BÁO CÁO THỰC HÀNH

MÔN: Phương Pháp Học Máy Trong An Toàn Thông Tin

LAB 1: Setting Up Your Machine Learning For Cybersecurity Arsenal

1. THÔNG TIN CHUNG:

Lớp: NT522.N11.ATCL

STT	Họ và tên	MSSV	Email
1	Trần Hoàng Khang	19521671	19521671@gm.uit.edu.vn
2	Lê Hồng Bằng	19520396	19520396@gm.uit.edu.vn
3	Nguyễn Tú Ngọc	20521665	20521665@gm.uit.edu.vn

2. NỘI DUNG THỰC HIỆN:

STT	Công việc	Kết quả tự đánh giá
1	Câu hỏi 1	100%
2	Câu hỏi 2	100%
3	Câu hỏi 3	100%
4	Câu hỏi 4	100%
5	Câu hỏi 5	100%
6	Câu hỏi 6	100%
7	Câu hỏi 7	100%
8	Câu hỏi 8	100%

BÁO CÁO CHI TIẾT

1. *Sinh viên cho ví vụ về phép cộng, trừ hai ma trận numpy.*

Xem chi tiết tại file Notebook (.ipynb)

2. *Sinh viên sử dụng pandas xử lý các yêu cầu sau:*

- Đọc CSV thành Dataframe và hiển thị

- Hãy chuyển index mặc định thành giá trị cột id

- Sắp xếp dữ liễu theo nhiều cột (sort)

- Chọn một cột cụ thể và hiển thị nó

- Chọn 2 hàng đầu tiền và hiển thị chúng

- Hãy chọ một hàng dựa trên một điều kiện giá trị của cột

- Thay đổi một vài giá trị thành NaN ở CSV, sau đó đọc lên thành Dataframe và thay thế chúng bằng giá trị 0

- Ở cột Z chuyển giá trị lớn hơn 90 là True và nhỏ hơn là False trong Dataframe

- Chuyển Dataframe trên thành 2 Dataframe d1 và d2; d1 chứ cột X và Y, d2 chứa cột Z; cuối cùng d3 là thành quả của nối 2 Dataframe d1 và d2

- Dùng tính năng thống kê hãy hiển thị kết quả thông kể các giá trị thuộc tính của Dataframe

Xem chi tiết tại file Notebook (.ipynb)

3. *Sinh viên tự tìm hiểu thực hiện lại ví dụ dùng mô hình Linear Regression trong thư viện scikit-learning bằng các thư viện sáu:*

- TensorFlow

- Keras

- PyTorch

Cho biết cảm nghĩ về việc dùng 4 thư viện này

Cảm nghĩ cá nhân *khi sử dụng các thư viện*

	Scikit-learning	**Keras**	**Pytorch**
Tổng quan	Một thư viện học máy nói chung, cung cấp các thuật toán cơ bản	Keras là một khung học sâu cấp cao hơn, nó tóm tắt nhiều chi tiết, làm cho mã trở nên đơn giản và ngắn gọn hơn so với trong PyTorch hoặc TensorFlow,	Một thư viện hỗ trợ nhiều phương tiện liên quan cho Deep Learning
Tính năng đáng chú ý	Cung cấp các thuật toán chuyên về học máy Decision Tree, Logistic Regression , … etc.	Cung cấp các API nhất quán và đơn giản, nó giảm thiểu số lượng hành động của users cần thiết cho các trường hợp sử dụng phổ biến	• **Autograd** - một thuật toán có thể tự động tính toán độ dốc của các hàm của bạn, được xác định theo các hoạt động cơ bản • Các quy trình tối ưu hóa dựa trên **Gradient** để tối ưu hóa quy mô lớn, dành riêng cho tối ưu hóa mạng thần kinh
Tốc độ		Keras chậm hơn so với Pytorch	Pytorch có tốc độ thực thi cao hơn, phù hợp cho hiệu suất cao
Dataset size		Bởi lý do trên nên Keras phù hợp cho dataset nhỏ	Pytorch có thể vận hành trên dataset lớn
Case sử dụng	Chủ yếu khi cần sử dụng các thuật toán ***Machine Learning*** truyền thống	• Phù hợp khi sử dụng ***Deep Learning***, cả 2 thư viện đều update các mô hình Neuron Network hiện đại nhất khá tốt và nhanh. • Tùy vào từng ngữ cảnh và một số tiêu chí đánh giá để chọn lọc thư viện (xem tại đây), không có thư viện nào hoàn toàn tốt hơn	

Nguồn tham khảo (*từ ý kiến cộng đồng*):

Keras vs PyTorch - GeeksforGeeks

Pytorch Vs Tensorflow Vs Keras: Here are the Difference You Should Know (simplilearn.com)

4. *Sinh viên hoàn thành code phát hiện spam với* ***SVMs*** *và* ***Linear regression***

Xem chi tiết tại file Notebook (.ipynb)

5. *Sinh viên cho biết chức năng của phương thức* ***genfromtxt()*** trong thư viện numpy

Hàm **genfromtxt()** tải dữ liệu từ tệp văn bản, xử lý các giá trị bị miss (theo các tham số parameter truyền vào).

