



fit@hcmus

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH ĐẠI HỌC KHOA HỌC TỰ NHIÊN

ĐỒ ÁN CUỐI KỲ

DỰ BÁO LƯỢNG GẠO TIÊU THỤ Ở CÁC NƯỚC CHÂU Á VÀ MỸ LATINH

Giáo viên hướng dẫn

Thầy Trần Trung Kiên

Nhóm thực hiện: 18

Họ và tên

MSSV

Trần Hữu Chí Bảo

18120288

Trần Thanh Tùng

18120258

Nội dung trình bày

- ☐ Giới thiệu đề tài
- ☐ Thu thập dữ liệu
- ☐ Khám phá dữ liệu
- ☐ Tiền xử lý
- ☐ Mô hình hóa
- ☐ Đánh giá kết quả
- ☐ Nhìn lại quá trình
- ☐ Tài liệu tham khảo

Đề tài

- ❑ Thu thập dữ liệu về sản xuất lúa gạo của 1 quốc gia trong 1 năm, dân số, GDP bình quân đầu người của nước đó.
- ❑ Câu hỏi: Lượng gạo tiêu thụ được tính theo công thức nào từ các thông tin trên.
- ❑ Ý nghĩa: Dự báo trước được lượng gạo tiêu thụ vào những năm tới, từ đó cân bằng cán cân xuất nhập khẩu gạo và đảm bảo an ninh lương thực,...

Thu thập dữ liệu

1. Dữ liệu về sản xuất lúa gạo:

- Nguồn dữ liệu: ricepedia.org
- Định dạng: Bảng
- Phương pháp thu thập dữ liệu: Parse html

Element	1961	1962	1963	1964	1965	1966	1967	1968	1969	1970	1971	1972
Arable land (000 Ha)	5,550.00	5,550.00	5,550.00	5,550.00	5,550.00	5,550.00	5,570.00	5,590.00	5,600.00	5,630.00	5,630.00	5,650.00
Rice area (000 Ha)	4,744.00	4,888.86	4,496.52	4,987.80	4,826.30	4,681.30	4,795.80	4,893.80	4,930.00	4,724.40	4,692.10	4,900.00
Paddy yield (t/Ha)	1.90	1.99	2.14	1.94	1.94	1.81	1.92	1.71	1.79	2.15	2.23	2.19
Paddy Production (000 t)	8,997.40	9,747.04	9,622.67	9,697.03	9,369.70	8,463.50	9,188.40	8,366.15	8,815.00	10,173.30	10,447.00	10,748.20
Milled production (000 t)	6,001.27	6,501.28	6,418.32	6,467.92	6,249.59	5,645.15	6,128.66	5,580.22	5,879.60	6,785.59	6,968.15	7,169.05
Rice imports (000 t)	18.50	211.50	450.00	420.00	329.59	894.19	1,250.00	1,230.00	1,030.00	1,260.00	690.00	880.00
Rice exports (000 t)	182.25	89.76	229.32	59.58	2.96	12.52	3.44	2.36	20.08	18.48	6.00	3.00
Total rice consumption (000 t)	5,833.23	6,256.78	6,336.93	6,529.92	6,584.77	6,631.98	6,874.22	6,839.97	6,977.51	7,443.89	7,516.12	7,709.83
Fertilizer usage (NPK) (000 t)	16.04	17.12	18.92	16.58	14.05	8.29	20.91	20.30	49.21	55.29	51.10	45.69

Thu thập dữ liệu

2. Dữ liệu về dân số:

- Nguồn: <https://www.worldbank.org/>
- Định dạng dữ liệu: json
- Phương pháp thu thập dữ liệu: Dùng API của world bank.

❑ Cú pháp:

`api.worldbank.org/v2/country/ + “mã quốc gia” +
/indicator/SP.POP.TOTL? + “format=json”`

Thu thập dữ liệu

3. Dữ liệu về GDP bình quân đầu người :

- Nguồn dữ liệu:
<https://countryeconomy.com/>
- Định dạng: bảng
- Phương pháp thu thập dữ liệu: Parse html

Evolution: GDP per capita Vietnam		
Date	GDP per capita	GDP P.C. Annual Growth
2018	2,525\$	8.4%
2017	2,330\$	7.3%
2016	2,172\$	4.1%
2015	2,086\$	1.9%
2014	2,047\$	7.8%
2013	1,899\$	8.5%
2012	1,751\$	14.3%
2011	1,532\$	18.1%
2010	1,297\$	9.8%
2009	1,181\$	2.3%
2008	1,154\$	25.4%
2007	920\$	15.5%
2006	797\$	13.9%
2005	700\$	15.9%
2004	604\$	23.4%
2003	489\$	11.1%
2002	440\$	6.5%
2001	413\$	2.9%
2000	402\$	7.2%
1999	375\$	3.8%
1998	361\$	-0.3%
1997	362\$	7.2%

