

Introduction to Machine Learning in Engineering Science

National Cheng Kung University

Department of Engineering Science

Instructor: Chi-Hua Yu

Lab 1

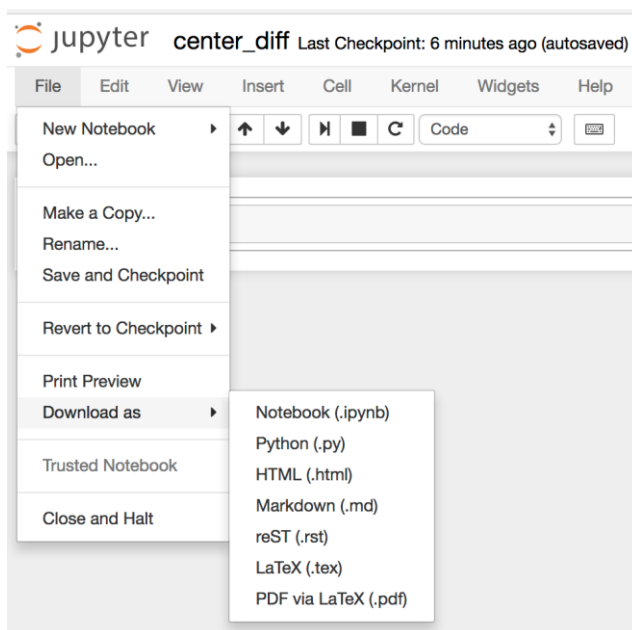
Programming, Due 21:00, Thursday, October 7th, 2021

Submission within 24 hours.

Late submission before post of solution: score*0.8 (the solution will usually be posted within a week); no late submission after the post of solution

Lab Submission Procedure (請仔細閱讀)

1. You should submit your Jupyter notebook and Python script (*.py, in Jupyter, click File, Download as, Python (*.py)).



2. Name a folder using your student id and lab number (e.g., n96081494_lab1), put all the python scripts into the folder and zip the folder (e.g., n96081494_lab 1.zip).
3. Submit your lab directly through the course website.

Total 120%

1. **(20%)** Name your Jupyter notebook `cm2ftin_table` and Python script `cm2ftin_table.py`. Write a Python program that prints out a table with centimeter 145, 150, 155, ..., 210 in the first column and the corresponding feet and inches in the second column. Below is a sample output:

```
cm  ft  in
145, 4.0, 9.086614114999996
150, 4.0, 11.055118049999997
```

Introduction to Machine Learning in Engineering Science

National Cheng Kung University

Department of Engineering Science

Instructor: Chi-Hua Yu

```
155, 5.0, 1.02362198499999983
160, 5.0, 2.99212591999999994
165, 5.0, 4.9606298549999993
170, 5.0, 6.92913378999999945
175, 5.0, 8.8976377249999996
180, 5.0, 10.8661416599999997
185, 6.0, 0.83464559499999978
190, 6.0, 2.8031495299999999
195, 6.0, 4.771653465
200, 6.0, 6.7401574000000001
205, 6.0, 8.7086613350000002
210, 6.0, 10.6771652700000003
```

2. (20%) Name your Jupyter notebook `sin_approx` and Python script `sin_approx.py`. Using a `while` loop to implement an approximation of sine function with polynomial:

$$\sin(x) \approx x - \frac{x^3}{3!} + \frac{x^5}{5!} - \frac{x^7}{7!} + \dots$$

Compute the right-hand side for powers up to $N = 25$. Below is a sample output:

```
sin(1.6) approximation is 0.9995736030415051
```

3. (20%) Name your Jupyter notebook `dna_seq` and Python script `dna_seq.py`. A DNA sequence contains four base letters A, C, G, T. Genomes of different species vary with respect to the proportion of Gs and Cs in their DNA as opposed to Ts and As.

Write a function `gc_content` that returns the GC content of a given DNA sequence represented as a string. Use the following simple script and sample outputs to test your function:

```
In [2]: 1 seq50 = 'GGAACCTT'
        2 gc_content(seq50)
```

```
Out[2]: 0.5
```

```
In [5]: 1 seq75 = 'GCGCGCATTA'
        2 gc_content(seq75)
```

```
Out[5]: 0.6
```

```
In [8]: 1 seq60 = 'AGCAGCCGCT'
        2 gc_content(seq60)
```

```
Out[8]: 0.7
```

4. Bank of LAiMM wants to predict who are possible defaulters for the consumer loans product. They have data about historic customer behavior based on what they have observed. Hence when they acquire new customers, they want to predict who is riskier and who is not. You are required to use the training dataset to identify patterns that predict “potential” defaulters. Name your Jupyter notebook `SVM` and report `SVM_report.docx`.

