TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN

-----

# BÁO CÁO

# ĐỒ ÁN 3 – LINEAR REGRESSION


**Bộ môn :** Toán ứng dụng và thống kê cho Công nghệ thông tin
**Giảng viên lý thuyết :** Vũ Quốc Hoàng
**Giảng viên thực hành :** Nguyễn Văn Quang Huy
Trần Thị Thảo Nhi
Phan Thị Phương Uyên

**Sinh viên thực hiện**
Lớp : 20CLC05
MSSV **:** 20127206
Tên : Vũ Đình Duy Khánh

**MỤC LỤC**

# I. Thông tin cá nhân

Lớp : 20CLC05

MSSV **:** 20127206

Tên : Vũ Đình Duy Khánh

# II. Các chức năng đã hoàn thành

| STT | Chức năng | Mức độ hoàn thành |
|---|---|---|
| **1** | Sử dụng toàn bộ 10 đặc trưng | 100% |
| **2** | Xây dựng mô hình sử dụng duy nhất 1 đặc trưng | 100% |
| **3** | Tự xây dựng mô hình, tìm mô hình cho kết quả tốt nhất | 100% |

# III. Liệt kê thư viện, hàm

## 1. Liệt kê thư viện

- Pandas : làm việc dễ dàng và trực quan với cấu trúc dạng bảng của dữ liệu đầu vào
- Numpy : tính toán trên mảng nhiều chiều, kích thước lớn với các hàm đã được tối ưu, thích hợp với dữ liệu đầu vào
- Scikit-learn (sklearn) : cung cấp nhiều thuật toán, công cụ xử lý các bài toán mạnh mẽ, dễ sử dụng, dễ code, cho hiệu quả thời gian cao

## 2. Liệt kê hàm

- sklearn.metrics.mean_squared_error[1]
- def RMSE(test, pred)
  - Ý tưởng : RMSE là căn bậc hai của bất giá trị trả về của hàm sklearn.metrics.mean_squared_error
  - Input :
    + test (array (n_samples,) or (n_samples, n_outputs)): 1 giá trị mục tiêu kiểm tra
    + pred (array (n_samples,) or (n_samples, n_outputs)) : giá trị mục tiêu ước tính
  - Output :
    + rmse (float) : Giá trị rmse
- def CrossValidation_5Fold(x_train_clone, Y_train_clone)
  - Ý tưởng :
    + Dùng hàm sklearn.model_selection.KFold[2](n_splits = 5,shuffle = True) để chia và trộn mô hình thành 5 phần bằng nhau, train(4 phần) và test(1 phần)
    + Tính RMSE 5 lần sau đó trả về giá trị trung bình

- Input :
    + X_train_clone (numpy) : DataFrame huấn luyện dưới dạng numpy
    + Y_train_clone (numpy) : Series chứa 1 giá trị mục tiêu kiểm tra dưới dạng numpy
- Output :
    + rmse (float) : giá trị rmse

- def new_paradigm_with_avg_2_best_feature(best_feature, second_feature, best_feature_name, second_feature_name)
    - Ý tưởng : tính đặc trưng mới với kết quả là trung bình 2 đặc trưng tốt nhất
    - Input :
        + best_feature (numpy) : mảng dữ liệu đặc trưng tốt nhất
        + second_feature (numpy) : mảng dữ liệu đặc trưng tốt thứ 2
        + best_feature_name (string) : tên đặc trưng tốt nhất
        + second_feature_name (string) : tên đặc trưng tốt thứ 2
    - Output : kết quả (pd.Series)
- def new_paradigm_sum_of_sqrt_2_best_feature (best_feature, second_feature, best_feature_name, second_feature_name)
    - Ý tưởng : tính đặc trưng mới với kết quả là tổng của căn bậc 2 2 đặc trưng tốt nhất
    - Input :
        + best_feature (numpy) : mảng dữ liệu đặc trưng tốt nhất
        + second_feature (numpy) : mảng dữ liệu đặc trưng tốt thứ 2
        + best_feature_name (string) : tên đặc trưng tốt nhất
        + second_feature_name (string) : tên đặc trưng tốt thứ 2
    - Output : kết quả (pd.Series)
- def new_paradigm_with_sum_of_root4_2_best_feature(best_feature, second_feature, best_feature_name, second_feature_name)
    - Ý tưởng : tính đặc trưng mới với kết quả là tổng của căn bậc 4 2 đặc trưng tốt nhất
    - Input :
        + best_feature (numpy) : mảng dữ liệu đặc trưng tốt nhất
        + second_feature (numpy) : mảng dữ liệu đặc trưng tốt thứ 2
        + best_feature_name (string) : tên đặc trưng tốt nhất
        + second_feature_name (string) : tên đặc trưng tốt thứ 2
    - Output : kết quả (pd.Series)
- def new_paradigm_with_thinness_age_5_19(train, test)
    - Ý tưởng : tính đặc trưng mới với kết quả là tổng của thinness age 5-9 và thinness age 10-19
    - Input :
        + best_feature (numpy) : mảng dữ liệu đặc trưng tốt nhất

