Full Stack AI Projects

AI Specialization

Henry Ruiz
GDE ML
@devharuiz
https://haruiz.github.io/

What is this course about?

How to :

Build and design applications and products with AI.

- A **machine learning model** can only begin to **add value to an organization** when that model's insights routinely become available to the users for which it was built

Create and Lead ML groups

Establish a working relationship between software engineers, infrastructure managers and data scientists.

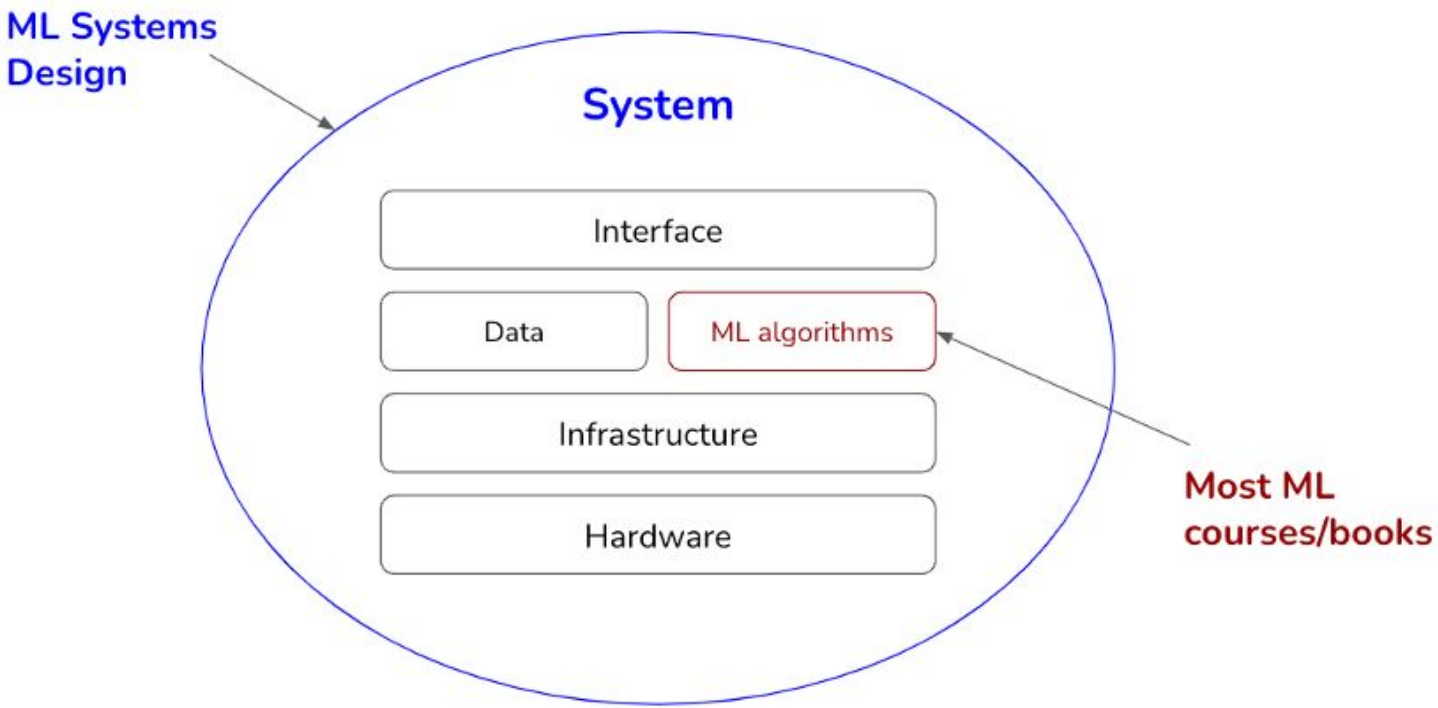
- **Software engineers** who want to build robust and responsible systems meeting the specific challenges of working with ML components
- **Data scientists** who want to facilitate getting a prototype model into production

Describe and Understand some of the most common ML deployment scenarios

Evaluate, testing and Monitoring ML systems

ML in production

ML Systems design



What is this course NOT about

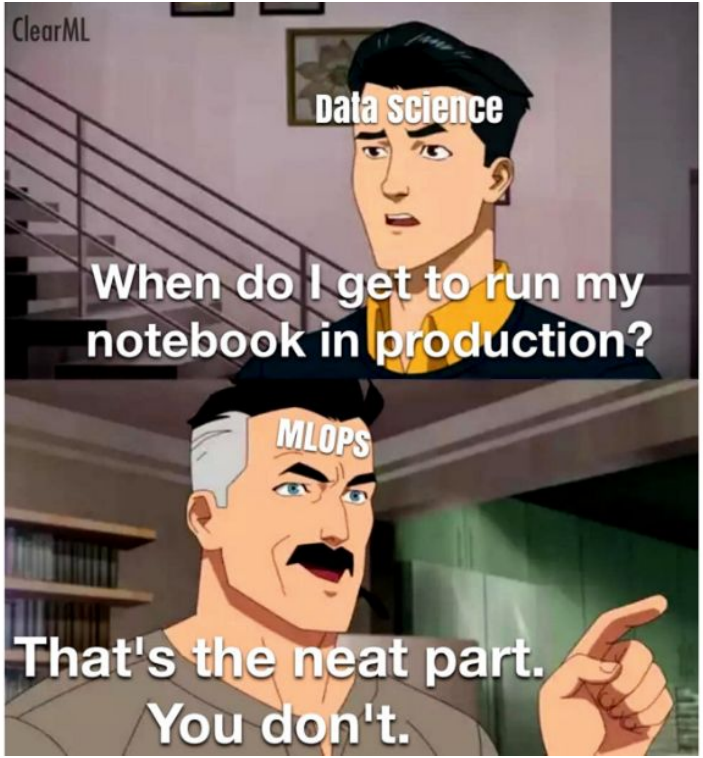
- Machine learning/deep learning algorithms
  - CS 229: Machine Learning
  - CS 230: Deep Learning
  - CS 231N: Convolutional Neural Networks for Visual Recognition
  - CS 224N: Natural Language Processing with Deep Learning
- Computer systems
  - CS 110: Principles of Computer Systems
  - CS 140E: Operating systems design and implementation
- UX design
  - CS 147: Introduction to Human-Computer Interaction Design
  - DESINST 240: Designing Machine Learning: A Multidisciplinary Approach

What is machine learning system design?

