Nonprobalistic or non parametric classifiers

Machine Learning

Dr. Neeta Nain

Department of Computer Science and Engineering
Malviya National Institute of Technology, Jaipur

2020

Outline

k - Nearest Neighbor

- k-NN is a popular classification technique owing to its simple, and intuitively appealing, specification
- Classify a measurement x to one of C classes as
  1. determine the k nearest training data vectors to the measurement x, using an appropriate distance metric
  2. assign x to the class with the most representatives (votes) within the set of k nearest vectors
- k is selected as a trade-off between choosing a value large enough to reduce the sensitivity to noise, and choosing a value small enough that the neighborhood does not extend to the domain of other classes

Nearest Neighbor

Single *NN* by passes the problem of probability densities completely and classifies an unknown sample belonging to the same class as the most similar or "nearest" sample point in the training set.

Nearest

Nearest can mean the smallest Euclidean distance (usual distance between two points) in n-dimensional feature space, $a = (a_1, a_2, \cdots, a_n)$ and $b = (b_1, b_2, \cdots, b_n)$, defined by $d_e(a, b) = \sqrt{\sum_{i=1}^{n}(b_i - a_i)^2}$

Square root can be avoided as the smallest squared distance to the sample being classified also has the smallest distance to sample.

Manhattan / City block / Taxi cab distance

Sum of absolute differences in each measure as $d_{cb}(a, b) = \sum_{i=1}^{n} |b_i - a_i|$. Compensates the great emphasis placed by *ED* on those features for which the dissimilarity is large. Also saves time.

Distance Measures

Maximum distance (Lagrange metric)

Considers only the most dissimilar pair of features as
$d_m(a, b) = \max_{i=1}^{n} |b_i - a_i|$

Minkowski distance

A generalization of the three is $d_p(a, b) = \{\sum_{i=1}^{n} |b_i - a_i|^p\}^{1/p}$ called L_p norm distance. Hence, the Manhattan and the ED are of norms L_1 and L_2, respectively.

Mahalanobis distance

Measures the statistical distance between two classes (i, j) of Gaussian mixtures having mean μ_i and μ_j, and a common covariance matrix Σ_{ij}
$d_m(i, j) = (\mu_i - \mu_j)^T \Sigma_{ij}^{-1} (\mu_i - \mu_j)$

Distance Measures

Cosine metric

Measure of similarity between two non zero vectors of an inner product space that measures the cosine of the angle between them. It is thus a judgment of orientation and not magnitude: two vectors with the same orientation have a cosine similarity of 1, at 90^O have a similarity of 0, and two vectors diametrically opposed have a similarity of -1, independent of their magnitude. $d_c(x, y) = \frac{xy}{|x||y|} = \frac{\sum_{i=1}^{d} x_i y_i}{\sqrt{\sum_{i=1}^{d} x_i^2}\sqrt{\sum_{i=1}^{d} x_i^2}}$

Canberra metric and Lance-Williams metric

The Canberra distance is a numerical measure of the distance between pairs of points in a vector space, introduced in 1966, as $d_c(x, y) = \frac{1}{d}\sum_{i=1}^{d} \frac{|x_i - y_i|}{x_i + y_i}$ and refined in 1967 by G. N. Lance and W. T. Williams. It is a weighted version of L (Manhattan) distance, has been used as a metric for comparing ranked lists. $d_{LW}(x, y) = \frac{\sum_{i=1}^{d} |x_i - y_i|}{\sum_{i=1}^{d} (x_i + y_i)}$

Histograms

- Easiest way of obtaining an approx density function $\hat{p}(x)$ from sampled data if no parametric form is assumed for the underlying density is to form a histogram
- The range of feature variable x is divided into a finite no of adjacent intervals called bins that include all of the data
- The no or fraction of samples falling within each interval is then plotted as a function of x, $f(x)$, as a bar graph (assuming a constant density within each interval of x)
- To use histogram as an estimate of true underlying continous density function, the area under the histogram must $= 1$
- Area under the density in each interval j is equal to the fraction of the total no N of samples that are in the interval: $p_j = \frac{n_j}{N_{\omega j}}$
- Once the approx density function is determined, decisions can be made using Bayes' theorem
- For discrete feature x, its range can be divided into intervals and the same technique can be used to fit the distribution of the samples that have each value of x_i can be used as an estimate of discrete distribution $P(x_i)$

