Semi-supervised Learning

Karamoustou Vasiliki, Papadouli Vasiliki, Panagiotou Dimitra

November 2021

1 Data

The dataset consists of 14 feature variables and 1 class label that quantifies the approval decision. Not much is known about the 14 features themselves for the sake of confidentiality and for convenience of processing statistical algorithms. This means that even the feature names are not present, but whether they are continuous or categorical is known. I have similarly labelled which columns are continuous (N), categorical (C) and if they require further encoding (C_enc)$inthedataset$.

Due to the fact that we deal with a classification problem and for the purposes of this project it is essential to apply semi-supervised techniques, we add a new column in the dataset that contains only 100 labelled observations and the rest of the original dataset is marked as unlabelled (by replacing the original value to a new one (-1) which stands for the don't know value).

2 Semi-Supervised Learning

Semi-supervised learning is the branch of machine learning concerned with using labelled as well as unlabelled data to perform certain learning tasks. Generally, the main intrest of research on semi-supervised learning is focused on classification. Semi-supervised classification methods are particularly relevant to scenarios where labelled data is scarce. In those cases, it may be difficult to construct a reliable supervised classifier. This situation occurs in application domains where labelled data is expensive or difficult obtain, like computer-aided diagnosis, drug discovery and part-of-speech tagging. If sufficient unlabelled data is available and under certain assumptions about the distribution of the data, the unlabelled data can help in the construction of a better classifier.

3 Inductive vs Transductive Learning

The main difference is that during transductive learning, you have already encountered both the training and testing datasets when training the model. However, inductive learning encounters only the training data when training the

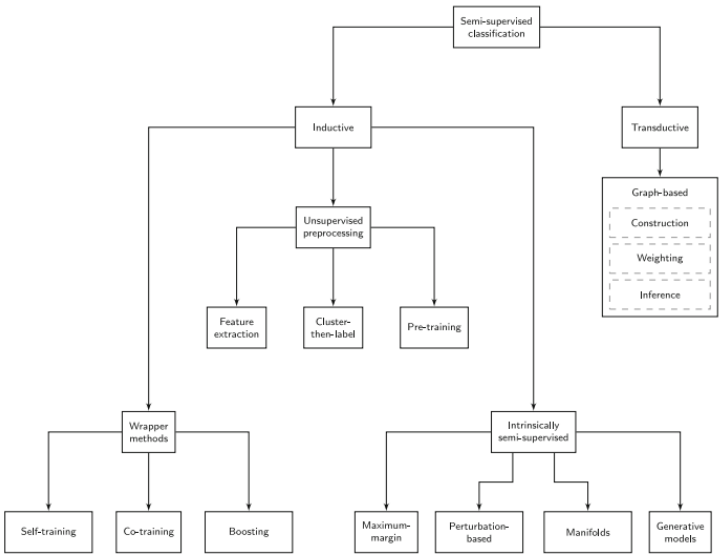

Figure 1: Taxonomy of semi supervised techniques

model and applies the learned model on a dataset which it has never seen before. Transduction does not build a predictive model. If a new data point is added to the testing dataset, then we will have to re-run the algorithm from the beginning, train the model and then use it to predict the labels. On the other hand, inductive learning builds a predictive model. When you encounter new data points, there is no need to re-run the algorithm from the beginning. In more simple terms, inductive learning tries to build a generic model where any new data point would be predicted, based on an observed set of training data points. Here you can predict any point in the space of points, beyond the unlabelled points. In contrary, transductive learning builds a model that fits the training and testing data points it has already observed. This approach predicts labels of unlabelled points using the knowledge of the labelled points and additional information.

4 Inductive Learning

4.1 Definition

Induction is reasoning from observed training cases to general rules, which are then applied to the test cases.

Inductive learning is the same as what we commonly know as traditional supervised learning. We build and train a machine learning model based on a labelled training dataset we already have. Then we use this trained model to

predict the labels of a testing dataset which we have never encountered before.

4.2 Wrapper Methods

A simple approach to extending existing, supervised algorithms to the semi-supervised setting is to first train classifiers on labelled data, and to then use the predictions of the resulting classifiers to generate additional labelled data. The classifiers can then be re-trained on this pseudo-labelled data in addition to the existing labelled data. Such methods are known as wrapper methods: the unlabelled data is pseudo-labelled by a wrapper procedure, and a purely supervised learning algorithm, unaware of the distinction between originally labelled and pseudo-labelled data, constructs the final inductive classifier.

4.2.1 Self Training

Self-training methods consist of a single supervised classifier that is iteratively trained on both labelled data and data that has been pseudo-labelled in previous iterations of the algorithm. At the beginning of the self-training procedure, a supervised classifier is trained on only the labelled data. The resulting classifier is used to obtain predictions for the unlabelled data points. Then, the most confident of these predictions are added to the labelled data set, and the supervised classifier is re-trained on both the original labelled data and the newly obtained pseudo-labelled data. This procedure is typically iterated until no more unlabelled data remain.

	precision	recall	f1-score	support
0.0	0.76	0.89	0.82	115
1.0	0.82	0.64	0.72	92
accuracy			0.78	207
macro avg	0.79	0.76	0.77	207
weighted avg	0.78	0.78	0.77	207

Figure 2: Classification Report from Self Training Method

4.3 Unsupervised Preprocessing

Unsupervised preprocessing use the unlabelled data and labelled data in two separate stages. Typically, the unsupervised stage comprises either the automated extraction or transformation of sample features from the unlabelled data (feature extraction), or the unsupervised clustering of the data (cluster-then-label), or the initialization of the parameters of the learning procedure (pre-training).