Thu thập dữ liệu

❑ Sau khi thu thập dữ liệu ta tạo được 1 dataframe hoàn chỉnh

	Arable land	Paddy yield	Milled production	Rice exports	Fertilizer usage	Rice area	Paddy Production	Rice imports	Total rice consumption	GDP	Population
0	159,756.00	1.71	43,089.53	15.85	16.63	37,757.81	64,602.00	532.05	41,577.75	119	567868018
1	160,186.00	1.60	39,264.92	14.99	17.28	36,687.81	58,867.95	300.35	41,593.65	123	581087256
2	161,044.00	1.73	44,073.36	17.78	17.63	38,285.01	66,077.01	264.00	42,310.84	144	594770134
3	161,501.00	1.57	39,786.55	41.43	15.93	37,888.40	59,650.00	109.47	41,890.58	163	608802600
4	161,838.00	1.86	48,925.78	18.29	21.59	39,475.41	73,352.00	276.82	47,148.31	158	623102897
...
1447	4,145.00	5.30	210.25	133.47	82.09	59.48	315.21	0.90	44.78	4347	6248020
1448	4,305.00	5.19	272.30	203.89	97.64	78.61	408.25	1.14	52.41	5298	6333976
1449	4,415.00	4.95	264.13	262.06	83.30	80.00	396.00	0.99	56.60	5151	6421512
1450	4,500.00	6.30	411.80	366.28	96.92	98.00	617.40	1.12	57.86	5883	6510276
1451		6.70				120.00	804.00			6050	6599526

1452 rows × 11 columns

Khám phá dữ liệu

- ❑ Dữ liệu có bao nhiêu dòng và bao nhiêu cột?
- ❑ Mỗi dòng có ý nghĩa gì? Có vấn đề các dòng có ý nghĩa khác nhau không?
- ❑ Dữ liệu có các dòng bị lặp không?
- ❑ Ý nghĩa của các cột trong dataframe:
 1. **Arable land**: Diện tích đất có thể canh tác được, đơn vị nghìn ha.
 2. **Paddy yield**: Năng suất, đơn vị tấn/ha.
 3. **Miled Production**: Sản lượng gạo, đơn vị nghìn tấn.
 4. **Rice exports** : Lượng gạo xuất khẩu, đơn vị nghìn tấn.
 5. **Fertilizer usage**: Lượng phân bón sử dụng, đơn vị nghìn tấn.
 6. **Rice area**: Diện tích trồng lúa, đơn vị nghìn ha.
 7. **Paddy production**: Sản lượng lúa, đơn vị nghìn tấn.
 8. **Rice imports**: Lượng gạo nhập khẩu, đơn vị nghìn tấn.
 9. **Total rice consumption**: Tổng lượng gạo tiêu thụ, đơn vị nghìn tấn.
 10. **GDP**: Tổng thu nhập bình quân đầu người, đơn vị \$.
 11. **Population**: Tổng dân số, đơn vị người

Khám phá dữ liệu

- ❑ Đưa ra câu hỏi: Lượng gạo tiêu thụ được tính theo công thức nào từ các thông tin trên.
- ❑ Ý nghĩa: Dự báo trước được lượng gạo tiêu thụ vào những năm tới, từ đó cân bằng cán cân xuất nhập khẩu gạo và đảm bảo an ninh lương thực,...

Khám phá dữ liệu cột output

1. Cột output (cột “total rice consumption”)

❑ Kiểu dữ liệu của output:

```
[ ] # Cột output hiện có kiểu dữ liệu gì?  
data_df['Total rice consumption'].dtype  
  
dtype('float64')
```

❑ Cột output có giá trị thiếu không ?

```
[ ] # Cột output có giá trị thiếu không?  
data_df['Total rice consumption'].isna().sum()
```

173



Loại các dòng mà thiếu output

Tiền xử lý

- ❑ Sau khi đã khám phá dữ liệu ta tiến hành tách tập dữ liệu ban đầu thành tập train, tập validation và tập test.

Tách theo tỉ lệ: 60% : 20% : 20% lần lượt cho bộ train, bộ validation, bộ test.

