



AET's
Atharva College of Engineering, Malad(W)
Approved by AICTE, New Delhi, DTE, Mumbai
Affiliated to University of Mumbai, ISO certified 9001:2015
Department of Information Technology
Academic Year: 2021-22

Webinar on “ Machine Learning ”

Date:	26th March 2022
Time:	3.30 pm TO 5.00 pm
Platform Used:	Google meet
Speaker details:	Mr. Bhavya Gada and Mr. Shivang Ghandhi
No. of attendants:	30+
Organized by:	INFT department
Faculties assigned:	Yogita Shelar, Pranoti Nage, Dhanashree Salvi B.

Objective: The Department of Information Technology organized a webinar on “**Machine Learning**”. The session enabled all the participants to understand many topics as listed below-

Steps in developing a Machine Learning Application

Building a machine learning application is an iterative process and follows a set of sequences. Below are the steps involved in for developing machine learning applications:

1. Problem framing

- This first step is to frame a machine learning problem in terms of what we want to predict and what kind of observation data we have to make those predictions. Predictions. be a yes/no label (binary classification) or a category (multiclass

classification) or a real number (regression). generally a label or a target answer: It may

2. Collect and clean the data

- Once we frame the problem and identify what kind of historical data we have for prediction modeling, the next step is to collect the data from a historical database or from open datasets or from any other data sources.
- Not all the collected data is useful for a machine learning application. We may need to clean the irrelevant data, which may affect the accuracy of prediction or may take additional computation without aiding in the result.

3. Prepare data for ML application

- Once the data ready for the machine learning algorithm, we need to transform the data in the form that the ML system can understand. Machines cannot understand an image or text. We need to convert it into numbers. It also requires building data pipeline depending on the machine learning application needs.

4. Feature engineering

- Sometimes a raw data may not reveal all the facts about the targeted label. Feature engineering is a technique to create additional features combining two or more existing features with an arithmetic operation that is more relevant and sensible.
- For example: In a compute engine, it is common for RAM and CPU usage to reach 95%, but something is messy when RAM usage is at 5% and CPU is at 93%. We can use a ration of RAM to CPU usage as a new feature, which may provide a better prediction. If we are using deep learning, it will automatically build features itself; we do not need explicit feature engineering.

5. Training a model

- Before we train the model, we need to split the data into training and evaluation sets, as we need to monitor how well a model generalizes to unseen data. Now, the algorithm will learn the pattern and mapping between the feature and the label.
- The learning can be linear or non-linear depending upon the activation function and algorithm. There are a few hyper parameters that affect the learning as well as training time such as learning rate, regularization, batch size, number of passes (epoch), optimization algorithm, and more.

6. Evaluating and improving model accuracy

- Accuracy is a measure to know how good or bad a model is doing on an unseen validation set. Based on the current learnings, we need to evaluate how a model is doing on a validation set. Depending on the application, we can use different accuracy metrics. For e.g. for classification we may use, precision and recall or F1 Score; for object detection, we may use IoU (intersection over union).
- If a model is not doing well, we may classify the problem in either of class 1) overfitting and 2) under-fitting.
- When a model is doing well on the training data, but not on the validation data, it is the over-fitting scenario. Somehow model is not generalizing well. The solution for the problem includes regularizing algorithm, decreasing input features, eliminating the redundant feature, and using resampling techniques like k-fold cross-validation.
- In the under-fitting scenario, a model does poor on both training and validation dataset. The solution to this may include training with more data, evaluating different algorithms or architectures, using more number passes, experimenting with learning rate or optimization algorithm.
- After an iterative training, the will learn a model to represent those labels from input data and this model can be used to predict on the unseen data.

Steps to Complete a Machine Learning Project

3. Use SMOTE synthetically generate sampling technique. It creates synthetic samples of your minority class to avoid overfitting.

