Machine Learning and its Applications

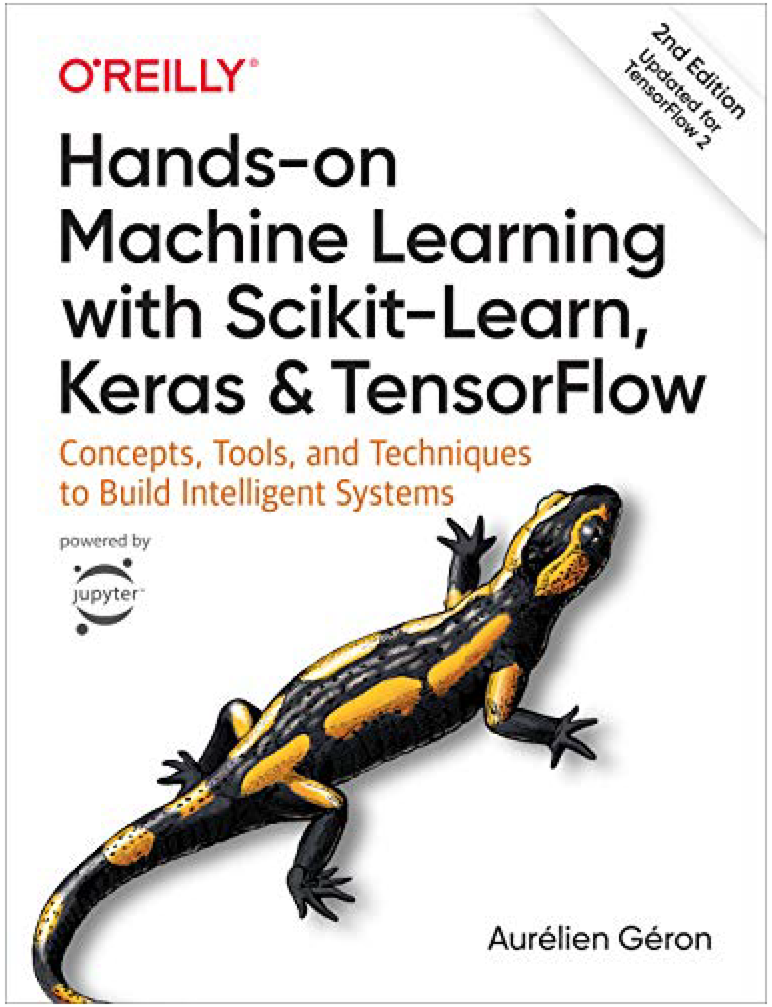
Week 1: Introduction to Machine Learning

Urban Information Lab

Learning Objective

1. Understand general information regarding ML & AI
2. Set up Anaconda & Python
3. Understand basic functions in Python

Textbook (Useful if you have one)



O'REILLY

2nd Edition
Updated for TensorFlow 2

Hands-on Machine Learning with Scikit-Learn, Keras & TensorFlow

Concepts, Tools, and Techniques to Build Intelligent Systems

powered by jupyter

Aurélien Géron

Session 1

1. Understand general information regarding ML & AI

Human Learning?

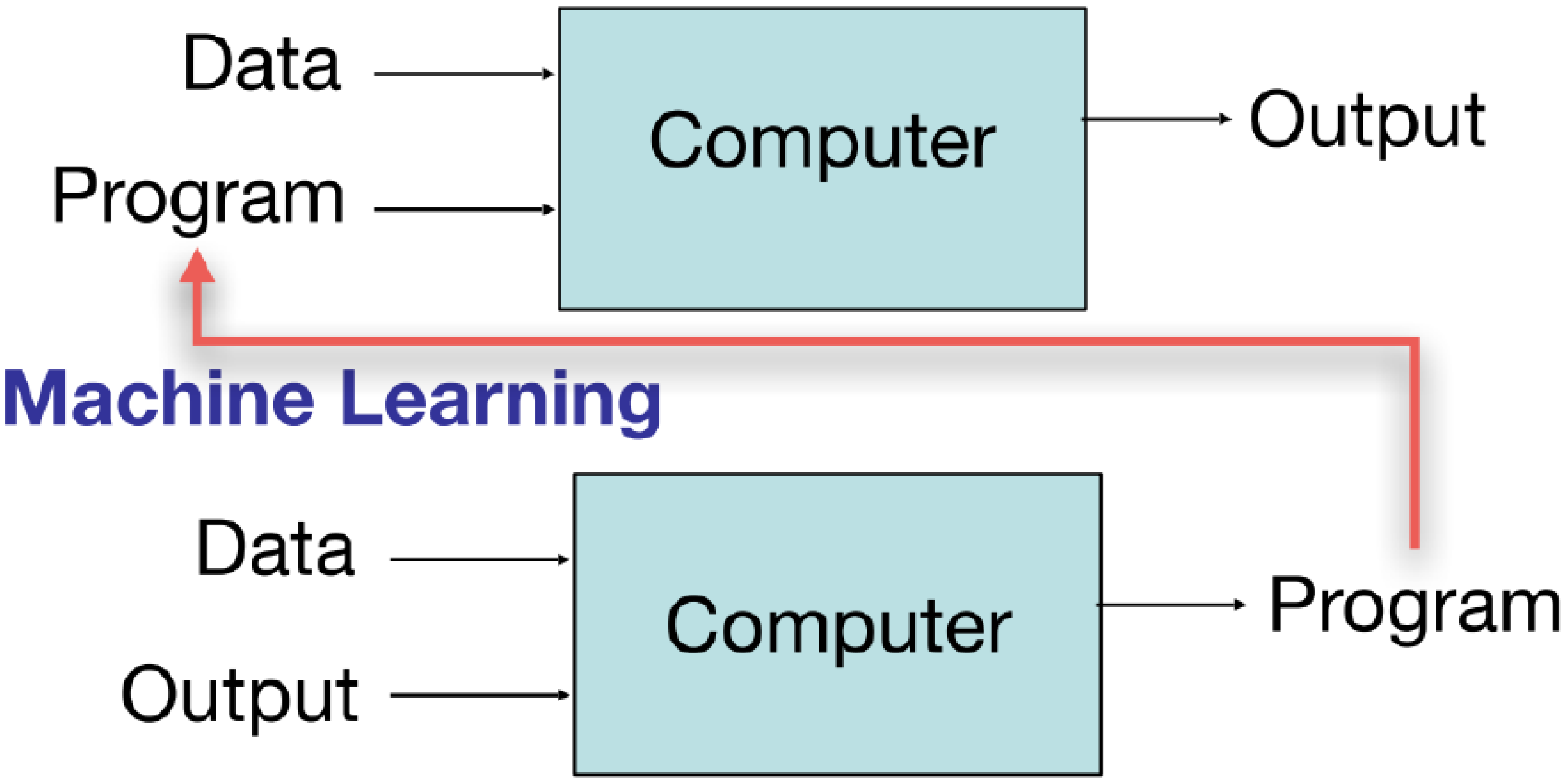
1. We learn from the things that happen to use – our experiences
2. Learning new things by acquiring appropriate responses to the events
3. Learning applies both humans and animals

Machine Learning?

1. Field of study that gives computers the ability to learn without being explicitly programmed (Arthur Samuel, 1959).
2. Algorithms that automatically detect patterns in data -> uncover patterns
3. Algorithms that improve their performance (P), at some task (T), with experience €

Comparison

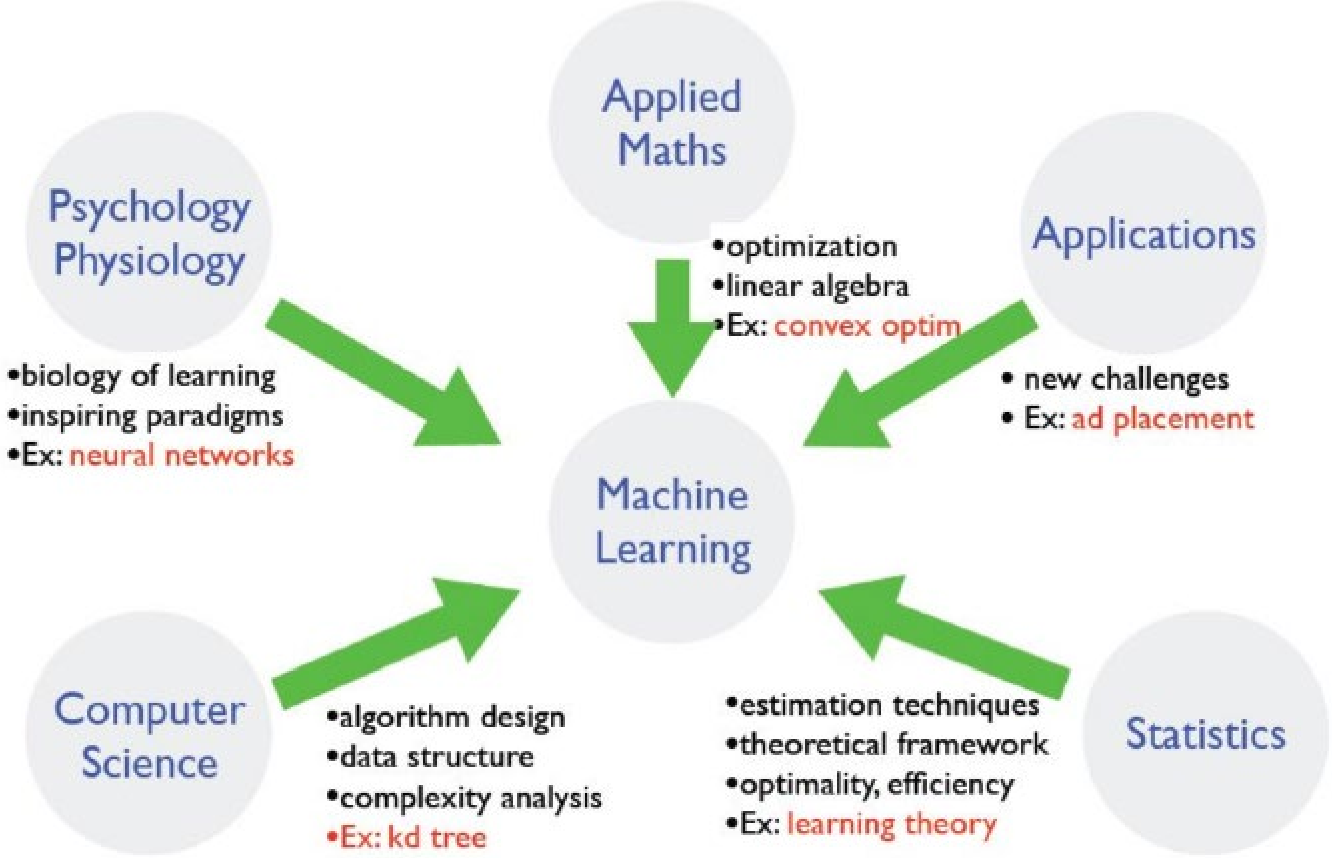
- **Traditional Programming**



- **Machine Learning**

slide by Pedro Domingos, Tom Mitchel, Tom Dietterich

Where does ML fit in?



Applied Maths

- optimization
- linear algebra
- Ex: convex optim

Psychology Physiology

- biology of learning
- inspiring paradigms
- Ex: neural networks

Applications

- new challenges
- Ex: ad placement

Machine Learning

Computer Science

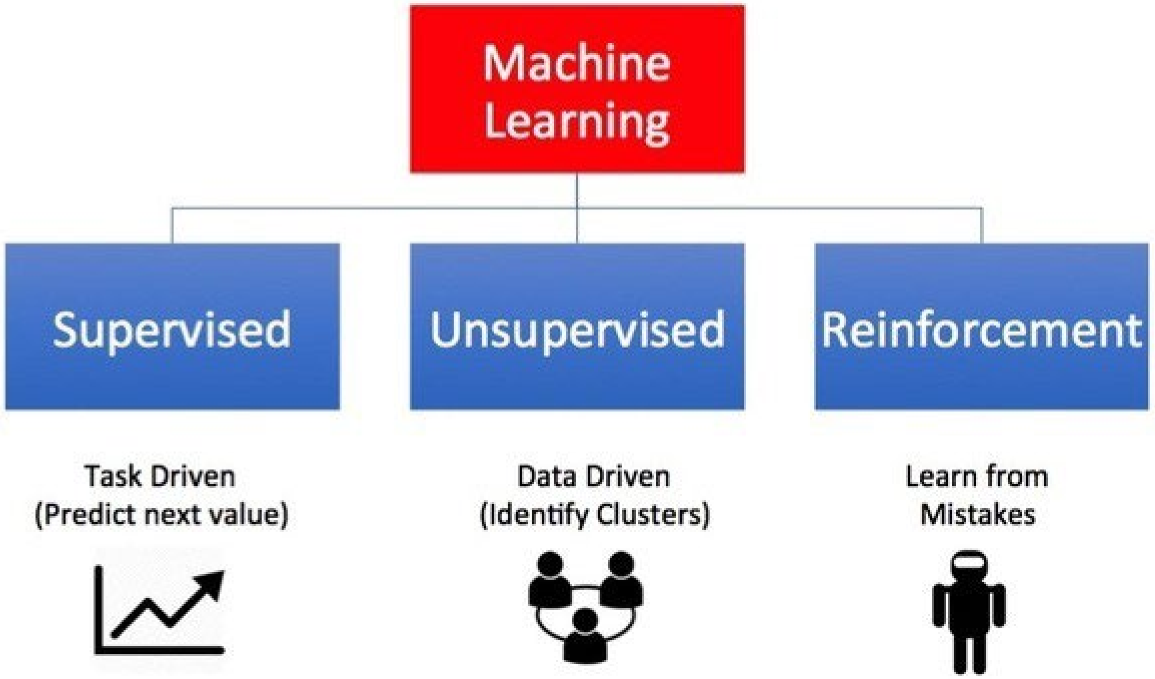
- algorithm design
- data structure
- complexity analysis
- Ex: kd tree

Statistics

- estimation techniques
- theoretical framework
- optimality, efficiency
- Ex: learning theory

slide by Fei Sha

Types of Machine Learning



Machine Learning

Supervised

Unsupervised

Reinforcement

Task Driven
(Predict next value)