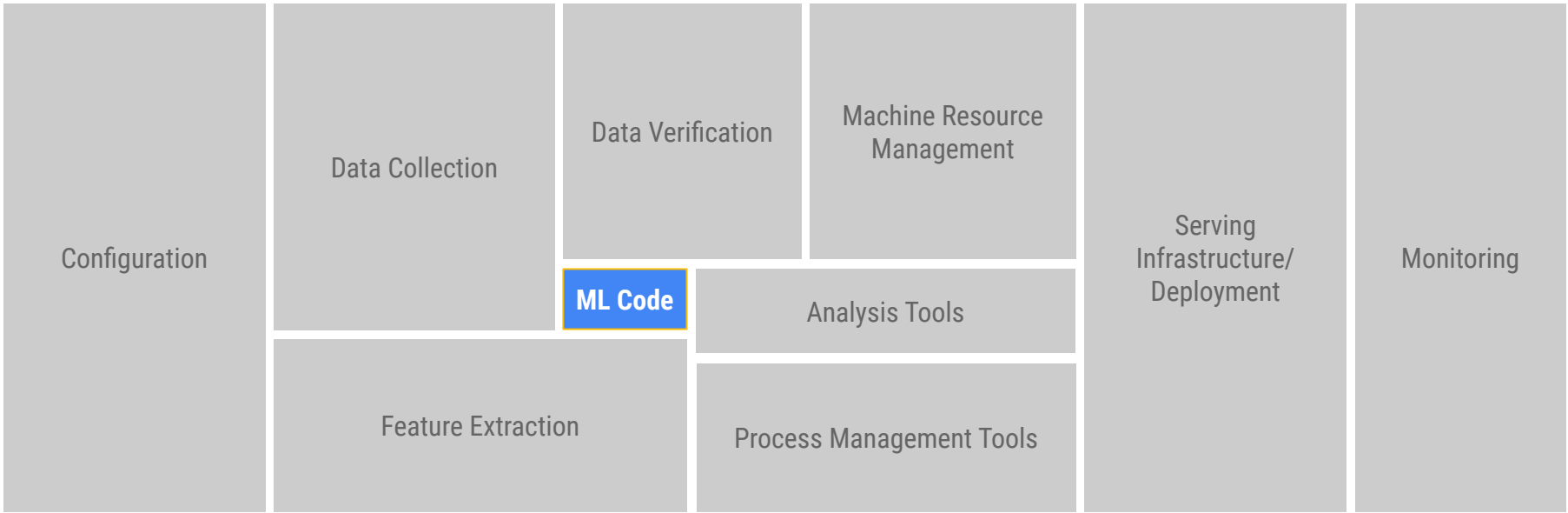
The process of **defining the interface, algorithms, data, infrastructure, and hardware** for a machine learning system to satisfy specified requirements.

The course should help you to address some of the below questions ...

You've trained a model, now what?
What are different components of an ML system?
How to engineer features?
How to evaluate your models, both offline and online?
What's the difference between online prediction and batch prediction?
How to serve a model on production?
How to continually monitor and deploy changes to ML systems?



… a production solution requires so much more

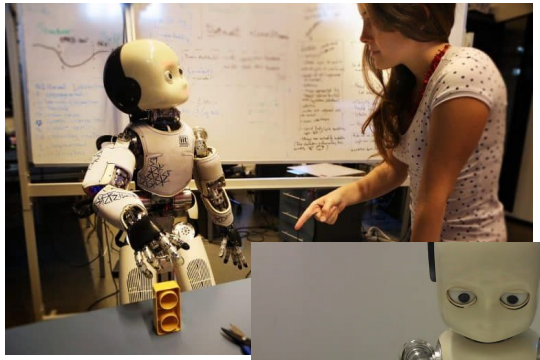


Model's deployment course could potentially be the continuation

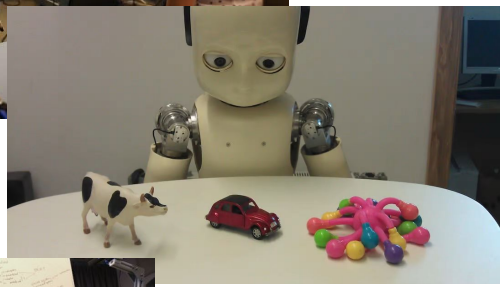
Model deployment is one of the most important steps in the ML pipeline. We have spent a lot of time and effort playing around with different models, training and tuning our model hyperparameters, so after evaluating its performance and obtaining that long-awaited score, now is the time to release it to the world, exposing this to real use.

It sounds like graduation time!!.

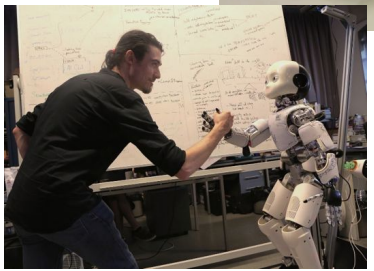
Training



Testing



Evaluation

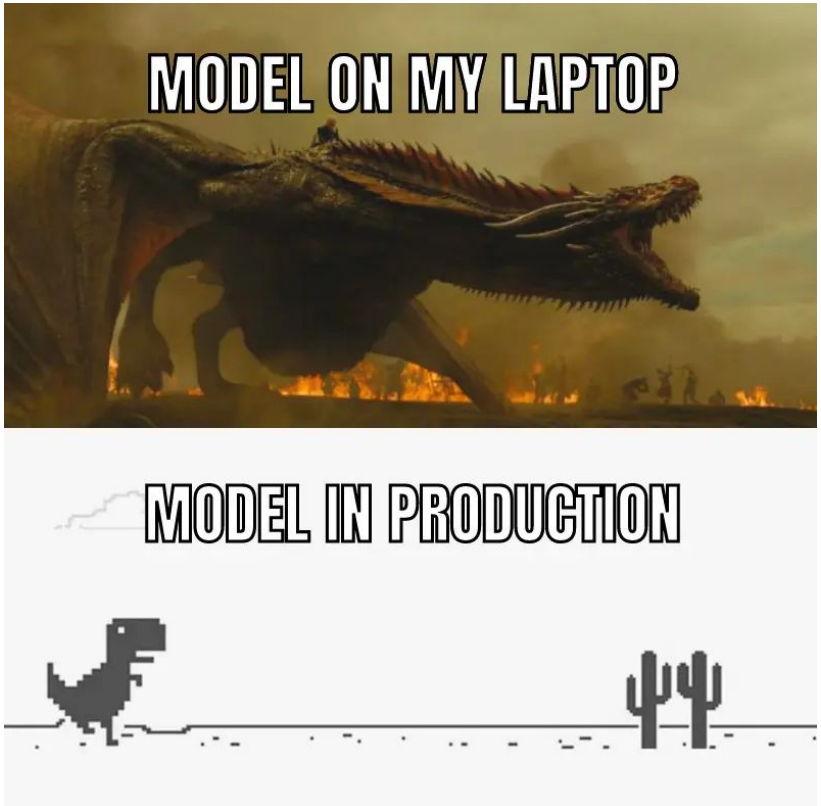


Deployment

Model Deployment

A machine learning model can only begin to add value to an organization when that model's insights routinely become available to the users for which it was built.

The process of taking a trained ML model and making its predictions available to users or other systems is known as deployment.



Course Methodology

Three hours of classes in two sections every week

Friday 6:30pm – 9:30 pm

Saturday 11:00 am – 2:00 pm

1 Final project and a paper discussion

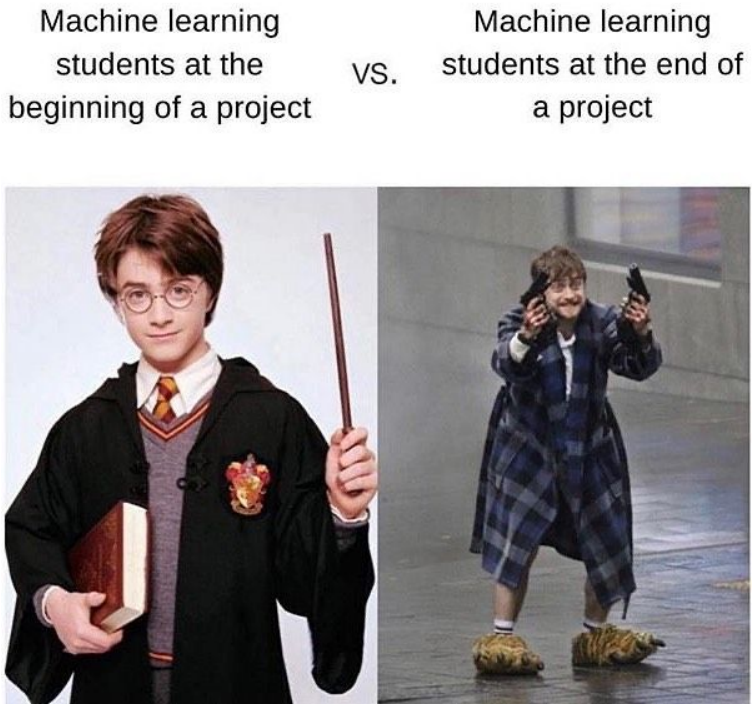
5 hours of lecture, 1 working on the project (optional)

Setting up Machine Learning Projects

Like any other software solution, ML systems require a **well-structured methodology** to maximize the success rate of the implementation.

