Applied Machine Learning - from Problem Defining to Model in Production

- Peiqing Zhang, 08.06.2022

“Well begun is half done.”

- before the well begun – not limited to ML

- after the well begun – focus on binary classification

Before the well begun – define the problem

What I have seen in the past years

projects project

"Happy ~~families~~ are all alike; every unhappy ~~family~~ is unhappy in its own way." – "*Anna Karenina*" by Leo Tolstoy

Here Tolstoy means that for a family to be happy, several key aspects must be given (such as good health of all family members, acceptable financial security, and mutual affection). If there is a deficiency in any one or more of these key aspects, the family will be unhappy. – "*The Anna Karenina principle: A concept for the explanation of success in science*" by Lutz Bornmann, Werner Marx

A health project I am going through recently…

- I sprained my ankle last July in Galdhøpiggen

Me: asked google, did nothing, still jumpped a lot

4 weeks

General doctor: checked, didn’t find anything

10 months

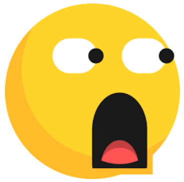
MR and X-ray

Orthopedist: called me, asked me to check it myself

physiotherapist: did thourough examinations, referred to the MR and X-ray results, revealed issues didn’t shown there but matches my painpoints, explained well, tried treatment relieved pain immediately, told me the plan

- pictures from internet

Lessons learnt here



- AAAAAAA (examed and confirmed by MR)
- BBBBBBBB (examed and confirmed by MR)
- CCCCCCCC (only shown by exam)
- DDDDDDD (only shown by X-ray)

Customers do not always know their own problems well, though they feel something is wrong:

- Thourough examination + deep insights from data

Helps with understanding the problems

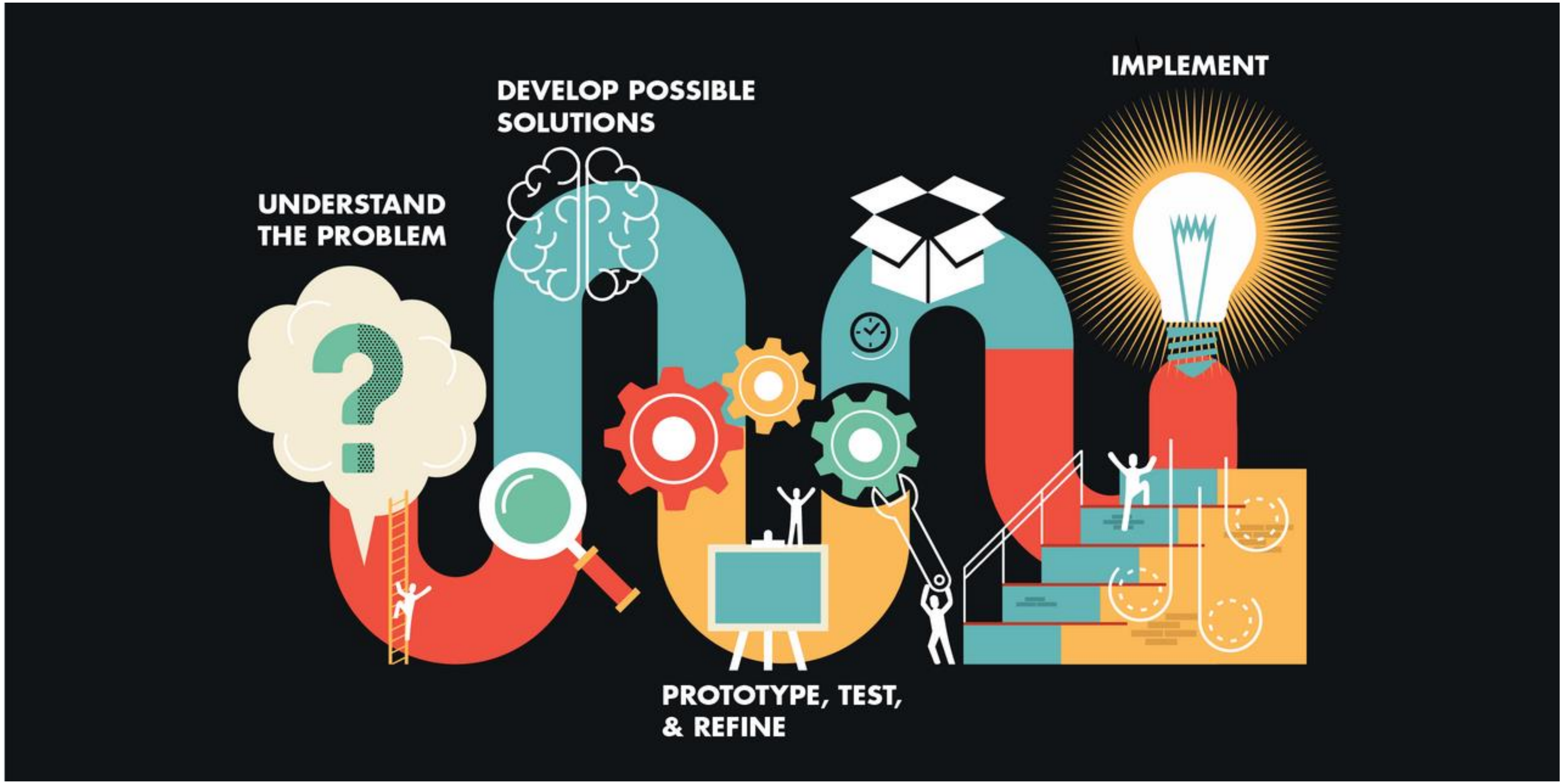
- Transparency
- Fast prototype and test
- A concrete plan