+ Code

+ Text

```
[31] # Tách (tập train, validation) và tập test theo tỉ lệ 80%:20%  
new_X_df, test_X_df, new_y_sr, test_y_sr = train_test_split(X_df, y_sr, test_size=0.2, random_state=1)
```


Tách tập train và tập validation

```
[53] # Tách tập huấn luyện và tập validation theo tỉ lệ 75%:25%  
train_X_df, val_X_df, train_y_sr, val_y_sr = train_test_split(new_X_df, new_y_sr, test_size=0.25, random_state=1)
```

Tiền xử lý

1. Khám phá dữ liệu tập train:

Kiểm tra kiểu dữ liệu của các cột input:

 Arable land float64
Paddy yield float64
Milled production float64
Rice exports float64
Fertilizer usage float64
Rice area float64
Paddy Production float64
Rice imports float64
GDP float64
Population int64
dtype: object

❑ Các giá trị dạng số được phân bố:

	Arable land	Paddy yield	Milled production	Rice exports	Fertilizer usage	Rice area	Paddy Production	Rice imports	GDP	Population
missing_ratio	0.0	0.00	0.00	29.7	1.20	0.0	0.0	11.00	0.0	0.000000e+00
min	200.0	0.68	9.14	0.0	0.00	14.7	13.7	0.00	26.0	1.563093e+06
lower_quartile	1715.5	2.30	215.00	1.0	17.50	112.8	322.3	2.20	456.5	9.739812e+06
median	3700.0	3.00	1070.60	16.9	61.00	471.0	1605.2	34.70	1409.0	2.078307e+07
upper_quartile	14020.5	4.10	5788.80	258.4	151.00	2413.9	9116.1	237.00	3268.5	6.111428e+07
max	163618.0	8.38	106186.40	11300.1	2241.37	44900.0	159200.0	4671.22	44674.0	1.280846e+09



Một số cột có tỷ lệ missing value khá cao.

Tiền xử lý

1. Loại bỏ thuộc tính không cần thiết:

Thuộc tính paddy yield được loại bỏ vì đây là thuộc tính suy diễn, có thể suy ra thuộc tính này từ 2 thuộc tính là Paddy Production và Rice area theo công thức:

$$\text{Paddy yield} = \text{Paddy Production} / \text{Rice area}$$

2. Điền giá trị thiếu cho các cột:

Khi khám phá tập train ta thấy được rằng có khá nhiều cột có giá trị thiếu.

Vì thế ta dùng phương pháp KNN (k-nearest neighbor) để điền giá trị cho các giá trị này



Cuối cùng ta tạo pipeline cho quá trình này, nhằm tránh rò rỉ dữ liệu (data leakage)

