# IBEHS 4C03: Statistical Methods in Biomedical Engineering

## Data Preprocessing

Carol Bassim , DMD, MHS
Assistant Professor, CLA
Division of Education and Innovation
Department Of Medicine

location: MDCL
phone: (905) 525-9140
email: bassimc@mcmaster.ca
web: http://ibiomed.mcmaster.ca

McMaster University

BRIGHTER WORLD

# Data in Engineering 13

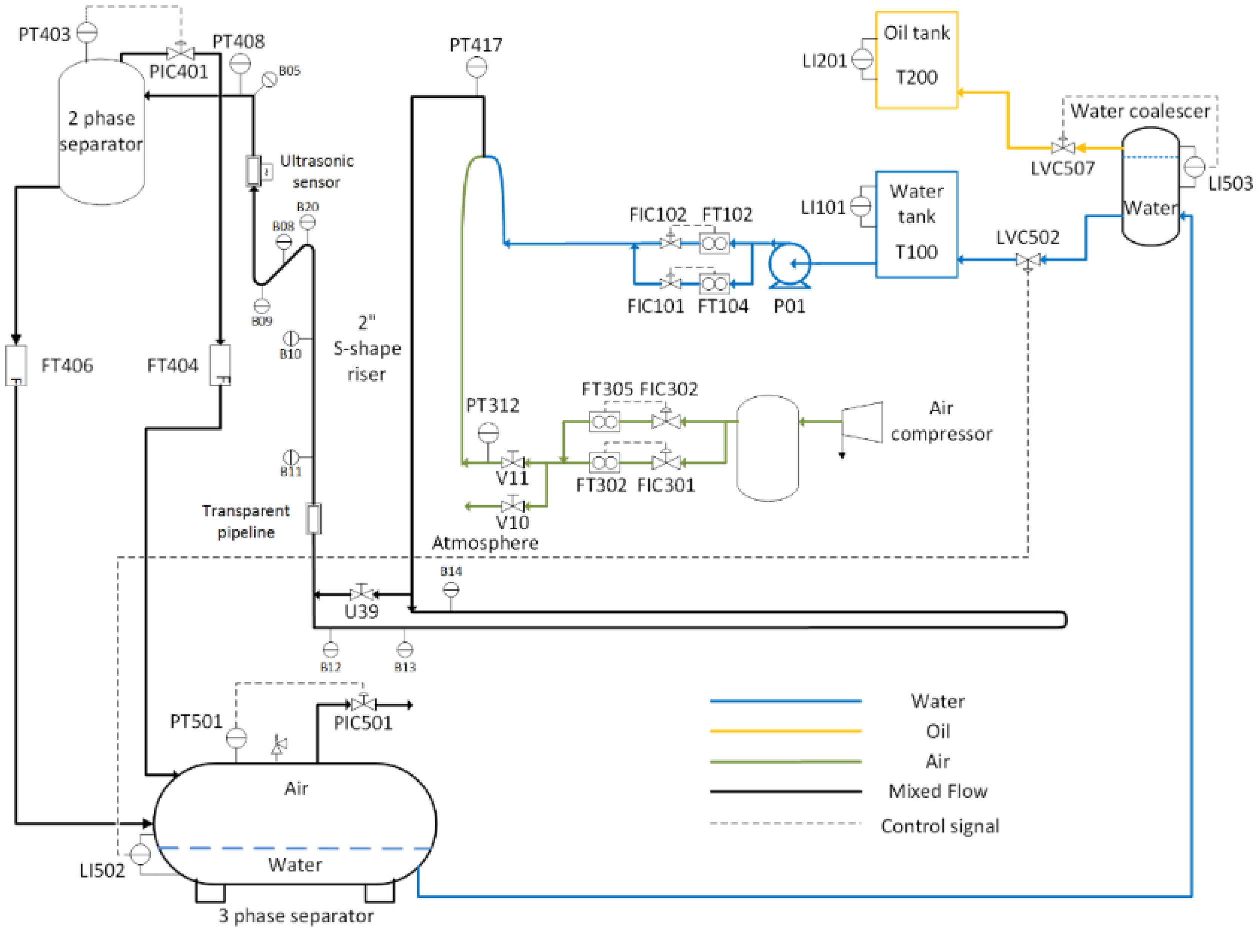
McMaster University

BRIGHTER WORLD

