Week 4

- Data Clean up

Data cleanup and Anomalies

- Mid-term project schedule
- Scope of "Cleaning" and context
- File encoding
- Data Characterization
- Missing data
- Sanity checks
- Anomalies
- Outliers
- Homework

Post your idea on the discussion board under Midterm Section

Midterm Due

October 13, 2021

What is in scope for "data cleanup":

- *Data formatting*
- *Data characterization*
- *Sanity checks*
- *Anomaly identification*
- *Missing data*

These tasks enable "exploration."

The tasks are not sequential and are not ordered.

These tasks improve your understanding of the data set.

What you should walk away with tonight

- Be able to explain at least four sanity checks on data
- Apply remediation techniques for missing entries in data
- Identify outliers (anomalies) in data and document the action (or inaction taken)

Cleaning data is an investment

Before cleaning data, make sure outcome is relevant to your customers

Fake data can be useful
to help enumerate
requirements and expectations

Be Skeptical; Don't trust your data

- Can I contact the author if I have questions or concerns?
- Does data appear to be regularly updated and checked for errors?
- Does data come with information as to how it was acquired and what types of samples were used in its acquisition?
- Is there another source of data that can verify and validate this dataset?
- **Given my overall knowledge of the topic**, does this data seem plausible?

Source: page 128 of "Data Wrangling with Python"

- ~~Schedule for semester~~
- ~~Mid-term project schedule~~
- ~~Scope of "Cleaning" and context~~
- File encoding
- Data characterization
- Missing data
- Sanity checks
- Anomalies
- Outliers
- Homework

"Unreadable" files due to encodings

Have you seen text that looks like this?

Author: GuÃ°rÃºn GuÃ°mundsdÃ³ttir. Title: Introduction to character encoding (æ–‡å—ç¬¦å·åŒ–å…¥é–€). Copyright Â© 2004-2007 W3CÂ® (MIT, ERCIM, Keio).

Source: https://www.w3.org/International/questions/qa-what-is-encoding

"Unreadable" files due to encodings

Author: GuÃ°rÃºn GuÃ°mundsdÃ³ttir. Title: Introduction to character encoding (æ–‡å—ç¬¦å·åŒ–å…¥é–€). Copyright Â© 2004-2007 W3CÂ® (MIT, ERCIM, Keio).

Was that what the author intended?
Was the file corrupted?
Why does this happen?

Source: https://www.w3.org/International/questions/qa-what-is-encoding

"Unreadable" files due to encodings

Intended text:

Author: Guðrún Guðmundsdóttir. Title: Introduction to character encoding (文字符号化入門). Copyright © 2004-2007 W3C® (MIT, ERCIM, Keio).

but it may actually display like this:

Author: GuÃ°rÃºn GuÃ°mundsdÃ³ttir. Title: Introduction to character encoding (æ–‡å—ç¬¦å·åŒ–å…¥é–€). Copyright Â© 2004-2007 W3CÂ® (MIT, ERCIM, Keio).

Source: https://www.w3.org/International/questions/qa-what-is-encoding

Root cause: translating 1 and 0 to text

- Computers store data as sequences of ones and zeros.
- Humans prefer to use a larger set of symbols

--> Which symbols are relevant depends on your situation

--> The translation from "10010110101" to a symbol depends on which convention you use

ASCII

"American Standard Code for Information Interchange"
--> identifies the problem in the name

ASCII represents 8 bytes data
ASCII only has 256 different characters.
Motive: concise