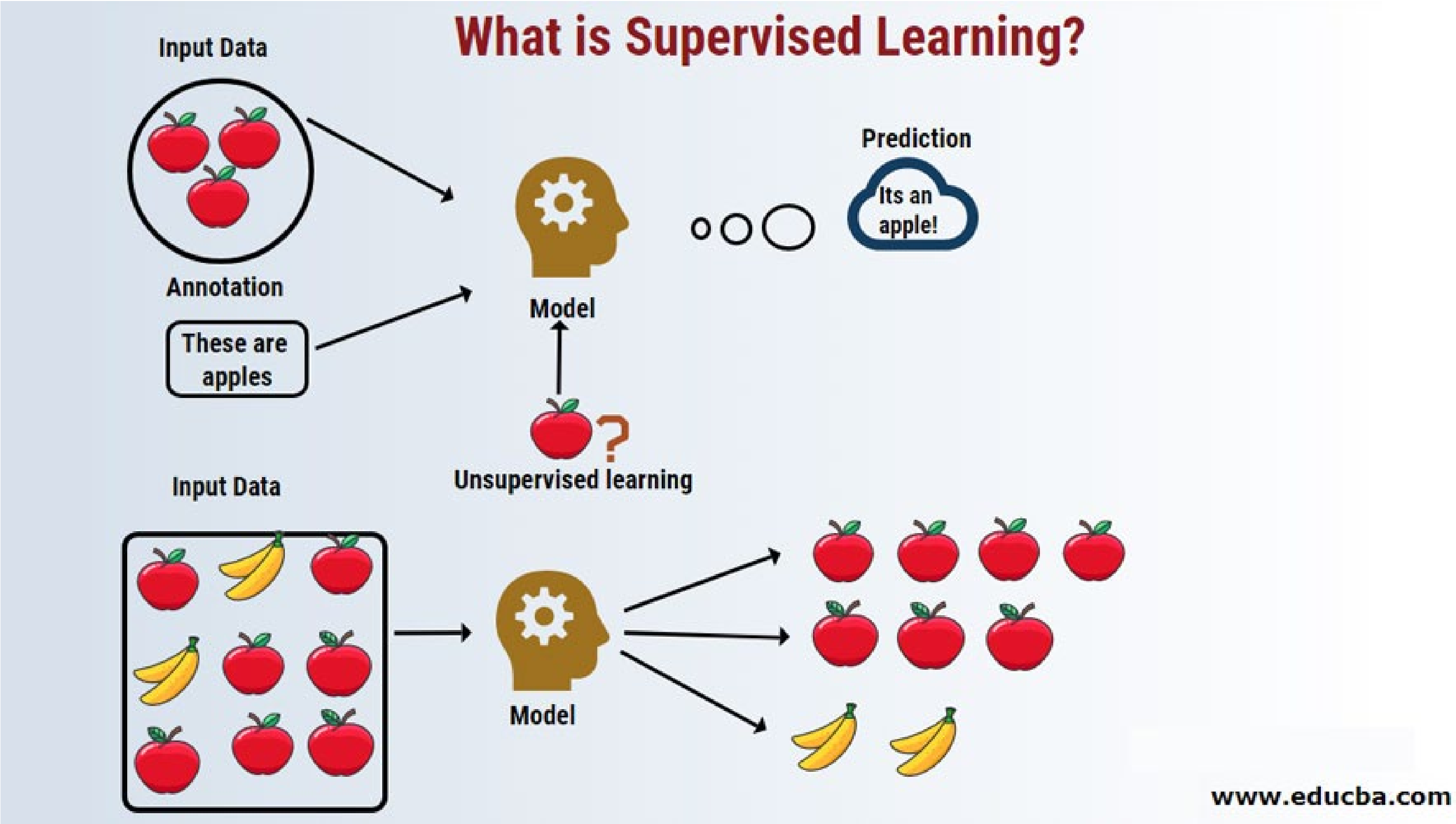
Data Driven
(Identify Clusters)

Learn from
Mistakes

Supervised Learning

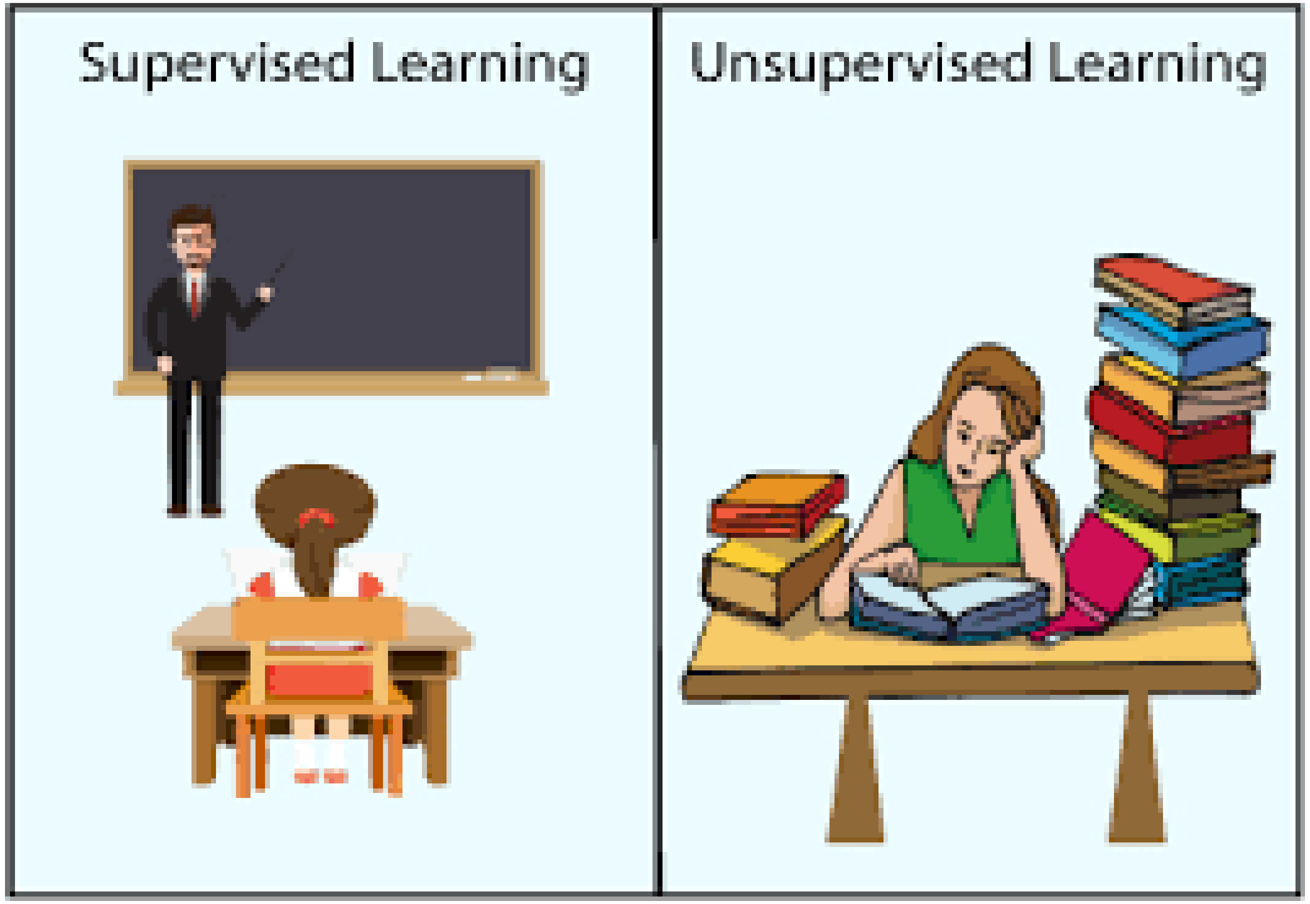
(Most widely used)

1. Classification (e.g. spam filter)
2. Regressor
3. Recommendation System
4. Text Analysis



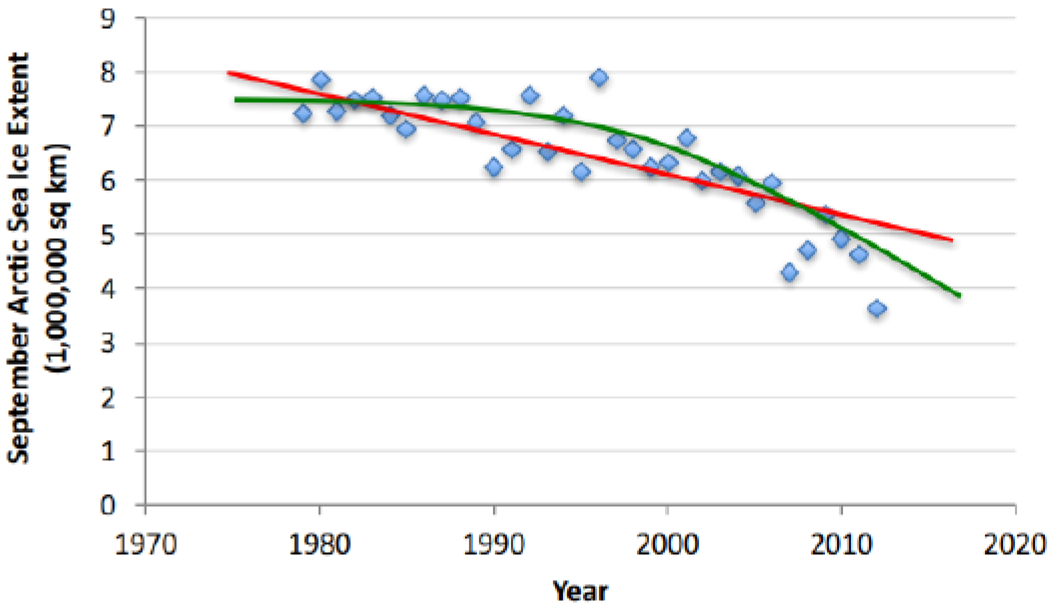
Unsupervised Learning

1. Clustering
2. Low-dimensional manifold
3. Topic modelling
4. Recommendation System


Supervised Learning
Unsupervised Learning

Supervised Learning: Regression

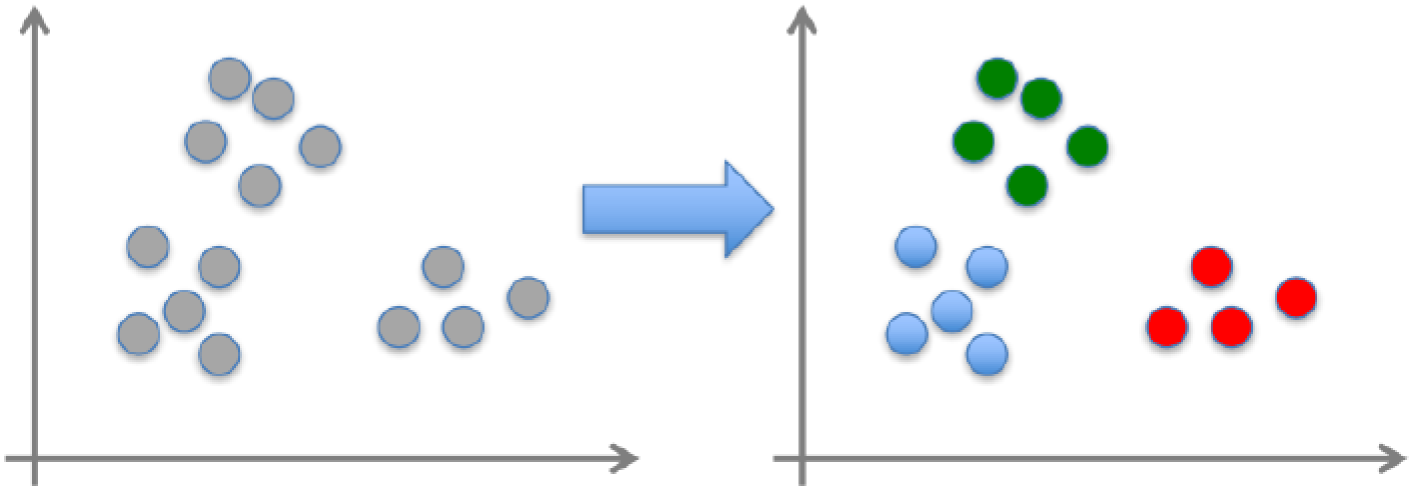
- Given (x_1, y_1), (x_2, y_2), …, (x_n, y_n)
- Learn a function $f(x)$ to predict y given x
  - y is real-valued == regression



Data from G. Witt. Journal of Statistics Education, Volume 21, Number 1 (2013)

Unsupervised Learning

- Given x_1, x_2, …, x_n (without labels)
- Output hidden structure behind the x's
  - E.g., clustering



Semi-supervised Learning
(Also name in reinforcement learning)

