BINUS UNIVERSITY

People
Innovation
Excellence

Course : COMP6577 – Machine Learning
Effective Period : February 2020

Classification Trees

Session 21 & 22

Learning Outcome

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

Outline

- Introduction
- Ordinary Binary Classification Trees
- Challenge
- Drawback
- Random Forest
- Case Study

Introduction

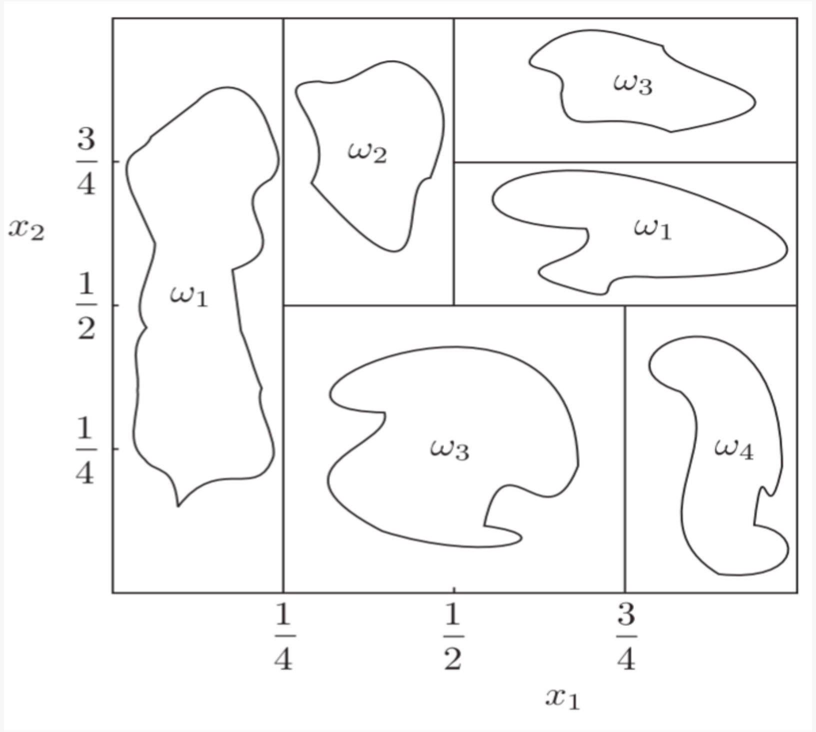
- Classification trees are based on a simple, yet powerful, idea, and they are among the most popular techniques for classification.
- They are multistage systems, and classification of a pattern into a class is achieved sequentially.
- Through a series of tests, classes are rejected in a sequential fashion until a decision is finally reached in favor of one remaining class.
- Each one of the tests, whose outcome decides which classes are rejected, is of a binary "Yes" or "No" type and is applied to a single feature.
- The goal is to present the main philosophy around a special type of trees known as ordinary binary classification trees (OBCT). They belong to a more general class of methods that construct trees, both for classification as well as regression, known as classification and regression trees (CART)