Char	Dec	Oct	Hex	Char	Dec	Oct	Hex	Char	Dec	Oct	Hex
(sp)	32	0040	0x20	@	64	0100	0x40	`	96	0140	0x60
!	33	0041	0x21	A	65	0101	0x41	a	97	0141	0x61
"	34	0042	0x22	B	66	0102	0x42	b	98	0142	0x62
#	35	0043	0x23	C	67	0103	0x43	c	99	0143	0x63
$	36	0044	0x24	D	68	0104	0x44	d	100	0144	0x64
%	37	0045	0x25	E	69	0105	0x45	e	101	0145	0x65
&	38	0046	0x26	F	70	0106	0x46	f	102	0146	0x66
'	39	0047	0x27	G	71	0107	0x47	g	103	0147	0x67
(	40	0050	0x28	H	72	0110	0x48	h	104	0150	0x68
)	41	0051	0x29	I	73	0111	0x49	i	105	0151	0x69
*	42	0052	0x2a	J	74	0112	0x4a	j	106	0152	0x6a
+	43	0053	0x2b	K	75	0113	0x4b	k	107	0153	0x6b
,	44	0054	0x2c	L	76	0114	0x4c	l	108	0154	0x6c
-	45	0055	0x2d	M	77	0115	0x4d	m	109	0155	0x6d
.	46	0056	0x2e	N	78	0116	0x4e	n	110	0156	0x6e
/	47	0057	0x2f	O	79	0117	0x4f	o	111	0157	0x6f
0	48	0060	0x30	P	80	0120	0x50	p	112	0160	0x70
1	49	0061	0x31	Q	81	0121	0x51	q	113	0161	0x71
2	50	0062	0x32	R	82	0122	0x52	r	114	0162	0x72
3	51	0063	0x33	S	83	0123	0x53	s	115	0163	0x73
4	52	0064	0x34	T	84	0124	0x54	t	116	0164	0x74
5	53	0065	0x35	U	85	0125	0x55	u	117	0165	0x75
6	54	0066	0x36	V	86	0126	0x56	v	118	0166	0x76
7	55	0067	0x37	W	87	0127	0x57	w	119	0167	0x77
8	56	0070	0x38	X	88	0130	0x58	x	120	0170	0x78
9	57	0071	0x39	Y	89	0131	0x59	y	121	0171	0x79
:	58	0072	0x3a	Z	90	0132	0x5a	z	122	0172	0x7a
;	59	0073	0x3b	[	91	0133	0x5b	{	123	0173	0x7b
<	60	0074	0x3c	\	92	0134	0x5c	|	124	0174	0x7c
=	61	0075	0x3d	]	93	0135	0x5d	}	125	0175	0x7d
>	62	0076	0x3e	^	94	0136	0x5e	~	126	0176	0x7e
?	63	0077	0x3f	_	95	0137	0x5f				

ASCII versus Unicode versus UTF

- Unicode represents 16 bytes data for supporting more characters.
- Unicode allows for up to 65,536 different characters.
- Languages such as Japanese and Chinese have thousands of characters.

UTF-8 is a compromise character encoding that can be as compact as ASCII (if the file is just plain English text) but can also contain any unicode characters (with some increase in file size). UTF stands for Unicode Transformation Format. The '8' means it uses 8-bit blocks to represent a character.

Relevance to you, the data scientist

- files do not adhere to a single convention
- Assuming an encoding for input files will break your process

These illustrate the

- https://donatstudios.com/CSV-An-Encoding-Nightmare
- https://www.joelonsoftware.com/2003/10/08/the-absolute-minimum-every-software-developer-absolutely-positively-must-know-about-unicode-and-character-sets-no-excuses/

"Unreadable" files due to encodings

In Pandas, try calling read_csv with

- encoding='latin1',
- encoding='iso-8859-1'
- encoding='cp1252'

(these are some of the various encodings found on Windows) citation

```
df= pd.read_csv('Region_count.csv',encoding ='latin1')
```

Suppose you can load the data into a dataframe.

--> *Don't jump immediately into getting a result*

Even when data can be read into your application, additional cleaning may be needed

Can you proceed without cleaning data?

Suppose you can load the data into a dataframe.

--> *Don't jump immediately into getting a result*

Even when data can be read into your application, additional cleaning may be needed

Cleaning and analytic development is an iterative process of finding issues and resolving them while also creating the desired outcome (without introducing bias)

- ~~Schedule for semester~~
- ~~Mid-term project schedule~~
- ~~Scope of "Cleaning" and context~~
- ~~File encoding~~
- Data characterization
- Missing data
- Sanity checks
- Anomalies
- Outliers
- Homework

Data structure

Here I assume data in a table, e.g. Pandas

Data doesn't necessarily arrive in a table:

- Unstructured text
- JSON
- XML
- Collection of images/audio/video
- Collection of mixed media (image/text/Powerpoint/Excel/video)

--> Need to characterize data regardless of data structure

Describe the data

- How big is the input file? (on disk, in memory)
- If input is recurring, how often does it arrive? On a schedule or event driven?
- In a table, how many rows and columns?
- What are the data types in each column? (May need transformation)

Demos; lots of demos

```
cleaning_data/loans_1_table_size.ipynb
cleaning_data/loans_2_data_types.ipynb
cleaning_data/loans_3_describe.ipynb
cleaning_data/loans_4_data_type_conversion.ipynb
cleaning_data/loans_5_unique_entries.ipynb
```

