

# 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

# 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.

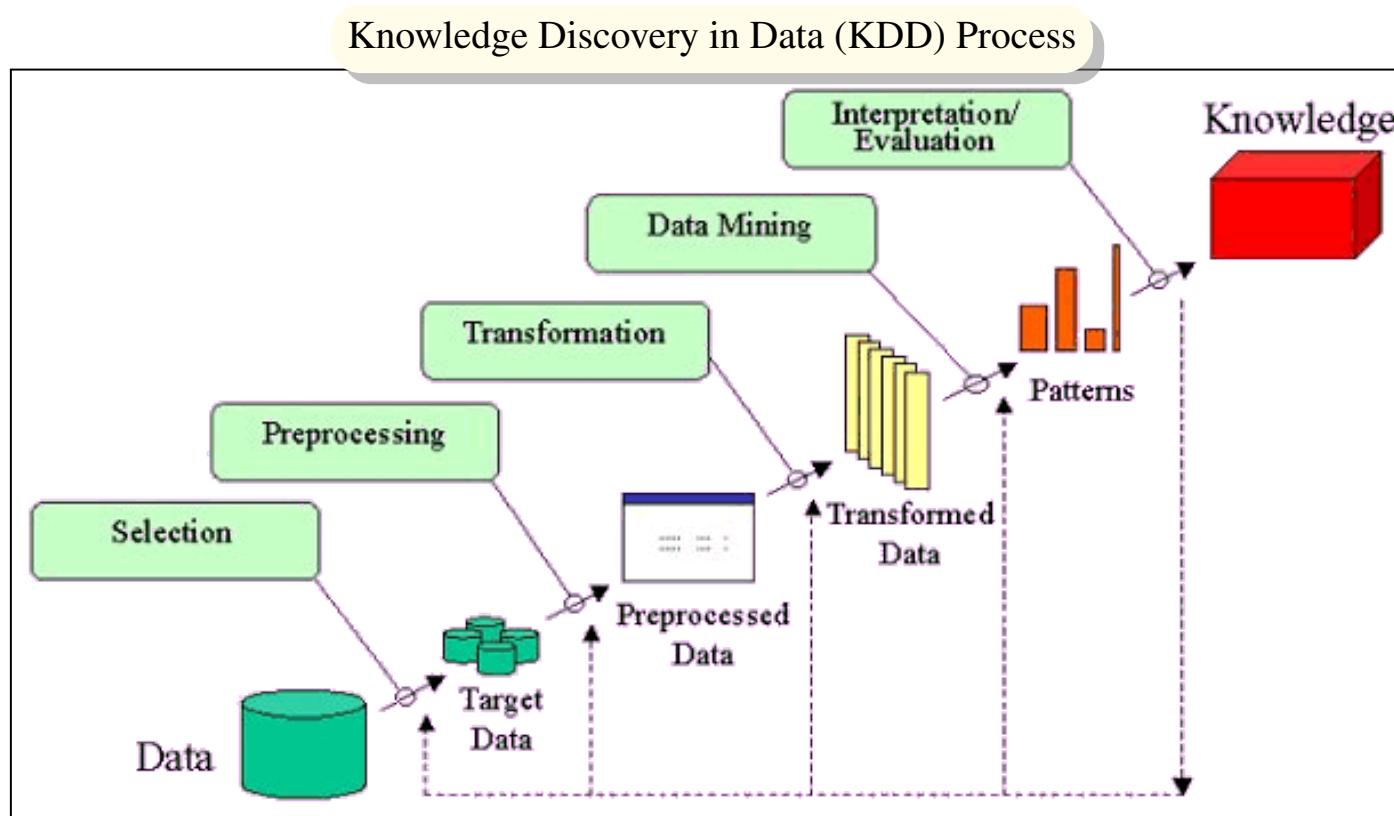
## Definition 1 (Data Mining)

