4/25/2024: Course Summary

① Tasks — What do we want to learn?
What resources are available?

Supervised Learning
- Learn to emulate function $x \to y$
- Data tells us correct y for some x's

Regression
$y \in \mathbb{R}$

Classification
y is discrete

Binary Classification
$y \in \{-1, 1\}$

Multi-class Classification
$y \in \{1, 2, 3, \ldots, k\}$

Unsupervised Learning
- Learn structure of x's
- Dataset only contains bunch of x's

Embeddings Word Vectors
For each word, associate it with a vector that capture word similarity, analogy, meaning

Clustering
Group structure

Dimensionality Reduction
Low dimensional structure

Reinforcement Learning
- Learn how to take good actions
- Data: After taking action, observe consequences

Bandits
State of world is fixed

Full RL
Actions change the state of the world

② Modeling — What "shape" does the desired solution have?

Parametric methods

- **Tabular Methods**: Remember predicted output for each possible input
  - e.g. Tabular Q-Learning
    - Stored a table containing prediction $\hat{Q}(s,a)$ for every state s & action a
- **Linear Model**: Can't enumerate all possible inputs so we assume $x \rightarrow y$ is a linear function
- **Neural Networks**: Assume $x \rightarrow y$ is complex non-linear function
  - MLP: Generic nonlinear function
  - CNN: Local structure matters, weight sharing
  - RNN: Sequential order matters, weight sharing
  - Transformer: Relationships between words matter, so use attention, weight sharing

add structure compared with basic MLP

- **Non-parametric Models**: Refer to training data to make predictions
  - K-NN: Similar x's have similar labels
  - Kernel methods: 2 motivations
    1. Similar x's have similar labels
    2. Doing linear method in more complex feature space $\phi(x)$

③ Loss Function: Quantifies how good/bad a possible solution is

Supervised Learning

Compare model prediction to true/desired output

- Regression: $(\underbrace{f(x)}_{\text{model's prediction}} - \underbrace{y}_{\text{true answer}})^2$
- Binary Classification: $-\log \sigma(\underbrace{\underbrace{y}_{\text{true label}} \cdot \underbrace{f(x)}_{\text{prediction}}}_{\text{margin}})$

Unsupervised Learning

Want "compressed" version of data, make it close to the original

- K-Means: $\sum_i \| \underbrace{x^{(i)}}_{\text{data}} - \underbrace{\mu_{z_i}}_{\text{assigned cluster mean}} \|^2$
- PCA: $\sum_i \| \underbrace{x^{(i)}}_{\text{data}} - \underbrace{\text{Proj}_W(x^{(i)})}_{\text{projection onto subspace spanned by } W} \|^2$

④ Optimization — How to minimize loss function w.r.t. model's parameters?

Correct when possible → • Direct Computation, e.g. set $\nabla_\theta \text{Loss}(\theta) = 0$

solve for θ

- Linear Regression: Normal Equations
- GMM M-step: Choose best N, Σ, π for each cluster

• Gradient Descent: $\theta^{(t+1)} \leftarrow \theta^{(t)} - \underbrace{\eta}_{\text{learning rate}} \nabla_\theta \text{Loss}(\theta^{(t)})$

- Neural networks (+ backpropagation to compute $\nabla_\theta \text{Loss}(\theta)$)
- Linear methods
- RL: Deep Q Networks, Policy gradient

• Alternating Minimization:

- K-Means
- EM

⑤ Maximum Likelihood Estimation — Come up with loss function via probabilistic story

Loss = negative log likelihood of data

- Linear / Logistic regression
- GMM
- Naive Bayes

⑥ Importance of Data — determines what you learn

- RL: Choice of action determines what you learn
  think about doing exploration
- Spurious Correlations
- Overfitting: too little data → more overfitting
  ↳ can mitigate w/ regularization or simpler model
  No substitute for more data