10-701 Machine Learning Review

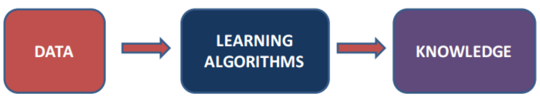
Frozen Heart

March 20, 2017

1 Intro

1.1 What is Machine Learning?

Algorithms that improve their knowledge towards some task with data.
Goal: improve knowledge with more data.



1.2 Three Axes of ML

1.2.1 Data

- Fully observed
- Partially observed
  - Systematically
  - Missing data

1.2.2 Algorithms

- Model-based Methods[1]
  - Probabilistic Model of the data
  - Parametric Models: fixed-size
  - Nonparametric Models: grow with the data
- Model-free Methods: No distribution model assumption

1.2.3 Knowledge/Tasks

- Prediction: estimate output given input.
  - Classification: discrete labels
  - Regression: continuous labels
- Description (unsupervised learning)

[1] refer to generative model (hw2)

2 Parametric Models: from data to models

2.1 A model for coin flips

Bernoulli distribution:

$$\begin{cases} P(X=1) & = & \theta \\ P(X=0) & = & 1-\theta \end{cases} \tag{1}$$

$$P(X) = \theta^X (1-\theta)^{1-X}$$

Flips are i.i.d. (independent, identically distributed)
Choose θ that maximizes the probability of observed data:

$$\begin{aligned} Probability\ of\ Data &= \mathbb{P}(X_1, X_2, \ldots, X_n; \theta) \\ &= \prod_{i=1}^n P(X_i) \\ &= \theta^{n_h}(1-\theta)^{n-n_h} \end{aligned} \tag{2}$$

2.2 Maximum Likelihood Estimator(MLE)

$$\begin{aligned} \hat{\theta} &= \arg\max_\theta \mathbb{P}(X_1, \ldots, X_n; \theta) \\ &= \arg\max_\theta \{\theta^{n_h}(1-\theta)^{n-n_h}\} \\ &= \arg\max_\theta \{n_h \log\theta + (n-n_h)\log(1-\theta)\} \\ &= \frac{n_h}{n} \end{aligned} \tag{3}$$

2.2.1 Consistency

- Estimator $\hat{\theta}$ converges (in probability) to the true value θ with more and more sample $n \to \infty$.
- For Bernoulli distribution, $\hat{\theta} = \frac{1}{n}\sum_{i=1}^n X_i \to \theta$ in probability as $n \to \infty$ by the **Law of Large Numbers**[2].

2.2.2 Unbiasedness

- Expectation $\mathbb{E}[\hat{\theta}]$ of the estimator $\hat{\theta}$ equals to the true value θ.
- For Bernoulli example:

$$\begin{aligned} \mathbb{E}(\hat{\theta}) &= \mathbb{E}(\frac{n_1}{n}) \\ &= \mathbb{E}(\frac{\sum_{i=1}^n X_i}{n}) \\ &= \frac{1}{n}\sum_{i=1}^n \mathbb{E}(X_i) \\ &= \mathbb{E}(X_1) \\ &= \theta \end{aligned} \tag{4}$$

[2] It does not apply to distributions for whom Expected values do not exist. One example of such a distribution is the Cauchy distribution where the mean and the variance are undefined.

2.3 Gaussian Distribution MLE

Gaussian Distribution:

$$P(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}}\exp(-\frac{(x-\mu)^2}{2\sigma^2}) = \mathcal{N}(\mu,\sigma^2)$$

- Affine transformation:
  - $X \sim \mathcal{N}(\mu,\sigma^2)$
  - $Y = aX + b \sim \mathcal{N}(a\mu + b, a^2\sigma^2)$
- Sum of Gaussians:
  - $X \sim \mathcal{N}(\mu_X,\sigma_X^2)$, $Y \sim \mathcal{N}(\mu_Y,\sigma_Y^2)$
  - $Z = X + Y \sim \mathcal{N}(\mu_X + \mu_Y, \sigma_X^2 + \sigma_Y^2)$

MLE for Gaussian mean and variance:

- $\hat{\mu}_{MLE} = \frac{1}{n}\sum_{i=1}^{n} x_i$
- $\hat{\sigma}^2_{MLE} = \frac{1}{n}\sum_{i=1}^{n}(x_i - \hat{\mu})^2$