+ second_feature (numpy) : mảng dữ liệu đặc trưng tốt thứ 2
+ best_feature_name (string) : tên đặc trưng tốt nhất
+ second_feature_name (string) : tên đặc trưng tốt thứ 2

- Output : kết quả (pd.Series)

- def Model_Avg_2_best_feature(X_train, X_test)
  - Ý tưởng : tạo mô hình mới với đặc trưng mới là kết quả hàm new_paradigm_with_avg_2_best_feature, xóa đi 2 đặc trưng cũ.
  - Input :
    + X_train (Dataframe) : chứa 10 đặc trưng huấn luyện ban đầu
    + X_test (Dataframe) : chứa 10 đặc trưng kiểm tra ban đầu
  - Output :
    + new_X_train (Dataframe) : chứa các đặc trưng huấn luyện mới
    + new_X_test (Dataframe) : chứa các đặc trưng kiểm tra mới
- def Model_Sum_Of_Sqrt_2_best_feature(X_train, X_test)
  - Ý tưởng : tạo mô hình mới với đặc trưng mới là kết quả hàm new_paradigm_with_sum_of_sqrt_2_best_feature, xóa đi 2 đặc trưng cũ.
  - Input :
    + X_train (Dataframe) : chứa 10 đặc trưng huấn luyện ban đầu
    + X_test (Dataframe) : chứa 10 đặc trưng kiểm tra ban đầu
  - Output :
    + new_X_train (Dataframe) : chứa các đặc trưng huấn luyện mới
    + new_X_test (Dataframe) : chứa các đặc trưng kiểm tra mới
- def Model_Sum_Of_Root4_2_best_feature(X_train, X_test)
  - Ý tưởng : tạo mô hình mới với đặc trưng mới là kết quả hàm new_paradigm_with_sum_of_root4_2_best_feature, xóa đi 2 đặc trưng cũ.
  - Input :
    + X_train (Dataframe) : chứa 10 đặc trưng huấn luyện ban đầu
    + X_test (Dataframe) : chứa 10 đặc trưng kiểm tra ban đầu
  - Output :
    + new_X_train (Dataframe) : chứa các đặc trưng huấn luyện mới
    + new_X_test (Dataframe) : chứa các đặc trưng kiểm tra mới
- def Model_Sqrt_2_best_feature(X_train, X_test)
  - Ý tưởng : tạo mô hình mới với 1 đặc trưng là kết quả của căn bậc 2 đặc trưng tốt nhất cộng căn bậc 2 đặc trưng tốt thứ 2
  - Input :
    + X_train (Dataframe) : chứa 10 đặc trưng huấn luyện ban đầu
    + X_test (Dataframe) : chứa 10 đặc trưng kiểm tra ban đầu
  - Output :
    + train_fea (Dataframe) : chứa đặc trưng huấn luyện mới
    + test_fea (Dataframe) : chứa đặc trưng kiểm tra mới

