Introduction to Big Data Analysis
Classification : Part 2

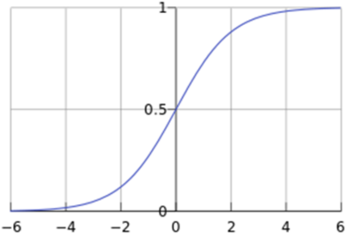
Zhen Zhang

Southern University of Science and Technology

Outlines

Logistic Regression

- Not regression, but a classification method
- Connection with linear regression : $y = w_0 + w_1 x + \epsilon$, y is binary (0 or 1) ; then $E(y|x) = P(y=1|x) = w_0 + w_1 x$; but $w_0 + w_1 x$ may not be a probability
- Find a function to map it back to $[0, 1]$: Sigmoid function $g(z) = \frac{1}{1+e^{-z}}$ with $z = w_0 + w_1 x_1 + \ldots + w_d x_d$



- Equivalently, $\log \frac{P(y=1|x)}{1-P(y=1|x)} = w_0 + w_1 x_1 + \ldots + w_d x_d$, logit transform $logit(z) = \log \frac{z}{1-z}$

MLE for Logistic Regression

- The prob. distribution for two-class logistic regression model is

$$Pr(y = 1|\mathbf{X} = \mathbf{x}) = \frac{\exp(\mathbf{w}^T\mathbf{x})}{1 + \exp(\mathbf{w}^T\mathbf{x})},$$

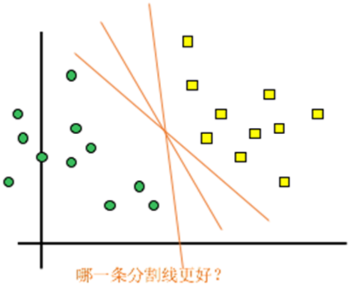
$$Pr(y = 0|\mathbf{X} = \mathbf{x}) = \frac{1}{1 + \exp(\mathbf{w}^T\mathbf{x})}.$$

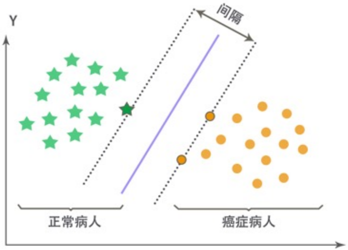
- Let $P(y = k|\mathbf{X} = \mathbf{x}) = p_k(\mathbf{x}; \mathbf{w})$, $k = 0$ or 1. The likelihood function is defined by $L(\mathbf{w}) = \prod_{i=1}^{n} p_{y_i}(\mathbf{x}_i; \mathbf{w})$
- MLE estimate of $\mathbf{w}$: $\hat{\mathbf{w}} = \arg\max_{\mathbf{w}} L(\mathbf{w})$
- Solve $\nabla_{\mathbf{w}} \log L(\mathbf{w}) = 0$ by Newton-Raphson method

Outlines

Support Vector Machine (SVM)

- Use hyperplane to separate data : maximize margin
- Can deal with low-dimensional data that are not linearly separated by using kernel functions
- Decision boundary only depends on some samples (support vectors)





Linear SVM

- Training data : $\{(\mathbf{x}_1, y_1), \ldots, (\mathbf{x}_n, y_n)\}$, $y_i \in \{-1, +1\}$
- Hyperplane : $S = \mathbf{w}^T\mathbf{x} + b$; decision function : $f(\mathbf{x}) = \mathrm{sign}(\mathbf{w}^T\mathbf{x} + b)$

$$\left.\begin{array}{l} f(\mathbf{x}_i) > 0 \Leftrightarrow y_i = 1 \\ f(\mathbf{x}_i) < 0 \Leftrightarrow y_i = -1 \end{array}\right\} \Rightarrow y_i f(\mathbf{x}_i) > 0$$

- Geometric margin between a point and hyperplane : $r_i = \frac{y_i(\mathbf{w}^T\mathbf{x}_i + b)}{\|\mathbf{w}\|_2}$
- Margin between dataset and hyperplane : $\min\limits_i r_i$
- Maximize margin : $\max\limits_{\mathbf{w}, b} \min\limits_i \frac{y_i(\mathbf{w}^T\mathbf{x}_i + b)}{\|\mathbf{w}\|_2}$