# Data in Engineering: You generate a lot of data

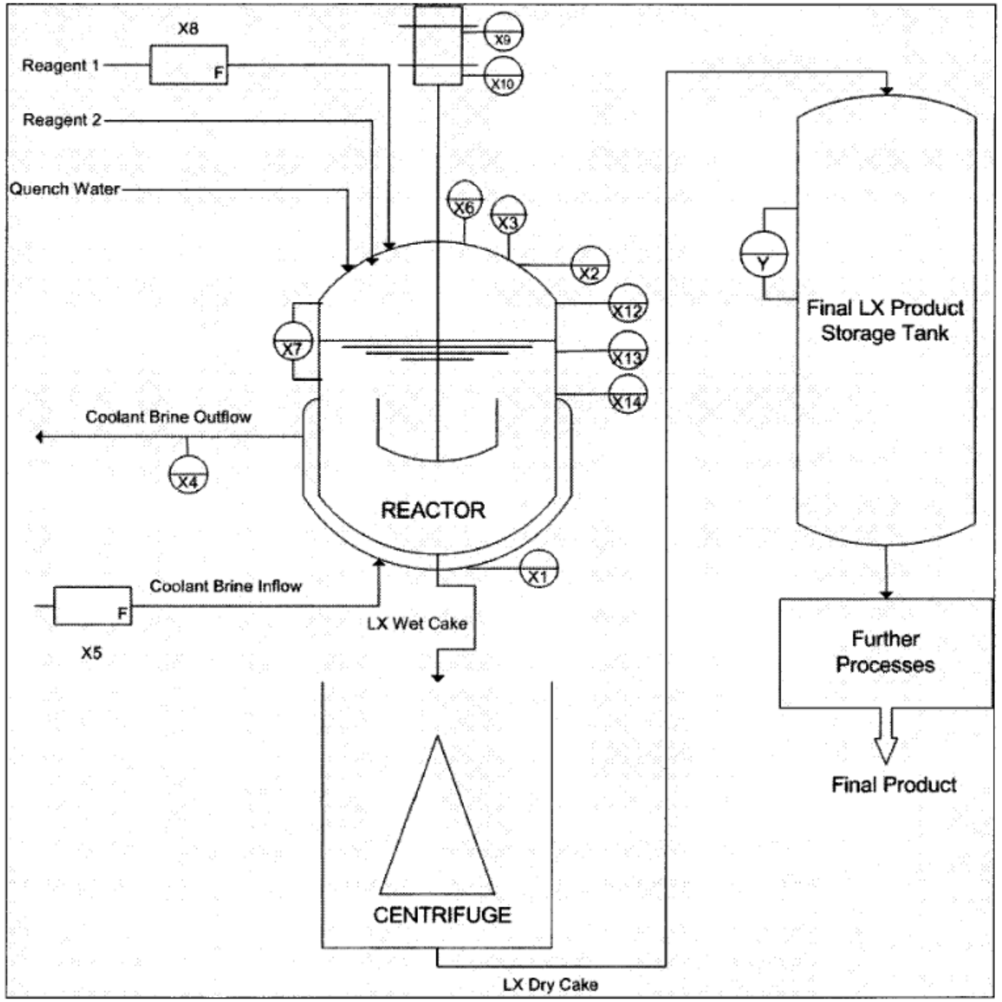
| Measured variable | Sampling rate | Availability | Platform |
|---|---|---|---|
| Process variables | 1 Hz | Continuous | DeltaV |
| Alarm, event, change logs | Event driven | Discrete event | DeltaV |
| Doppler ultrasonic sensor | 10 kHz | 60 s | LabView |
| High frequency pressure sensors | 5 kHz | 60 s | LabView |
| Videos | - | 30-60 s | Camera |

