Course	: COMP6577 – Machine Learning
Effective Period	: December 2019

The Nearest Neighbor rule (KNN) & Logistic Regression

Session 17 & 18

People
Innovation
Excellence

Learning Outcome

- LO3: Student be able to experiment classification and clustering algorithm from given dataset

Outline

- Bayesian Vs K-NN
- The Nearest Neighbor rule
- Example of KNN
- Logistic regression
- Multiclass logistic regression
- Case Study

Bayesian Vs K-NN

- Bayesian rule provides the optimal solution with respect to the classification error probability, but its application requires the estimation of the respective conditional pdfs; which is not an easy task, once the dimensionality of the feature space assumes relatively large values.
- The k-nearest neighbor (k-NN) rule is a typical nonparametric classifier and it is one among the most popular and well-known classifiers.
- In spite of its simplicity, it is still in use and stands next to more elaborate schemes.

The Nearest Neighbor Rule

- Consider N training points, (y_n,x_n), n=1,2,...,*N*, for an *M*-class classification task. At the heart of the method lies a parameter *k*, which is a user-defined parameter. Once *k* is selected, then given a pattern, *x*, assign it to the class in which the majority of its *k* nearest (according to a metric, e.g., Euclidean or Mahalanobis distance) neighbors, among the training points, belong.
- The parameter *k* should not be a multiple of *M*, in order to avoid ties. The simplest form of this rule is to assign the pattern to the class in which its nearest neighbor belongs, meaning *k*=1.

The Nearest Neighbor Rule (2)

- It turns out that this conceptually simple rule tends to the Bayesian classifier if (a) $N \to \infty$, (b)$k \to \infty$, and (c) $k/N \to 0$.
- More specifically, it can be shown that the classification errors P_{NN} and P_{kNN} satisfy, asymptotically, the following bounds:

$P_B \le P_{NN} \le 2P_B$ For the k=1 NN rule and,

$P_B \le P_{kNN} \le P_B + \sqrt{\frac{2PNN}{k}}$ for the more general k-NN version

- P_B is the error corresponding to the optimal Bayesian classifier.

- By those two formulae, if one has an easy task (as indicated by the very low value of PB), the NN rule can also do a good job. This, of course, is not the case if the problem is not an easy one and larger error values are involved.
- The bound in the second formula says that for large values of k (provided, of course, N is large enough), the performance of the k-NN tends to that of the optimal classifier. In practice, one has to make sure that k does not get values close to N, but remains a relatively small fraction of it.

- The Bayesian classifier exploits the statistical information for the data distribution while the k-NN does not take into account such information.
- The reason is that if N is a very large value (hence the space is densely populated) and *k* is a relatively small number, with respect to *N*, then the nearest neighbors will be located very close to *x*.
- Due to the **continuity** of the involved pdfs, the values of their posterior probabilities will be close to $P(\omega i|x)$, $i = 1, 2, \ldots, M$.
- For large enough *k*, the majority of the neighbors must come from the class that scores the maximum value of the posterior probability given *x.*

Drawback

- Every time a new pattern is considered, its distance from all the training points has to be computed, then selecting the *k* closest to it points.

Remarks

- The use of the k-nearest rule concept can also be adopted in the context of the regression task.
- Given an observation, *x,* one searches for its k closer input vectors in the training set, denoted as $x_{(1)}$, ... , $x_{(k),}$ and computes an estimate of the output value, $\hat{y}$, as an average of the respective outputs in the training set, represented by

