Linear Algorithms: An Introduction

Mia Feng

March 22, 2019

Catalogue

- Definition
- Classification
  - Perceptron
  - Logistic Regression
  - SVM
- Regression
  - OLS
  - Bayes Regression
- Comparision

What are LAs?

Linear Algorithms

$$f(\mathrm{x}) = w_1 x_1 + w_2 x_2 + \cdots + w_d x_d + b. \tag{1}$$

To be clear,

$$f(\mathrm{x}) = \mathrm{w}^{\mathrm{T}}\mathrm{x} + b, \tag{2}$$

where $\mathrm{w} = (w_1; w_2; \ldots; w_d)$, $\mathrm{x} = (x_1; x_2; \ldots; x_d)$.

Tasks

- classification
- regression

General Processes

- Set hypothesis
  e.g. $y = x_1 + 2x_2$
- Define cost function
  e.g. RMSE, MSE
  Accurate, Recall
- Find the optimal solution
  e.g. gradient descent (GD)⇒Large dataset?
  stochastic gradient descent (SGD)⇒optimal?
  batch gradient descent⇒reasonable?
- Evaluate and model selection: cross validation + regularization
  hold-out⇒accidental error?
  leave-one-out⇒large dataset?
  k-fold⇒reasonable?

Symbols and Notation

- J: cost function/control function
- f (x): the hypothesis, a function of x with parameters w, where w is a scalar or a vector.
- b: intercept (scalar)
- x: feature, which is a vector or a scalar
- X: collection of features.
- y: label, which is a vector or a scalar
- ε: Gaussian noise, aka follows a Guassian distribution with zero mean and variance σ^2
- η: learning rate

Perceptron-1

Idea: Minimize the number of misclassified samples.

$$J(\mathrm{w}, b) = -\sum_{\mathrm{x}_i \in \mathrm{X}} y_i \left(\mathrm{w}^{\mathrm{T}} \mathrm{x}_i + b\right). \tag{3}$$

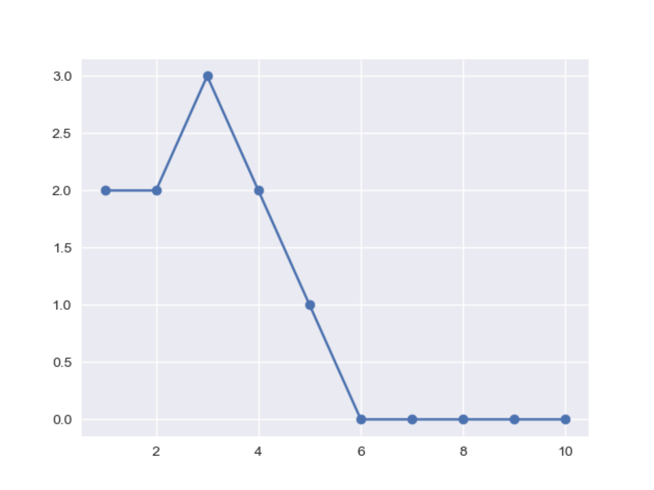
Solve: SGD

1. Initialize w, b.
2. Repeat until no misclassified samples{
   a. Choose (x_i, y_i) randomly,
   b. If $y_i \left(\mathrm{w}^{\mathrm{T}} \mathrm{x}_i + b\right) \leq 0$:

$$\begin{aligned} \mathrm{w} &\leftarrow \mathrm{w} + \eta y_i \mathrm{x}_i, \\ b &\leftarrow b + \eta y_i. \end{aligned} \tag{4}$$

}

Perceptron-2



The number of misclassified samples in each iteration.

Logistic Regression

Idea: Log MLE

$$J(\mathrm{w}) = -\sum_{i=1}^{m} y_i \ln f(\mathrm{x}_i) + (1 - y_i) \ln (1 - f(\mathrm{x}_i)). \tag{5}$$

Solve: GD

1. Initialize w, b.
2. Repeat until convergence{

$$\begin{aligned} \mathrm{w} &\leftarrow \mathrm{w} - \eta \tfrac{1}{m} \sum_{i=1}^{m} (f(\mathrm{x}_i) - y_i)\mathrm{x}_i, \\ b &\leftarrow b - \eta \tfrac{1}{m} \sum_{i=1}^{m} (f(\mathrm{x}_i) - y_i). \end{aligned} \tag{6}$$