helps with building the trust

- picture from internet
- diagnosis results are masked

Design Thinking

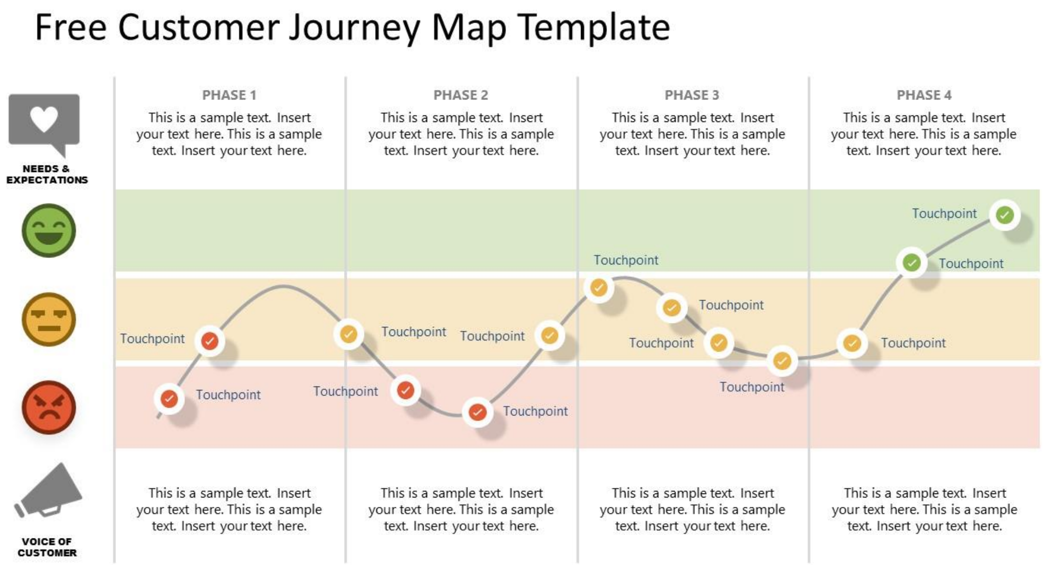
Design thinking is **a powerful approach to new product development that begins with understanding unmet customer needs**.

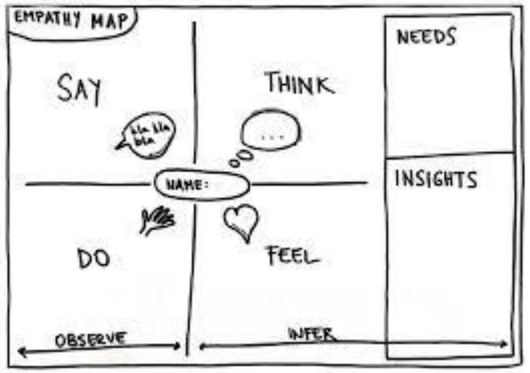


- For customers, and with customers
- Be transparent
- Make idea very clear and tangible
- Concrete and executable plan

https://executive.mit.edu/course/mastering-design-thinking/a056g00000URaa4AAD.html

Understand Problem: Reasearch and Empathy





- Interviews (ask if it is allowed to record)
- Observations - working side by side
- Directly with those who facing the issues (not managers only)
- Try to understand what the data means in the real-world

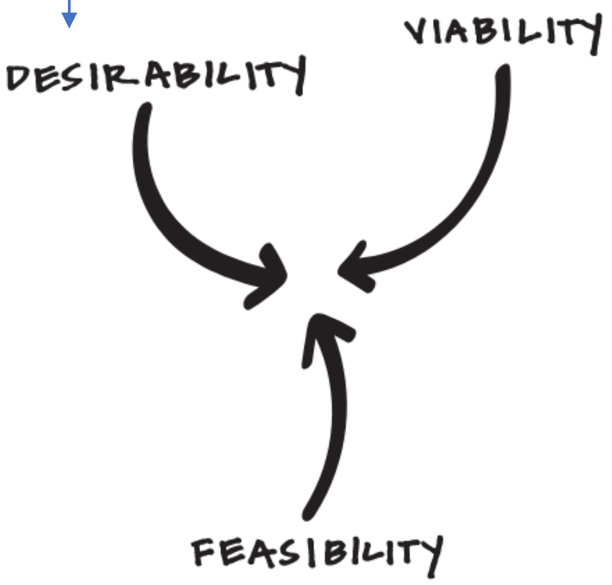
- **Data Analysis (for customers and with customers) – like X-Ray and MR!**

Develop Possible Solutions: Ideate and Selection

From research :

- what if? what we might?
- what first?

Research

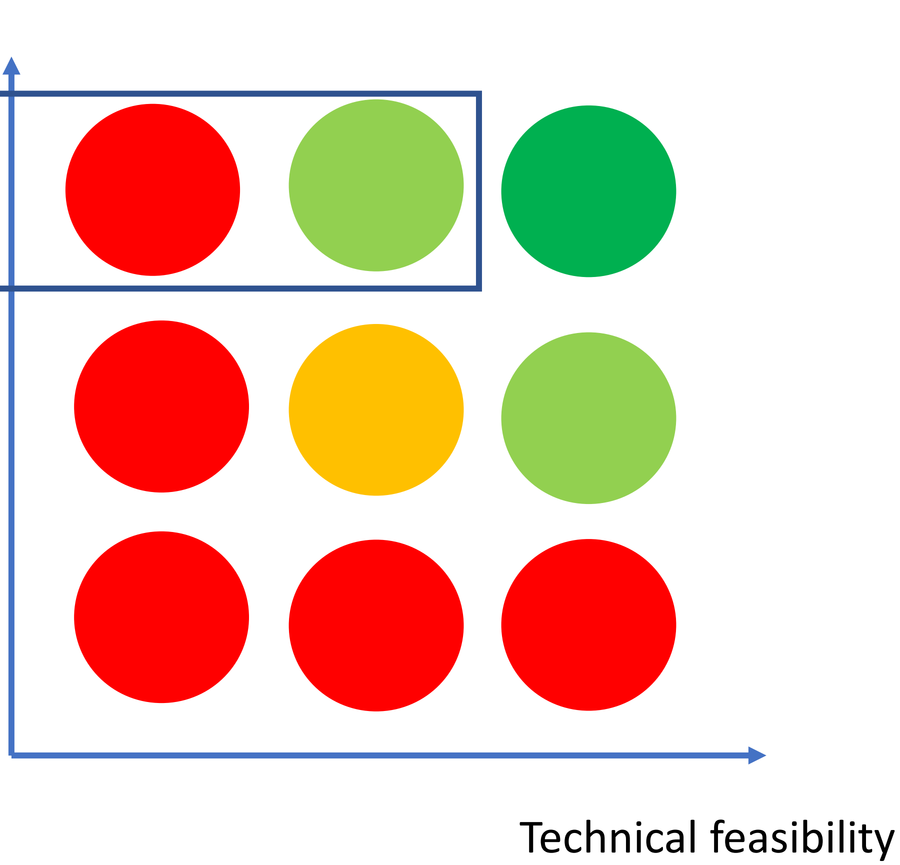


The intersection where design thinking lives
IDEO

- pictures from internet

Make (long-term) plan

as long as budget and resources fit it…



Make multiple selection matrics as needed

Develop Possible Solutions: Scoping - Size

- Goal/Objective: improve health
- KPI: BMI
- KPI Target: BMI 18.5 and 25

KPI level, with one actionable measure (e.g. sport more, or eat less)

If hard to related to KPI, perhaps need to check if the business KPIs are defined properly:
1) There can only be a few (“Key”).
2) It must matter if the numbers change (“Indicator”)
3) You need to be able to do something about it (“Performance”)

- *“Rome wasn't built in a day.”*

So does a company. All the issues will not be solved in a day.