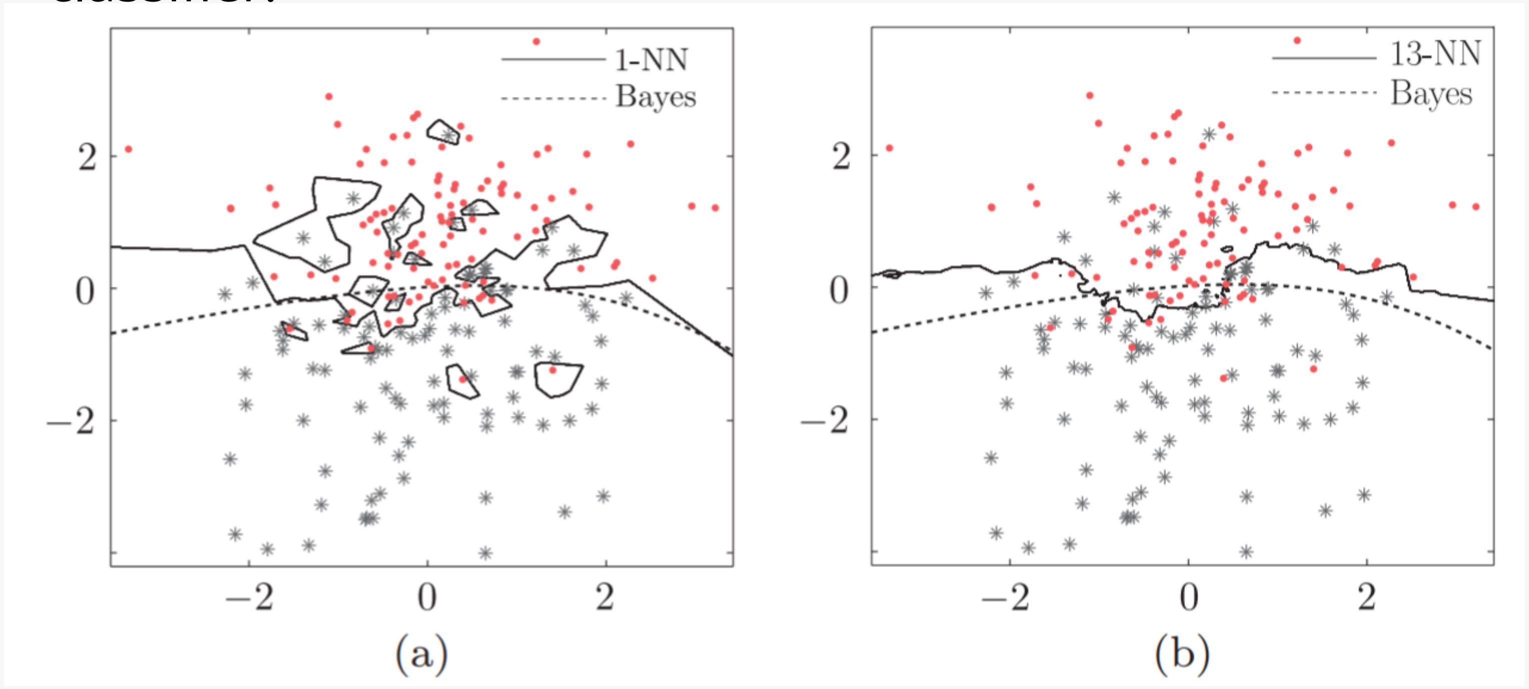
$$\hat{y} = \frac{1}{k} \sum_{i=1}^{k} y_{(i)}$$

People
Innovation
Excellence

People
Innovation
Excellence

Example

- Given an example that illustrates the decision curves for a two-class classification task in the two-dimensional space, obtained by the Bayesian, the 1-NN and the 13-NN classifier.



- A number of N = 100 data are generated for each class by Gaussian distributions. The decision curve of the Bayes classifier has the form of a parabola, while the 1-NN classifier exhibits a highly nonlinear nature. The 13-NN rule forms a decision line close to the Bayesian one.
- The figure shows a two-class classification task. The dotted curve corresponds to the optimal Bayesian classifier. The full line curves correspond to (a) the 1-NN and (b) the 13-NN classifiers. Observe that the 13-NN is closer to the Bayesian one.

Logistic Regression

- In Bayesian classification, the assignment of a pattern in a class is performed based on the posterior probabilities, P(ωi|x). The posteriors are estimated via the respective conditional pdfs, which is not, in general, an easy task.
- The goal in this session is to model the posterior probabilities directly, via the logistic regression method.
- This name has been established in the statistics community, although the model refers to classification and not to regression.
- This is a typical example of the discriminative modeling approach, where the distribution of data is of no interest.