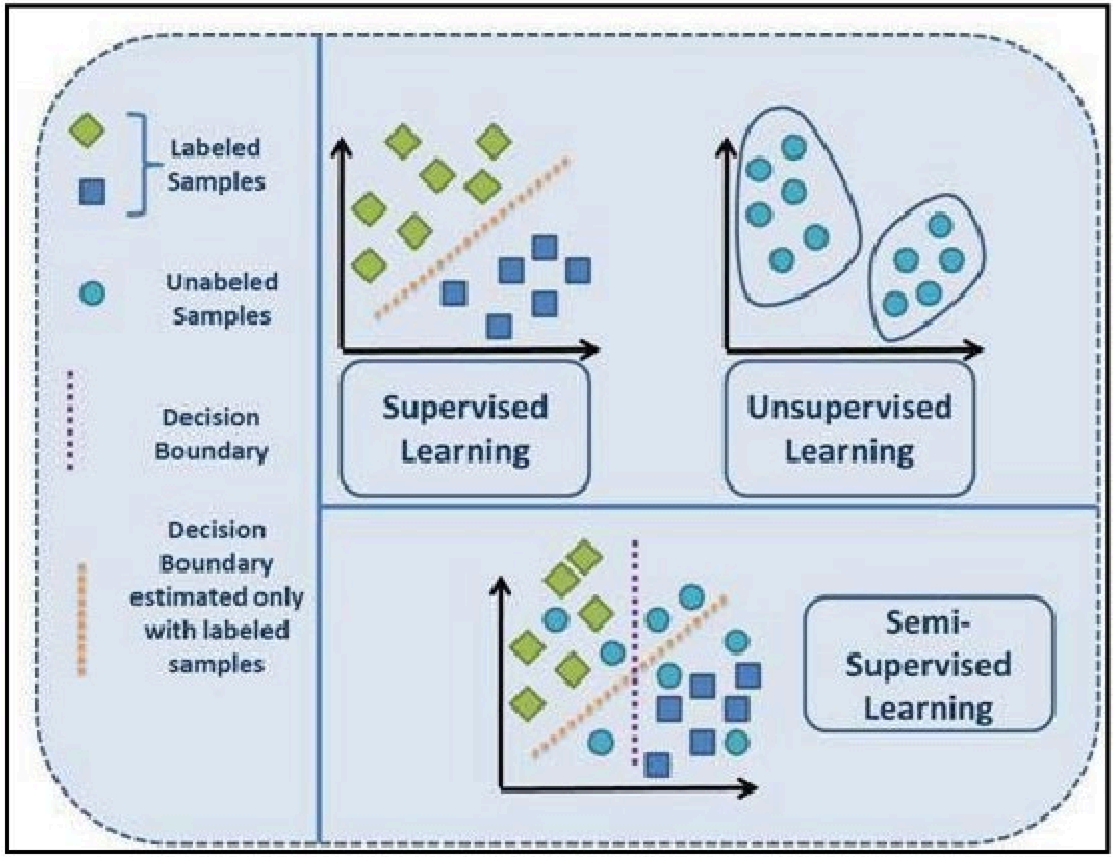
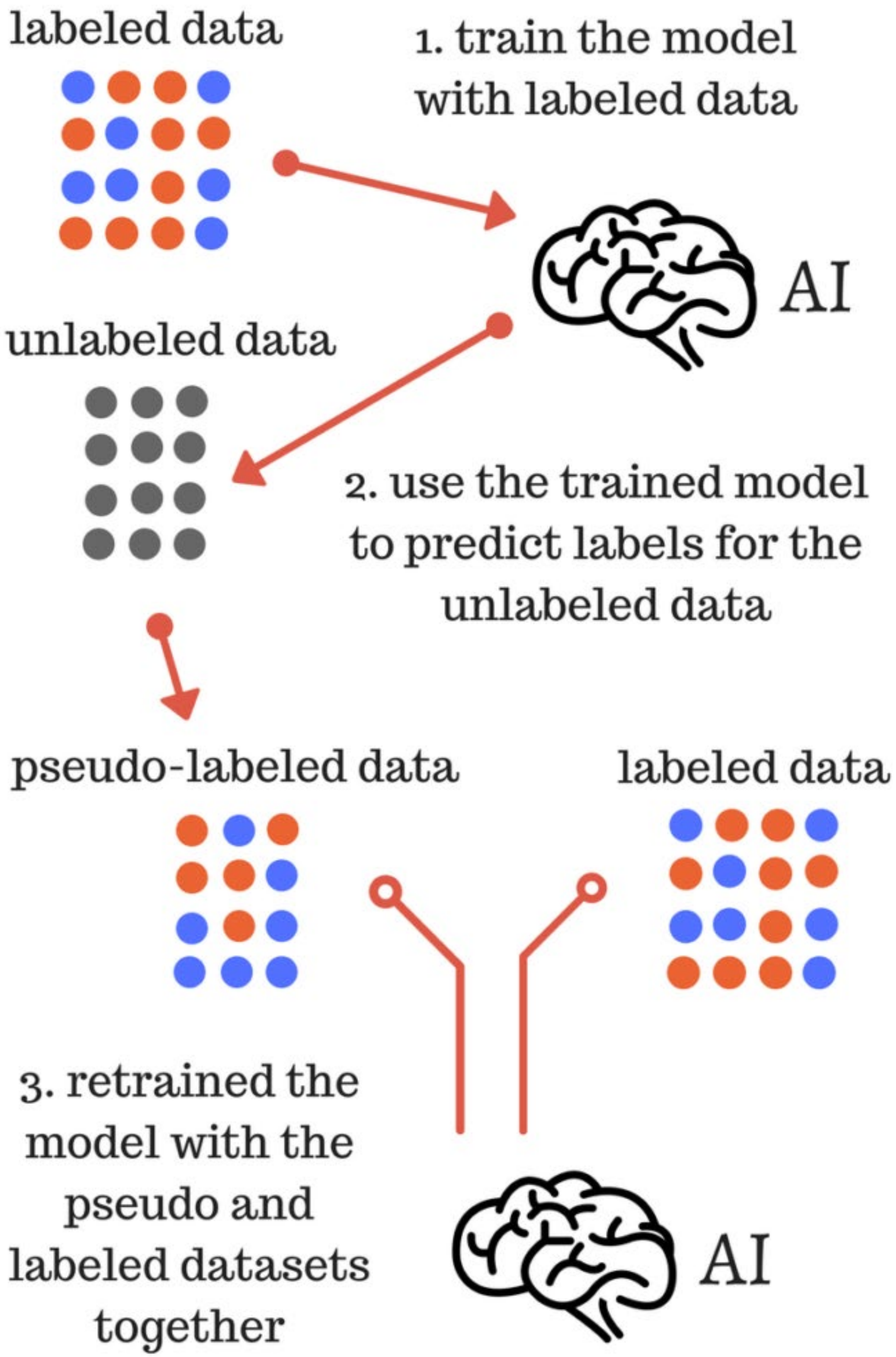
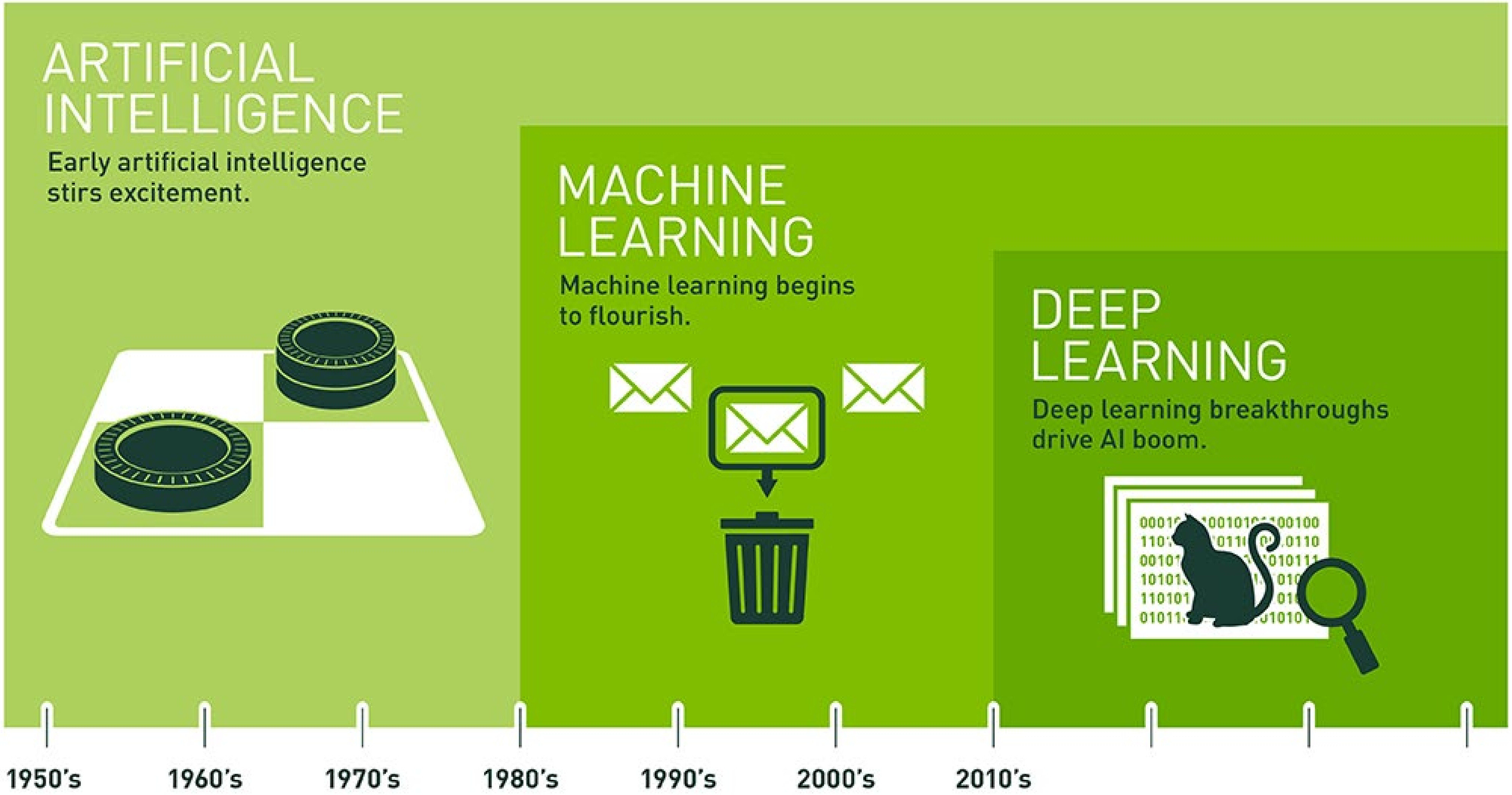


Fig. 1Semi-Supervised Learning


labeled data
1. train the model with labeled data
AI
unlabeled data
2. use the trained model to predict labels for the unlabeled data
pseudo-labeled data
labeled data
3. retrained the model with the pseudo and labeled datasets together
AI

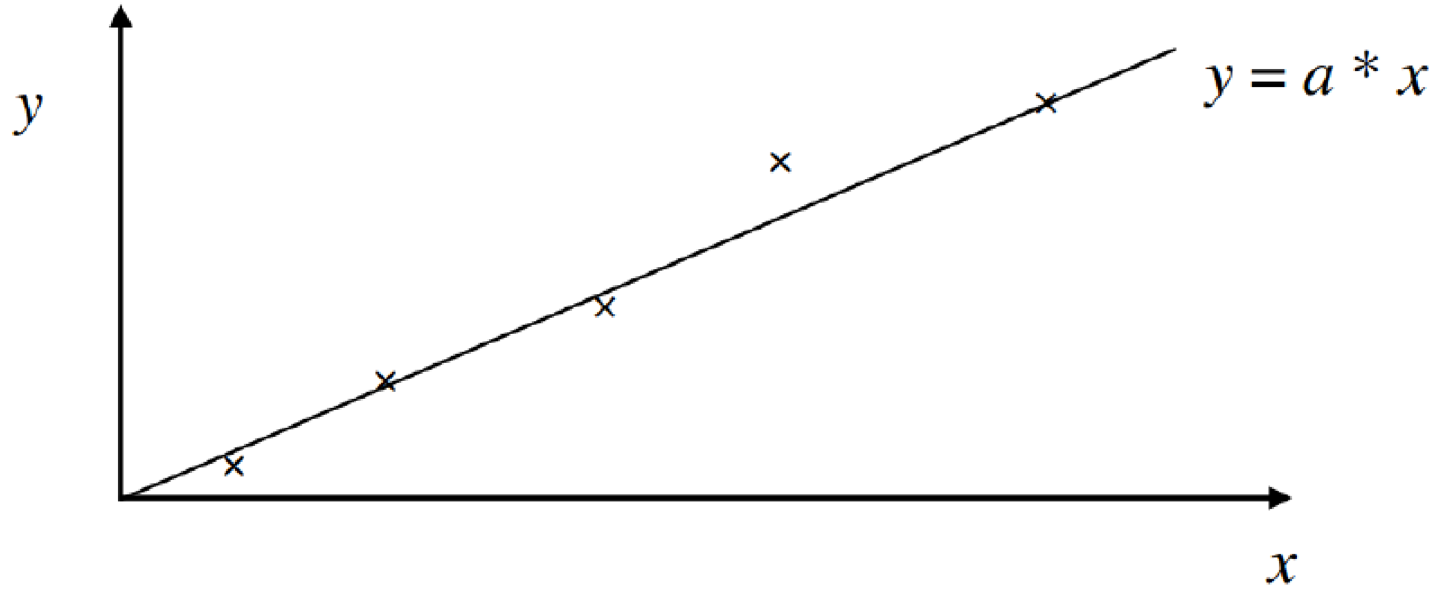
It actually existed decade ago…


ARTIFICIAL INTELLIGENCE
Early artificial intelligence stirs excitement.
MACHINE LEARNING
Machine learning begins to flourish.
DEEP LEARNING
Deep learning breakthroughs drive AI boom.
1950's
1960's
1970's
1980's
1990's
2000's
2010's

Since an early flush of optimism in the 1950s, smaller subsets of artificial intelligence – first machine learning, then deep learning, a subset of machine learning – have created ever larger disruptions.

Empirical Inference

- Drawing conclusions from empirical data (observations, measurements)
- Example 1: scientific inference

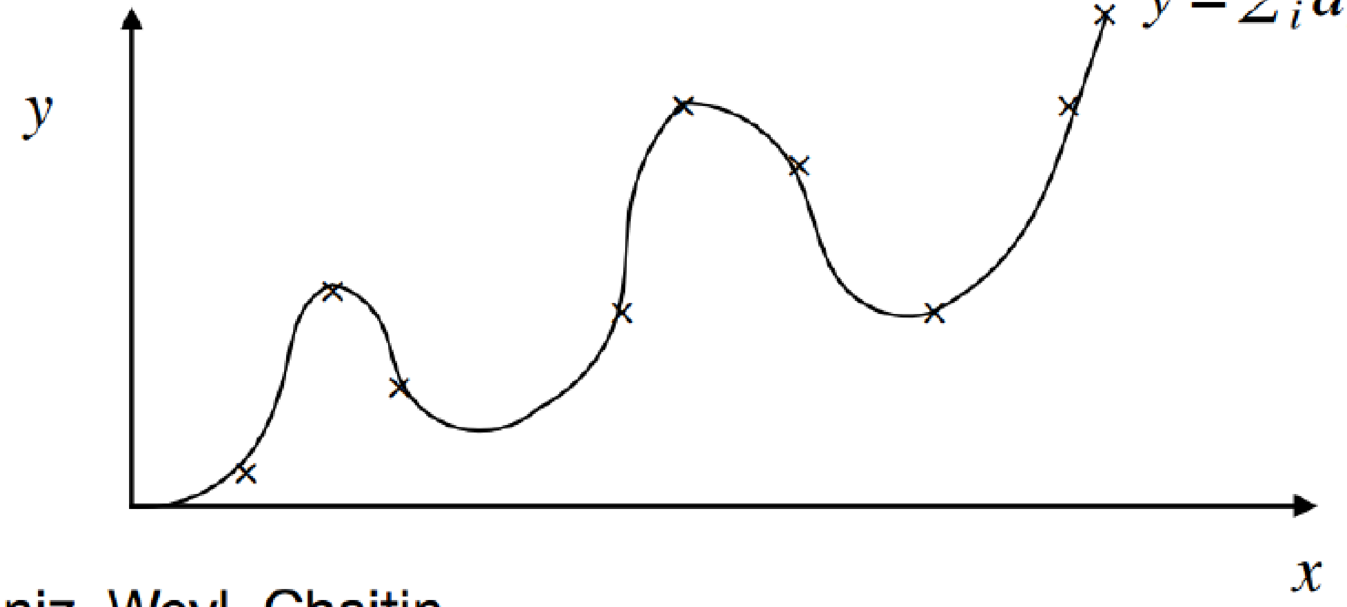


slide by Bernhard Schölkopf

Empirical Inference

- Drawing conclusions from empirical data (observations, measurements)
- Example 1: scientific inference

