BDBP-207 Machine Learning Laboratory - Notes & Excercises

Shyam Rajagopalan,
IBAB, Bengaluru

1 Jan 2025

In God we trust, all others bring data.

- William Edward Deming (1900-1993)

Lab 1 - Functions - 6 Jan 2025

Learning Goals

In this lab, you will learn linear and nonlinear functions and their derivatives.

By the end of this lab, you should be able to

1. Understand different types of functions that are useful for ML algorithms.
2. How exactly they are implemented in standard python libraries

Exercises

1. Implement A^TA - A = [1 2 3
   4 5 6]
2. Implement y = 2x + 3 and plot x, y [start=-100, stop=100, num=100]
3. Implement y = $2x^2$+3x+4 and plot x, y in the range [start=--10, stop=10, num=100]
4. Implement Gaussian PDF - mean = 0, sigma = 15 in the range[start=-100, stop=100, num=100]
5. Implement y = x^2, its derivative and plot both the function and derivative in the range

Lab 2 - ML model using scikit-learn - 7 Jan 2025

Learning Goals

In this lab, you will learn to use scikit-learn and build a machine learning model.

By the end of this lab, you should be able to

1. Learn basic usage of the scikit-learn module and ML model

Exercises

1. Implement california housing prediction model using scikit-learn - walkthro' of bdbp207_californiahousing.py
2. Complete the following tutorial
    a. https://inria.github.io/scikit-learn-mooc/python_scripts/datasets_california_housing.html

Lab 3 - Linear Regression - 13 Jan 2025

Learning Goals

In this lab, you will learn how to use scikit-learn and build a linear regression model.

By the end of this lab, you should be able to

1. Understand how to fit a linear model to a data
2. Learn many relevant functionalities in scikit-learn related to model preprocessing, training and evaluation.

Exercises

3. Implement a linear regression model using scikit-learn for the simulated dataset - simulated_data_multiple_linear_regression_for_ML.csv - to predict the disease score from multiple clinical parameters
4. Use the above simulated CSV file and implement the following from scratch in Python
    - Read simulated data csv file
    - Form x and y
    - Write a function to compute hypothesis
    - Write a function to compute the cost
    - Write a function to compute the derivative
    - Write update parameters logic in the main function

Lab 4 - Gradient Descent Implementation - 18 Jan 2025

Learning Goals

In this lab, you will develop a gradient descent algorithm from scratch using Python

By the end of this lab, you should be able to

1. Understand clearly how gradient descent algorithms work

Exercises

1. Implement gradient descent algorithm from scratch using Python
2. Use your implementation and train ML models for both californiahousing and simulated datasets and compare your results with the scikit-learn models.
3. Implement normal equations method from scratch and compare your results on a simulated dataset (disease score fluctuation as target) and the admissions dataset (https://www.kaggle.com/code/erkanhatipoglu/linear-regression-using-the-normal-equation). You can compare the results with scikit-learn and your own gradient descent implementation.
4. Plot the data points and the obtained regression line from all three approaches and compare the outcome.

Lab 5 - Logistic Regression - 20 Jan 2025

Learning Goals

In this lab, you will implement a logistic regression classifier using scikit-learn

By the end of this lab, you should be able to

1. Understand clearly how stochastic gradient descent algorithms work
2. How to implement a ML classifier using logistic regression
3. How to handle tabular datasets for ML model development

Exercises

1. Implement Stochastic Gradient Descent algorithm from scratch
2. Implement sigmoid function in python and visualize it
3. Compute the derivative of a sigmoid function and visualize it
4. Implement logistic regression using scikit-learn for the breast cancer dataset - https://www.kaggle.com/datasets/uciml/breast-cancer-wisconsin-data

Lab 6 - k-fold Cross Validation & Model Selection - 21 Jan 2025

Learning Goals

In this lab, you will implement k-fold CV from scratch

By the end of this lab, you should be able to

1. Understand the ML training process
2. The importance of k-fold cross validation
3. Model selection

Exercises

1. K-fold cross validation. Implement for K = 10. Implement from scratch, then, use scikit-learn methods.
2. Data normalization - scale the values between 0 and 1. Implement code from scratch.
4. Data standardization - scale the values such that mean of new dist = 0 and sd = 1. Implement code from scratch.
5. Use validation set to do feature and model selection.

Lab 7 - Data preprocessing - 27 Jan 2025

Learning Goals

In this lab, you will understand preprocessing and mechanism to train a ML model.

By the end of this lab, you should be able to

1. The importance of data processing

Exercises

1. Perform 10-fold cross validation for SONAR dataset in scikit-learn using logistic regression. SONAR dataset is a binary classification problem with target variables as Metal or Rock. i.e. signals are from metal or rock.
2. Compute SONAR classification results with and without data pre-processing (data normalization). Perform data pre-processing with your implementation and with scikit-learn methods and compare the results.

