MATH604 DATA SCIENCE CAPSTONE PROJECT

Szu-Chi Chung

Department of Applied Mathematics, National Sun Yat-sen University

Lectures

- Class hours: Tue. (9:10-12:00)
  - Classroom: 理 SC-4009-1

- Lecture: Szu-Chi Chung (鍾思齊)
  - Office: 理 SC 2002-4
  - Office hour: Mon. 16:10~18:10 and Wed. 16:10~18:10

- Facebook
  - https://www.facebook.com/groups/335202601842081

Textbook and requirement

- Textbook: *Hands-On Machine Learning with Scikit-Learn, Keras, and TensorFlow, 2nd Edition*
  - Authors: Aurélien Géron
  - https://github.com/ageron/handson-ml2
- Reference book: *An Introduction to Statistical Learning with Applications in R*
  - Authors: James, Witten, Hastie, and Tibshirani
  - https://www.statlearning.com/
- For the data processing reference book: *Python for Data Analysis Data Wrangling with Pandas, NumPy, and IPython*
  - Authors: Wes McKinney
  - https://github.com/wesm/pydata-book

Textbook and requirement

- You should have basic knowledge about statistics and modeling
  - Please refer to our course website for resources and MOOC to review the basic concepts if needed
    - https://phonchi.github.io/nsysu-math604/

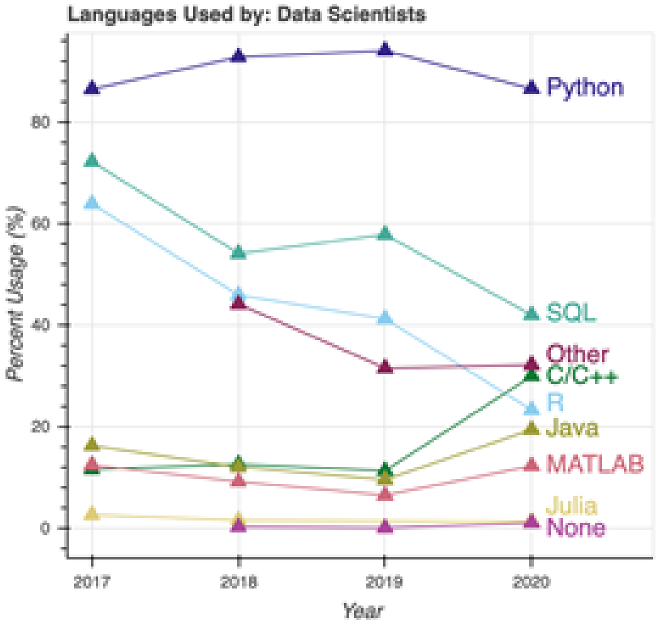
- Programming language: Python
  - You are asked to use python to implement the assignment, midterm and final
  - Since it is the most popular language in the field of data science
  - It is free and easy to learn
  - The homework and related material will be available in the course website

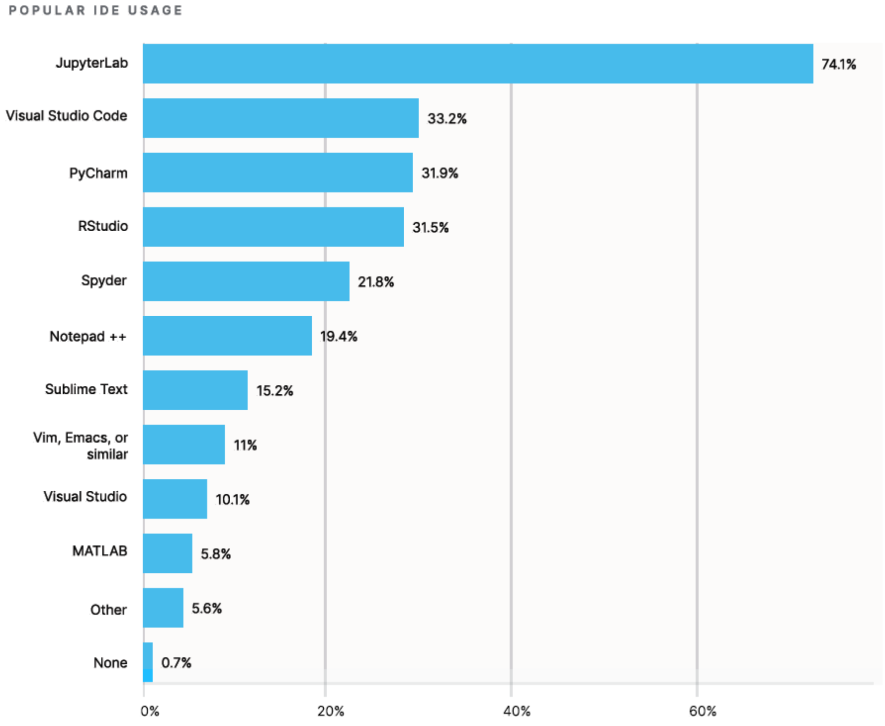
Grading policy

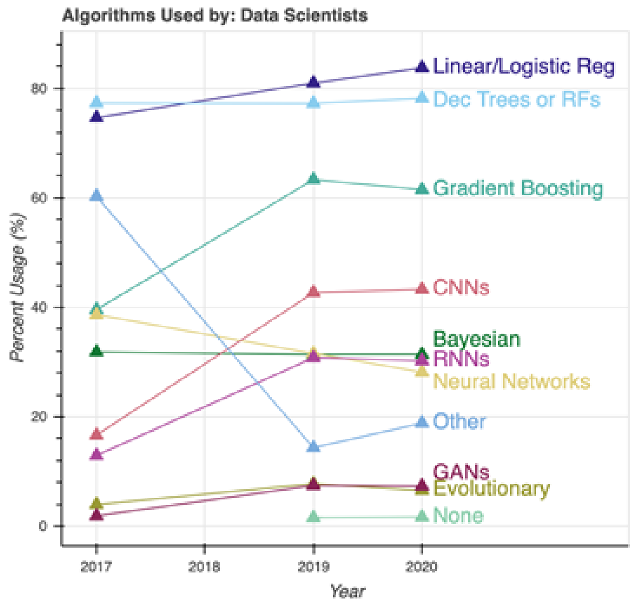
- Grading
  - Homework 20% (Both conceptual and coding part, about 4 times)
  - Midterm project 40% (We will provide a dataset and should use deep learning)
  - Final project 40% (You are free to choose any dataset and any analysis method)
- Midterm project:
  - Organize a team of 2 persons
  - Must hand in a report
- Final project:
  - Organize a team of 2 persons
  - Presentation will be held on 6/7 and 6/14
  - Must hand in a report
  - Score will be the summation of students (20%) and lecturer (20%).

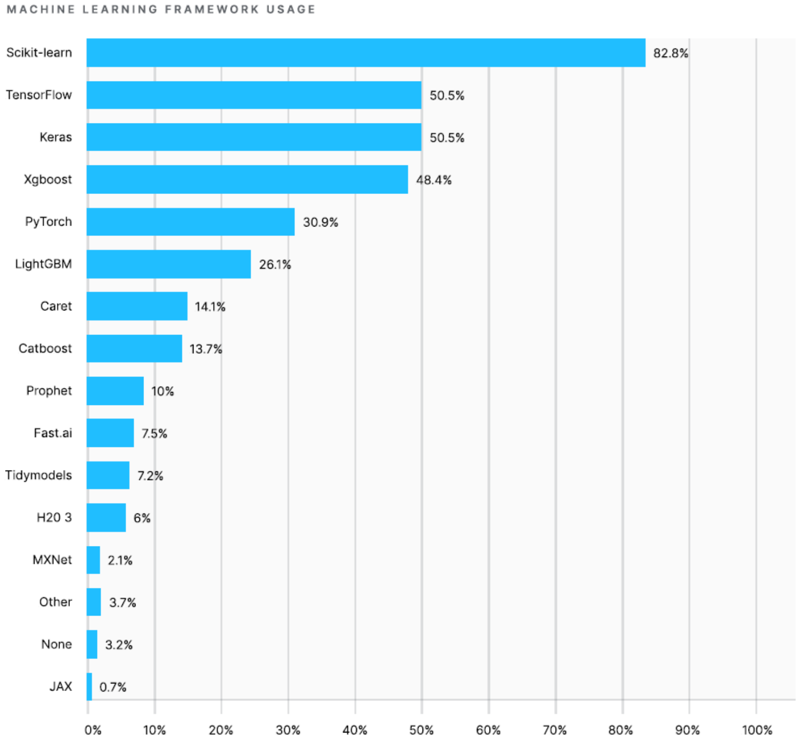
Dataset and competition

- Dataset
  - Dataset search platform provided by National Development Council
    - https://data.gov.tw/
  - Kaggle
    - https://www.kaggle.com/datasets
  - Google dataset search
    - https://datasetsearch.research.google.com/
- Competition
  - Kaggle
    - https://www.kaggle.com/competitions
  - Tbrain
    - https://tbrain.trendmicro.com.tw/





