Problem Solving with AI Techniques
Machine Learning

Paul Weng

UM-SJTU Joint Institute

VE593, Fall 2018

JOINT INSTITUTE
交大密西根学院

Introduction

What is Machine Learning?

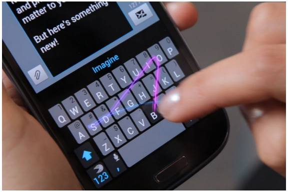
- Machine learning: technique that gives a computer system the ability to learn to perform a given task
- learn = improve itself as it sees more data, observations, interactions
- machine learning = "programming with data"

Why do we need Machine Learning?

- Some tasks are difficult to program
- Hand-coded programs are not adaptive

Applications

- Smart keyboard: Predict next word



- Credit scoring in finance: Predict financial health

- Visual search: Caption for an image


The man at bat readies to swing at the pitch while the umpire looks on.


A large bus sitting next to a very tall building.

Project 3: Recognizing Handwritten Digits

- Cleaned, normalized dataset of 70k images
- Hello world problem in machine learning
- One of the early success of artificial neural networks

High-level framework

The high-level framework is described as follows:

- Goal: learn a mapping from an input set $\mathcal{X}$ to an output set $\mathcal{Y}$
- Given $\boldsymbol{X} = (\boldsymbol{x}^1, \boldsymbol{x}^2, \dots, \boldsymbol{x}^N)^\intercal, \boldsymbol{y} = (y^1, y^2, \dots, y^N)^\intercal$
- Assumption: $\boldsymbol{X}, \boldsymbol{y}$ (statistically) representative of elements in $\mathcal{X}, \mathcal{Y}$

- What are $\boldsymbol{x}$ and y in the previous examples?

Classes of Learning Problems

Different classes of problems that depend on how much supervision is provided:

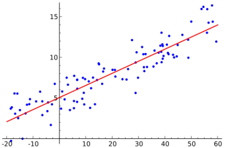
- Supervised Learning $(\boldsymbol{X}, \boldsymbol{y})$
- Weakly-supervised Learning (inexact or inaccurate $\boldsymbol{y}$)
- Semi-supervised Learning (y^k known only for some k)
- Reinforcement Learning
- Unsupervised Learning ($\boldsymbol{X}$ only!)

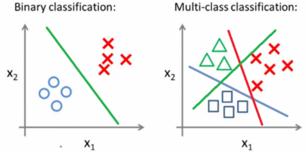
- Many other problems: active learning, transfer learning, multi-task learning, life-long learning....

Supervised Learning

Different classes of supervised learning depending on $\mathcal{Y}$:

- Regression: For each $\boldsymbol{x}$, predict a continuous y
- Classification: For each $\boldsymbol{x}$, predict a discrete y
  - binary classification if $|\mathcal{Y}| = 2$
  - multi-class classification otherwise
  - Classification can be turned into regression by prediction $\mathbb{P}(Y \mid X)$
- Structured prediction: For each $\boldsymbol{x}$, predict a structured object y (e.g., sequence, tree, graph, policy...)





The man at bat readies to swing at the pitch while the umpire looks on.

A large bus sitting next to a very tall building.

Formal Framework

- Input set $\mathcal{X}$
  e.g., $\mathbb{R}^n$, images, words
- Output set $\mathcal{Y}$
  e.g., $\mathbb{R}$, $\{0, 1\}$, sentences, actions
- Loss function $\ell : \mathcal{Y} \times \mathcal{Y} \to \mathbb{R}_+$
  e.g., squared error, 0,1-loss
- Concept class $\mathcal{C} \subset \mathcal{Y}^{\mathcal{X}}$
  e.g., linear functions from $\mathcal{X}$ to $\mathcal{Y}$, Bayes nets
- Hypothesis class $\mathcal{H} \subset \mathcal{Y}^{\mathcal{X}}$
  e.g., linear functions from $\mathcal{X}$ to $\mathcal{Y}$, Bayes nets
- Generative models $\mathcal{P}$ = set of probability distributions over $\mathcal{X} \times \mathcal{Y}$

- Why do we need a loss function? concept class? generative model?