Lab 8 - Regularization - 28 Jan 2025

Learning Goals

In this lab, you will understand L1 and L2-regularization and encoding mechanisms needed to handle categorical data.

By the end of this lab, you should be able to

1. Understand how regularization helps in avoiding overfitting
2. Understand the methods to handle categorical data
3. Scikit-learn methods for encoding

Exercises

1. Implement L2-norm and L1-norm from scratch
2. Build a classification model for wisconsin dataset using Ridge and Lasso classifier using scikit-learn
4. Implement ordinal encoding and one-hot encoding methods in Python from scratch.
5. Use breast_cancer.csv (https://raw.githubusercontent.com/jbrownlee/Datasets/master/breast-cancer.csv) and use scikit learn methods, OrdinalEncoder, OneHotEncoder(sparse=False), LabelEncoder to implement complete Logistic Regression Model.
   Good reference:
   https://machinelearningmastery.com/one-hot-encoding-for-categorical-data/

Lab 9 - Decision Trees - 3 Feb 2025

Learning Goals

In this lab, you will implement the decision tree algorithm.

By the end of this lab, you should be able to

1. Implement a decision tree regressor using scikit-learn
2. Implement a decision tree classifier using scikit-learn

Exercises

1. Write a program to partition a dataset (simulated data for regression) into two parts, based on a feature (BP) and for a threshold, t = 80. Generate additional two partitioned datasets based on different threshold values of t = [78, 82].
2. Implement a regression decision tree algorithm using scikit-learn for the simulated dataset.
3. Implement a classification decision tree algorithm using scikit-learn for the simulated dataset.

Lab 10 - Decision Tree Components - 4 Feb 2025

Learning Goals

In this lab, you will understand components of the decision tree algorithm.

By the end of this lab, you should be able to

1. Implement a different set of functions needed for the decision tree algorithm

Exercises

1. Implement entropy measure using Python. The function should accept a set of data points and their class labels and return the entropy value.
2. Implement information gain measures. The function should accept data points for parents, data points for both children and return an information gain value.

Lab 11 - Decision Tree Classifier from scratch - 10 Feb 2025

Learning Goals

In this lab, you will implement the decision tree classification algorithm without using any library.

By the end of this lab, you should be able to

1. Implement decision tree classification algorithm in Python

Exercises

1. Implement decision tree classifier without using scikit-learn using the iris dataset. Fetch the iris dataset from scikit-learn library.

Lab 12 - Decision Tree Regressor from scratch - 11 Feb 2025

Learning Goals

In this lab, you will implement the decision tree regression algorithm without using any library.

By the end of this lab, you should be able to

1. Implement decision tree regression algorithm in Python

Exercises

1. Implement a decision regression tree algorithm without using scikit-learn using the diabetes dataset. Fetch the dataset from scikit-learn library.

Lab 13 - Bagging & Random Forest - 17 Feb 2025

Learning Goals

In this lab, you will implement the bagging tree algorithm using scikit-learn.

By the end of this lab, you should be able to

1. Implement bagging tree algorithm using scikit-learn

Exercises

1. Implement bagging regressor and classifier using scikit-learn. Use diabetes and iris datasets.
2. Implement bagging regressor without using scikit-learn
3. Implement Random Forest algorithm for regression and classification using scikit-learn. Use diabetes and iris datasets.

Lab 14 - AdaBoost - 18 Feb 2025

Learning Goals

In this lab, you will implement the AdaBoost tree algorithm with and without using scikit-learn.

By the end of this lab, you should be able to

1. Understand the inner workings of AdaBoost in detail.

Exercises

1. Implement Adaboost classifier using scikit-learn. Use the Iris dataset.
2. Implement Adaboost classifier without using scikit-learn. Use the Iris dataset.

Lab 15 - Gradient Boost - 3 Mar 2025

Learning Goals

In this lab, you will implement the Gradient Boost tree algorithm using scikit-learn. Also, implement some of the components of the XGBoost algorithm from scratch.

By the end of this lab, you should be able to

1. Understand the components of XGBoost in detail.

2. Understanding of the XGBoost algorithm in detail.

Exercises

1. Implement Gradient Boost Regression and Classification using scikit-learn. Use the Boston housing dataset from the ISLP package for the regression problem and weekly dataset from the ISLP package and use Direction as the target variable for the classification.
2. Write a Python program to compute the similarity score, gain and output values used in XGBoost algorithm. Plot similarity scores vs lambda (regularization parameter) for different values of lambda (0, 0.1, 0.5, 1) and observe how similarity scores vary with lambda.

