

Data Mining 2

Topic 01 : Module Introduction

Lecture 01 : Module Overview

Dr Kieran Murphy

Department of Computing and Mathematics, WIT.
(kmurphy@wit.ie)

Spring Semester, 2022

Outline

- Module motivation and aims.
- The three components of a Machine Learning Problem
- Data mining / Machine Learning workflow

What is Data Mining ?

We are drowning in data but starving for knowledge!

Necessity is the mother of invention \Rightarrow Data Mining \approx Automated analysis of massive data sets.

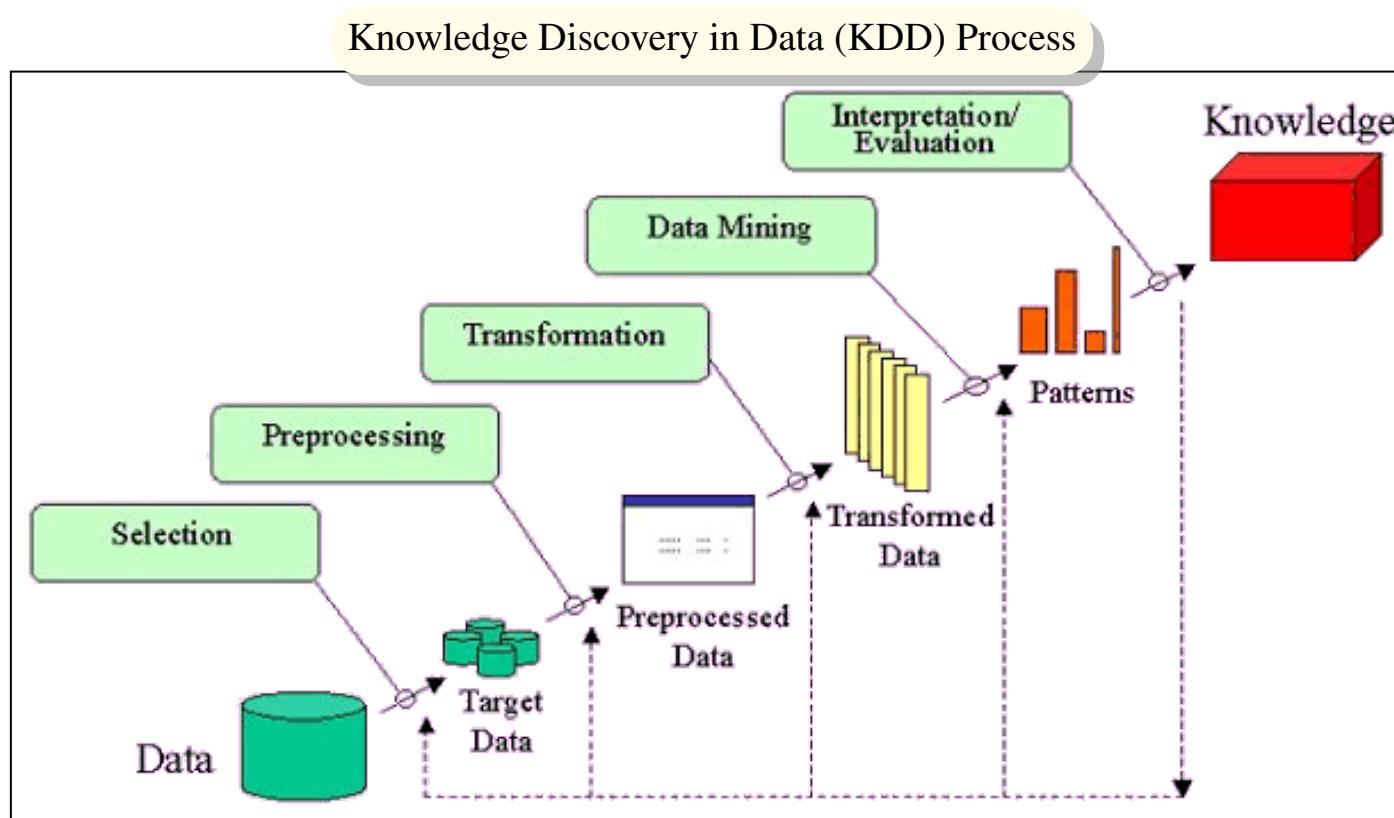
Definition 1 (Data Mining)

The **non-trivial** extraction of **implicit**, **previously unknown** and potentially **useful** knowledge from data in large data repositories

- non trivial — obvious knowledge is not useful (we already know it)
- implicit — hidden difficult to observe knowledge
- previous unknown — if known then, why go to this effort?
- potentially useful — actionable easy to understand

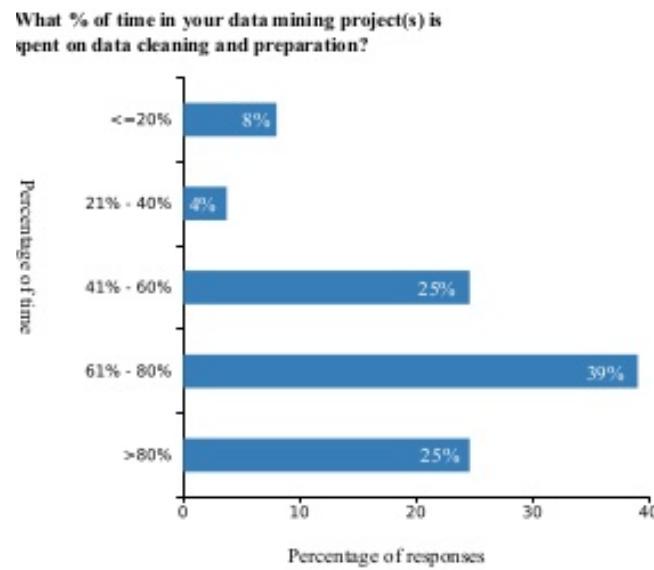
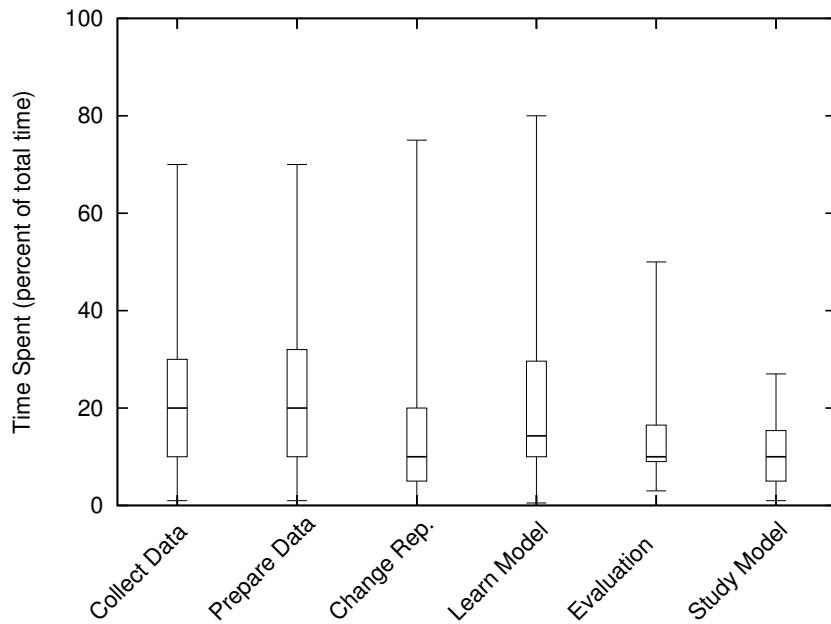
Data Mining vs Knowledge Discovery in Data (KDD)

- Data mining and KDD are often used interchangeably.
- Actually data mining is only a part of the KDD process.