If someone tells you a single ML model solves everything - it is a fraud.

Scoping and Prototype

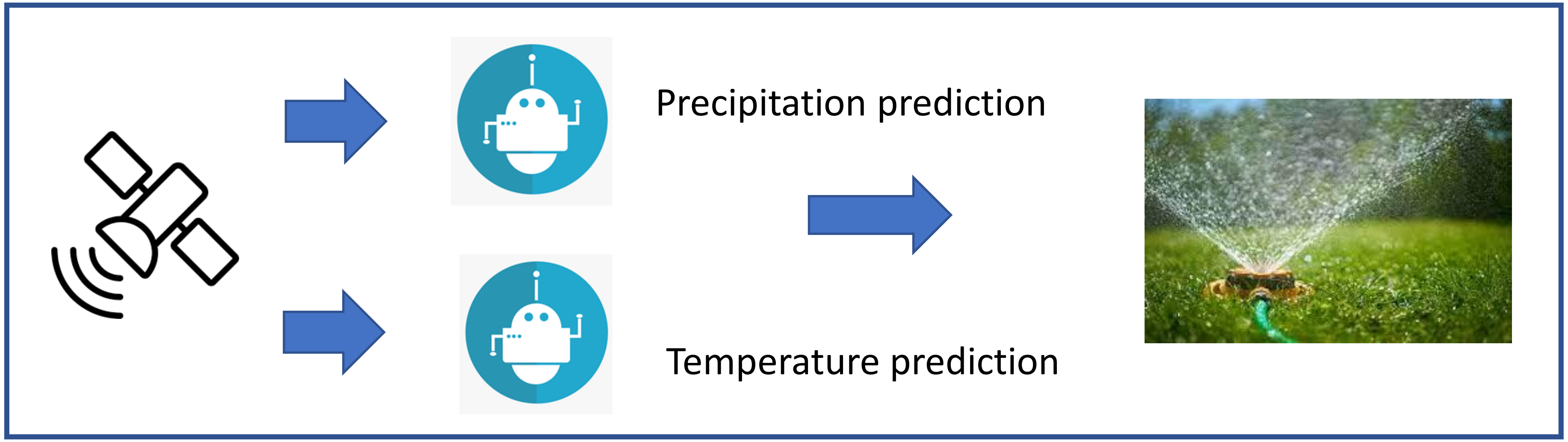
A stand-alone machine learning model doesn't solve any business issue, it is just a group of math equations, only gives:

- A probablity of in one category (e.g. the probability of rain or not)

<u>OR</u>

- A number (e.g. the temperature)

(without talking reinforcement learning and unsupervised learning here)



- pictures from internet

When scope and prototype the idea, the full end-to-end implementation shall be clean

Jump Back: Ideate

Watering the lawn

«what if…»: know precipitation in the coming week
«what we might»: predict the precipitation in the coming week

Why frame ML model(s) into possible solution:

- Actions can be taken based on the model's output, i.e. the cure to the pain point can be mapped on knowing an estimated
    - probablity, or
    - number
- In reality: there can be different ways to map the problem

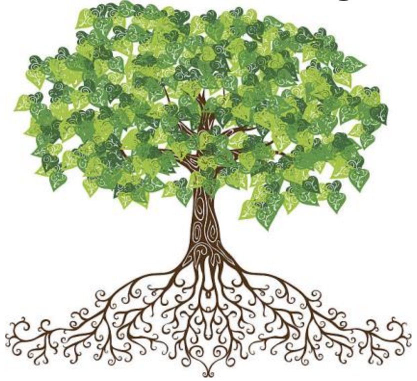
Should not because:

- we want a whatever ML model,
- Or «*If all you have is a hammer, everything looks like a nail*» - Abraham Maslow

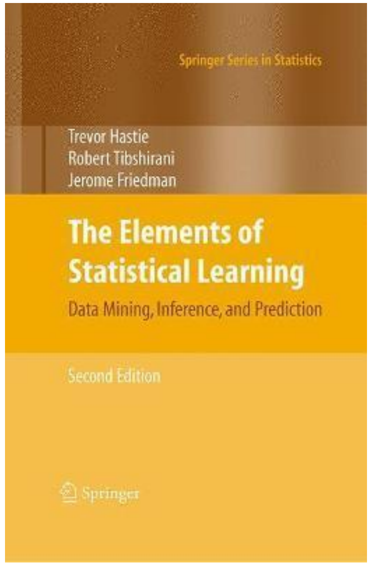
- picture from internet

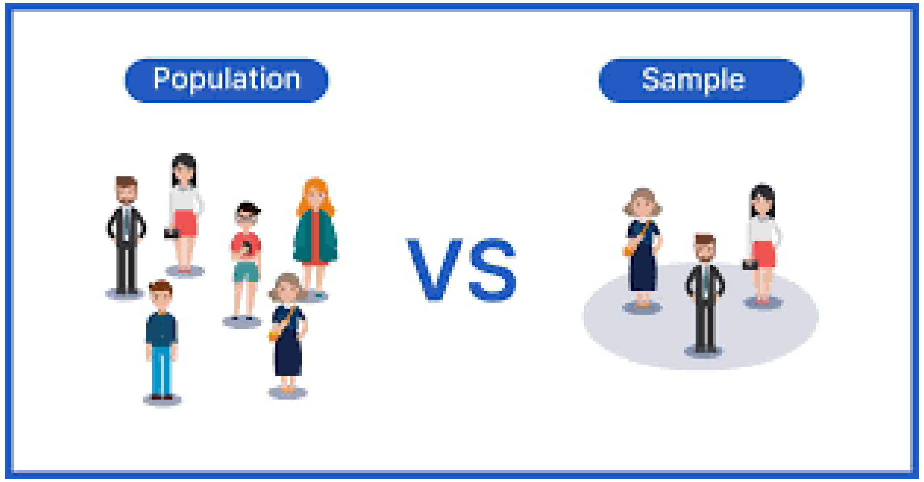
Jump back: Feasibility

Machine Learning



Statistics





Population and sampling is Machine Learning's fundation too!

Estimate feasibility:

- Samples should be representive, i.e. accurately reflect the **characteristics** of the larger group.
- **Characteristics** (cor)related to the target are available (features have predictive power for the target)
- Target are labelled correctly, etc.