}

SVM-1

Idea: Maximize the geometry distance between hyperplanes

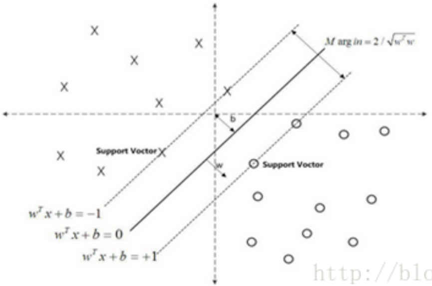
$$\begin{aligned} \min_{\mathrm{w},b} \quad & \frac{1}{2}\left\|\mathrm{w}\right\|^2, \\ s.t. \quad & y_i\left(\mathrm{w}\mathrm{x}_i + b\right) - 1 \geq 0, \quad i = 1, 2, \cdots, m; \end{aligned} \tag{7}$$

to be simpler,

$$\begin{aligned} \min_{\alpha} \quad & \frac{1}{2}\sum_{i,j=1}^{m} \alpha_i\alpha_j y_i y_j \mathrm{x}_i^{\mathrm{T}}\mathrm{x}_j - \sum_{i=1}^{m} \alpha_i, \\ s.t. \quad & \sum_{i=1}^{m} \alpha_i y_i = 0, \\ & \alpha_i \geq 0, \quad i = 1, 2, \cdots, m. \end{aligned} \tag{8}$$

SVM-2

如下图所示，中间的实线便是寻找到的最优超平面（Optimal Hyper Plane），其到两条虚线的距离相等，这个距离便是几何间隔$\tilde{\gamma}$，两条虚线之间的距离等于$2\tilde{\gamma}$，而虚线上的点则是支持向量。由于这些支持向量刚好在边界上，所以它们满足$y(w^Tx+b)=1$（还记得我们把 functional margin 定为 1 了吗？上节中：处于方便推导和优化的目的，我们可以令$\hat{\gamma}$ =1），而对于所有不是支持向量的点，则显然有$y(w^Tx+b)>1$。



Hyperplane and support vector

SVM-3: See equations (p125-p131) in Book by Hang Li.

算法 7.5（SMO 算法）

输入：训练数据集 $T=\{(x_1,y_1),(x_2,y_2),\cdots,(x_N,y_N)\}$，其中，$x_i\in\mathcal{X}=\mathbf{R}^n$，$y_i\in\mathcal{Y}=\{-1,+1\}$，$i=1,2,\cdots,N$，精度 ε；

输出：近似解 $\hat{\alpha}$.

（1）取初值 $\alpha^{(0)}=0$，令 $k=0$；

（2）选取优化变量 $\alpha_1^{(k)},\alpha_2^{(k)}$，解析求解两个变量的最优化问题 (7.101) ～ (7.103)，求得最优解 $\alpha_1^{(k+1)},\alpha_2^{(k+1)}$，更新 α 为 $\alpha^{(k+1)}$；

（3）若在精度 ε 范围内满足停机条件

$$\sum_{i=1}^{N}\alpha_i y_i=0$$

$$0\leqslant\alpha_i\leqslant C,\quad i=1,2,\cdots,N$$

$$y_i\cdot g(x_i)=\begin{cases}\geqslant 1, & \{x_i\mid\alpha_i=0\}\\ =1, & \{x_i\mid 0<\alpha_i<C\}\\ \leqslant 1, & \{x_i\mid\alpha_i=C\}\end{cases}$$

其中，

$$g(x_i)=\sum_{j=1}^{N}\alpha_j y_j K(x_j,x_i)+b$$

则转 (4)；否则令 $k=k+1$，转 (2)；

（4）取 $\hat{\alpha}=\alpha^{(k+1)}$.

SMO algorithm

Ordinary Least Square

Idea: Minimize MSE

$$J(\mathrm{w}) = \frac{1}{2}\sum_{i=1}^{m}\left(f(\mathrm{x_i}) - y_i\right)^2. \tag{9}$$

Solve: GD

Repeat until convergence{

$$w_j = w_j + \eta\sum_{i=1}^{m}\left(y_i - f(\mathrm{x}_i)\right). \tag{10}$$

}

Bayes Regression

Idea: MAP

$$\log p\,(y|\mathrm{X}) = -\frac{1}{2}y^{\mathrm{T}}K^{-1}y - \frac{1}{2}\log|K| - \frac{m}{2}\log 2\pi, \tag{11}$$

where K is the covariance matrix of training data X.
For more details, read "GPR for ML" by Carl Edward Rasmussen (p8-p22).

Solve

Repeat until convergence or out of the iteration limitations.

Classification and Regression

cost function

- Regression: RMSE, MSE etc.
- Classification: the number of misclassified samples, recall, likelihood etc.

General

Derivative

Determined and Stochastic

- Determined:
  - Geometry: SVM
  - The number of misclassified samples: Pereptron
  - Euclidean distance: OLS
- Stochastic:
  - Frequency: Logsitc, OLS($\leftrightarrow$ MLE if data follow unbiased Gaussian distribution)?
  - Bayesian: Bayesian regression.

Relation

Bayesian regression $\leftrightarrow$ OLS + random noise
prior $\leftrightarrow$ regularization