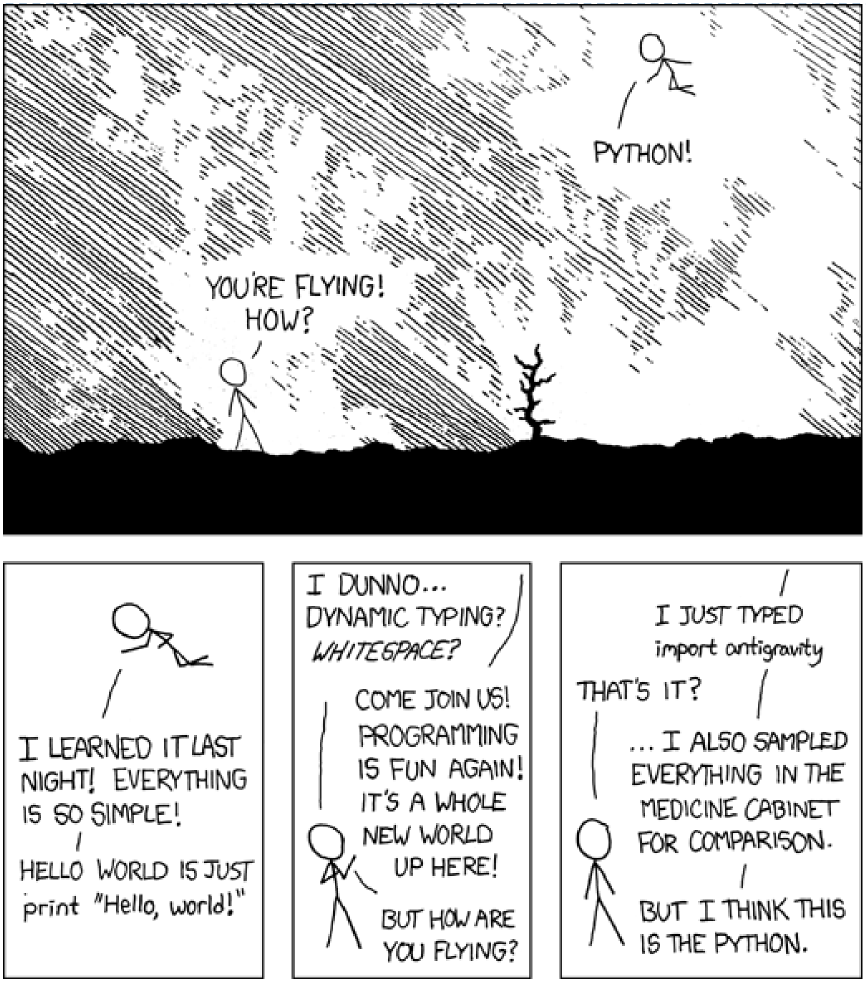




- https://www.kaggle.com/kaggle-survey-2020

Learning Python

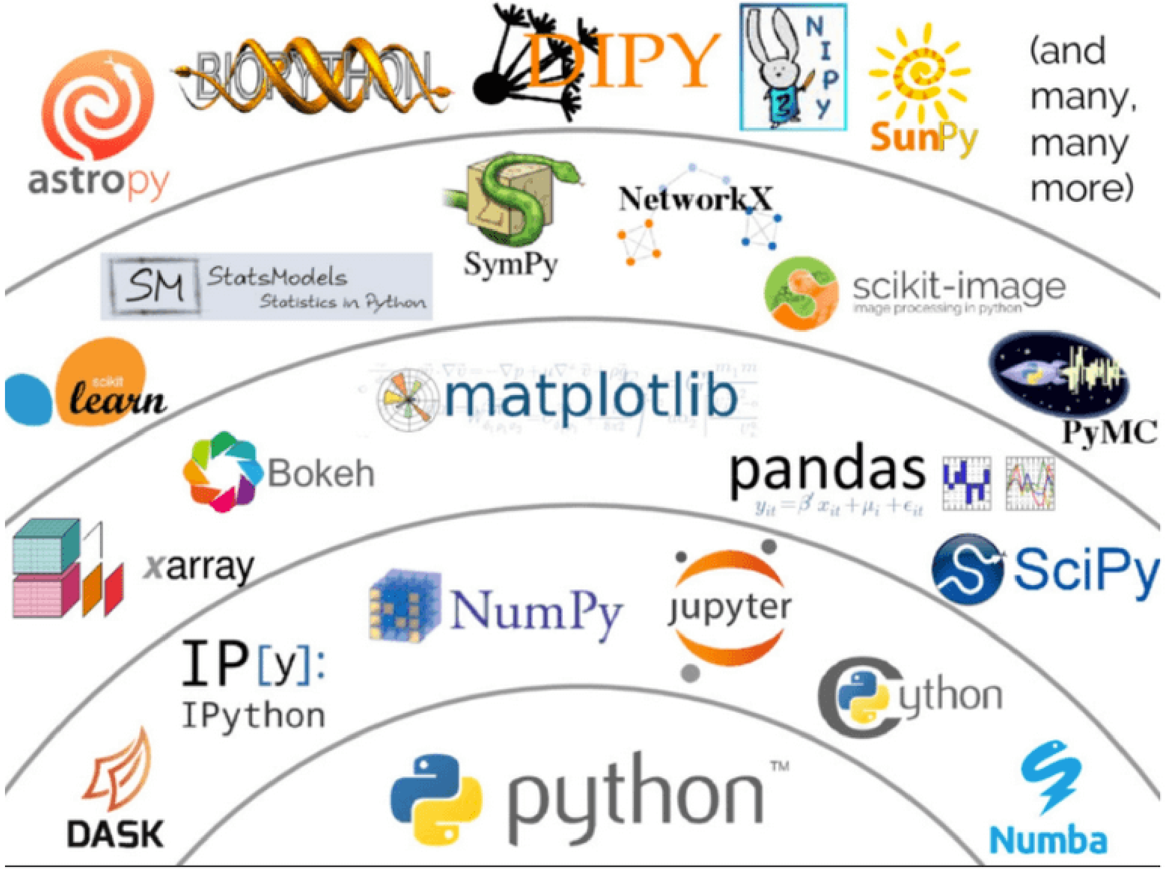
- Python
  - **Learn X in Y minutes**
  - Kaggle Python tutorial
  - Python for Everybody
  - Python 台灣社群
- Python scientific computing
  - https://scipy-lectures.org/
  - More
- Python for R and Matlab users
  - http://mathesaurus.sourceforge.net/r-numpy.html
  - https://numpy.org/doc/stable/user/numpy-for-matlab-users.html


PYTHON!
YOU'RE FLYING! HOW?
I LEARNED IT LAST NIGHT! EVERYTHING IS SO SIMPLE!
HELLO WORLD IS JUST print "Hello, world!"
I DUNNO... DYNAMIC TYPING? WHITESPACE?
COME JOIN US! PROGRAMMING IS FUN AGAIN! IT'S A WHOLE NEW WORLD UP HERE!
BUT HOW ARE YOU FLYING?
I JUST TYPED import antigravity
THAT'S IT?
... I ALSO SAMPLED EVERYTHING IN THE MEDICINE CABINET FOR COMPARISON.
BUT I THINK THIS IS THE PYTHON.

https://xkcd.com/353/

The Pydata Stack

- In 2017, a keynote at PyCon presented a schematic of the scientific Python stack
  - Project Jupyter and IPython for interactive computing and IDEs
  - NumPy for numerical array computing
    - Numba for just-in-time compilation
    - Cython for ahead-of-time compilation
  - Pandas for dataframe (Labeled array)
  - Scikit-learn for modeling
  - Seaborn and Bokeh for visualization
- Install Anaconda
  - https://www.anaconda.com/products/individual



Source: https://coiled.io/pydata-dask/

Environment

- Jupyter notebook
  - Colab - https://colab.research.google.com/
  - Kaggle - https://www.kaggle.com/docs/notebooks
  - Jupyterlab - https://www.anaconda.com/products/individual
- Markdown (Use on https://hackmd.io/, github, jupyter notebook...)
  - Learning
    - https://commonmark.org/
    - https://learnxinyminutes.com/docs/markdown/
- Cloud service
  - Google computing platform