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

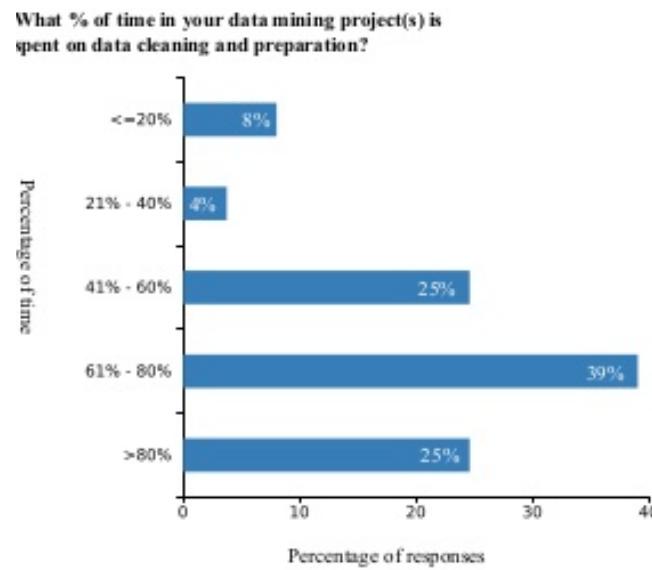
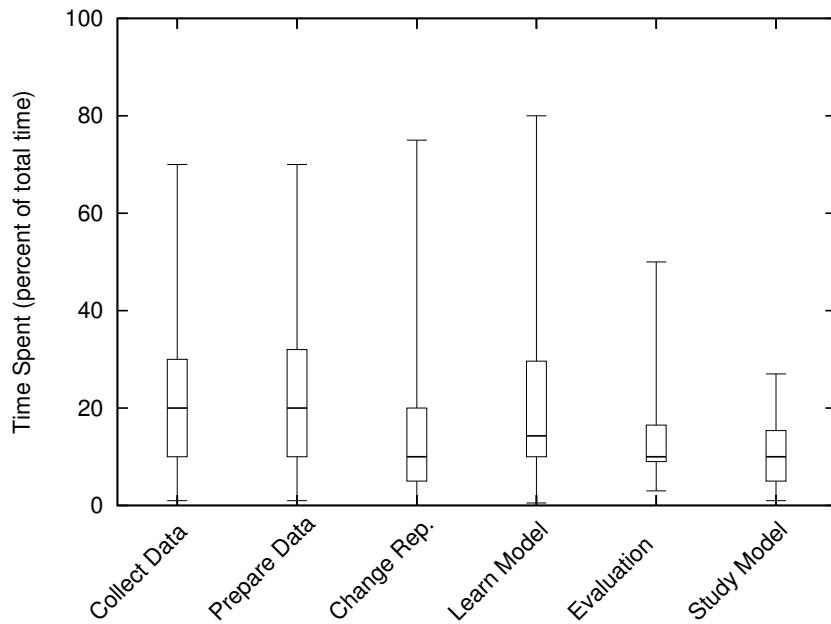
# Data Mining vs Knowledge Discovery in Data (KDD)

- 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

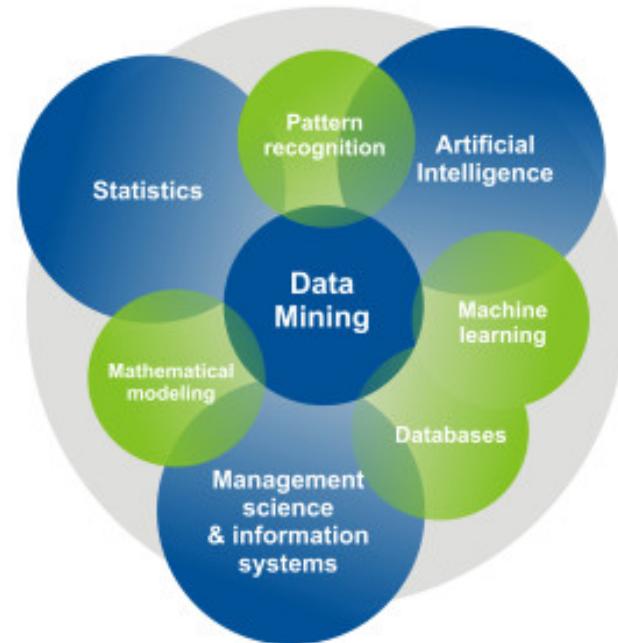
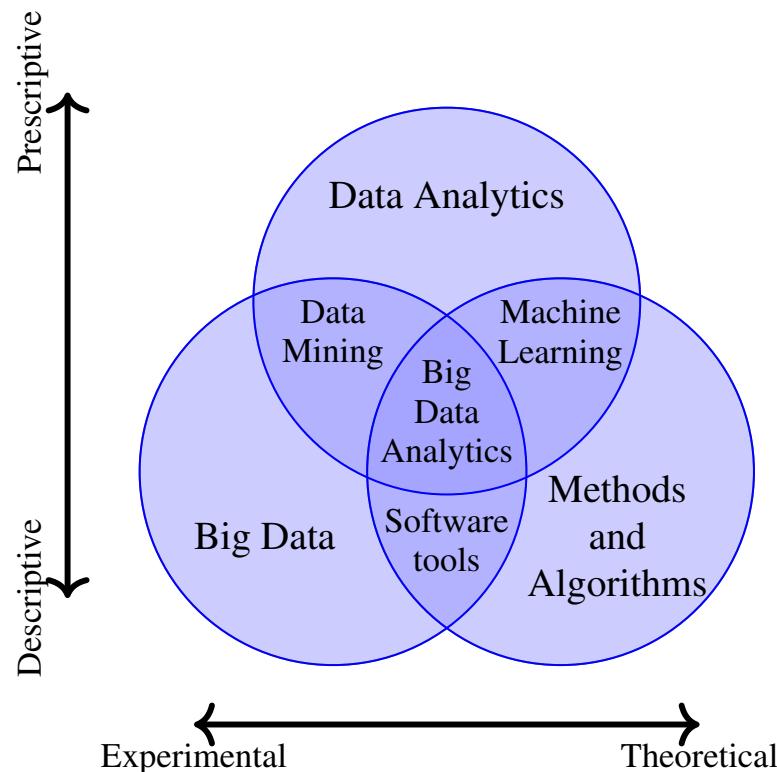


