.pyplot.pie**



Taken from https://matplotlib.org/gallery/index.html (https://matplotlib.org/gallery/index.html)

Exploring Data Visually

Combines scatter plots and histograms. Data is from the Boston Housing dataset available from scikit-learn. Use **pandas.scatter_matrix**



As a data scientist

- Most (approximately 70%) time in data science is spent on cleaning and organsising data

- Collecting data sets can also be time consuming

- Little time is spent refining algorithms

- The tools and techniques you will learn in the following two lectures on NumPy and Pandas are

  well adapted for data cleaning (and many other tasks).

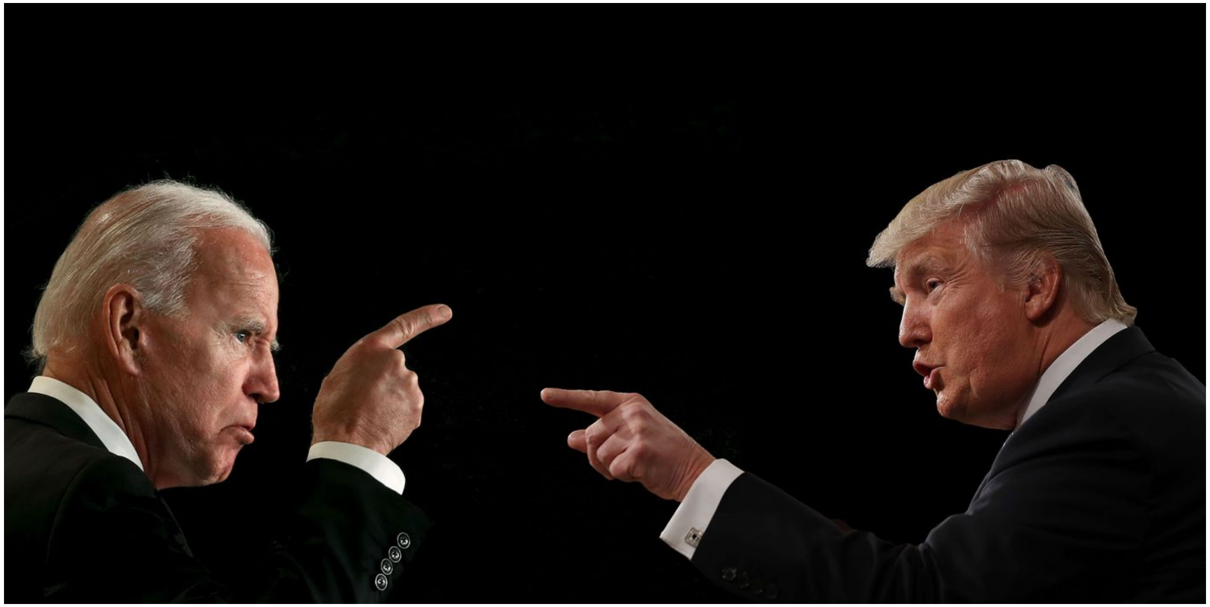
Next slide shows struggle to obtain good data.

Common Data Quality Issues

- Missing data is a common problem. A good solution is to build a simple model to estimate the missing values. Pandas has good tools for this.
    - Sequenced Treatment Alternatives to Relieve Depression (STAR*D) data: 17% obs. are missing.
    - The schizophrenia study: over 50% obs. are missing.
    - The Nefazodone-CBASP clinical trial study: 5% obs. are missing.
- Duplicate data is another common problem. Again Pandas has tools for this.
- Incorrect values in the dataset.
- In engineering methods have been developed for correcting measurements where there are networks of sensors, some of which may have failed. This is often called *Data validation and reconciliation*

Missing data

Taken from https://www.cnbc.com/2016/02/21/is-trump-vs-hillary-inevitable.html (https://www.cnbc.com/2016/02/21/is-trump-vs-hillary-inevitable.html)