In addition:

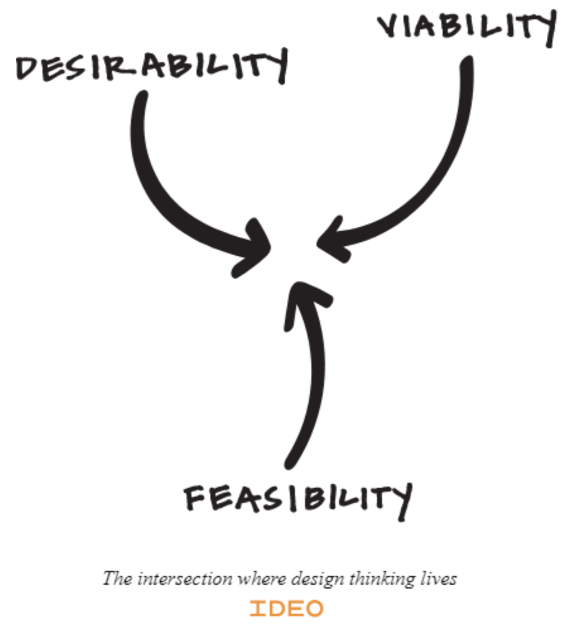
- enough computational power, resources, budget, potential quality (compared to non-ML solutions) etc.
- feasibility of the overall end-to-end solution

We will talk about data mismatch and high bias later.

- pictures from internet

When manager think they have a good idea…

- It shall still go through this process
- What can be worse: 10 managers all think they have a better idea than others



Make the ideas very clear and tangible is easier for management to control and setup milestones as well…

"Rome wasn't built in a day."

- It was built a block by block
- If you don't see even a single house on day 999, you probably will not be able to see a full Rome city on day 1000

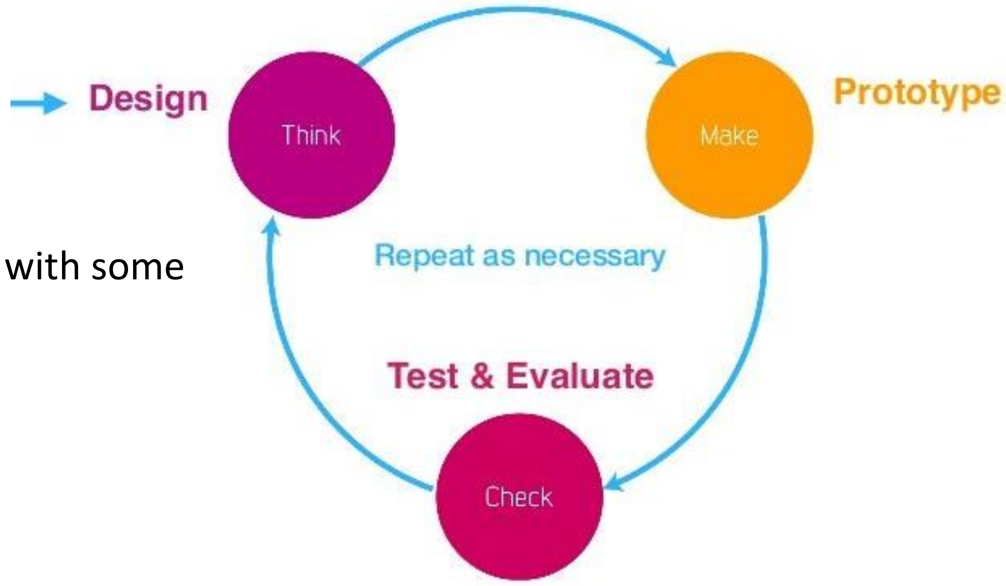
- picture from IDEO

In relality: most pains don’t need ~~surgery~~ ML model

Many times:

- rule-based (data engineering related)
- ad-hoc analysis
- dashboard

already can relieve some pains, suitable for running the prototype - test circle fast with some touchable results (a key to build the trust)



Prototype can be:

- Analysis/Dashboard with limited data
- SQL script extracts data (subset)
- etc.

After prototyping:

- Sit together with the customers, check, observe, and improve. Create a project after the prototype is satisifying

Remember it is in the problem defining phase – not a project yet, no sprint – do it as fast as possible.

- picture from internet

Kick-off: Project inception

Who:

Core team doing the work and the sponsoring stakeholders

What:

- Vision (long-term) and Goals (or the next few months)
- Non-goals (not immediate goals, not essential to have now)
- Risks (everyone shall identify, repeat the exercise in the end)
- Personas/Workflows/Stories/Estimation on critical stories
- Prioritization/next steps

- Team: ideally - two-pizza size, flat (for transparency and accountability), product owner/project manager can pause or stop to release the resources

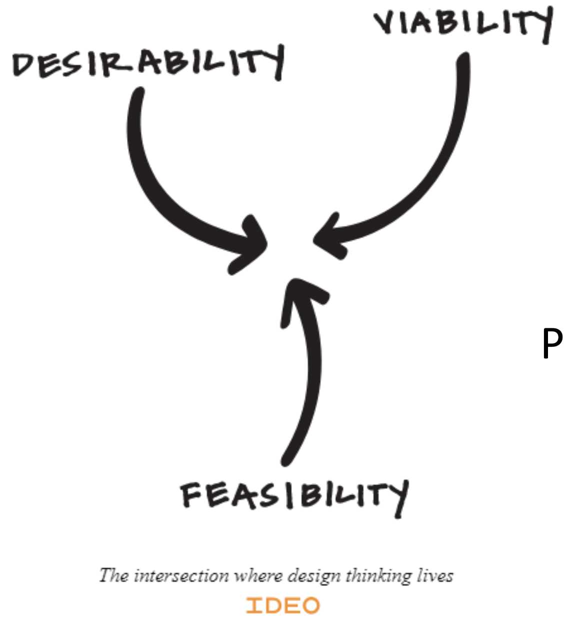
Everything should be documented.

- pictures from internet

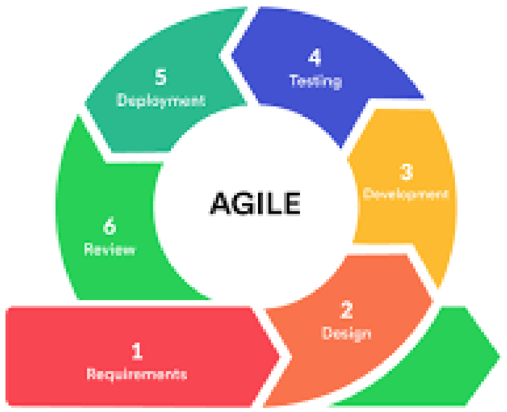
Recapture

“Well begun is half done.”