ML algorithms are the less challenging part. The hard part is **making algorithms work with other software infrastructure components to solve real-world problems.**

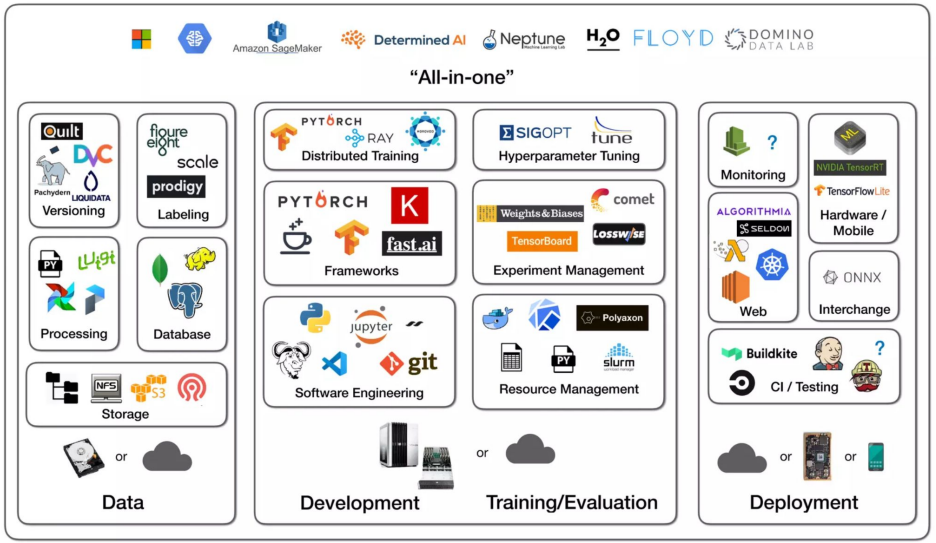
Non-ML components cause 60/96 failures, and 60% of the models never make it into production.



Infrastructure & Tooling

The number of **available tools to work with ML seems endless.**

Selecting the appropriate tools depends on:
the kind of problem, **type of solution, deployment scenario, capacity building**, team experience, cost, hardware and software infrastructure, etc.




Google
Cloud Platform
Azure
amazon
web services™
IBM Watson™
FLOYD
scikit
learn
TensorFlow
PyTorch
NumPy
Pandas
DASK
Numba
astropy
gprlib
OPTUNA
SciPy
OpenCV

Data Management

ML is all about data

Data can be represented and stored differently.

We will explore different storage and data management solutions in this part of the course.

Machine Learning Teams

Machine Learning talents are expensive and scarce.

ML teams have diverse roles.

Managing and leading ML and Data Science teams require unique skills.

Here we will learn more about these roles' importance and impact within the organization.

Training, Debugging and Design patterns

In engineering disciplines, **design patterns capture best practices and solutions to commonly occurring problems.**

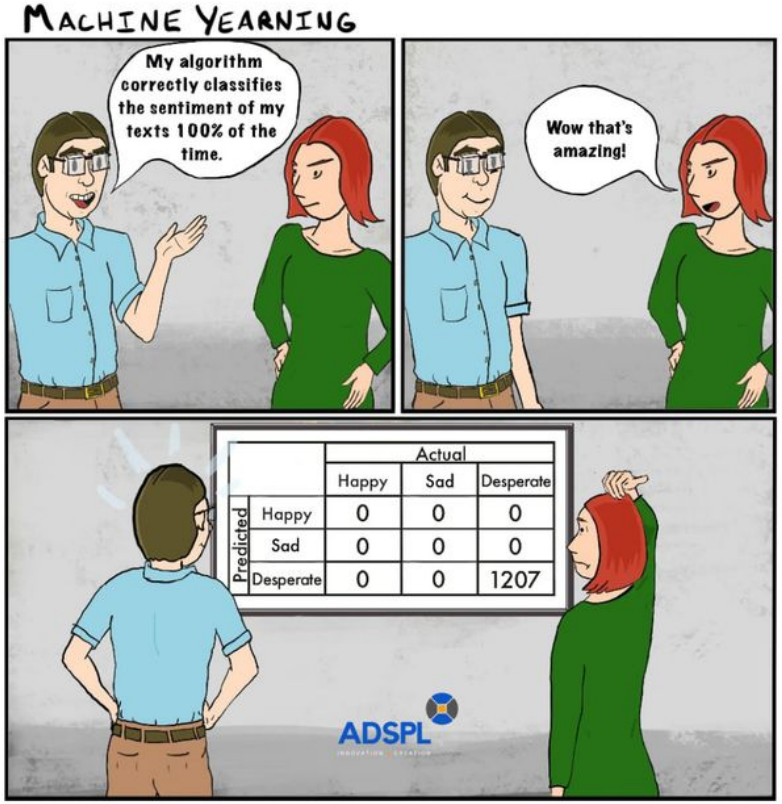
They codify the knowledge and experience of experts into advice that all practitioners can follow.

This course will introduce some design patterns or repeatable solutions to issues in ML engineering for training and debugging ML systems.

Testing and Deployment

A machine learning model can only begin to add value to an organization when that **model's insights routinely become available to the users for which it was built.**

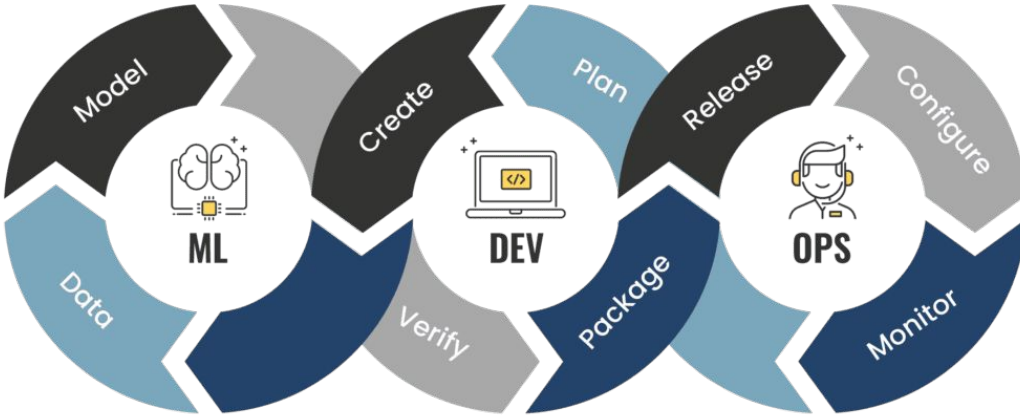
Let's learn together about **troubleshooting and deploying ML models in production.**



MLOps

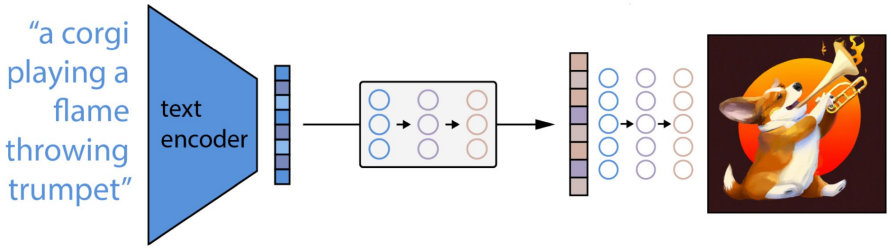
MLOps is **a methodology for ML engineering that unifies ML system development** (the ML element) with **ML system operations** (the Ops element).