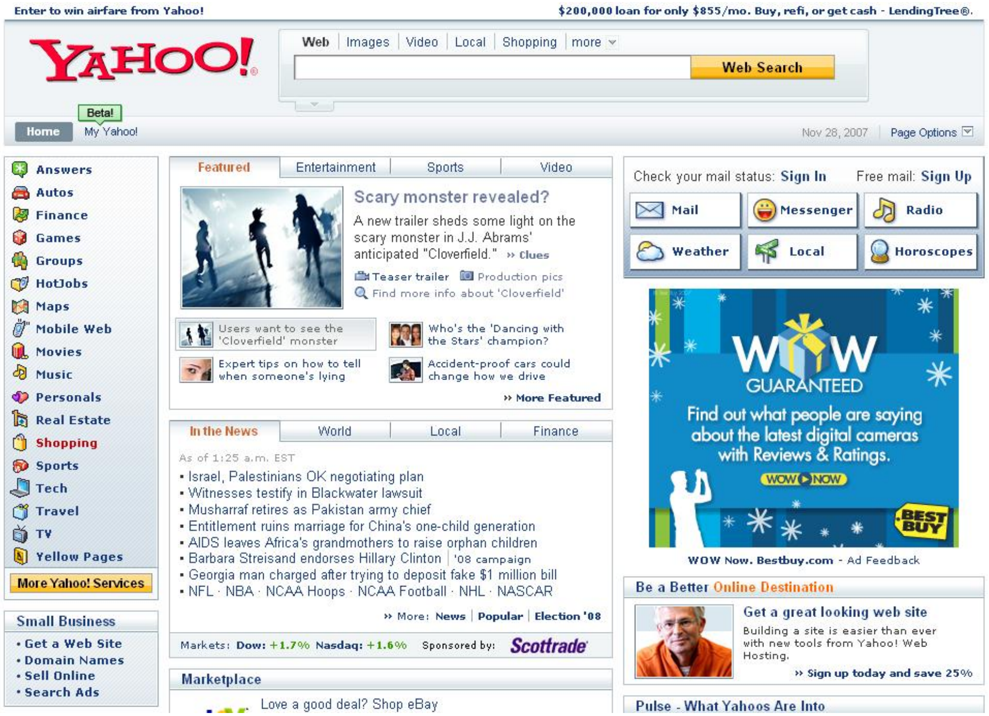


Taken from https://www.bloomberg.com/features/2019-trump-or-biden-quotes-quiz/ (https://www.bloomberg.com/features/2019-trump-or-biden-quotes-quiz/)

Changes in the world of data

- high-dimensional data ($p \gg n$), e.g., genetic data (dimension reduction, penalized regression, random projection)
- functional data, e.g., time series, images (functional principle component analysis, deep learning)
- big data/massive data (subsampling, divide and conquer, parallel computing)
  - volume of data in the modern world: 90% of the world's data generated in the last *two years* (https://www.sciencedaily.com/releases/2013/05/130522085217.htm)
  - an that was in 2013

Examples of big data



Enter to win airfare from Yahoo!

$200,000 loan for only $855/mo. Buy, refi, or get cash - LendingTree®.

Scary monster revealed?

A new trailer sheds some light on the scary monster in J.J. Abrams' anticipated "Cloverfield." » Clues

Teaser trailer · Production pics

Find more info about 'Cloverfield'

Users want to see the 'Cloverfield' monster

Who's the 'Dancing with the Stars' champion?

Expert tips on how to tell when someone's lying

Accident-proof cars could change how we drive

» More Featured

As of 1:25 a.m. EST

- Israel, Palestinians OK negotiating plan
- Witnesses testify in Blackwater lawsuit
- Musharraf retires as Pakistan army chief
- Entitlement ruins marriage for China's one-child generation
- AIDS leaves Africa's grandmothers to raise orphan children
- Barbara Streisand endorses Hillary Clinton | '08 campaign
- Georgia man charged after trying to deposit fake $1 million bill
- NFL · NBA · NCAA Hoops · NCAA Football · NHL · NASCAR

» More: News | Popular | Election '08

Markets: Dow: +1.7% Nasdaq: +1.6% Sponsored by: Scottrade

Love a good deal? Shop eBay

Find out what people are saying about the latest digital cameras with Reviews & Ratings.

WOW Now. Bestbuy.com - Ad Feedback

Be a Better Online Destination

Get a great looking web site

Building a site is easier than ever with new tools from Yahoo! Web Hosting.