- Understand the problem for customers, and with customers
- Look into:
    - Desirablity
    - Business viability
    - Technical feasibility, for ML
        - not only the feasibility of building a model
            - Samples are representive
            - Characteristics exist in data
            - Correctly labelled
        - but also end-to-end implementation
- Make idea very clear and tangible
- Concrete and executable plan
- Be transparent



Problem defining



Problem solving

- pictures from internet

Break – next part will be a bit more technical, and ML focused

After the well begun – solve the problem

Before we start…

math or statistical model builders

painters

create the abstracts of the real world
formulas and numbers v.s. shapes and colours

- pictures from internet

Mass–energy equivalence:

E = MC2

describe the relationship between mass and energy

Poisson distribution: $$P(X) = \frac{\lambda^x e^{-\lambda}}{X!}$$

describe the pattern of things happen during a time interval

- both are simple and beautiful formulas

- When no strong relationship or obvious pattern…

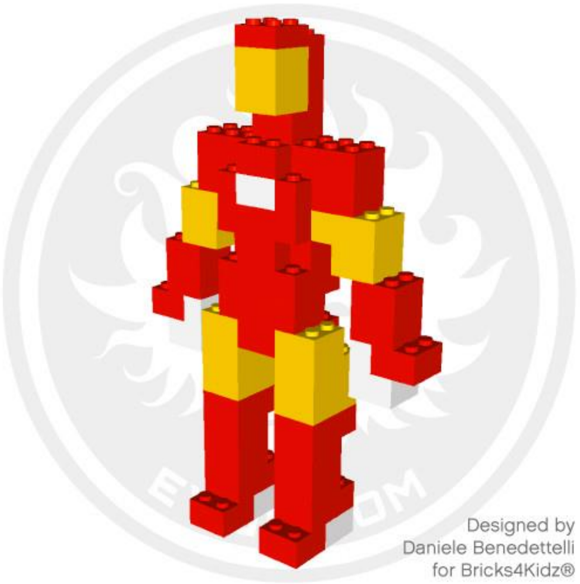
a dog or a cat?

Can we still use math to paint the world?

- Yes, but need models can handle more complexity…

pictures from FOD.no (Foreningen for Omplassering av Dyr)

Like lego bricks…

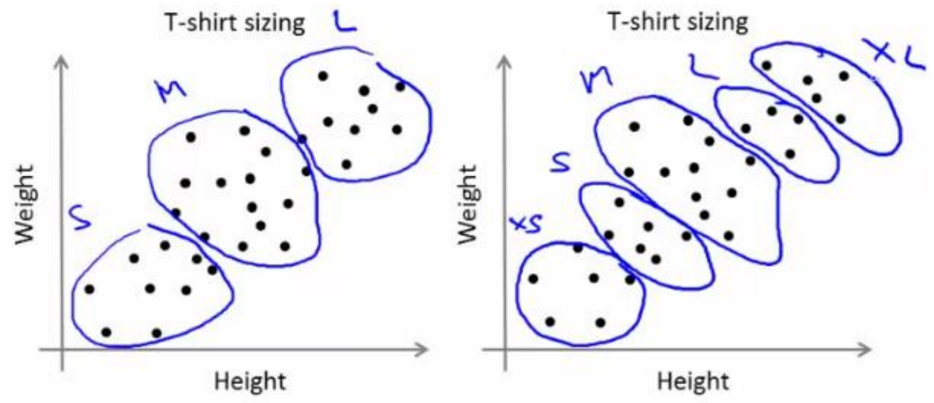


Basic unit is a piece of math… some package are with basic bricks, some are with more variants…

- pictures from internet

Machine Learning

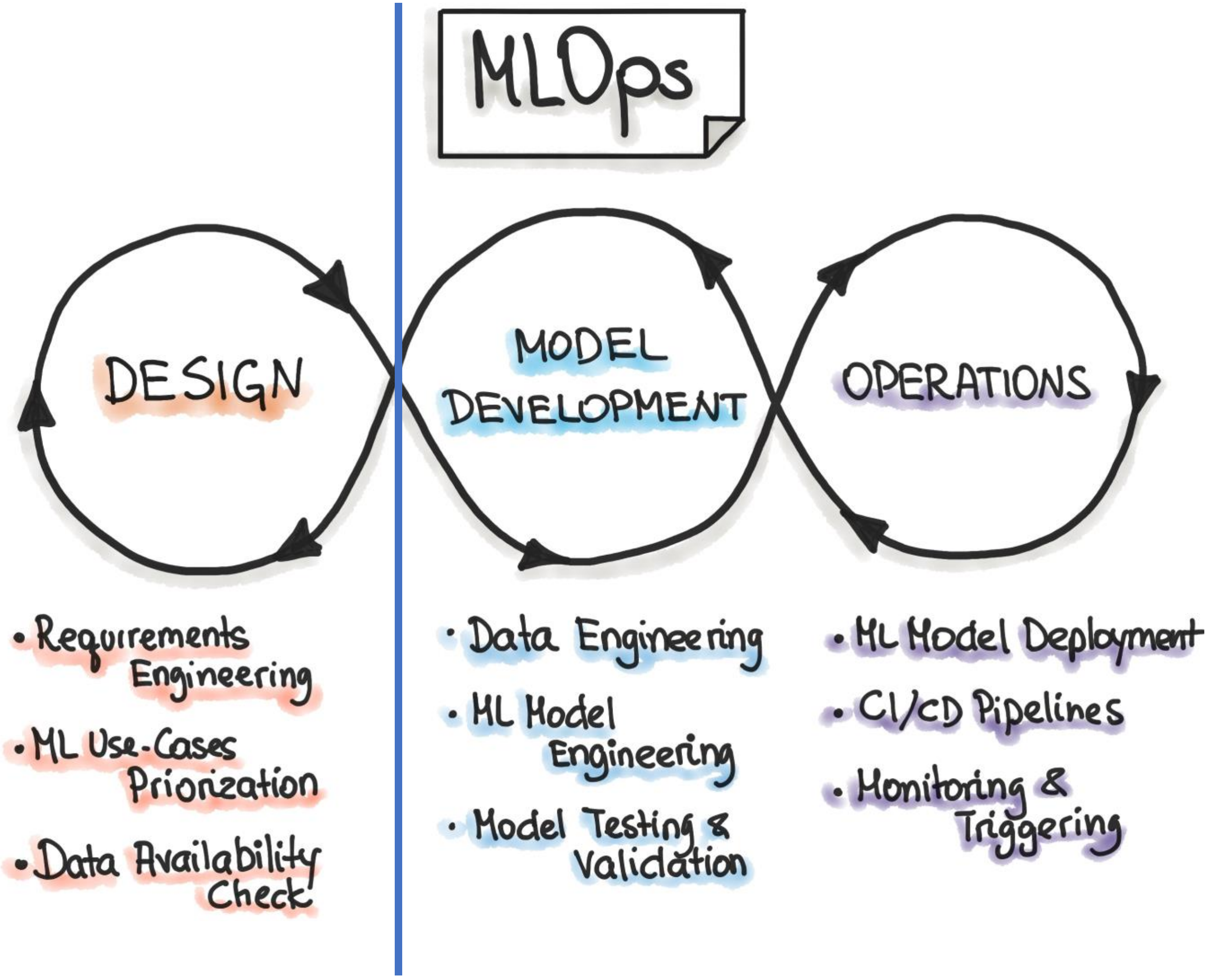
- Unsupervised learning: model finds patterns in structure from the data without any help
  - Common application: clustering
- Supervised learning: with example outputs
  - Classification: output variable is category
    - multi class classification: A, B, C, or more
    - **binary classification: yes or no**
  - Regression: output variable is numeric
- Reinforcement learning: learns by interacting with environment and learns to take action to maximize the total reward