This course will discuss how MLOps maximize the capacities and resources of ML teams by providing a set of standardized processes and technology capabilities for building, deploying, and operationalizing ML systems rapidly and reliably.



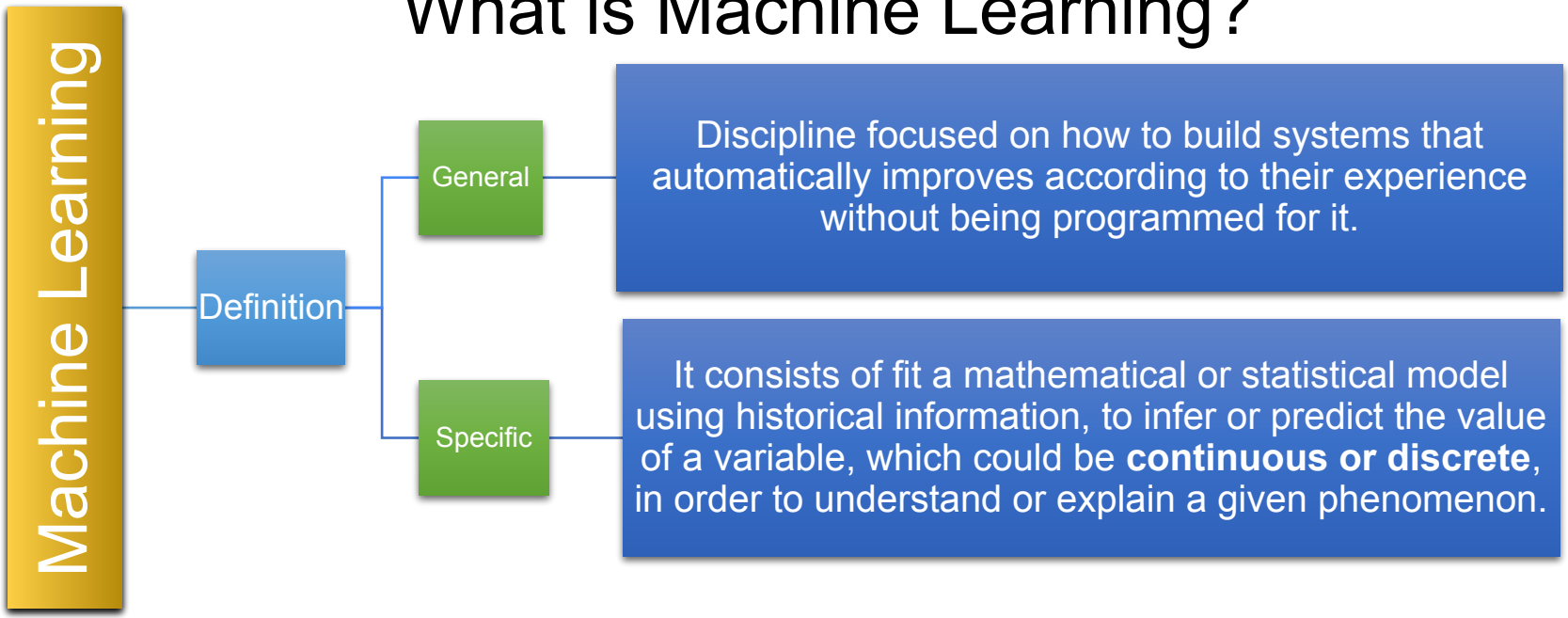
Research Areas

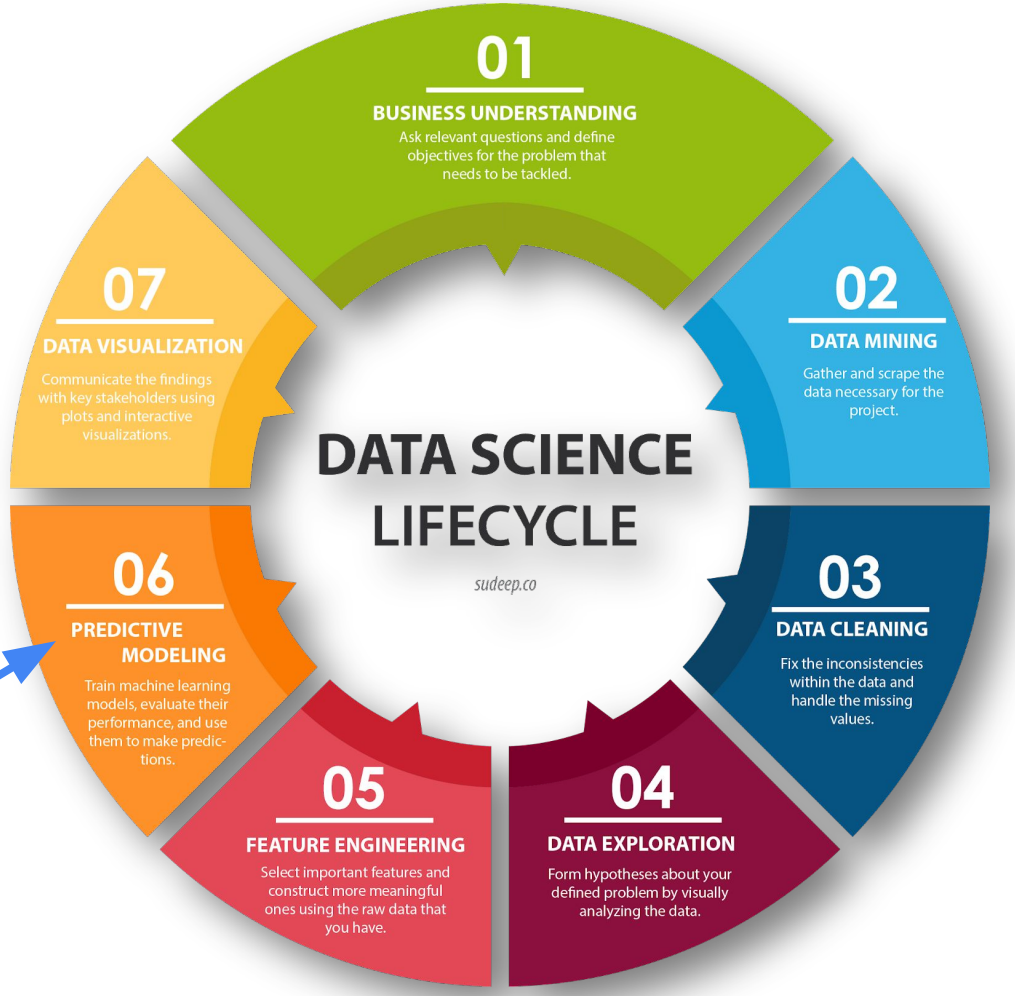
In this section, the current advances in AI will be discussed, such as **Attention-based models, Transformers, Diffusion models, and Multimodal models**



ML basic concepts review

What is Machine Learning?



