Examples of Machine Learning Algorithms for Optical Network Control and Management

Alba P. Vela*, **Marc Ruiz, and Luis Velasco**
Universitat Politècnica de Catalunya (UPC), Barcelona, Spain
**e-mail: apvela@ac.upc.edu*

ABSTRACT
Machine learning (ML) offers a great variety of algorithms that can be used in the context of optical networks. In particular, ML algorithms might be applied for classification and to detect patterns, among others. Both, can help to facilitate improving its performance, as well as to understand the behavior of optical networks. In this paper, we review two of these ML algorithms, one for classification and the other for clustering. Illustrative examples of the application of such supervised and unsupervised ML algorithms applied to optical networks are presented.
Keywords: machine learning, support vector machine, clustering, data visualization.

1. INTRODUCTION

Machine learning (ML) is omnipresent nowadays as a technique to find relations between inputs and outputs, where no specific analytical equations (or algorithms) can be found to perform such transformation. In fact, as stated in [1]: *What we lack in knowledge, we make up for in data*. ML is an incredibly powerful tool for Big Data, predicting future behaviour.

Algorithms for ML can be classified into three different categories according to the input received data [2], [3]:

- **Supervised learning**, when the training data contains examples (data points) of the input vectors and their corresponding target vectors, in other words, when a *label* is available for a subset of the dataset. The training set is used to generate a function to map inputs with the desired outputs until the model achieves a certain level of accuracy. This kind of algorithms is often used for *classification*; predicting to which category does a new data point belong.
- **Unsupervised learning** is applied when the training data consists of a set of input vectors without any corresponding target value to predict. As an example, *clustering* targets at finding collection of data points with similar features among them but different from other data points in input data.
- **Reinforcement learning** is based on training the machine to make specific decisions. Initially it is continuously self-trained using trial and error. Then, it learns from experience and captures this knowledge to make accurate decisions, such as Markov decision process.

Another way to classify ML algorithms is according to their function similarity:

- **Regression algorithms** deal with modelling the relationship between variables; i.e., simple and multiple linear regression, as well as polynomial regression.
- **Classification algorithms**, aiming at deciding at which already know group does a new point belong; i.e., *support vector machines* (SVM), Naive Bayes, and decision tree classification.
- **Clustering algorithms** are focused on modelling as centroid-based; i.e., *K-means*.

Several works in the literature in the field of Optical Networking use ML algorithms. For instance, authors in [4] use ML algorithms for failure classification, whereas the authors in [5] use predictive models based on ML that are used for network reconfiguration. In fact, the number of applications of ML is huge. As for architectures to support ML, the authors in [6] present a framework named CASTOR that collects monitoring data, which can be used for many purposes. For a tutorial in ML and applications for optical networks we refer the reader to [7].

Another application of ML is for data visualization [8]. Data visualization is a technique that involves processing large amounts of (raw) data aiming at discovering patterns and presenting them in a clear way using graphical resources. It is worth noting that data visualization is extremely helpful to discover hidden behavior in the network.

In this paper, we go insight two popular algorithms for classification (SVM) and clustering (K-means) which can be used for optical network related problems, such as, soft failure detection. The rest of the paper is organized as follows. Section 2 gives a short introduction to SVM and K-means algorithms. Then, Section 3 presents the benefits of SVM for filter related soft-failure detection. Section 4 shows the application of the K-means algorithm; this algorithm can be particularly useful for further data visualization purposes. Finally, Section 5 concludes the paper.

2. INTRODUCTION TO SVM AND K-MEANS

2.1 Support Vector Machine (SVM)

The SVM is a supervised learning classification algorithm mainly used for binary classification. We start from a dataset, where every sample in the dataset represents a point in a n-dimensional space (let us assume a 2D space for simplicity), and each feature is represented as a particular coordinate in such space. For illustrative purposes, let us consider the example in Fig. 1a where the data points in a dataset are represented with colors (are labeled), identifying the two different classes. Note that every example is positioned using (x_1, x_2) coordinates, where x_1 and x_2 are two different features.