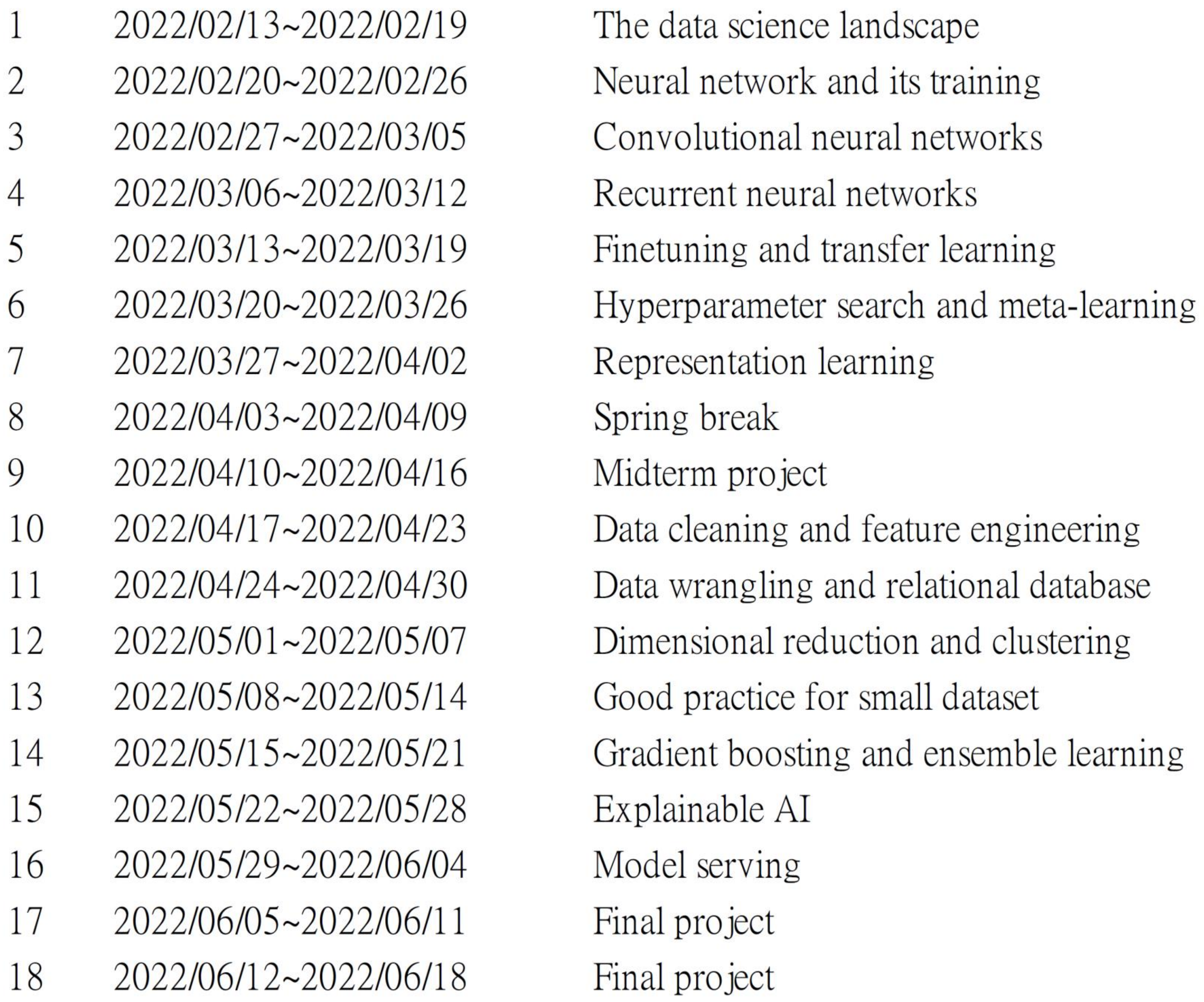
Our aim

- This Data Science Capstone aims to focus on the practical aspect of data science in the real world. In the capstone, students will learn to engage on a real-world project requiring them to apply skills from the entire data science pipeline: preparing, organizing, and transforming data, constructing a model, and evaluating results. Moreover, advanced modeling methods, including neural networks and gradient boosting, will also be covered

Related to other course

- More theory
  - Mathematical statistics
  - Statistical inference
- More about modeling
  - Machine learning
  - Statistical learning and data mining
  - Deep learning
- Apply to specific domain
  - Analysis of financial time series
  - Survival analysis
- Other important topics: High performance computing, Database systems…

Schedule



1	2022/02/13~2022/02/19	The data science landscape
2	2022/02/20~2022/02/26	Neural network and its training
3	2022/02/27~2022/03/05	Convolutional neural networks
4	2022/03/06~2022/03/12	Recurrent neural networks
5	2022/03/13~2022/03/19	Finetuning and transfer learning
6	2022/03/20~2022/03/26	Hyperparameter search and meta-learning
7	2022/03/27~2022/04/02	Representation learning
8	2022/04/03~2022/04/09	Spring break
9	2022/04/10~2022/04/16	Midterm project
10	2022/04/17~2022/04/23	Data cleaning and feature engineering
11	2022/04/24~2022/04/30	Data wrangling and relational database
12	2022/05/01~2022/05/07	Dimensional reduction and clustering
13	2022/05/08~2022/05/14	Good practice for small dataset
14	2022/05/15~2022/05/21	Gradient boosting and ensemble learning
15	2022/05/22~2022/05/28	Explainable AI
16	2022/05/29~2022/06/04	Model serving
17	2022/06/05~2022/06/11	Final project
18	2022/06/12~2022/06/18	Final project

End-to-End Machine Learning Project

Szu-Chi Chung

Department of Applied Mathematics, National Sun Yat-sen University

Data

- We live in a world that’s drowning in data
  - Websites track every user’s every click
  - Your smartphone is building up a record of your location every second of every day
  - People wear smart watch that are always recording their heart rates, movement habits, diet, and sleep patterns
  - Smart cars collect driving habits, smart homes collect living habits, and smart marketers collect purchasing habits
  - The internet itself represents a huge graph of knowledge that contains an enormous cross-referenced encyclopedia; domain-specific databases about movies, music, sports results…
- Buried in these data are answers to countless questions that no one’s ever thought to ask

Data

- Facebook asks you to list your hometown and your current location, ostensibly to make it easier for your friends to find and connect with you. But it also analyzes these locations to identify global migration patterns and where the fanbases of different sport teams live

- As a large retailer, Costco tracks your purchases and interactions, both online and in-store. And it may, for example, uses the data to predict which of its customers are pregnant, to better market baby-related purchases to them

- Some others also use data to make government more effective, to help the homeless, and to improve public health

What Is Data Science?

- Data science is an interdisciplinary field that uses scientific methods, algorithms and systems to extract knowledge and insights from noisy, structured and unstructured data, and apply knowledge and actionable insights from data across a broad range of application domains

- A data scientist often creates <u>programming code</u>, and combines it with <u>statistical knowledge</u> to create insights from data

What Is Data Science?

- The field encompasses preparing data for analysis, formulating data science problems, analyzing data, developing data-driven solutions, and presenting findings to inform high-level decisions in a broad range of application domains

- As such, it incorporates skills from computer science, statistics, information science, mathematics, information visualization, graphic design, complex systems, communication and business

- In 2015, the American Statistical Association identified database management, statistics and machine learning, and distributed and parallel systems as the three emerging foundational professional communities of data science

Who Are Data Scientists?

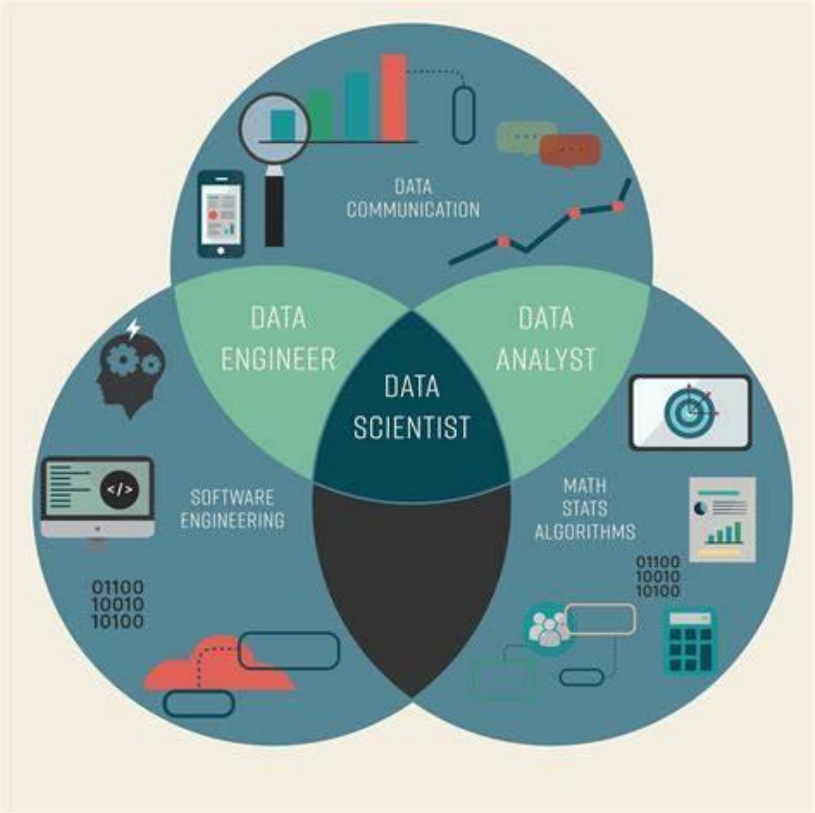
- There’s a joke that says a data scientist is someone who knows more statistics than a computer scientist or more computer science than a statistician

