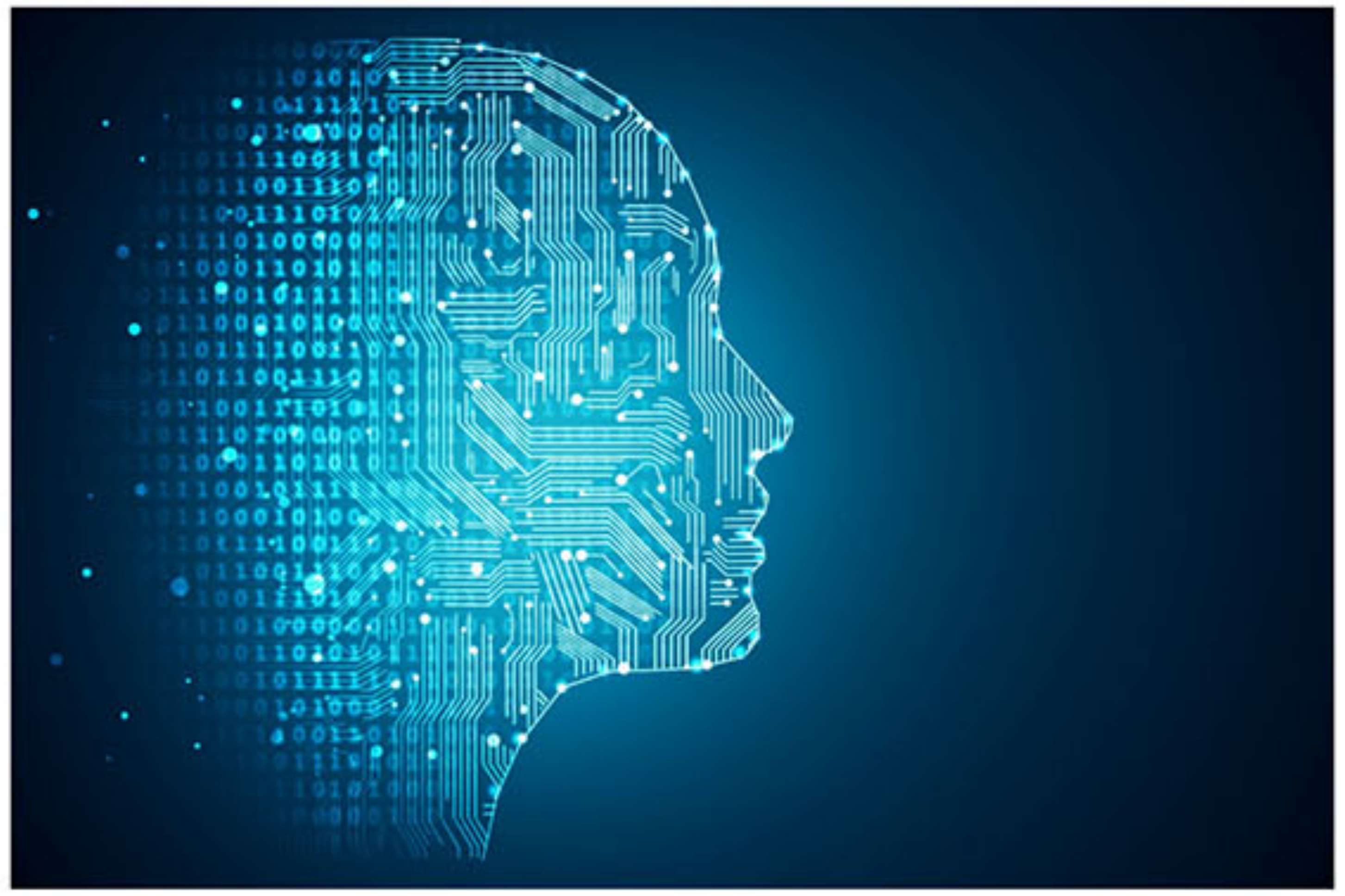
PRINCETON UNIVERSITY

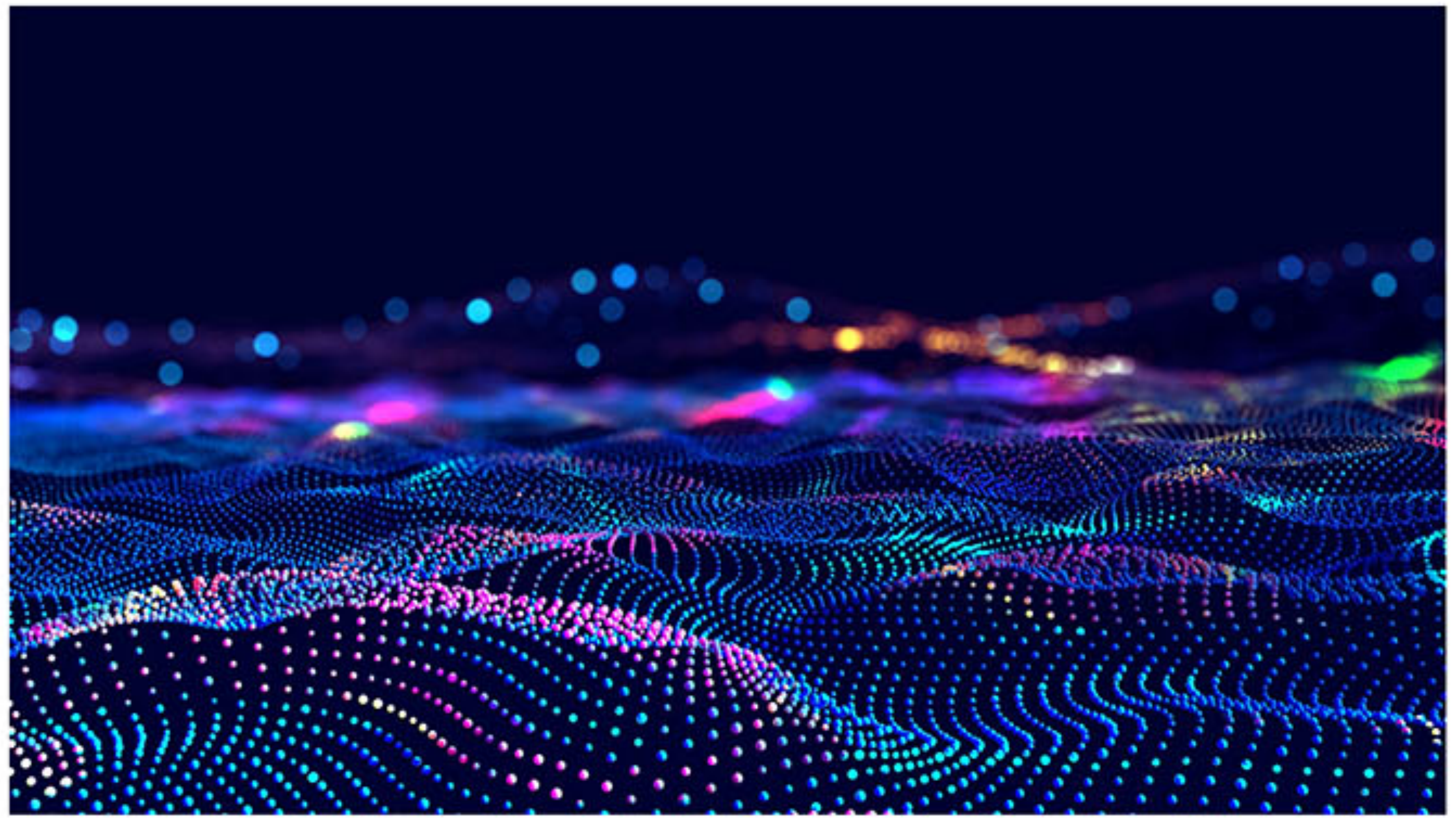
CENTER FOR STATISTICS AND MACHINE LEARNING

A Hands–On Introduction to Machine Learning



Wintersession 2025
January 15–17,21

Julian Gold
Gage DeZoort



With materials from:

Brian Arnold, Gage DeZoort, Julian Gold, Jonathan Halverson, Christina Peters, Savannah Thias, Amy Winecoff

DataX
Accelerating Scientific Discovery at Princeton

Wintersession 2025 with PICSciE/RC

20 hours of machine learning training

Instructors

Sarah-Jane Leslie, Professor of Philosophy and CSML, and NAM Co-Director
Julian Gold, DataX Data Scientist, CSML
Gage DeZoort, Postdoctoral Research Associate and Lecturer, Physics
Simon Park, Graduate Student, Computer Science and PLI
Abhishek Panigrahi, Graduate Student, Computer Science and PLI
Christian Jespersen, Graduate Student, Astrophysical Sciences
Rafael Pastrana, Graduate Student, Architecture
Quinn Gallagher, Graduate Student, Chemical and Biological Engineering