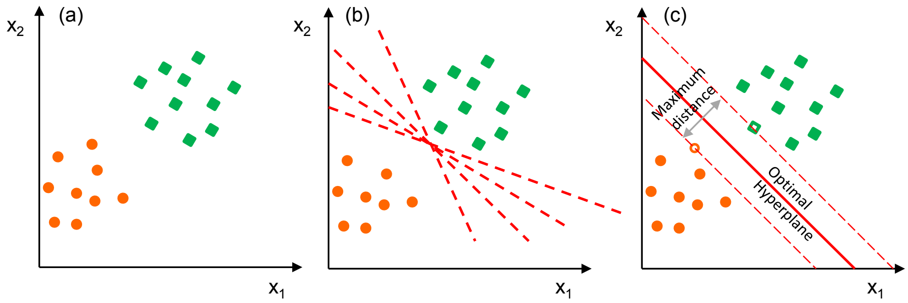

Figure 1: (a) Two-class classification problem (dots and squares); (b) Different possible choices for the optimal hyperplane; (c) The optimal hyperplane with the support vectors over the margins.

Given a training dataset, a SVM can be found in order to build a model to predict the class of a new unclassified sample. The target of the training phase is then to find a boundary, called hyperplane, that separates data points into two groups (classes) of similar characteristics. When the two groups can be perfectly divided by a hyperplane, they are said to be linearly separable [3]. Infinite hyperplanes can be found separating the data points into two different classes; the choice of the best hyperplane could be based on the one that represents the largest separation between the two classes. As observed in Fig. 1 the choice of the optimal hyperplane is not straightforward; infinite hyperplanes, lines in this 2D example, can be used to correctly separate both classes, as suggested in Fig. 1b. The SVM algorithm solves an optimization problem to find the optimal hyperplane, the one that maximizes the distance between the classes, as presented in Fig. 1c.

In certain cases, data is not linearly separable by a simple hyperplane. To extend SVM these situations, the so-called soft margins need to be considered that allows some data points to fall on the incorrect side of the hyperplane. In the maximization problem, a cost C represents the penalty of a not perfect classification. The importance of the parameter C will be illustrated in Section 3.

2.2 Clustering

Among the clustering algorithms that can be found in ML, *K*-means is one of the most popular. *K*-means is designed to find a given number *K* of clusters in a dataset. This technique assigns each of the *n* data points in the input dataset to one of the *K* clusters aiming at minimizing the differences within each cluster and maximizing the differences between the clusters[3].

For illustrative purposes, let us consider the example in Fig. 2a, where data points are represented in a 2D space, as for the SVM example above, but now, they are all of the same color, meaning that data is unlabeled. Although it is unclear in view of the distribution of data, how many clusters can be found, and which data points belong to which cluster, let us assume that we target at finding 2 different clusters. Note that the decision of the number of clusters to be found is usually not based some external information, not on the existing data.

K-means selects *K* points, named centroids, that do not need necessarily to belong to the dataset, and places at random in the problem space, as presented in Fig. 2b. Then, each of the data points is assigned to the closest centroid, as suggested in Fig. 2c. For the sake of clarity, in the example we are considering Euclidean distances. Now, every data point is represented with the color of the centroid thus, forming *K* clusters. Once we have the clusters, the new centroid representing the clusters can be computed the center of gravity of the cluster (Fig. 2d). With the new set of centroids, steps b-d are repeated (Fig. 2e-2g), since according to the new centroids, some points are now not assigned to the closer cluster. The process is repeated for a number of fixed iterations or until no reassignments can be done. The final clusters after an iterative process are shown in Fig. 2h.

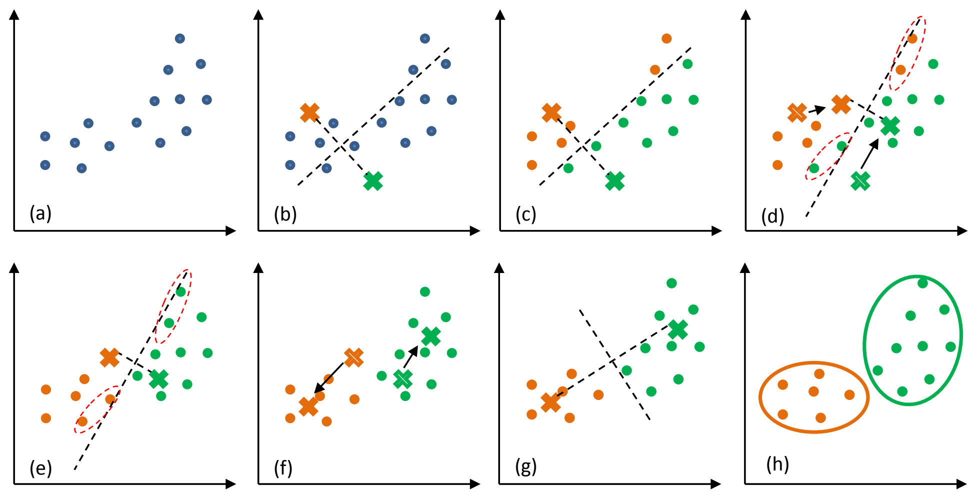

Figure 2: (a) Input data; (b) Selection of random centroids; (c) Assigning data points to clusters; (d) New centroids placement; (e) Data points reassigned; (f), (g) Iteration of the previous process; (h) Final clusters.