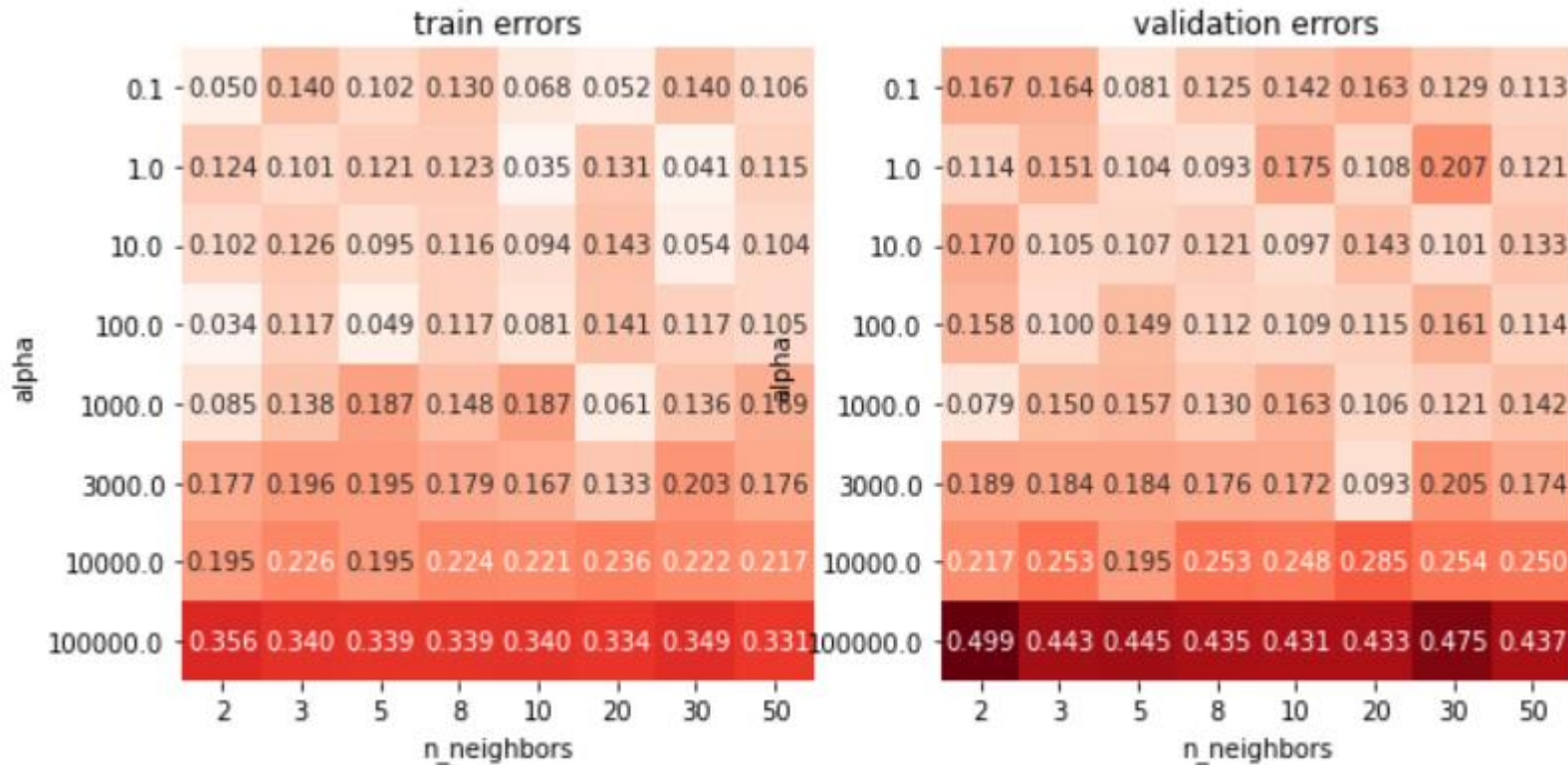
Mô hình hóa

- Thực hiện transform trên tập validation.
- Tiến hành lựa chọn mô hình tốt nhất:
 - ☐ Mô hình Linear Regression
 - ☐ Mô hình Neural Network



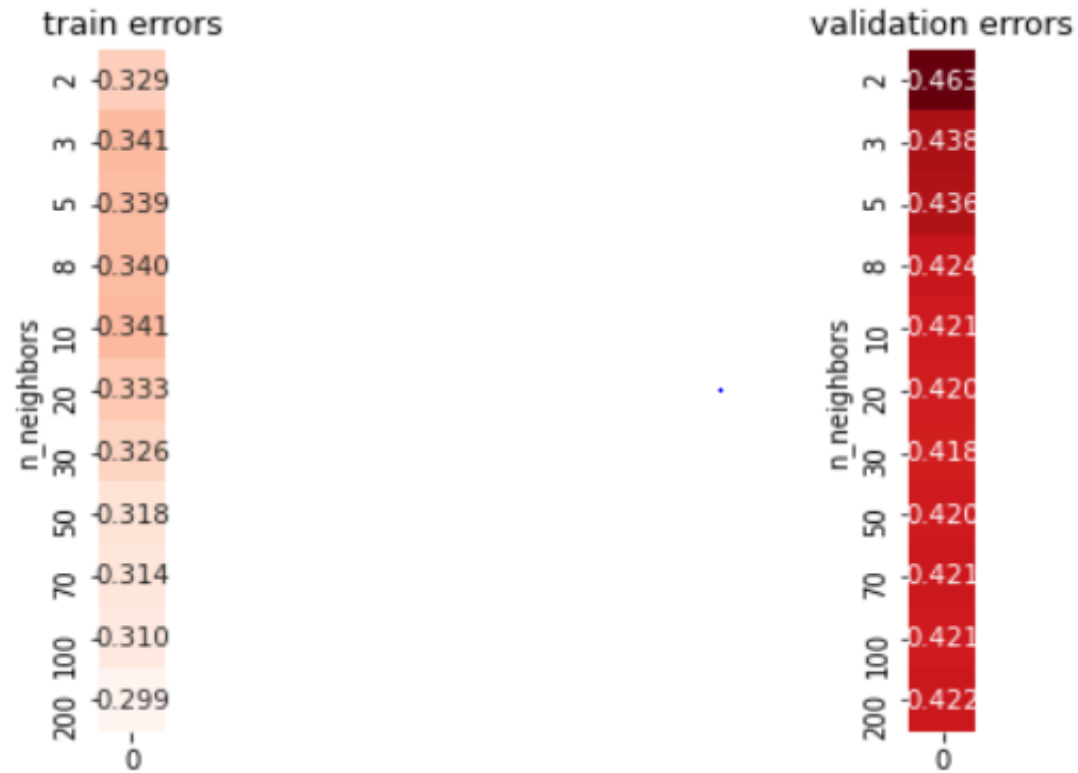
Tạo một Pipeline hoàn chỉnh để chứa quá trình tiền xử lý phía trên và mô hình MLP phía dưới này.