4.3.1 Cluster then Label

Many semi-supervised learning algorithms use principles from clustering to guide the classification process. Cluster-then-label approaches usually apply an unsupervised or semi-supervised clustering algorithm to all available data, and use the resulting clusters to guide the classification process. In our case, we cluster both labelled and unlabelled coming from train dataset and then we use the majority of labelled instances in each cluster to assign a label in the whole cluster. That way, training dataset which previously contained both labelled and unlabelled data, now contains only labelled data. In this dataset we apply a supervised learning method (Logistic Regression) and then based on that learner we predict our test data originating from the initial splitting.

	precision	recall	f1-score	support
0.0	0.80	0.91	0.85	115
1.0	0.87	0.72	0.79	92
accuracy			0.83	207
macro avg	0.83	0.82	0.82	207
weighted avg	0.83	0.83	0.82	207

Figure 3: Classification Report from Unsupervised Preprocessing Method

4.4 Intrinsically semi-supervised method

4.4.1 Support Vector Machines

The objective of an SVM is to find a decision boundary that maximizes the margin, which is defined as the distance between the decision boundary and the data points closest to it.

The concept of semi-supervised SVMs, or S3VMs, is similar: we want to maximize the margin, and we want to correctly classify the labelled data. However, in the semi-supervised setting, an additional objective becomes relevant: we also want to minimize the number of unlabelled data points that violate the margin. Since the labels of the unlabelled data points are unknown, those that violate (i.e. lie within) the margin are penalized based on their distance to the closest margin boundary.

S3VMs were proposed by Vapnik (1998), who motivated the problem from a more transductive viewpoint: instead of optimizing only over the weight vector, bias and slack variables, he proposed to also optimize over the label predictions y^U . This formulation is equivalent to optimization problem, since any labelling yjU can only be optimal if, for each $y^i \quad y^U$, x_i is on the correct side of the decision boundary. Otherwise, a better solution could be obtained by simply inverting the labelling of x_i .

	precision	recall	f1-score	support
0	0.54	1.00	0.70	15
1	1.00	0.13	0.24	15
accuracy			0.57	30
macro avg	0.77	0.57	0.47	30
weighted avg	0.77	0.57	0.47	30

Figure 4: Classification Report from S3VM Method

5 Transductive Learning

5.1 Definition

Transduction is reasoning from observed, specific (training) cases to specific (test) cases.

In contrast to inductive learning, transductive learning techniques have observed all the data beforehand, both the training and testing datasets. We learn from the already observed training dataset and then predict the labels of the testing dataset. Even though we do not know the labels of the testing datasets, we can make use of the patterns and additional information present in this data during the learning process.

5.1.1 Label Propagation Algorithm

A graph is builded with the data samples as nodes.A weighted edge is put between each pair of samples. The closer the samples are, the higher the weight. Labels do not matter at this point.To get the label of an unlabeled sample,a random walk is started originating in this sample. One step of the walk consists of jumping from one node to an adjacent node randomly. Edges with a higher weight are chosen with higher probability.The probability that the random walk enters a blue node first is calculated. If it is larger than 50%,label the node blue, otherwise red.

Ex.Let us start in the lower white, unlabeled node. To proceed, we have to define probabilities for jumping to the other unlabeled node, one of the two blue nodes and the red node. An easy way to do this is by normalizing. There are four outgoing edges with the weights 1 (leading to the blue node), 0.25 (other blue node), 0.5 (other unlabeled node), and 0.5 (red node). So, for example, let us just define the probability to jump to the red node as $0.5/(1+0.25+0.5+0.5)=2/9$. Jumping to the closer blue node happens with a probability of $1/(1+0.25+0.5+0.5)=4/9$.

Using these probabilities, there is a lot of theory involved in how to compute

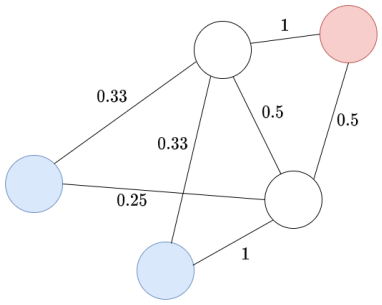

Figure 5: mixed samples

the probabilities in ending up in a blue or red node first. It can be done via Markov chains.

	Landing in a blue node first	Landing in a red node first
Starting in the top unlabeled node	46%	54%
Starting in the bottom unlabeled node	65%	35%

With this result, we can say that the upper unlabeled node could belong to the red class, while the bottom one should be blue.

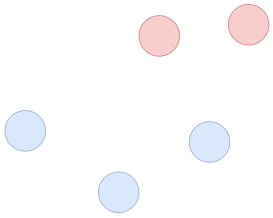

Figure 6: final samples

	precision	recall	f1-score	support
0.0	0.64	0.98	0.77	115
1.0	0.93	0.30	0.46	92
accuracy			0.68	207
macro avg	0.79	0.64	0.62	207
weighted avg	0.77	0.68	0.63	207

Figure 7: Classification Report from Label Propagation Method