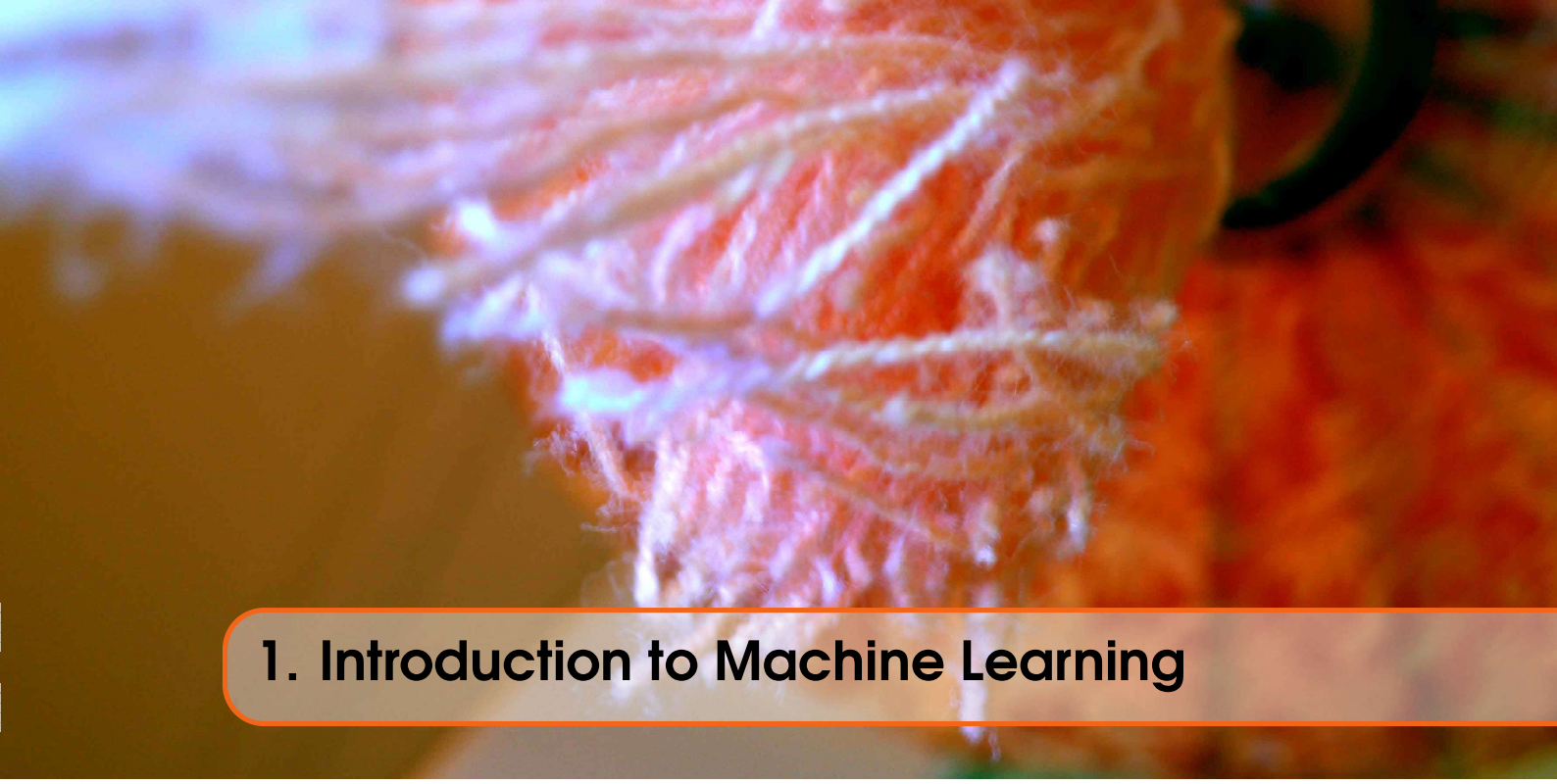

1. Introduction to Machine Learning

1.1 Introduction

We are living in a digitized age where information technology is attached to everything in our daily lives and bringing a revolution in the industry, healthcare, education, entertainment etc with the concept of Industry 4.0, Health4.0. A huge amount of data is generated from various applications, social media, IoT devices and financial markets. Plenty of data is being analyzed for the improvement in business, production, and services. Machine learning provides the methods of automated data analysis, predicting and forecasting the future and recommending the area and means of improvement. We can define machine learning as a set of algorithms, or methods that can automatically identify hidden patterns in data and make decisions for the unseen data under uncertainty.