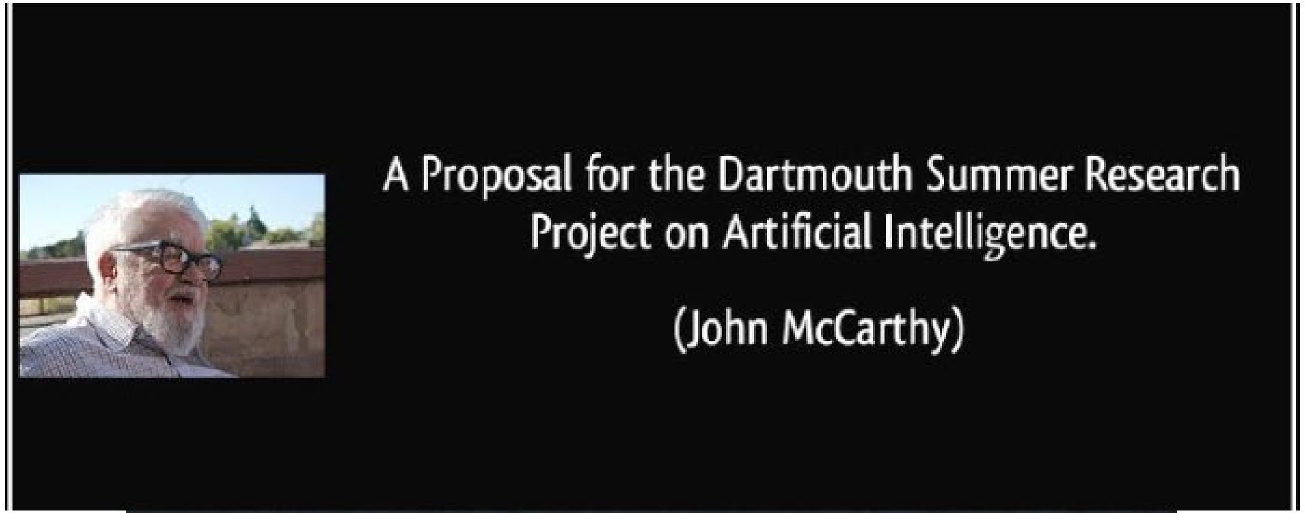
$$y = \Sigma_i \, a_i \, k(x, x_i) + b$$



slide by Bernhard Schölkopf

Leibniz, Weyl, Chaitin

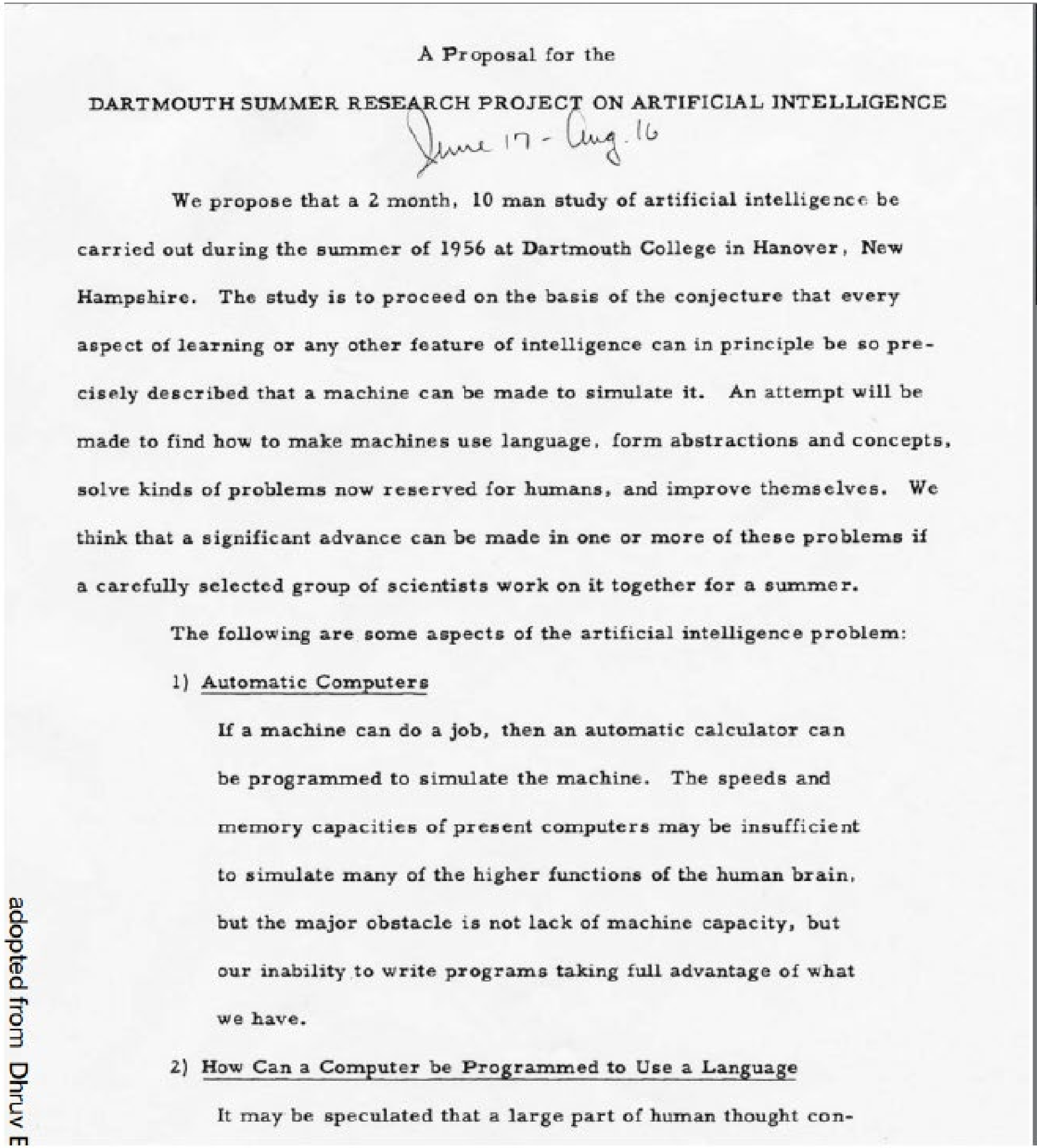
A Brief History of AI



A Proposal for the Dartmouth Summer Research Project on Artificial Intelligence.

(John McCarthy)

slide by Dhruv Batra



A Proposal for the

DARTMOUTH SUMMER RESEARCH PROJECT ON ARTIFICIAL INTELLIGENCE

June 17 - Aug. 16

We propose that a 2 month, 10 man study of artificial intelligence be carried out during the summer of 1956 at Dartmouth College in Hanover, New Hampshire. The study is to proceed on the basis of the conjecture that every aspect of learning or any other feature of intelligence can in principle be so precisely described that a machine can be made to simulate it. An attempt will be made to find how to make machines use language, form abstractions and concepts, solve kinds of problems now reserved for humans, and improve themselves. We think that a significant advance can be made in one or more of these problems if a carefully selected group of scientists work on it together for a summer.

The following are some aspects of the artificial intelligence problem:

1) Automatic Computers

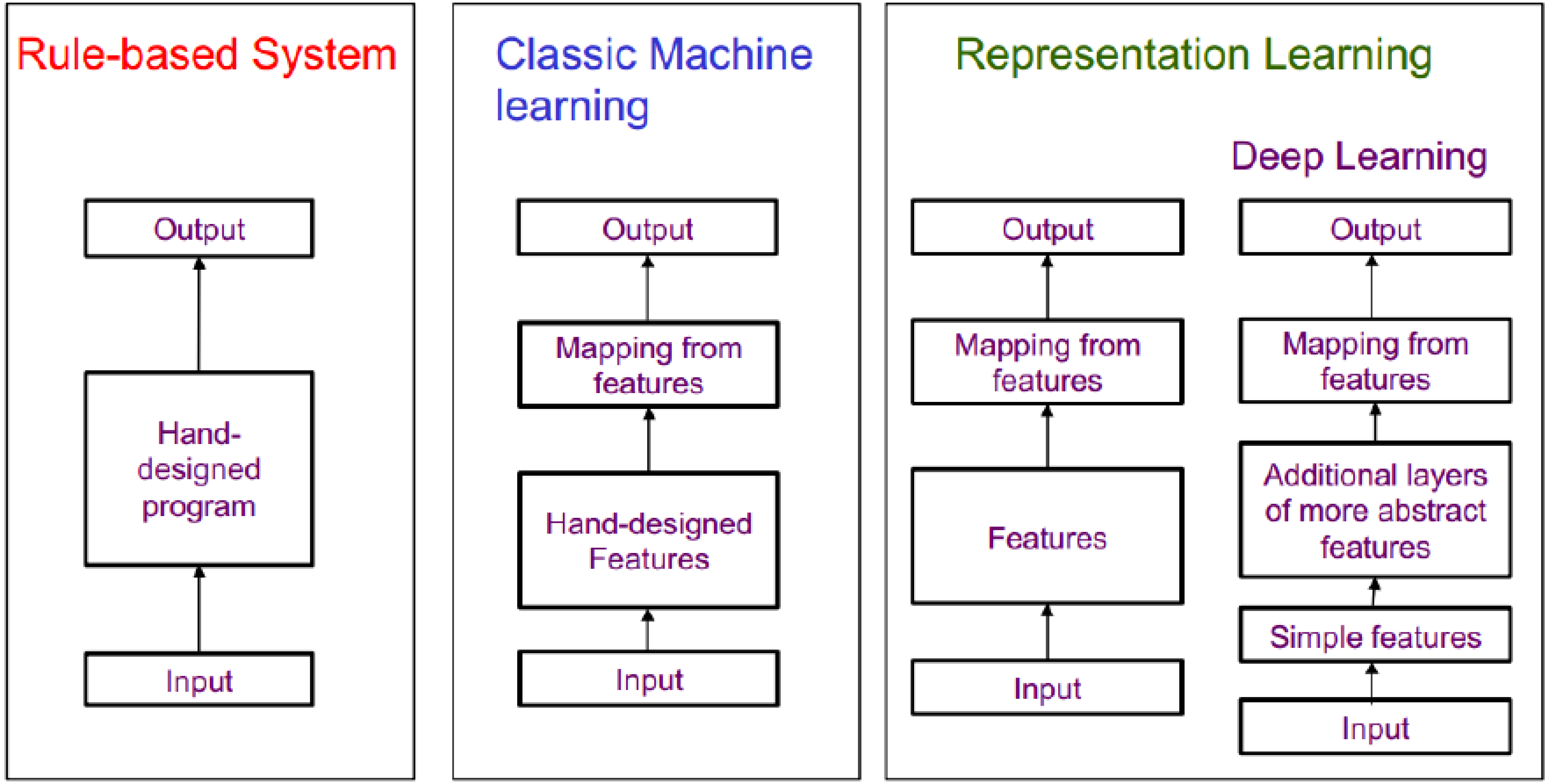
If a machine can do a job, then an automatic calculator can be programmed to simulate the machine. The speeds and memory capacities of present computers may be insufficient to simulate many of the higher functions of the human brain, but the major obstacle is not lack of machine capacity, but our inability to write programs taking full advantage of what we have.

2) How Can a Computer be Programmed to Use a Language

It may be speculated that a large part of human thought con-

adopted from Dhruv E

Summary of AI Models



▨ Shaded boxes indicate components that can learn from data

History of Machine Learning

- Neural Networks (1960)
- Multi-layer Perceptions (1985)
- Restricted Boltzman Machines (1986)
- Support Vector Machine (1995)
- Deep Belief Networks – New interest in deep learning (2005) CNN
- Deep Recurrent Neural Network (2009)
- Convolutional DBN (2010)
- Max Pooling CDBN (2011)

Why are things working today?

- More compute power
- More data
- Better algorithms/ models

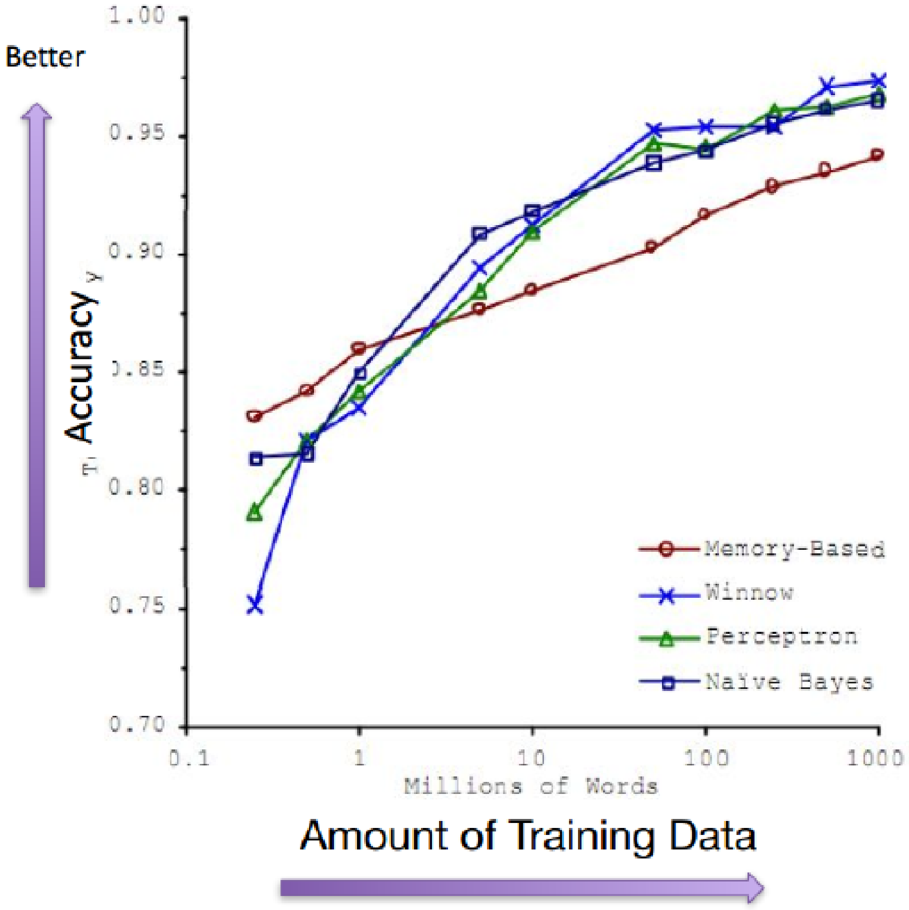


Figure Credit: Banko & Brill, 2011

slide by Dhruv Batra

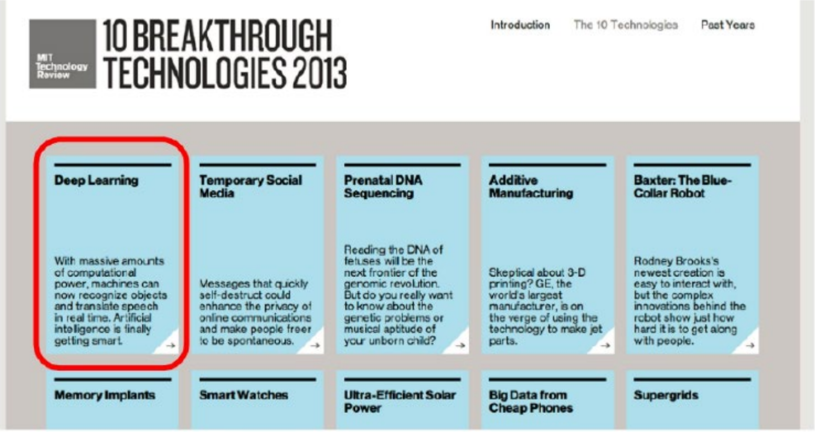
Then, why suddenly got attention?

1. Improve in computer technology (better graphic, GPU, etc)

2. Advent of Big Data

-> Able to train and test more sophisticated model

3. Python



MIT Technology Review

10 BREAKTHROUGH TECHNOLOGIES 2013

Introduction The 10 Technologies Past Years

Deep Learning

With massive amounts of computational power, machines can now recognize objects and translate speech in real time. Artificial intelligence is finally getting smart.

Temporary Social Media

Messages that quickly self-destruct could enhance the privacy of online communications and make people freer to be spontaneous.

Prenatal DNA Sequencing

Reading the DNA of fetuses will be the next frontier of the genomic revolution. But do you really want to know about the genetic problems or musical aptitude of your unborn child?

Additive Manufacturing

Skeptical about 3-D printing? GE, the world's largest manufacturer, is on the verge of using the technology to make jet parts.

Baxter: The Blue-Collar Robot

Rodney Brooks's newest creation is easy to interact with, but the complex innovations behind the robot show just how hard it is to get along with people.