Mô hình Neural Network



Kết quả độ lỗi khi thử nghiệm nhiều giá trị 2 siêu tham số α và $n_neighbor$

Mô hình Linear Regression



Kết quả độ lỗi khi thử nghiệm nhiều giá trị của siêu tham số $n_neighbor$

Kết quả

- ❑ Chọn mô hình MLP do có độ lỗi thấp hơn
- ❑ Kết quả độ lỗi của mô hình :

Đánh giá mô hình tìm được

```
[48] (1 - full_pipeline.score(new_X_df,new_y_sr)) * 100 # do loi tren toan bo tap huan luyen va tap validation  
0.08494704724988367
```

```
[49] (1 - r2_score(test_y_sr, full_pipeline.predict(test_X_df))) * 100 # do loi tren tap test  
0.2548413750649625
```

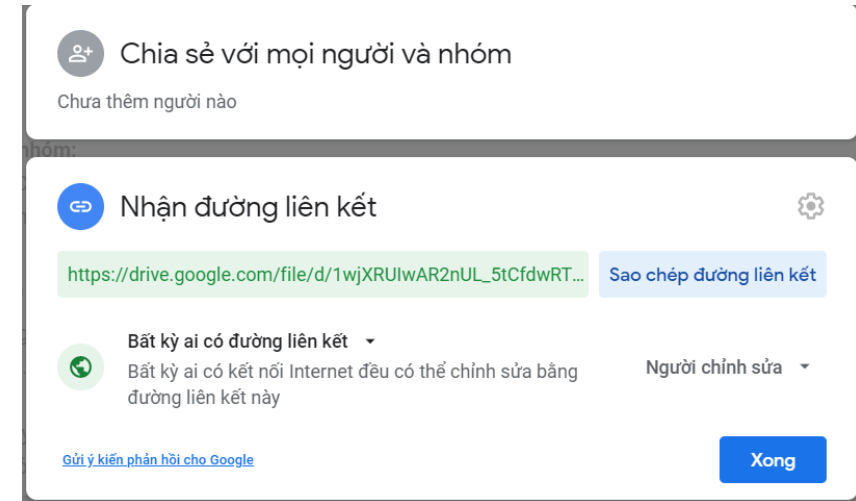


Mô hình có độ chính xác khá cao khi độ lỗi trên tập test chỉ là 0.25%, đạt được độ tin cậy mong đợi.

Quản lý phiên bản và hợp tác nhóm

1. Hợp tác nhóm:

- ❑ Sử dụng tính năng chia sẻ của google colab, bằng cách này cả 2 có thể code cùng 1 lúc cũng như biết được rằng người kia đang code tới đâu hay là 1 người code 1 người test...



2. Quản lý phiên bản:

- ❑ Công cụ: github



Nhìn lại quá trình làm đồ án

- Khó khăn:
 - ❑ Khó khăn trong việc đi tìm nguồn dữ liệu chính thống với thông tin chính xác, ít nhiễu, ít thô.
 - ❑ Khó khăn trong việc tìm hiểu knowledge domain của dữ liệu.
- Những điều hữu ích học được:
 - ❑ Được làm việc hoàn chỉnh trên một mô hình khoa học dữ liệu.
 - ❑ Kỹ năng khám phá và tiền xử lý dữ liệu.
 - ❑ Kỹ năng đọc và tìm hiểu dữ liệu, cũng như nghiên cứu các mô hình.

Tài liệu tham khảo

1. <https://likegeeks.com/python-correlation-matrix>
2. <https://worldbank.github.io/debt-data/api-guide/ids-api-guide-python-1.html>
3. <https://scikit-learn.org/stable/modules/generated/sklearn.impute.KNNImputer.html>

**Cảm ơn thầy đã theo dõi phần trình bày
của nhóm**