Choosing Histograms Intervals

- If a small no of wide intervals is used, the no of samples falling within each interval will be relatively large, so the height and thus the area within the interval can be estimated quite correctly
- The resulting approx density will be flat over large regions and any fine structure in the true distribution is lost
- Using large no of histogram intervals can preserve the structure of the true density, but with too many intervals the confidence in their height decreses
- With large no of intervals, most of the apparaent structure depends on only a few samples, and thus cannot be very significant
- Tend to perceive structure in the data even when the "structure" is due to random fluctuations, leads to "overfitting", which degrades performance

Histograms Intervals

- One rule of thumb is to choose the no of intervals $= \sqrt{N}$, the no of intervals and the average no of samples per interval will be equal, giving a sort of equal precision to both scales
- Theoretical results indicate if the the true density $p(x)$ has finite continuous first and second derivatives, the mean square error between $p(x)$ and the estimated $\hat{p}(x)$ is minimized when the no of intervals is proportional to the cube root of the no of samples
- Histograms are not restricted to $1D$, e.g., $p(x, y)$ can be approx by dividing both x and y into intervals, and determining the no of samples that fall within each rectangular histogram bin with dimensions $\triangle x$ and $\triangle y$
- The volume under the surface of this $2D$ histogram is then normalized to 1, to yield an estimate of the density funtion $p(x, y)$

Histograms Intervals

- The square root rule of thumb can be generalized to produce an equal precison rule. When there are n features, the $(n+1)^{st}$ root is used
- Histogram technique becomes impractical for spaces of high dimensions, e.g., for $5D$, in which each of 5 discrete features could have 10 possible values (or each continous feature is divided into 10 intervals), there would be $10^5 = 100,000$ histogram bins, and several times this no of samples would be required to obtain a reasonable estimate of $p(x_1, \cdots, x_5)$. Sample sets of this size are usually not available
- If there were $1,000$ samples and 10 features, the equal precision rule would recommend $1000^{\frac{1}{11}} = 1.874$ intervals for each feature, with an average of 1.874 samples per bin. Rounding this to two intervals (high or low) for each feature would give $2^{10} = 1,024$ bins, with an average of $\frac{1000}{1024}$ or about one sample per bin
  - This may not produce a useful histogram, in many cases, many of the bins contain no data, so the average no of samples in the populated bins could be considerably larger

k - Nearest Neighbor

- Difficult to obtain a satisfactory estimate of the density everywhere when the samples are not evenly distributed
- Fix the number of samples, k, and let the sample width change so that each region contains exactly k samples.

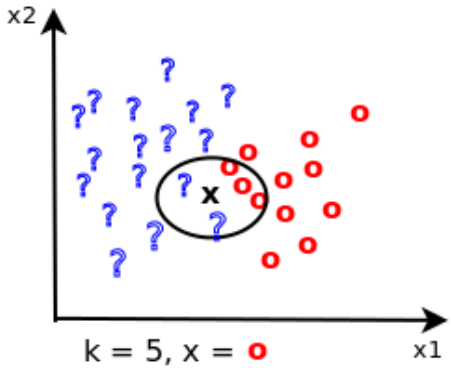
k-NN

The k-NN process starts at the test point and grows a region until it encloses k training samples and labels the test point x by a majority vote of these k samples.

"If it dances like a peacock, phe-ea-ao-n like a peacock, and looks like a peacock, then it is probably a peacock"

k - Nearest Neighbor

- For two classes, k should be odd to avoid a tie, larger values of k are more likely to resolve ties
- The region would be circular (or spherical in $3D$) if the data have been normalized
- Larger the value of k, the smoother the classification boundary; and the smaller the value of k, the more convoluted the bounday
- k should be large enough to reduce the sensitivity to noise, and small enough that the neighborhood does not extend to the domain of other classes