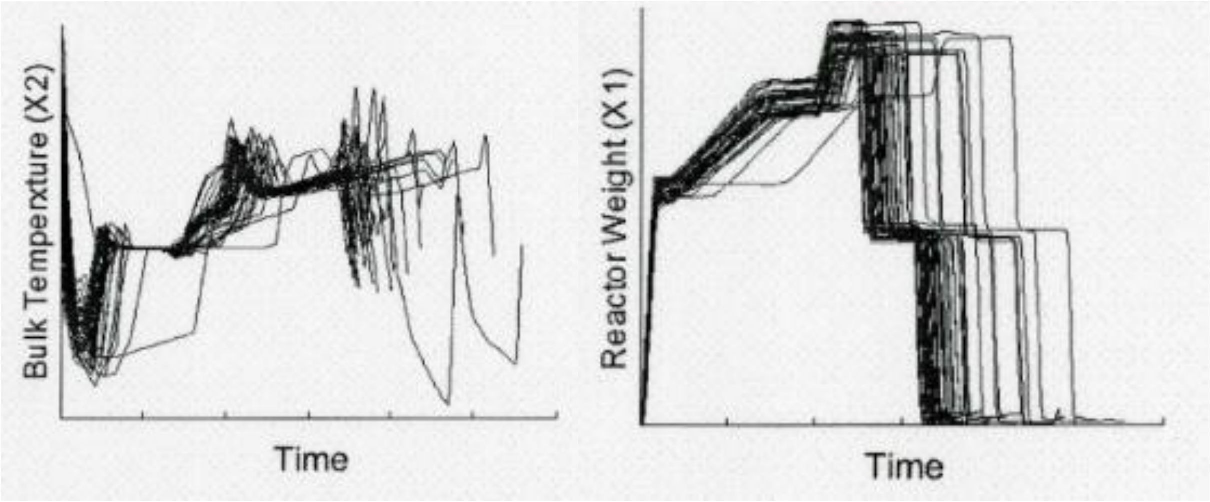
- 29 measure process variables
- 9 high frequency pressure sensors
- 2 cameras
- > 3 GB of data *per day*




A. Stief, R. Tan, Y. Cao, J. R. Ottewill, N. F. Thornhill, J. Baranowski, A heterogeneous benchmark dataset for data analytics: Multiphase flow facility case study, Journal of Process Control, 79 (2019) 41–55, DOI: https://doi.org/10.1016/j.jprocont.2019.04.009

# Data in Engineering




- 13 measured process variables
- *58 batches*

From Cecilia Rodrigues' M.A.Sc thesis, 2006, McMaster University, used with permission