Visualizing data

`cleaning_data/visualize_baseball.ipynb`

Visual clean up

Browse the tabs on the page

- https://python-graph-gallery.com/134-how-to-avoid-overplotting-with-python/

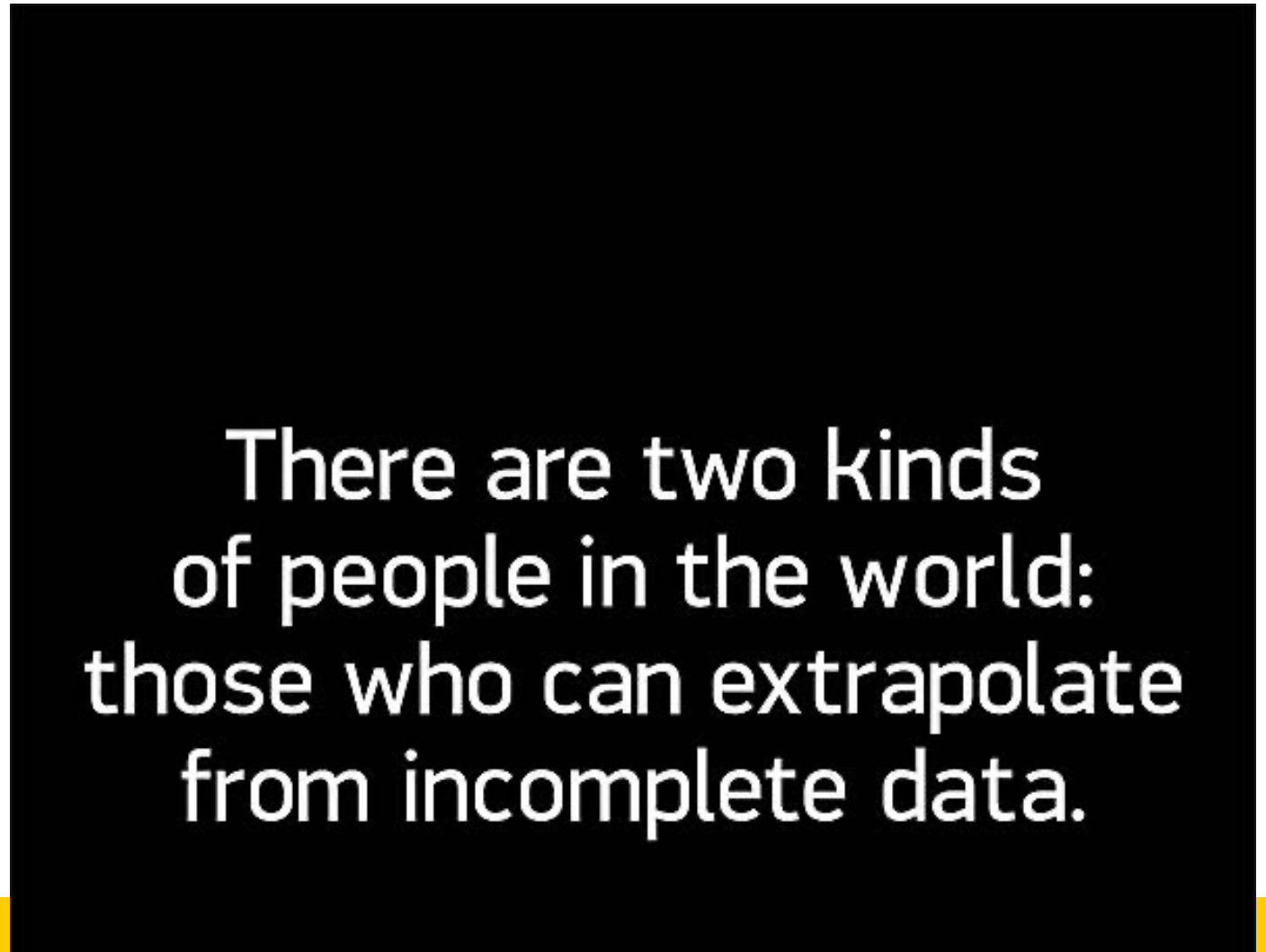
To view different layouts of the same data