Memory Implants

Smart Watches

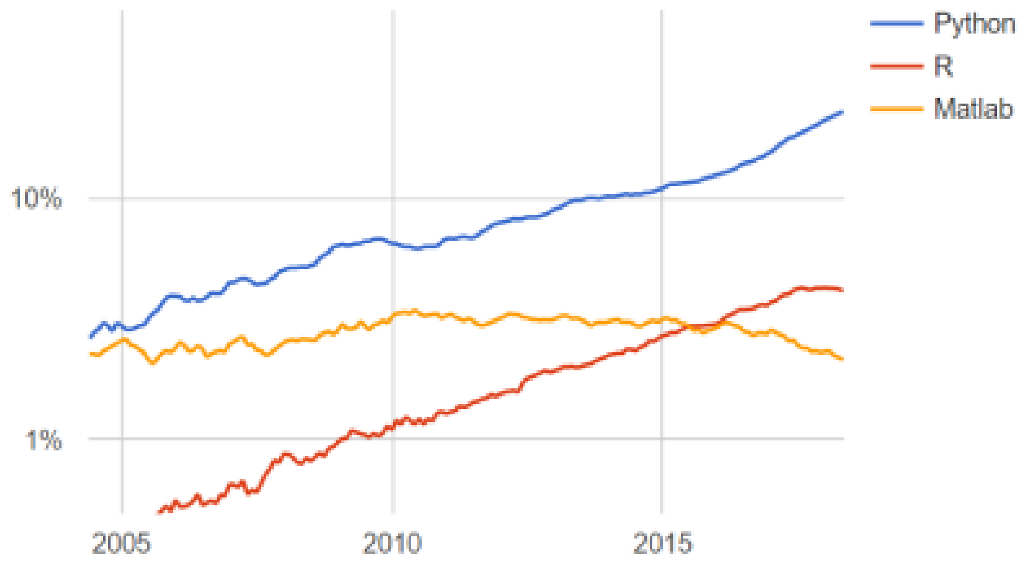
Ultra-Efficient Solar Power

Big Data from Cheap Phones

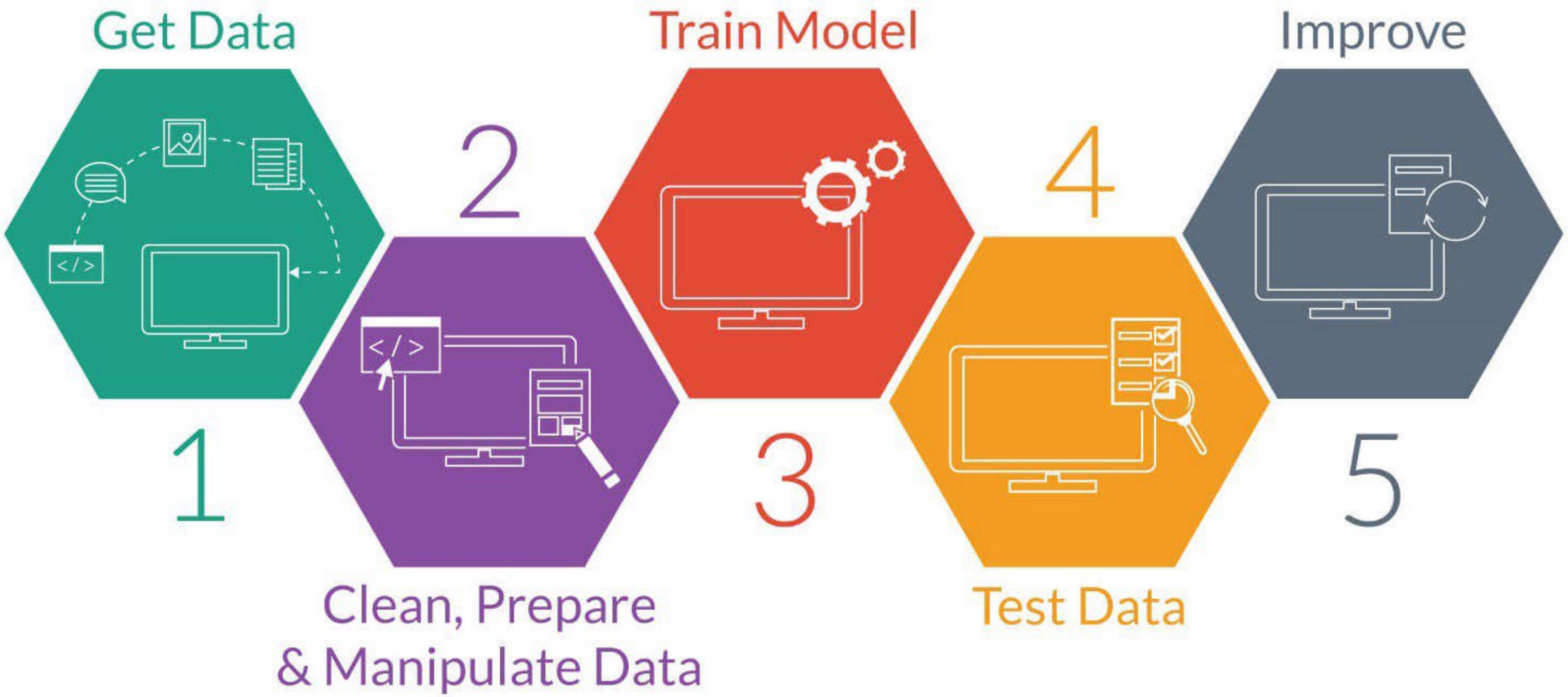
Supergrids

MIT Technology Review, April 23rd, 2013

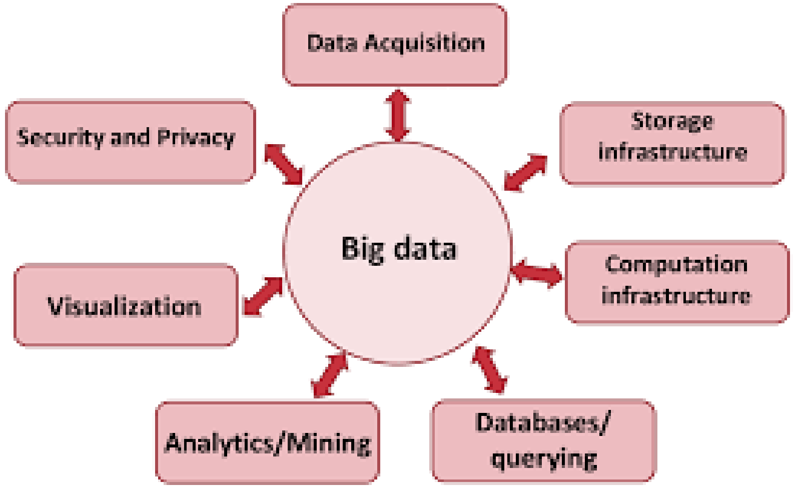
PYPL PopularitY of Programming Language

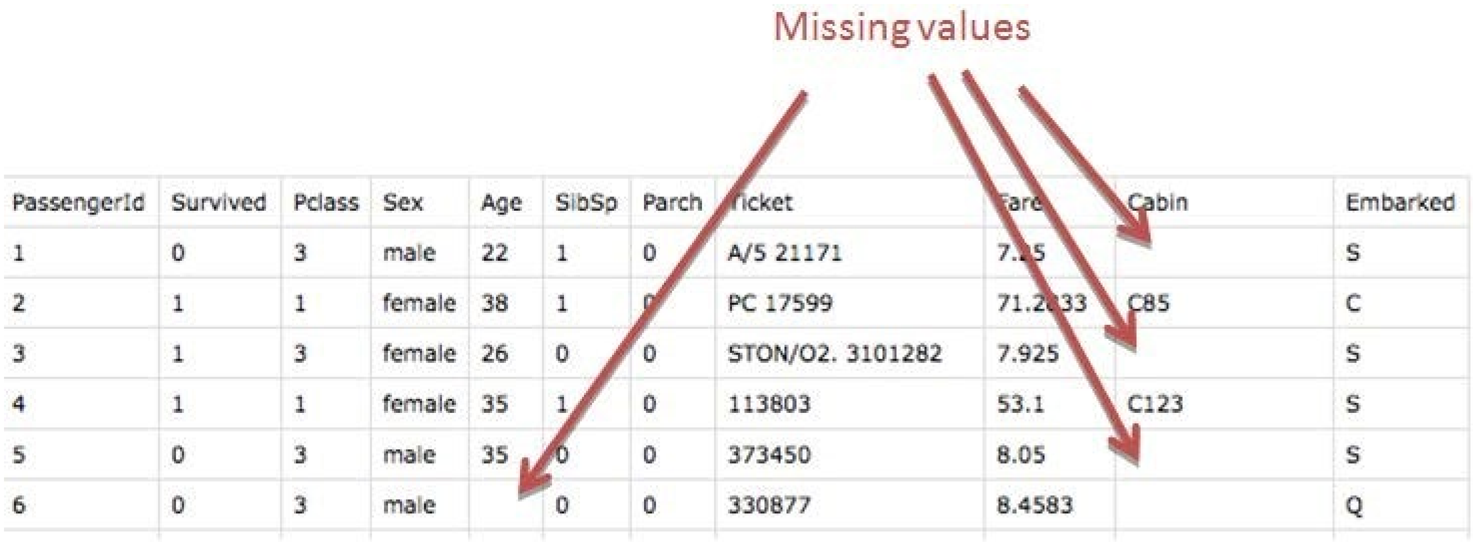


ML Process (1)


Get Data
1
2
Clean, Prepare
& Manipulate Data
Train Model
3
4
Test Data
Improve
5

ML Process (2) Step 1&2



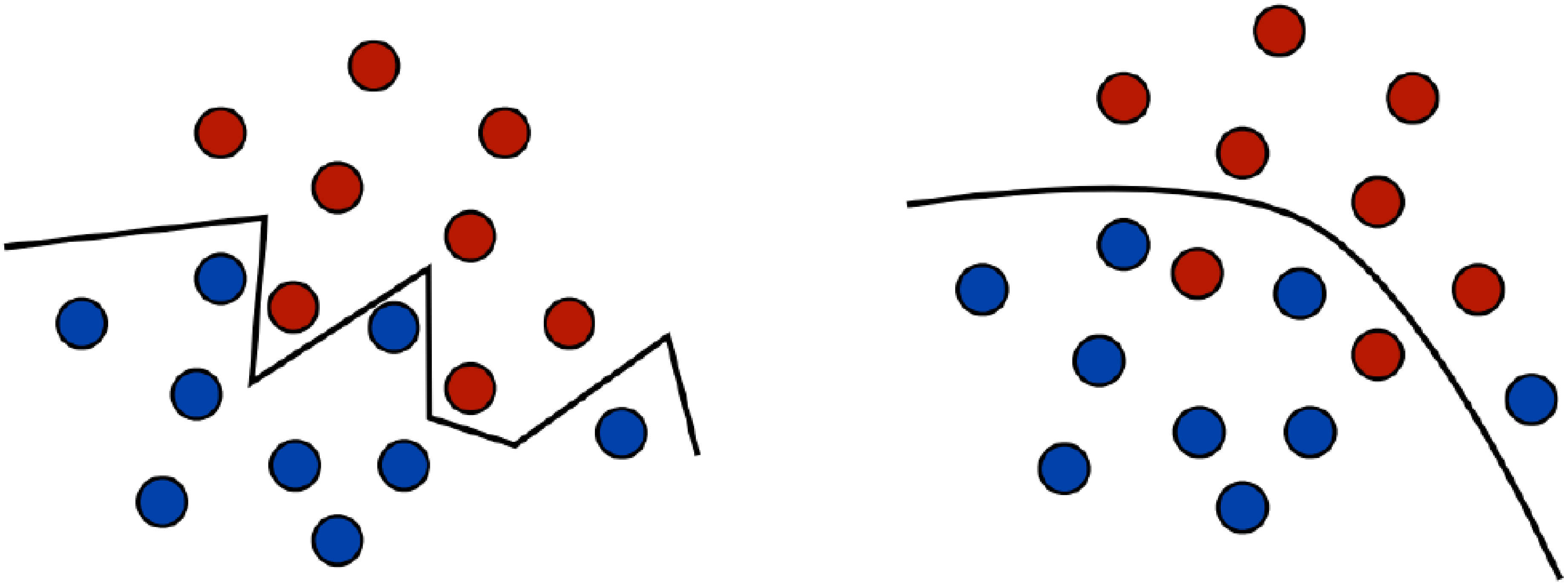

Missing values

PassengerId	Survived	Pclass	Sex	Age	SibSp	Parch	Ticket	Fare	Cabin	Embarked
1	0	3	male	22	1	0	A/5 21171	7.25		S
2	1	1	female	38	1	0	PC 17599	71.2833	C85	C
3	1	3	female	26	0	0	STON/O2. 3101282	7.925		S
4	1	1	female	35	1	0	113803	53.1	C123	S
5	0	3	male	35	0	0	373450	8.05		S
6	0	3	male		0	0	330877	8.4583		Q

1) Where do we get Big but with **good density** data?

2) How are we going to replace missing values?
-> Drop? Fill up? Optimization?

Learning ≠ Fitting



Notion of simplicity/complexity.

→ How do we define complexity?

Learning Problem Definition

- Improving some measure of performance P when executing some task T through some type of training experience E
- Example: Learning to detect credit card fraud
- **Task T**
  - Assign label of fraud or not fraud to credit card transaction

- **Performance measure P**
  - Accuracy of fraud classifier

  With higher penalty when fraud is labeled as not fraud
- **Training experience E**
  - Historical credit card transactions labeled as fraud or not

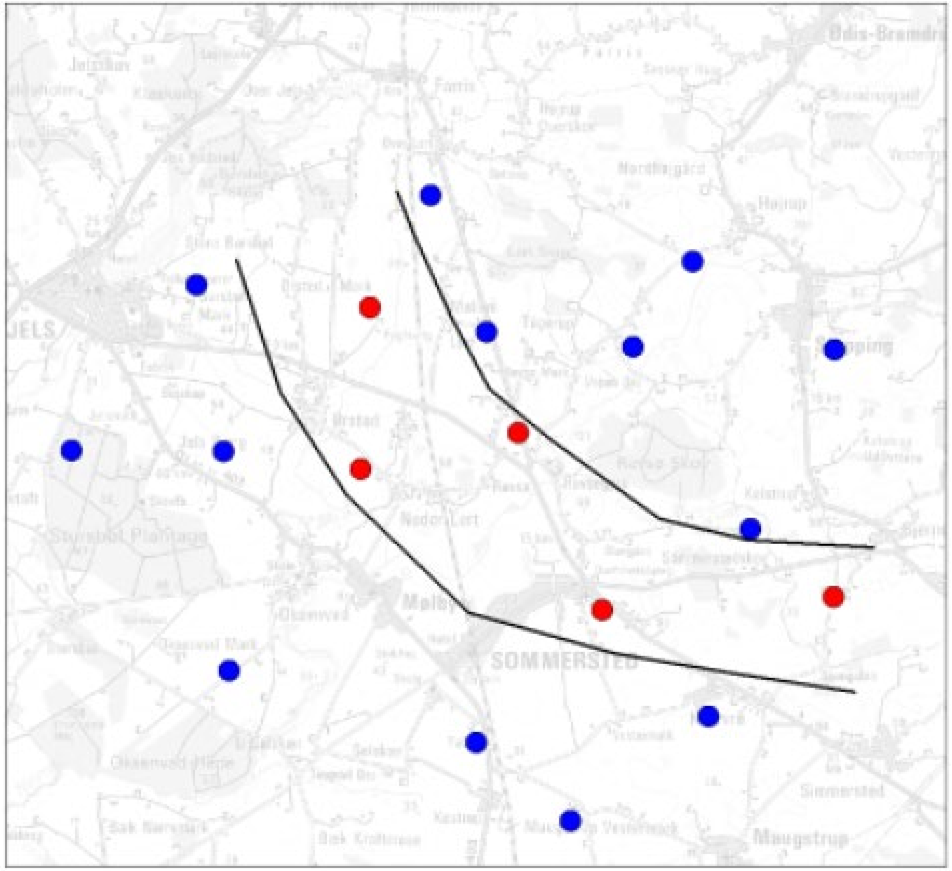
Good Data Density?