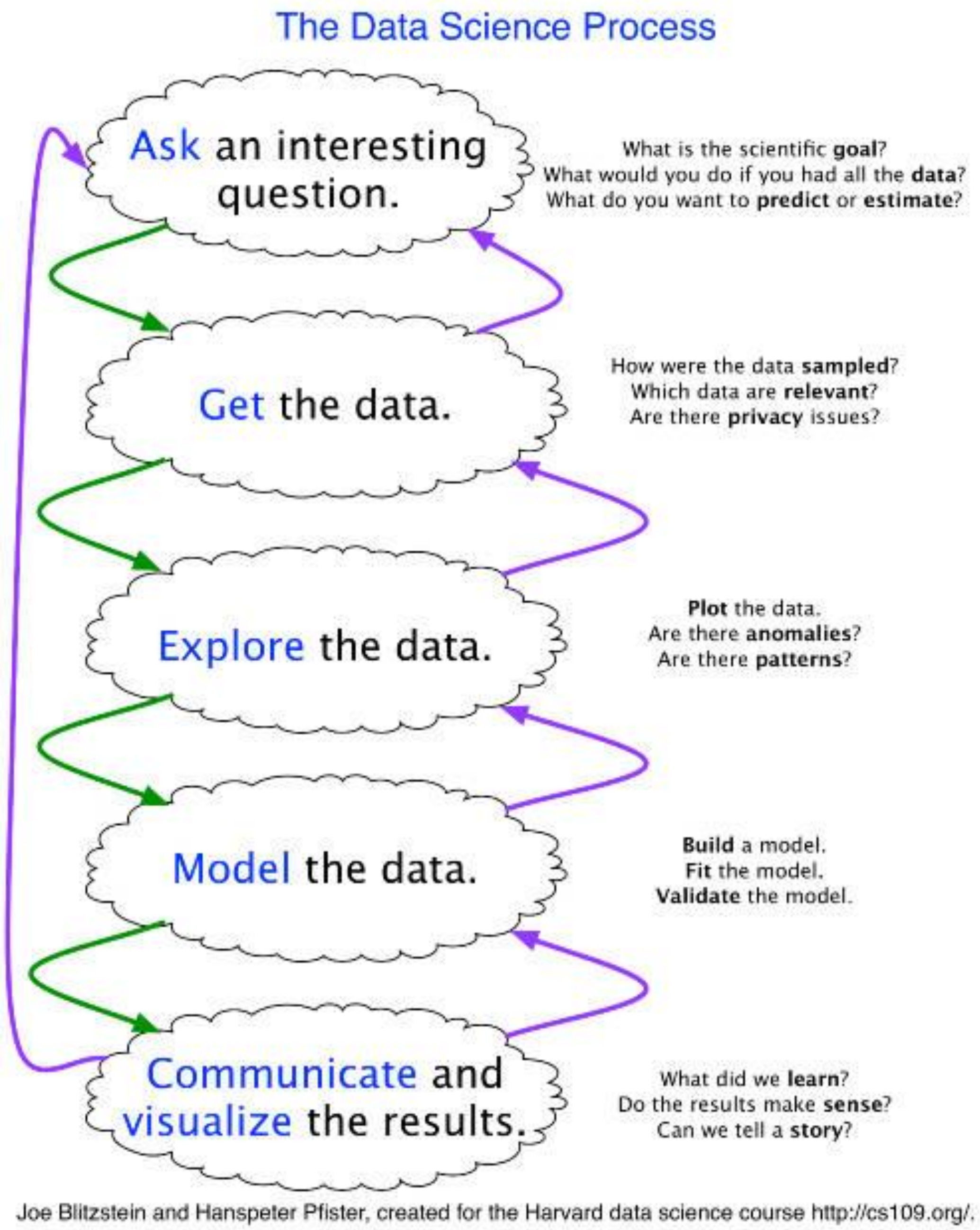
# Data Preprocessing

**Data preprocessing** is the manipulation and/or dropping of data before it is used in order to ensure or enhance performance.

We say we like data, but we don't…
We like insights from data
- Bad Data Handbook (McCallum 2013)

# Data Science Workflows

1. Blitzstein and Pfister workflow: The Data Science Process




“the data science workflow is not a linear process, instead it’s non-linear and extremely iterative”

# Data Science Workflows

1. CRISPT-DM: Cross-Industry Standard Process for Data

Phase 1: Business Understanding
Phase 2: Data Understanding
Phase 3: Data Preparation
Phase 4: Modeling
Phase 5: Evaluation
Phase 6: Deployment


"the standard process model was led by five companies, and has been added to by IBM"

# Data Science Workflows

1. OSEMN
    - Obtain
    - Scrub
    - Explore (Exploratory Data Analysis)
    - Model
    - iNterpret

A taxonomy of data science: by Hilary Mason and Chris Wiggibs
https://sites.google.com/a/isim.net.in/datascience_isim/taxonomy

“people often remember the framework by recalling how close sounding OSEMN is to “possum” or “awesome””

# Project Steps

1. Define the Problem
2. Data Collection and Assembly
3. Data Preprocessing
   - Cleaning
   - Data Exploration
   - Visualization and Descriptions
   - Feature engineering
4. Data Analysis and/or Model Building
5. Model and/or Test Evaluation and Interpretation
6. Reporting, Dissemination, and Communication

# Why clean data?

- Data rarely arrives with a quality guarantee
- **Data typically arrives with little documentation** of where exactly it came from, how it was gathered and what to watch out for when using it
- Relatively simply analysis can provide a lot of insight into new data sets
- **‘Bad’ data can give erroneous results**
- What is bad data?
  - Technical issues: missing data, malformed records, etc.
  - Data you can’t access, data that changed since last time you looked at it
  - **Bad data is data that gets in the way**
  - **“Garbage in, garbage out”**

# Steps in Data Preprocessing

14

# Common Steps to Data Preprocessing

Common things to check in your data:

1) Understand the data format
2) Field validation
3) Value validation
4) Missing data
5) Scaling
6) Dealing with categorical data

# Common Steps to Data Preprocessing

Common things to check in your data:

1) Understand the data format
2) Field validation
3) Value validation
4) Missing data
5) Scaling
6) Dealing with categorical data

- Format of the files?
  - e.g., .csv, .json, data base connection, SCADA (Supervisory Control and Data Acquisition)?
- Encoding of the file?
  - e.g., date/time format

# Common Steps to Data Preprocessing

Common things to check in your data:

1) Understand the data format
2) Field validation
3) Value validation
4) Missing data
5) Scaling
6) Dealing with categorical data

- Where are the data fields coming from?
- Do sensor tags need to be matched to physical unit?
- What are the units for all fields?
- Are they the correct format? e.g. Website visits should be an integer not a decimal value
- Are the data types consistent with what you want them to be?