Mỗi dòng bỏ qua **n** dòng được chỉ định trong *skip_header* được phân tách bởi tham số *delimiter* và các ký tự theo sau *comments* sẽ bị loại bỏ.

Return: Trả về một mảng kiểu 'ndarray'

→ Thích hợp với dataset "phishing_dataset.csv" vì không có hàng header đầu tiên (*skip_header* không cần chỉ định). Đầu vào là file csv nên *delimiter* là ',' (comma). Định nghĩa kiểu dữ liệu là dtype.int32

Xem cách dùng với tất cả tham số tại official document của Numpy:

numpy.genfromtxt — NumPy v1.23 Manual

6. *Sinh viên hoàn thiện code Decision trees trên và đánh giá kết quả nhận được so với phương pháp Logistic regression.*

Xem chi tiết tại file Notebook (.ipynb)

So sánh kết quả:

Logistic Regression:

```
[38]  1 accuracy = 100.0 * accuracy_score(testing_target, predictions)
      2 print("Logistic Regression accuracy: %s" % str(accuracy))

      Logistic Regression accuracy: 91.67797376752601
```

Decision Tree:

```
[43]  1 accuracy = 100.0 * accuracy_score(testing_target, predictions)
      2 print("Decision Tree accuracy: %s" % str(accuracy))

      Decision Tree accuracy: 96.38172772501132
```

➔ Độ chính xác của *Decision Tree* mang lại kết quả tốt hơn *Logistic Regression* khá đáng kể

Lý do: Theo như tìm hiểu, Decision Tree thích hợp với dữ liệu được gán nhãn và phân biệt chỉ với 2 giá trị "có" hoặc "không có", khi chạy tay thì cây đơn giản chỉ chọn lựa theo nhị phân → Độ chính xác cao

V	W	X	Y	Z	AA	AB	AC	AD	AE
1	1	-1	-1	-1	-1	1	1	-1	-1
1	1	-1	-1	0	-1	1	1	1	-1
1	1	1	-1	1	-1	1	0	-1	-1
1	1	-1	-1	1	-1	1	-1	1	-1
-1	1	-1	-1	0	-1	1	1	1	1
1	1	1	1	1	-1	1	-1	-1	1
1	1	1	-1	-1	-1	1	0	-1	-1
1	1	-1	-1	0	-1	1	0	1	-1
1	1	1	-1	1	1	1	0	1	1
1	1	1	-1	0	-1	1	0	1	-1
1	1	-1	1	1	1	1	-1	-1	1
1	1	-1	-1	-1	-1	1	0	-1	-1
-1	1	1	-1	-1	-1	1	0	1	-1
1	1	-1	-1	0	-1	1	1	1	-1
1	1	1	-1	1	-1	1	-1	1	1
1	1	1	-1	-1	-1	1	0	1	-1
1	1	1	-1	0	-1	1	1	-1	-1
1	1	-1	1	1	-1	1	1	-1	-1
1	1	1	-1	-1	1	1	-1	-1	1
-1	-1	1	-1	0	-1	1	0	-1	1
-1	1	-1	1	1	-1	1	-1	-1	1
1	1	-1	1	-1	-1	1	0	-1	1
1	1	1	1	0	-1	1	-1	1	1
1	1	1	1	1	-1	1	-1	1	1
1	1	1	1	-1	-1	1	0	1	1

Nhãn với 2 giá trị -1 và 1

Còn Logistic Regression predict giá trị dựa theo một đồ thị nên có thể sẽ không theo sát được dataset.

***7.** Sinh viên thực hiện code phát hiện phising website bằng mô hình học máy Logistic regression và Decision trees với train và test trên tập dữ liệu https://www.kaggle.com/shashwatwork/phishing-dataset-for-machinelearning*

Xem chi tiết tại file Notebook (.ipynb)

8. *Sinh viên thực hiện code phát hiện phising website bằng mô hình học máy Logistic regression hoặc Decision trees với train và test trên tập dữ liệu phishtank. Tham khảo cách xử lý và trích xuất thuộc tích https://github.com/surajr/URL-Classification*

Xem chi tiết tại file Notebook (.ipynb)