See [A Comparative Study of Data Mining Process Models \(KDD, CRISP-DM and SEMMA\)](#)

Data Mining (Model Building) is less than half of Data Mining

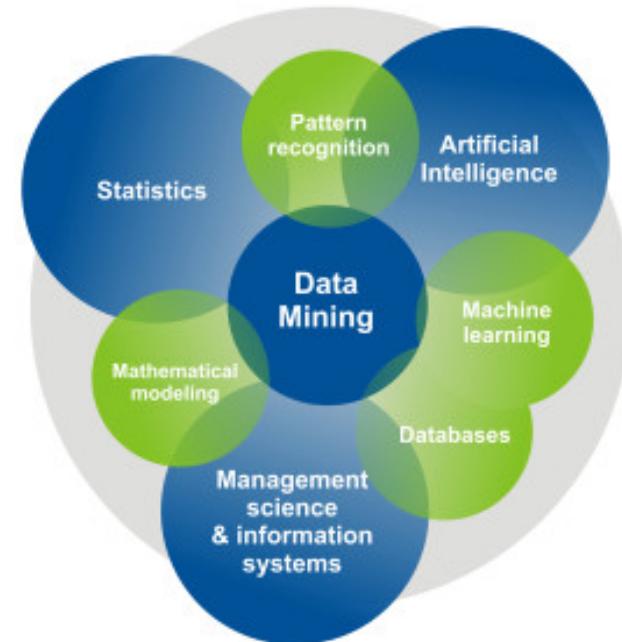
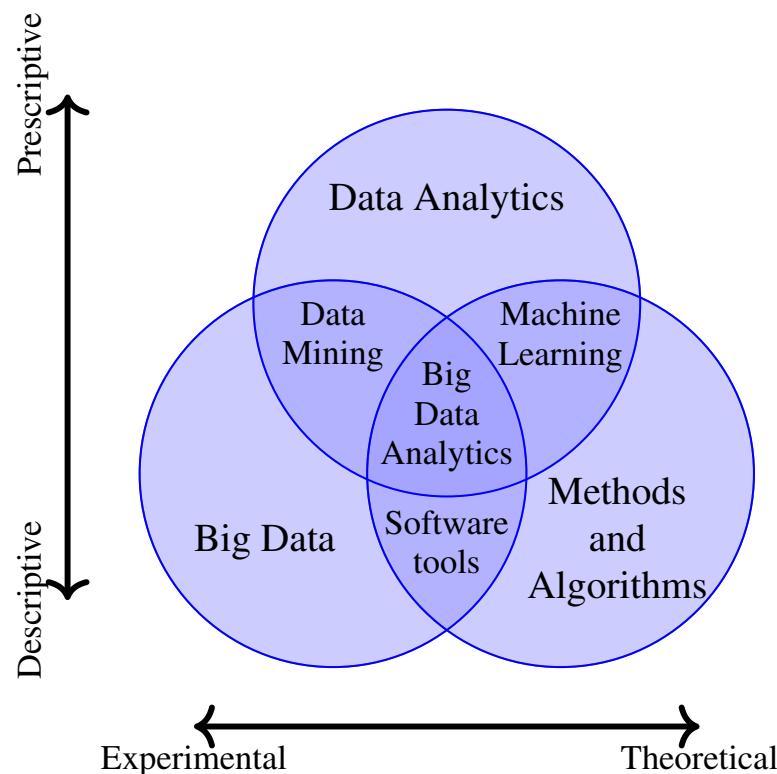


- Boxplots: median is 20% on collecting data, 20% on preparing data, and 10% on changing data representation — all before starting on model.
- Bar chart — data cleaning and preparation consumes at least 80% of project time for 25% of the participants, and 61% to 80% for another 39%.

See [Study on the Importance of and Time Spent on Different Modeling Steps, 2012](#)

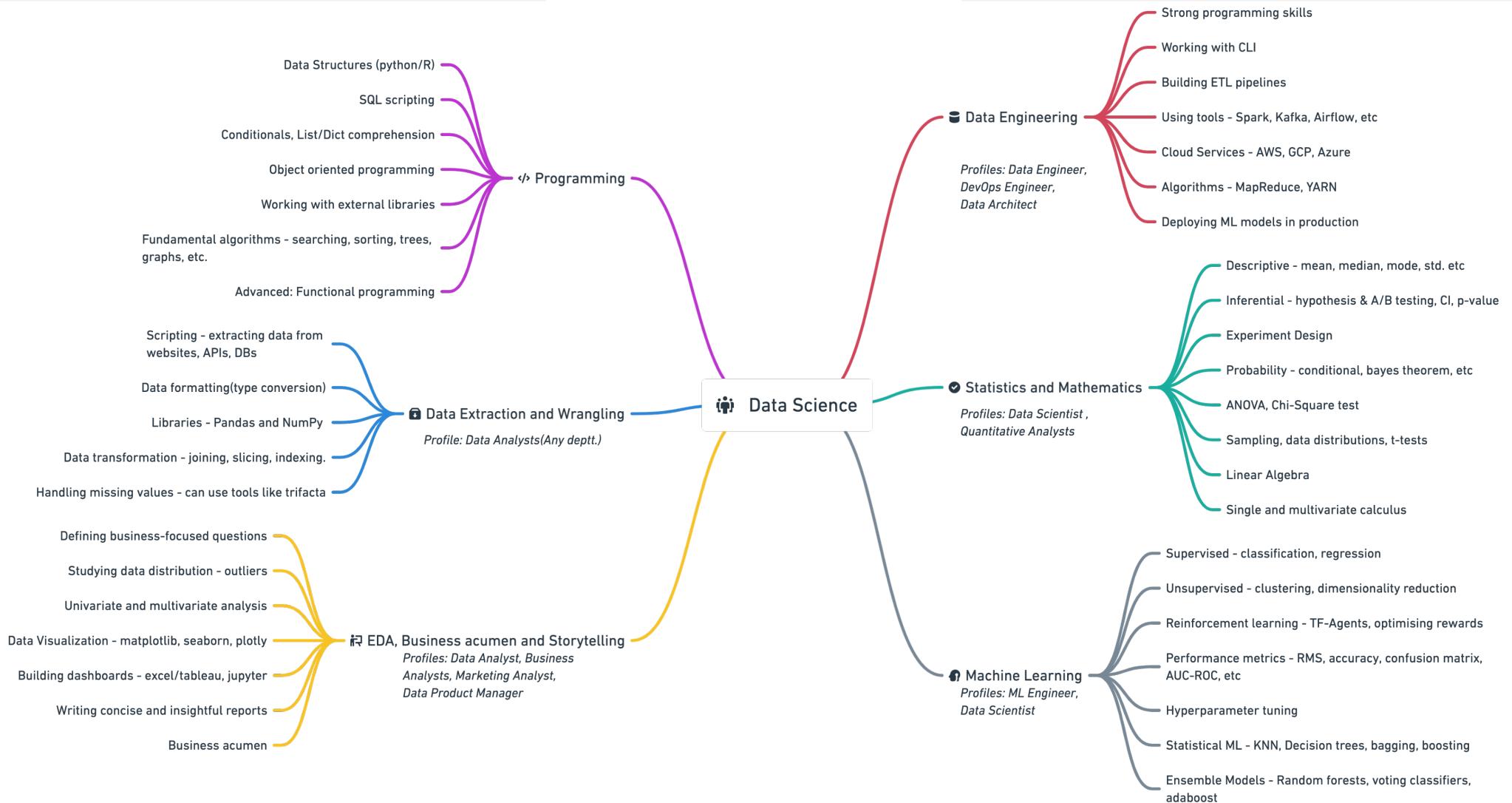
Related Disciplines — Data Mining vs Data Analytics vs Data Science[†]

- Data Mining is about finding the patterns in a data set, and using these patterns to make predictions.
- Data Science is a field of study which includes everything from Big Data Analytics, Data Mining, Predictive Modelling, Data Visualisation, Mathematics, and Statistics.



*In other words, have we titled this module correctly? Probably not, and it should be called Data Analytics 2 or Data Science 2

Data Science Mind Map



What? Why? and How? What?

Data Science in 2021

There are still some skeptics ...

The screenshot shows the homepage of mindatters.ai. The header features the "MIND MATTERS" logo in red. Below it is a large image of a brain composed of circuit boards, set against a dark blue background with glowing blue lines representing neural connections. A prominent headline reads "AI: STILL JUST CURVE FITTING, NOT FINDING A THEORY OF EVERYTHING". Below this, a quote from Judea Pearl is displayed: "The AI Feynman algorithm is impressive, as the New York Times notes, but it doesn't devise any laws of physics." At the bottom, there's a section about Judea Pearl and a link to a recent New York Times article.

