Suggested Teaching Guidelines for

Practical Machine Learning

PG-DBDA September 2023

Duration: 60 hours Theory and 60 hours Lab

Objective: Practicing Machine Learning Algorithms

Prerequisites: Good knowledge of Python Programming and Statistics

Evaluation method:
Theory exam– 40%
Lab Exam - 40%
Internal exam- 20%

List of Books / Other training material

Textbook:

1. Introduction to Machine Learning with Python - A Guide for Data Scientists, Muller Andreas / Shroff Publishers

Reference Book:

1. Machine Learning with R by Brett Lantz
2. Machine Learning for Big Data: Hands- On for Developer by Jasaon Bell, Wiley
3. Machine Learning: Hands-on for Developers and Technical Professionals
4. Machine Learning: A Bayesian and Optimization Perspective
5. Introduction to Machine Learning, Third Edition
6. R in Action, Robert Kabakoff

Note:

- PyTorch Framework should be taught in Lab Hours

Note: Each session having 2 Hours

Session 1 & 2:

- What is machine learning?
- Algorithm types of Machine learning
- Supervised and Unsupervised Learning
- Uses of Machine learning
- Evaluating ML techniques
- Introduction to Scikit Learn
- Performing ML using Scikit Learn

Assignments:

- Explore scikit learn Library.
- Explore Datasets Online (can refer Kaggle, UCI ML, etc.)
  a) Load dataset in google colab.
  b) Print first five values and last five values in dataset.
  c) check correlation between fields present in dataset

Session 3 & 4:

- Clustering
- Hierarchical Clustering &K means
- Distance Measure and Data Preparation – Scaling & Weighting
- Evaluation and Profiling of Clusters
- Hierarchical Clustering
- Clustering Case Study
- Principal Component analysis

Assignments:

- Download "mall_customers.csv" dataset from Kaggle.
  (a) Form n no. of clusters according to your observation.
  (b) Get wss value for each cluster.
  (c) find best K value

Session 5, 6 & 7:

- Decision Trees
- Classification and Regression Trees
- Random forest, Gradient boosting Machines, Model Stacking
- CAT Boost
- XG Boost

Assignments:

- Implement Random Forest, SVM, Logistic regression classification algorithm and check for classification report, f1 score for all three algorithms.

Session 8 & 9:

- Bayesian analysis and Naïve bayes classifier
- Assigning probabilities and calculating results
- Discriminant Analysis (Linear and Quadratic)
- K-Nearest Neighbors Algorithm

Assignments:

- Implement K-Nearest Neighbors Algorithm

Session 10

- Association rules mining
- Apriori

Assignments:

- Implement Food Product recommendation system for shop using Apriori.

Session 11 & 12:

- Linear Regression
- Logistic Regression
- Polynomial Regression
- Stepwise Regression
- Ridge Regression
- Lasso Regression

- Elastic Net Regression

Assignments:

- Download Dataset, perform linear, Ridge, Lasso, Polynomial regression and check for MAE, MSE, RMSE and also check F1 score and explain with conclusion.

Session 13:

- Support Vector Machines
- Basic classification principle of SVM
- Linear and Nonlinear classification (Polynomial and Radial)

Assignments:

- Download Air Quality Dataset from Kaggle Predict Air Quality Index using Linear regression and classify it into five categories using SVM (i.e. Very good, good, moderate, poor, worst)

Session 14 & 15:

- Moving average, Exponential Smoothing, Holt's Trend Methods, Holt-Winters' Methods for seasonality
- Autocorrelation (ACF & PACF), Auto-regression, Auto-regressive Models, Moving Average Models
- ARMA &ARIMA

Assignments:

- What is Auto correlation, explain its purpose Also download one data set and calculate Auto correlation.
- Explain ARMA and ARIMA model, what is purpose of this models in time series and Explain difference between them.

Session 16 & 17:

- ML in Real Time
- Algorithm Performance Metrics
- ROC and AOC
- Confusion Metrix
- F1 Score
- MSE and MAE

Assignments:

- Explain what is Confusion matrix, F1 score, MSE, RMSE, MAE and classification report.
- Load a dataset in Jupyter notebook and implement any classification algorithm and generate classification report and accuracy score.

Session 18:

- Recommendation Systems
  - Data Collection & Storage, Data Filtering
  - Collaborative Filtering
  - Factorization Methods
  - Evaluation Metrics: Recall, Precision, RMSE, Mean Reciprocal Rank, MAP at K, NDCG

Assignments:

- Download Movie dataset from kaggle and implement the movie recommendation system.

Session 19

- Anomaly detection
- Point, Contextual and Collective Anomaly
- Supervised and Unsupervised anomaly detection

Assignments:

- What is Anomaly detection? Explain its Purpose.

Session 20:

- Concept of Model Ensembling

Session 21 ,22 & 23:

- Introduction to Deep Learning
- Introduction to Tensor flow and Keras
- Introduction to Auto-encoders
- Neural Network and its applications
- Single layer neural Network
- Activation Functions: Sigmoid, Hyperbolic Tangent, ReLu
- Overview of Back propagation of errors

Assignments:

- Explore Tensor Flow and Keras Libraries.
- Implement Different Activation functions on datasets in Jupyter Notebook.

Session 24, 25 & 26:

Deep Learning Essentials

- Early Stopping for Preventing Overfitting
- Dropout
- Training Methods for Neural Network (High-Level Overviews only)
  - Update of weights with single training set element, Batch Training, Mini-batch Training, Stochastic Gradient Descent
  - Training Methods for Neural Network (High-Level Overviews only)
- Classic Backpropagation
- Momentum Backpropagation
- ADAM
  - L1 and L2 Regularization

Assignments:

- Implement L1 and L2 Regularization
- Implement Gradient Problems

Session 27 & 28

Convolutional Neural Network using PyTorch

- Introduction to PyTorch Framework
- Pytorch vs Tensor flow
- Convolutional Concept
- Inception Network
- Transfer Learning
- Data Augmentation
- Object Detection
- YOLO Algorithm (High-Level Overview)

Assignments:

- Install PyTorch. Explore the documentation of PyTorch Library.
- Implement YOLO Algorithm.

Session 29 & 30

Recurrent Neural Network (RNN) using Pytorch

- RNN Concept
- Types of RNNs
- Vanishing gradients with RNNs
- Gated Recurrent Unit (GRU) - (High-Level Overview only)
- Long Short-Term Memory (LSTM) - (High-Level Overview only)

Assignments:

- Implement RNN using PyTorch
- Implement LSTM and GRU

Case Studies:

- Real-time end-to-end practical ML project including front end with the deployment process.
- Cookie Cutter