3. APPLYING SVM FOR SOFT-FAILURE DETECTION

Let us consider an illustrative example for optical networks where SVMs can be applied for classification; that of soft failures affecting established lightpaths [4]. Two different filter-related soft-failures are considered in Fig. 3, where the solid line represents the optical spectrum of the expected normal signal at the measurement point, whist the solid area represents the optical spectrum of the signal with failure. In the case of filter shift, a 10 GHz shift to the right was applied (Fig. 2a), whereas the signal is affected by a 20GHz in filter tightening (Fig. 2b). In the work in [4], the authors consider a number of features that help to characterize an optical signal. Among those features, in this paper we use two: *bandwidth*, as a measure of the amount filtering suffered by a signal, and *symmetry*, as a measure of filtering affecting more to one of the sides of the signal.

The benefit of the SVM algorithm relies on the fact that, given a labeled dataset (Fig. 4a), the algorithm can be trained to classify a new signal; two classes are considered here *Class 1* refers to filter shift, while *Class 2* represents filter tightening. Fig. 4b shows the testing dataset. In this case, although the two classes are linearly separable, we will study whether is better to completely separate them or not by selecting different costs *C* and analysing the obtained results.

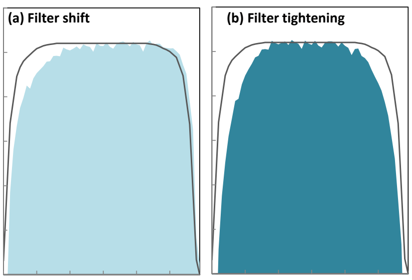

Figure 3. Example of filter-related failures.

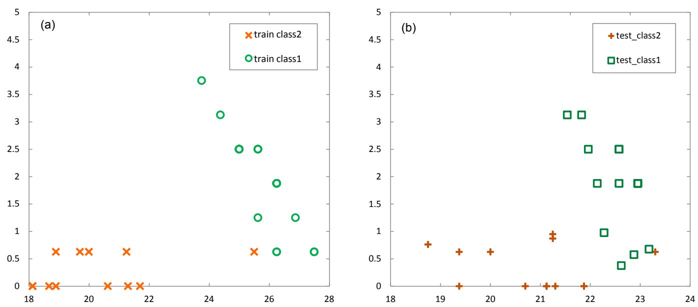

Figure 4. Data points that characterize two types of filter soft failure.

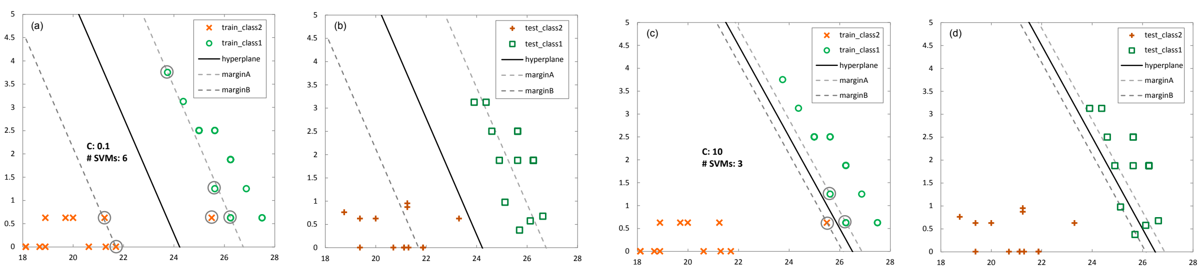

Figure 5. SVM hyperplane and margins computed with training data (Y axis is bandwidth and X axis symmetry) for two different costs (a) C= 0.1 and (c) C= 10. (b) and (d) use SVM to classify the testing data.

Figure 5 presents an illustrative example of the use of lineal kernel with different cost values ($C = 0.1$ and $C = 10$). Parameter *C* allows to decide the penalty of misclassifying data; a low *C* value prioritizes simplicity (soft margin), while high *C* values prioritize making as few mistakes, tending to overfitting.