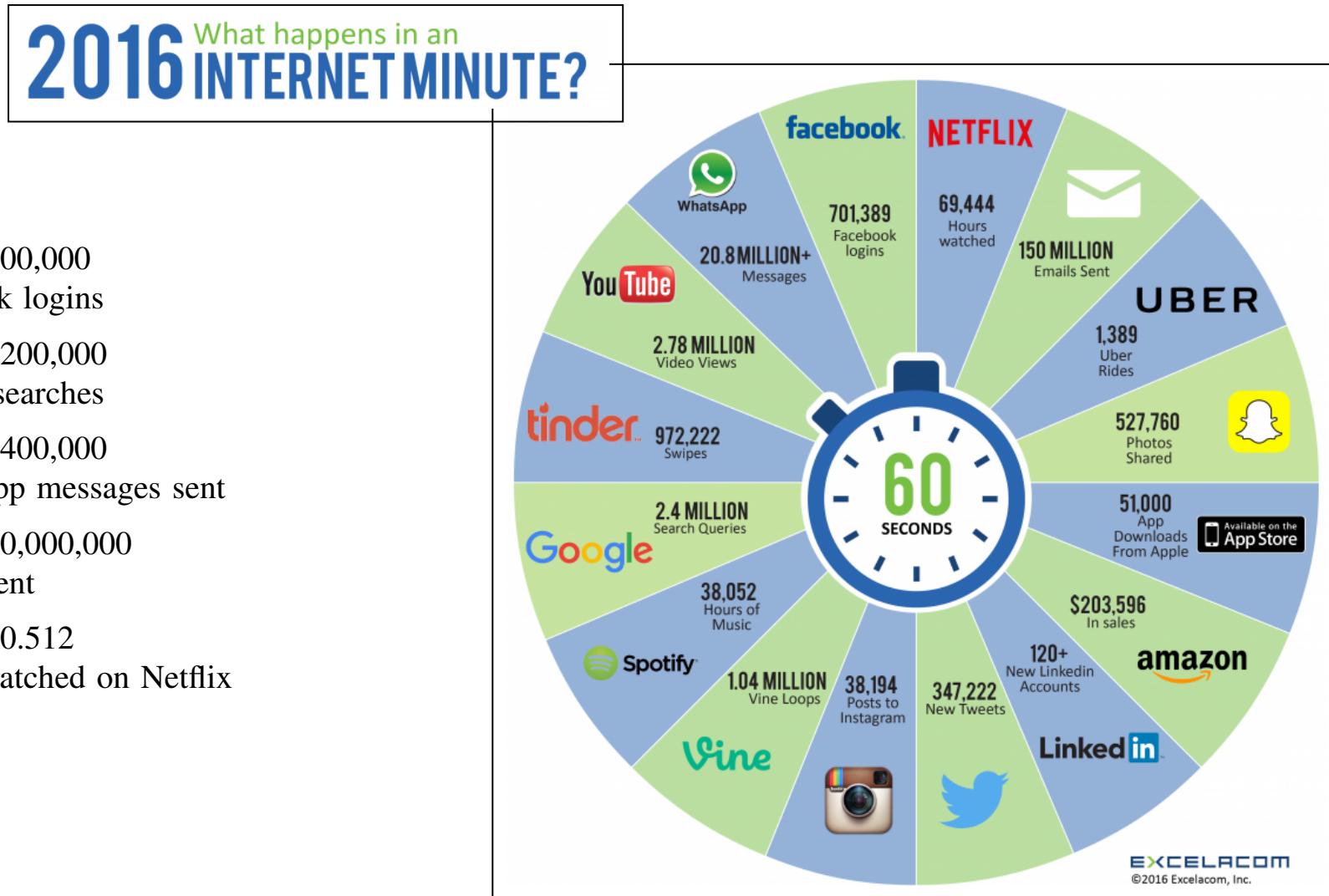
... lower barriers and models as assets ...

The screenshot shows the booste.io website. It features a large red banner with the text "Machine Learning, Without The Code" and a "Get Started" button. Below the banner is a screenshot of the booste software interface, which displays a pipeline flowchart with four stages: Data Acc., Labeling, Training, and Deployment. A specific model, YoloV3, is shown with its architecture diagram and deployment options. To the right of the interface, the text "We handle the entire ML pipeline." is followed by a list of four steps: Data Collection, Data Annotation, Model Training, and Model Deployment, each accompanied by an icon.

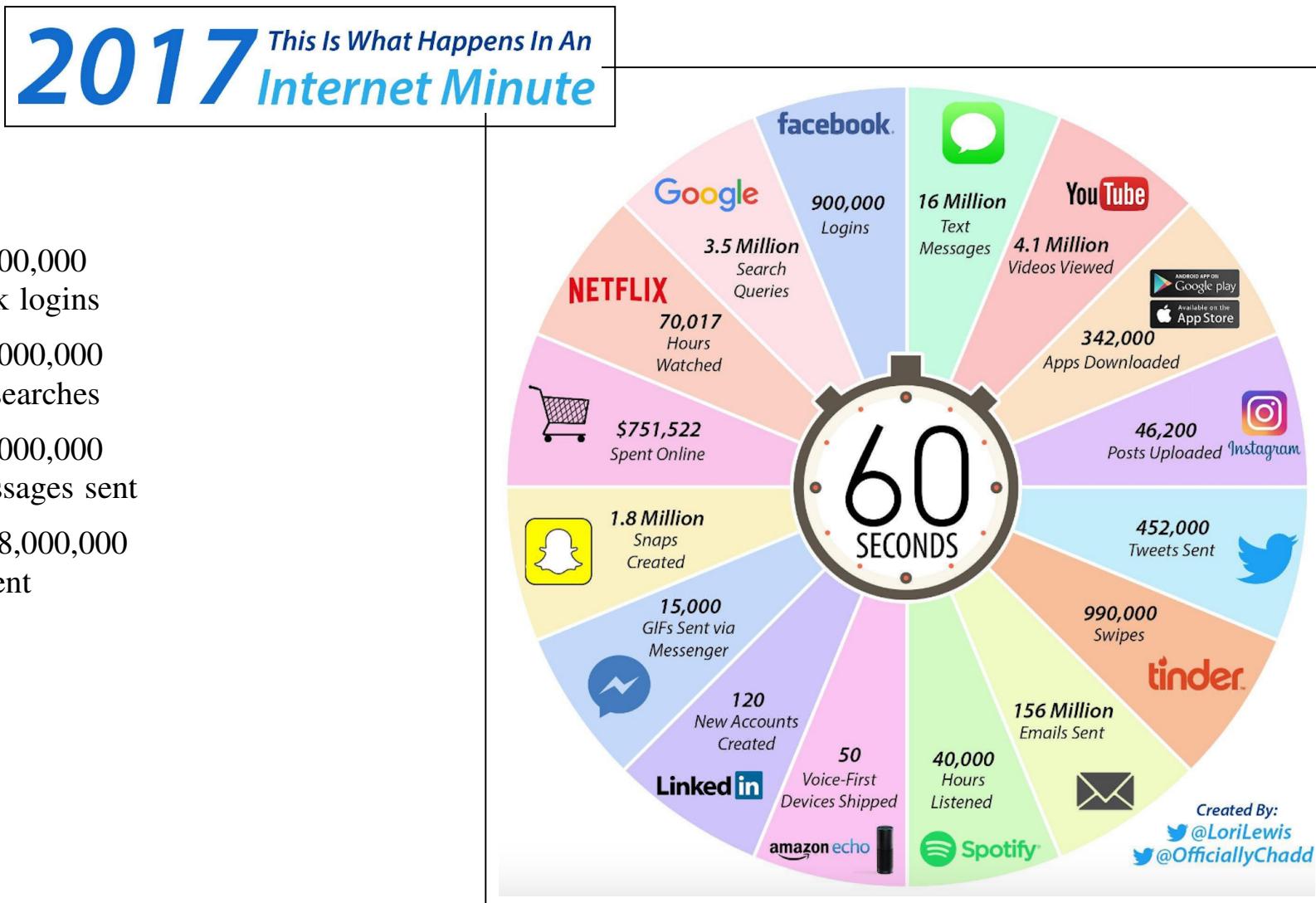
... MLOps

The screenshot shows the MLflow.org website. The header includes the MLflow logo and links for DOCS, COMMUNITY, and CODE. The main content area features a dark blue background with a blue wavy network graphic. The text "An open source platform for the machine learning lifecycle" is prominently displayed. On the right, there's a "Latest News" sidebar with links to recent releases and announcements. At the bottom, there are three icons with text: "WORKS WITH ANY ML LIBRARY, LANGUAGE & EXISTING CODE", "RUNS THE SAME WAY IN ANY CLOUD", and "DESIGNED TO SCALE FROM 1 USER TO LARGE ORGS", along with the Apache Spark logo.