Formulation as Constrained Optimization

- Without loss of generality, let $\min_i y_i(\mathbf{w}^T\mathbf{x}_i + b) = 1$ (multiply $\mathbf{w}$ and b by the same proper constant)
- Maximize margin is equivalent to

$$\max_{\mathbf{w},b} \frac{1}{\|\mathbf{w}\|_2}, \quad \text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geqslant 1, i = 1, \ldots, n$$

- Further reduce to

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2, \quad \text{s.t. } y_i(\mathbf{w}^T\mathbf{x}_i + b) \geqslant 1, i = 1, \ldots, n$$

- This is primal problem : quadratical programming with linear constraints, computational complexity is $O(p^3)$ where p is dimension

Method of Lagrange Multipliers

- Introduce $\alpha_i \geqslant 0$ as Lagrange multiplier of constraint $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geqslant 1$
- Lagrange function :
  $L(\mathbf{w}, b, \alpha) = \frac{1}{2}\|\mathbf{w}\|_2^2 - \sum_{i=1}^{n} \alpha_i[y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1]$
- Since

$$\max_{\alpha} L(\mathbf{w}, b, \alpha) = \begin{cases} \dfrac{1}{2}\|\mathbf{w}\|_2^2, & y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 \geqslant 0, \forall i \\ +\infty, & y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 < 0, \exists i \end{cases}$$

- Primal problem is equivalent to the minimax problem

$$\min_{\mathbf{w},b} \max_{\alpha} L(\mathbf{w}, b, \alpha)$$

Dual problem

- When slater condition is satisfied, $\min\max \Leftrightarrow \max\min$
- Dual problem : $\max_{\alpha} \min_{\mathbf{w},b} L(\mathbf{w}, b, \alpha)$
- Solve for inner minimization problem :

$$\nabla_{\mathbf{w}} L = 0 \Longrightarrow \mathbf{w}^* = \sum_i \alpha_i y_i \mathbf{x}_i$$

$$\frac{\partial L}{\partial b} = 0 \Longrightarrow \sum_i \alpha_i y_i = 0$$

- Plug into L : $L(\mathbf{w}^*, b^*, \alpha) = \sum_i \alpha_i - \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_i y_i y_j (\mathbf{x}_i^T \mathbf{x}_j)$
- Dual optimization :

$$\min_{\alpha} \frac{1}{2} \sum_i \sum_j \alpha_i \alpha_i y_i y_j (\mathbf{x}_i^T \mathbf{x}_j) - \sum_i \alpha_i,$$

$$\text{s.t. } \alpha_i \geqslant 0, i = 1, \ldots, n, \sum_i \alpha_i y_i = 0$$

KKT conditions

- Three more conditions from the equivalence of primal and minimax problems

$$\begin{cases} \alpha_i^* \geqslant 0, \\ y_i((\mathbf{w}^*)^T \mathbf{x}_i + b^*) - 1 \geqslant 0, \\ \alpha_i^* [y_i((\mathbf{w}^*)^T \mathbf{x}_i + b^*) - 1] = 0. \end{cases}$$

- These together with two zero derivative conditions form KKT conditions
- $\alpha_i > 0 \Rightarrow y_i(\mathbf{w}^T \mathbf{x}_i + b^*) = 1$
- Index set of support vectors $S = \{i | \alpha_i > 0\}$
- $b = y_s - \mathbf{w}^T \mathbf{x}_s = y_s - \sum_{i \in S} \alpha_i y_i \mathbf{x}_i^T \mathbf{x}_s$
- More stable solution : $b = \frac{1}{|S|} \sum_{s \in S} \left(y_s - \sum_{i \in S} \alpha_i y_i \mathbf{x}_i^T \mathbf{x}_s \right)$

Sequential Minimal Optimization (SMO) Algorithm

- Invented by John C. Platt (1998)
- Coordinately optimize dual problem, select two variables and fix others, then dual problem reduces to one variable quadratic programming with positivity constraint
  1. Initially, choose α_i and α_j
  2. Fix other variables, solve for α_i and α_j
  3. Update α_i and α_j, redo step 1 iteratively
  4. Stop until convergence
- How to choose α_i and α_j? choose the pair far from KKT conditions the most
- Computational complexity $O(n^3)$
- Easy to generalize to high dimensional problem with kernel functions