- Boxplots: median is 20% on collecting data, 20% on preparing data, and 10% on changing data representation — all before starting on model.
- Bar chart — data cleaning and preparation consumes at least 80% of project time for 25% of the participants, and 61% to 80% for another 39%.

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

# Related Disciplines — Data Mining vs Data Analytics vs Data Science<sup>†</sup>

- 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 [mindmatters.ai](https://mindmatters.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 the New York Times 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's Turing Award win and a quote from him. A "Get Started" button is visible at the bottom right.

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

The screenshot shows the [booste.io](https://booste.io) website. The main heading is "Machine Learning, Without The Code". Below it is a sub-headline: "Add custom machine learning models to your project, while hardly lifting a finger." A "Get Started" button is present. To the right, there are two windows demonstrating the platform's interface. The top window shows a configuration screen for a "YoloV3" model, including fields for "Model Type" (YoloV3, VGG, BERT, CRF-2, NMT, Fast NST), "Custom Classes", and "Training Data" (Google Images, Contractor, Upload). The bottom window shows a flowchart of the ML pipeline: Data Collection → Labeling → Training → Deployment, with a specific "Deploying YoloV3 Model To Endpoint..." step highlighted.

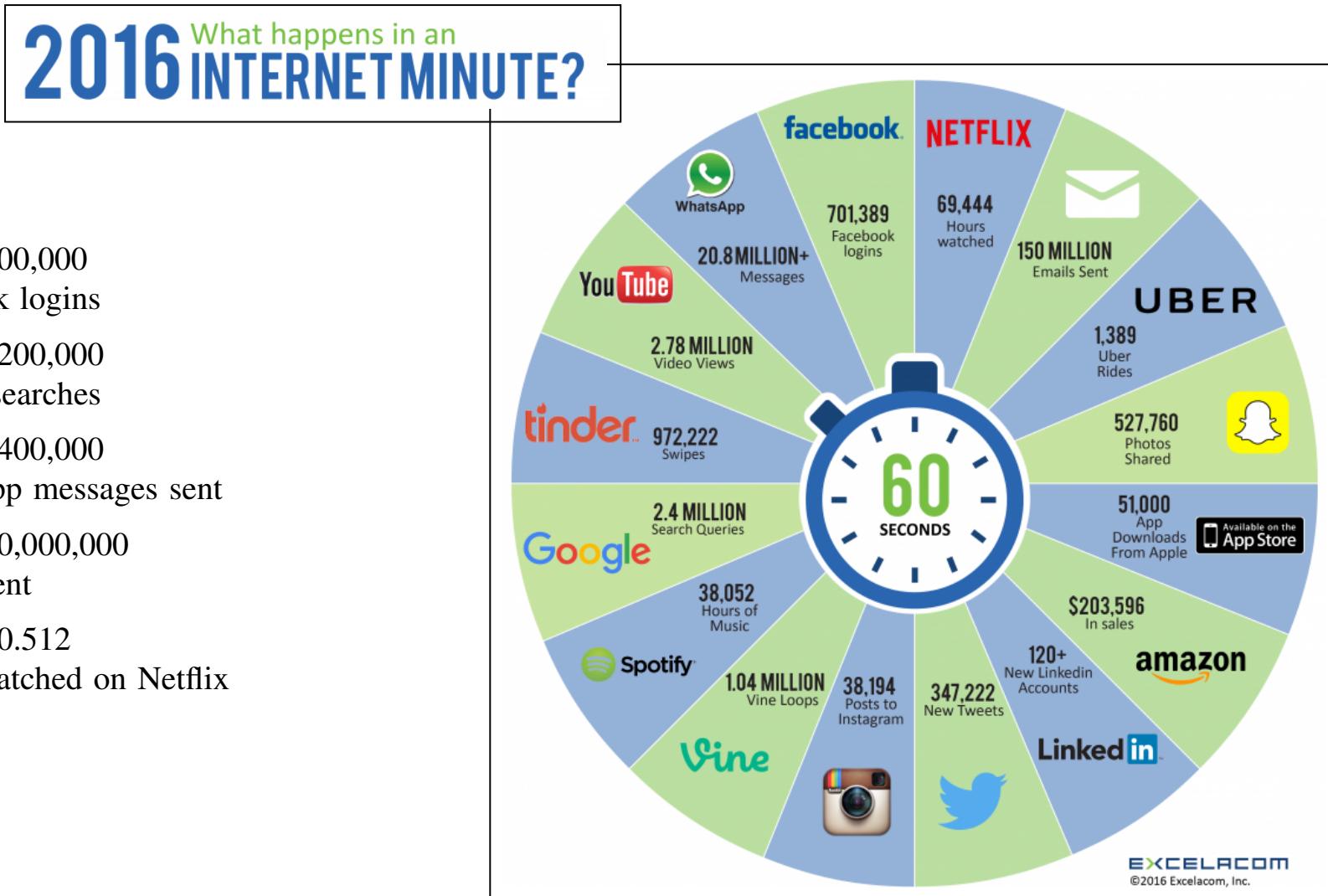
We handle the entire ML pipeline.