2.3.1 The Biased Variance of a Gaussian

The unbiased variance estimator: $\hat{\sigma}^2_{unbiased} = \frac{n}{n-1}\hat{\sigma}^2_{MLE}$
Proof:

$$\begin{array}{rcl} \mathbb{E}(\sigma^2_{MLE}) & = & \mathbb{E}(\frac{1}{n}\sum_{i=1}^{n}(x_i-\hat{\mu})^2) \\ & = & \frac{1}{n}\mathbb{E}(\sum_{i=1}^{n}x_i^2 - n\hat{\mu}^2) \\ & = & \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}(x_i^2) - \mathbb{E}(\hat{\mu}^2) \\ & = & \mathbb{E}(x_i^2) - \mathbb{E}(\hat{\mu}^2) \\ & = & (\sigma^2(x_i) + \mathbb{E}(x_i)^2) - (\sigma^2(\hat{\mu}) + \mathbb{E}(\hat{\mu})^2) \\ & = & \sigma^2(x_i) - \sigma^2(\hat{\mu}) \\ & = & \sigma^2(x_i) - \sigma^2(\frac{1}{n}\sum_{i=1}^{n}x_i) \\ & = & \sigma^2(x_i) - \frac{1}{n^2}\sigma^2(\sum_{i=1}^{n}x_i) \\ & = & \sigma^2(x_i) - \frac{1}{n^2}n\sigma^2(x_i) \\ & = & \frac{n-1}{n}\sigma^2(x_i) \end{array} \quad (5)$$

2.4 Convergence Rates of Estimator

2.4.1 Simple Bound (Hoeffding's Inequality)

$$P(|\hat{\theta} - \theta^*| \geq \epsilon) \leq 2\exp(-2n\epsilon^2)$$

2.5 PAC* (Probably Approximate Correct) Learning

2.6 Computational Issues of MLE

When number of parameters, or number of samples n is large, computing the MLE is a large-scale optimization problem.

3 Parametric Models: Prior Information

3.1 Bayesian Learning

Given a prior knowledge to estimate the model.
Bayesian Learning:

$$P(\theta|\mathcal{D}) = \frac{P(\mathcal{D}|\theta)P(\theta)}{P(\mathcal{D})}$$

or equivalently

$$P(\theta|\mathcal{D}) \propto P(\mathcal{D}|\theta)P(\theta)$$

- $P(\theta|\mathcal{D})$: posterior
- $P(\mathcal{D}|\theta)$: likelihood
- $P(\theta)$: prior

Likelihood measures the fitness between data and parameters, Prior is the knowledge how possible the parameters to be.

3.2 Conjugate Priors

- Closed-form representation of posterior
- $P(\theta)$ and $P(\theta|D)$ have the same algebraic form as a function of θ

For Binomial(Bernoulli), conjugate prior is Beta distribution:

- $P(D|\theta) = \theta^{\alpha_H}(1-\theta)^{\alpha_T}$
- $P(\theta) = \frac{\theta^{\beta_H - 1}(1-\theta)^{\beta_T - 1}}{B(\beta_H,\beta_T)} \sim Beta(\beta_H,\beta_T)$
- $P(\theta|D) \sim Beta(\beta_H + \alpha_H, \beta_T + \alpha_T)$
- Mode of Beta distribution $Beta(\alpha_H,\alpha_T)$: $\frac{\alpha_H - 1}{\alpha_H + \alpha_T - 2}$

3.3 Maximum A Posteriori Estimation (MAP)

Choose θ that maximizes a posterior probability:

$$\begin{array}{rcl} \hat{\theta}_{MAP} & = & \arg\max_\theta P(\theta|D) \\ & = & \arg\max_\theta P(D|\theta)P(\theta) \end{array} \quad (6)$$

3.4 Regularized MLE

Constrained MLE:

$$\max_\theta \log \mathbb{P}(D;\theta)$$

$$s.t. \mathcal{R}(\theta) \leq C$$

Regularized MLE:

$$\max_\theta\{\log \mathbb{P}(D;\theta) + \lambda\mathcal{R}(\theta)\}$$

- l_2 regularization: (Ridge?)

$$\mathcal{R}(\theta) = ||\theta||_2^2 = \sum_{j=1}^{p} \theta_j^2$$

- l_1 regularization: (Lasso)

$$\mathcal{R}(\theta) = ||\theta||_1 = \sum_{j=1}^{p} |\theta_j|$$

4 Linear Regression

4.1 Bayes Optimal Rule

$$f^* = \arg\min_f \mathbb{E}[loss(Y, f(X))]$$

4.2 Linear Regression

$$\hat{f}_n^L = \arg \min_{f \in \mathcal{F}_L} \frac{1}{n} \sum_{i=1}^{n} (f(X_i) - Y_i)^2$$

$$f(X) = X\beta$$

$$\hat{\beta} = \arg\min_\beta \frac{1}{n} (A\beta - Y)^T (A\beta - Y)$$

If $(A^T A)$ is invertible,

$$\hat{\beta} = (A^T A)^{-1} A^T Y$$

4.3 Regularized Least Squares

Guarantee solution uniqueness by adding a regular constraint.
Ridge Regression:

$$\begin{aligned} \hat{\beta}_{MAP} &= \arg\min_\beta \textstyle\sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda||\beta||_2^2 \\ &= \arg\min_\beta (A\beta - Y)^T (A\beta - Y) + \lambda||\beta||_2^2 \\ &= (A^T A + \lambda I)^{-1} A^T Y \end{aligned} \tag{7}$$

Lasso Regression:

$$\hat{\beta}_{MAP} = \arg\min_\beta \sum_{i=1}^{n} (Y_i - X_i\beta)^2 + \lambda||\beta||_1$$

More sparse solution.