Spatial data density

High data density is crucial for reliable mapping results !

Suppose you have indications of buried valleys visible in some of your boreholes (red color)

– possible conclusion 1



Good Data Density?

Spatial data density

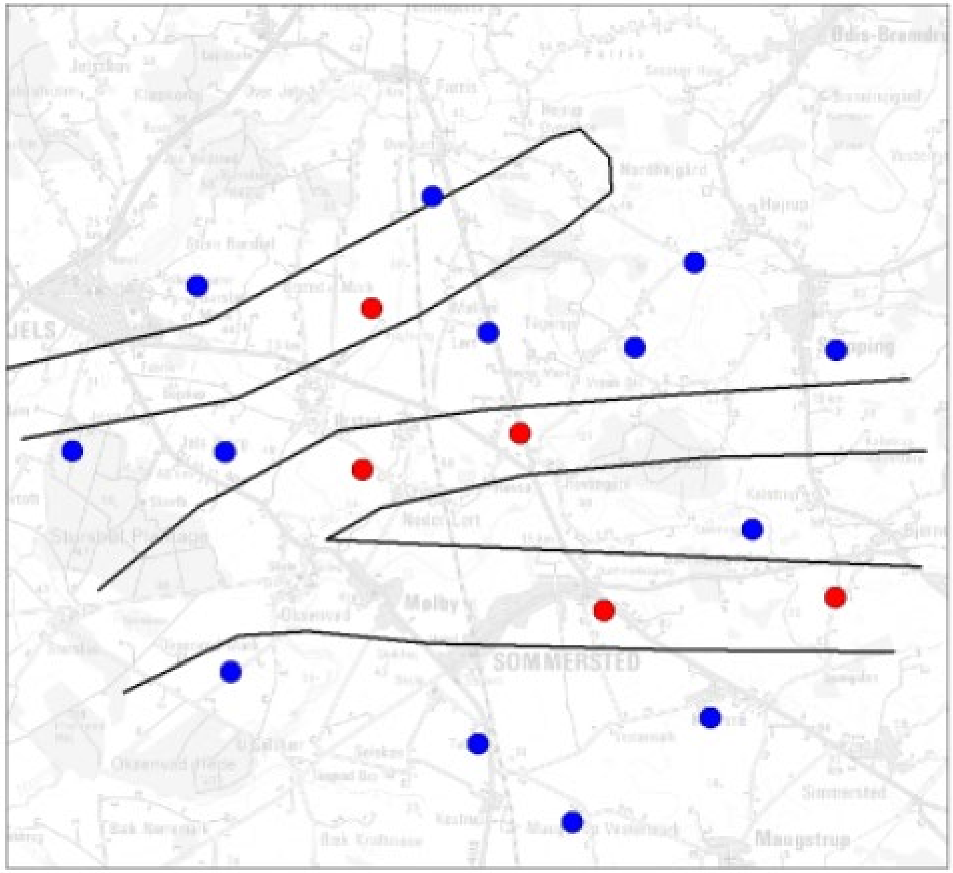
High data density is crucial for reliable mapping results !

Suppose you have indications of buried valleys visible in some of your boreholes (red color)

– possible conclusion 1

– possible conclusion 2

Which one is the most likely?



Good Data Density?

Spatial data density

High data density is crucial for reliable mapping results !

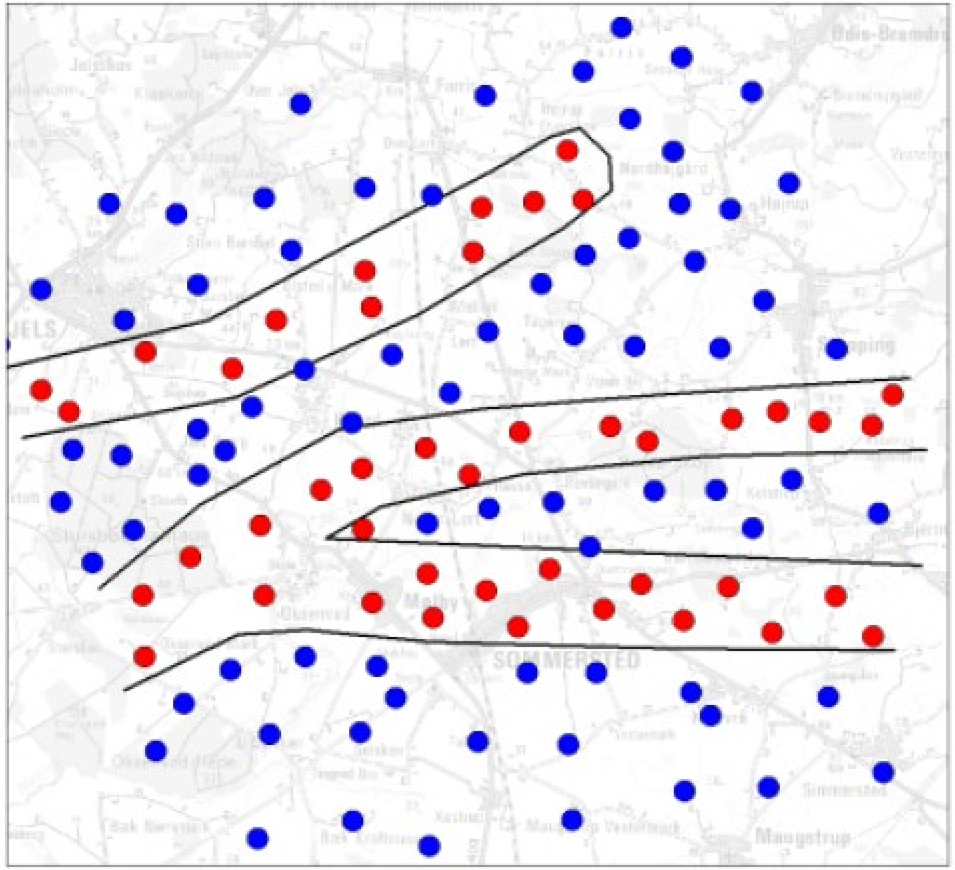
Suppose you have indications of buried valleys visible in some of your boreholes (red color)

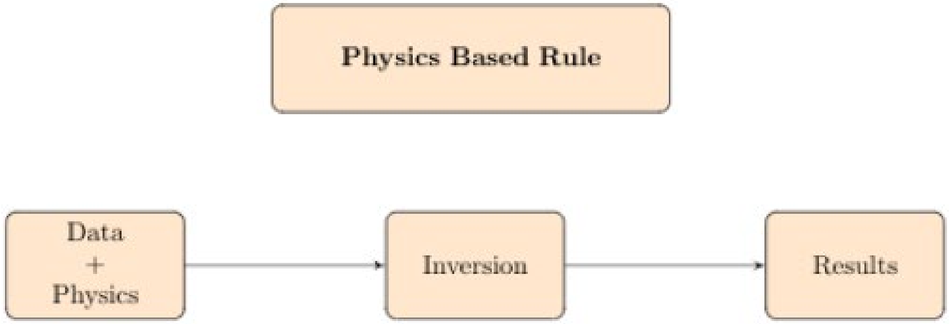
– possible conclusion 1

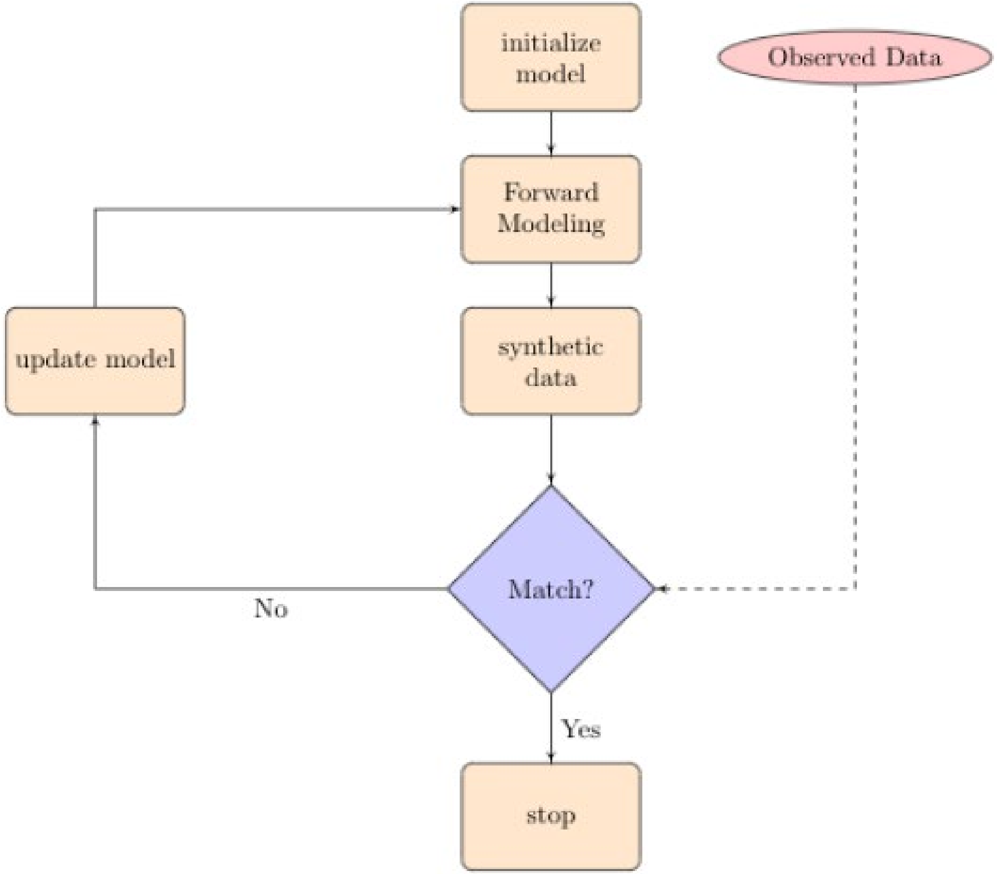
– possible conclusion 2

Which one is the most likely?

A good data density makes it easier to decide!







- **Data Vector**

$$\mathbf{d} = [d_1, d_2, .., d_N]^T$$

- **Model Vector**

$$\mathbf{m} = [m_1, m_2, .., m_M]^T$$

- **In general**

$$N \neq M.$$

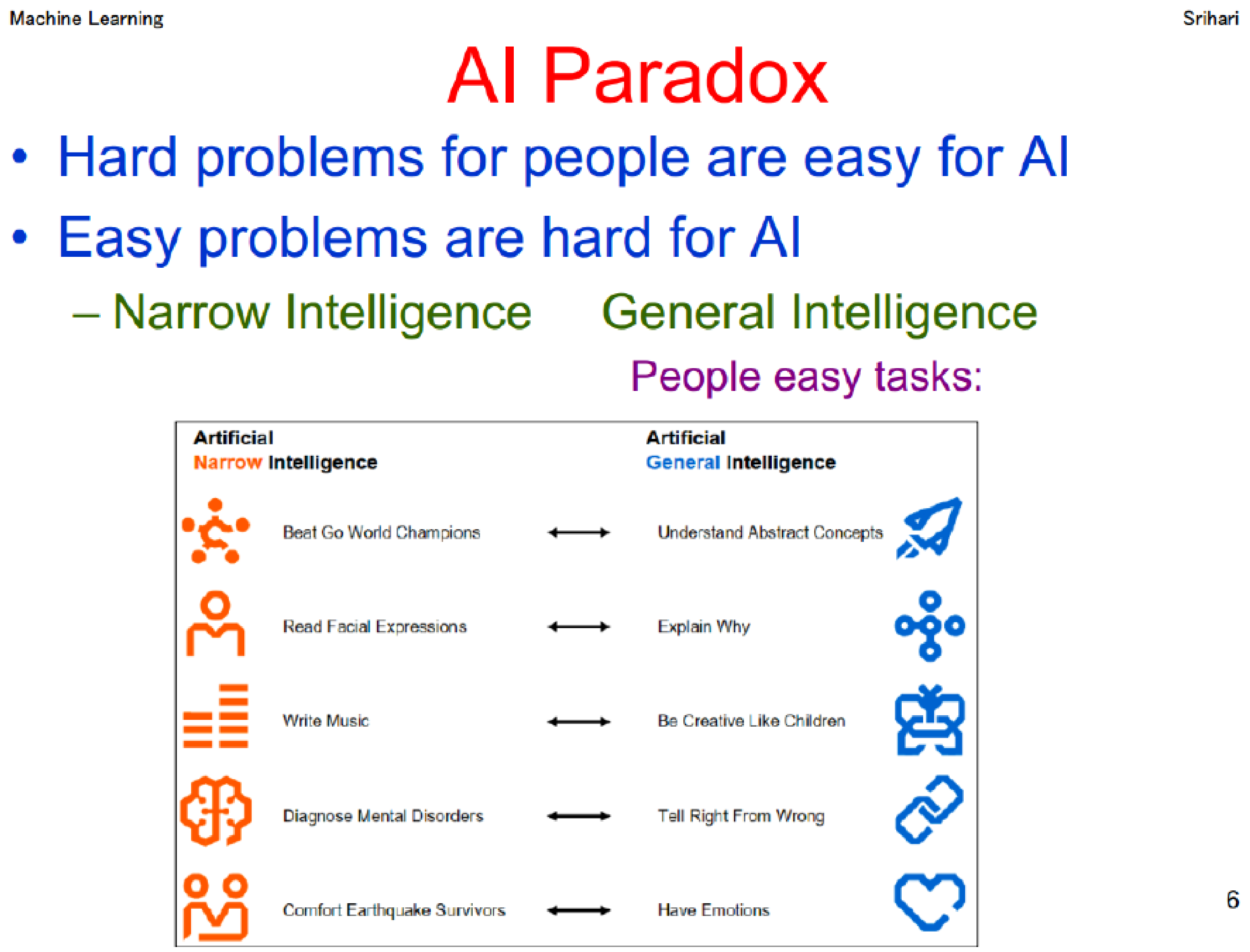
- **Forward Modeling**

$$\mathbf{d} = g(\mathbf{m})$$

- **Linear Problem**

$$\mathbf{d} = \mathbf{G}\mathbf{m}$$

Paradox of AI



AI Paradox

- Hard problems for people are easy for AI
- Easy problems are hard for AI
  - Narrow Intelligence General Intelligence