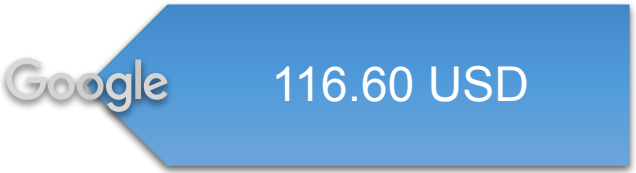


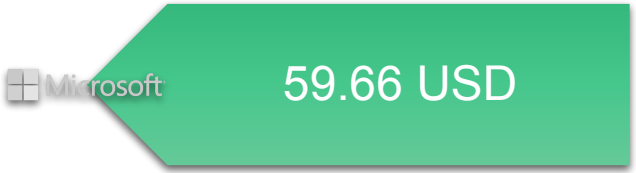
Machine Learning

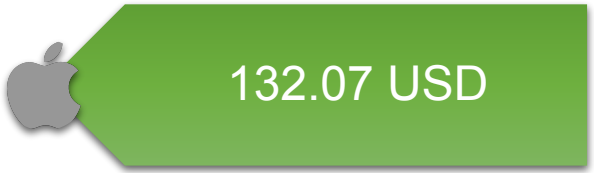
ML problems

Regression

What's tomorrow's stock price? (Continuous)

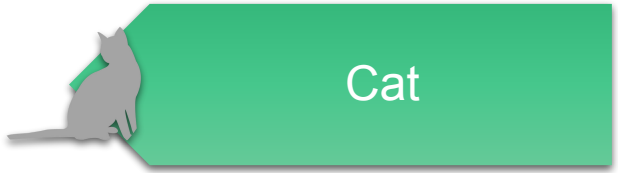


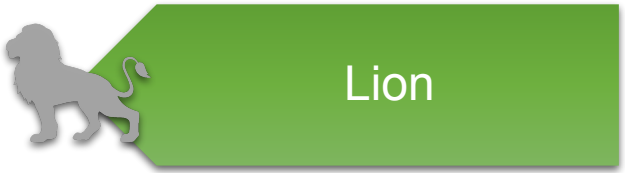




Classification

What's in this picture? (Discrete)



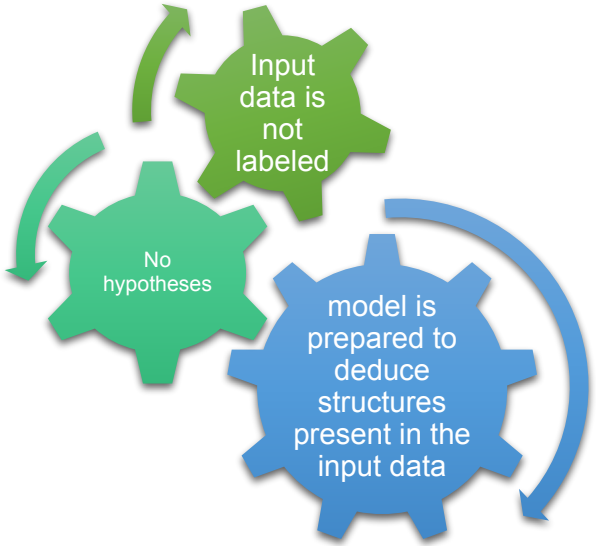


How does the machine Learn?

Unsupervised Learning

Data driven

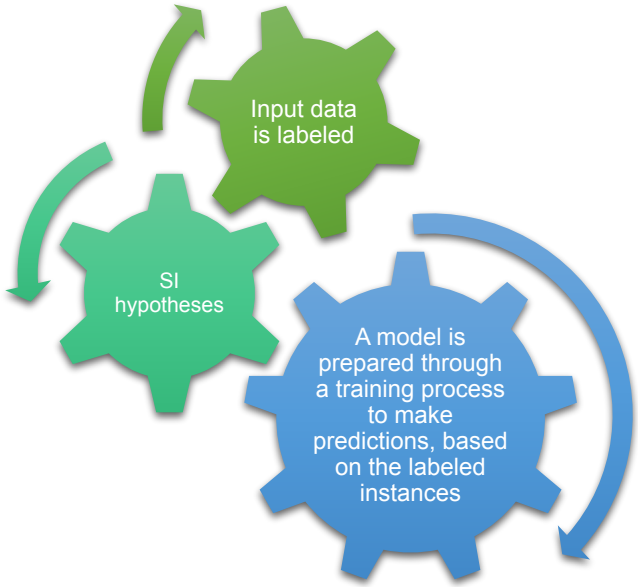
Learning a function that describe the
structure of the unlabeled data by revealing the hidden relationships between the data points



Supervised Learning

Task driven

Based on the provided training examples(pairs inputs and expected output), **Learning** a function that maps an input to an output by creating a decision boundaries across the classes.



Reinforcement learning

Learn from mistakes

We have an agent learning to perform a given task by interacting with the environment and receiving feedback based on its actions

Industry vs Academia

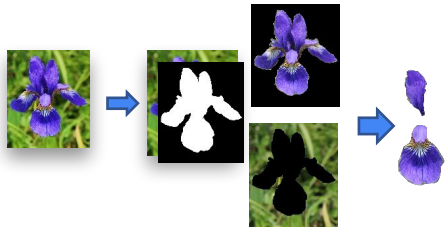
Machine Learning practitioner

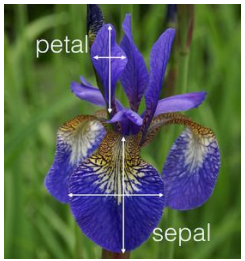
- Tasks Oriented:
- Queries databases.
- Cleaning data.
- Writing scripts to transform data and probe algorithms.
- Play around with libraries.
- Make all easy.
- Find the best model writing custom code.

Machine Learning Research

- Research Oriented:
- Read papers.
- Implement the algorithm from scratch.
- Translate Math into code.
- Reducing algorithms.
- Use beauty math to develop their own model

Features Engineering Approach (Features are known)





	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa

segmentation

Representation and Description

Pre-processing

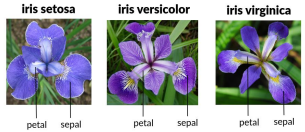
Knowledge Base

Recognition and interpretation (Train Model)