Holly Johnson, Graduate Student, Electrical and Computer Engineering

Introduction to Machine Learning for Humanists and Social Scientists → **A Hands-On Introduction to Machine Learning** → **Getting Started with LLMs with Princeton Language and Intelligence** / **Graph Neural Networks for Your Research** / **Machine Learning for the Physical Sciences**

Introduction to Machine Learning for Humanists and Social Scientists	Part 1	Mon Jan. 13	10 AM-12 PM
	Part 2	Tue Jan. 14	10 AM-12 PM
A Hands-On Introduction to Machine Learning	Part 1	Wed Jan. 15	2-4 PM
	Part 2	Thu Jan. 16	2-4 PM
	Part 3	Fri Jan. 17	2-4 PM
	Part 4	Tue Jan. 21	2-4 PM
Getting Started with LLMs with Princeton Language and Intelligence	Part 1	Wed Jan. 22	2-4 PM
	Part 2	Thu Jan. 23	2-4 PM

NATURAL & ARTIFICIAL MINDS

CENTER FOR STATISTICS AND MACHINE LEARNING

DataX
Accelerating Scientific Discovery at Princeton

PLi PRINCETON Language + Intelligence

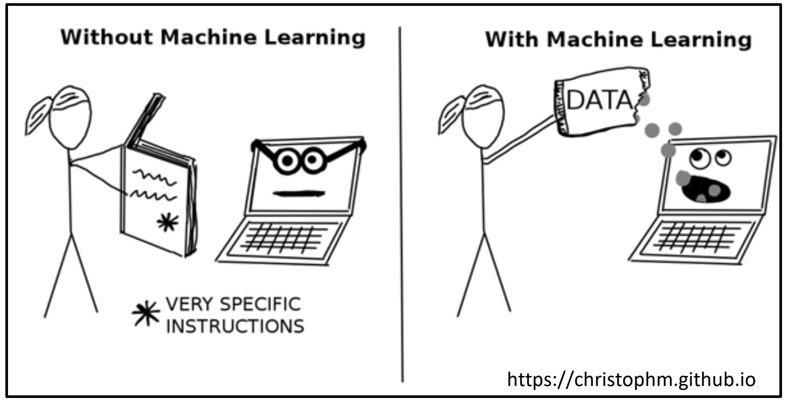
https://researchcomputing.princeton.edu/workshops