Visual design matters to your story. See the animation on the page

- https://www.darkhorseanalytics.com/blog/data-looks-better-naked

- ~~Mid-term project schedule~~
- ~~Scope of "Cleaning" and context~~
- ~~File encoding~~
- ~~Data characterization~~
- Missing data
- Sanity checks
- Anomalies
- Outliers
- Homework

Missing data



Missing data

Great walk through:
https://pandas.pydata.org/pandas-docs/stable/user_guide/missing_data.html

What to do about missing data? Options:

- Proceed as is
- Fill in with fixed value
- Fill using adjacent values
- Interpolation

https://en.wikipedia.org/wiki/Imputation_(statistics)

First, characterize

- cleaning_data/loans_empty_entries_1.ipynb
- cleaning_data/loans_empty_entries_2_visualize_sparseness.ipynb

What to do with NaN entries?

- Drop all columns where every value is NaN

```
df=df.dropna(how='all',axis=1)
```

- Replace NaN with numeric value out of bounds

```
df=df.fillna(-1)
```

If distribution of variable is known

- If variable adheres to known distribution, sample replacement values or supplement insufficient data from that distribution

- Be careful: missing values may have a bias, so simply inspecting the existing entries may yield a distribution that isn't representative

Interpolation of missing variables based on adjacent values

- `df.interpolate()`

AirBnB machine learning needs clean data

Example of real-world application

Goal: Binary classification of fraud/not fraud

- fraud prediction models using numerical and categorical
- Use k nearest neighbors to find adjacent entities
- Then fill in missing values based on adjacent values

https://medium.com/airbnb-engineering/overcoming-missing-values-in-a-random-forest-classifier-7b1fc1fc03ba

- ~~Mid-term project schedule~~
- ~~Scope of "Cleaning" and context~~
- ~~File encoding~~
- ~~Data characterization~~
- ~~Missing data~~
- Sanity checks
- Anomalies
- Outliers
- Homework

sanity checks for numerical variables

Is numerical variable within a bounded range?

- Cost of candy bar is $132,402 USD
- Cost of a sofa is $0.0042

For numerical variables, sort or find max and min

How to sanity check

- For numeric values, review the maximum and minimum

sanity checks for numerical variables

Variance too wide or not as wide as expected

- 15 of 28 bus arrival times between 3:09:12pm and 3:09:34pm

For numerical variables, measure variance

sanity checks for numerical variables

Are values unusual?

- Ages of 100 people, but only ages are 10,20,30,40

For numerical variables,
a histogram would show this if
bin count is appropriate

How to sanity check

- For numeric values, review the maximum and minimum
- For numeric values, inspect the distribution

sanity checks for numerical variables

Check units of all columns to make sure they are meaningful

index	count	distance	time	rate
		miles	*horses*	*movies per hour*
1	53	4	4.2	2
2	42	24	3.2	45
3	25	5	6.4	5
4	32	2	1.4	5

Pro-tip: Make headers human readable

Data cleanup includes helping your own interpretation

```
df.columns=['speed in mph', 'height in ft']
```

sanity checks for numerical variables

Are values non-sensical?

- Negative values (i.e. a length)
- Fractional values for counting (i.e., there are 4.28 cows)
- Percentages of a whole that exceed 100% (i.e., 153% of the chapters in a book)

For numerical variables, sort or find max and min

How to sanity check

- For numeric values, review the maximum and minimum
- For numeric values, inspect the distribution
- Check data type
  - Numeric values that should be integers

Categorical variables: are there rare outcomes?

```
for this_column in df.columns:
    print(this_column,"has",df[this_column].nunique(),"unique entries")
    print(df[this_column].value_counts().head(10))
```

sanity checks for numerical variables

Values that oscillate (usually temporal variation)

- https://en.wikipedia.org/wiki/Diurnal_temperature_variation
- https://en.wikipedia.org/wiki/Diurnal_cycle

Activity: Brainstorm sanity checks for text

First, enumerate common text variables as a group on the whiteboard

Activity: Brainstorm sanity checks for text

- What constraints would you expect for these?
- What exceptions would be indicators of problems?