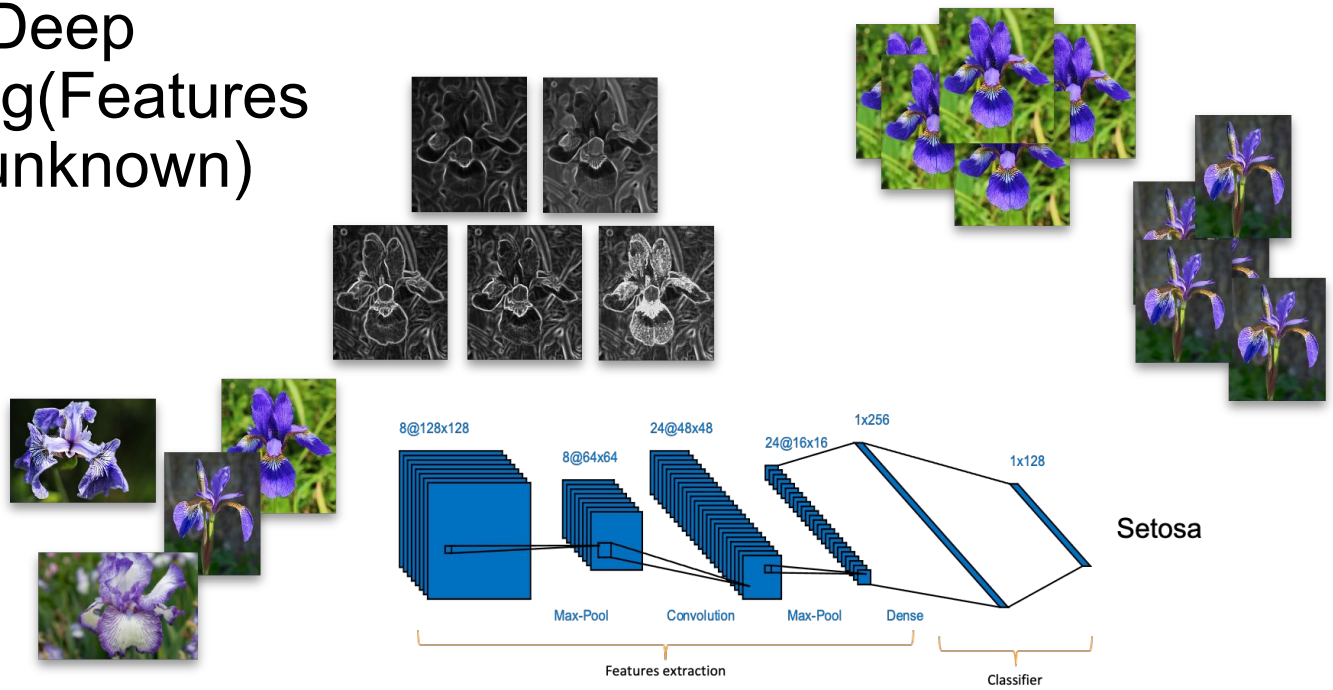
Results

Proficiency issue

Image Acquisition



Deep learning(Features are unknown)

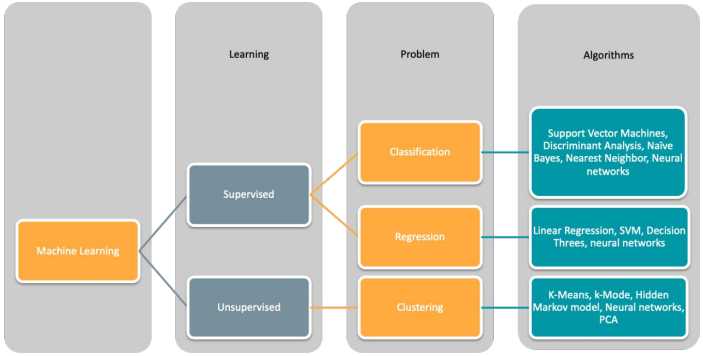


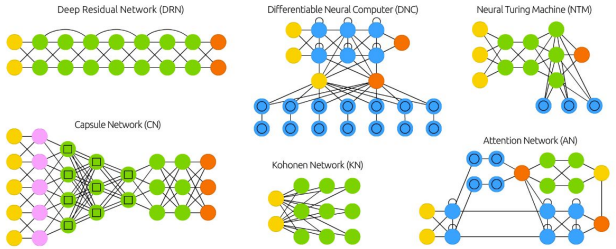
	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
5	5.0	3.6	1.4	0.2	setosa
6	5.4	3.9	1.7	0.4	setosa
7	4.6	3.4	1.4	0.3	setosa
8	5.0	3.4	1.5	0.2	setosa