The first machine learning application used by the regular user is detecting spam mail. The spam filter was introduced in 1990, and improved the lives of hundreds of millions of people. today, we are using and getting benefits of ML throughout the day. Typing these sentences in my text document uses ML to detect and correct the words and sentences. The application of ML includes Language translation, Optical Character Recognition(OCR), product recommendation, service recommendation, diagnosis of diseases, and many more.

1.2 Types of Machine Learning

Supervised Learning: Supervised learning means that training of the system is done under the supervision of an instructor. In supervised learning, the training data is fed to the system with the inputs and desired output called labels. From the observed data, the input-output pair, it find out the relationships between input and output. Output is determined for the unseen input using the relationship forecast the result. The output variable is called the target variable and the set of inputs or features are called predictors.

Unsupervised learning: In unsupervised learning, the system is trained without supervision, i.e. the training data is not labelled. Unsupervised learning algorithms create clusters from the given input set based on the characteristics of the inputs. It determines the cluster for the new observation

and makes decisions based on the behaviour of other observations in the same cluster. Anomaly detection is another example of the unsupervised learning. It studies the pattern of the observed data and notifies the user of the new observation if the pattern does not match.
Semi-supervised Learning: It is applicable for the dataset where a little bit of data is labelled and the remaining is unlabelled. Semi-supervised learning is the combination of supervised and unsupervised learning. It clusters the data using an unsupervised learning algorithm and if at least one instance is labelled in a cluster, the same label is assigned to all the instances of the same group.
Reinforcement learning: The reinforcement algorithm is completely different from the other types of machine learning algorithms. The reinforcement learning system is called an agent, which acts upon the environment to achieve a specific goal. The agent has to discover the actions that help the agent reach the goal or improve its utility value and the actions that reduce its utility value or put it far from the goal. Agent receives reward on getting close to goal or increasing utility value and receives punishment on performing actions that decrease the utility value. It learns through the trial and error search.

1.3 Model as a function

Model as a function: A predictor is a function that, when given a particular input example, produces an output. For simplicity we are considering here that the output is a single number i.e. a real number or scalar value.

$$f(x) : \mathbb{R}^{N\times d} \rightarrow \mathbb{R} \tag{1.1}$$

where x is the d dimensional input and the function f returns a real number.
The general case of all functions

$$f(x) = \beta_0 + \beta^T x \tag{1.2}$$

where β_0 and β are unknown. Training process approxiamte the the values of these unknown.

1.4 Model as Probability distribution

Input data can introduce noise, affecting the targeted output, but machine learning techniques can uncover the true relationship between input and output amidst this noise. To achieve this, it's essential to quantify the noise's impact. Given that noise is inherent in all data, predictions should include an element of uncertainty or a confidence level for the predicted values. Machine learning techniques adopt a probabilistic approach, treating the target variable as a distribution of possible functions rather than a single function.

$$P(y|x) = P(y|f(x)) = P(y|x,\beta) \tag{1.3}$$

1.5 Learrning in Machine Learning

Learning means adapting some specific skills through the acquisition of knowledge and experience for the improvement of performance at a specific task. According to Tom Michell[**mitchell1997introduction**] "A computer is said to learn from experience E with respect to some class of task T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E."
For example, a computer program checking grammar might improve its performance as a measure of its ability to detect and correct the grammatical mistakes made by the user. It learns through the experience of suggesting and acceptance of the error to the user. Any learning problem requires identifying these features of the problem: the class of task, the performance measure for the improvement, and some experience or knowledge. For a handwritten digit recognition system:

1. Task T: recognizing and classifying handwritten digits
2. Performance measure P: percentage of digits identified correctly
3. Training experience E: a collection of handwritten digits with label

For a learning problem in hand, task T is given, dataset in the form of experience E has to be collected, and one of the performance measure P has to choose from the set of defined performance measures. All that remains is to specify learning of the model. Learning is to find out the parameters or unknown or weights that the resulting model perform well for the unseen data.