- Data Collection
- Data Annotation
- Model Training
- Model Deployment

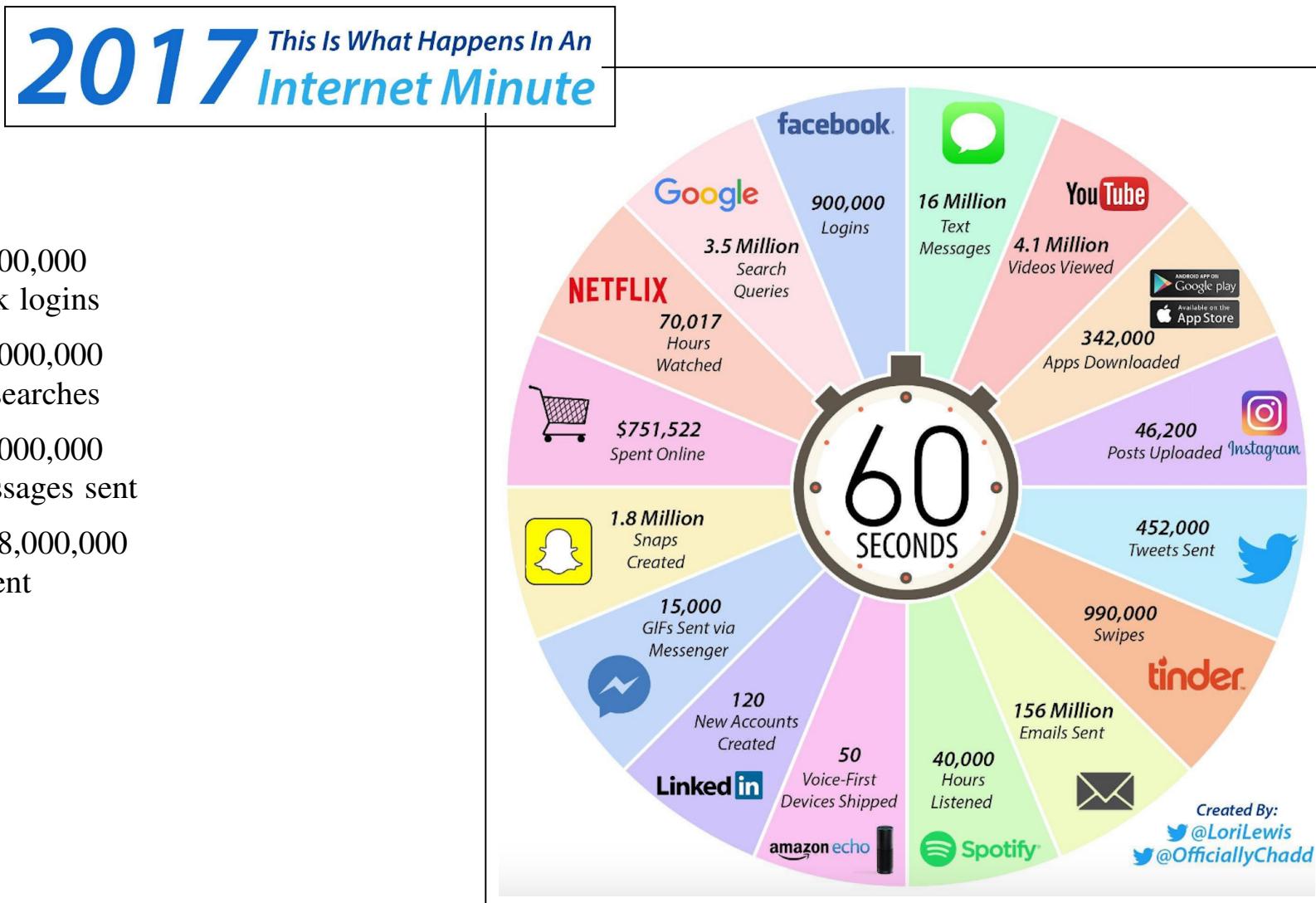
... MLOps

The screenshot shows the [MLflow](https://mlflow.org) website. The header features the "MLflow" logo. Below it is a large blue banner with a wavy dot pattern. The main headline is "An open source platform for the machine learning lifecycle". On the