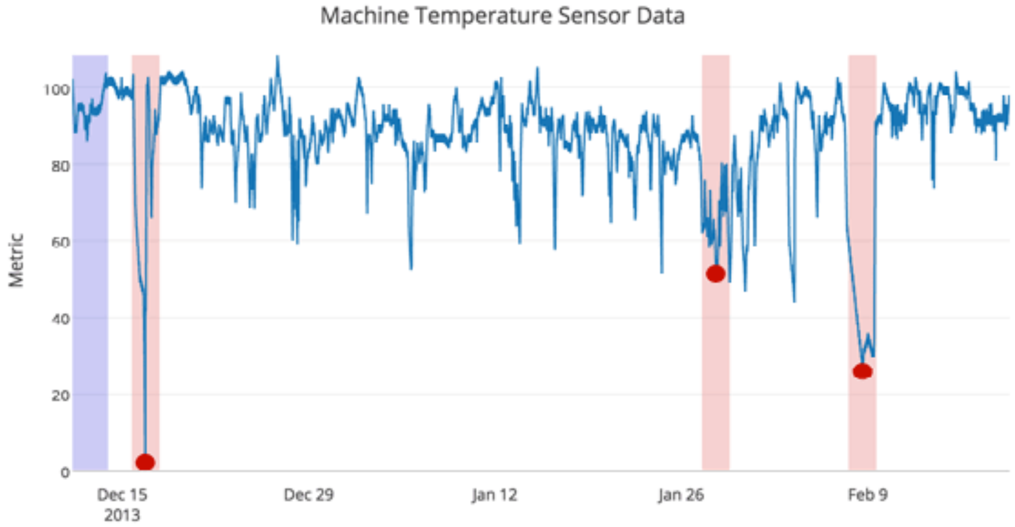
Mini-Course Outline

Date	Topic	Instructor
Wed. 1/15	Machine Learning Overview and Simple Models	Julian Gold
Thu. 1/16	Model Evaluation and Improving Performance	Julian Gold
Fri. 1/17	Introduction to Neural Networks	Gage DeZoort
Tue. 1/21	Survey of Neural Network Architectures	Gage DeZoort
Wed.+Thu. 1/22-1/23	Getting Started with LLMs with PLI	Simon Park, Abhishek Panigrahi
Wed. 1/22	Graph Neural Networks for Your Research	Gage DeZoort
Wed. 1/22	Machine Learning for the Physical Sciences	C. Jespersen, R. Pastrana, Q. Gallagher, H. Johnson

What is machine learning?

1. building and understanding methods that ‘**learn**’ by using **data** to improve performance on some set of tasks
2. using and developing computer systems that can **learn** and adapt without following explicit instructions, by using algorithms and statistical models to analyze and draw inferences from patterns in **data**.





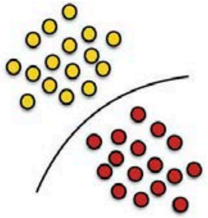
Also known as:
- pattern recognition
- artificial intelligence
- data mining
- predictive analytics

Goal is often to use data to create an algorithm/model that
- makes accurate predictions
- is interpretable, revealing (previously unknown) patterns in data

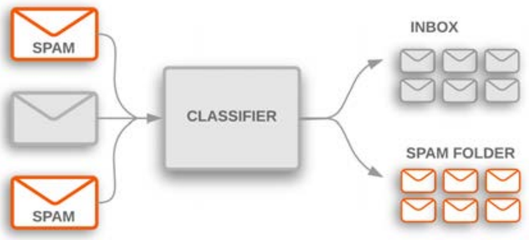
ML tasks

1. building and understanding methods that ‘**learn**’ by using **data** to improve performance on some set of tasks