First, enumerate common text variables as a group on the whiteboard

Example sanity checks for text

- Street Address: 59@4 Thomas Street
- Name: Alice, Be#$, Kate
- Name: Buck, Robert, 4' cable, Cindy
- Categorical: Yes/No/Cats
- Lists elements
  - List of names: ['Bob', 'Mary', 524]
  - List of colors: ['Blue', 'Red', 'Orange', 'Fruit']
  - List of states: ['Missouri', 'Wisconsin', 'Mexico']
  - List sizes: len(list_of_days_in_week) indicates 9
- Data that claims to be XML, but is actually JSON
- Data that claims to be text and is actually binary or otherwise encoded

Causes for outliers and anomalies and invalid data

- Data entry errors (human errors)
- Measurement errors (instrument errors)
- Experimental errors (data extraction or experiment planning/executing errors)
- Intentional (dummy outliers made to test detection methods)
- Data processing errors (data manipulation or unintended mutations)
- Sampling errors (extracting or mixing data from wrong or various sources)
- Natural (not an error; actual novelty in data)

source

https://en.wikipedia.org/wiki/Outlier#Causes

What to do when sanity checks indicate issue?

- Fix minor formatting issues i.e., with Regular Expressions (**document your process**)

- Revisit the data collection process
  - –Talk to the data owner; your understanding may be incomplete
  - –Review code used to collect data

Data cleaning: 1 is an exception, 2 is a pattern

Consistency of data comparing before cleanup with after cleanup: the cleaning shouldn't have altered essential aspects of the data

- The point is that there is **not** a "right" choice.
- Document your action and your justification.
- Publicize decisions you made
- Publicize your documentation of how you arrived at the decision

Review of sanity checks

- Is numerical variable within a bounded range?
- Variance too wide or not as wide as expected
- Check units of all columns to make sure they are meaningful
- Are values non-sensical?
- Are values unusual?
- Values that oscillate (usually temporal variation)
- Text anomalies

- ~~Mid-term project schedule~~
- ~~Scope of "Cleaning" and context~~
- ~~File Encodings~~
- ~~Data Characterization~~
- ~~Sanity checks~~
- Anomalies
- Outliers
- Missing data
- Homework

Detecting anomalies in data isn't routine

Data scientists often trust their data; this can be a mistake

- In text, Zipf's law applies to word frequency
  - Use case: distinguish natural language from gibberish
- In text, letter frequency depends on language
  - Use case: language detection
- For numerical data, Benford's law applies to leading digits
  - Use case: fraud dection [citation, citation]

- ~~Mid-term project schedule~~
- ~~Scope of "Cleaning" and context~~
- ~~File Encodings~~
- ~~Data Characterization~~
- ~~Sanity checks~~
- ~~Anomalies~~
- Outliers
- Missing data
- Homework