- pictures from https://www.coursera.org/learn/machine-learning

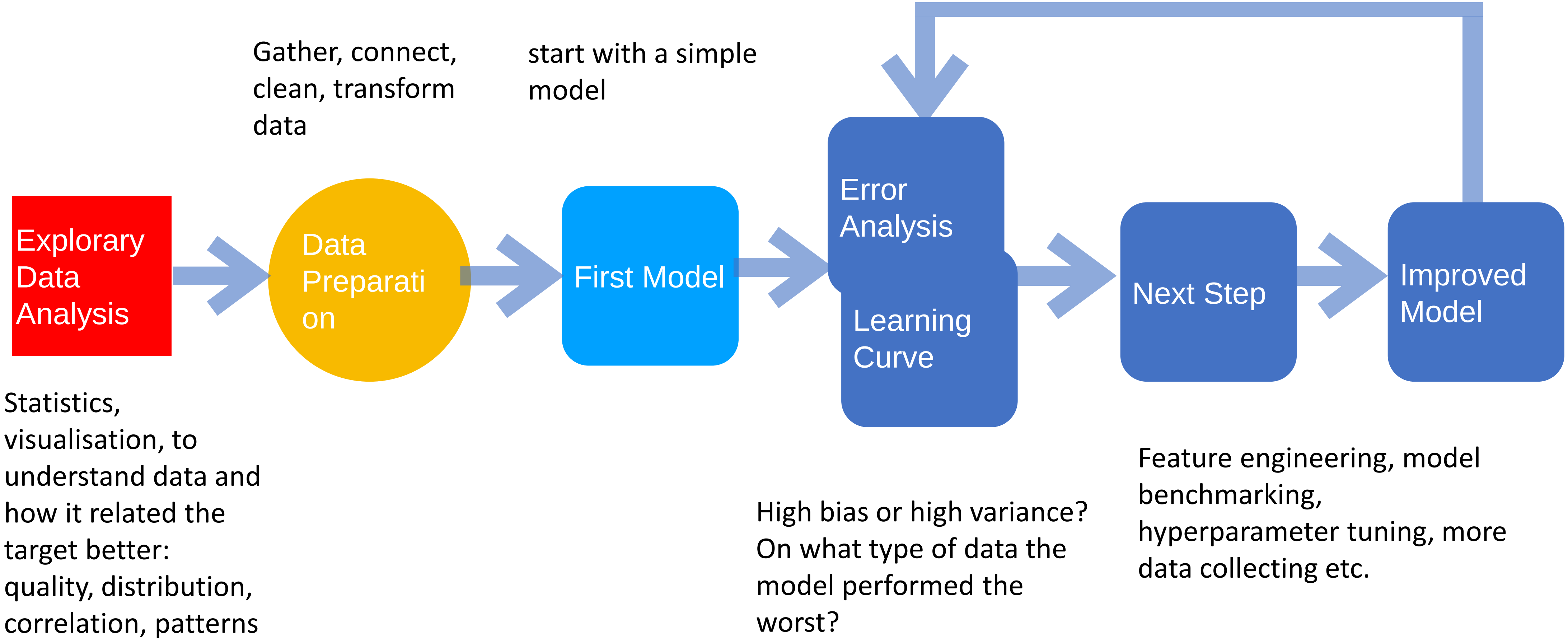
MLOps

- Covered the design part in the first part (briefly)
- Going to talk about
  - Model Development
  - Operations



From ml-ops.org – I cannot 100% agree with what is written here, but the picture is better than my hand writing…

Model Development: General Approach



Classification: imbalanced data

Proportion of minority class takes small part of the data set (e.g. fraud detection, AML)

- Try training on the true distribution, if it doesn’t work well:
  - Undersampling
  - Oversampling
  - SMOTE
  - Weight
  - Recommended approach: undersampling + upweighting

Homework: why imbalanced data may be a problem? (hint: think about sampling, gradient descent, especially mini-batch) ;)

Data preparation

- Many things to consider there: missing value, outliers, skewed data, feature scaling (normalization, standardization), correlated features, encoding, merge…
- Think about why and for what
- Can also depent on algorithms, e.g: some algorithms are affected by range of features, like KNN, SVM, linear regressions etc, therefore requires feature scaling, but some are not, e.g. tree-based models (homework: why? hint: think about how these algorithms work)

Few popular algorithms for classification

	SVM	KNN	Tree-based
Pros:	• Effective in the higher dimension. • When the number of features are more than training examples. • When classes are separable • Outliers have less impact. • Extreme imbalanced data	• Intuitive case • Simple and easy to implement	• Can deal with multiple features which may are correlated • Can handle both numeric and categorical features easily • Don't require much data pre-processing (e.g. normalization)
Cons:	• Slow when large data set • In case of overlapped classes • Appropriate hyperparameters and appropriate kernel function can be tricky	• Slow when large data set	• Overfitting • Less interpretable • A large grid search during tuning – computational expensive

- not necessary be able to write code from scratch, but need to know how they work, pros and cons

Performance metrics

- If I am developing an AML model, will accuracy be the best performance measure?