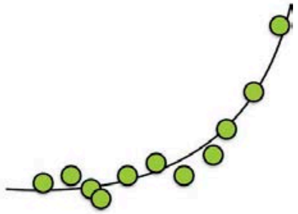
classification



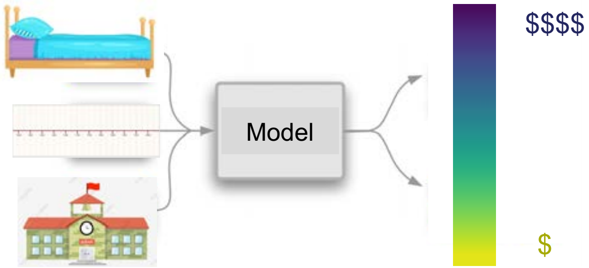
predict input data labels



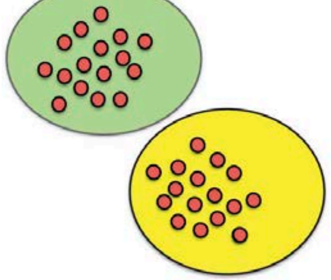
regression



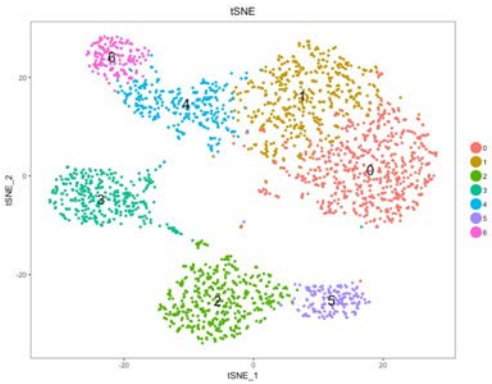
predict input data values



clustering



group input data

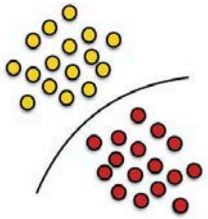


task images from from Carrasquilla 2020

CENTER FOR STATISTICS AND MACHINE LEARNING

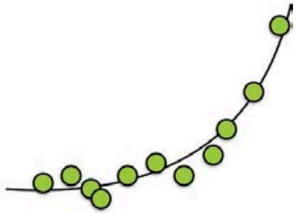
ML tasks

classification



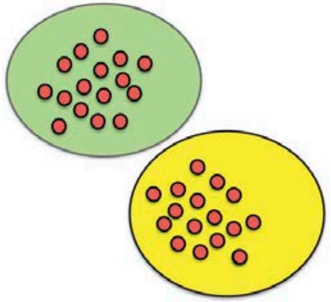
predict input data labels

regression



predict input data values

clustering



group input data

supervised

response (label/value) y_i

features $x_{i,1}, x_{i,2} \dots x_{i,n}$

sample i

ML learns the relationship between the features and response

unsupervised

$x_{i,1}, x_{i,2} \dots x_{i,n}$

ML learns patterns/groupings

Terminology

$$\text{sample } i \left\{ \boxed{\begin{matrix} y_i \\ x_{i,1}, x_{i,2} \dots x_{i,n} \end{matrix}} \right. \begin{matrix} \text{response (label/value)} \\ \text{features} \end{matrix}$$

sample i

- sample
- data point
- observation

y_i

- response
- target
- class (if categorical)
- outcome
- dependent variable

$x_{i,1}, x_{i,2} \dots x_{i,n}$

- features
- predictors
- descriptors
- attributes
- covariates
- independent variables

many terms in English, but the **math** is always the same!

Discrete/categorical or continuous values!

sample i $\left\{ \boxed{\begin{array}{c} y_i \\ x_{i,1}, x_{i,2} \dots x_{i,n} \end{array}} \right.$ response (label/value) features

examples

response y_i

- a sample's disease status (discrete)
- a sample's height/length (continuous)
- a house's market value (continuous)

features $(x_{i,1}, x_{i,2} \dots x_{i,n})$

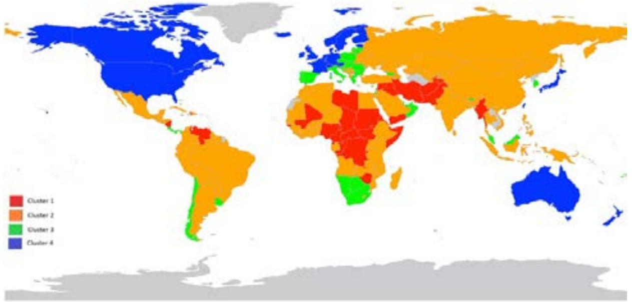
- the presence of a mutation in genome (discrete)
- cigarettes smoked per week (continuous)
- the age of a house (continuous)

Why use machine learning?

we want to
- know a future event
- make decision based on information
- look for useful patterns in data

examples
- supervised
    - should I sell this stock?
    - how many copies will this book sell?
    - will this customer move their business to a different company?
    - how much will my house sell for in the current market?
    - does a patient have a specific disease?
    - based on past choices, which movies will interest this viewer?
    - which people should we match in our online dating service?
    - will this patient respond to this therapy?

- unsupervised
    - how do customers differ from one another?
    - how are countries different in terms of socio-economic/health?
    - how many cell types are in my sample?



Why use machine learning?

<u>supervised</u>

- prediction
  - predict response value for new samples
- inference
  - understand *how* and *why* a model works