Breaking Ties

- Ties may be broken arbitrarily, may not always be sensible as the same x_{new} may be assigned to different classes, if it is tested more than once
- x_{new} may be assigned to the class of the nearest neighbor
- x_{new} may be assigned to the class, out of the classes with tying values of k', that has nearest mean vector to x (mean vector calculated over the k' samples)
- x_{new} may be assigned to the most compact class of the tying classes, i.e. to the one for which the distance to the k'^{th} member is the smallest
- More generally we can weight the votes according to distance such that the votes from closer points have greater influence, making ties highly unlikely

k - Nearest Neighbor classifier

- If $k = 1$ we have the NN classifier, recommended as the baseline classifier, to which the performance of more complicated classifiers should be compared
- If the ED metrix is used
  $d_e(x_i, x)^2 = |x_i - x|^2 = x^T x - 2x^T x_i + x_i^T x_i$
- Therefore, the NN rule is to assign the test sample to the class of the training data vector x_m satisfying

$$x^T x_m - \frac{1}{2} x_m^T x_m > x^T x_i - \frac{1}{2} x_i^T x_i \quad \forall i \neq m$$

- The rule has the form of a piecewise linear discriminant function

k - Nearest Neighbor Density Estimates

- k-NN rule is infact $\equiv$ to applying Bayes' rule to class conditional density estimates using a simple nonparametric method of density estimation
- The prob that a point $x^{'}$ falls within a volume V centered at a point x is $\theta = \int_{v(x)} p(x)dx$ where the integral is over the volume V. For small volume $\theta \sim p(x)V$
- The prob, θ, may be approximated by the prop of samples falling within V. If k, no of samples out of n, falling within V (k is a function of x) then $\theta \sim \frac{k}{n}$
- Combining the above we get an approxi for the density $\hat{p}(x) = \frac{k}{nV}$
- The k-NN approach to density estimation is to fix the prob, $\frac{k}{n}$ or $\equiv$ for a given no of samples n, to fix k and to determine the volume V which contains k samples centered on the point x

k - Nearest Neighbor

- For e.g., if x_k is the k^{th}-NN point to x, then V may be taken to be a sphere, centered at x of radius $|x - x_k|$
- The ratio of the prob to this volume gives the density estimate
- This is in contrast to the basic histogram approach where is to fix the cell size and then determine the no of points lying within
- If k, is too large value then the estimate will be smoothed and fine details averaged out. If too small, then the prob density estimate is likely to be spiky
- The density estimate is not infact a density. The integral under the curve is infinite, because for large enough $|x|$, the estimate varies as $\frac{1}{|x|}$, however, the density estimator is asymptotically unbiased and consistent if

$$\lim_{n\to\infty} k(n) = \infty, \quad \lim_{n\to\infty} \frac{k(n)}{n} = 0$$

k - Nearest Neighbor Decision Rule

- We can use the density estimate for decision rule
- Suppose that in the first k samples there are k_m in class ω_m (so that $\sum_{m=1}^{C} k_m = k$)
- Let the total no of samples in class ω_m be n_m (so that $\sum_{m=1}^{C} n_m = n$)
- The class density, $p(x|\omega_m)$, can be estimated as $\hat{p}(x|\omega_m) = \frac{k_m}{n_m V}$ and the prior prob, $p(\omega_m)$, as $\hat{p}(\omega_m) = \frac{n_m}{n}$
- The decision rule is to assign x to ω_m if $\hat{p}(\omega_m|x) \geq \hat{p}(\omega_j|x) \quad \forall\, j \neq m$
- Using Bayes' theorem $\frac{k_m}{n_m V}\frac{n_m}{n} \geq \frac{k_j}{n_j V}\frac{n_j}{n} \quad \forall\, j \neq m$
- Assign x to ω_m if $k_m \geq k_j \quad \forall\, j \neq m$

Performance of k Nearest Neighbor

- There is essentially no training involved in *k-NN* method
- It is intuitive, analytically tractable and very simple to implement, excellent empirical performance, and it lends itself easily to parallel implementation
- Can be used adaptively, as it uses local information (e.g., stretching the region in the dimension of the lowest variance)
- Can handle both binary and multiclass data and makes no assumption about the parameter form of the decision boundary. Decision surfaces are non-linear.
- Missclassification rate high due to disadvantage of outliers
- Performs well when training pattern size is large, but it increases computational complexity
- *k-NN* error rate can never be less than the Bayes error rate (chooses the most probable class/optimal choice). The *NN* error rate approaches the Bayes error rate when
  - Bayes error rate approaches either 0 or 1
  - When $k \to \infty$ and $\frac{\text{number of taining patterns}(n)}{k} \to \infty$