People
Innovation
Excellence

Logistic Regression

- The two-class case: The starting point is to model the ratio of the posteriors as

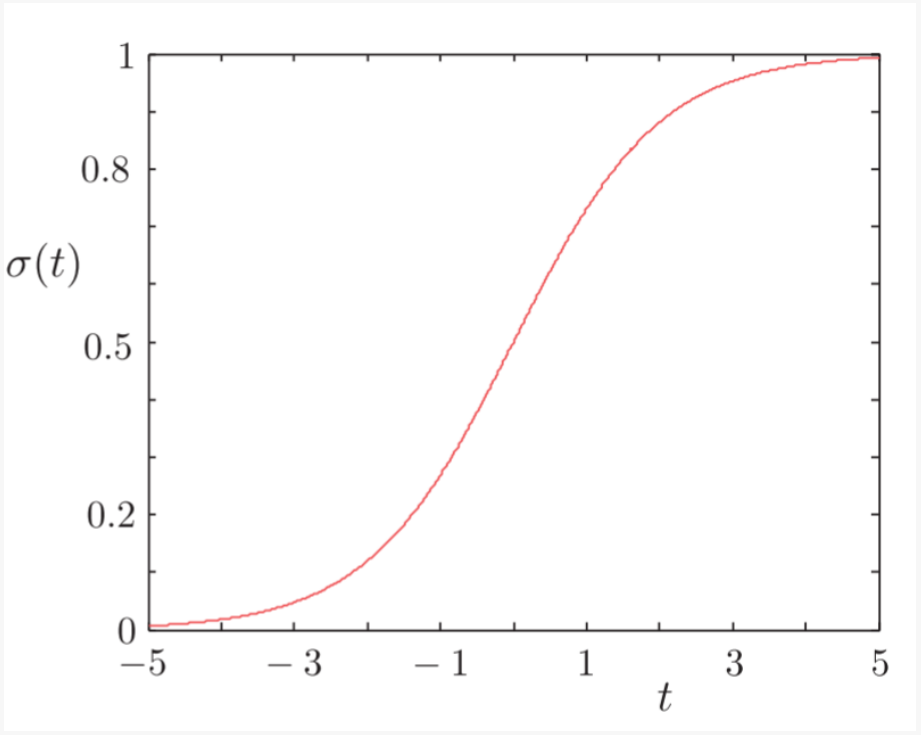
$$\boxed{\ln \frac{P(\omega_1|\boldsymbol{x})}{P(\omega_2|\boldsymbol{x})} = \boldsymbol{\theta}^{\mathrm{T}}\boldsymbol{x} : \quad \text{Two-class Logistic Regression,}}$$

- where the constant term, θ_0, has been absorbed in θ. Taking into account that *P(ω1|x)* + *P(ω2|x)* = 1, and defining t := $\boldsymbol{\theta}^T\mathbf{x}$**,** it is readily seen that the model is equivalent to

$$P(\omega_1|\boldsymbol{x}) = \sigma(t)$$
$$\sigma(t) := \frac{1}{1+\exp(-t)},$$

$$P(\omega_2|\boldsymbol{x}) = 1 - P(\omega_1|\boldsymbol{x}) = \frac{\exp(-t)}{1+\exp(-t)}$$

Sigmoid Link Function



- The function σ (t) is known as the logistic sigmoid or sigmoid link function and it is shown in the figure.

BINUS UNIVERSITY

People Innovation Excellence

Logistic Regression

- Assuming the data in the classes follow Gaussian distributions with $\Sigma_1 = \Sigma_2 \equiv \Sigma$ and for simplicity that $P(\omega_1) = P(\omega_2)$, the latter of the previously stated equations is written as

$$\ln \frac{P(\omega_1|\boldsymbol{x})}{P(\omega_2|\boldsymbol{x})} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)^{\mathrm{T}} \Sigma^{-1} \boldsymbol{x} + \text{constants}$$

- In other words, when the distributions underlying the data are Gaussians with a common covariance matrix, then the log ratio of the posteriors is a linear function. Thus, in logistic regression, all we do is adopt such a model, irrespective of the data distribution.
- Moreover, even if the data are distributed according to Gaussians, it may still be preferable to adopt the logistic regression formulation instead of that in the equation above.

- In the latter formulation, the covariance matrix has to be estimated, amounting to $O(l^2/2)$ parameters. The logistic regression formulation only involves $l + 1$ parameters.
- That is, once we know about the linear dependence of the log ratio on $\boldsymbol{x}$, we can use this a priori information to simplify the model.
- Assuming that the Gaussian assumption is valid, if one can obtain good estimates of the covariance matrix, employing this extra information can lead to more efficient estimates, in the sense of lower variance.
- In practice, it turns out that using the logistic regression is, in general, a safer bet compared to the linear discriminant analysis (LDA).