<u>unsupervised</u>

- learn underlying structure of data

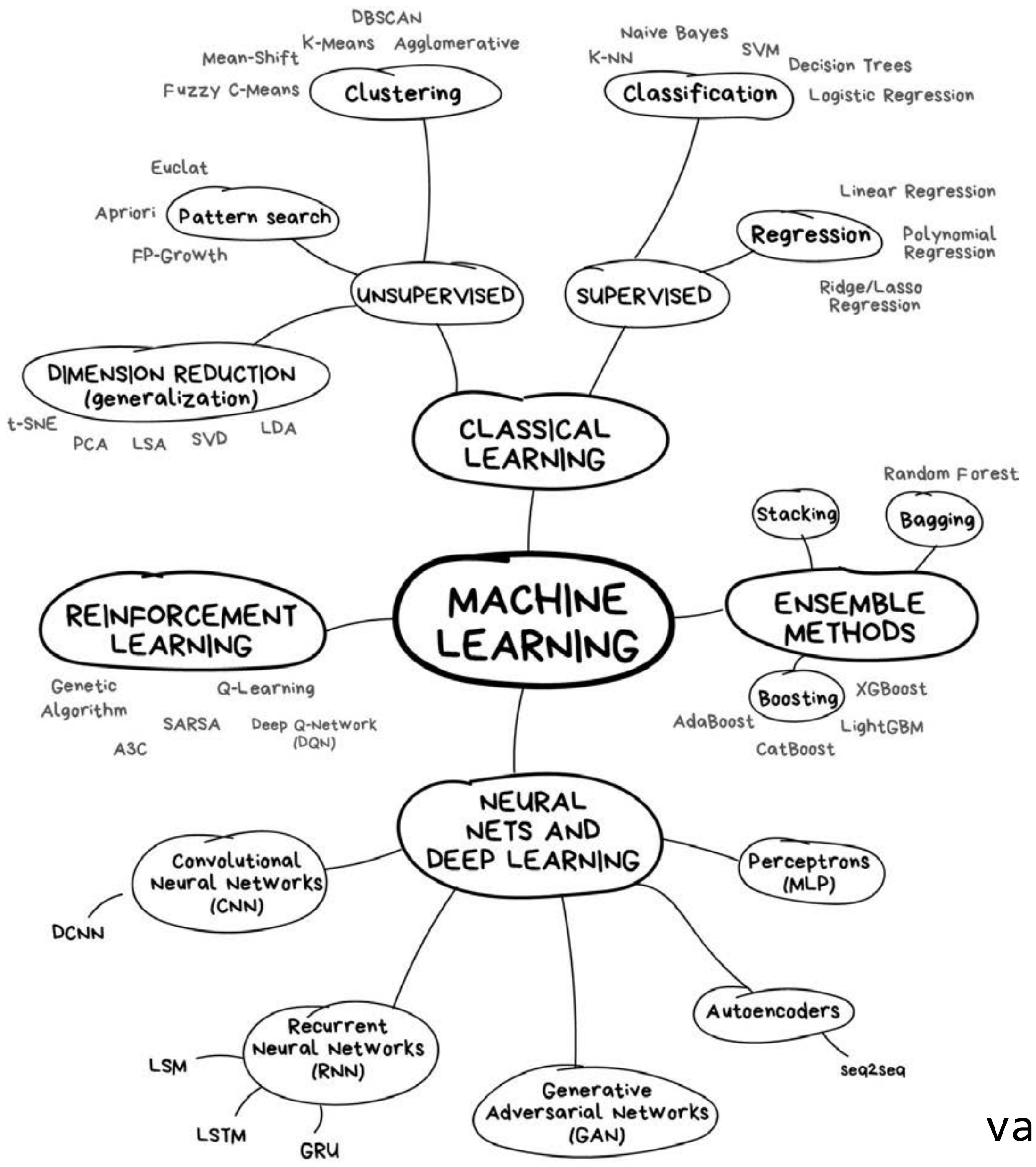
Overview of Machine Learning Process

Datasets? Input features? Targets? Evaluation metrics?

1. Define the problem to be solved.
2. Split the data into train / validation / test.
3. Run the validation loop:
   a. Choose a set of models.
   b. Train each model by optimizing its parameters on the training set.
   c. Evaluate the performance of each model on the validation set.
   d. Repeat until performance is satisfactory.
4. Evaluate final performance on the test set.

Overview of Machine Learning Process

1. Define the problem to be solved.
2. Split the data into train / validation / test.
3. Run the validation loop:
   a. Choose a set of models.
   b. Train each model by optimizing its parameters on the training set.
   c. Evaluate the performance of each model on the validation set.
   d. Repeat until performance is satisfactory.
4. Evaluate final performance on the test set.


MACHINE LEARNING
CLASSICAL LEARNING
UNSUPERVISED
SUPERVISED
Clustering
DBSCAN
K-Means
Agglomerative
Mean-Shift
Fuzzy C-Means
Classification
Naive Bayes
K-NN
SVM
Decision Trees
Logistic Regression
Pattern search
Euclat
Apriori
FP-Growth
Regression
Linear Regression
Polynomial Regression
Ridge/Lasso Regression
DIMENSION REDUCTION (generalization)
t-SNE
PCA
LSA
SVD
LDA
ENSEMBLE METHODS
Stacking
Bagging
Random Forest
Boosting
XGBoost
AdaBoost
LightGBM
CatBoost
REINFORCEMENT LEARNING
Genetic Algorithm
Q-Learning
SARSA
Deep Q-Network (DQN)
A3C
NEURAL NETS AND DEEP LEARNING
Convolutional Neural Networks (CNN)
DCNN
Perceptrons (MLP)
Recurrent Neural Networks (RNN)
LSM
LSTM
GRU
Generative Adversarial Networks (GAN)
Autoencoders
seq2seq