» Sign up today and save 25%

- Get a Web Site
- Domain Names
- Sell Online
- Search Ads

- Yahoo! Front Page Today Module User Click Log Dataset, version 1.0 (1.1 GB).
- contains a fraction of user click log for news articles displayed in the Featured Tab of the Today Module on Yahoo! Front Page during the first ten days in May 2009.
- a total of 45,811,883 user visits to the Today Module.

Clever algorithms are very important...

- The Apollo landing relied on algorithmic developments such as the Kalman Filter to process noisy data from multiple sensors.

- Big Data has been powered by algorithms such as Google's PageRank

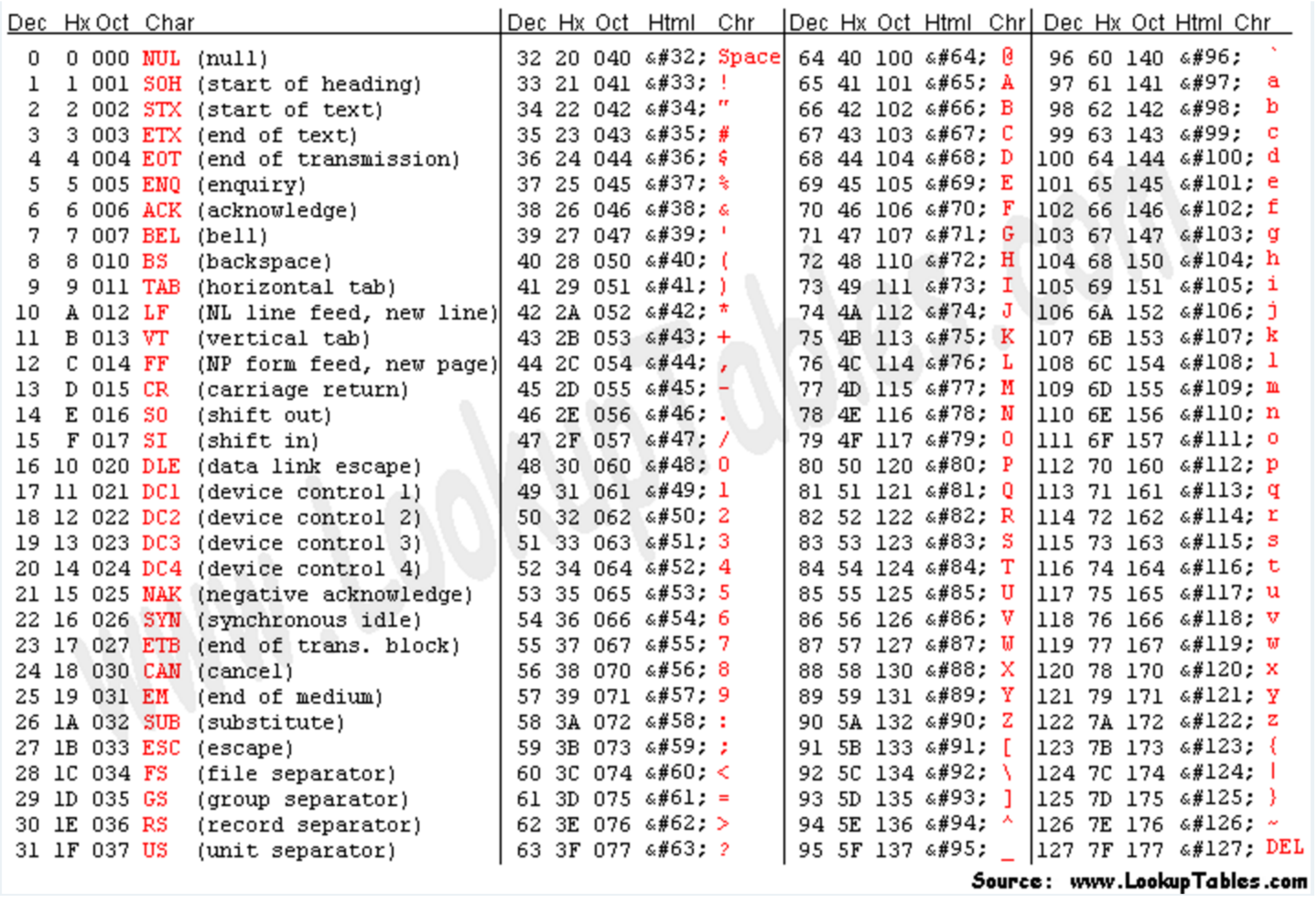
Examples of small data

- Sequenced Treatment Alternatives to Relieve Depression (STAR*D) data: 383 obs.
- The schizophrenia study: 165 obs.
- The Nefazodone-CBASP clinical trial study: 681 obs.
- ACTG 175 study: 2139 obs.
- A Data from the InternationalWarfarin Pharmacogenetics Consortium: 3848 obs.