Lab 16 - XGBoost - 4 Mar 2025

Learning Goals

In this lab, you will implement the XGBoost tree algorithm using scikit-learn.

By the end of this lab, you should be able to

1. Fully understand the ensemble learning process
2. Understanding of the XGBoost algorithm in detail.

Exercises

1. Write a Python program to aggregate predictions from multiple trees to output a final prediction for a regression problem.
2. Write a Python program to compute the residual values. Assume three trees are constructed.
3. Implement XGBoost classifier and regressor using scikit-learn

Lab 17 - Kernel Methods - 10 Mar 2025

Learning Goals

In this lab, you will develop programs to understand components involved in Kernel transformation.

By the end of this lab, you should be able to

1. Understand data transformations to higher dimensional space
2. Perform operations on the higher dimensional space

Exercises

1. Project discussions and getting started
2. Implement a feature mapping function called Transform()

$\phi : \mathbb{R}^2 \rightarrow \mathbb{R}^3$ defined by:

$$\phi(x_1, x_2) = (x_1^2, \sqrt{2}x_1x_2, x_2^2)$$

Use following set of samples

x1	x2	Label
1	13	Blue
1	18	Blue
2	9	Blue
3	6	Blue
6	3	Blue
9	2	Blue
13	1	Blue
18	1	Blue
3	15	Red
6	6	Red
6	11	Red
9	5	Red
10	10	Red
11	5	Red
12	6	Red
16	3	Red

Plot these points. Then transform these points using your “Transform” function into 3-dim space. Plot the points and manipulate the points so that you can see a separating plane in 3D.

1. Let x1 = [3, 6], x2 = [10, 10]. Use the above “Transform” function to transform these vectors to a higher dimension and compute the dot product in a higher dimension. Print the value.
2. Implement a polynomial kernel K(a,b) = a[0]**2 * b[0]**2 + 2*a[0]*b[0]*a[1]*b[1] + a[1]**2 * b[1]**2 . Apply this kernel function and evaluate the output for the same x1 and x2 values. Notice that the result is the same in both scenarios demonstrating the power of kernel trick.

3. Try this tutorial for plotting decision boundaries - https://scikit-learn.org/stable/auto_examples/svm/plot_svm_kernels.html

Lab 18 - RBF Kernel and SVM - 11 Mar 2025

Learning Goals

In this lab, you will implement kernel methods and support vector machine algorithms.

By the end of this lab, you should be able to

1. Understand the effectiveness of RBF Kernel
2. Build a model using SVM

Exercises

1. Consider the following dataset. Implement the RBF kernel. Check if RBF kernel separates the data well and compare it with the Polynomial Kernel.

x1	x2	Label
6	5	Blue
6	9	Blue
8	6	Red
8	8	Red
8	10	Red
9	2	Blue
9	5	Red
10	10	Red
10	13	Blue

11	5	Red
11	8	Red
12	6	Red
12	11	Blue
13	4	Blue
14	8	Blue

3. Try classifying classes 1 and 2 from the iris dataset with SVMs, with the 2 first features. Leave out 10% of each class and test prediction performance on these observations. https://scikit-learn.org/stable/tutorial/statistical_inference/supervised_learning.html#supervised-learning-tut - Check the solution code to learn about various plots.
4. Implement twitter sentiment prediction using SVM - Try different kernel functions and compare the results. https://www.kaggle.com/code/langkilde/linear-svm-classification-of-sentiment-in-tweets/notebook

Lab 19 - Evaluation Metrics - 17 Mar 2025

Learning Goals

In this lab, you will learn about various performance metrics used to evaluate model performance.

By the end of this lab, you should be able to

1. Know the differences between different metrics
2. Get a thorough understanding of when to use these metrics

Exercises

For the heart.csv dataset, build a logistic regression classifier to predict the risk of heart disease. Vary the threshold to generate multiple confusion matrices. Implement a python code to calculate the following metrics

- Accuracy
- Precision
- Sensitivity
- Specificity
- F1-score
- Plot the ROC curve

- AUC

Lab 20 - Unsupervised Learning - PCA and Clustering - 18 Mar 2025

Learning Goals

In this lab, you will understand unsupervised learning methods.