Structured data

Non-Structured data

Fit data



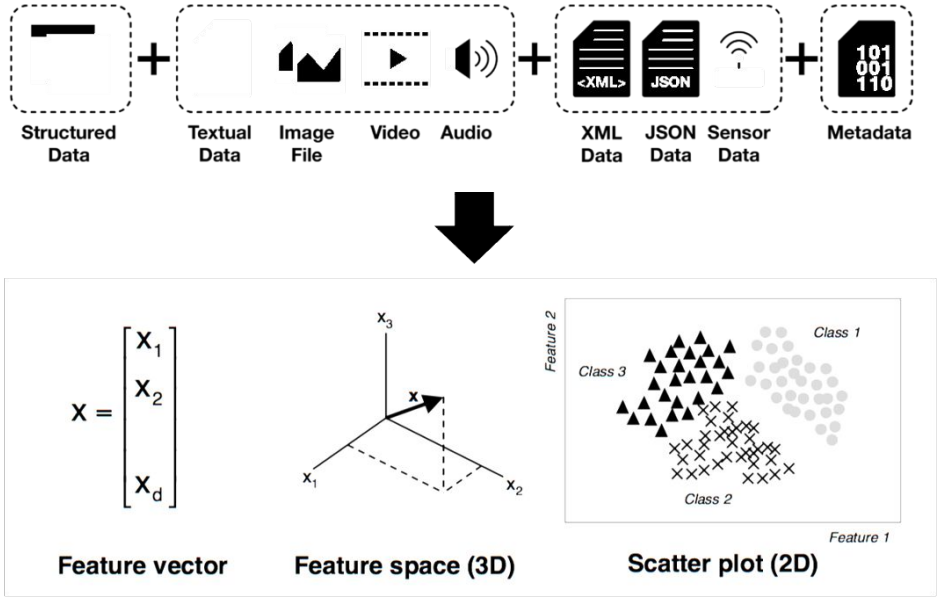


https://www.asimovinstitute.org/neural-network-zoo/

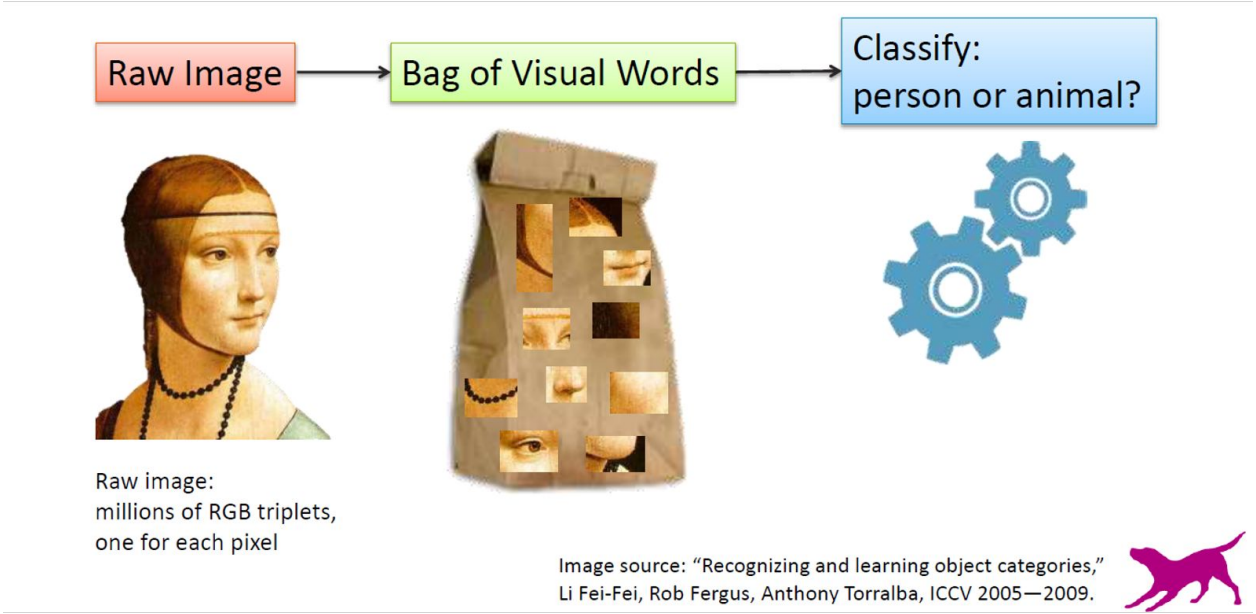
Algorithm/Architecture Selection

Model

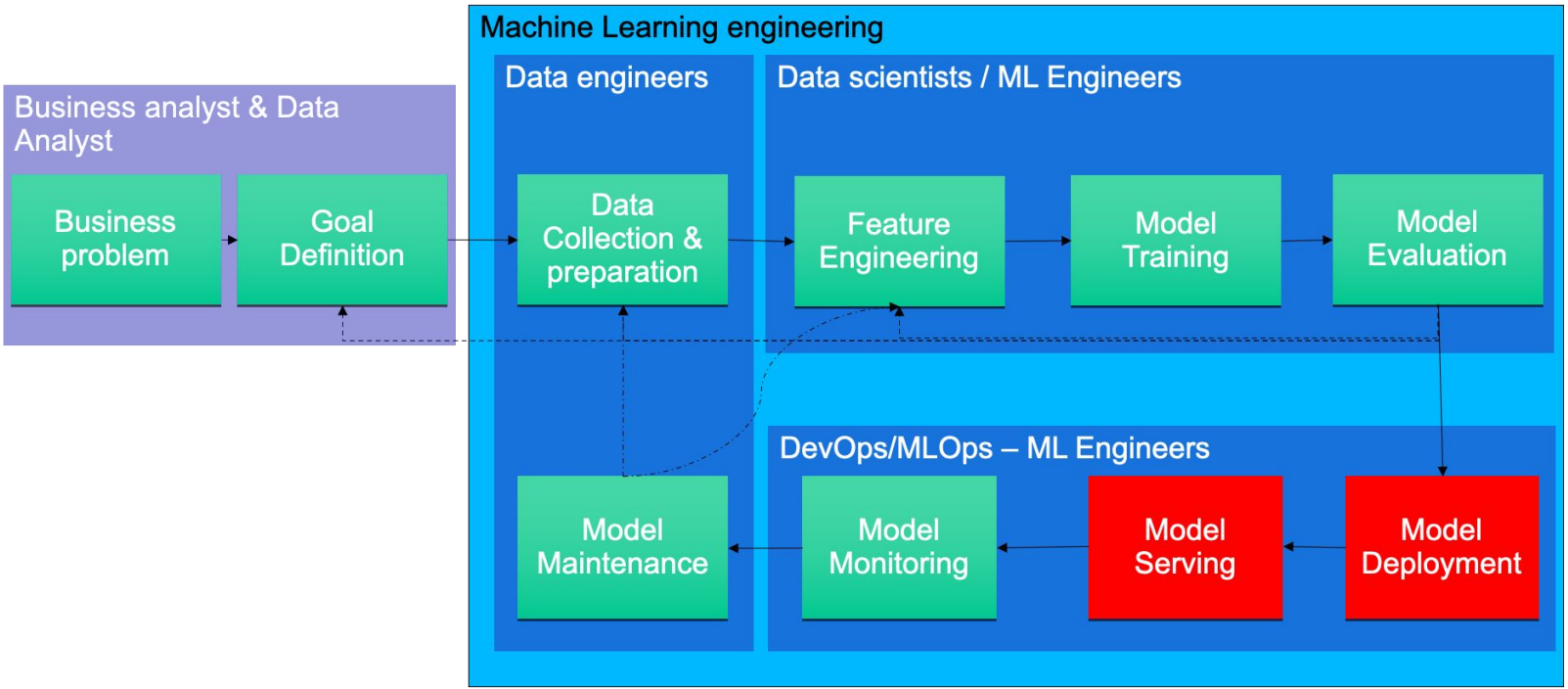
How our data should be presented to the model?



Feature space



ML Learning Pipeline

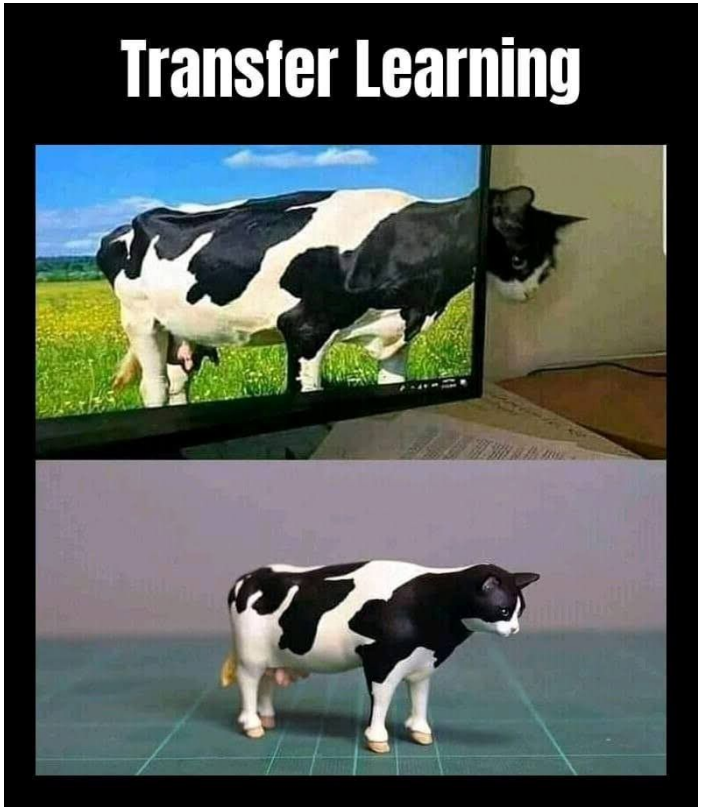


Transfer Learning

Transfer learning is when a model developed for one task is reused to work on a second task. Fine-tuning is one approach to transfer learning where you change the model output to fit the new task and train only the output model.

In **Transfer Learning** or Domain Adaptation, we train the model with a dataset. Then, we train the same model with another dataset that has a different distribution of classes, or even with other classes than in the first training dataset).

In **Fine-tuning**, an approach of Transfer Learning, we have a dataset, and we use let's say 90% of it in training. Then, we train the same model with the remaining 10%. Usually, we change the learning rate to a smaller one, so it does not have a significant impact on the already adjusted weights.



Setting up Machine Learning Projects

Based on the statistics, according to a 2019 Report, 85% of AI projects fail to deliver On their intended promise to business

Why do so many projects fail?