BINUS UNIVERSITY

People
Innovation
Excellence

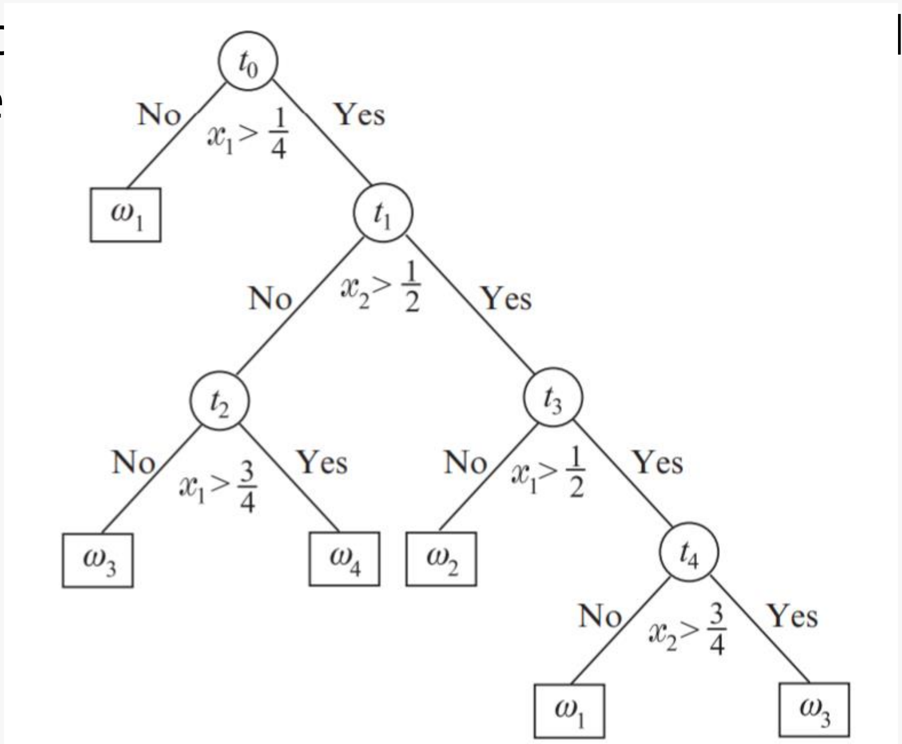
Ordinary Binary Classification Trees

- The basic idea around OBCTs is to partition the feature space into (hyper) rectangles; that is, the space is partitioned via hyperplanes, which are parallel to the axes.
- This is illustrated in the following figure.



Partition of the two-dimensional features space, corresponding to three classes, via a classification (OBCT) tree.

- The partition of the space in (hyper)rectangles is performed via a series of "questions" of this form: is the value of the feature $x_i < a$? This is also known as the **splitting criterion**.
- The sequenc l via the use of a tree



People
Innovation
Excellence

- Each node of the tree performs a test against an individual feature and, if it is not a leaf node, it is connected to two descendant nodes: one is associated with the answer “Yes” and the other with the answer “No.”
- Starting from the root node, a path of successive decisions is realized until a leaf node is reached. Each leaf node is associated with a single class. The assignment of a point to a class is done according to the label of the respective leaf node. This type of classification is conceptually simple and easily interpretable.

BINUS UNIVERSITY

Example

- In a medical diagnosis system, one may start with a question: is the temperature high? If yes, a second question can be: is the nose runny? The process carries on until a final decision concerning the disease has been reached.
- Also, trees are useful in building up reasoning systems in artificial intelligence. The existence of specific objects, which is deduced via a series of related questions based on the values of certain (high-level) features, can lead to the recognition of a scene or of an object depicted in an image.
- Once a tree has been developed, classification is straightforward.

People
Innovation
Excellence

Challenge

- The major challenge lies in constructing the tree, by exploiting the information that resides in the training data set.
- The main questions one is confronted with while designing a tree are:
  - Which splitting criterion should be adopted?
  - When should one stop growing a tree and declare a node as final?
  - How is a leaf node associated with a specific class?

Splitting criterion

- We have already stated that the questions asked at each node are of the type is $x_i < a$? The goal is to select an appropriate value for the threshold value a.
- Assume that starting from the root node, the tree has grown up to the current node, *t*. Each node, *t*, is associated with a subset $X_t \subseteq X$ of the training data set, *X*. This is the set of the training points that have survived to this node, after the tests that have taken place at the previous nodes in the tree.
- For example in the tree (previous slide) a number of points, which belong to, say, class ω_1, will not be involved in node t_1 because they have already been assigned in a previously labeled leaf node

- The purpose of a splitting criterion is to split X_t into two disjoint subsets, namely X_{tY} , and X_{tN}, depending on the answer to the specific question at node *t*. For every split, the following is true:

$$X_{tY} \cap X_{tN} = \emptyset,$$

$$X_{tY} \cup X_{tN} = X_t.$$

- The goal in each node is to select which feature is to be tested and also what is the best value of the corresponding threshold value *a*. The adopted philosophy is to make the choice so that every split generates sets, X_{tY} , X_{tN}, which are more class-homogeneous compared to $X_{t.}$
- In other words, the data in each one of the two descendant sets must show a higher preference to specific classes, compared to the ancestor set