4. An Adabooster to look at it. Learning from imbalanced Data.

Shivangkumar is presenting

Shivangkumar, Bhavya, Deepali, You, Apeksha, Nil 23 others

Steps to Complete a Machine Learning Project

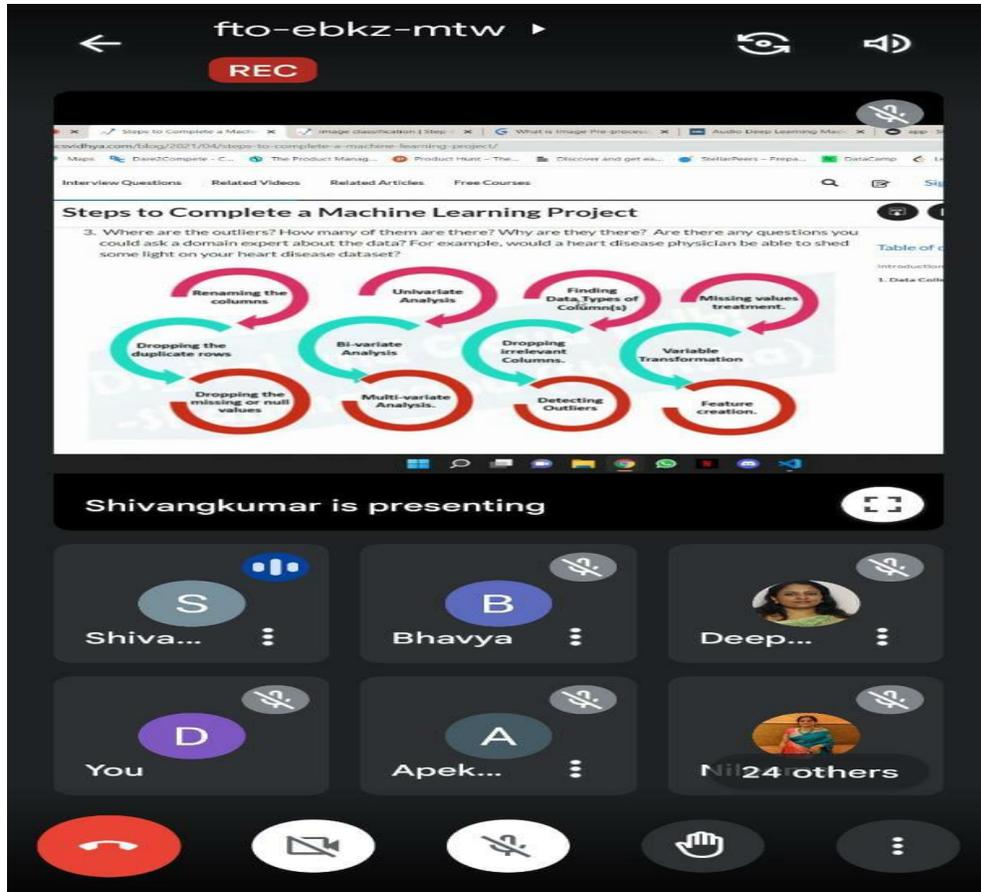
on data that isn't there!

1. **Single Imputation:** Fill with mean, a median of the column.
2. **Multiple Imputation:** Model other missing values and with what your model finds.
3. **KNN:** nearest neighbor. Fill data with a value of nearest neighbor that is closest.
4. Many more, such as random imputation, last observation carried forward (for time series), missing indicator, and more to go.

- **Feature engineering:** Creating values into numbers. A machine learning model require all values to be numerical.
- **One-hot encoding:** Turn all unique values into bits of 0s and 1s where the target value is 1 and the rest are 0s. For example, when a car colors green, red, blue, a car's color feature would be represented as [1, 0, 0] and red one would be [0, 1, 0].
- **Label Encoder:** Turn labels into distinct numerical values. For example, if your target variables are different animals, such as dog, cat, bird, these could become 0, 1, and 2, respectively.
- **Embedding encoding:** Learn a representation amongst all the different data points. For example, a language model it's a representation of how different words are related to each other. Embedding is also becoming more widely available for structured tabular data.
- **Feature normalization:** Scaling or standardization. When you're comparing variables, use an different scale (eg. number_of_bathrooms between 1 and 5 and cost_of_home between 500 and 300000), first, some machine learning algorithms don't perform very well. Scaling and standardization help in this.

Shivangkumar is presenting

Shivangkumar, Bhavya, Deepali, You, Apeksha, Nil 23 others



Prof. Deepali Maste
HOD
Dept. of Information Technology
ACE, Mumbai

Dr. S. P. Kallurkar
Principal
ACE, Mumbai