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0, 1\}^3$ and $y \in \{0, 1\}$
- How many functions $f : \{0, 1\}^3 \to \{0, 1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0,1\}^3$ and $y \in \{0,1\}$
- How many functions $f : \{0,1\}^3 \to \{0,1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1		
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0,1\}^3$ and $y \in \{0,1\}$
- How many functions $f : \{0,1\}^3 \to \{0,1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1	1	
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0, 1\}^3$ and $y \in \{0, 1\}$
- How many functions $f : \{0, 1\}^3 \rightarrow \{0, 1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1	1	2^6
0	1	0		
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0,1\}^3$ and $y \in \{0,1\}$
- How many functions $f : \{0,1\}^3 \to \{0,1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1	1	2^6
0	1	0	1	
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0,1\}^3$ and $y \in \{0,1\}$
- How many functions $f : \{0,1\}^3 \to \{0,1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1	1	2^6
0	1	0	1	2^5
0	1	1		
1	0	0		
1	0	1		
1	1	0		
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0,1\}^3$ and $y \in \{0,1\}$
- How many functions $f : \{0,1\}^3 \to \{0,1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1	1	2^6
0	1	0	1	2^5
0	1	1	1	2^4
1	0	0	1	2^3
1	0	1	1	2^2
1	1	0	1	
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0,1\}^3$ and $y \in \{0,1\}$
- How many functions $f : \{0,1\}^3 \to \{0,1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1	1	2^6
0	1	0	1	2^5
0	1	1	1	2^4
1	0	0	1	2^3
1	0	1	1	2^2
1	1	0	1	2^1
1	1	1		

- We need an Inductive Bias!

Need for Concept Class

- Assume $\boldsymbol{x} \in \{0,1\}^3$ and $y \in \{0,1\}$
- How many functions $f : \{0,1\}^3 \rightarrow \{0,1\}$ are there?
- Assume we have seen the following examples. Can we generalize?

x_1	x_2	x_3	y	# consistent concepts
0	0	0	1	2^7
0	0	1	1	2^6
0	1	0	1	2^5
0	1	1	1	2^4
1	0	0	1	2^3
1	0	1	1	2^2
1	1	0	1	2^1
1	1	1	1	2^0

- We need an Inductive Bias!

Graphical Models

For example, using Bayes nets:

- Choose a structure (or learn from data)
- Learn parameters from data
- Use Bayes net for inference

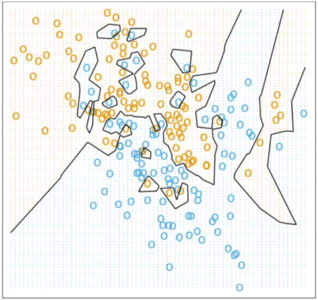
- Issue: structure hard to define/learn, inference may be hard to compute for complex structure

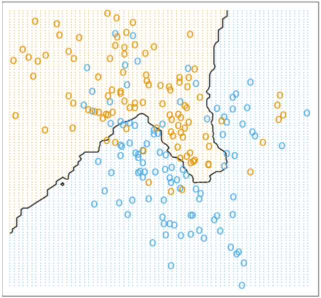
Naive Bayes:

- Idea: assume that all the X^i's are independent
- Works surprisingly well

k-Nearest Neighbors

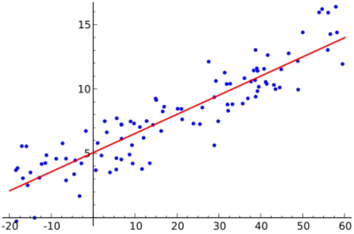
- Principle: for new $\boldsymbol{x}$, compute response as function of k nearest neighbors of $\boldsymbol{x}$ in dataset $\mathcal{D} = \{(\boldsymbol{x}^i, y^i)\}$
  - Classification: majority vote
  - Regression: average
- Issues:
  - High computational/space requirements if dataset large
  - Doesn't scale in high dimension
  - Which distance to use?