Soft Margin

- When data are not linear separable, introduce slack variables (tolerance control of fault) $\xi_i \geqslant 0$
- Relax constraint to $y_i(\mathbf{w}^T\mathbf{x}_i + b) \geqslant 1 - \xi_i$
- Primal problem :

$$\min_{\mathbf{w},b} \frac{1}{2}\|\mathbf{w}\|_2^2 + C\sum_{i=1}^{n}\xi_i$$

$$\text{s.t.} \quad y_i(\mathbf{w}^T\mathbf{x}_i + b) \geqslant 1 - \xi_i, \xi_i \geqslant 0, i = 1, \dots, n$$

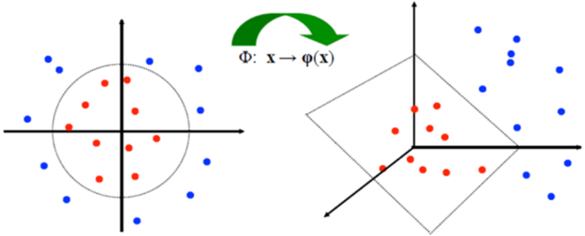
- Similar derivation to dual problem :

$$\min_{\alpha} \frac{1}{2}\sum_{i}\sum_{j}\alpha_i\alpha_i y_i y_j(\mathbf{x}_i^T\mathbf{x}_j) - \sum_{i}\alpha_i,$$

$$\text{s.t.} \quad 0 \leqslant \alpha_i \leqslant C, i = 1, \dots, n, \sum_{i}\alpha_i y_i = 0$$

Nonlinear SVM

- Nonlinear decision boundary could be mapped to linear boundary in high-dimensional space
- Modify objective function in dual problem : $\frac{1}{2}\sum_i \sum_j \alpha_i \alpha_i y_i y_j (\phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)) - \sum_i \alpha_i$
- Kernel function as inner product : $K(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)^T \phi(\mathbf{x}_j)$



Kernel Methods

- Reduce effect of curse of dimensionality
- Different kernels lead to different decision boundaries
- Popular kernels :

Kernel	Definition	Parameters
Polynomial	$(\mathbf{x}_1^T\mathbf{x}_2+1)^d$	d is positive integer
Gaussian	$e^{-\frac{\lVert\mathbf{x}_1-\mathbf{x}_2\rVert^2}{2\delta^2}}$	$\delta>0$
Laplacian	$e^{-\frac{\lVert\mathbf{x}_1-\mathbf{x}_2\rVert}{\delta^2}}$	$\delta>0$
Fisher	$\tanh(\beta\mathbf{x}_1^T\mathbf{x}_2+\theta)$	$\beta>0,\theta<0$

Pros and Cons

- Where it is good
  - Applications in pattern recognition : text classification, face recognition
  - Easy to deal with high-dimensional data with kernels
  - Robust (only depends on support vectors), and easy to generalize to new dataset
- Disadvantage
  - Poor for ultra high dimensional data
  - Low computational efficiency for nonlinear SVM when sample size is large
  - Poor interpretability without probability

Outlines

Linear Discriminant Analysis (LDA)

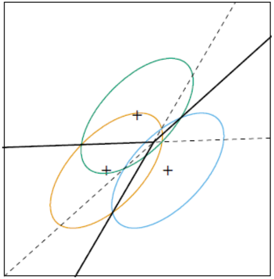
- Bayes Classifier amounts to know the class posteriors $\mathrm{P}(Y|\mathbf{X})$ for optimal classification : $k^* = \arg\max_k \mathrm{P}(Y = k|\mathbf{X})$
- Let $\pi_k = \mathrm{P}(Y = k)$ be the prior probability, $f_k(\mathbf{x}) = \mathrm{P}(\mathbf{X} = \mathbf{x}|Y = k)$ be the density function of samples in each class $Y = k$
- By Bayes theorem, $\mathrm{P}(Y|\mathbf{X} = \mathbf{x}) \propto f_k(\mathbf{x})\pi_k$ (Recall naive Bayes)
- Assume $f_k(x)$ is multivariate Gaussian : $f_k(\mathbf{x}) = \frac{1}{(2\pi)^{p/2}|\Sigma_k|^{1/2}} e^{-\frac{1}{2}(\mathbf{x}-\mu_k)^T \mathbf{\Sigma}_k^{-1}(\mathbf{x}-\mu_k)}$, with a common covariance matrix $\mathbf{\Sigma}_k = \mathbf{\Sigma}$, sufficient to look at the log-ratio