right, there's a "Latest News" sidebar with links to recent releases (MLflow 1.13.1, 1.13.0, 1.12.1) and an announcement about PyTorch integration. At the bottom, there are three icons: one for working with various ML libraries, one for running code in the cloud, and one for scaling to large organizations using Apache Spark.

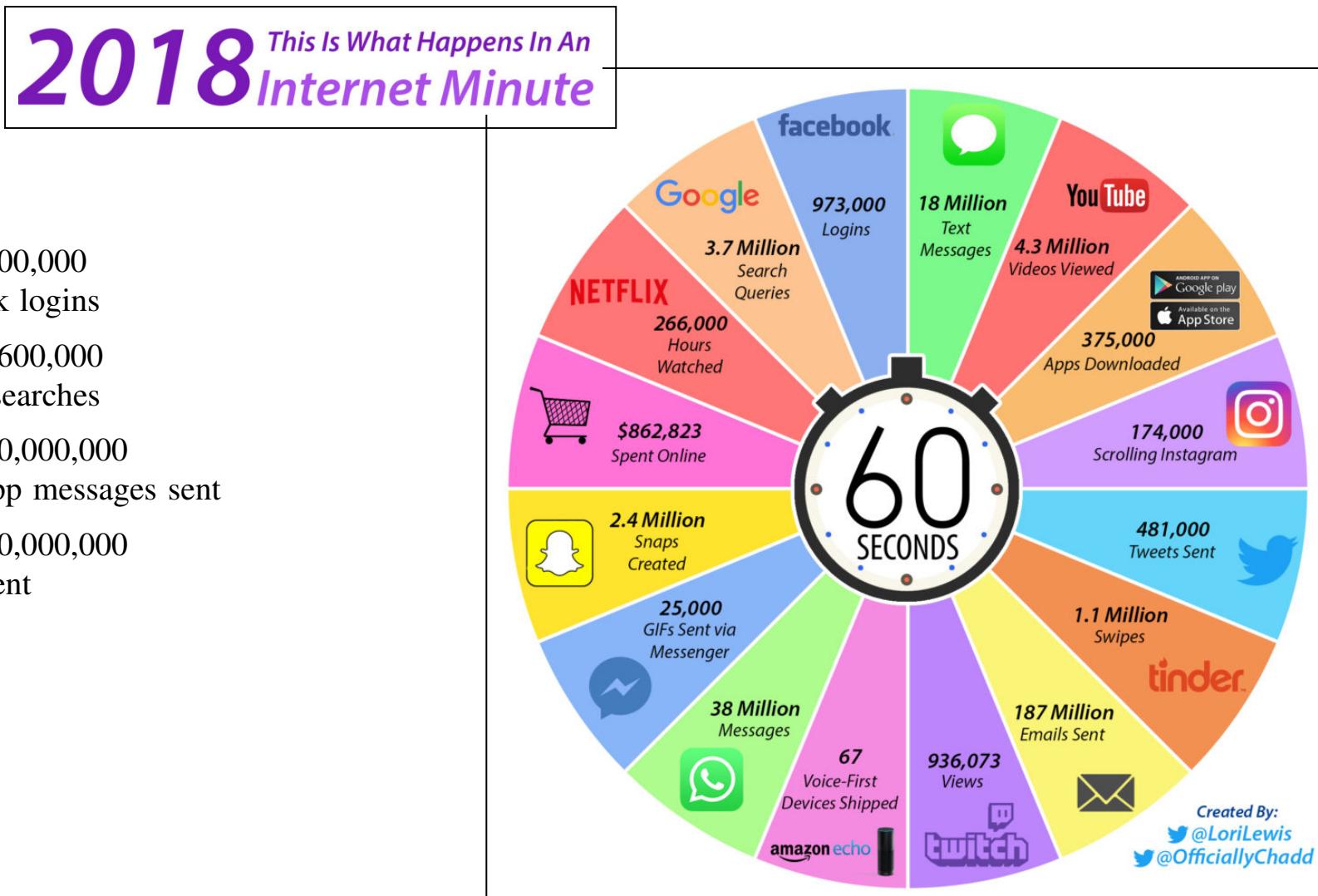
# How Much Data?