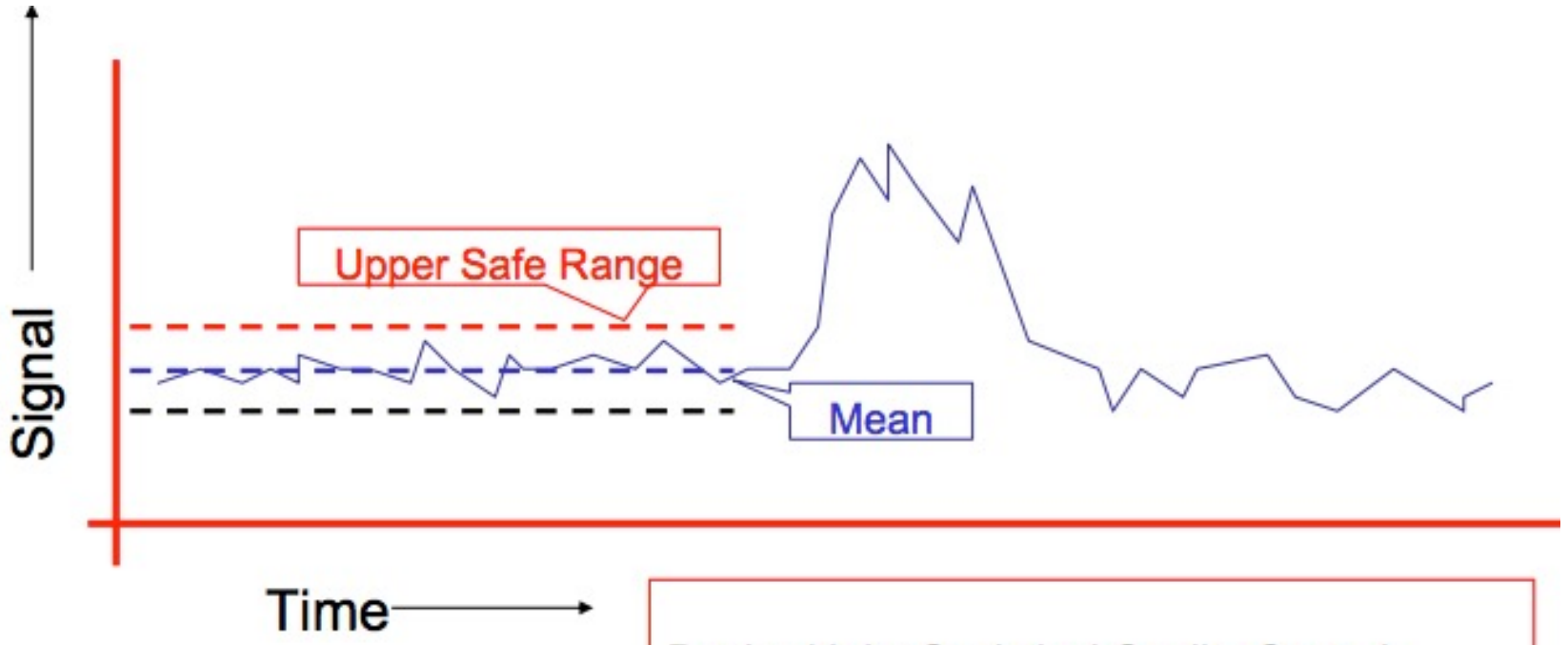
"Outliers" imply a normal population

Define "normal", then determine which points deviate from that definition
https://en.wikipedia.org/wiki/Anomaly_detection

There are statistical tests to run to measure deviation, e.g.,

- https://en.wikipedia.org/wiki/Grubbs%27s_test_for_outliers

This is a common need in data science; there are a variety of techniques



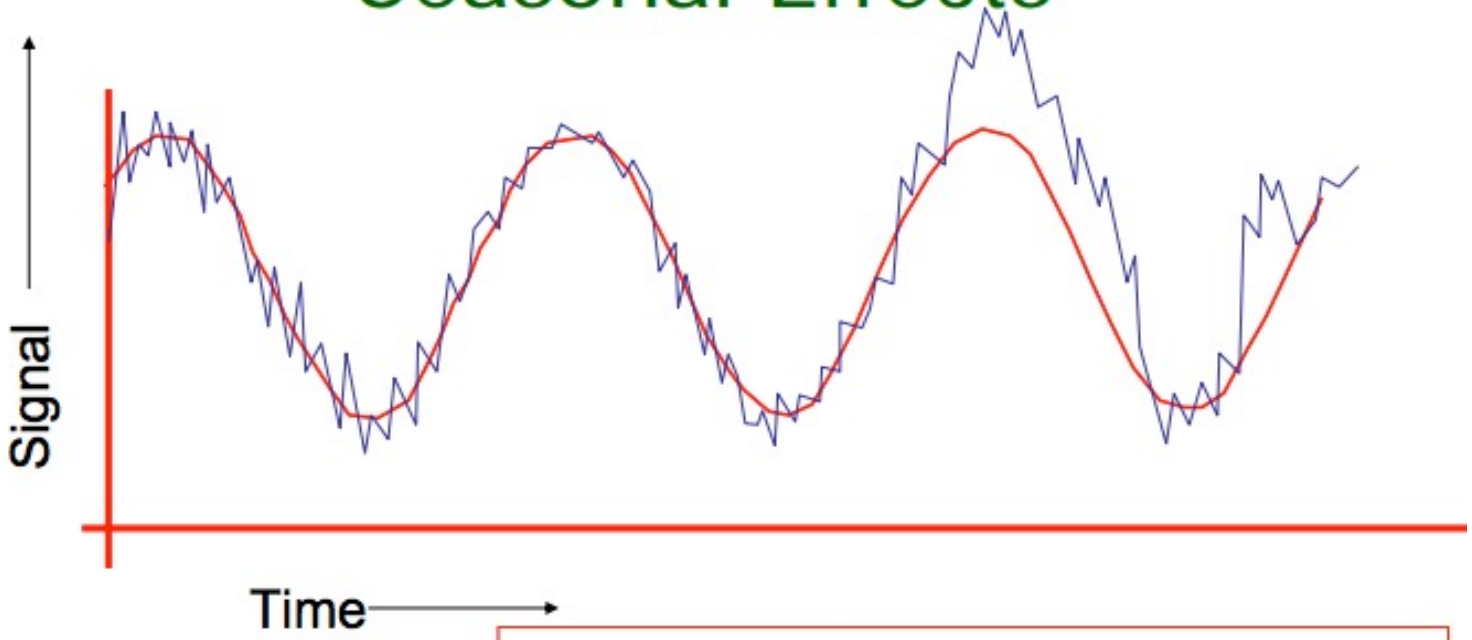
Dealt with by Statistical Quality Control

Record the mean and standard deviation up the the current time.