Sources: [20][21][22][23][24][25][26][27] view · talk · edit

		Predicted condition			
	Total population $= P + N$	**Positive (PP)**	**Negative (PN)**	Informedness, bookmaker informedness (BM) $= TPR + TNR - 1$	Prevalence threshold (PT) $= \frac{\sqrt{TPR \times FPR} - FPR}{TPR - FPR}$
Actual condition	**Positive (P)**	**True positive (TP),** hit	**False negative (FN),** type II error, miss, underestimation	True positive rate (TPR), recall, sensitivity (SEN), probability of detection, hit rate, power $= \frac{TP}{P} = 1 - FNR$	False negative rate (FNR), miss rate $= \frac{FN}{P} = 1 - TPR$
	Negative (N)	**False positive (FP),** type I error, false alarm, overestimation	**True negative (TN),** correct rejection	False positive rate (FPR), probability of false alarm, fall-out $= \frac{FP}{N} = 1 - TNR$	True negative rate (TNR), specificity (SPC), selectivity $= \frac{TN}{N} = 1 - FPR$
	Prevalence $= \frac{P}{P + N}$	Positive predictive value (PPV), precision $= \frac{TP}{PP} = 1 - FDR$	False omission rate (FOR) $= \frac{FN}{PN} = 1 - NPV$	Positive likelihood ratio (LR+) $= \frac{TPR}{FPR}$	Negative likelihood ratio (LR−) $= \frac{FNR}{TNR}$
	Accuracy (ACC) $= \frac{TP + TN}{P + N}$	False discovery rate (FDR) $= \frac{FP}{PP} = 1 - PPV$	Negative predictive value (NPV) $= \frac{TN}{PN}$ $= 1 - FOR$	Markedness (MK), deltaP (Δp) $= PPV + NPV - 1$	Diagnostic odds ratio (DOR) $= \frac{LR+}{LR-}$
	Balanced accuracy (BA) $= \frac{TPR + TNR}{2}$	F_1 score $= \frac{2PPV \times TPR}{PPV + TPR} = \frac{2TP}{2TP + FP + FN}$	Fowlkes–Mallows index (FM) $= \sqrt{PPV \times TPR}$	Matthews correlation coefficient (MCC) $= \sqrt{TPR \times TNR \times PPV \times NPV} - \sqrt{FNR \times FPR \times FOR \times FDR}$	Threat score (TS), critical success index (CSI), Jaccard index $= \frac{TP}{TP + FN + FP}$

What to choose depends on the business case

Performance metrics

threhold	tn	fp	fn	tp	accuracy	sensitivity (tp/p)	miss rate (1 - sensitivity)	precision
0.1	1063.0	8491.0	331.0	13880.0	0.628782	0.976708	0.023292	0.620446
0.2	3195.0	6359.0	1913.0	12298.0	0.651925	0.865386	0.134614	0.659163
0.3	5288.0	4266.0	4100.0	10111.0	0.647970	0.711491	0.288509	0.703276
0.4	6930.0	2624.0	6331.0	7880.0	0.623185	0.554500	0.445500	0.750190
0.5	8038.0	1516.0	8227.0	5984.0	0.590027	0.421082	0.578918	0.797867
0.6	8734.0	820.0	9792.0	4419.0	0.553461	0.310956	0.689044	0.843482
0.7	9139.0	415.0	11107.0	3104.0	0.515169	0.218422	0.781578	0.882069
0.8	9382.0	172.0	12265.0	1946.0	0.476667	0.136936	0.863064	0.918791
0.9	9506.0	48.0	13421.0	790.0	0.433242	0.055591	0.944409	0.942721

An example: model in general with good precision but bad miss rate.

Your (measured) goal decides how to train

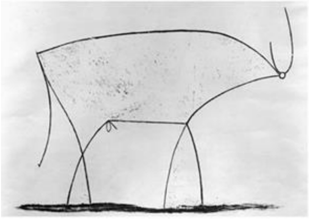
If you are **uncertain** about business cost, **use AUC, Gini Coefficient** (2* AUC - 1), they measure how well the model ranks good on top of bad.

- picture from internet

Data split

- Dev/Training – if the model is with high bias
- Validation – if the model is with high variance
- Test – if any data mismatch (i.e. samples in training data set reflects real-world – if test samples representive real-world)

training data



underfit – high bias

overfit – high variance

mismatch

- maybe not the most precise illustration, but hope you get what I mean

Homework: find out what is cross validation, how to do data split on time series.

- drawings by Pablo Picasso, pictures from internet

What to do next?

If high variance:

- Get more training data
- Trying smaller sets of features
- Increasing regularization (hyper-)parameter
- Simpler models

If high bias:

- Adding features
- Adding polynomial features
- Decreasing regularization (hyper-)parameter
- Train longer/better optimization algorithms
- Use models which can handle complexity

- error analysis on which data the model performed the worst (find way to improve)

AutoML

“AutoML enables developers with limited machine learning expertise to train ~~high-quality~~ models ~~specific to their business needs~~” (https://cloud.google.com/automl)

- Big «For/While-loop»: try a group of algorithms, hyper-parameters, see which combination gives the best performance in terms of the performance measures you select.
- Seems all ML frameworks/platforms provide a convenient way for it
- In practice: do random search first, and then grid search in a smaller promising area

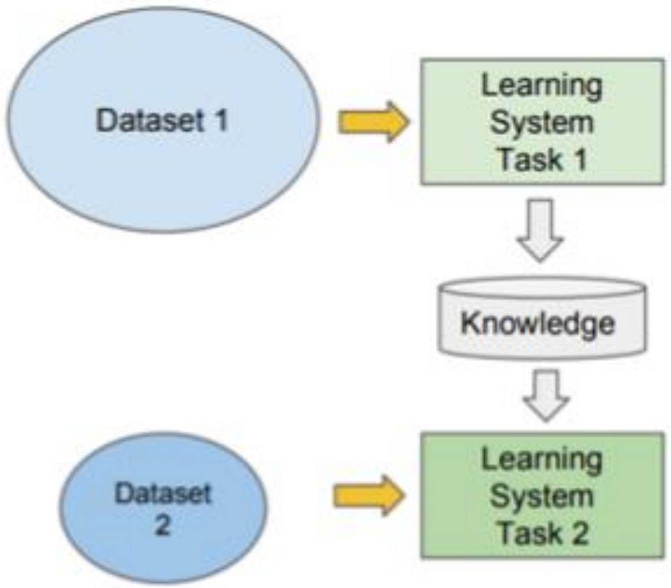
- Like weight penguins one by one, find the heaviest as the «penguin of the year»

- picture from internet

Transfer Learning (for Deep Learning)