- The parameter vector, θ, is estimated via the ML method applied on the set of training samples, (y_n, x_n), $n = 1, 2, \ldots, N$, $y_n \in \{0, 1\}$. The likelihood function can be written as

$$P(y_1, \ldots, y_N; \boldsymbol{\theta}) = \prod_{n=1}^{N} \left(\sigma(\boldsymbol{\theta}^T \boldsymbol{x}_n)\right)^{y_n} \left(1 - \sigma(\boldsymbol{\theta}^T \boldsymbol{x}_n)\right)^{1-y_n}$$

- Usually, we consider the negative log-likelihood given by

$$L(\boldsymbol{\theta}) = -\sum_{n=1}^{N} \left(y_n \ln s_n + (1 - y_n) \ln(1 - s_n)\right)$$

Also known as cross-entropy error

where $s_n := \sigma(\theta^T x_n)$.

People
Innovation
Excellence

- Minimization of L(θ) with respect to θ is carried out iteratively by any iterative minimization scheme, such as the steepest descent or Newton's method. Both schemes need the computation of the respective gradient, which in turn is based on the derivative of the sigmoid link function:

$$\frac{d\sigma(t)}{dt} = \sigma(t)\big(1 - \sigma(t)\big)$$

- The gradient is given by

$$\nabla L(\boldsymbol{\theta}) = \sum_{n=1}^{N} (s_n - y_n)\boldsymbol{x}_n = X^{\mathrm{T}}(\boldsymbol{s} - \boldsymbol{y}),$$

where $X^{\mathrm{T}} = [\boldsymbol{x}_1, \ldots, \boldsymbol{x}_N], \quad \boldsymbol{s} := [s_1, \ldots, s_N]^{\mathrm{T}}, \quad \boldsymbol{y} = [y_1, \ldots, y_N]^{\mathrm{T}}.$

- The Hessian matrix is given by

$$\nabla^2 L(\boldsymbol{\theta}) = \sum_{n=1}^{N} s_n(1 - s_n)\boldsymbol{x}_n\boldsymbol{x}_n^{\mathrm{T}} = X^{\mathrm{T}}RX,$$

where $R := \operatorname{diag}\left\{s_1(1 - s_1), \ldots, s_N(1 - s_N)\right\}$

- Note that because $0 < s_n < 1$, by definition of the sigmoid link function, matrix R is positive definite; hence, the Hessian matrix is also positive definite.
- The negative log-likelihood function is convex, which guarantees the existence of a unique minimum.

- Two of the possible iterative minimization schemes to be used are:
  - Steepest descent $$\boldsymbol{\theta}^{(i)} = \boldsymbol{\theta}^{(i-1)} - \mu_i X^{\mathrm{T}}(\boldsymbol{s}^{(i-1)} - \boldsymbol{y}).$$
  - Newton's scheme

$$\begin{aligned}\boldsymbol{\theta}^{(i)} &= \boldsymbol{\theta}^{(i-1)} - \mu_i \left(X^{\mathrm{T}}R^{(i-1)}X\right)^{-1} X^{\mathrm{T}}(\boldsymbol{s}^{(i-1)} - \boldsymbol{y}) \\ &= \left(X^{\mathrm{T}}R^{(i-1)}X\right)^{-1} X^{\mathrm{T}}R^{(i-1)}\boldsymbol{z}^{(i-1)}\end{aligned}$$

where $\boldsymbol{z}^{(i-1)} := X\boldsymbol{\theta}^{(i-1)} - \left(R^{(i-1)}\right)^{-1}(\boldsymbol{s}^{(i-1)} - \boldsymbol{y})$

the involved quantities are iteration-dependent and the resulting scheme is known as **iterative reweighted least squares scheme (IRLS)**

- Maximizing the likelihood may run into problems if the training data set is linearly separable. In this case, any point on a hyperplane, $\theta^T x = 0$, that solves the classification task and separates the samples from each class (note that there are infinite many such hyperplanes), results in $\sigma(x) = 0.5$, and every training point from each class is assigned a posterior probability equal to one.
- Thus, ML forces the logistic sigmoid to become a step function in the feature space and equivalently $||\theta|| \to \infty$. This can lead to overfitting and it is remedied by including a regularization term, $||\theta||^2$, in the respective cost function.