By the end of this lab, you should be able to

1. Understand the dimensionality reduction and PCA
2. Understand k-means and hierarchical clustering algorithms

Exercises

1. Complete Lab exercises, Sec 12.5 Unsupervised Learning - PCA and Clustering - from the ISLP book. No need for Matrix Completion exercise. However, read that offline to understand SVD in action.
2. Work on your project

Lab 21 - K-Means Algorithm - 24 Mar 2025

Learning Goals

In this lab, you will implement K-means clustering algorithm from scratch

By the end of this lab, you should be able to

1. Understanding the k-means algorithm in detail.

Exercises

1. Implement K-Means algorithm ground-up using Python
2. Work on your project

Lab 22 - Hierarchical Clustering - 25 Mar 2025

Learning Goals

In this lab, you will implement the hierarchical clustering algorithm

By the end of this lab, you should be able to

1. Understand the hierarchical clustering in detail.

Exercises

2. Work on NCI data - build classification model after reducing the gene expression features using hierarchical clustering. Compare this with the PCA approach
3. Work on your project

Lab 23 - Generative Models - 1 Apr 2025

Learning Goals

In this lab, you will implement a generative model and use it for prediction

By the end of this lab, you should be able to

1. Understand the discriminatory and generative models better
2. Implement generative models

Exercises

1. Develop prediction model for Iris.csv using joint probability distribution approach
    a. Use only the first two features, SepalLengthCm, SepalWidthCm and the target variable
    b. Add random noise to the features
    c. Discretize the feature values
    d. Build a decision tree model with max_depth = 2, then, compare the accuracy of this model with the joint probability distribution method
2. Work on your project

Lab 24 - Bayesian Learning - 5 Apr 2025

Learning Goals

In this lab, you will implement a Bayesian learning algorithm

By the end of this lab, you should be able to

1. Do inference on bayesian networks
2. Develop bayesian learning model

Exercises

1. Class exercises on Bayesian Network Inference and Bayesian Learning
2. Implement Naive Bayes classifier for spam detection using scikit-learn library - use the dataset from https://www.kaggle.com/datasets/vishakhdapat/sms-spam-detection-dataset/data

Lab 23 - IA2 Exam - 7 Apr 2025

1. Internal Assessment 2 Exam

Lab 24 - Multiclass classification - 8 Apr 2025

Learning Goals

In this lab, you will implement a real-world image classifier involving multiple classes.

By the end of this lab, you should be able to

1. Understand to handle image modality
2. Develop a real-world classifiers involving multiple classes

Exercises

3. Use CIFAR10 dataset and develop a ML model for image classification using kNN
2. Work on your project

Lab 25 - Explainable AI - 19 Apr 2025

Learning Goals

In this lab, you will use SHAP for explaining ML models

By the end of this lab, you should be able to

1. Use the SHAP tool and generate different plots
2. Interpret models

Exercises

3. Learn XAI - SHAP. Use SHAP in one of your models and plot beeswam plot. Study the influencing features. - https://shap.readthedocs.io/en/latest/

Lab 26 - Linear Algebra - 21 Apr 2025

Learning Goals

In this lab, you will practice more linear algebra problems needed for optimization

By the end of this lab, you should be able to

1. Clearly understand advanced linear algebra concepts

Exercises

1. Can you tell whether the matrix, A = [9 -15; -15 21] is positive definite?
2. Find the eigenvalues of the given Hessian at the given point
    a. [12x^2 -1; -1 2] at (3, 1)
4. Determine the concavity of
    a. f(x, y) = x^3 + 2y^3 -xy at (i) (0,0, (ii) (3, 3), (iii) (3, -3)
5. f(x, y) = 4x + 2y - x^2 - 3y^2
    a. Find the gradient. Use that to find critical points, (x, y) that makes gradient 0
    b. Use the Eigenvalues of the Hessian at the point to determine whether the critical point is a minimum, maximum or neither

Lab 27 - Real world case studies - 22 Apr 2025

Learning Goals

In this lab, you will develop ML models on biological datasets

By the end of this lab, you should be able to

1. Process biological data for ML model development

Exercises

1. Use the following datasets
    a. https://archive.ics.uci.edu/dataset/401/gene+expression+cancer+rna+seq
    b. https://www.kaggle.com/code/singhakash/dna-sequencing-with-machine-learning
2. Write python programs to extract features from DNA and RNA seq data.
3. Develop ML classifiers using above features utilizing DNA and RNA seq datasets.

Lab 27 - Optimization Theory - 28 Apr 2025

Learning Goals

In this lab, you will learn Maximum Likelihood Estimation.

By the end of this lab, you should be able to

1. Learn methods you will have a understanding of defining log likelihoods and optimize them.

Exercises

1. Simulate a dataset of 1000 points from a Normal distribution with mu=10, sd=3. Write a log-likelihood function and optimize it to find the mu and sigma.
2. Work on a project

Lab 28 - Project Implementation - 29 Apr 2025

Learning Goals

In this lab, you will learn to present your scientific work to larger audiences

By the end of this lab, you should be able to

1. Stitch various components of a model development
2. Present your research to your peers

Exercises

1. Complete the ML development for your project.
2. Project presentations

========== End ==========