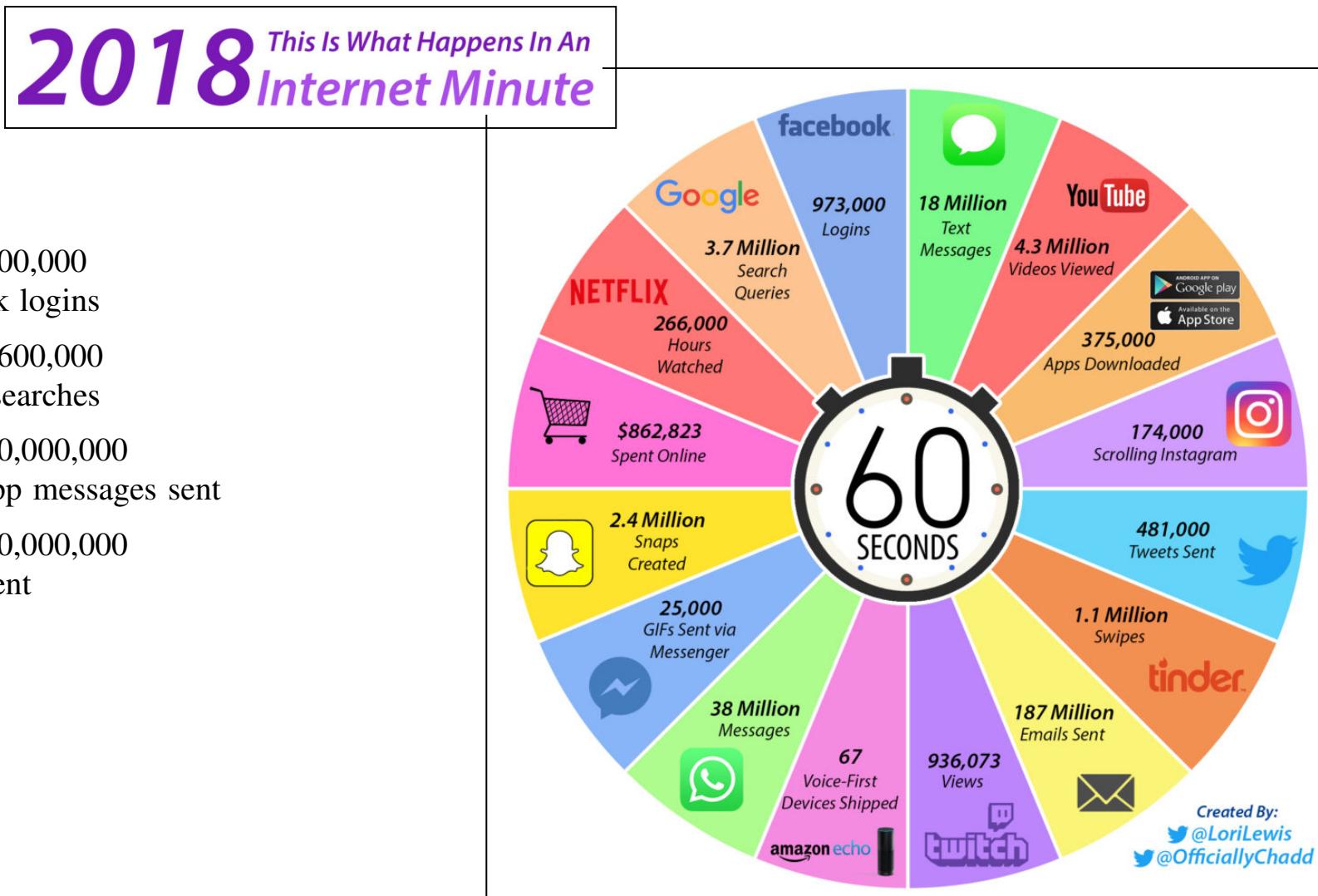
How Much Data?



How Much Data?



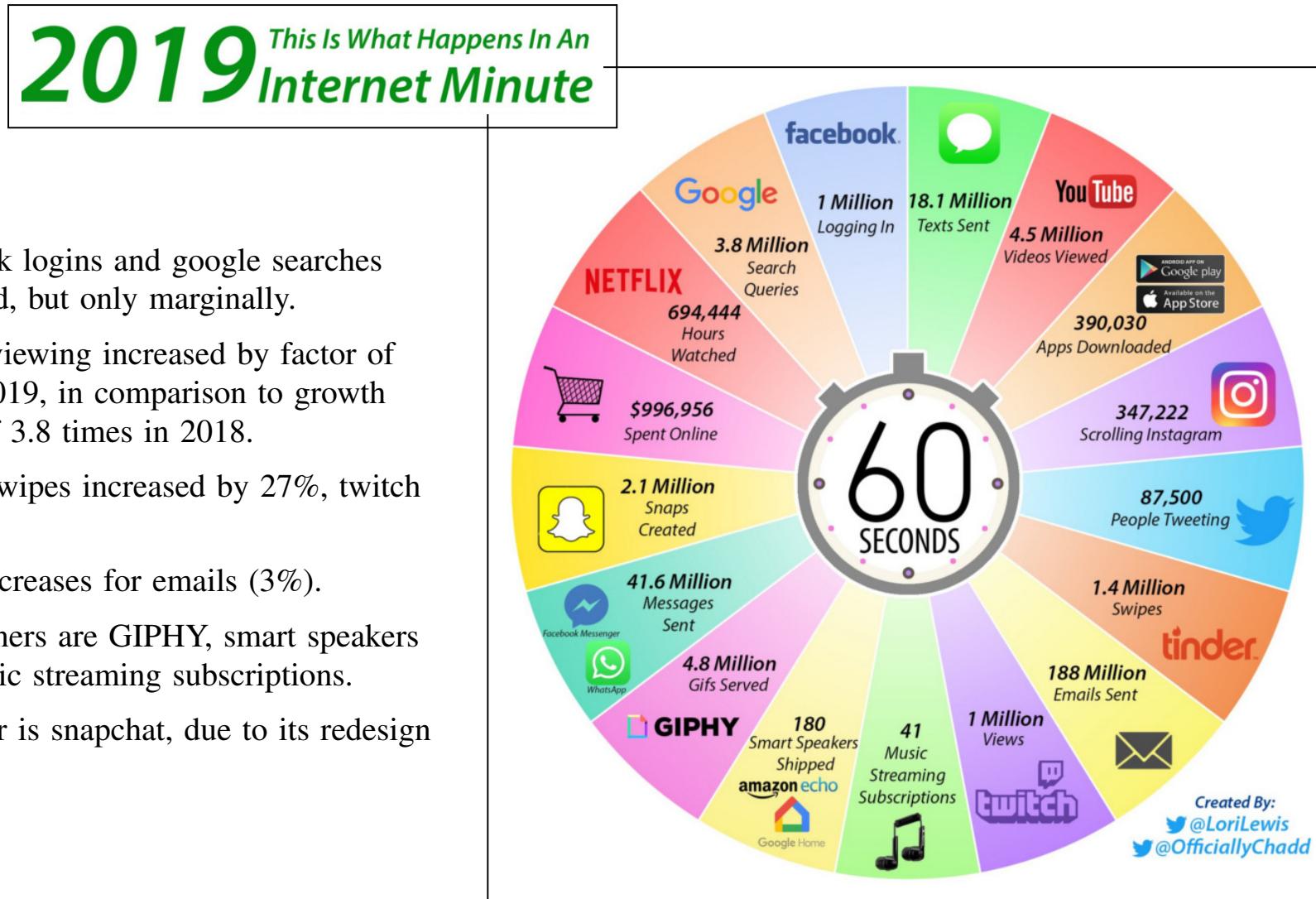
How Much Data?