5 Logistic Regression

Assumes the following functional form for $P(Y|X)$

$$P(Y=0|X) = \frac{1}{1+\exp(w_0 + \sum_i w_i X_i)}$$

5.1 Linear Classifier

Decision boundary: $w_0 + \sum_i w_i X_i = 0$

5.2 Training Logistic Regression

$$\hat{w}_{MLE} = \arg\max_w \prod_{i=1}^{n} P(X_i, Y_i|w)$$

$P(X)$ and $P(X|Y)$ are unknown. Discriminative philosophy[3]

$$\hat{w}_{MCLE} = \arg\max_w \prod_{i=1}^{n} P(Y_i|X_i, w)$$

5.2.1 Conditional Log Likelihood

$$P(Y=0|X,w) = \frac{1}{1+\exp w_0 + \sum_i w_i X_i}$$

$$P(Y=1|X,w) = \frac{\exp(w_0 + \sum_i w_i X_i)}{1+\exp(w_0 + \sum_i w_i X_i)}$$

$$\begin{aligned} l(w) &= \ln \textstyle\prod_i P(y^i|x^i, w) \\ &= \textstyle\sum_i [y^i(w_0 + \sum_j^d w_j x_j^i \\ &\quad - \ln(1+\exp(w_0 + \sum_j^d w_j x_j^i))] \end{aligned} \tag{8}$$

- no-closed form solution
- $l(w)$ is concave function of w

5.2.2 Gradient Ascent for LR

$$\frac{\partial l(w)}{\partial w_0} = \sum_i [y^i - P(Y^i = 1|x^i, w)]$$

$$\frac{\partial l(w)}{\partial w_j} = \sum_i x_j^i [y^i - P(Y^i = 1|x^i, w)]$$

[3] Don't waste effort learning $P(X)$, focus on $P(Y|X)$, that's all that matters for classification.

6 Naive Bayes Classifier

6.1 Optimal Classification

Optimal predictor: $f^* = \arg\min_f P(f(X) \neq Y)$
Optimal classifier:

$$\begin{aligned} f^*(x) &= \arg\max_{Y=y} P(Y=y|X=x) \\ &= \arg\max_{Y=y} P(X=x|Y=y)P(Y=y) \end{aligned} \tag{9}$$

- Class conditional density: $P(X=x|Y=y)$
- Class prior: $P(Y=y)$

Naive Bayes Classifier is a model based approach: to model these two terms.

6.2 Gaussian Bayes Classifier

- $P(Y=y) = p_y$ ($K-1$ if K labels)
- $P(X=x|Y=y \sim N(\mu_y, \Sigma_y)$ ($\frac{Kd+Kd(d+1)}{2} = O(Kd^2)$ if d features)

Binary classification:

$$P(X=x|Y=y) = \frac{1}{\sqrt{(2\pi)^d |\Sigma_y|}} \exp\left(-\frac{(x-\mu_y)^T \Sigma_y^{-1} (x-\mu_y)}{2}\right)$$

$$\frac{P(Y=1|X=x)}{P(Y=0|X=x)}$$

$$= \sqrt{\frac{|\Sigma_0|}{\Sigma_1}} \exp\left(-\frac{(x-\mu_1)^T \Sigma_1^{-1}(x-\mu_1)}{2} - \frac{(x-\mu_0)^T \Sigma_0^{-1}(x-\mu_0)}{2}\right)$$

$$\cdot \frac{\theta}{1-\theta}$$

If $\Sigma_0 = \Sigma_1$, then quadratic part cancels out and equation is linear.

6.3 Naive Bayes Classifier

Naive assumption: Features are independent given class:

$$P(X_1, \ldots, X_d|Y) = \prod_{i=1}^{d} P(X_i|Y)$$

$$f_{NB} = \arg\max_y P(x_1, \ldots, x_d|y)P(y) = \arg\max_y \prod_{i=1}^{d} P(X_i|Y)P(y)$$

$P(X_i = x_i|Y=y) \sim N(\mu_i, \sigma_i^2)$ (2Kd)

- Features are not conditionally independent.
- Insufficient data $\rightarrow$ MLE to be 0. Typically use MAP estimates.