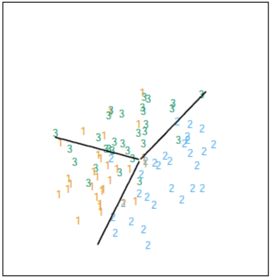
$$\log \frac{\mathrm{P}(Y = k|\mathbf{X} = \mathbf{x})}{\mathrm{P}(Y = l|\mathbf{X} = \mathbf{x})} = \log \frac{\pi_k}{\pi_l} - \frac{1}{2}(\mu_k + \mu_l)^T \mathbf{\Sigma}^{-1}(\mu_k - \mu_l) + \mathbf{x}^T \mathbf{\Sigma}^{-1}(\mu_k - \mu_l)$$

for the decision boundary between class k and l

Discriminant Rule

- Linear discriminant functions :
  $\delta_k(\mathbf{x}) = \mathbf{x}^T \mathbf{\Sigma}^{-1} \mu_k - \frac{1}{2} \mu_k^T \mathbf{\Sigma}^{-1} \mu_k + \log \pi_k$
- Decision rule : $k^* = \arg\max_k \delta_k(\mathbf{x})$
- Sample estimate of unknowns : $\hat{\pi}_k = N_k / N$, where
  $N = \sum_{k=1}^K N_k$, $\hat{\mu}_k = \frac{1}{N_k} \sum_{y_i=k} \mathbf{x}_i$,
  $\hat{\Sigma} = \frac{1}{N-K} \sum_{k=1}^K \sum_{y_i=k} (\mathbf{x}_i - \hat{\mu}_k)(\mathbf{x}_i - \hat{\mu}_k)^T$



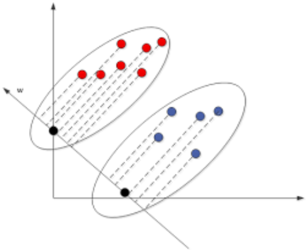


Two-class LDA

- LDA rule classifies to class 2 if

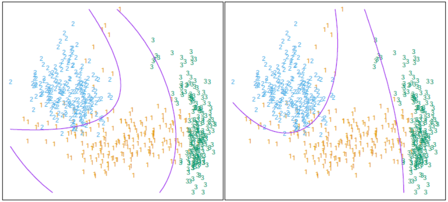
$$(\mathbf{x} - \frac{\hat{\mu}_1 + \hat{\mu}_2}{2})^T \mathbf{\Sigma}^{-1} (\hat{\mu}_2 - \hat{\mu}_1) + \log \frac{\hat{\pi}_2}{\hat{\pi}_1} > 0$$

- Discriminant direction : $\beta = \mathbf{\Sigma}^{-1}(\hat{\mu}_2 - \hat{\mu}_1)$
- Bayes misclassfication rate $= 1 - \Phi(\beta^T(\mu_2 - \mu_1)/(\beta^T \mathbf{\Sigma} \beta)^{\frac{1}{2}})$, where $\Phi(x)$ is the Gaussian distribution function



Other Variants

- Quadratic discriminant analysis (QDA) :
  $\delta_k(\mathbf{x}) = -\frac{1}{2}\log|\mathbf{\Sigma}_k| - \frac{1}{2}(\mathbf{x}-\mu_k)^T\mathbf{\Sigma}_k^{-1}(\mathbf{x}-\mu_k) + \log\pi_k$
- Regularized discriminant analysis : $\hat{\mathbf{\Sigma}}_k(\alpha) = \alpha\hat{\mathbf{\Sigma}}_k + (1-\alpha)\hat{\mathbf{\Sigma}}$
- Computations for LDA :
  1. Sphere the data with respect to $\hat{\mathbf{\Sigma}} = \mathbf{UDU}^T$: $\mathbf{X}^* = \mathbf{D}^{-\frac{1}{2}}\mathbf{U}^T\mathbf{X}$. Then the common covariance estimate of $\mathbf{X}^*$ is $\mathbf{I}_p$
  2. Classsify to the closest class centroid in the transformed space, taking into account of the class prior probabilities π_k's
- Reduced-Rank LDA : see dimensionality reduction



Outlines

References

- 数据分析导论，博雅大数据学院
- 周志华，机器学习，2016
- T. Hastie, R. Tibshirani, and J. Friedman, The Elements of Statistical Learning : Data mining, Inference, and Prediction, 2nd Edition, 2009