Figure 5a shows the hyperplane computed for $C = 0.1$; for this value, 6 support vectors are needed. Figure 5b evaluate the just obtained model on the test dataset. The same study has been considered for $C = 10$ (Fig. 5c and 5d). In this case, the number of support vectors decreases down to 3. In particular, with $C = 10$, the constraint penalizes classification mistakes thus, avoiding misclassification. However, for small values of C, the optimization chooses a bigger-margin hyperplane in contrast with the case of high values of C. Table 1 summarizes the results presented in Fig. 5c presenting confusion matrix with true/false negative/positive.

Table 1. Confusion matrix.

	Real			
	C=0.1		**C=10**	
Prediction	Class 1	Class 2	Class 1	Class 2
Class 1	15	6	15	2
Class 2	0	8	0	12

4. USING K-MEANS FOR DATA VISUALIZATION

Let us now consider another interesting example concerning optical networks, where the clustering algorithm can be particularly useful to discover patterns that are used for visualization purposes. We assume an optical network with a set of established lightpaths, where its bit error rate (BER) is being monitored. By performing a set of transformations, data points are represented using two features normalized in the range [0, 1] (Fig. 6a): *i*) *BER ratio* compares the measured BER against the expected one for a lightpath, and *ii*) *BER trend* reports the evolution of the measured BER with time [8]. It can be seen that data points are mostly groped around small horizontal values. However, if we apply the clustering algorithm to find the centroids we can obtain more information about the BER distribution (Fig. 6b). Finally, a bubbles chart is used for visualization, which takes advantage of the use of colors and sizes (Fig. 6c); the bubble size gives information about the number of lightpaths that it includes, while its position gives information of the relation between BER features, and finally, the color represent the L2-norm of the bubble position, which gives information about the severity.

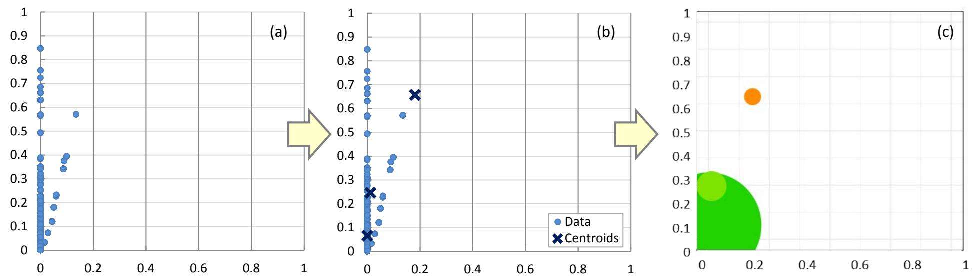

Figure 6. Applying clustering algorithm (Y axis is BER ratio and X BER trend): (a) monitored, (b) clustering centroids and (c)bubbles visualization chart.

5. CONCLUSIONS

In this work, we have introduced the main concepts of two interesting ML algorithms and illustrated their application for optical networks control and management. First, the SVM classification algorithm has been introduced aiming at targeting soft-failure detection by analyzing the optical spectrum of the signal. Secondly, the interest of clustering algorithms arises to find knowledge on unlabeled data. The K-means algorithm has been used for clustering lightpaths BER monitoring data used for visualization purposes.

ACKNOWLEDGEMENTS

The research leading to these results has received funding from the EC through the METRO-HAUL project (G.A. no. 761727), from the Spanish MINECO TWINS project (TEC2017-90097-R), and from the ICREA Institution.

REFERENCES

[1] E. Alpaydın, *Introduction to Machine Learning*, Second Edition, The MIT Press, 2010.
[2] Ch. Bishop, *Pattern Recognition and Machine Learning*, Springer-Verlag, 2006.
[3] Brett Lantz, *Machine Learning with R*, Packt Publishing Ltd, 2nd Edition 2013.
[4] A. P. Vela *et al.*, "Soft failure localization during commissioning testing and lightpath operation," *IEEE/OSA Journal of Optical Communications and Networking (JOCN)*, vol. 10, pp. A27-A36, 2018.
[5] F. Morales *et al.*, "Dynamic core VNT adaptability based on predictive metro-flow traffic models," *IEEE/OSA Journal of Optical Communications and Networking (JOCN)*, vol. 9, pp. 1202-1211, 2017.
[6] Ll. Gifre *et al.*, "Autonomic disaggregated multilayer networking," *IEEE/OSA Journal of Optical Communications and Networking (JOCN)*, vol. 10, pp. 482-492, 2018.
[7] D. Rafique and L. Velasco, "Machine learning for network automation [Invited Tutorial], *IEEE/OSA Journal of Optical Communications and Networking (JOCN)*, 2018.
[8] A. P. Vela, M. Ruiz, and L. Velasco, "Applying data visualization for failure localization," in *Proc. OFC*, 2018.