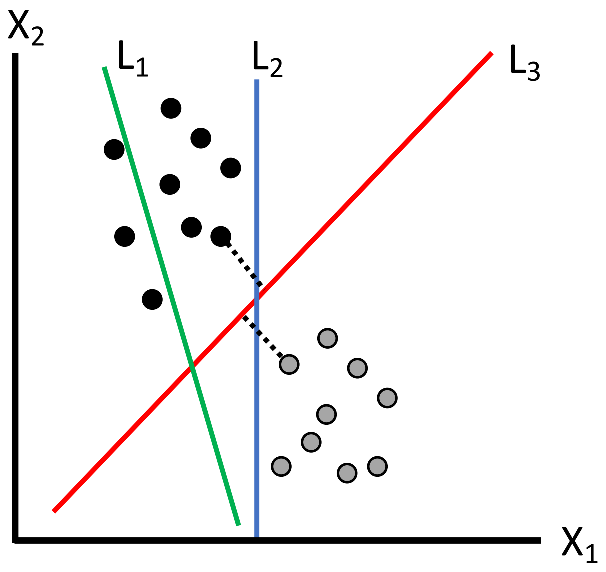
vas3k.com

Overview of Machine Learning Process

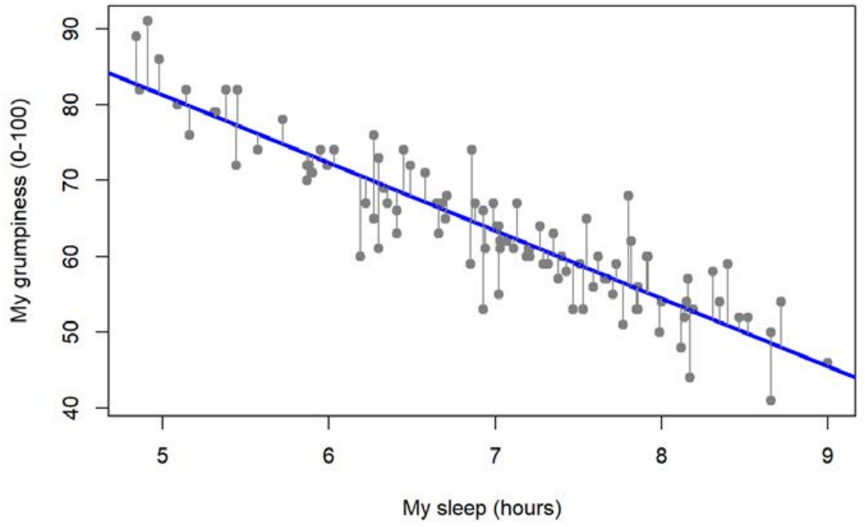
1. Define the problem to be solved.
2. Split the data into train / validation / test.
3. Run the validation loop:
   a. Choose a set of models.
   b. Train each model by optimizing its parameters on the training set.
   c. Evaluate the performance of each model on the validation set.
   d. Repeat until performance is satisfactory.
4. Evaluate final performance on the test set.

Model Training

e.g. find slope of line that **best** separates training labels



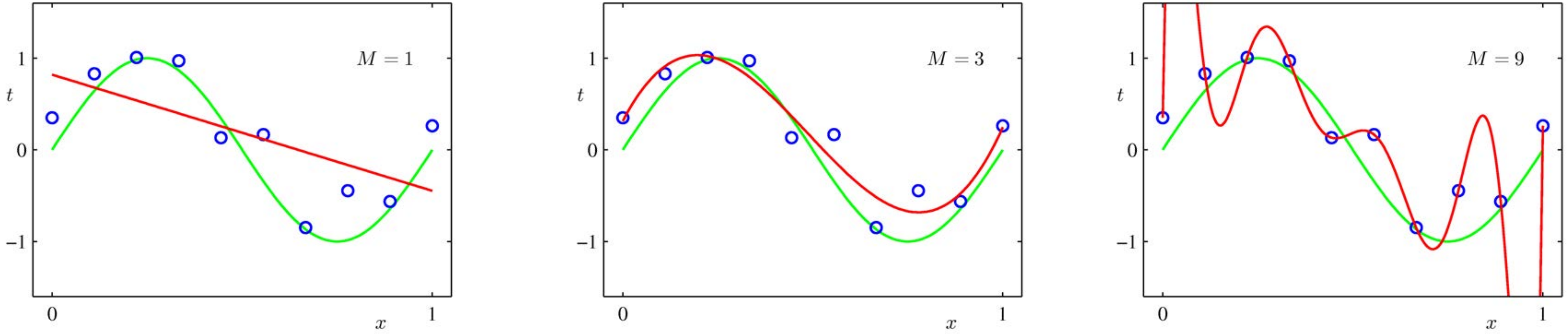
e.g. find slope of line that **best** predicts training values



Overview of Machine Learning Process

1. Define the problem to be solved.
2. Split the data into train / validation / test.
3. Run the validation loop:
   a. Choose a set of models.
   b. Train each model by optimizing its parameters on the training set.
   c. Evaluate the performance of each model on the validation set.
   d. Repeat until performance is satisfactory.
4. Evaluate final performance on the test set.

Model Complexity

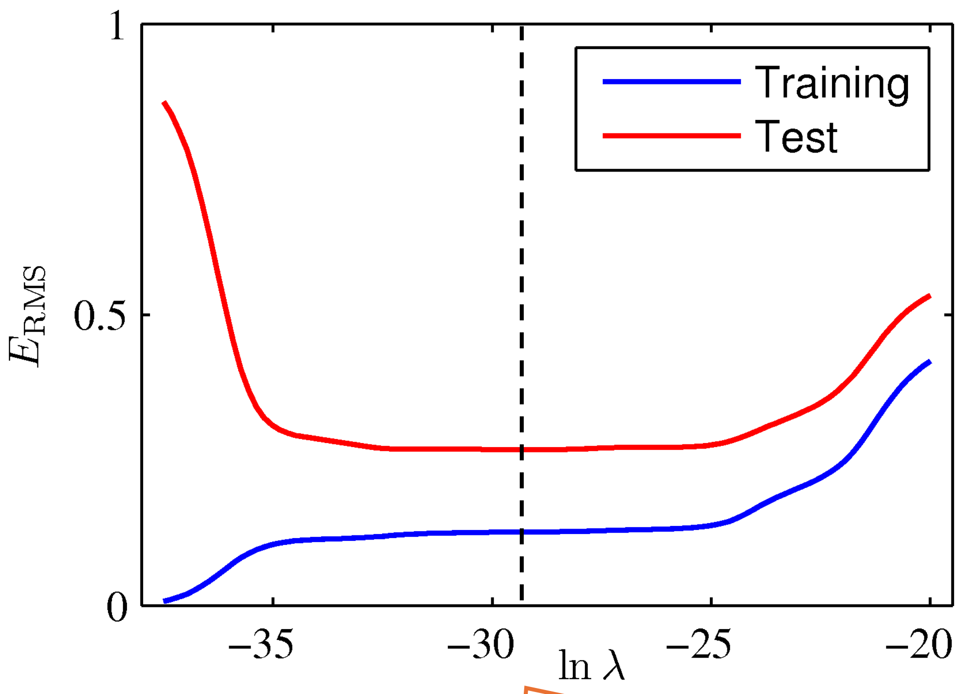


Underfitting
(misses general trend)

Overfitting
(captures noise)

Figure credit: Bishop, Christopher M. 2006. *Pattern Recognition and Machine Learning*.