# Common Steps to Data Preprocessing

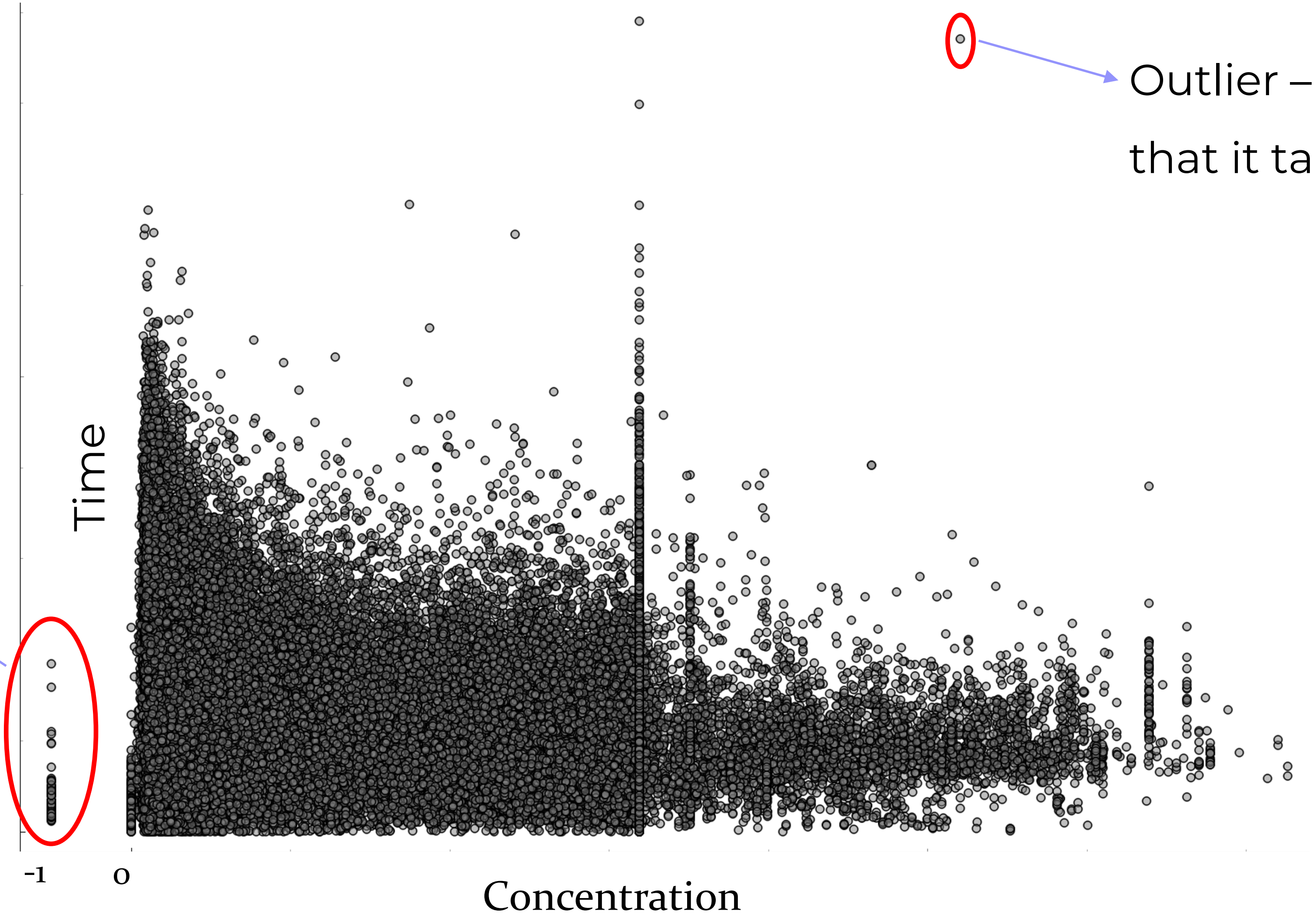
Common things to check in your data:

1) Understand the data format
2) Field validation
3) Value validation
4) Missing data
5) Scaling
6) Dealing with categorical data

- Are there any nonsensical data?
  - e.g. Wikipedia had 1.06 E69 page views on June 7th 2011
  - 1,060,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000,000
  - For reference: radius observable universe: 8.8 E26 m
- Outliers and nonsensical data are different
- Nonsensical data can be removed (treated then as missing data)

https://dumps.wikimedia.org/other/pagecounts-raw/2011/2011-06
https://en.wikipedia.org/wiki/Observable_universe/

# Outlier vs. Nonsensical data




Outlier – possible that it takes this long

Negative concentration?!