Basic units of data

- Bits
    - smallest unit of storage, a 0 or 1
    - with n bits, can store 2^n patterns - so one byte can store 256 patterns

- Bytes
  - eight *bits* = one *byte*
  - ASCII (American Standard Code for Information Interchange) - represented characters, such as `A` represented as 65



Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040	&#32;	Space	64	40	100	&#64;	@	96	60	140	&#96;	`
1	1	001	SOH (start of heading)	33	21	041	&#33;	!	65	41	101	&#65;	A	97	61	141	&#97;	a
2	2	002	STX (start of text)	34	22	042	&#34;	"	66	42	102	&#66;	B	98	62	142	&#98;	b
3	3	003	ETX (end of text)	35	23	043	&#35;	#	67	43	103	&#67;	C	99	63	143	&#99;	c
4	4	004	EOT (end of transmission)	36	24	044	&#36;	$	68	44	104	&#68;	D	100	64	144	&#100;	d
5	5	005	ENQ (enquiry)	37	25	045	&#37;	%	69	45	105	&#69;	E	101	65	145	&#101;	e
6	6	006	ACK (acknowledge)	38	26	046	&#38;	&	70	46	106	&#70;	F	102	66	146	&#102;	f
7	7	007	BEL (bell)	39	27	047	&#39;	'	71	47	107	&#71;	G	103	67	147	&#103;	g
8	8	010	BS (backspace)	40	28	050	&#40;	(	72	48	110	&#72;	H	104	68	150	&#104;	h
9	9	011	TAB (horizontal tab)	41	29	051	&#41;	)	73	49	111	&#73;	I	105	69	151	&#105;	i
10	A	012	LF (NL line feed, new line)	42	2A	052	&#42;	*	74	4A	112	&#74;	J	106	6A	152	&#106;	j
11	B	013	VT (vertical tab)	43	2B	053	&#43;	+	75	4B	113	&#75;	K	107	6B	153	&#107;	k
12	C	014	FF (NP form feed, new page)	44	2C	054	&#44;	,	76	4C	114	&#76;	L	108	6C	154	&#108;	l
13	D	015	CR (carriage return)	45	2D	055	&#45;	-	77	4D	115	&#77;	M	109	6D	155	&#109;	m
14	E	016	SO (shift out)	46	2E	056	&#46;	.	78	4E	116	&#78;	N	110	6E	156	&#110;	n
15	F	017	SI (shift in)	47	2F	057	&#47;	/	79	4F	117	&#79;	O	111	6F	157	&#111;	o
16	10	020	DLE (data link escape)	48	30	060	&#48;	0	80	50	120	&#80;	P	112	70	160	&#112;	p
17	11	021	DC1 (device control 1)	49	31	061	&#49;	1	81	51	121	&#81;	Q	113	71	161	&#113;	q
18	12	022	DC2 (device control 2)	50	32	062	&#50;	2	82	52	122	&#82;	R	114	72	162	&#114;	r
19	13	023	DC3 (device control 3)	51	33	063	&#51;	3	83	53	123	&#83;	S	115	73	163	&#115;	s
20	14	024	DC4 (device control 4)	52	34	064	&#52;	4	84	54	124	&#84;	T	116	74	164	&#116;	t
21	15	025	NAK (negative acknowledge)	53	35	065	&#53;	5	85	55	125	&#85;	U	117	75	165	&#117;	u
22	16	026	SYN (synchronous idle)	54	36	066	&#54;	6	86	56	126	&#86;	V	118	76	166	&#118;	v
23	17	027	ETB (end of trans. block)	55	37	067	&#55;	7	87	57	127	&#87;	W	119	77	167	&#119;	w
24	18	030	CAN (cancel)	56	38	070	&#56;	8	88	58	130	&#88;	X	120	78	170	&#120;	x
25	19	031	EM (end of medium)	57	39	071	&#57;	9	89	59	131	&#89;	Y	121	79	171	&#121;	y
26	1A	032	SUB (substitute)	58	3A	072	&#58;	:	90	5A	132	&#90;	Z	122	7A	172	&#122;	z
27	1B	033	ESC (escape)	59	3B	073	&#59;	;	91	5B	133	&#91;	[	123	7B	173	&#123;	{
28	1C	034	FS (file separator)	60	3C	074	&#60;	<	92	5C	134	&#92;	\	124	7C	174	&#124;	|
29	1D	035	GS (group separator)	61	3D	075	&#61;	=	93	5D	135	&#93;	]	125	7D	175	&#125;	}
30	1E	036	RS (record separator)	62	3E	076	&#62;	>	94	5E	136	&#94;	^	126	7E	176	&#126;	~
31	1F	037	US (unit separator)	63	3F	077	&#63;	?	95	5F	137	&#95;	_	127	7F	177	&#127;	DEL