By Month

- 42,033,600,000 Facebook logins
- 162,237,600,000 Google searches
- 1,641,600,000,000 WhatsApp messages sent
- 8,078,400,000,000 emails sent

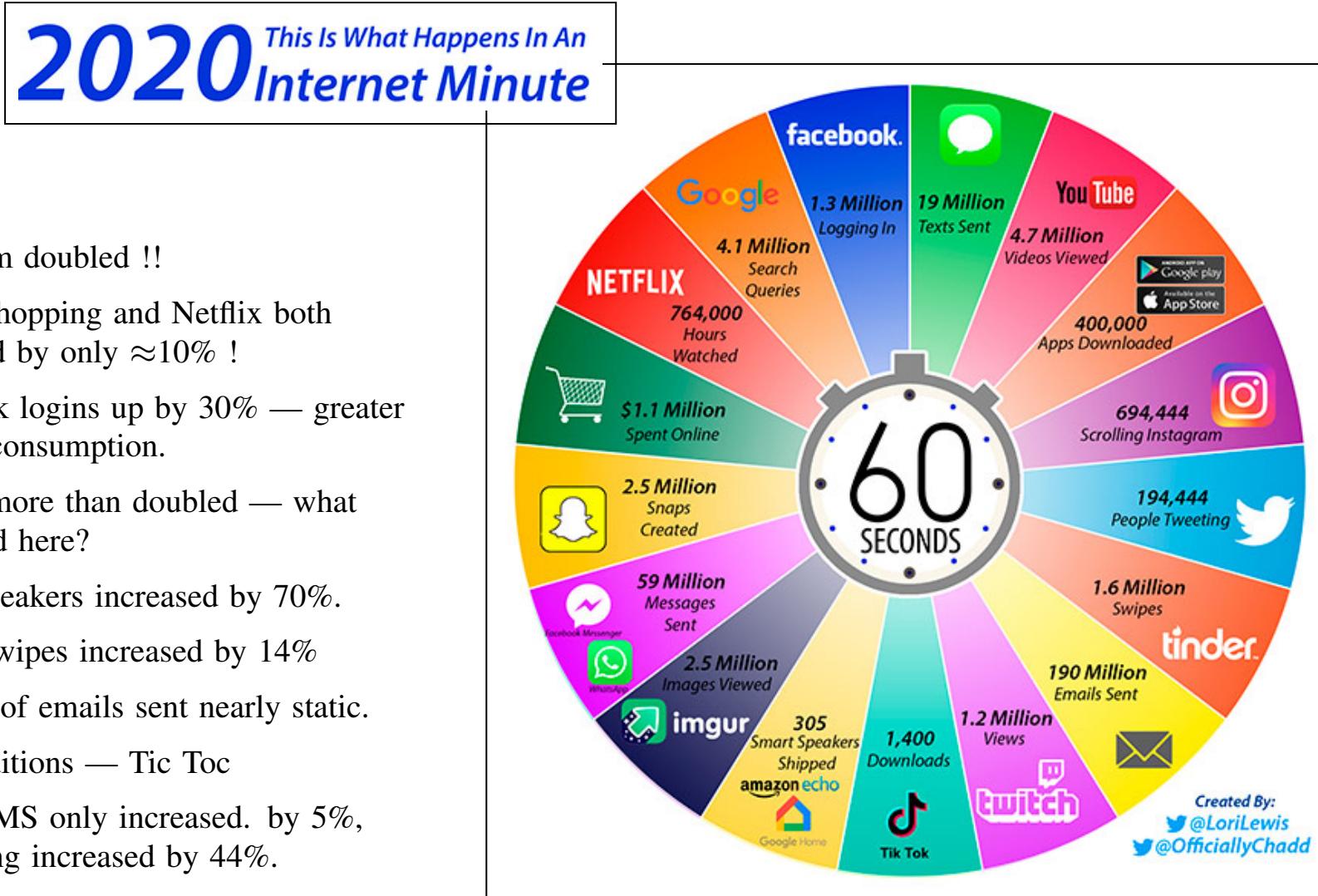
How Much Data?



- ## By Month

- Facebook logins and google searches increased, but only marginally.
 - Netflix viewing increased by factor of 2.6 in 2019, in comparison to growth factor of 3.8 times in 2018.
 - Tinder swipes increased by 27%, twitch by 20%.
 - Small increases for emails (3%).
 - Big winners are GIPHY, smart speakers and music streaming subscriptions.
 - Big loser is snapchat, due to its redesign issues.

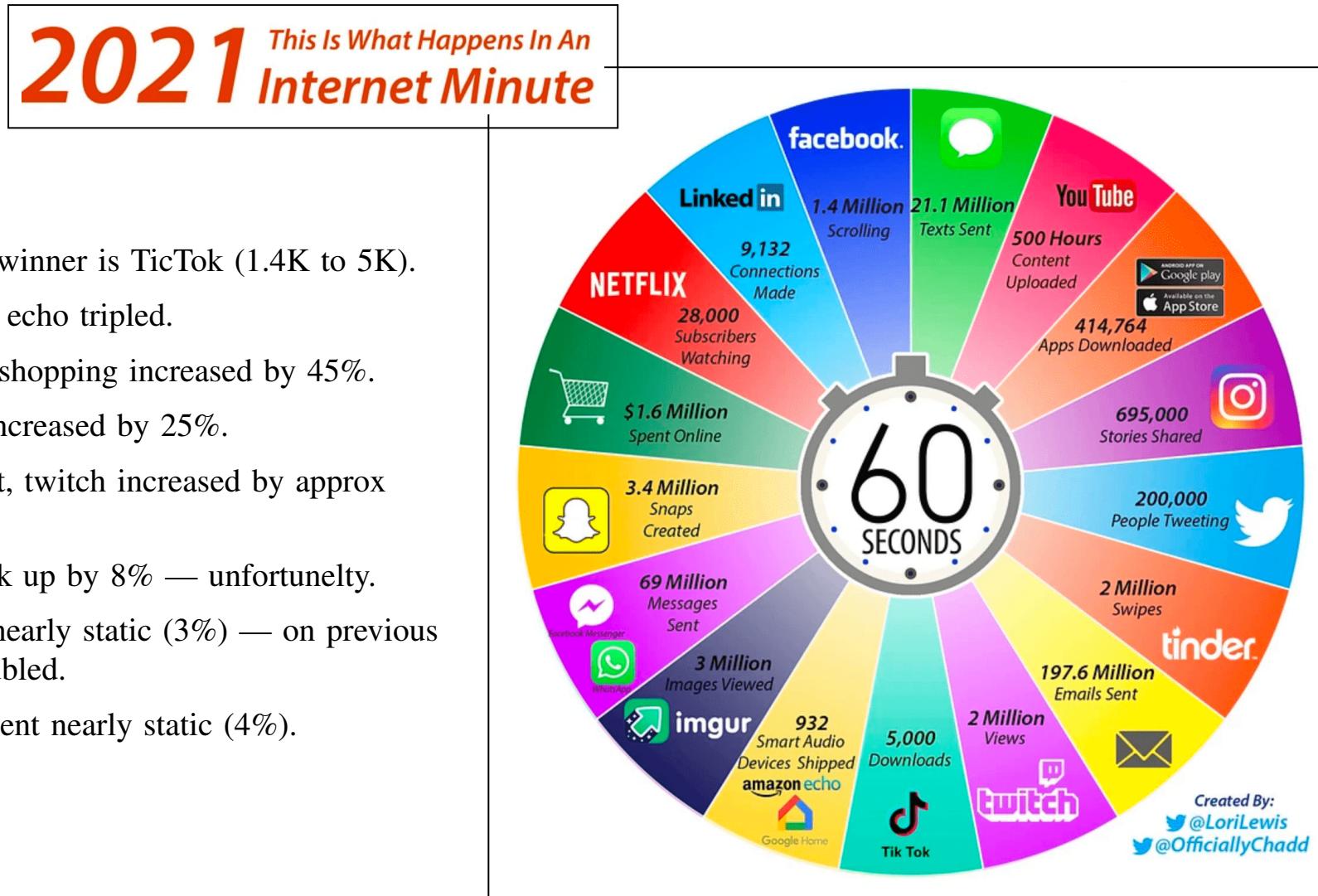
How Much Data?



By Month

- Instagram doubled !!
- Online shopping and Netflix both increased by only $\approx 10\%$!
- Facebook logins up by 30% — greater “news” consumption.
- Twitter more than doubled — what happened here?
- Smart speakers increased by 70%.
- Tinder swipes increased by 14%
- Number of emails sent nearly static.
- New additions — Tic Tac
- While SMS only increased. by 5%, messaging increased by 44%.