Effect of Complexity on Test Performance



Best model has this value!

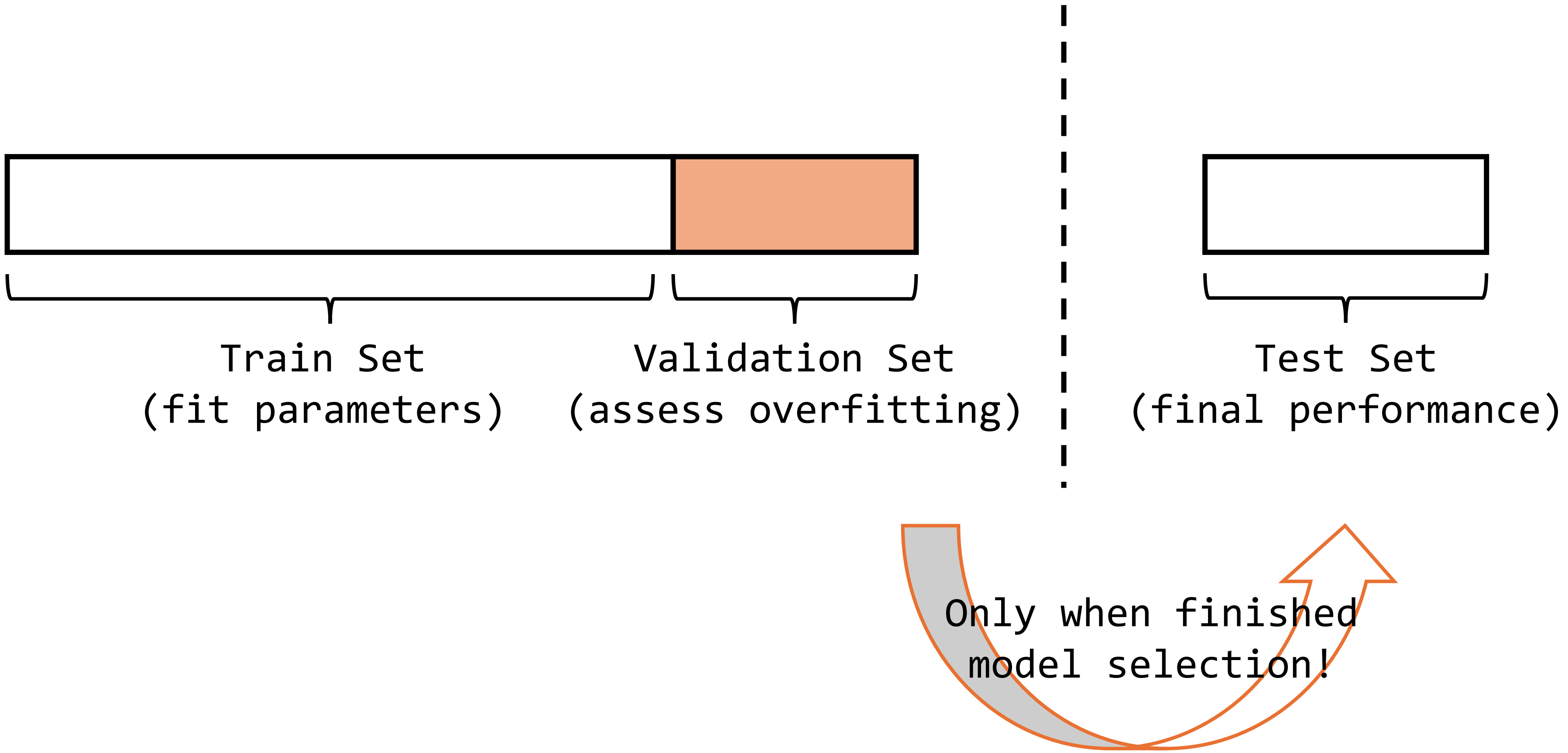
Figure credit: Bishop, Christopher M. 2006. *Pattern Recognition and Machine Learning*.

DataX
Accelerating Scientific Discovery at Princeton

Overview of Machine Learning Process

1. Define the problem to be solved.
2. Split the data into train / validation / test.
3. Run the validation loop:
   a. Choose a set of models.
   b. Train each model by optimizing its parameters on the training set.
   c. Evaluate the performance of each model on the validation set.
   d. Repeat until performance is satisfactory.
4. Evaluate final performance on the test set.

Role of the Validation Set



Overview of Machine Learning Process

1. Define the problem to be solved.
2. Split the data into train / validation / test.
3. Run the validation loop:
    a. Choose a set of models.
    b. Train each model by optimizing its parameters on the training set.
    c. Evaluate the performance of each model on the validation set.
    d. Repeat until performance is satisfactory.
4. Evaluate final performance on the test set.