Multiclass Logistic Regression

- The *M*-class case: For the more general *M*-class classification task, the logistic regression is defined for *m* = 1, 2, …, *M*, as

$$\boxed{P(\omega_m|\boldsymbol{x}) = \frac{\exp(\boldsymbol{\theta}_m^{\mathrm{T}}\boldsymbol{x})}{\sum_{j=1}^{M}\exp(\boldsymbol{\theta}_j^{\mathrm{T}}\boldsymbol{x})} : \quad \text{Multiclass Logistic Regression.}}$$

- The previous definition is easily brought into the form of a linear model for the log ratio of the posteriors. Divide, for example, by P(ω_M|x) to obtain

$$\ln\frac{P(\omega_m|\boldsymbol{x})}{P(\omega_M|\boldsymbol{x})} = (\boldsymbol{\theta}_m - \boldsymbol{\theta}_M)^T\boldsymbol{x} = \hat{\boldsymbol{\theta}}_m^T\boldsymbol{x}.$$

- Let us define, for notational convenience,

$$\phi_{nm} := P(\omega_m | \boldsymbol{x}_n), \quad n = 1, 2, \ldots, N,\ m = 1, 2, \ldots, M,$$

And $$t_m := \boldsymbol{\theta}_m^{\mathrm{T}} \boldsymbol{x}, \quad m = 1, 2, \ldots, M.$$

- The likelihood function is now written as

$$P(\boldsymbol{y}; \boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M) = \prod_{n=1}^{N} \prod_{m=1}^{M} (\phi_{nm})^{y_{nm}}$$

where $y_{nm} = 1$ if $x_n \in \omega_m$ and zero otherwise

- The respective negative log-likelihood function becomes

$$L(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M) = -\sum_{n=1}^{N}\sum_{m=1}^{M} y_{nm} \ln \phi_{nm}$$

- Minimization with respect to θ_m, *m* = 1, … , *M*, takes place iteratively.
- The following gradients are used

$$\frac{\partial \phi_{nm}}{\partial t_j} = \phi_{nm}(\delta_{mj} - \phi_{nj}),$$

where δ_{mj} is one if *m* = *j* and zero otherwise. Also

$$\nabla_{\boldsymbol{\theta}_j} L(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M) = \sum_{n=1}^{N} (\phi_{nj} - y_{nj})\boldsymbol{x}_n.$$

- The respective Hessian matrix is an *(lM)* × *(lM)* matrix, comprising *l* × *l* blocks. Its *k*, *j* block element is given by

$$\nabla_{\boldsymbol{\theta}_k} \nabla_{\boldsymbol{\theta}_j} L(\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M) = \sum_{n=1}^{N} \phi_{nj}(\delta_{kj} - \phi_{nk}) \boldsymbol{x}_n \boldsymbol{x}_n^T.$$

- The Hessian matrix is also positive definite, which guarantees uniqueness of the minimum as in the two-class case.

People
Innovation
Excellence

Probit regression

- Instead of using the logistic sigmoid function in P(ω1|x) = σ (t) (for the two-class case), other functions can also be adopted. A popular function in the statistical community is the **probit function**, which is defined as:

$$\Phi(t) := \int_{-\infty}^{t} \mathcal{N}(z|0,1)dz$$
$$= \frac{1}{2}\left(1 + \frac{1}{\sqrt{2}}\text{erf}(t)\right),$$

- where erf is the error function defined as

$$\text{erf}(t) = \frac{2}{\sqrt{\pi}} \int_{0}^{t} \exp\left(-\frac{z^2}{2}\right) dz.$$

- In other words, $P(\omega_1|t)$ is modeled to be equal to the probability of a normalized Gaussian variable to lie in the interval (−∞, t]. The graph of the probit function is very similar to that of the logistic one.

Case Study

Given data of Singapore Airbnb which can be downloaded in this link

https://www.kaggle.com/jojoker/singapore-airbnb

1. From the downloaded data, try to apply K-NN to classify the data. Discuss the result.
2. Apply Logistic regression algorithm to do classification for the Singapore Airbnb data above.

End of Session 17 & 18

People
Innovation
Excellence

References

- Sergios Theodoridis. (2015). *Machine Learning: a Bayesian and Optimization Perspective*. Jonathan Simpson. ISBN: 978-0-12-801522-3. Chapter 7.
- https://www.kaggle.com/jojoker/singapore-airbnb