How Much Data?



Delivery

Resources

- All lecture slides, handouts and datasets: GitHub — datamining2-202122.github.io/live
- All activities: quizzes and assignments: Moodle — moodle.wit.ie/course/view.php?id=182862

Delivery

- Two 1-hour lectures and one 2-practical session.
 - Lecture sessions can tend to get very non-interactive so to help avoid this please ask questions.
 - Lectures and practical sessions may be recorded — in the sessions that I record I will post links in slack.
- Slack
 - Will use this for all last minute posts and individual/group Q+A, particularly for assignments.

Strategy to handle module

- Prepare — review material in advance of the sessions, install/download the software/datasets.
- Interact — yes, this is rich coming for an introvert mathication, but we live in strange times.
- Time management — give tasks a serious/focused effort, but when stuck ask for help.

Assessment Structure — 100% Continuous Assessment

Covering skills

- Data Wrangling + Feature Engineering (pandas and friends)
- NLP, Text processing (regex)
- Model building and optimisation (skilearn, tensorflow, ...)

Breakdown

- Metric:
 - 20% Student engagement + 80% Demonstration of skills/understanding
- Activities:
 - Moodle quizzes based on analysing datasets / model building / etc.
 - Data science problems with mixture of Kaggle style grading and traditional grading.

Calandar

- Week 14/15 end of semester individual review interview (zoom).
- 4 weeks + reading week + 4 weeks + Easter break (2 weeks) + 3 weeks + 3 weeks for CA
12 teaching weeks

Three Components of a Machine Learning Problem

It is easy to get lost among the multitude of choices one needs to make when given data mining problem.
A good decomposition is the following:

Representation	Evaluation	Optimization
Instances <i>K</i> -nearest neighbor Support vector machines Hyperplanes Naive Bayes Logistic regression Decision trees Sets of rules Propositional rules Logic programs Neural networks Graphical models Bayesian networks Conditional random fields	Accuracy/Error rate Precision and recall Squared error Likelihood Posterior probability Information gain K-L divergence Cost/Utility Margin	Combinatorial optimization Greedy search Beam search Branch-and-bound Continuous optimization Unconstrained Gradient descent Conjugate gradient Quasi-Newton methods Constrained Linear programming Quadratic programming

[†]A Few Useful Things to Know about Machine Learning, Domingos, 2012.

3 Components — Representation

Representation	Evaluation	Optimization
Instances K -nearest neighbor Support vector machines	Accuracy/Error rate Precision and recall Squared error	Combinatorial optimization Greedy search Beam search

Representation refers to formulating the problem as a machine learning problem — typically a classification problem, a regression problem or a clustering problem.

- How do we represent the input?
- What features to use?
- How do we learn additional features?
- With each type of problem, we have multiple subtypes.

For example which classifier? a decision tree, a neural network, a support vector machine, a hyperplane that separates the two classes etc.

3 Components — Evaluation

Representation	Evaluation	Optimization
Instances K -nearest neighbor Support vector machines	Accuracy/Error rate Precision and recall Squared error	Combinatorial optimization Greedy search Beam search

Evaluation refers to an **objective function** or a scoring function, to distinguish good models from a bad model.

- For a classification problem, we need this function to know if a given classifier is good or bad. A typical function can be based on the number of errors made by the classifier on a test set, using precision and recall.
- For a regression problem, it could be the squared error, or likelihood. Do we include regularisation? etc

3 Components — Optimisation

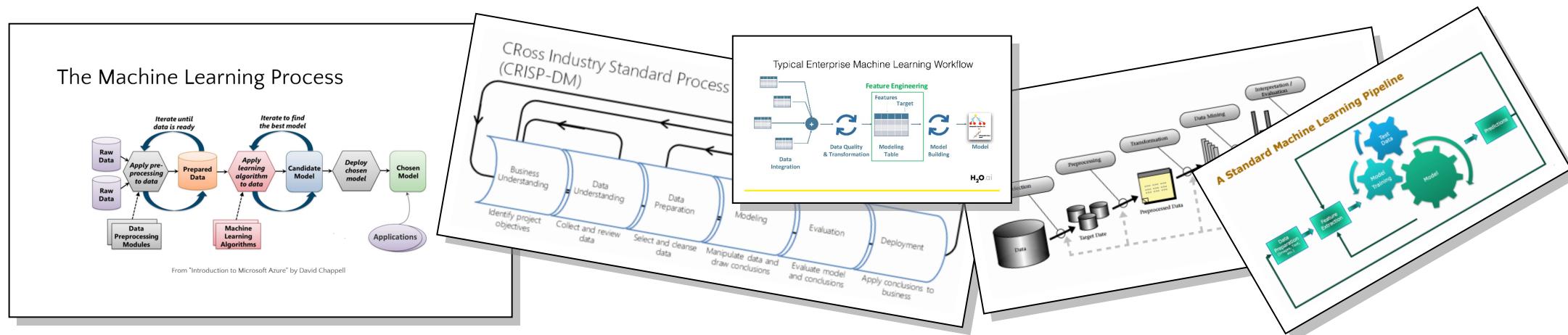
Representation	Evaluation	Optimization
Instances K -nearest neighbor Support vector machines	Accuracy/Error rate Precision and recall Squared error	Combinatorial optimization Greedy search Beam search

Optimisation is concerned with searching among the models in the language for the highest scoring model.

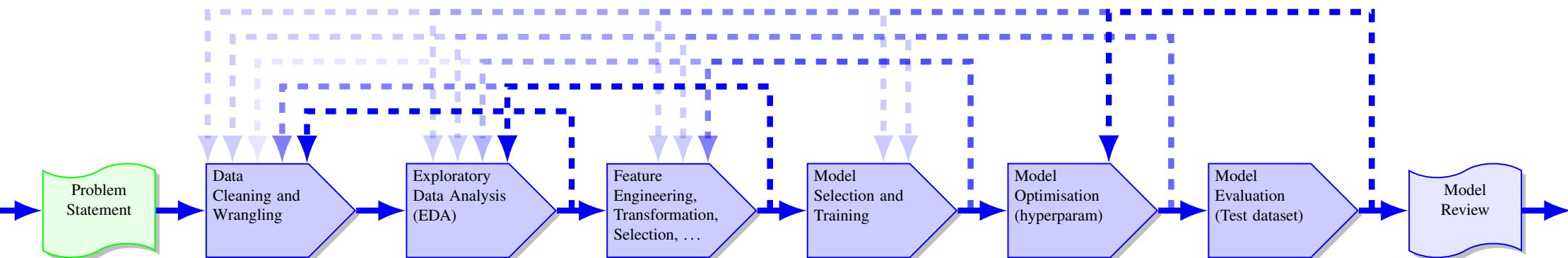
- How do we search among all the alternatives?
- Can we use some greedy approaches, branch and bound approaches, gradient descent, linear programming or quadratic programming methods.

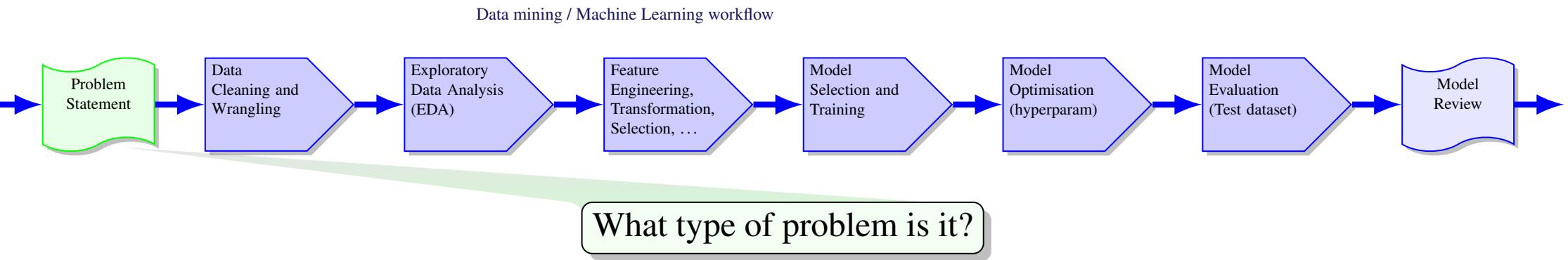
Data Mining Workflow

► There are many, many ...