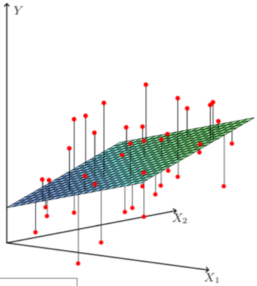


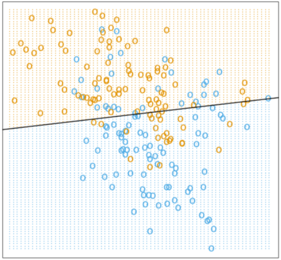


from Hastie et al.

Linear Models

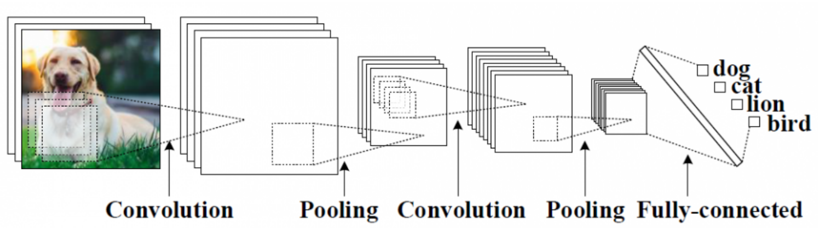


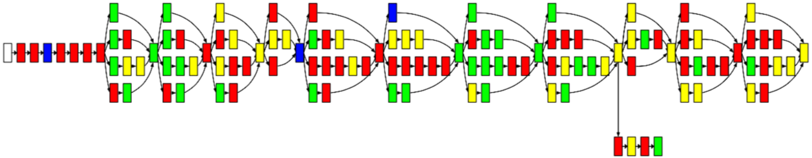




from Hastie et al.

Artificial Neural Networks





Learning Problem

- Expected risk or error: $R_\mu(h) = \mathbb{E}_{(X,Y)\sim\mu}[\ell(h(X), Y)]$ with $\mu \in \mathcal{P}$
- Ideally, $h^* = \arg\min_h R_\mu(h)$
- Bayes risk: $\min_h R_\mu(h)$
- h is Bayes optimal if $R_\mu(h)$ is equal to the Bayes risk
- Hard to solve because
    - μ is not known
    - optimization is over any h

Bayes Classifier

- Expected Loss with 0-1 Loss $\ell(h(\boldsymbol{x}), y) = [h(\boldsymbol{x}) \neq y]$
- Expected loss is the probability of error: $R_\mu(h) = \mathbb{P}(h(X) \neq Y)$
- Theorem. The Bayes classifier defined as $h^*(\boldsymbol{x}) = \arg\max_y \mathbb{P}(y \mid \boldsymbol{x})$ reaches the Bayes error.

Bayes Classifier

- Expected Loss with 0-1 Loss $\ell(h(\boldsymbol{x}), y) = [h(\boldsymbol{x}) \neq y]$
- Expected loss is the probability of error: $R_\mu(h) = \mathbb{P}(h(X) \neq Y)$
- Theorem. The Bayes classifier defined as $h^*(\boldsymbol{x}) = \arg\max_y \mathbb{P}(y \mid \boldsymbol{x})$ reaches the Bayes error.
- Proof. For simplicity written in the discrete case:

$$h^* = \arg\min_h R_\mu(h) = \arg\min_h \sum_{\boldsymbol{x}} \sum_y \ell(h(\boldsymbol{x}), y)\mu(\boldsymbol{x}, y)$$

$$h^*(\boldsymbol{x}) = \arg\min_{y'} \sum_y \ell(y', y)\mu(\boldsymbol{x}, y)$$

$$h^*(\boldsymbol{x}) = \arg\min_{y'} \sum_{y \neq y'} \mu(\boldsymbol{x}, y)$$

$$h^*(\boldsymbol{x}) = \arg\max_{y'} \mu(\boldsymbol{x}, y') = \arg\max_y \mathbb{P}(\boldsymbol{x}, y)/\mathbb{P}(\boldsymbol{x})$$

Bayes Regressor

- Expected Loss with squared error loss $\ell(h(\boldsymbol{x}), y) = (h(\boldsymbol{x}) - y)^2$
- Expected loss is mean squared error: $R_\mu(h) = \mathbb{E}[(h(X) - Y)^2]$

- Theorem. The Bayes regressor defined as $h^*(\boldsymbol{x}) = \arg\max_y \mathbb{E}(y \mid \boldsymbol{x})$ reaches the Bayes error.