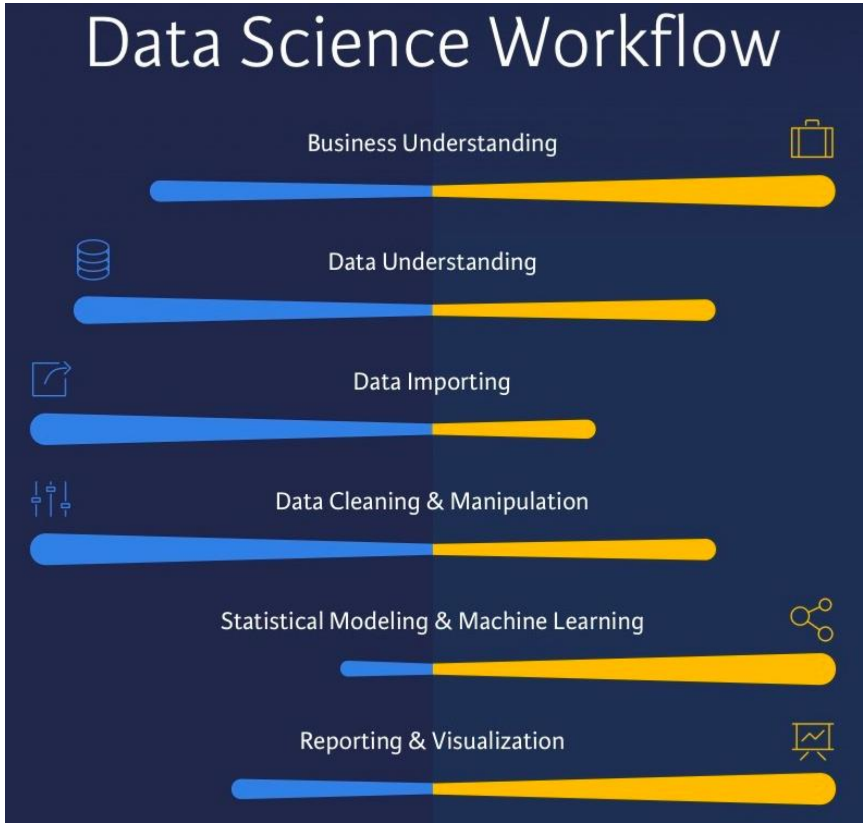
- In fact, some data scientists are statisticians, some are software engineers, some are …
  - In short, pretty much no matter how you define data science, you’ll find practitioners for whom the definition is totally, absolutely wrong

- We’ll say that a data scientist is someone who extracts insights from messy data. Today’s world is full of people trying to turn data into insight!

Who Are Data Scientists?

- Data scientist

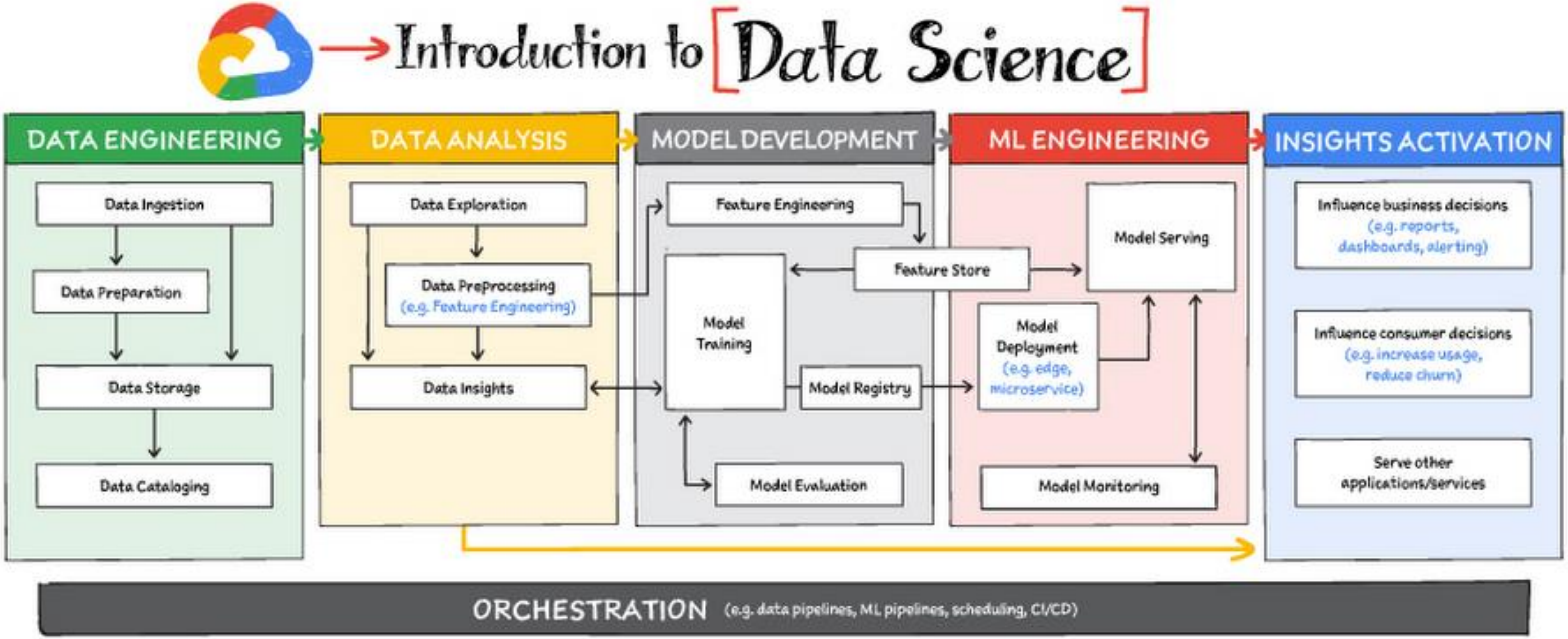




https://www.datacamp.com/community/blog/data-scientist-vs-data-engineer
https://k21academy.com/microsoft-azure/data-science-vs-data-analytics-vs-data-engineer/

- Data scientists roadmap
  - https://github.com/AMAI-GmbH/AI-Expert-Roadmap

The Pipeline



Introduction to [Data Science]
DATA ENGINEERING
Data Ingestion
Data Preparation
Data Storage
Data Cataloging
DATA ANALYSIS
Data Exploration
Data Preprocessing
(e.g. Feature Engineering)
Data Insights
MODEL DEVELOPMENT
Feature Engineering
Feature Store
Model Training
Model Registry
Model Evaluation
ML ENGINEERING
Model Serving
Model Deployment
(e.g. edge, microservice)
Model Monitoring
INSIGHTS ACTIVATION
Influence business decisions
(e.g. reports, dashboards, alerting)
Influence consumer decisions
(e.g. increase usage, reduce churn)
Serve other applications/services
ORCHESTRATION (e.g. data pipelines, ML pipelines, scheduling, CI/CD)