# How Much Data?



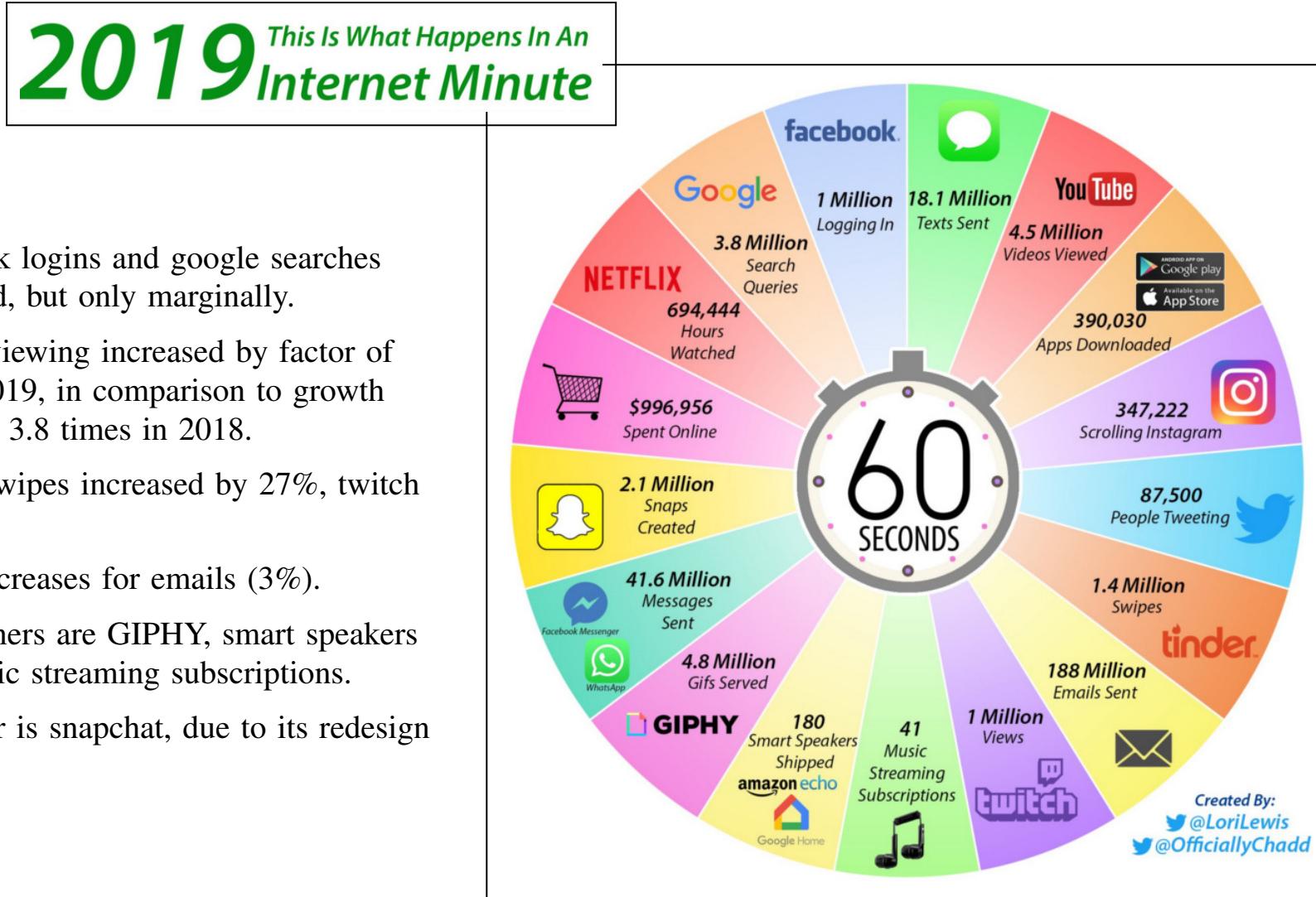
# How Much Data?