Signal an alarm if we go outside 3 sigmas

Source: https://www.autonlab.org/_media/tutorials/biosurv01.pdf

Seasonal Effects



Fit a periodic function (e.g. sine wave) to previous data. Predict today's signal and 3-sigma confidence intervals. Signal an alarm if we're off.

Reduces False alarms from Natural outbreaks.

Different times of year deserve different thresholds.

Source: https://www.autonlab.org/_media/tutorials/biosurv01.pdf

Dealing with Outliers

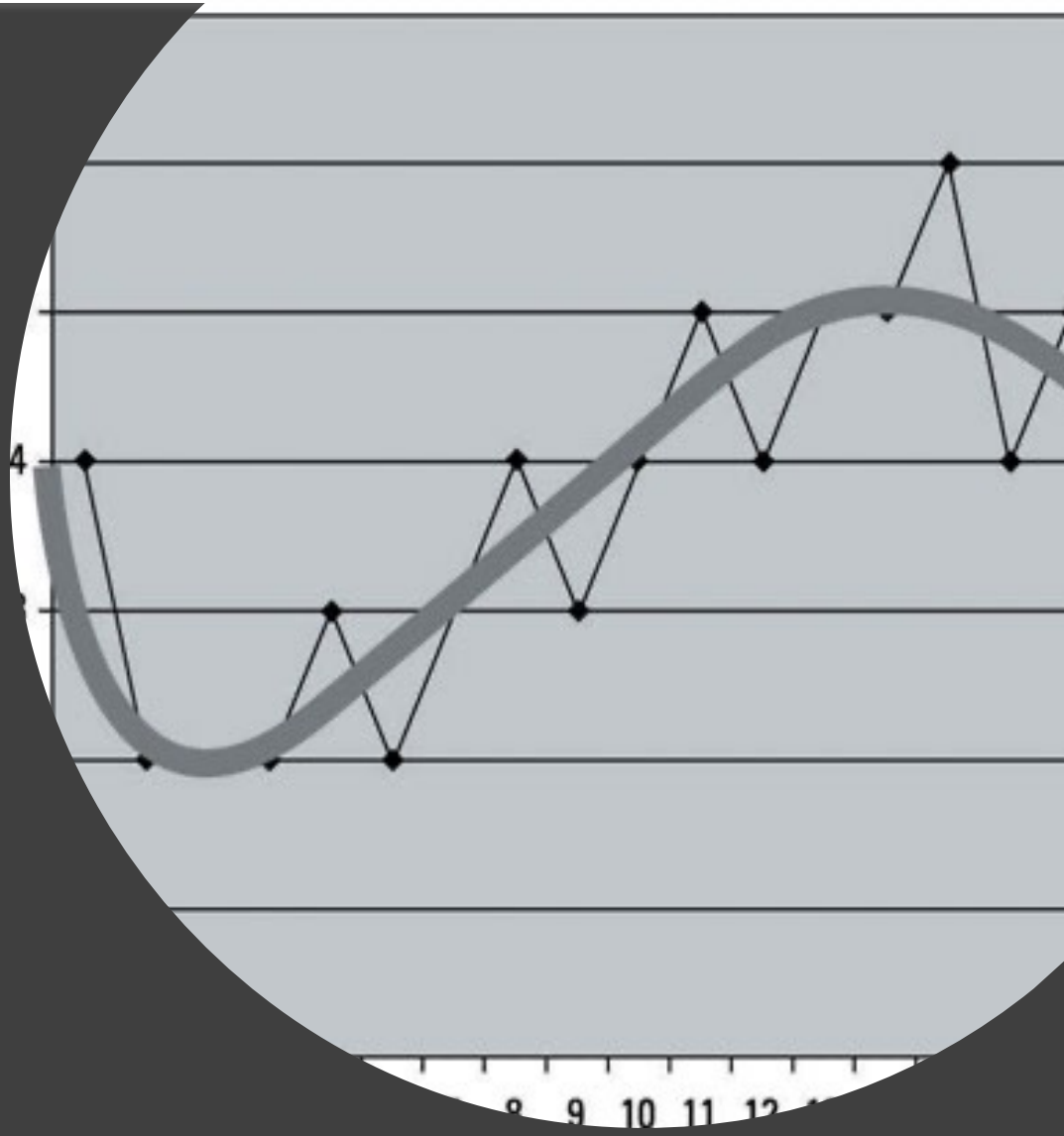
- You want to clean your data, not manipulate or change it
- Be able to justify removal.
- Be explicit in your final conclusions if you removed outliers to normalize your data.