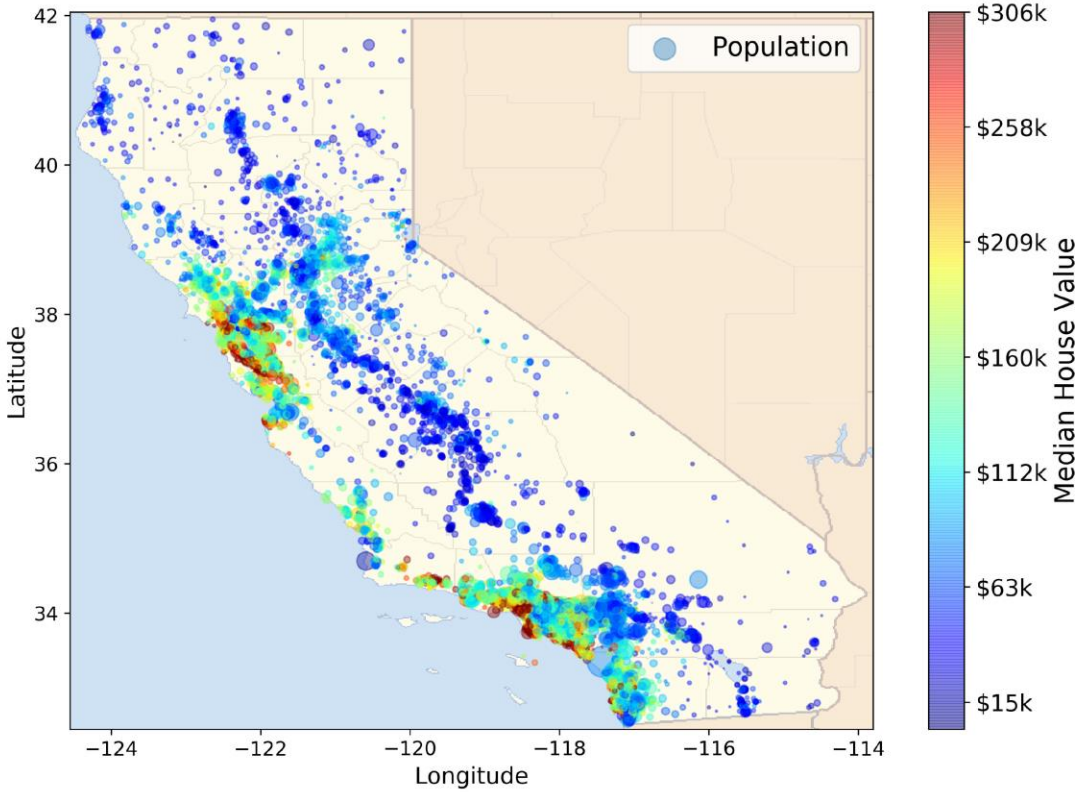
https://cloud.google.com/blog/topics/developers-practitioners/intro-data-science-google-cloud

The Checklist

1. Look at the big picture
2. Get the data
3. Discover and visualize the data to gain insights
4. Prepare the data for Machine Learning algorithms
5. Select a model and train it
6. Fine-tune your model
7. Present your solution
8. Launch, monitor, and maintain your system

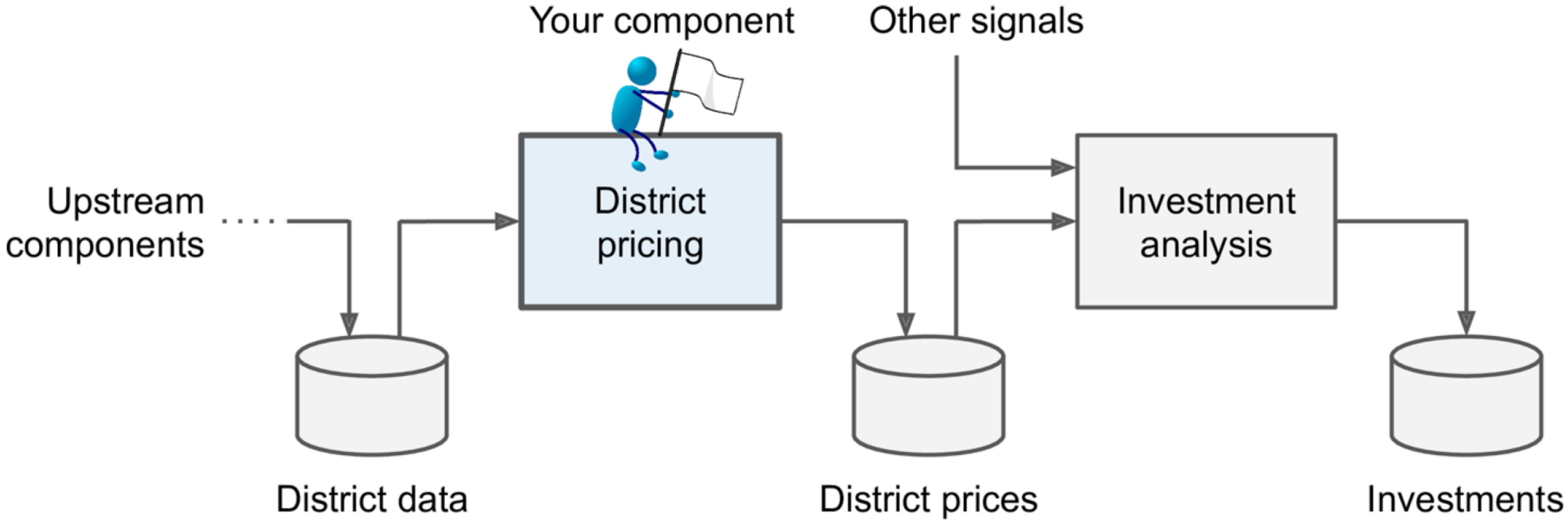
1. Look at the Big Picture

- Your first task is to use California census data to build a model of housing prices in the state
  - This data includes metrics such as the population, median income, and median housing price for each districts in California
  - Your model should learn from this data and be able to predict the median housing price in any district, given all the other features



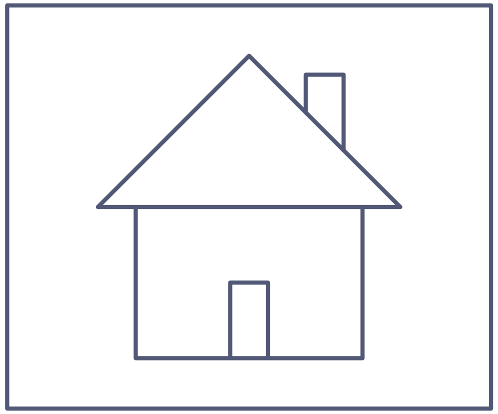
1. Look at the Big Picture – Business objective

- How does the company expect to use and benefit from this model?
  - Your model's output (a prediction of a district's median housing price) will be fed to another ML system, along with many other signals. This downstream system will determine whether it is worth investing in a given area or not



1. Look at the Big Picture

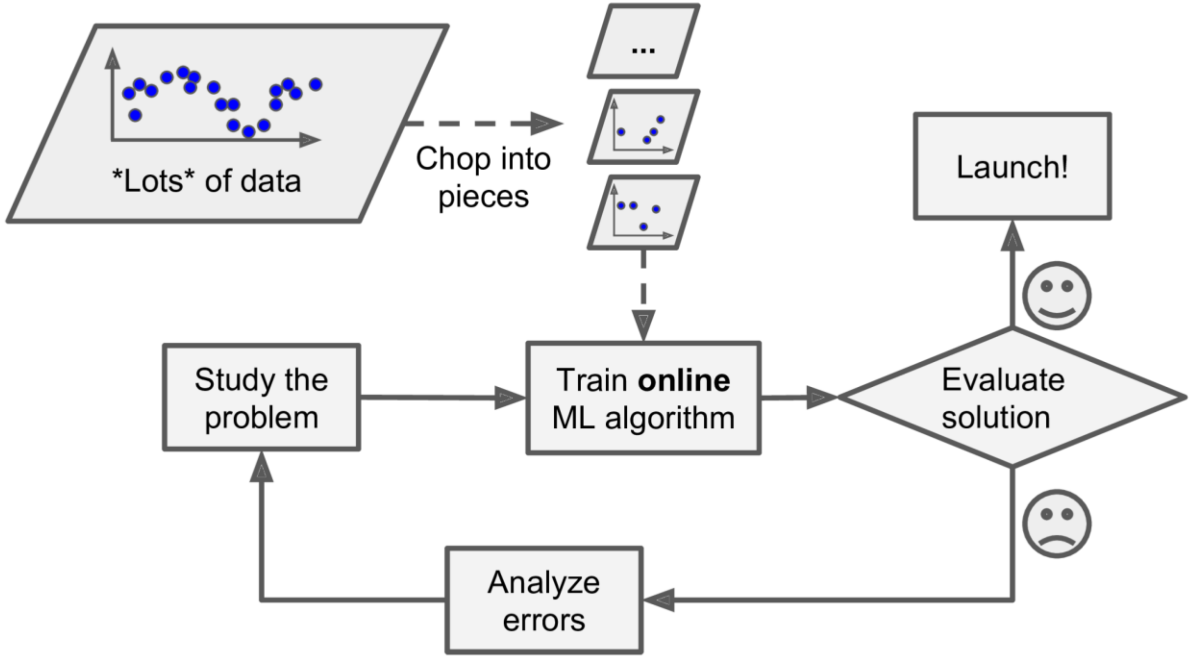
- What the current solution looks like (if any)?
  - The district housing prices are currently estimated manually by experts: a team gathers up-to-date information about a district, and when they cannot get the median housing price, they estimate it using complex rules
  - This is costly and time-consuming, and their estimates are not great which often off by more than 20%. This is why the company thinks that it would be useful to train a model to predict a district's median housing price, given other data about that district



1. Look at the Big Picture - Frame the Problem

- What kind of task?