Performance of k Nearest Neighbor

- It is considered a lazy learning algorithm
- It defers data processing until it receives a request to classify an unlabeled (test) example
- Susceptible to curse of dimensionality, need increasing amounts of taining data as the dimensionality (no. features) increases, which can be compensated by
  - using weighted attributes during the computation of proximity measures, and
  - by eliminating irrelevant attributes a priori
- Significant overhead is computation of a large no of distance for each sample, and then discards any intermediate results unlike eager learning classifiers like trees and rule based; starts mapping as soon as the training data become available

k - Nearest Neighbor - Useful Tips

- Rule of thumb - Use at least 10 times as many training samples per class as the number of features
- Heuristic value of k, could be an integer approx of the square root of the number of training patterns of a class (with fewest training patterns). If no of samples in (C_i are $< \frac{k}{c}$) no testing sample will ever be classified as C_i
- For life effecting decisions - it is more useful to modify the criterion to assign a new vector to a particular class if atleast l of k nearest neighbors are in that particular class
  - the penalty for misclassifying one class (abnormal as normal - false negatives) is much greater than the penalty for misclassifying the other class (e.g., normal as abnormal - false positives) and
  - when there is an unbalanced training set, with many more samples in one class than the other

k - Nearest Neighbor - Feature Normalization

- Normalize the features to avoid unnecessary bias, so that there is equal potential influence of each feature.
- Normalizing according to range of the samples is a bit unrealistic, because it depends only on the highest and the lowest values of the features instead
  - normalize each feature x_i to have zero mean μ and unit σ by replacing x_i with $z_i = \frac{(x_i - \mu_i)}{\sigma_i}$ when ED is used, and
  - to make each feature have the same mean absolute deviation $MAD_i = \frac{1}{n}\sum_{j=1}^{n} |x_{ij} - M_i|$ of x_{ij} from its median M_i when city block distance is used
  - if a given $\triangle x_i$ in one portion of a feature range is considered more important than the same $\triangle x_i$ in another portion of its range, perform nonlinear transformation like log, exp, or power function on the feature

Feature Scaling Approaches

The input variables must be scaled to ensure that the *NN* rule is independent of measurement units

Scale to unit length or Maximum scaling

To scale the components of a feature vector to unit length: by dividing each component by the Euclidean length $x_j^{'} = \frac{x_j}{||x_j||}$ or the maximum of the feature vector:$x_j^{'} = \frac{x_j}{U_j}$

Range scaling or rescaling

To rescale the range of features to $[0, 1]$ or $[1, 1]$ by $x_j^{'} = \frac{x_j - L_j}{U_j - L_j}$

Profiles

To segment lines and words from an image $x_j^{'} = \frac{x_j}{\sqrt{\sum_{i=1}^{n}(y_{i,j}^2)}\sqrt{\sum_{j=1}^{d}(x_j^2)}}$

k - Nearest Neighbor - Feature Scaling

- Features could be scaled according to their importance or desired contribution to decision making
  - let the range or σ of each feature, or the weight of each normalized feature be proportional to the accuracy of the feature, defined as the probability of a correct decision when that feature is used alone for decision making, or
  - make the scale factor of each feature proportional to some power of a of this probability of being correct when the feature is used alone.
- Combining nonlinear accuracy weighting with the Minkowski distance concept produces a distance metric with two adjustable parameters $d_{ar} = \sum_{i=1}^{n} \{P(C|i)^a |x_{iu} - x_{icj}|\}^r$ where $P(C|i)$ is the probability of being correct when only feature i is used, which is estimated by actually classifying a data set on the basis of that single feature alone. Taking $r = 1$ and $a = 0$ yields the unweighted city block distance and setting $r = 2$ and $a = 0$ gives the unweighted ED squared

Other Nearest Neighbor Techniques

- $k + k$- NN, finding the k NN from each class, and computing, for each class, the average distance from the test sample. The class that has the samllest average distance is chosen
- Uses k-NN, regardless of their classes, but then uses weighted vote from each sample rather than a simple majority or plurality voting rule. The weighted votes are then summed from each class, and the class with the largest total vote is chosen
- Nearest prototype $\equiv$ to single NN, except only one typical sample from each class is used for the refernces set of possible neighbors
- Edited NN, use only a portion of the original training set as a reference set of potential neighbors and the rest of the set is discarded or edited out