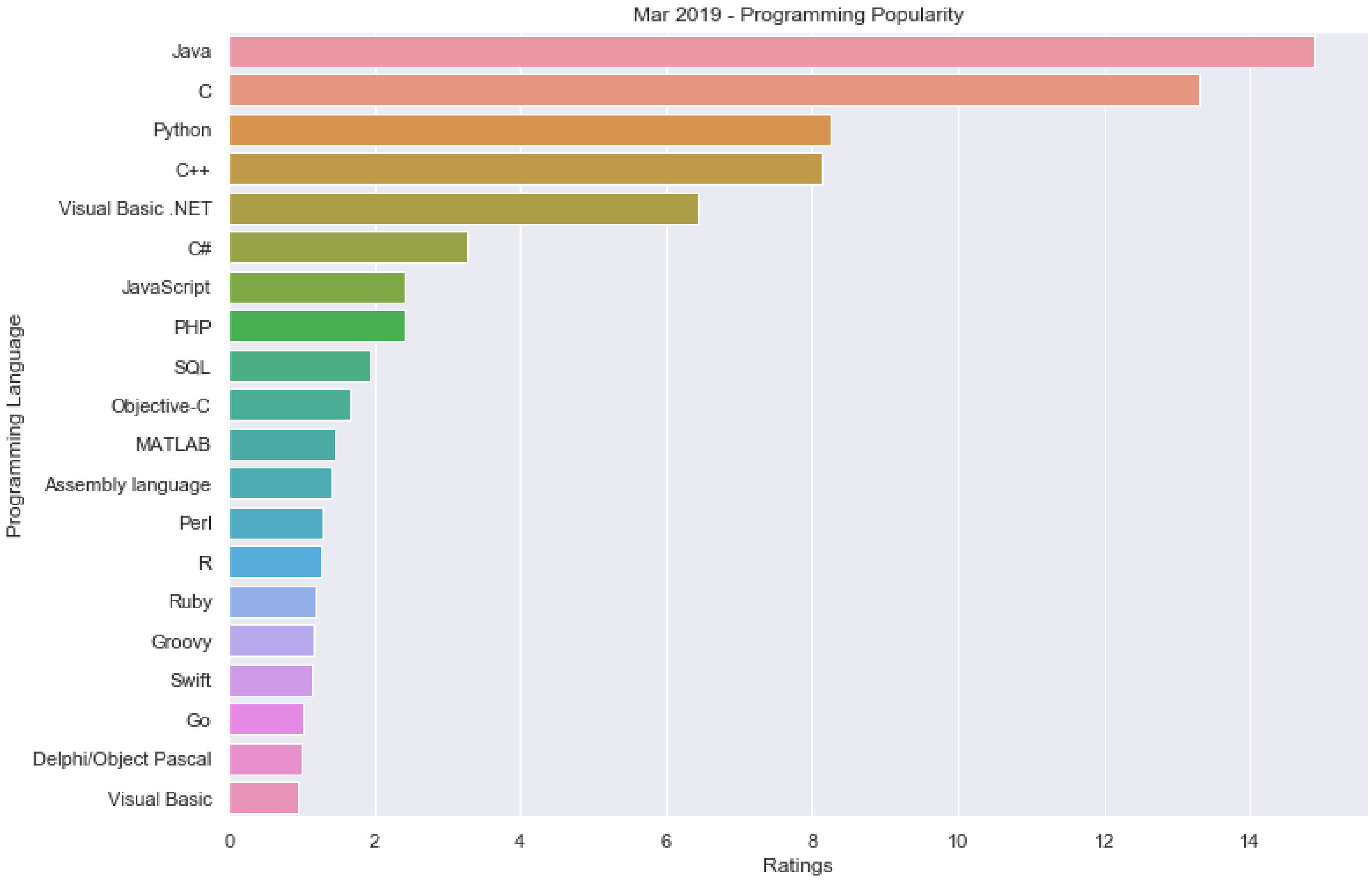
Source: www.LookupTables.com

multi-byte units

unit	abbreviation	total bytes	nearest decimal equivalent
kilobyte	KB	1,024^1	1000^1
megabyte	MB	1,024^2	1000^2
gigabyte	GB	1,024^3	1000^3