This is both an ethical issue and a trust issue.

Source: page 167 of "Data Wrangling with Python"

Smoothing data:

Find the signal in the noise by discarding variance which is considered noisy



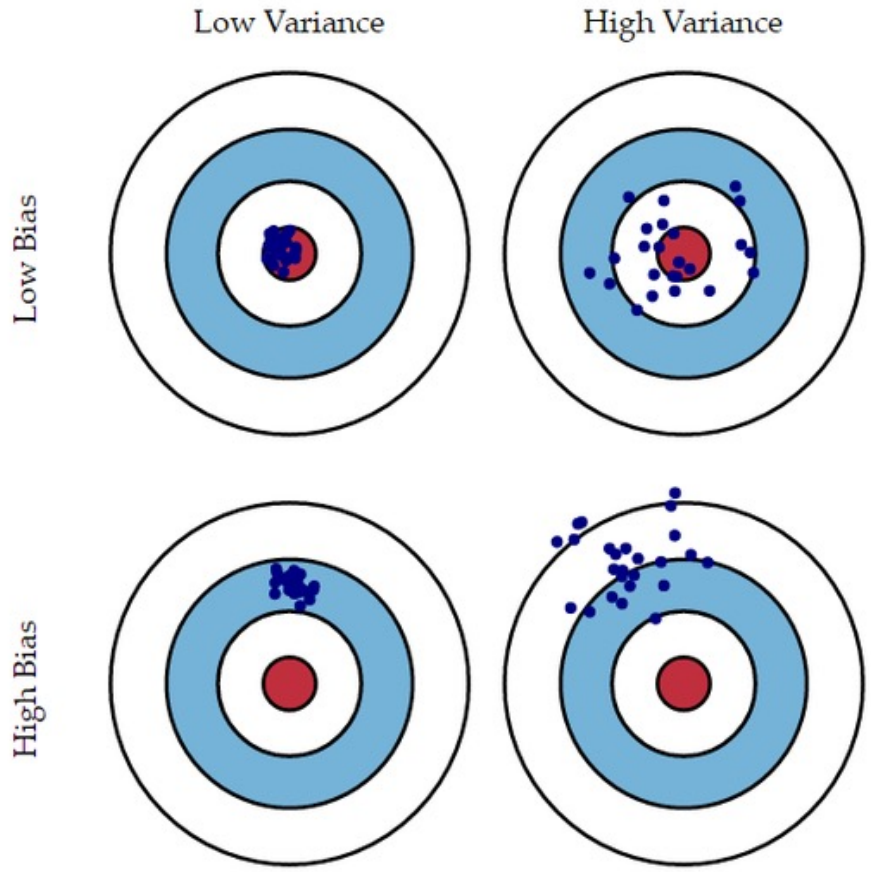
Smoothing data

When is it useful?

- Highlight data trends and focus the story on the relevant aspects
- Ignore data that distracts from the story being told

What are the dangers?

- Excessively smoothed models have a high bias (not close to truth), even if they have low variance. Models that are rough have high variances and low bias (close to truth).



- ~~Mid-term project schedule~~
- ~~Scope of "Cleaning" and context~~
- ~~File Encodings~~
- ~~Data Characterization~~
- ~~Sanity checks~~
- ~~Missing data~~
- ~~Anomalies~~
- ~~Outliers~~
- Homework

How to be a successful data scientist

- Have multiple projects active at all times (though not too many)
  - Be able to split attention
  - Enables cross pollination of ideas
  - If something fails/is waiting/terminates, you can switch to something else
  - Don't spread yourself too thinly

- Deliver results early and often
  - Constant dialog and feedback
  - Be able to explain value to organization