1. It is a typical *supervised learning* task, since you are given labeled training examples
2. It is also a *multiple regression* task and a *univariate regression* problem, since we are using multiple features and trying to predict single value for each district
3. There is no particular need to adjust to changing data rapidly, and the data is small enough to fit in memory, so plain *batch learning* should do just fine



1. Look at the Big Picture - Select a Performance Measure

- A typical performance measure for regression problems is the Root Mean Square Error (RMSE) or l_2 norm

$$RMSE(X,h) = \sqrt{\frac{1}{m}\sum_{i=1}^{m}(h(x^{(i)}) - y^{(i)})^2}$$

  - Suppose that there are many outlier districts. In that case, you may consider using the mean absolute error

$$MAE(X,h) = \frac{1}{m}\sum_{i=1}^{m}\left|h\left(x^{(i)}\right) - y^{(i)}\right|$$

  - The higher the norm index, the more it focuses on large values and neglects small ones. This is why the RMSE is more sensitive to outliers than the MAE
- What would the minimum performance needed?

1. Look at the Big Picture - Check the Assumptions

- Lastly, it is good practice to list and verify the assumptions that have been made so far (by you or others)
  - This can help you catch serious issues early on. For example, the district prices that your system outputs are going to be fed into a downstream ML system, and you assume that these prices are going to be used as such
  - But if the downstream system converts the prices into categories (e.g., “cheap,” “medium,” or “expensive”) and then uses those categories instead of the prices themselves? If that’s so, then the problem should have been framed as a classification task, not a regression task!

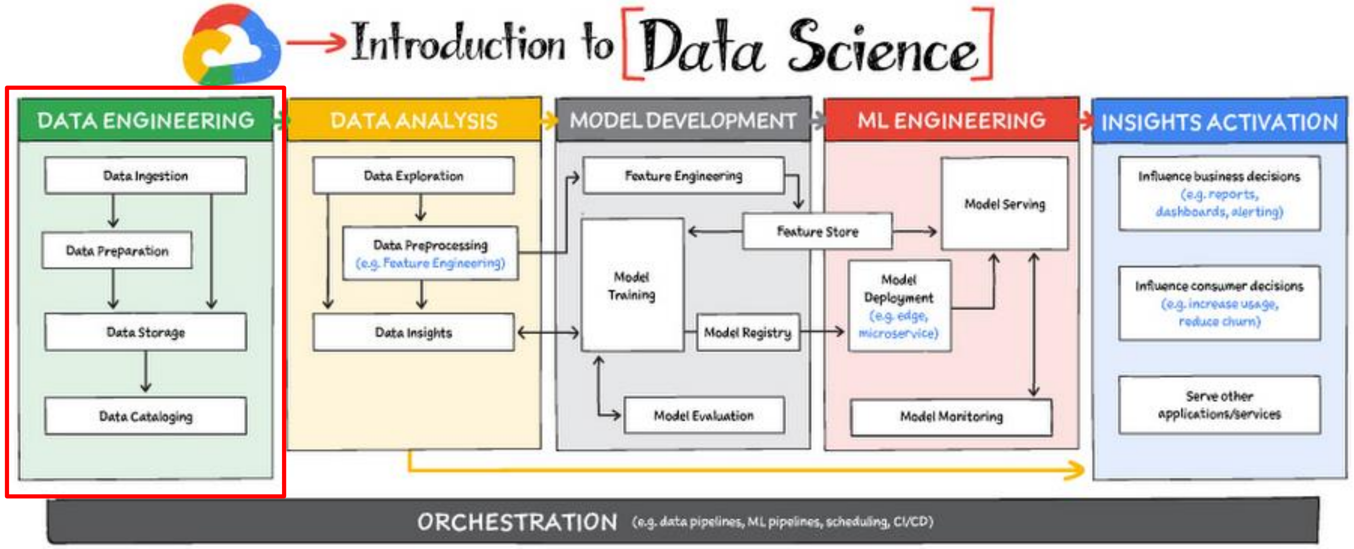
2. Get the Data

- Find and get the data
  - List the data you need and how much you need
  - Find and document where you can get the data
  - Check how much space it will take

- In typical environments your data would be available in a relational database (or some other common data store) and spread across multiple tables/documents/files
  - To access it, you would first need to get your credentials and access authorizations and familiarize yourself with the data schema
  - You should check legal constraint here

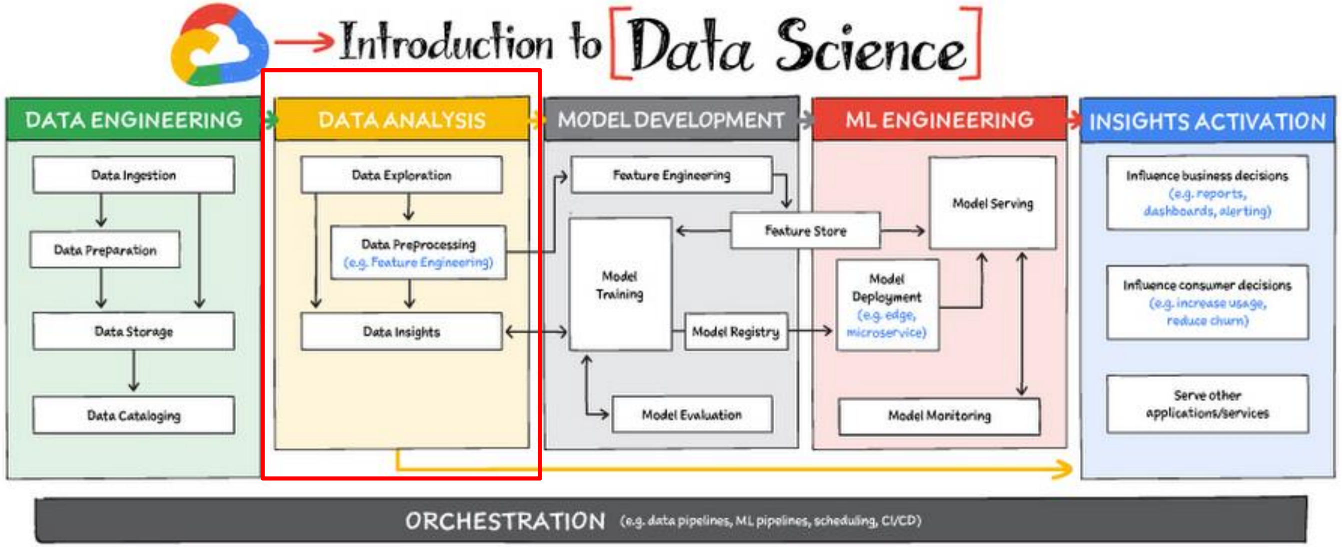
2. Get the Data - Create the workspace and download the data

- Work in an isolated environment may be prefer
  - Anaconda
- Get the data and convert the data to a format that you can manipulate
- Check the size and type of the data
- Create a test set and put it aside


Introduction to Data Science
DATA ENGINEERING
Data Ingestion
Data Preparation
Data Storage
Data Cataloging
DATA ANALYSIS
Data Exploration
Data Preprocessing (e.g. Feature Engineering)
Data Insights
MODEL DEVELOPMENT
Feature Engineering
Feature Store
Model Training
Model Registry
Model Evaluation
ML ENGINEERING
Model Serving
Model Deployment (e.g. edge, microservice)
Model Monitoring
INSIGHTS ACTIVATION
Influence business decisions (e.g. reports, dashboards, alerting)
Influence consumer decisions (e.g. increase usage, reduce churn)
Serve other applications/services
ORCHESTRATION (e.g. data pipelines, ML pipelines, scheduling, CI/CD)

3. Explore the data

- Create a copy of the data for exploration (sampling it down to a manageable size if needed)
- Visualizing the data
  - Study each attributes and its characteristics
  - Looking for correlations
  - Identify promising transformation