Nearest Neighbor Error Rates

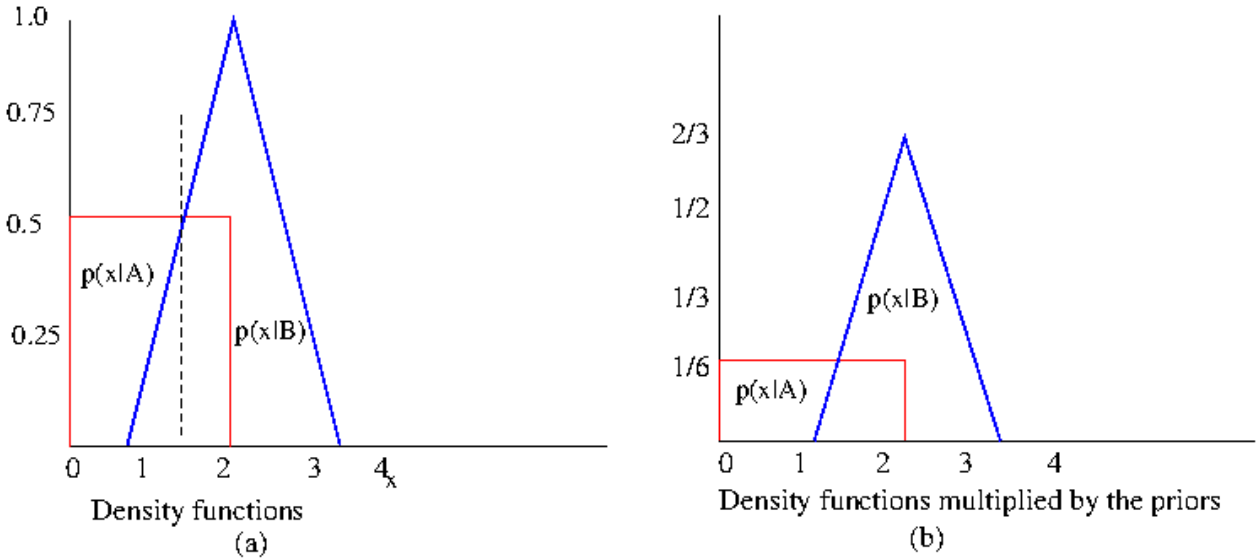
- Expected error rate for NN can never be $<$ of a Bayesisn classifier (always chooses the most probable class, which is an optimal choice)
- Let for a member of class C_i with a feature vector of $\mathbf{x}$, the prob that it will have a NN from C_i and be correctly classified by NN technique is $P(C_i|\mathbf{x})$ at any $\mathbf{x}$
- The prob density for finding a sample to be classified from class C_i at $\mathbf{x}$ is $p(\mathbf{x}|C_i)$
- The expected value of the prob of correct classification for members of the class C_i, averaged over all possible values of $\mathbf{x}$, each weighted by its prob of occurrence, is $P(C|C_i)_{NN} = \int_{R^n} P(C_i|\mathbf{x})p(\mathbf{x}|C_i)d\mathbf{x}$, the multiple integration is performed over the entire n-dim space R^n
- Bayes' Theorem yields $P(C|C_i)_{NN} = P(C_i)\int_{R^n} \frac{p(\mathbf{x}|C_i)^2}{p(\mathbf{x})}d\mathbf{x}$ where $p(\mathbf{x}) = \sum_i P(C_i)p(\mathbf{x}|C_i)$, is the mixture density over all the classes