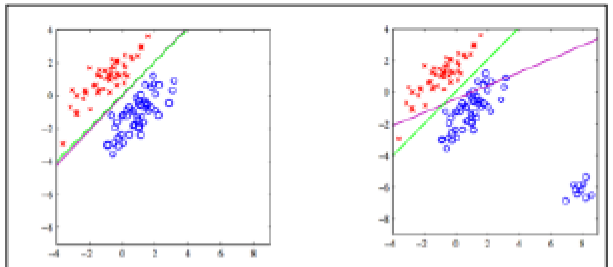
People easy tasks:

Artificial Narrow Intelligence		Artificial General Intelligence
Beat Go World Champions	⟷	Understand Abstract Concepts
Read Facial Expressions	⟷	Explain Why
Write Music	⟷	Be Creative Like Children
Diagnose Mental Disorders	⟷	Tell Right From Wrong
Comfort Earthquake Survivors	⟷	Have Emotions

The Machine Learning approach

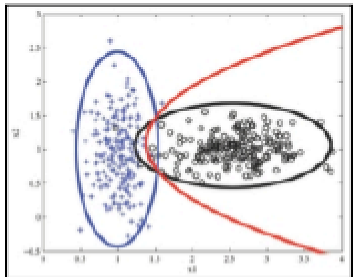
- Difficulties of hard-coded approach suggests:
  - Allow computers to learn from experience
- First determine what features to use
- Learn to map the features to outputs

Linear classifier



Magenta: Least Squares
Green: Logistic Regression (more robust)

Quadratic classifier



2D binary classification with Naive Bayes. A density contour is drawn for the Gaussian model of each class and the decision boundary is shown in red

Session 2

2. Set up Anaconda & Python

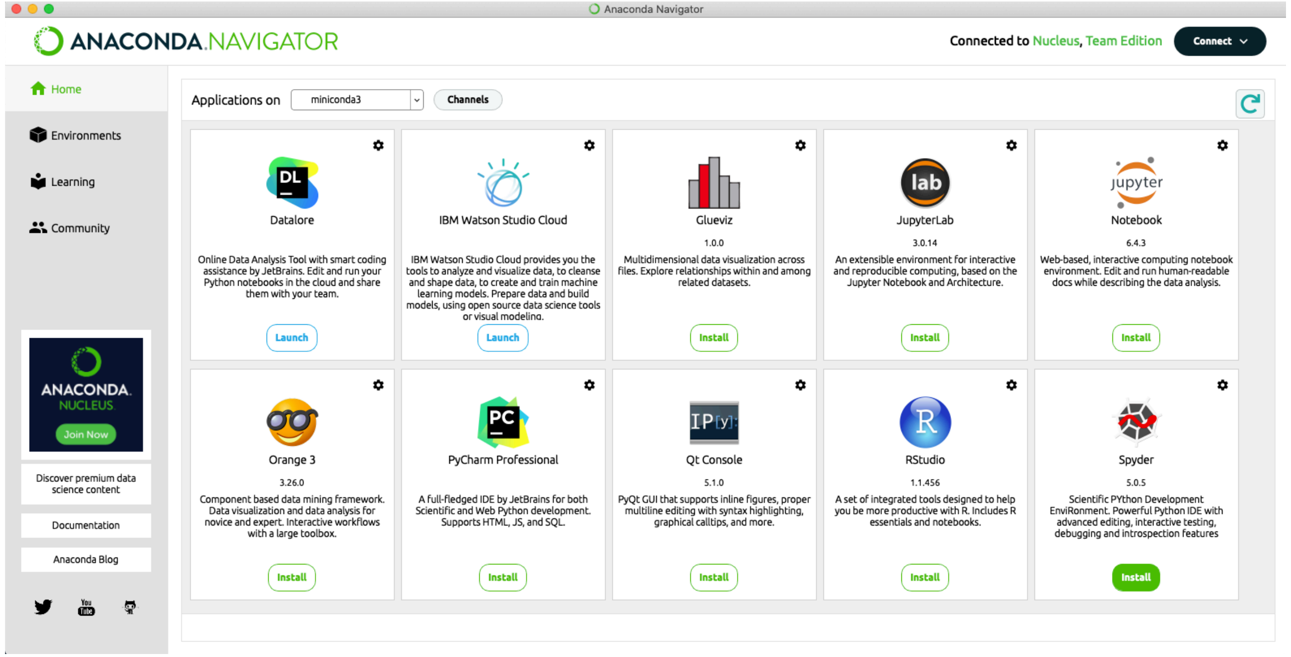
https://www.anaconda.com/products/individual

Set Up

Install Anaconda
Install jupyter notebook

(For week 1-3, we are using jupyter notebook)

Install recent python



Datalore

Online Data Analysis Tool with smart coding assistance by JetBrains. Edit and run your Python notebooks in the cloud and share them with your team.

IBM Watson Studio Cloud

IBM Watson Studio Cloud provides you the tools to analyze and visualize data, to cleanse and shape data, to create and train machine learning models. Prepare data and build models, using open source data science tools or visual modeling.

Glueviz 1.0.0

Multidimensional data visualization across files. Explore relationships within and among related datasets.

JupyterLab 3.0.14

An extensible environment for interactive and reproducible computing, based on the Jupyter Notebook and Architecture.

Notebook 6.4.3

Web-based, interactive computing notebook environment. Edit and run human-readable docs while describing the data analysis.

Orange 3 3.26.0

Component based data mining framework. Data visualization and data analysis for novice and expert. Interactive workflows with a large toolbox.

PyCharm Professional

A full-fledged IDE by JetBrains for both Scientific and Web Python development. Supports HTML, JS, and SQL.

Qt Console 5.1.0

PyQt GUI that supports inline figures, proper multiline editing with syntax highlighting, graphical calltips, and more.

RStudio 1.1.456

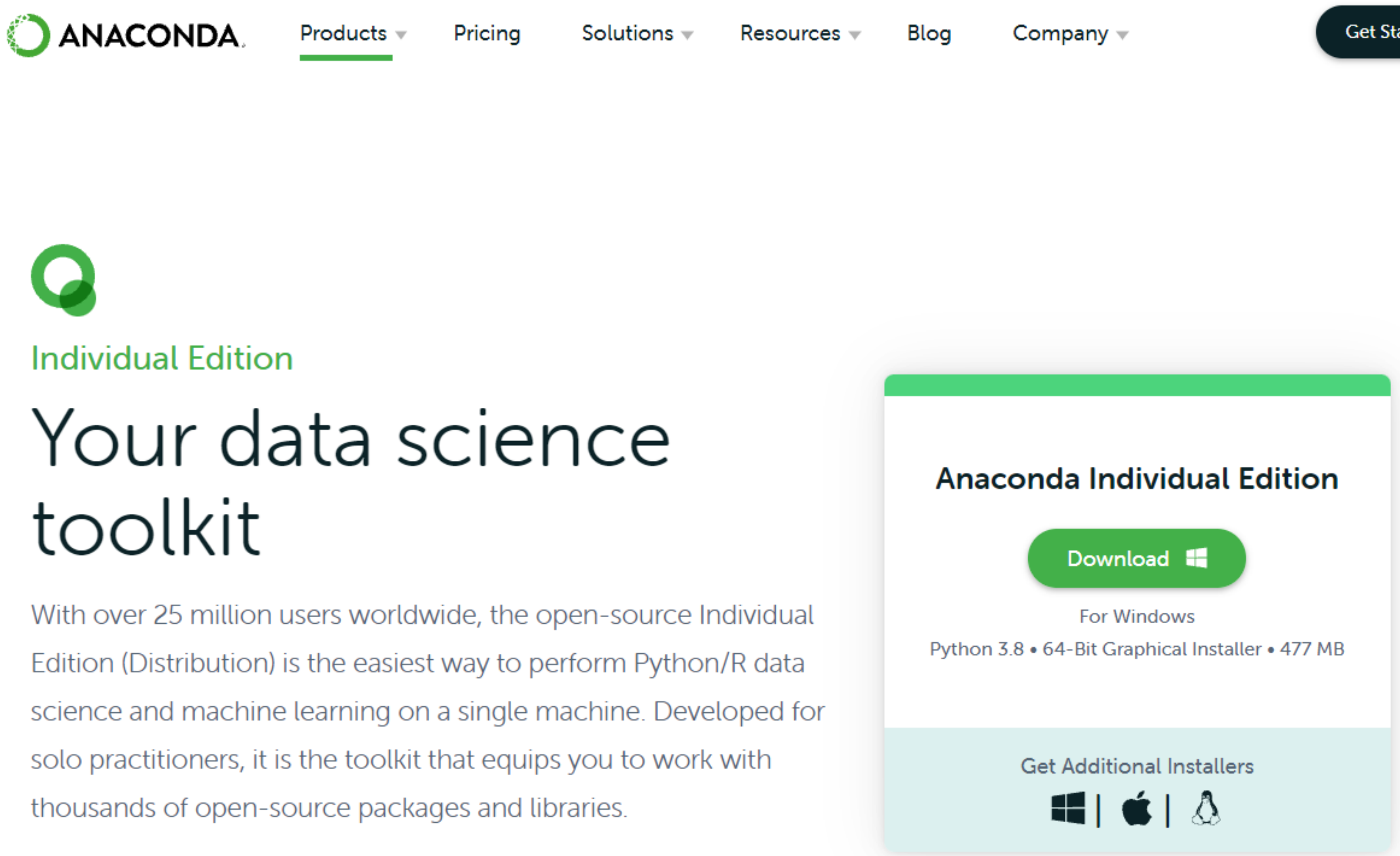
A set of integrated tools designed to help you be more productive with R. Includes R essentials and notebooks.

Spyder 5.0.5

Scientific PYthon Development EnviRonment. Powerful Python IDE with advanced editing, interactive testing, debugging and introspection features

Code easily and link to terminal

Google Colab is also useful



Individual Edition

Your data science toolkit

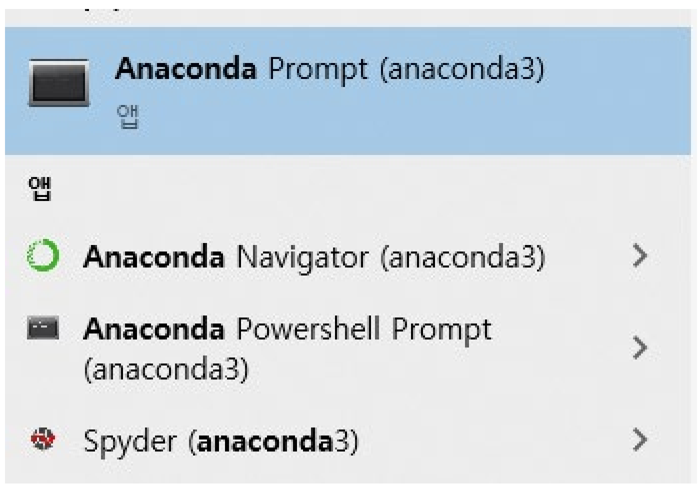
With over 25 million users worldwide, the open-source Individual Edition (Distribution) is the easiest way to perform Python/R data science and machine learning on a single machine. Developed for solo practitioners, it is the toolkit that equips you to work with thousands of open-source packages and libraries.

Anaconda Individual Edition

For Windows

Python 3.8 • 64-Bit Graphical Installer • 477 MB

After Installation


Anaconda Prompt (anaconda3)

앱

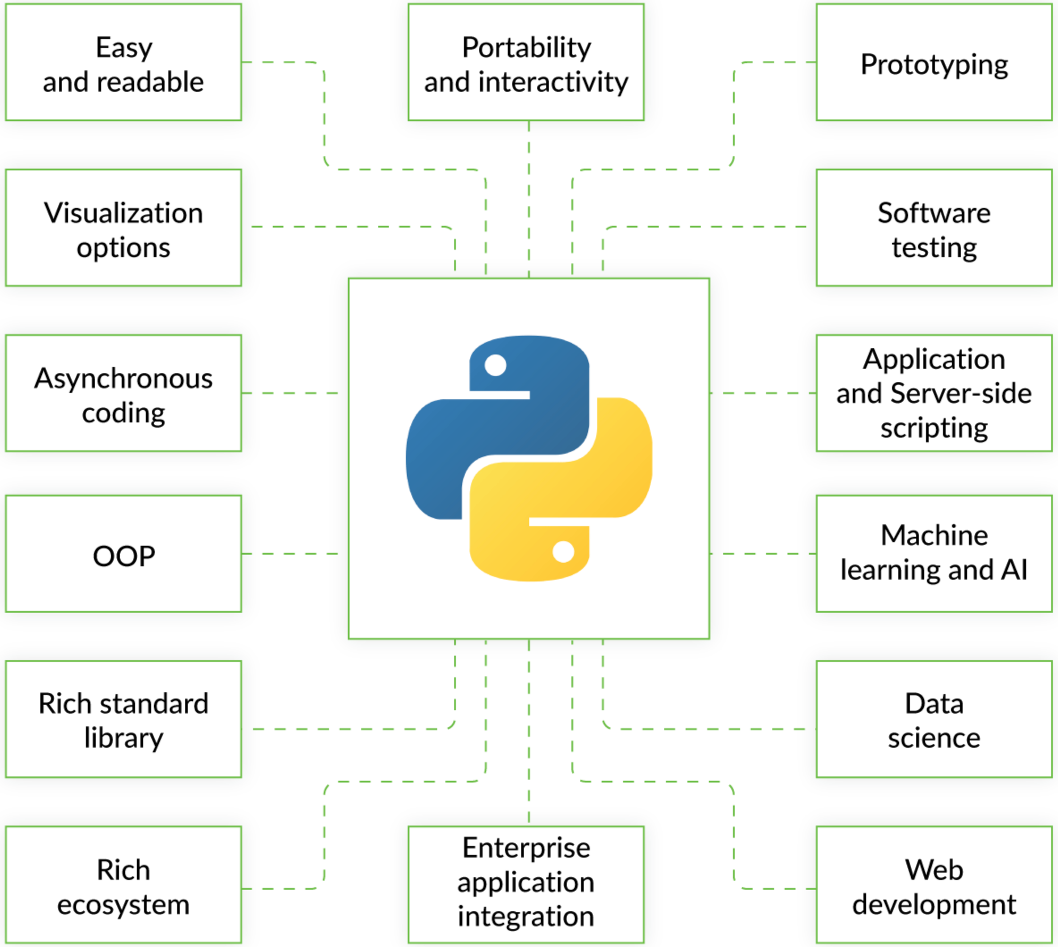
앱

Anaconda Navigator (anaconda3)

Anaconda Powershell Prompt (anaconda3)

Spyder (anaconda3)

Why Python?