terabyte	TB	1,024^4	1000^4
petabyte	PB	1,024^5	1000^5
exabyte	EB	1,024^6	1000^6
zettabyte	ZB	1,024^7	1000^7
yottabyte	YB	1,024^8	1000^8

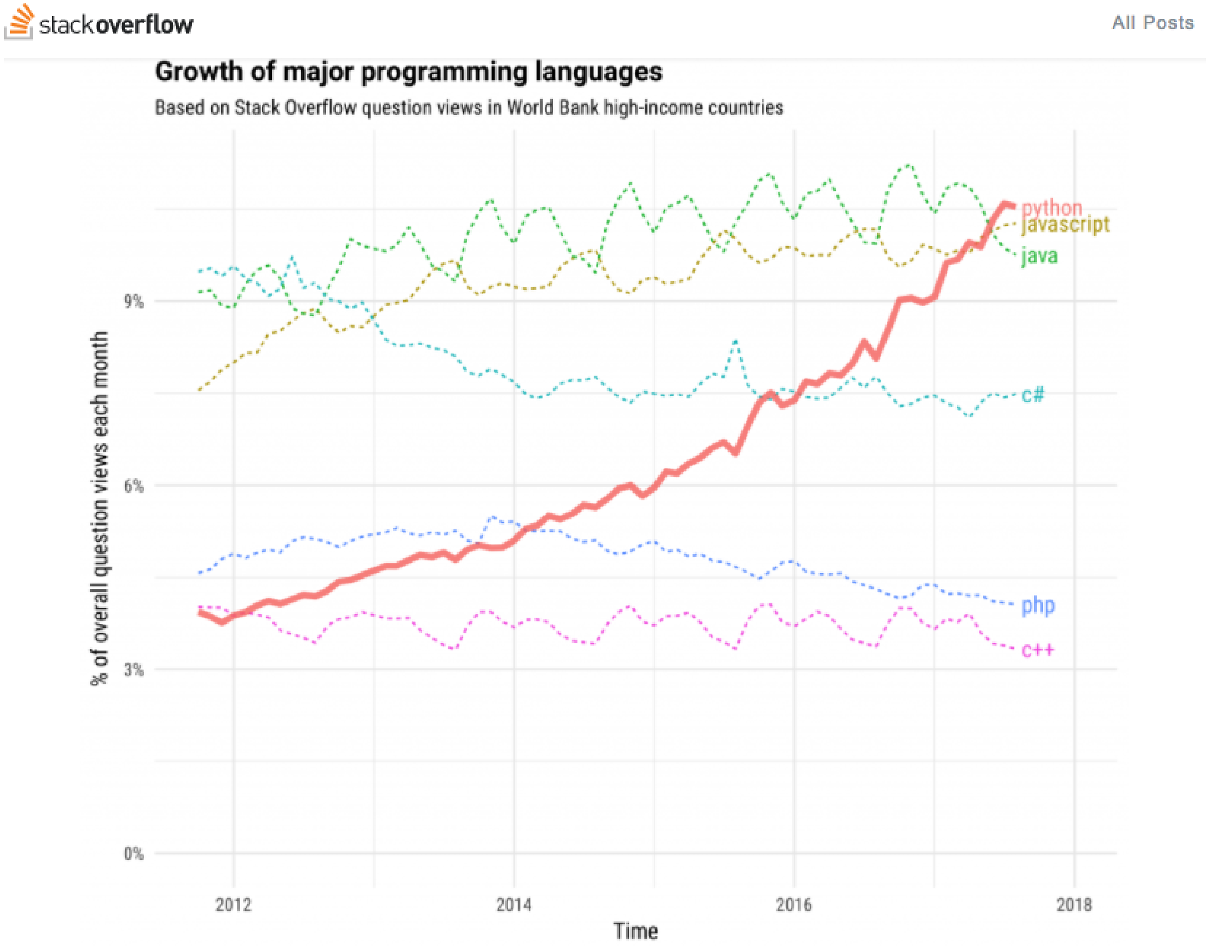
- this is why 1GB is greater than 1 billion bytes