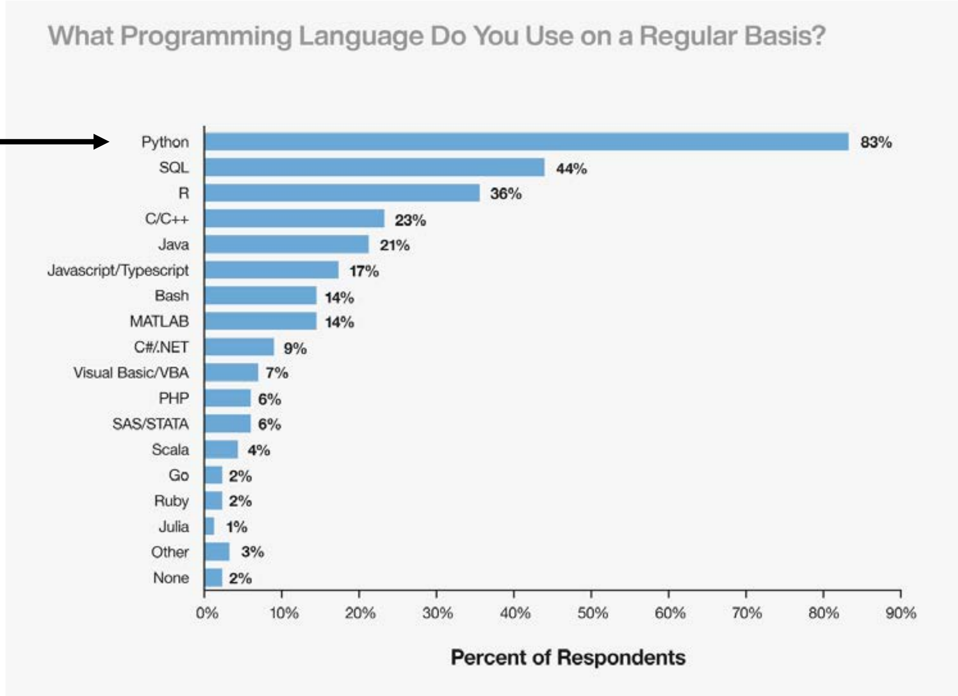
Simplified Machine Learning Process

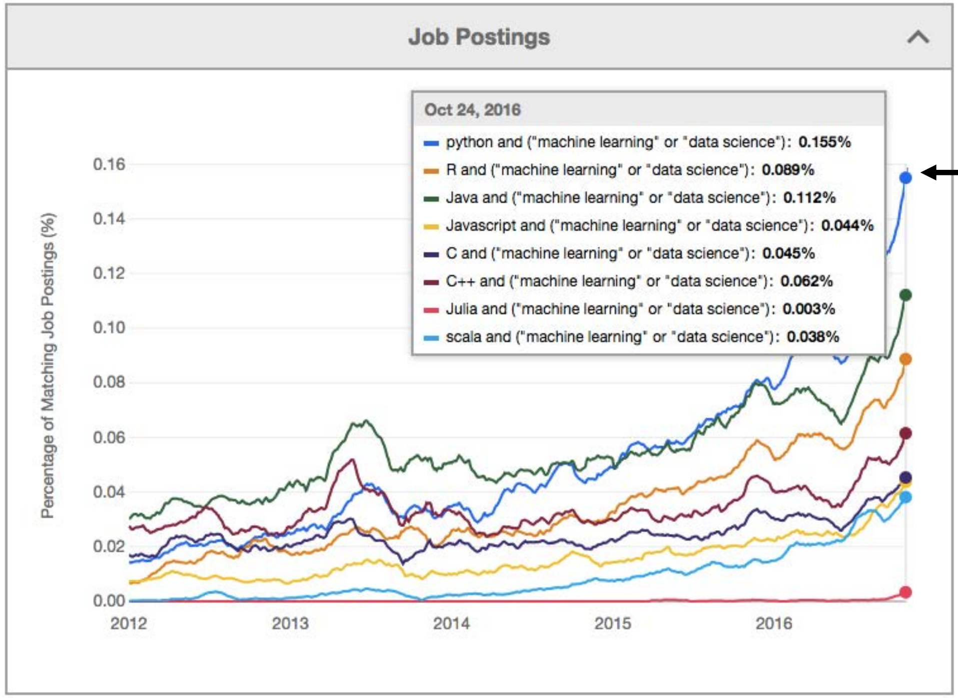
1. Download a dataset from the internet.
2. Use the predefined train / test split.
3. ~~Run the validation loop:~~
    a. Choose a ~~set of~~ model~~s~~.
    b. Train ~~each~~ model by optimizing its parameters on the training set.
    c. ~~Evaluate the performance of each model on the validation set.~~
    d. ~~Repeat until performance is satisfactory.~~
4. Evaluate final performance on the test set.

Machine learning in python

- several options for building ML models
- Python most popular and most in demand (job postings)
- R also popular in statistics and biology communities

python





python

Fantastic Python libraries

- data analysis
  - Pandas: great for analyzing and manipulating data tables
  - Seaborn: simple functions -> detailed visualization, integrated with Pandas
  - Matplotlib: visualization

- machine learning
  - numpy: fast, powerful data structures for matrices
  - scikit-learn: simple, efficient, accessible tools for ML
  - Keras: neural networks
  - TensorFlow
  - PyTorch
  - …

today we will use **numpy** and **scikit-learn**!

ML Coding Tour in Python!

Open the iPython notebook from this link!

https://github.com/PrincetonUniversity/intro_machine_learning/tree/main/day1

Intro to *K*-nearest neighbors (KNN)

- simple but powerful
- can be used for classification *or* regression!
- algorithm
  1. for a given test sample (yellow dot), find the *K* nearest training samples in feature space
  2a. for **classification**, assign label by majority vote
  2b. for **regression**, assign value by mean of neighbors

K is a tunable parameter!

- choose value that gives better predictions on test data