- For example, assume that the data in X_t consist of points that belong to four classes, ω_1, ω_2, ω_3, ω_4. The idea is to perform the splitting so that most of the data in X_{tY} belong to, say, ω_1, ω_2 and most of the data in X_{tN} to ω_3, ω_4. In the adopted terminology, the sets X_{tY} and X_{tN} should be purer compared to X_t. Thus, we must first select a criterion that measures impurity and then compute the threshold value and choose the specific feature (to be tested) to maximize the decrease in node impurity.
- For example, a common measure to quantify impurity of node, t, is the

$$I(t) = -\sum_{m=1}^{M} P(\omega_m|t) \log_2 P(\omega_m|t).$$

where log2(·) is the base-two logarithm. The maximum value of I(t) occurs if all probabilities are equal (maximum impurity), and the smallest value, which is equal to zero, when only one of the probability values is one and the rest equal zero

- Probabilities are approximated as

$$P(\omega_m|t) = \frac{N_t^m}{N_t}, \quad m = 1, 2, \ldots, M$$

where $I(t_Y)$ and $I(t_N)$ are the impurities associated with the two new sets, respectively.

- The goal now becomes to select the specific feature, x_i, and the threshold at so that $\Delta I(t)$ becomes maximum.
- This will now define two new descendant nodes of t, namely, t_N and t_Y ; thus, the tree grows with two new nodes.

People
Innovation
Excellence

- A way to search for different threshold values is the following:
  - For each one of the features, x_i, $i = 1, 2, \dots, l$, rank the values, x_{in}, $n = 1, 2, \dots, N_t$, which this feature takes among the training points in X_t.
  - Then define a sequence of corresponding threshold values, a_{in}, to be halfway between consecutive distinct values of x_{in}.
  - Then test the impurity change that occurs for each one of these threshold values and keep the one that achieves the maximum decrease.
  - Repeat the process for all features, and finally, keep the combination that results in the best maximum decrease.

- Besides entropy, other impurity measuring indices can be used. A popular alternative, which results in a slightly sharper maximum compared to the entropy one, is the so-called **Gini index**, defined as:

$$I(t) = \sum_{m=1}^{M} P(\omega_m|t)\big(1 - P(\omega_m|t)\big)$$

- This index is also zero if one of the probability values is equal to 1 and the rest are zero, and it takes its maximum value when all classes are equiprobable.

People
Innovation
Excellence

Stop-splitting rule

- One possible way is to adopt a threshold value, *T*, and stop splitting a node once the maximum value $\Delta I(t)$,for all possible splits, is smaller than *T*.
- Another possibility is to stop when the cardinality of X_t becomes smaller than a certain number or if the node is pure, in the sense that all points in it belong to a single class.

Class assignment rule

- Once a node, t, is declared to be a leaf node, it is assigned a class label; usually this is done on a majority voting rationale. That is, it is assigned the label of the class where most of the data in X_t belong.

Pruning a tree

- Experience has shown that growing a tree and using a stopping rule does not always work well in practice; growing may either stop early or may result in trees of very large size.
- A common practice is to first grow a tree up to a large size and then adopt a pruning technique to eliminate nodes. Different pruning criteria can be used; a popular one is to combine an estimate of the error probability with a complexity measuring index.

Drawback

- A major drawback associated with the tree classifiers is that they are **unstable**.
- A small change in the training data set can result in a very different tree. The reason for this lies in the hierarchical nature of the tree classifiers. An error that occurs in a node at a high level of the tree propagates all the way down to the leaves below it.

People
Innovation
Excellence

Random Forests

- Bagging (Bootstrap Aggregating) is a technique that can reduce the variance and improve the generalization error performance.
- The basic idea is to create a number of B variants, X_1, X_2, ... , X_B, of the training set, X, using bootstrap techniques, by uniformly sampling from X with replacement. For each of the training set variants, X_i, a tree, T_i, is constructed.
- The final decision for the classification of a given point is in favor of the class predicted by the majority of the subclassifiers, T_i, $i = 1, 2, \dots, B$.
- Random forests use the idea of bagging in tandem with random feature selection. The difference with bagging lies in the way the decision trees are constructed.

- The feature to split in each node is selected as the best among a set of *F* randomly chosen features, where *F* is a user-defined parameter. This extra introduced randomness is reported to have a substantial effect in performance improvement.
- Random forests often have very good predictive accuracy and have been used in a number of applications, including for body pose recognition in terms of Microsoft’s popular Kinect sensor.

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

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

- Do classification for the Singapore Airbnb data above using classification trees method.

End of Session 21 & 22

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