## 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

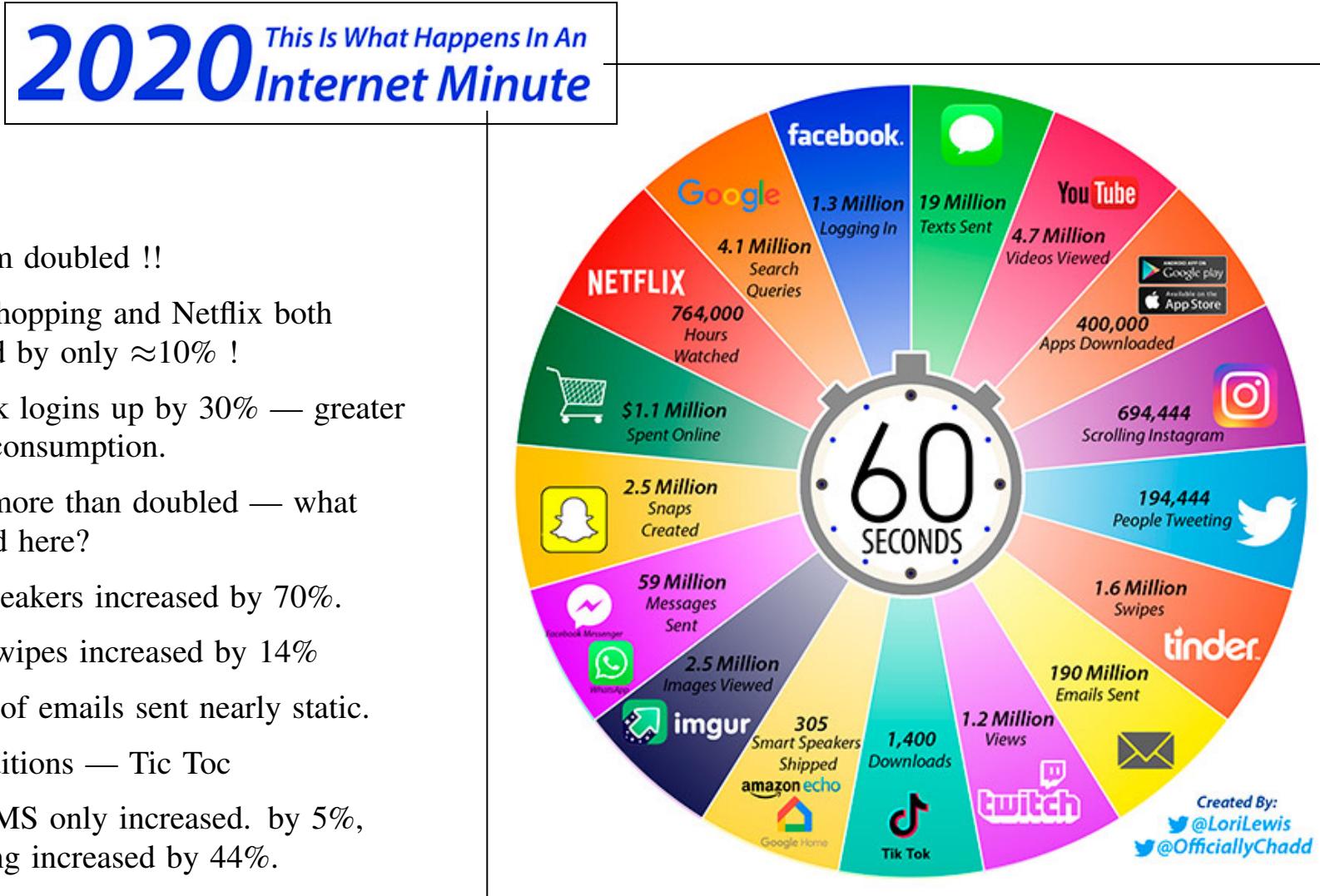
# How Much Data?