Easy and readable

Portability and interactivity

Prototyping

Visualization options

Software testing

Asynchronous coding

Application and Server-side scripting

OOP

Machine learning and AI

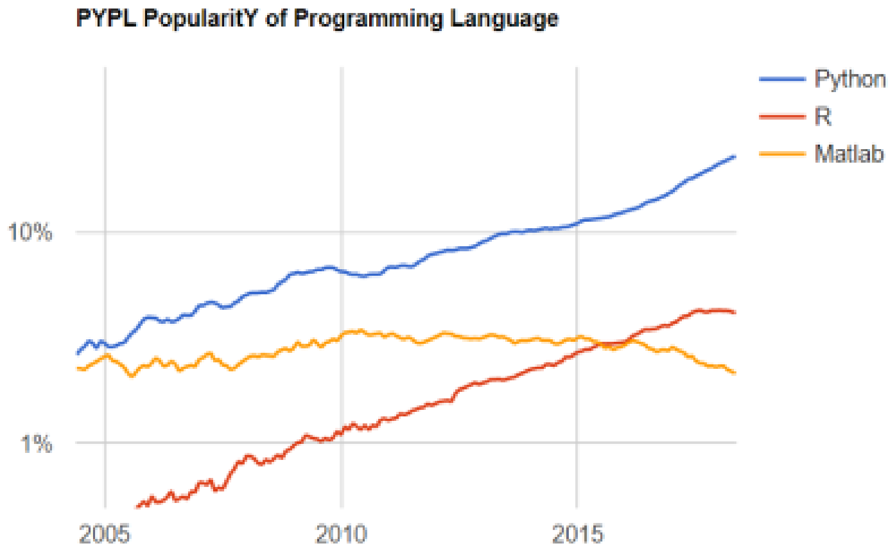
Rich standard library

Data science

Rich ecosystem

Enterprise application integration

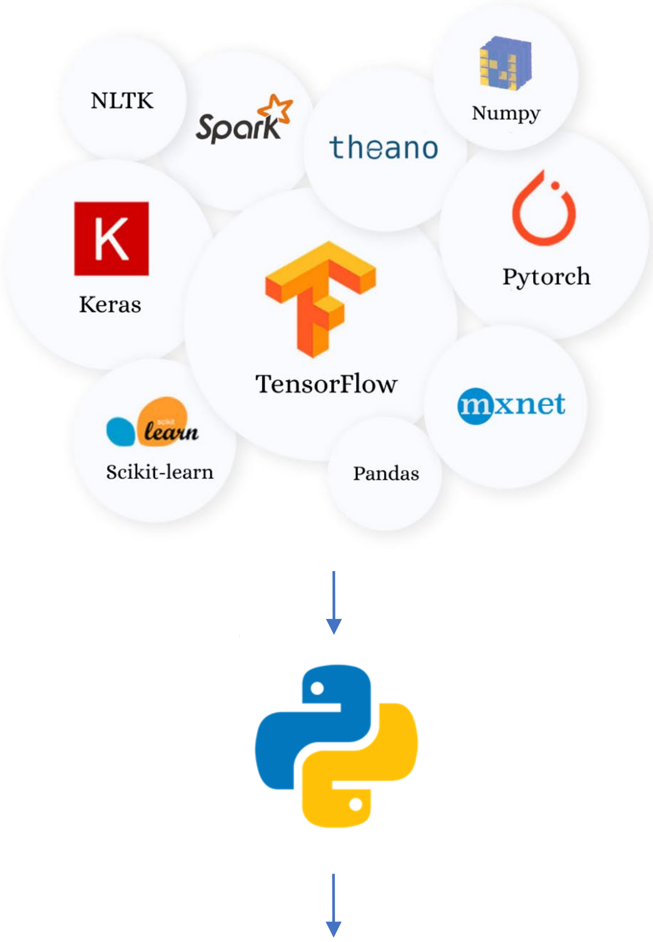
Web development



PYPL PopularitY of Programming Language

Computer scientists and smart people already coded for you

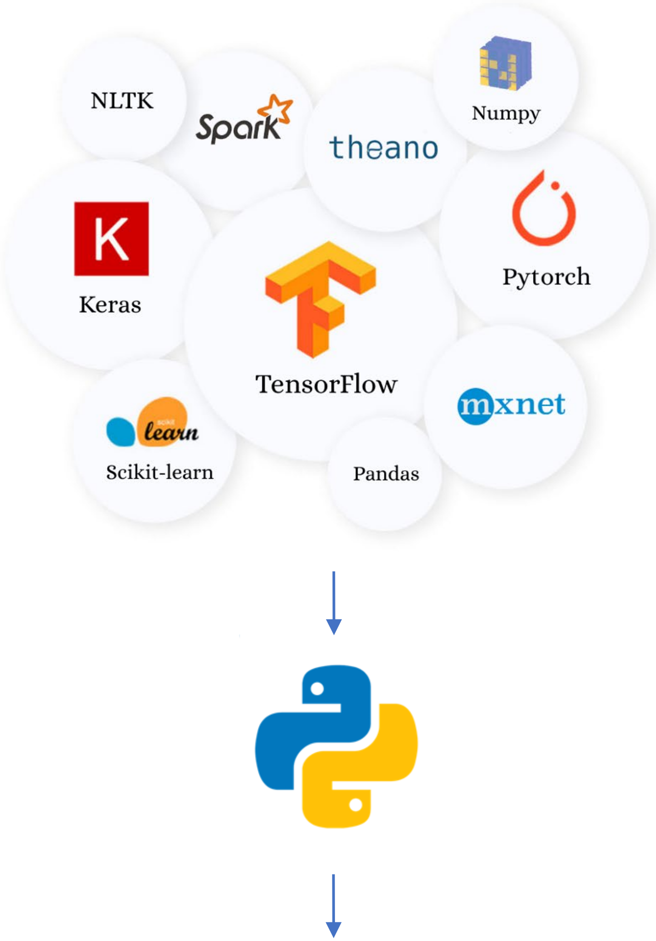
ML libraries



Real World Application

Computer scientists and smart people already coded for you

ML libraries



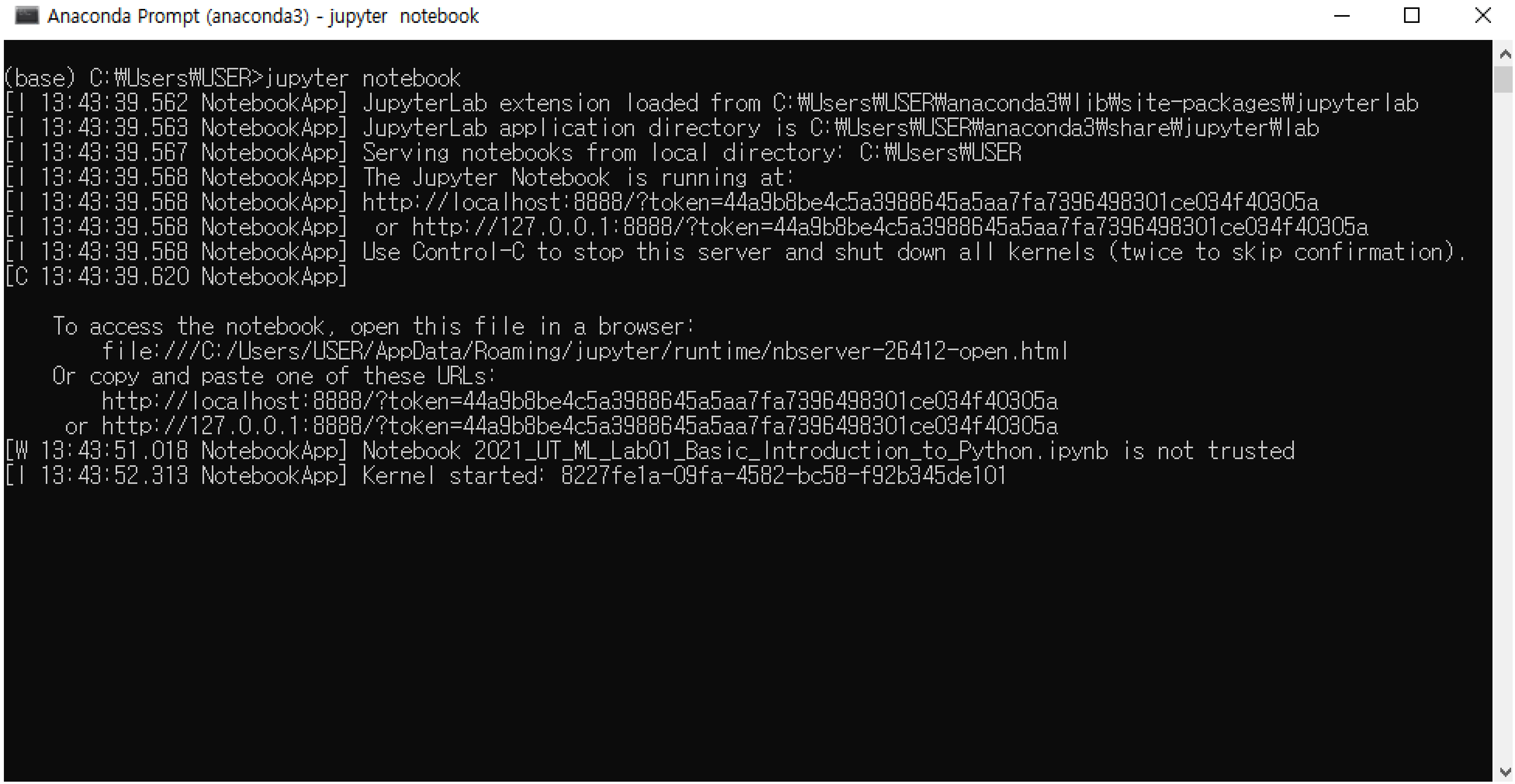
Real World Application

pip install python3

If you are in Frontier,
You will be working to
build up the codes

But, in application part,
You are duty is to
understand how libraries
apply and choose the
one that fits your RQ

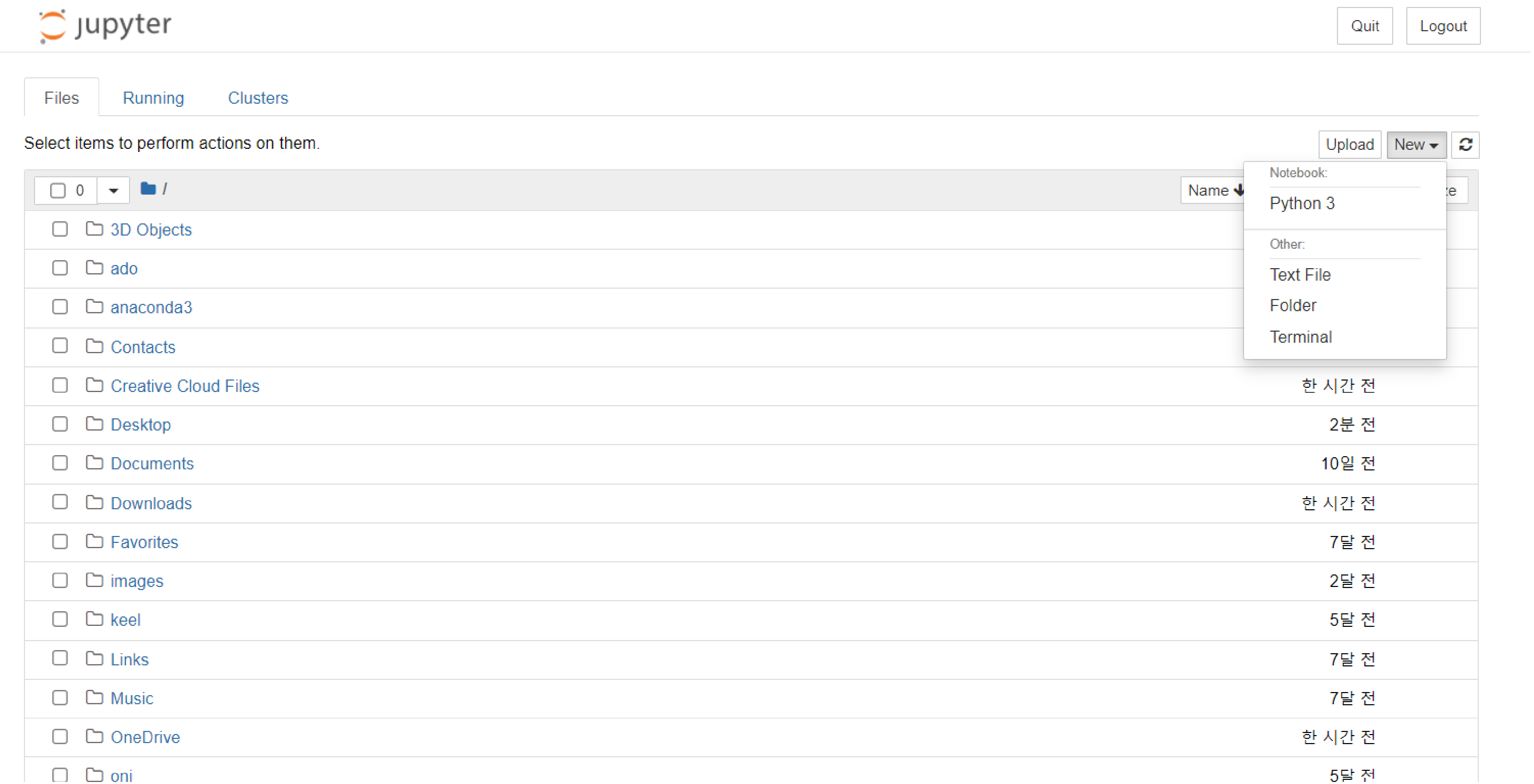
Type: jupyter notebook in Anaconda Prompt



```
(base) C:₩Users₩USER>jupyter notebook
[I 13:43:39.562 NotebookApp] JupyterLab extension loaded from C:₩Users₩USER₩anaconda3₩lib₩site-packages₩jupyterlab
[I 13:43:39.563 NotebookApp] JupyterLab application directory is C:₩Users₩USER₩anaconda3₩share₩jupyter₩lab
[I 13:43:39.567 NotebookApp] Serving notebooks from local directory: C:₩Users₩USER
[I 13:43:39.568 NotebookApp] The Jupyter Notebook is running at:
[I 13:43:39.568 NotebookApp] http://localhost:8888/?token=44a9b8be4c5a3988645a5aa7fa7396498301ce034f40305a
[I 13:43:39.568 NotebookApp]  or http://127.0.0.1:8888/?token=44a9b8be4c5a3988645a5aa7fa7396498301ce034f40305a
[I 13:43:39.568 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 13:43:39.620 NotebookApp]

    To access the notebook, open this file in a browser:
        file:///C:/Users/USER/AppData/Roaming/jupyter/runtime/nbserver-26412-open.html
    Or copy and paste one of these URLs:
        http://localhost:8888/?token=44a9b8be4c5a3988645a5aa7fa7396498301ce034f40305a
     or http://127.0.0.1:8888/?token=44a9b8be4c5a3988645a5aa7fa7396498301ce034f40305a
[W 13:43:51.018 NotebookApp] Notebook 2021_UT_ML_Lab01_Basic_Introduction_to_Python.ipynb is not trusted
[I 13:43:52.313 NotebookApp] Kernel started: 8227fe1a-09fa-4582-bc58-f92b345de101
```

Click New -> Python3



Select items to perform actions on them.

3D Objects	
ado	
anaconda3	
Contacts	
Creative Cloud Files	한 시간 전
Desktop	2분 전
Documents	10일 전
Downloads	한 시간 전
Favorites	7달 전
images	2달 전
keel	5달 전
Links	7달 전
Music	7달 전
OneDrive	한 시간 전
oni	5달 전