Programming language popularity: TIOBE index



Taken from https://towardsdatascience.com/visualize-programming-language-popularity-using-tiobeindexpy-f82c5a96400d (https://towardsdatascience.com/visualize-programming-language-popularity-using-tiobeindexpy-f82c5a96400d)

Programming language popularity



Taken from [https://hackernoon.com/top-3-most-popular-programming-languages-in-2018-and-their-annual-salaries-51b4a7354e06 (https://hackernoon.com/top-3-most-popular-programming-languages-in-2018-and-their-annual-salaries-51b4a7354e06)](https://hackernoon.com/top-3-most-popular-programming-languages-in-2018-and-their-annual-salaries-51b4a7354e06)

Open Source Software

- Free computer software which the user can modify and distribute within the terms of a licence
- https://www.python.org/download/releases/3.3.5/license/ (https://www.python.org/download/releases/3.3.5/license/)
- Collaborative development has created diverse and very powerful software ecosystems
- Both major data science languages - Python and R are Open-source
- Python files are saved with the .py extension. These files on their own are called modules.
- Modular structure permits users to build an environment exactly suited to their needs.

In Python Everything is an Object

- objects have *classes*, meaning they represent a "type" of object,for example *string* or *function*
- *attributes* are features of objects or variables in a class
- *methods* are functions

Data types: Generically

- objects are *bound* to an identifier, e.g.

```
In [7]:
temperature = 98.6
print(temperature)
print(id(temperature))
```

```
98.6
92603568
```

- here, `temperature` is a variable name assigned to the literal floating-point object with the value of 98.6
- in Python, this is an instance of the **float** class
- function `id` returns the identity of an object

In [6]:

```
temperature1 = 98.6
print(temperature1 is temperature)
print(temperature1 == temperature)
```

```
92603616
False
True
```

- variable names in R and Python are *case-sensitive*
- some variable names are typically reserved, e.g.

```
False, True, None, or, and   # Python
  FALSE, TRUE, NA, NAN         # R
```

- All programming languages use comments, for humans to read
    - this is anything that follows the # character in both Python and R

> *"Let us change our traditional attitude to the construction of programs: Instead of imagining that our main task is to instruct a computer what to do, let us concentrate rather on explaining to human beings what we want a computer to do." -- Donald Knuth, Literate Programming (1984)*

- "immutable" objects cannot be subsequently changed

Python class	**Immutable**	**Description**	**R class**
bool	Yes	Boolean value	logical
int	Yes	integer number	integer
float	Yes	floating-point number	numeric
list	No	mutable sequence of objects	list
tuple	Yes	immutable sequence of objects	-
str	Yes	character string	character
set	No	unordered set of distinct objects	-
NumPy array	No	mutable array	-
dict	No	dictionary	(named) list

(indexing data cont.)