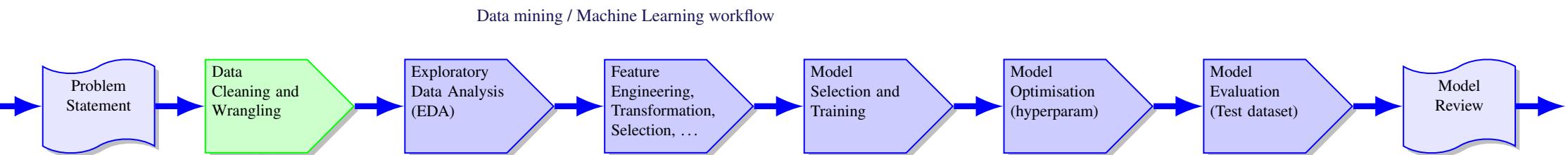


› So why not make YADMW (Yet Another Data Mining Workflow)?



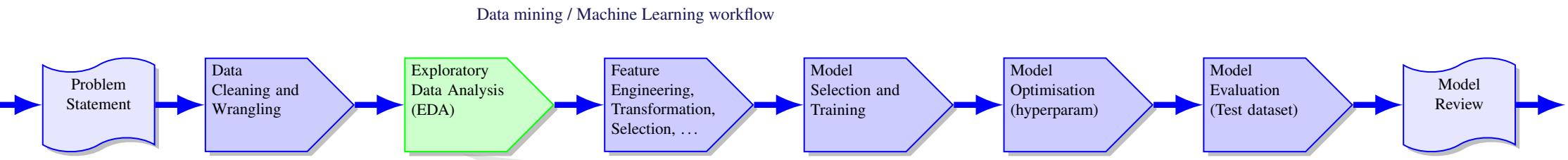


- Exploratory data analysis
Do we just want to see what the data says?
- Association / Rule finding
Are we searching for relations/patterns?
- Hypothesis testing (Statistical)
Do we have a theory we wish to test?
- Model building
Do we wish to build a representation of some pattern within the data?
 - **Supervised** ⇐ data split into input variables (**features**) and output variable(s) (**target(s)**)
 - **Classification** (target is **categorical**) vs **regression** (target is **continuous**)
 - **Unsupervised** ⇐ no target
 - **Clustering** — grouping similar cases



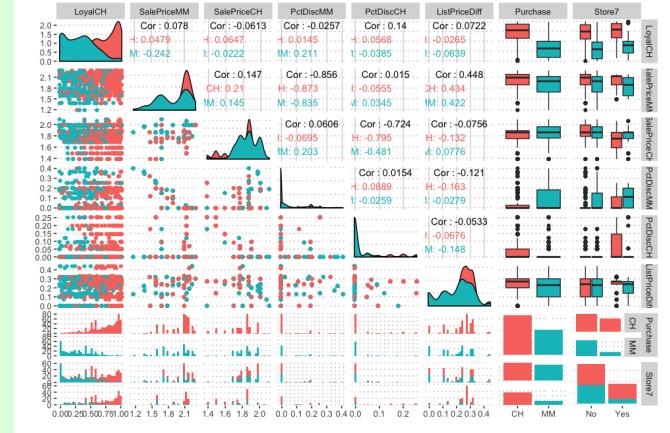
How to import and prepare data for subsequent analysis/processing?

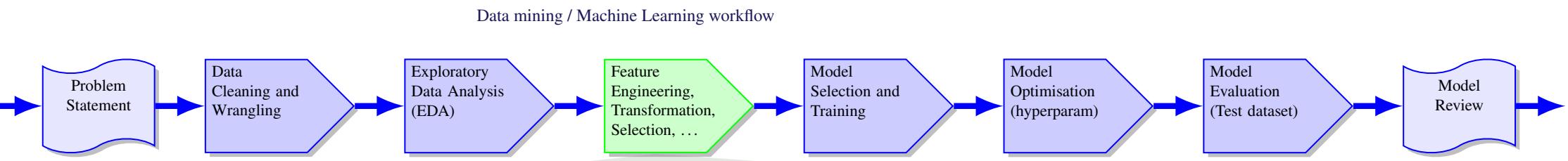
- Multiple file formats
 - Pandas supports a wide collection of file formats but default options often need to be changed to suit data.
 - Main file format (Comma Separated Values ([csv](#))) does not support meta-data, is slow, and results in large files
 ⇒ use other formats ([pickle](#), [feather](#)) to store datasets between steps in the workflow.
- Assumptions made by input parser can be important (i.e., bite you when you least expect)
 - Scientists rename human genes to stop Microsoft Excel from misreading them as dates
 - Pandas vs excel use different heuristics to decide on data type of each variable.
- Sub-tasks
 - Check dimension (number of [rows/cases](#), number of [columns/variables](#)).
 - Check data types ([categorical](#), [ordinal](#), or [numerical \(discrete/continous\)](#)) of each variable.
 - Check for missing values, encoding errors, etc.
 - Merge tables, apply filters, and general data wrangling to generate (tabular) dataset suitable for EDA.



What is the data telling us?

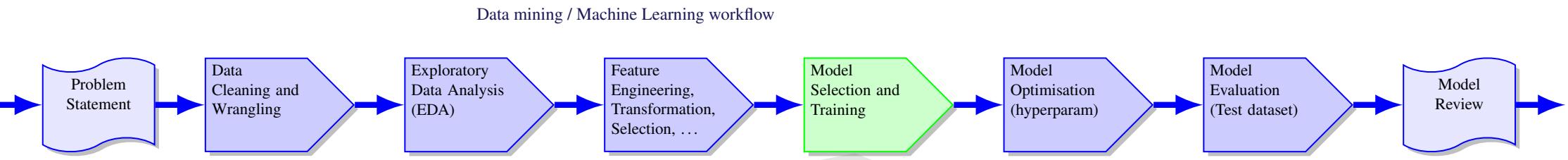
- Univariate descriptive statistics — examine each variable
 - What are typical values?
 - What is the variation / spread / range?
 - What does the data look like ...bell curve, bath tub curve, etc. ?
- Bi- / multi- variable descriptive statistics
 - Identifying relationships between variables.
- All descriptive statistics methods summarise data:
 - ✓ A summary is good since it helps to focus on simpler and important aspects.
 - ✗ A summary is bad if it focuses on irrelevant or the wrong aspects.
 - ⇒ Need to use multiple methods, be aware of their strengths/deficiencies.





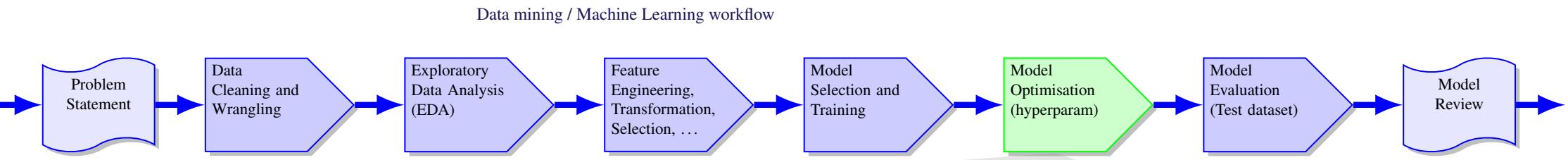
Can we transform, encode/bin, select, . . . , the given features to improve model training?

- Better features can mean:
 - Better model performance and reduce training times.
 - Simpler models become applicable — think linear/logistic regression.
 - More explainable models — the future of machine learning (hopefully).
 - Cheaper and easier models to deploy.
- Feature selection reduces the number of features used in the model:
 - Drop features that have low variability.
 - Drop features that have no relation to target.
 - Drop features that are highly related to other features — **multicollinearity**.
 - Keep features whose addition to model have the largest improvement in model score.
- Feature extraction merges existing features to generate (hopefully) fewer features with essentially all the variation.



