**Python\_Lesson4: Python Programming**

**Lesson Overview:**

In this lesson we will introduce classification.

b. Classification algorithm

c. Scikit learn

d. Advanced concept related to machine learning algorithm like overfitting, underfitting, cross validation, evaluation for clustering methods

**Use Case Description:**

k-nearest neighbor classifier

**Programming elements:**

Classification

**Source Code:**

<https://umkc.box.com/s/cra3wkg3y7c4yolhrler01tq5d42mc1j>

**Data Set:**

**URL:** <https://umkc.box.com/s/ea6wn1cidukan67t02j60nmp1ljln3kd>

**Dataset description:** <https://www.kaggle.com/uciml/glass/downloads/glass-classification.zip/1>

The name of target Column is **Type**

**In class programming:**

1. find the correlation between ‘survived’ (target column) and ‘sex’ column for the Titanic use case in class.

Do you think we should keep this feature?

2. Implement Naïve Bayes method using scikit-learn library

Use dataset available in <https://umkc.box.com/s/ea6wn1cidukan67t02j60nmp1ljln3kd>

Use **train\_test\_split** to create training and testing part

Evaluate the model on testing part using score and

classification\_report(y\_true, y\_pred)

3. Implement linear SVM method using scikit library

Use the same dataset above

Use **train\_test\_split** to create training and testing part

Evaluate the model on testing part using score and

classification\_report(y\_true, y\_pred)

Which algorithm you got better accuracy? Can you justify why?

**ICP Submission Guidelines (for In Class students):**

1. ICP Submission is in pairs of two students.

2. Once completed, must be presented to TA or Instructor before the completion of the class

3. Submission after class is considered as a late submission. (Check the late submission policy in the syllabus)

4. ICP Code with brief explanation should be pushed to GitHub. Submit GitHub link through the Feedback Form: <https://docs.google.com/forms/d/e/1FAIpQLSdmJkDgBMxr4qv73c9y5k1jtky44-sMmOI1v1jFtNEbUJ6H9A/viewform>

**Online Submission Guidelines (for Online students):**

1. Submit your source code and documentation to GitHub and represent the work through wiki page properly (submit your screenshots as well. The screenshot should have both the code and the output)

2. Comment your code appropriately

3. Video Submission (2 – 3 min video showing the demo of the ICP, with brief voice over on the code explanation)

4. Submission after class is considered as a late submission. (Check the late submission policy in the syllabus)

5. Use the following Google link to submit your ICP # (GitHub wiki page link for ICP #): https://docs.google.com/forms/d/e/1FAIpQLSdmJkDgBMxr4qv73c9y5k1jtky44-sMmOI1v1jFtNEbUJ6H9A/viewform

**Evaluation Criteria:**

1. Completeness of Features

2. Code Quality (<https://en.wikipedia.org/wiki/Best_coding_practices>)

3. Time

4. Feedback Submission

**Note:** *Cheating, plagiarism, disruptive behavior and other forms of unacceptable conduct are subject to strong sanctions in accordance with university policy. See detailed description of university policy at the following URL:* [*https://catalog.umkc.edu/special-notices/academic-honesty/*](https://catalog.umkc.edu/special-notices/academic-honesty/)