- index from 0 or from 1?
    - where an index begins counting, when addressing elements of a data object
    - most languages index from 0 (https://en.wikipedia.org/wiki/Comparison_of_programming_languages_%28
      reference_list)
    - human ages - do they index from 0?

```
In [23]: string_example = 'Hello World'
         string_example[0:5]

Out[23]: 'Hello'
```

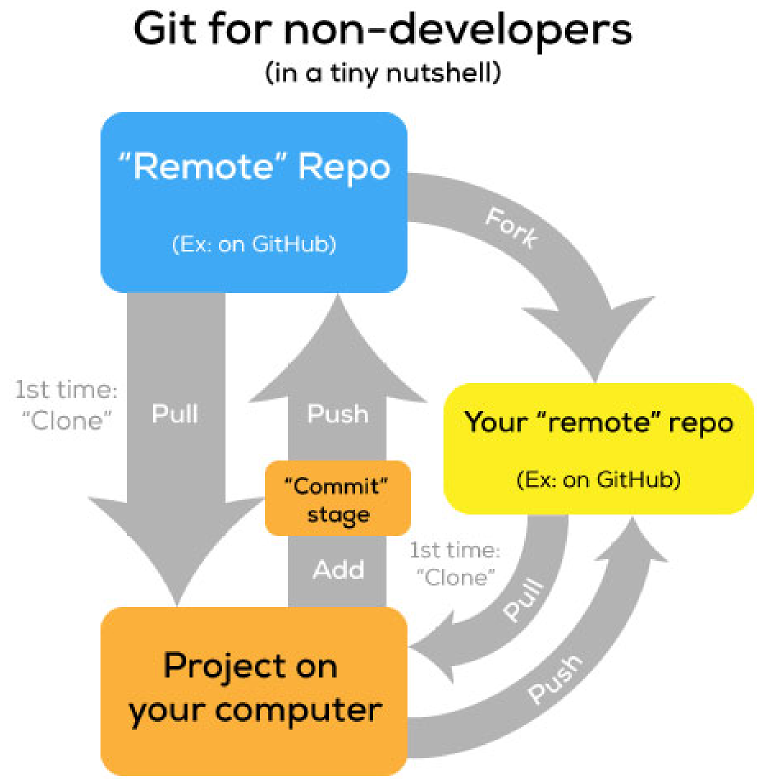
- Python indexes from 0

```
Be warned!
```

git

- `git`: a version control system
- Allows for complete history of changes, branching, staging areas, and flexible and distributed workflows

- simplified workflow (from [Anita Cheng's excellent blog post (http://anitacheng.com/git-for-non-developers)](http://anitacheng.com/git-for-non-developers))



GitHub

- a website and hosting platform for git repositories

- GitHub classroom (https://classroom.github.com)
- Free stuff for students! https://education.github.com/pack (https://education.github.com/pack)

More great resources for using git/GitHub

- An easy git Cheatsheet (http://rogerdudler.github.io/git-guide/files/git_cheat_sheet.pdf), by Nina Jaeschke and Roger Dudler
- git - the simple guide (http://rogerdudler.github.io/git-guide/) by Roger Dudler

git Example

Fixing a broken Python Jupyter notebook

This Jupyter notebook needs de-bugging:

https://github.com/lse-st445/lectures/week01/DebugExercise.ipynb (https://github.com/lse-st445/lectures/week01/DebugExercise.ipynb)

How to fix it:

- clone the repository
- edit the file
- stage the changes
- commit the changes
- issue a "pull request"

Markdown (and other markup languages)

- Idea of a "markup" language: HTML, XML, LaTeX
- "Markdown"
    - Created by John Gruber as a simple way for non-programming types to write in an easy-to-read format that could be converted directly into HTML
    - No opening or closing tags
    - Plain text, and can be read when not rendered
- Markdown has many "flavours" (https://github.com/commonmark/CommonMark/wiki/Markdown-Flavors)

Markdown example

This is a markdown example.

- bullet list 1
- bullet list 2

> *"I love deadlines. I like the whooshing sound they make as they fly by.(https://www.brainyquote.com/quotes/quotes/d/douglasada134151.html?src=t_funny)"*
> *-- Douglas Adams*

```
# Markdown example

This is a markdown example
* bullet list 1
* bullet list 2

> "[I love deadlines. I like the whooshing sound they make as they fly by.](http
s://www.brainyquote.com/quotes/quotes/d/douglasada134151.html?src=t_funny)"
-- _Douglas Adams_
```

Upcoming

- **Lab**: Working with Jupyter and Github
- **Next week**: Python and NumPy Data Structures