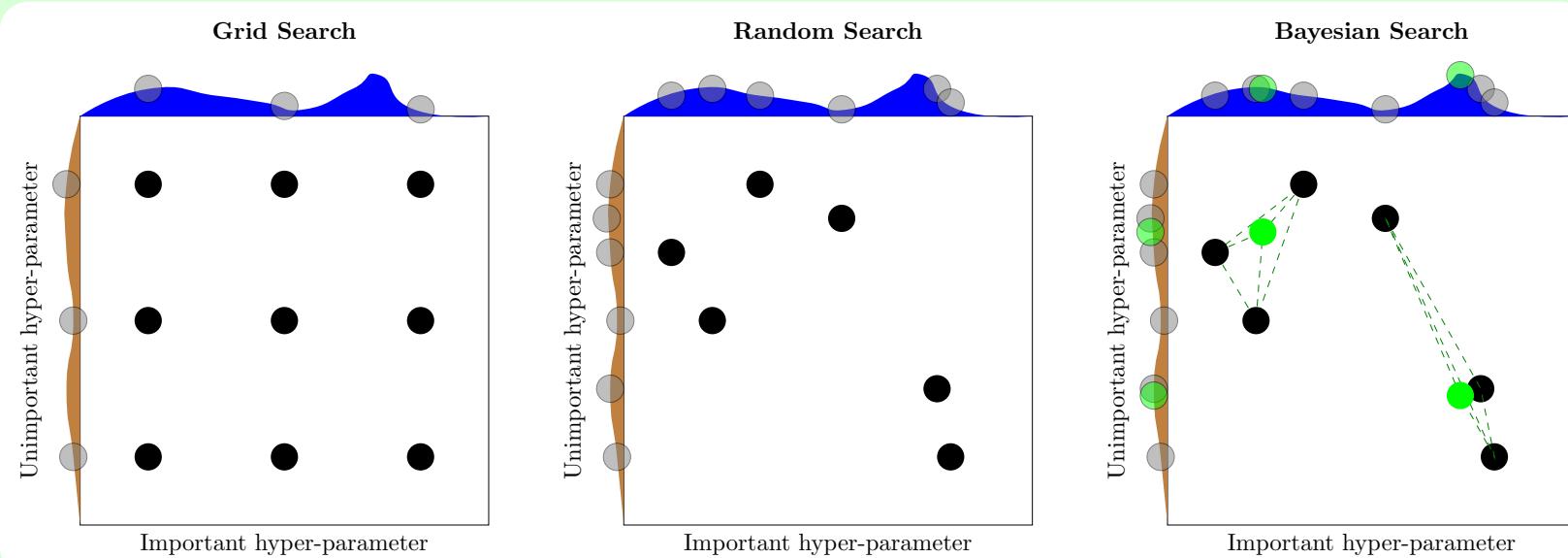
Which models are suitable?

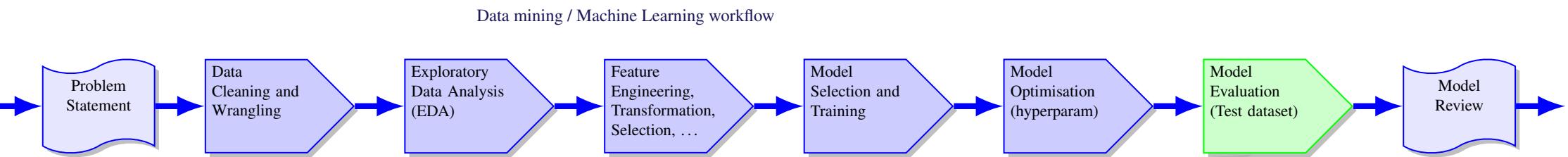
- Models vary greatly in terms of capabilities/deficiencies — usually aim to build a short list of candidate models, which are subsequently optimised in the next step.
- Select models based on different algorithms/approaches.
- Select (loss function and) evaluation metric.
 - **Loss function** is used to train model, **evaluation metric** is used to evaluate model (post training).
- Relative model performance can help identify issues with data.
 - Outliers can negatively affect linear regression but have smaller impact on decision tree based models.



How do we determine optimal values of the hyper-parameters?

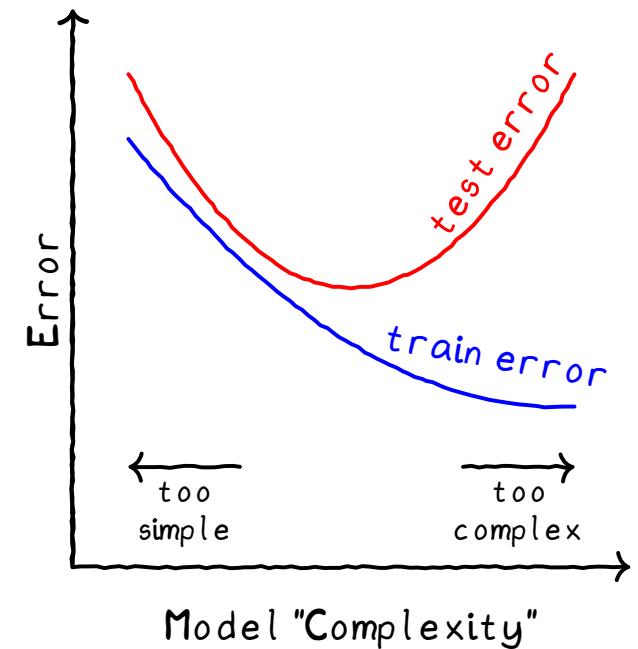
- Most models have options which control how a model “learns” from the training data.
- Three search strategies: Grid search < Random search ≪ Bayesian search

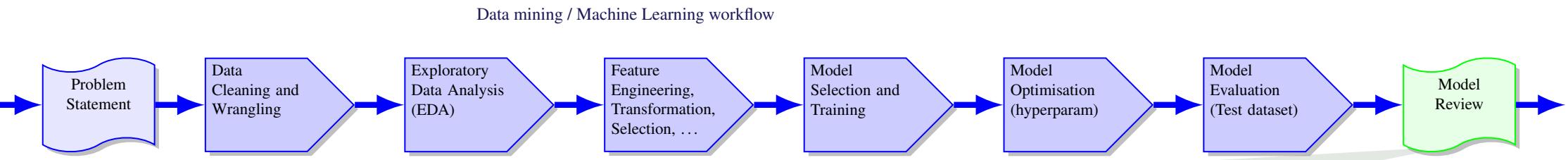




How well does the model generalise (to unseen data)?

- In the machine learning approach (vs statistical approach) we rely on model performance on **unseen data** to evaluate models.
 - Split data into train/test, only use train dataset for all modelling decisions.
 - [Data leakage \(MachineLearningMastery article\)](#), where information outside the train dataset is used in model building.
- Is there evidence for overfitting?
 - Does the model perform much better on training dataset than on the test dataset?
- Multiple techniques to address overfitting:
 - Regularisation (linear / logistic regression).
 - Trimming (decision trees).
 - Dropout (neural networks), Batch normalisation (CNN).



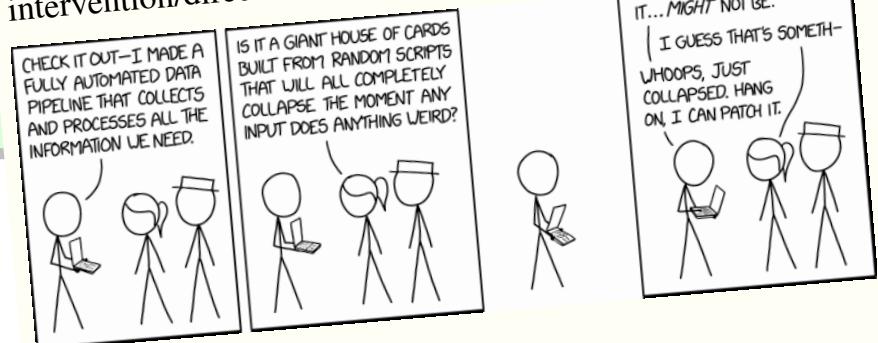


How well have we addressed the problem statement?

- At what level of **accuracy** (or other metrics) does a model become useful?
 - This is a business, medical, ... decision
 - The larger the relative payoff the weaker the model can be and still be useful.
- OK, finally ready to implement/deploy model ...
 - Separate skillset / concerns
 - MLOps = ML + DevOps
 - Monitoring of model drift needed.
- towards data science What is MLOps — Everything You Must Know to Get Started

Q: Why don't we automate all of this ~~stuff~~?

Tools are getting better and easier to use, but need intervention/direction (data can be weird in weird ways)



– xkcd.com/2054