Machine Learning Working Notes

Huang Xiao XIAOHU@IN.TUM.DE
Technische Universität München, Boltzmannstr.3, D-85743 Garching b. München, Germany

Abstract

Machine learning is a fast pacing discipline in many working fields, especially it is now regarded as the most impacting subject in artificial intelligence. In this working notes, I summarize some important notes during my study of machine learning. For the completeness, references are included for readers who are reading this article. Note that this working note is only distributed and shared with author's acknowledge and confirmation. It is not intended as a publishable research paper or tutorial.

1. Gaussian Process

1.1. Regression

Gaussian process is an important nonparametric regression model which looks for an optimal functional in a space of functions, that minimizes a loss function, although the loss function needs not to be explicitly defined. (Rasmussen, 2006)

Give a training dataset $\mathcal{D} = \{x_i\}_{i=1}^n$ with $x_i \in R^d$, *i.i.d* drawn from certain distribution, we are interested at the predictive distribution of unknown target for the test sample x_*, denoted as f_*. Suppose a prior over $\boldsymbol{y}$ given input $\boldsymbol{X}$ is a n-variable Gaussian distribution,

$$\boldsymbol{y} \sim \mathcal{N}(\boldsymbol{0}, K(\boldsymbol{X}, \boldsymbol{X}))$$

where $K(\boldsymbol{X}, \boldsymbol{X})$ defines a covariance function over $\boldsymbol{X}$. Therefore the posterior of f_* given the training dataset $\mathcal{D}$ is also a Gaussian.

$$f_*|\boldsymbol{y}, \boldsymbol{X}, x_* \sim \mathcal{N}(\mu_*, \Sigma_*^{-1})$$

where the sufficient statistics can be derived using Bayesian theorem,

$$\mu_* = K_{X_*,X}\left[K_{X,X} + \sigma_n^2 I\right]^{-1}\boldsymbol{y} \quad (1)$$

$$\Sigma_*^{-1} = K_{X_*,X_*} - K_{X_*,X}\left[K_{X,X} + \sigma_n^2 I\right]^{-1} K_{X,X_*} \quad (2)$$

where σ_n^2 is the noise level and $K_{\cdot,\cdot}$ represents a shorthand for covariance matrix. Q: Here comes the question of how to estimate the parameters for the covariance K.

To obtain the Eq.(1)-(2), we can use the following trick. Given two variables $(\boldsymbol{x}, \boldsymbol{y})$ following a Gaussian distribution,

$$\begin{bmatrix}\boldsymbol{x}\\ \boldsymbol{y}\end{bmatrix} \sim \mathcal{N}\left(\begin{bmatrix}\mu_x\\ \mu_y\end{bmatrix}, \begin{bmatrix}A & C\\ C^T & B\end{bmatrix}\right)$$

Then we have the conditional distribution,

$$\boldsymbol{x}|\boldsymbol{y} \sim \mathcal{N}\left(\mu_x + CB^{-1}(\boldsymbol{y} - \mu_y), A - C^T B^{-1} C\right)$$

1.2. Classification

Using Gaussian process for classification task is a bit more complicated than regression. The main idea of it is to use a 'squash' function to convert a predicted function value within $[0, 1]$, *e.g.*, sigmoid function, cumulative Gaussian *pdf*

2. PCA

The basic idea of PCA is to maximally reduce information loss of projecting high dimensional data to lower dimension. Therefore, an intuitive consideration would be that we introduce first of all a projection matrix $\boldsymbol{u}$ on d-dimensional instance x, so that x is mapped on a lower m-dimensional space. Following column vector routine, we expect that matrix $\boldsymbol{u}$ as being $m \times d$. Now given a dataset $\boldsymbol{X} = \{x_i\}_{i=1}^n$, it will be projected on a m-dimensional space by $\boldsymbol{u}$. The objective of the projection is to maximize the covariance of data on the lower dimensional space, that is,

$$\max \frac{1}{n}\sum_{i=1}^{n} \|\boldsymbol{u}x - \boldsymbol{u}\bar{x}\|^2$$

that can be rewritten as,

$$\underset{\boldsymbol{u}}{\textit{maximize}} \quad \boldsymbol{u}S\boldsymbol{u}^T$$
$$\text{s.t.} \quad \boldsymbol{u}_i\boldsymbol{u}_i^T = 1,\ i = 1, \ldots, m \quad (3)$$

where S is the covariance of $\boldsymbol{X}$. To prevent $\boldsymbol{u}$ goes to infinity, we assume $\boldsymbol{u}$ has unit length, namely, $\boldsymbol{u}$ represents a set of basis of the lower dimension.

According to (3), we introduce m Lagrangian multipliers $\boldsymbol{\lambda}$ as a diagonal matrix, and we have

$$L = \underset{\boldsymbol{u}}{maximize} \quad \boldsymbol{u}S\boldsymbol{u}^T - \boldsymbol{\lambda}\boldsymbol{u}\boldsymbol{u}^T$$

Take the derivative of L with respect to $\boldsymbol{u}$, we have

$$\begin{aligned} \boldsymbol{u}S &= \boldsymbol{\lambda}\boldsymbol{u} \\ S &= \boldsymbol{u}^{-1}\lambda\boldsymbol{u} \end{aligned} \tag{4}$$

Since $\boldsymbol{u}$ is orthogonal, it is therefore not singular. We see that $\boldsymbol{u}$ and $\boldsymbol{l}ambda$ are the indeed the eigenvectors and eigenvalues for S respectively. And Eq.(4) is exactly the singular value decomposition of $S = U\Sigma V^T$, we can derive $\boldsymbol{u}$ and $\boldsymbol{\lambda}$ from S conveniently.

References

Rasmussen, Carl Edward. Gaussian processes for machine learning. MIT Press, 2006.