## 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.

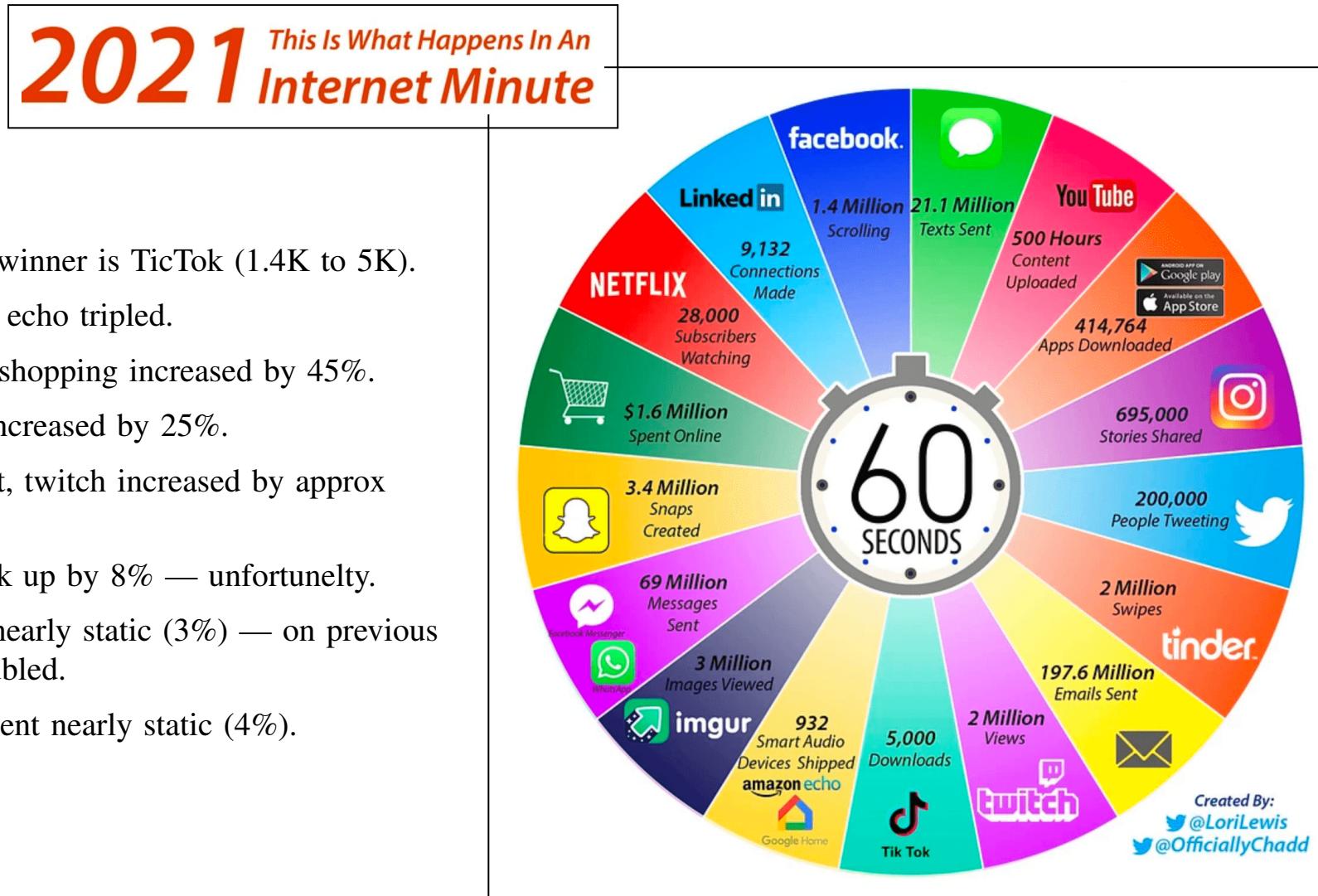
# How Much Data?



## 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?



## By Month

- Biggest winner is TicTok (1.4K to 5K).
- Amazon echo tripled.
- Internet shopping increased by 45%.
- Tinder increased by 25%.
- Snapchat, twitch increased by approx 20%.
- Facebook up by 8% — unfortunately.
- Twitter nearly static (3%) — on previous year doubled.
- Emails sent nearly static (4%).

# Delivery

## Resources

- All lecture slides, handouts and datasets: [GitHub — datamining2-202122.github.io/live](https://GitHub — datamining2-202122.github.io/live)
- All activities: quizzes and assignments: [Moodle — moodle.wit.ie/course/view.php?id=182862](https://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<br><i>K</i> -nearest neighbor<br>Support vector machines<br>Hyperplanes<br>Naive Bayes<br>Logistic regression<br>Decision trees<br>Sets of rules<br>Propositional rules<br>Logic programs<br>Neural networks<br>Graphical models<br>Bayesian networks<br>Conditional random fields | Accuracy/Error rate<br>Precision and recall<br>Squared error<br>Likelihood<br>Posterior probability<br>Information gain<br>K-L divergence<br>Cost/Utility<br>Margin | Combinatorial optimization<br>Greedy search<br>Beam search<br>Branch-and-bound<br>Continuous optimization<br>Unconstrained<br>Gradient descent<br>Conjugate gradient<br>Quasi-Newton methods<br>Constrained<br>Linear programming<br>Quadratic programming |

---

<sup>†</sup>A Few Useful Things to Know about Machine Learning, Domingos, 2012.

## 3 Components — Representation

| Representation  | Evaluation   | Optimization   |
|---|--|--|
| Instances<br>$K$ -nearest neighbor<br>Support vector machines | Accuracy/Error rate<br>Precision and recall<br>Squared error | Combinatorial optimization<br>Greedy search<br>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<br>$K$ -nearest neighbor<br>Support vector machines | Accuracy/Error rate<br>Precision and recall<br>Squared error | Combinatorial optimization<br>Greedy search<br>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<br>$K$ -nearest neighbor<br>Support vector machines | Accuracy/Error rate<br>Precision and recall<br>Squared error | Combinatorial optimization<br>Greedy search<br>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.