1.6 Loss Function for Training

The concept of loss function is involved with the learning of a model. The concept of the loss function is just the opposite of the performance measure. The optimal value of the parameters minimizes the value of the loss function. A loss function $L(y,\hat{y})$ takes the ground truth label and the prediction of the model as input and produces a non-negative error that presents an error in prediction. The goal of the learning is to find out the parameters β that best fit the training data. The meaning of best fit is that these parameters minimize the loss function for the known observation. A commonly used loss function for the regression problem is the squared loss given by

$$\mathscr{L}(y,\hat{y}) = (y-\hat{y})^2 = (y-f(x))^2 \tag{1.4}$$

The average loss for the entire training data can be defined as

$$\begin{aligned}\mathscr{L}(y,\hat{y}) &= \frac{1}{N}\sum_{i=1}^{N}(y_i-\hat{y}_i)^2 = \frac{1}{N}\sum_{i=1}^{N}(y_i-f(x_i))^2 \\ &= \frac{1}{N}\sum_{i=1}^{N}(y_i-x_i^T\beta)^2\end{aligned} \tag{1.5}$$

The loss function can be considered as a function $\mathscr{L}(y,x,\beta)$ of input, output, and parameter β. One method of finding out the parameter is to take the gradient of the loss function, set it to zero, and solve the equation for β. This is known as the sum squared error. The optimal parameters β_{ML} is find out by minimizing the loss function and written as

$$\beta_{ML} = argmin_{\beta}\mathscr{L}(Y,X,\beta) \tag{1.6}$$

1.7 Information Theory

Information theory is a field of study that deals with the quantification, and communication of information over transmission. Here we will discuss the basic concept of information theory that requires understanding and measuring uncertainty in model performance.

1.7.1 Entropy

The entropy of a random variable X with probability p_x, is a measure of uncertainty is denoted by $H(X)$. For a discrete variable X with K states, that entropy is defined by

$$H(X) = -\sum_{k=1}^{K} P(X=k)\log P(X=k) \tag{1.7}$$

For a random variable $X \in 0,1$ with the probability distribution $p = 0.98,0.02$, the value of $H(x) = 0.08$ and for the probability distribution $p = 0.5,0.5$, the entropy is 1.0. For the first

example, there is a high probability of getting 0, uncertainty is very low, hence the entropy is very low. For later example, the probability of getting 0 and 1 are 0.5 and 0.5 respectively, uncertainty is high, therefore entropy value is 1.0. The entropy value is the maximum if $P(X=k)=1/K$. For the binary random variable $X \in 0,1$ with the probability distribution $\theta,(1-\theta)$, the entropy becomes

$$H(X) = -\theta \log \theta + (1-\theta)\log(1-\theta) \tag{1.8}$$

It is called binary entropy function and written as $H(\theta)$

1.7.2 KL-Divergence

Kullback-Leibler(KL) divergence or relative entropy measures dissimilarity between two probability distributions p and q. It also defines the number of bits required to convert data from one distribution to another. KL divergence has plenty of usefulness in data science including assessing datasets, and model performance.

$$KL(p||q) = \sum_{k=1}^{K} p_k \log \frac{p_k}{q_k} \tag{1.9}$$

$$KL(p||q) = \sum_{k} p_k \log p_k - \sum_{k} p_k \log q_k = -H(p) + H(p,q) \tag{1.10}$$

Where $H(p,q)$ is called cross-entropy, and

$$H(p,q) = -\sum_{k} p_k \log(q_k) \tag{1.11}$$

1.7.3 Mutual Information

Sometimes we are interested in knowing how much information of input vector will tell us about the target variable. We can calculate the correlation coefficient between inputs X and output Y, that provides a degree of association among variables and is not useful for categorical variables. Another approach is to find out the mutual information, determining how similar the joint distribution $P(X,Y)$ is to the factored distribution $P(X)P(Y)$. The mutual information can be defined as

$$I(X,Y) = KL(P(X,Y)||P(X)P(Y)) = \sum_{x}\sum_{y} P(x,y) \log \frac{p(x,y)}{p(x)p(y)} \tag{1.12}$$

1.8 Evaluation Metrics

The field of machine learning started its journey long back with statistical methods, simple algorithms and recently leveraging the architecture of deep learning networks. ML is used for widespread application for various purposes. As it is applied in the heterogeneous sector, the actual requirement is also varied. Various performance measure of ML techniques is developed to the needs of the application. Some fundamental performance measures for supervised and unsupervised problems will be discussed here.