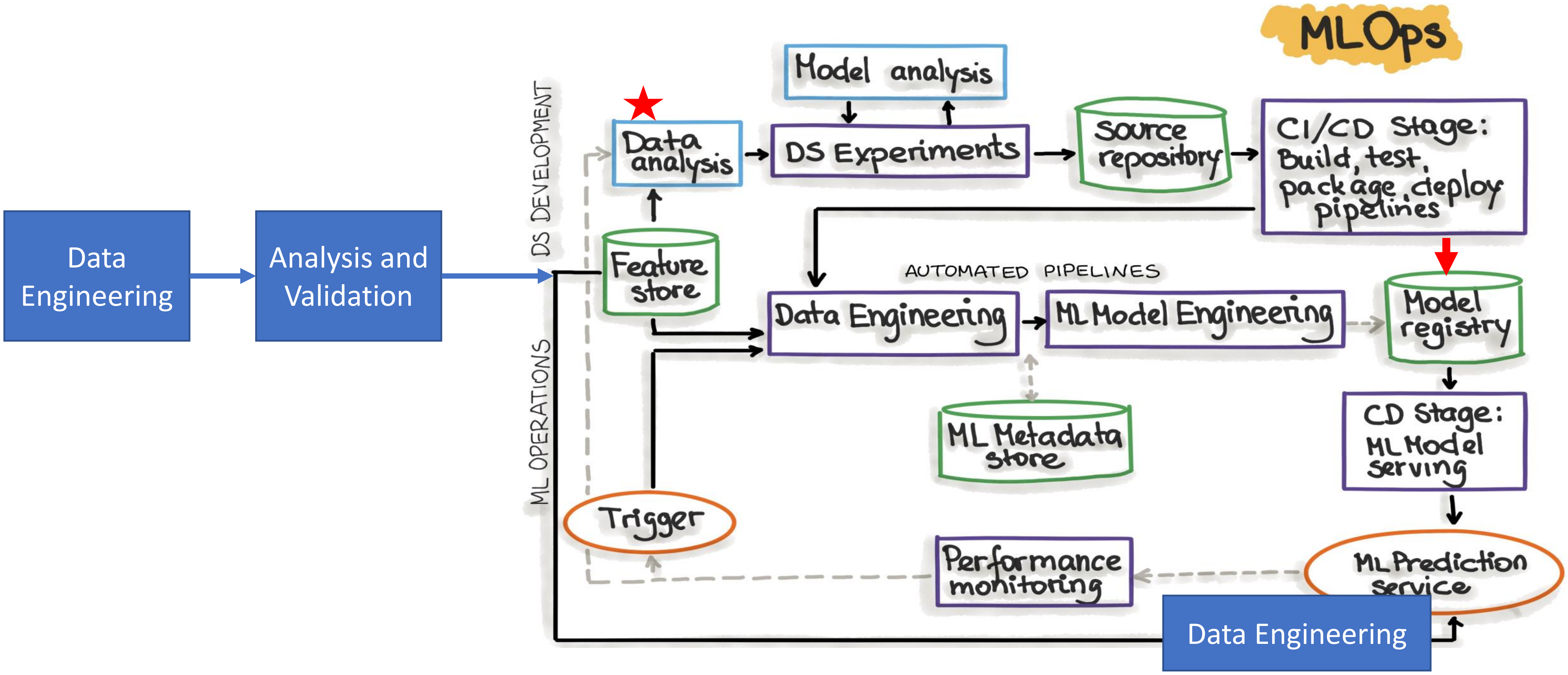
Common to use pre-trained models for computer vision and natural language processing

- fact: BERT: pre-trained from unlabeled data extracted from the BooksCorpus with 800M words and English Wikipedia with 2,500M words



Picture from: https://towardsdatascience.com/a-comprehensive-hands-on-guide-to-transfer-learning-with-real-world-applications-in-deep-learning-212bf3b2f27a

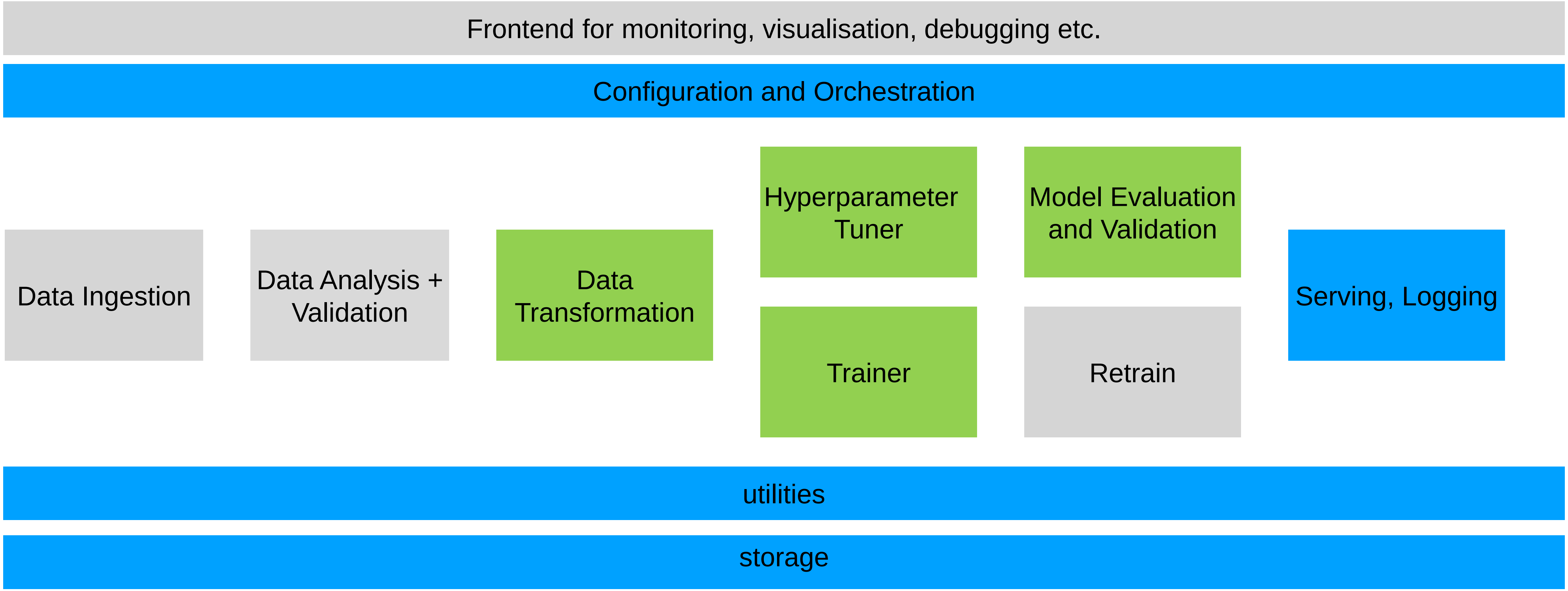
Production



From ml-ops.org

Overall Picture

Machine Learning Model in Production



Last homework

- A model looks perfect in development but works bad in production, what else can be the reason besides data mismatch?

If you are interested in ML:

- check this online course for some basic knowledge: https://www.coursera.org/learn/machine-learning
- remember to review statisitcs from school

Thank you for your time!