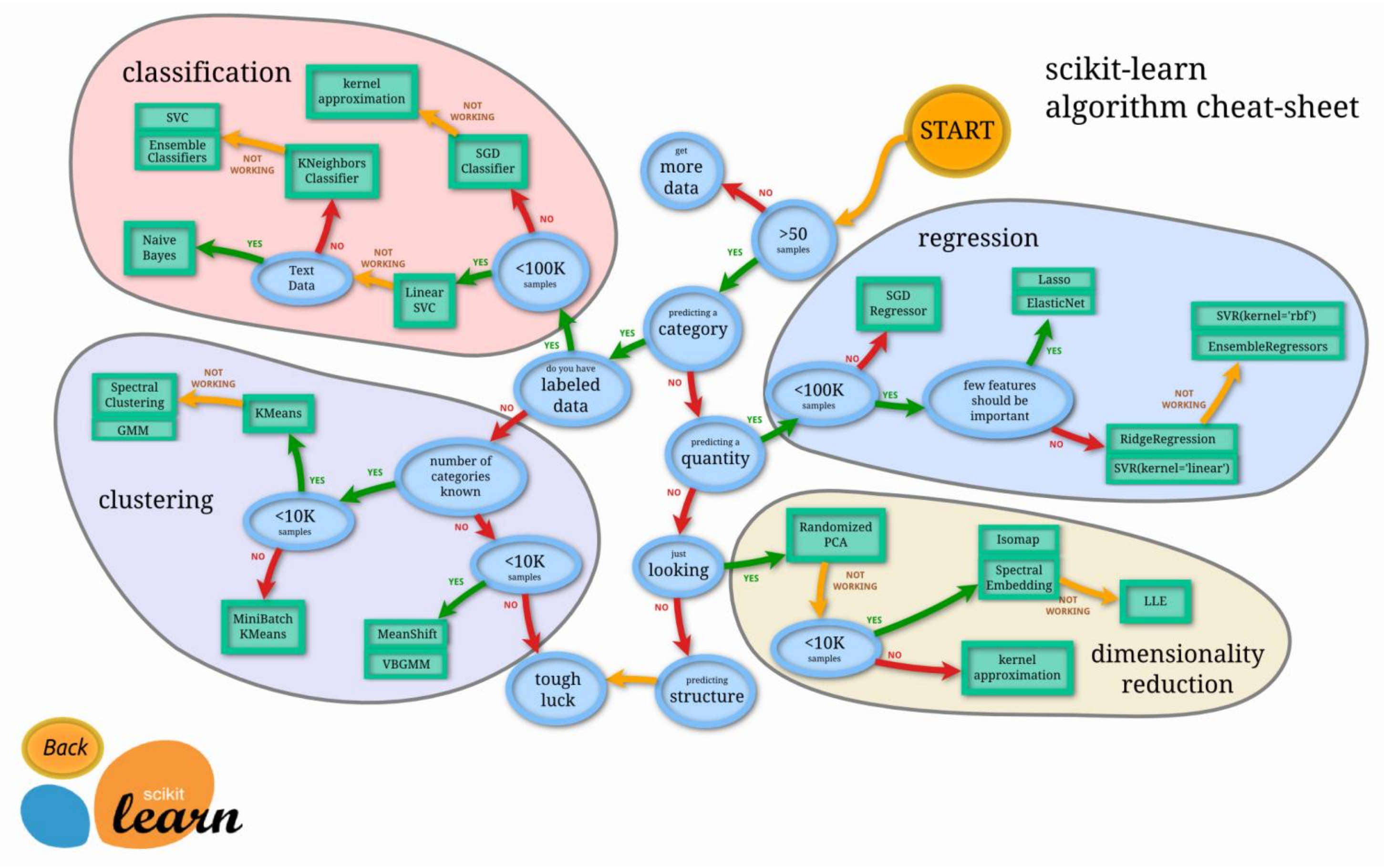
Introduction to [Data Science]
DATA ENGINEERING
Data Ingestion
Data Preparation
Data Storage
Data Cataloging
DATA ANALYSIS
Data Exploration
Data Preprocessing
(e.g. Feature Engineering)
Data Insights
MODEL DEVELOPMENT
Feature Engineering
Feature Store
Model Training
Model Registry
Model Evaluation
ML ENGINEERING
Model Serving
Model Deployment
(e.g. edge, microservice)
Model Monitoring
INSIGHTS ACTIVATION
Influence business decisions
(e.g. reports, dashboards, alerting)
Influence consumer decisions
(e.g. increase usage, reduce churn)
Serve other applications/services
ORCHESTRATION (e.g. data pipelines, ML pipelines, scheduling, CI/CD)

4. Prepare the data

- Data cleaning
  - Fixing outlier
  - Deal with missing data
- Feature selection
- Feature engineering
  - Handling text and categorical attributes
  - Decomposing features (date/time)
  - Aggregate features into promising ones
  - Add promising transforms of features
  - Discretize continuous feature
  - Feature scaling

5. Select a model and train it

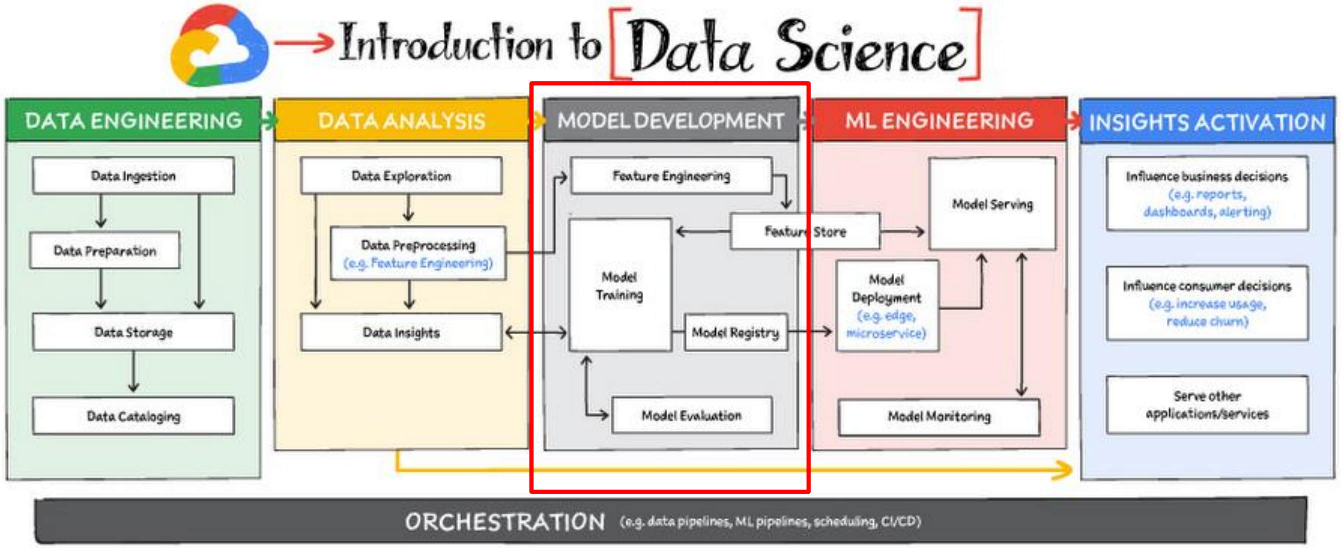
- Selecting models
  - Training and evaluating on the training set
  - Try different models
  - Analyze the most significant variables for each of them
  - Analyze the types of errors
  - Shortlist promising models, preferring models that make different types of errors



https://scikit-learn.org/stable/tutorial/machine_learning_map/index.html

6. Fine-Tune Your Model

- Fine-tune the hyperparameter using CV
  - Grid search
  - Random search
  - You can treat your data transformation choices as hyperparameters
  - You may want to use Bayesian optimization
- Try ensemble methods
- Measure performance on test set