ML is still in research - you shouldn't aim for 100% success rate

But, many are doomed to fail

Technically infeasible or poor scoped

Never make the leap to production

Unclear success criteria

Poor team management

Lifecycle of a ML project

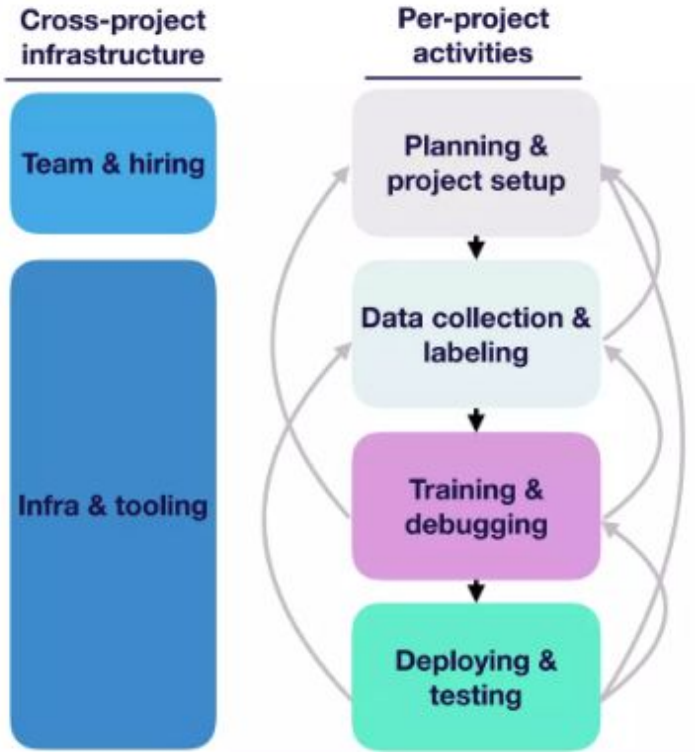
https://github.com/ageron/handson-ml/blob/master/ml-project-checklist.md

Phase 1 is **Project Planning and Project Setup**: At this phase, we want to decide the problem to work on, determine the requirements and goals, as well as figure out how to allocate resources properly.

Phase 2 is **Data Collection and Data Labeling**: At this phase, we want to collect training data (images, text, tabular, etc.) and potentially annotate them with ground truth, depending on the specific sources where they come from.

Phase 3 is **Model Training and Model Debugging**: At this phase, we want to implement baseline models quickly, find and reproduce state-of-the-art methods for the problem domain, debug our implementation, and improve the model performance for specific tasks.

Phase 4 is **Model Deployment and Model Testing**: At this phase, we want to pilot the model in a constrained environment, write tests to prevent regressions, and roll the model into production.



What also do you need to know?

Understand state of the art in your domain

Understand what is possible

Know what to try next

We will introduce most promising areas

Be aware that not the more complex solution will be the best, it will impact significantly your resources management plan.

ML is not alway the solution

Applied machine learning projects

Problem Definition: Understand and clearly describe the problem that is being solved.
Analyze Data: Understand the information available that will be used to develop a model.
Prepare Data: Discover and expose the structure in the dataset.
Evaluate Algorithms: Develop a test harness and baseline accuracy from which to improve.
Improve Results: Leverage results to develop more accurate models.
Present Results: Describe the problem and solution so that it can be understood by third parties.

What is the problem?

Informal description
Describe the problem as though you were describing it to a friend or colleague. This can provide a great starting point for highlighting areas that you might need to expand upon. It also provides the basis for a one sentence description you can use to share your understanding of the problem.
For example: I need a program that will tell me which tweets will get retweets.

Formalism
Tom Mitchell defines a useful formalism for describing a machine learning problem:
A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E.

Applied machine learning projects

Problem Definition: Understand and clearly describe the problem that is being solved.
Analyze Data: Understand the information available that will be used to develop a model.
Prepare Data: Discover and expose the structure in the dataset.
Evaluate Algorithms: Develop a test harness and baseline accuracy from which to improve.
Improve Results: Leverage results to develop more accurate models.
Present Results: Describe the problem and solution so that it can be understood by third parties.

Analyze the data

The objective of the data analysis step is to increase the understanding of the problem by better understanding the problems data. This involves providing multiple different ways to describe the data as an opportunity to review and capture observations and assumptions that can be tested in later experiments.
There are two different approaches I use to describe a given dataset:

1. Summarize Data: Describe the data and the data distributions.
2. Visualize Data: Create various graphical summaries of the data.

Applied machine learning projects

Problem Definition: Understand and clearly describe the problem that is being solved.
Analyze Data: Understand the information available that will be used to develop a model.
Prepare Data: Discover and expose the structure in the dataset.
Evaluate Algorithms: Develop a test harness and baseline accuracy from which to improve.
Improve Results: Leverage results to develop more accurate models.
Present Results: Describe the problem and solution so that it can be understood by third parties.

Prepare the data

The more disciplined you are in your handling of data, the more consistent and better results you are likely to achieve. The process for getting data ready for a machine learning algorithm can be summarized in three steps:

1. Select Data
2. Preprocess Data
3. Transform Data

Applied machine learning projects

Problem Definition: Understand and clearly describe the problem that is being solved.
Analyze Data: Understand the information available that will be used to develop a model.
Prepare Data: Discover and expose the structure in the dataset.
Evaluate Algorithms: Develop a test harness and baseline accuracy from which to improve.
Improve Results: Leverage results to develop more accurate models.
Present Results: Describe the problem and solution so that it can be understood by third parties.

Evaluate Algorithms

n this part you will step through a process to rapidly test algorithms and discover whether or not there is structure in your problem for the algorithms to learn and which algorithms are effective.
There are two considerations when evaluating algorithms:
1. Define your Test Harness
2. Spot Checking Algorithms

Applied machine learning projects

Problem Definition: Understand and clearly describe the problem that is being solved.
Analyze Data: Understand the information available that will be used to develop a model.
Prepare Data: Discover and expose the structure in the dataset.
Evaluate Algorithms: Develop a test harness and baseline accuracy from which to improve.
Improve Results: Leverage results to develop more accurate models.
Present Results: Describe the problem and solution so that it can be understood by third parties.

Improve results

When tuning algorithms you must have a high confidence in the results given by your test harness. This means that you should be using techniques that reduce the variance of the performance measure you are using to assess algorithm runs. I suggest cross validation with a reasonably high number of folds (the exact number of which depends on your dataset).
The three strategies you will learn about in this part are:

1. Algorithm Tuning
2. Ensembles
3. Extreme Feature Engineering

Applied machine learning projects

Problem Definition: Understand and clearly describe the problem that is being solved.
Analyze Data: Understand the information available that will be used to develop a model.
Prepare Data: Discover and expose the structure in the dataset.
Evaluate Algorithms: Develop a test harness and baseline accuracy from which to improve.
Improve Results: Leverage results to develop more accurate models.
Present Results: Describe the problem and solution so that it can be understood by third parties.

Presents results

Depending on the type of problem you are trying to solve, the presentation of results will be very different. There are two main facets to making use of the results of your machine learning endeavor:

1. Report the results
2. Operationalize the system

Software Methodologies

Waterfall	Agile	Scrum	XP(Extreme Programming)
• Most traditional and sequential choices. • Linear sequential approach • Each stage must be completed before moving on to the next • Best suited for projects with well-defined and stable requirements	• An iterative and incremental approach • Emphasizes teamwork, adaptability, and customer satisfaction • Used in rapidly changing environments and for projects with unclear requirements • Using the Agile approach, teams develop in short sprints or iterations, each of which includes a defined duration and list of deliverables, but in no particular order.	• Another way to implement the Agile approach, Scrum borrows from Agile's foundational beliefs and philosophy that teams and developers should collaborate heavily and daily. • Uses sprints, roles (e.g. product owner, scrum master, development team), and ceremonies (e.g. daily standup, sprint review, sprint retrospective) • Agile framework for managing and completing complex projects • Best suited for projects with rapidly changing requirements or in complex environments	• Agile methodology for software development • Emphasizes communication, simplicity, feedback, and courage • Practices include pair programming, test-driven development and continuous integration and deployment