Statistical Learning Problem

- Issue: We don't know μ, Bayes risk cannot be reached generally
- Idea: given $\mathcal{D} = \{(\boldsymbol{x}^i, y^i) \mid i = 1, \dots, N\}$ where $(\boldsymbol{x}^i, y^i) \sim \mu \in \mathcal{P}$, find $H : \mathcal{X} \to \mathcal{Y} \in \mathcal{H}$ that approximately minimizes the loss $\ell(H(X), Y)$ for $(X, Y) \sim \mu$
- Empirical Risk Minimization: solve:

$$H^* = \arg\min_{H \in \mathcal{H}} R_{\mathcal{D}}(H) \text{ where } R_{\mathcal{D}}(H) = \sum_{i=1}^{N} \ell(H(\boldsymbol{x}^i), y^i)$$

- What are the possible issues with this approach?

Statistical Learning Problem

- Issue: We don't know μ, Bayes risk cannot be reached generally
- Idea: given $\mathcal{D} = \{(\boldsymbol{x}^i, y^i) \mid i = 1, \ldots, N\}$ where $(\boldsymbol{x}^i, y^i) \sim \mu \in \mathcal{P}$, find $H : \mathcal{X} \to \mathcal{Y} \in \mathcal{H}$ that approximately minimizes the loss $\ell(H(X), Y)$ for $(X, Y) \sim \mu$
- Empirical Risk Minimization: solve:

$$H^* = \arg\min_{H \in \mathcal{H}} R_{\mathcal{D}}(H) \text{ where } R_{\mathcal{D}}(H) = \sum_{i=1}^{N} \ell(H(\boldsymbol{x}^i), y^i)$$

- What are the possible issues with this approach?
  - Empirical risk is only an approximation of the true risk

Statistical Learning Problem

- Issue: We don't know μ, Bayes risk cannot be reached generally
- Idea: given $\mathcal{D} = \{(\boldsymbol{x}^i, y^i) \mid i = 1, \ldots, N\}$ where $(\boldsymbol{x}^i, y^i) \sim \mu \in \mathcal{P}$, find $H : \mathcal{X} \to \mathcal{Y} \in \mathcal{H}$ that approximately minimizes the loss $\ell(H(X), Y)$ for $(X, Y) \sim \mu$
- Empirical Risk Minimization: solve:

$$H^* = \arg\min_{H \in \mathcal{H}} R_{\mathcal{D}}(H) \text{ where } R_{\mathcal{D}}(H) = \sum_{i=1}^{N} \ell(H(\boldsymbol{x}^i), y^i)$$

- What are the possible issues with this approach?
  - Empirical risk is only an approximation of the true risk
  - More complex hypothesis class can lead to smaller empirical risk

Statistical Learning Problem

- Issue: We don't know μ, Bayes risk cannot be reached generally
- Idea: given $\mathcal{D} = \{(\boldsymbol{x}^i, y^i) \mid i = 1, \ldots, N\}$ where $(\boldsymbol{x}^i, y^i) \sim \mu \in \mathcal{P}$, find $H : \mathcal{X} \to \mathcal{Y} \in \mathcal{H}$ that approximately minimizes the loss $\ell(H(X), Y)$ for $(X, Y) \sim \mu$
- Empirical Risk Minimization: solve:

$$H^* = \arg\min_{H \in \mathcal{H}} R_{\mathcal{D}}(H) \text{ where } R_{\mathcal{D}}(H) = \sum_{i=1}^{N} \ell(H(\boldsymbol{x}^i), y^i)$$

- What are the possible issues with this approach?
  - Empirical risk is only an approximation of the true risk
  - More complex hypothesis class can lead to smaller empirical risk
  - We are in fact interested in $\sum_{\boldsymbol{x}, y \in \mathcal{D}'} \ell(H(\boldsymbol{x}), y)$ where $\mathcal{D}'$ new data set