Nearest Neighbor Error Rates

- The prob that a member of class C_i is classified incorrectly (prob of error) is
  $P(\epsilon|C_i)_{NN} = 1 - P(C|C_i)_{NN} = 1 - P(C_i)\int_{R^n} \frac{p(\mathbf{x}|C_i)^2}{p(\mathbf{x})} d\mathbf{x}$
- Another useful equation is found by defining
  $P(\bar{C}_i) = \sum_{j \neq i} P(C_j) = 1 - P(C_i)$, then
- $$P(C|C_i)_{NN} = \int_{R^n} P(C_i|\mathbf{x})p(\mathbf{x}|C_i)d\mathbf{x} = \int_{R^n}[1 - P(\bar{C}_i|\mathbf{x})]p(\mathbf{x}|C_i)d\mathbf{x}$$
  $$= \int_{R^n} p(\mathbf{x}|C_i)d\mathbf{x} - \int_{R^n} P(\bar{C}_i|\mathbf{x})p(\mathbf{x}|C_i)d\mathbf{x}$$
  $$= 1 - \int_{R^n} P(\bar{C}_i|\mathbf{x})p(\mathbf{x}|C_i)d\mathbf{x}$$
  $$P(\epsilon|C_i)_{NN} = 1 - P(C|C_i)_{NN} = \int_{R^n} P(\bar{C}_i|\mathbf{x})p(\mathbf{x}|C_i)d\mathbf{x}$$
- Substituting Bayes' Theorem
  $P(\epsilon|C_i)_{NN} = P(\bar{C}_i)\int_{R^n} \frac{P(\mathbf{x}|\bar{C}_i)p(\mathbf{x}|C_i)}{p(\mathbf{x})} d\mathbf{x}$
- For overall error, prob of error of each class is weighted by the prior prob $\rightarrow P(\epsilon)_{NN} = \sum_{i=1}^{c} P(C_i)P(\epsilon|C_i)_{NN} =$
  $1 - \sum_{i=1}^{c} P(C_i)^2 \int_{R^n} \frac{p(\mathbf{x}|C_i)^2}{p(\mathbf{x})} d\mathbf{x}$

Exercise

- What are the Bayesian and the *NN* error rates for the two classes shown in Figure (a)



- Given that $P(A) = \frac{1}{3}, P(B) = \frac{2}{3}$. The two densities have been multiplied by their respective priors to produce Figure (b)

Bayes Error

- For the Bayesian case, the minimum error decision boundary , is $P(A)p(x|A) = P(B)p(x|B)$, lies at the value $1 \leq x \leq 2$ or $P(A)p(x|A) = \frac{1}{6}$, and $P(B)p(x|B) = \frac{2}{3}(x-1)$
- Equating $\frac{1}{6} = \frac{2}{3}(x-1)$, gives $x = \frac{5}{4}$
- The prob of error for members of class A is $P(\epsilon|A)_{Bayes} = \int_{\frac{5}{4}}^{2} p(x|A)dx = \int_{\frac{5}{4}}^{2} \frac{1}{2}dx = \frac{3}{8}$
- For class B, $P(\epsilon|B)_{Bayes} = \int_{1}^{\frac{5}{4}} p(x|B)dx = \int_{1}^{\frac{5}{4}} (x-1)dx = \frac{1}{32}$
- $P(\epsilon)_{Bayes} = \frac{1}{3} \times \frac{3}{8} + \frac{2}{3} \times \frac{1}{32} = \frac{7}{48} = 0.1458$

NN Error

- To obtain *NN* error rate we use
  $P(\epsilon|A)_{NN} = P(B)\int_{-\infty}^{\infty} \frac{p(x|A)p(x|B)}{p(x)}dx$
- The only region where both densities are nonzero is $1 \leq x \leq 2$, so the integral is zero outside this range
- Thus
  $$P(\epsilon|A)_{NN} = \frac{2}{3}\int_1^2 \frac{(1/2)\times(x-1)}{(1/3)\times(1/2)+(2/3)\times(x-1)}dx$$
  $$= \int_1^2 \frac{(x-1)}{2\times(x-1)+(1/2)}dx$$
- Making the substitution $y = x - 1$ gives
  $P(\epsilon|A)_{NN} = \int_0^1 \frac{y}{2y+1/2}dy = (1/4)[2y + 0.5 - 0.5\ln(2y + 0.5)]_0^1$
  $= 1/4(2 - 0.5\ln 2.5 + 0.5\ln 0.5) = 0.2988$
- Calculation for class B is same except the prior is $\frac{1}{3}$, so
  $P(\epsilon|B)_{NN} = \frac{1}{3}\int_1^2 \frac{(1/2)\times(x-1)}{(1/3)\times(1/2)+(2/3)\times(x-1)}dx = \frac{0.2988}{2} = 0.1494$
- $P(\epsilon)_{NN} = (1/3) \times (0.2988) + (2/3)(0.1494) = 0.1992$, which is 1.366 times the Bayesian error
- Note that the class B, which has twice the prob of class A, has half the overall error rate of class A, so the expected total number of errors for each type will be equal in the population