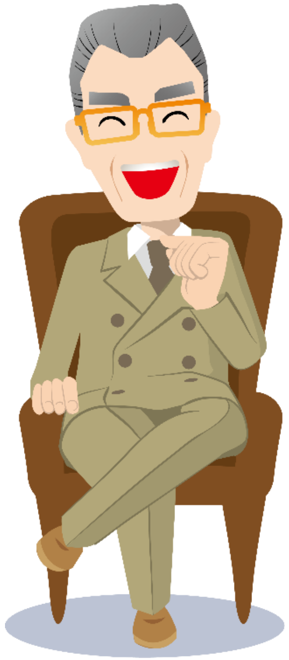


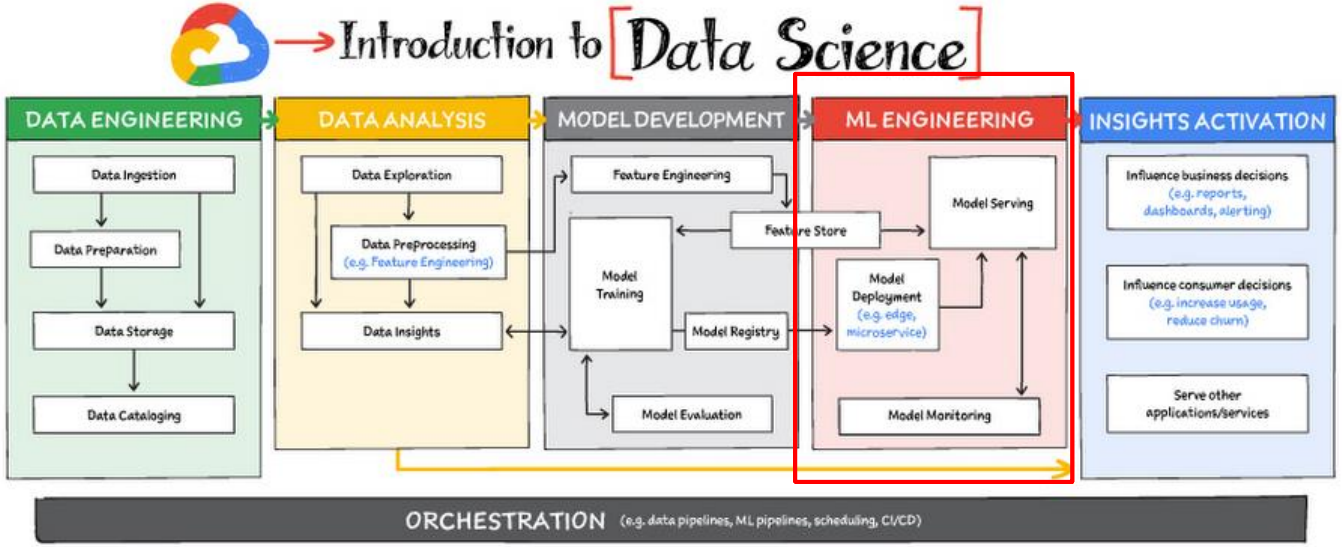
7. Present your solution

1. Document what you have done
2. Create a nice presentation
    1. Make sure you highlight the big picture first
3. Explain why your solution achieves the business objective
4. Don’t forget to present interesting points you noticed along the way
    1. Describe what worked and what did not
    2. List your assumptions and your system’s limitations
5. Ensure your key findings are communicated through beautiful visualizations or easy-to-remember statements (e.g., “the median income is the number-one predictor of housing prices”)

8. Launch, Monitor, and Maintain Your System

- Write monitoring code to check your system's live performance
  - Monitor input's quality
  - Retrain your model on a regular basis




Introduction to Data Science
DATA ENGINEERING
Data Ingestion
Data Preparation
Data Storage
Data Cataloging
DATA ANALYSIS
Data Exploration
Data Preprocessing
(e.g. Feature Engineering)
Data Insights
MODEL DEVELOPMENT
Feature Engineering
Feature Store
Model Training
Model Registry
Model Evaluation
ML ENGINEERING
Model Serving
Model Deployment
(e.g. edge, microservice)
Model Monitoring
INSIGHTS ACTIVATION
Influence business decisions
(e.g. reports, dashboards, alerting)
Influence consumer decisions
(e.g. increase usage, reduce churn)
Serve other applications/services
ORCHESTRATION (e.g. data pipelines, ML pipelines, scheduling, CI/CD)

Appendix

Working with Real Data

- Popular open data repositories
  - UC Irvine Machine Learning Repository
  - Kaggle datasets
  - Amazon’s AWS datasets
- Meta portals (they list open data repositories)
  - OpenDataMonitor
  - Awesome datasets
- Other pages listing many popular open data repositories
  - Wikipedia’s list of Machine Learning datasets
  - Quora.com
  - The datasets subreddit
  - https://www.analyticsvidhya.com/blog/2022/01/10-best-data-science-websites-to-find-datasets-for-your-next-ds-project/

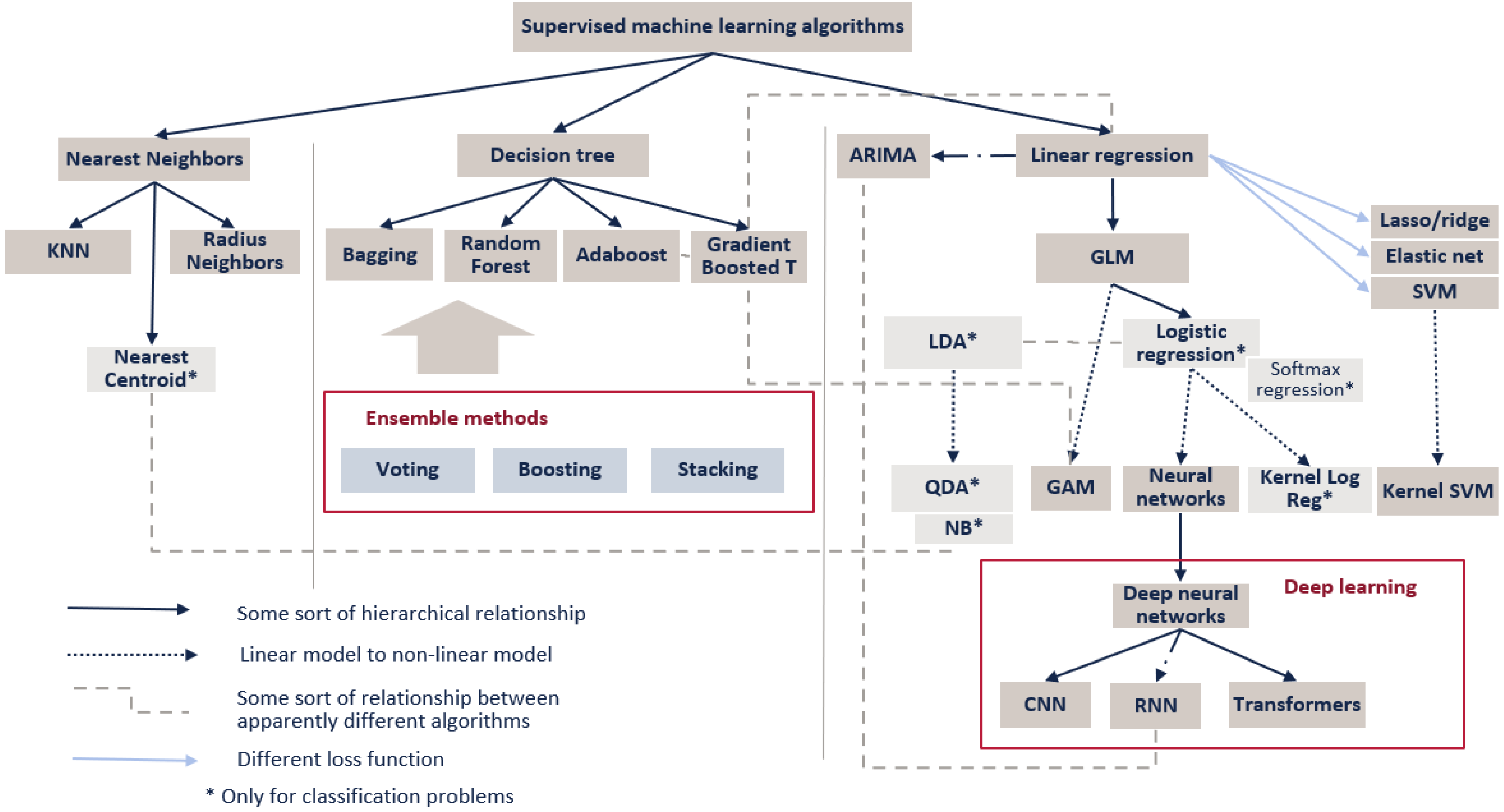
Resources

- Data science and modeling
  - https://github.com/GokuMohandas/MadeWithML
  - https://virgili0.github.io/Virgilio/#table-of-contents
  - MOOC Lectures
- Deep learning
  - https://github.com/fastai/fastbook
  - https://d2l.ai/

Resources

- Libraries
  - https://github.com/EthicalML/awesome-production-machine-learning
  - https://github.com/academic/awesome-datascience
  - https://storage.googleapis.com/deepmind-media/research/New_AtHomeWithAI%20resources.pdf
  - https://github.com/ml-tooling/best-of-ml-python
- Cheat sheet
  - https://github.com/aaronwangy/Data-Science-Cheatsheet
  - https://github.com/afshinea/stanford-cs-229-machine-learning
  - https://github.com/afshinea/stanford-cs-230-deep-learning

Supervised learning



Supervised machine learning algorithms

Nearest Neighbors

KNN

Radius Neighbors

Nearest Centroid*

Decision tree

Bagging

Random Forest

Adaboost

Gradient Boosted T

Ensemble methods

Voting

Boosting

Stacking

ARIMA

Linear regression

Lasso/ridge

Elastic net

SVM

GLM

LDA*

Logistic regression*

Softmax regression*

QDA*

NB*

GAM

Neural networks

Kernel Log Reg*

Kernel SVM

Deep learning

Deep neural networks

CNN

RNN

Transformers

→ Some sort of hierarchical relationship

⋯→ Linear model to non-linear model

- - Some sort of relationship between apparently different algorithms

→ Different loss function

* Only for classification problems

https://towardsdatascience.com/overview-of-supervised-machine-learning-algorithms-a5107d036296