Introduction to Machine Learning in Engineering Science

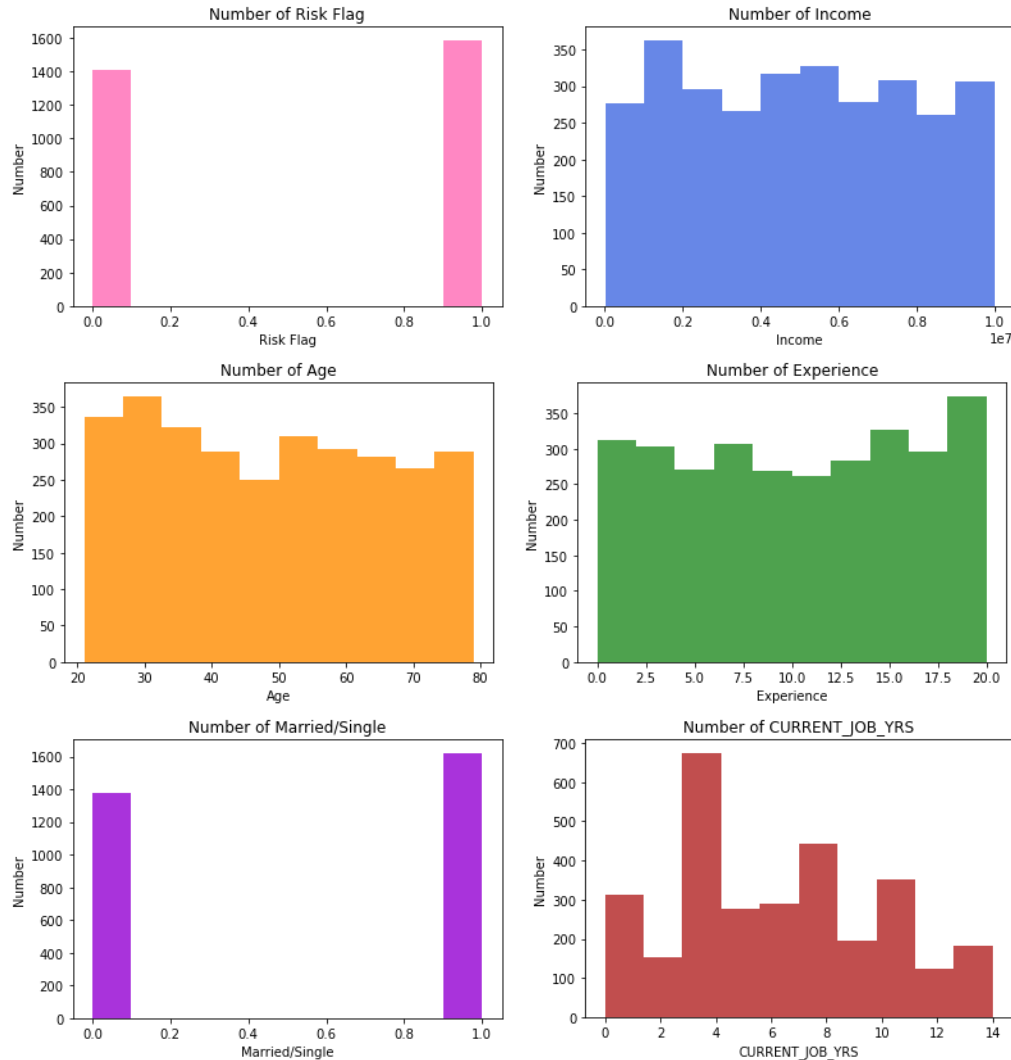
National Cheng Kung University

Department of Engineering Science

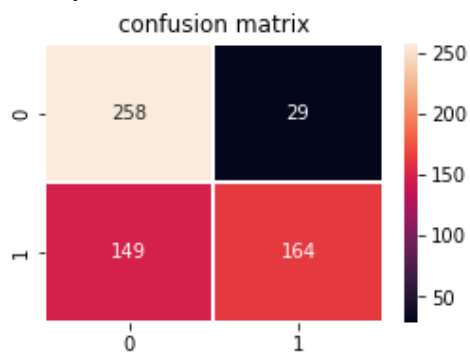
Instructor: Chi-Hua Yu

- (a) (20%) Plot the statics of training dataset. You may need to process training data before feeding them into model.

This figure is an example, please plot the statistics of training data that are fed into the model.



- (b) (40%) Use Support Vector Machine to identify possible defaulters, please report training history, confusion matrix and accuracy.



accuracy score: 0.7033