# Common Steps to Data Preprocessing

Common things to check in your data:

1) Understand the data format
2) Field validation
3) Value validation
4) Missing data
5) Scaling
6) Dealing with categorical data

- Many reasons for missing data
- Generally, don't want missing data
  - Can cause errors in statistical analysis
- Some methods to handle
  - Ignore/remove it – works well for data sets with few missing data (small percent of all data)
  - Use the previous value or interpolate
  - Replace with standard statistic value (e.g. mean, median, mode)

# Missing Data Example

Original Data

| Batch | Yield (g/L) |
|---|---|
| 0 | 83.5 |
| 1 | |
| 2 | 93.2 |
| … | |
| 1000 | 81.6 |

Makes the most sense given the context (not time series data, one missing data point)

Ignore/Drop It

| Batch | Yield (g/L) |
|---|---|
| 0 | 83.5 |
| 2 | 93.2 |

Carry Forward

| Batch | Yield (g/L) |
|---|---|
| 0 | 83.5 |
| 1 | 83.5 |
| 2 | 93.2 |

Interpolate

| Batch | Yield (g/L) |
|---|---|
| 0 | 83.5 |
| 1 | 88.4 |
| 2 | 93.2 |

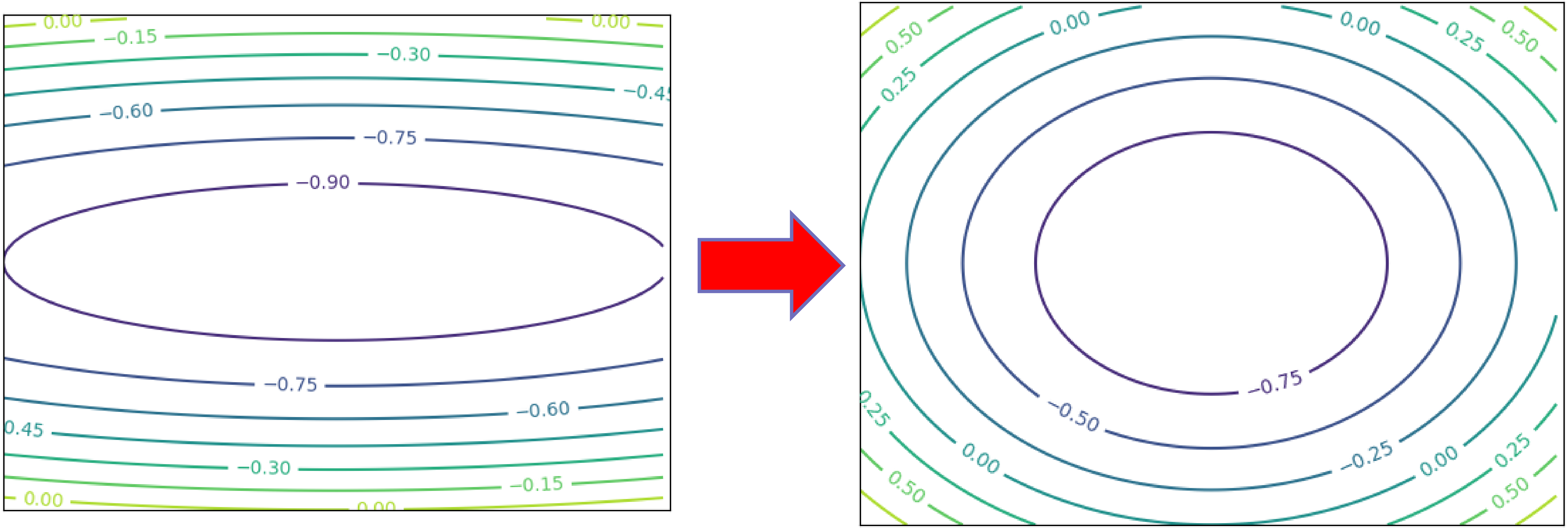
# Common Steps to Data Preprocessing

Common things to check in your data:

1) Understand the data format
2) Field validation
3) Value validation
4) Missing data
5) Scaling
6) Dealing with categorical data

- Variables at different scales can skew data models
- Normalization ensures that each variable contributes approximately proportionally to chosen metric
- Methods to normalize data:
    - Min-max normalization
    - Mean normalization
    - Standardization
    - Unit length scaling
- 'Best' normalization method depends on the application and the data

# Variable Scaling Motivation




Goal of normalization is to make the data less skewed

# Variable Scaling Methods

Min-max normalization

- Simplest method
- Rescale variable to the range [0,1] or [-1,1] depending on which is more meaningful
- The formula to rescale a set of values to the interval [0,1]

$$x_{normalized} = \frac{x - \min(x)}{\max(x) - \min(x)}$$

# Variable Scaling Methods

Mean normalization

- Center the data around the mean
- Will *not* have unit variance

$$x_{normalized} = \frac{x - average(x)}{\max(x) - \min(x)}$$

# Variable Scaling Methods

Standardization:

- Standardization scales the data to zero mean *and* unit variance

$$x_{normalized} = \frac{x - \bar{x}}{\sigma}$$

- Where $\bar{x}$ is the average of the x values and $\sigma$ is the standard deviation
- We will revisit this in Section 2 of the course (univariate statistics)

# Variable Scaling Methods

Unit scaling

- Scale the data such that the complete vector has a length of one
- Divide each component by the Euclidian length (a.k.a. the 2-norm: $\sqrt{x^2}$)

$$x_{normalized} = \frac{x}{||x||}$$

- Note in some applications it can be better to use other norms than the 2-norm

# Common Steps to Data Preprocessing

Common things to check in your data:

1) Understand the data format
2) Field validation
3) Value validation
4) Missing data
5) Scaling
6) Dealing with categorical data

- Data often contains categorical values
  - e.g., which unit processes the batch?
- Need to ‘reencode’ the categories into numeric values
- How you handle the categorical data in the analysis depends on the problem/algorithm you use

# Categorical Data Example

| Machine | Batch Time (s) |
|---|---|
| M1 | 1501 |
| M1 | 1940 |
| M2 | 1399 |
| M3 | 2093 |
| M3 | 1899 |
| M2 | 1476 |

Integer encoding

| Machine | Batch Time (s) |
|---|---|
| 0 | 1501 |
| 0 | 1940 |
| 1 | 1399 |
| 2 | 2093 |
| 2 | 1899 |
| 1 | 1476 |

BE CAREFUL WITH INTEGER ENCODINGS – IMPLIES ORDERING IN THE SET

# Categorical Data Example

| Machine | Batch Time (s) |
|---|---|
| M1 | 1501 |
| M1 | 1940 |
| M2 | 1399 |
| M3 | 2093 |
| M3 | 1899 |
| M2 | 1476 |

Binary encoding

| Machine 1 | Machine 2 | Machine 3 | Batch Time (s) |
|---|---|---|---|
| 1 | 0 | 0 | 1501 |
| 1 | 0 | 0 | 1940 |
| 0 | 1 | 0 | 1399 |
| 0 | 0 | 1 | 2093 |
| 0 | 0 | 1 | 1899 |
| 0 | 1 | 0 | 1476 |

# Data Preprocessing Summary

- **Real data is messy –** doesn’t come with a ‘how to’ guide
- **Data cleaning is a must** – no data set arrives perfect
- It takes time to understand a new data set before you can really begin to use the data
- No two data sets are alike – **no standard data preprocessing method exists**
  - The outlined steps provide a general guideline for data preprocessing
  - Data cleaning is learned by experience – what does your data need? What are you trying to do with it?

# Now what?

- You’ve been given a data set
- You’ve done a preliminary check of the data
  - You know where measurements are coming from and what their units are
  - You’ve eliminated data points that don’t make sense and transformed some of the variables
- Now you can start exploring and analyzing the data in more detail

# References


- Best Practices in Data Cleaning – Jason W Osborne (2013)
- Bad Data Handbook – Q Ethan McCallum (2013)
- Data Wrangling with Python – Jacqueline Kazil and Katharine Jarmul (2016)