1.8.1 Binary Classification Problem

In a binary classification problem, the goal is to predict whether an input vector belongs to the positive or negative class. Each predicted label can be categorized as one of four types[**rainio2024evaluation**]: TP, TN, FP, or FN. TP (True Positive) means the instance is positive and has been correctly predicted

as positive. TN (True Negative) indicates the instance is negative and has been correctly predicted as negative. FP (False Positive) occurs when a negative instance is incorrectly predicted as positive. FN (False Negative) happens when a positive instance is incorrectly predicted as negative. These counts of TP, FP, FN, and TN are organized in a 2x2 matrix called the confusion matrix.

The most commonly used evaluation metrics for binary classification are accuracy, sensitivity

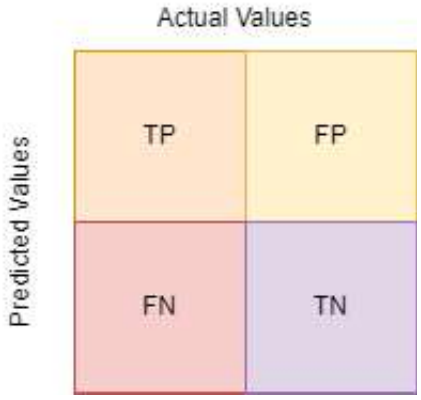

Figure 1.1: Confusion Matrix

(or recall), specificity, and precision. Accuracy measures the percentage of correctly classified instances out of the total instances and is defined as

$$Accuracy = \frac{TP+TN}{TP+FP+FN+TN} \tag{1.13}$$

Sensitivity or recall represents the percentage of positive instances classified correctly and is defined as

$$SensitivityorRecall = \frac{TP}{TP+FN} \tag{1.14}$$

Specificity represents the percentage of negative instances classified correctly and is defined as

$$Spe\,ficity = \frac{TN}{TN+FP} \tag{1.15}$$

Precision expresses the percentage of instances classified as positive and is defined as

$$Precision = \frac{TP}{TP+FP} \tag{1.16}$$

All the metrics are important for evaluating the performance of a model. However, sensitivity and specificity provide more insights into imbalanced datasets. In an imbalanced dataset, the distribution of the classes is not balanced; it may have more than 80% of the data from the positive class and the remaining data from the negative class, or vice versa.
The F1 score is defined as

$$F1-Score = \frac{2*Precision \times Recall}{precision+Recall} \tag{1.17}$$

A machine learning model provides the probability of belonging to the positive or negative class. A threshold value is used to determine the binary class. The default threshold is 0.5 for predicting the class of the target variable. If the probability is more than 0.5, it is considered a positive class instance; otherwise, it is considered a negative class instance. Another performance measure is the

receiver operating characteristic (ROC) curve. The ROC curve is obtained by plotting sensitivity against the false positive rate for all threshold values.
The area under the ROC curve, called the area under the curve (AUC), is another evaluation metric with values ranging from 0 to 1.
In multiclass classification problems, the model predicts the class label for an input instance from a set of $k \geq 3$ classes. The results of the classifier are presented in a confusion matrix format similar to binary classification. The meaning of the matrix elements is the same as in binary classification.

1.8.2 Evaluation Metrics for Regression

The efficiency or quality of a model is evaluated through different evaluation metrics or performance metrics. Performance metrics determine or measure how close the predicted values are to the actual values. Defining the performance metric is necessary before building and testing the models.
Most widely used performance measure for evaluationg prediction accuracy is Root Mean Squared Error (RMSE). RMSE is defined as

$$RMSE = \sqrt{\frac{1}{N}\sum_{t=1}^{N}(y_t - \hat{y}_t)} \tag{1.18}$$

MAPE is one of the most accepted measures for forecasting error([**kim2016new**]). MAPE is the average of absolute percentage error and is defined as

$$MAPE = \frac{1}{N}\sum_{t=1}^{N}\left|\frac{y_t - \widehat{y_t}}{y_t}\right| \tag{1.19}$$

Average error is calculated by taking the average of all the residuals where absolute value of each is consider so that negative and positive residuals do not cancel out. This is called the mean absolute error(MAE) and is defined as

$$MAE = \frac{1}{N}\sum_{t=1}^{N}|y_t - \widehat{y_t}| \tag{1.20}$$

The deviation of the estimated value from the actual value is the error, and how less the deviation occurs during forecasting is the measure of goodness. Here all the performance metrics calculate total deviation or error, and the lower value of these metrics indicates the potential of predicting.