Machine Learning using Python

Course Outline

Overview of the Training Course

This course provides a broad introduction to data science, machine learning, data mining, and statistical pattern recognition. It covers supervised and unsupervised learning, natural language processing, computer vision problems, time series analysis. You'll learn the essential theories, the interpretations of what comes out of machine learning execution and gain the practical know-how needed to quickly and powerfully apply these techniques to new problems in your own organization.

You will spend 40% time on theory lectures and demos and rest on lab exercises. Lab exercises will be based on problems hosted on platforms such as kaggle.com, UCI dataset.

Python programming language is used for lab exercises and demonstrations.

Duration: 4 Days

Prerequisite

- Strong programming knowledge in Python is required to perform the exercises during the class. If you want to brush up python knowledge, here is a quick tutorial.
- Basic knowledge of statistics, linear algebra is required to understand and implement some of the ML algorithms. You may revise the textbooks or online materials on these subjects.
- I strongly recommend taking one of the following online courses prior to the class.
  - Machine Learning by Andrew Ng available in coursera.org
  - Statistical Learning by Trevor Hastie and Rob Tibshirani available in online.stanford.edu
  - Analytics Edge available at edx.org

Day 1: Intro to Machine Learning and Data Preprocessing

Topic	Description

Introduction to Machine Learning	• What is machine learning, data science and it use cases
	• Supervised and Unsupervised Learning
	• Types of machine learning – choice of algorithm
	  ◦ Regression, Classification, Recommender
	  ◦ Parametric vs Non-parametric
	• Machine learning libraries

Introduction to Numpy and Scipy	• Fast numeric computation using numpy arrays • Linear Algebra – matrix multiplication, decompositions/factorization determinants • Common statistical functions using Scipy
Exploratory data analysis	• Descriptive statistics - mean, median, range, variance, standard deviation, variance, kurtosis, skewness, percentile • Correlation • Analysis of variance and F statistics • Chi-square test to test goodness of fit • Hypothesis tests

Introduction to Pandas	• Explore data in Dataframe - index, slice, filter, sort, join, aggregate and transfor
	• Handle Missing Data
	• Descriptive statistics
	  ◦ Correlation and covariance
	  ◦ Unique values, value count, distribution

Lab Exercises	• Analyzing the grouplens.org data using Pandas
Data Visualization (interactive session)	• Introduction to matplotlib and Seaborn • Types of chart, when to use which one • Interpretation from each type of chart • Subplots and overlay plots • Colors, styles, labels, legends

Feature Engineering (interactive session)	• Characteristics of good features • Handle missing values • Handle outliers • Handle categorical data • Feature scaling - standard scaler, normalizer (L1/L2), min-max scaler • Probability distribution of variables • Normality test for continuous variables • Power transformation • Polynomial transformation • Discretization • Sampling - down sampling, up sampling, stratified sampling • Feature selection based on significance • Multi-collinearity detection using correlation, variance inflation ratio etc. • Dimensionality reduction using PCA • Generate new features • Handle Time series data • Denormalization of structured data of RDBMS systems
Apache Spark for ML	• Features of apache Spark • Data structures in Spark • Dataframe DSL and Spark SQL • Spark integration with Pandas, Numpy, Scipy
Lab Exercise	• Explore ad-click prediction data from Outbrain

Learning Goals:

- Spot the use cases of machine learning
- Explore and transform data using numpy, pandas, Matplotlib
- Clean and prepare data for further processing

Day 2: Regression, Model Diagnostics and Tuning

Topic	Description
Linear Regression	Predict numeric outcomes with linear regression • Finding best fit lines with linear regression • Model evaluation metrics – R2, RMSE, MAE, F-stat • Shrinking coefficients to understand the data – Ridge regression, Lasso regression, Forward stage-wise regression, L1/L2 Penalty for regularization • Bias / variance trade off • Tree based regression (CART regression) • Understanding the cost function and minimize cost function using batch gradient descent algorithm • Variations of gradient algorithms - stochastic gradient descent, mini batch gradient descent for classification problems

Topic	Description
Lab Exercise	• Forecast insurance premium charges based on customer profile • Forecasting home price based on data from Kaggle.com challenge
Model Tuning and Deployment	• Combining transformers and estimators in a pipeline • Using k-fold cross validation to assess model performance • Debugging algorithms with learning and validation curve • Diagnosis bias and variance problems with learning curves • Fine tuning hyper parameters using grid search • Production readiness of ML application • Exposing ML model as web service
Lab Exercises	• Tune models built in the previous exercises

Learning Goals:

- Learn to predict numeric data (for example, revenue, price etc.)
- Learn to evaluate a regression model
- Learn to find important predictors for regression model
- Learn to validate and tune a regression model for best performance

Day 3: Classification and NLP

Topic	Description
Classification Algorithm	Predict categorical outcomes with classification • Logistic Regression for binary and multi class classification • Classification using decision trees, random forest • Classifying with KNN • Model evaluation using accuracy score, precision, recall, F-score, ROC plot, AUC, confusion matrix • Tune the hyper parameters using cross validation and grid search
Lab Exercise	• Predicting customer credit status customer profile • Find fraudulent transactions in credit card transactions • Reveal local attitudes from personal ads from craigslist.org RSS feed
Text Analytics	• Text analysis steps for topic modeling • Term Frequency – Inverse Term Frequency • POS tagging, lemmatization, stemming • Collecting raw text • Representing text • Categorize Documents by type • Determining sentiments • Gaining insights
Lab Exercise	• Working with IMDB comments dataset to analyze sentiments • StumbleUpon Evergreen classification Challenge • BBC Datasets - content based classification problem

Topic	Description
Boosting, Bagging and Ensemble	• What is boosting, bagging and pasting? • Putting multiple models in action • Building a hybrid model and compare performance metrics
Lab Exercise	• Build a boosted model for a classifier

Learning Goals:

- Learn to predict categorical data types, for example – customer segmentation, detection of email spam, stock market gainer or loser, fraud prediction
- Learn which metrics to use under what scenarios to measure performance
- Learn to find important predictor
- Evaluate and tune classification models
- Learning common tasks for text analytics

Day 4: Clustering and Neural Networks

Topic	Description
Clustering	• Working with unlabeled data – clustering analysis • K-means clustering • Hard (K-means) vs Soft clustering (Fuzzy c-means) • Find optimal number of clusters • Hierarchical Clustering
Lab Exercise	• Clustering users based social-media profile
Intro to Deep Learning	• What is Artificial Neural Network? • What is deep learning? • Introduction to Tensorflow and Keras • Optimization to gradient descent algorithms ◦ Batch vs mini-batch vs stochastic gradient descent ◦ Advanced initializers ◦ Momentum based methods ◦ Adaptive learning rate • Convolutional Neural Networks (CNN) and its use cases • Recurrent Neural Networks (RNN) and its use cases
Lab Exercises	• TensorFlow basics • Building regression model using TensorFlow • Building multi class classifier using TensorFlow • Classifier for image of hand-written digits (MINST) • Classifier for image of objects (CIFAR10)

Learning Goals:

- Learn to apply clustering to find natural groupings within a dataset
- Solve computer vision related problems

Appendix: Lab Setup

For lab exercises each student needs a laptop or a desktop with standard configuration 8 GB+ RAM and i5 CPU. Students will require administrative access to install software. For the lab exercise student will require to install the following software. Internet access is required for downloading test datasets, sample code, browse and documentations. Access to USB pen drive is required to copy materials from training instructor's machine.

- Anaconda distribution for python 3.6+ 64-bit version full version