- def Model_Root4_2_best_feature(X_train, X_test)
  - Ý tưởng : tạo mô hình mới với 2 đặc trưng tốt nhất bằng căn bậc 4 giá trị của nó
  - Input :
    + X_train (Dataframe) : chứa 10 đặc trưng huấn luyện ban đầu
    + X_test (Dataframe) : chứa 10 đặc trưng kiểm tra ban đầu
  - Output :
    + new_X_train (Dataframe) : chứa các đặc trưng huấn luyện mới
    + new_X_test (Dataframe) : chứa các đặc trưng kiểm tra mới
- def create_new_data(X_train, X_test)
  - Ý tưởng : dùng các hàm Model… ở trên tạo ra list các mô hình
  - Input :
    + X_train (Dataframe) : chứa 10 đặc trưng huấn luyện ban đầu
    + X_test (Dataframe) : chứa 10 đặc trưng kiểm tra ban đầu
  - Output :
    + new_train_data (list) : chứa các mô hình huấn luyện mới
    + new_test_data (list) : chứa các mô hình kiểm tra mới

# IV. Các yêu cầu đề bài :

## 1. Câu 1a

- Các bước thực hiện :
  + Dùng phương thức fit() của lớp OLSLinearRegression để train model với X_train và Y_train
  + Dùng phương thức predict() để đưa ra kết quả dự đoán với X_test sau đó đánh giá bằng hàm RMSE với kết quả dự đoán và giá trị thật Y_test

- Công thức hồi quy :

Life expectancy = w0*Adult Mortality + w1*BMI + w2*Polio + w3*Diphtheria + w4*(HIV/AIDS) + w5*GDP + w6*Thinness age 10-19 + w7*(Thinness age 5-9) + w8*(Income composition of resources) + w9*Schooling

- Ước lượng sai số RMSE : 7.064046430584209

## 2. Câu 1b

- Các bước thực hiện :
  + Thực hiện tuần tự với từng đặc trưng:
    - Dùng hàm sklearn.model_selection.KFold() có sẵn của thư viện sklearn với các tham số truyền vào n_splits = 5 để chia thành 5 phần, shuffle = True để xáo trộn dữ liệu

- Với mỗi nhóm trong 5 nhóm dữ liệu, lấy 1 nhóm làm tập kiểm tra, 4 nhóm còn lại làm tập huấn luyện
- Kết quả RMSE là trung bình của rmse 5 nhóm dữ liệu trên

+ Thực hiện so sánh tìm ra đặc trưng tốt nhất

- Công thức hồi quy :

Life expectancy = w*Schooling

- Ước lượng sai số RMSE : 10.26095039165537
- Nhận xét :

+ Đặc trưng tốt nhất : Schooling

+ Đặc trưng kém nhất : HIV/AIDS

+ Các đặc trưng như Schooling, HDI ảnh hưởng rất nhiều đến tuổi thọ, điều đó ngược lại với HIV/AIDS, GDP từ đó có thể xây dựng những mô hình tốt hơn.

+ Tuy kết quả được tạo ra từ đặc trưng tốt nhất trong tập dữ liệu, nhưng do chỉ sử dụng 1 đặc trưng so với mô hình 10 đặc trưng nên kết quả vẫn kém hơn so với ban đầu.

## 3. Câu 1c

- Từ câu 1b ta biết được những đặc trưng nào ảnh hưởng nhiều nhất đến tuổi thọ. Từ đó tìm ra các mô hình hiệu quả hơn mô hình gốc
- Các bước thực hiện vẫn như câu 1b nhưng thay vì thực hiện với các đặc trưng, ta thực hiện với các mô hình được xây dựng.
- Các mô hình được xây dựng :

| STT | Mô hình | Mô tả | RMSE |
|---|---|---|---|
| 0 | …. + (Thinness age 5-19) + …. | Bỏ 2 đặc trưng Thinness age 5-9 và 10-19<br>Thay vào đó là đặc trưng mới mang trung bình của 2 đặc trưng cũ | 7.864198 |
| 1 | … + (Schooling + HID)/2 | Bỏ 2 đặc trưng tốt nhất, thay vào đó là trung bình của nó | 8.103070 |
| 2 | … + $(\sqrt{Schooling} + \sqrt{Income\ composition\ of\ resources})$ | Đặc trưng mới là tổng căn 2 đặc trưng tốt nhất<br>Bỏ 2 đặc trưng cũ | 5.209554 |
| 3 | … + $\sqrt[4]{Schooling}$ + $\sqrt[4]{Income\ composition\ of\ resources}$ | 2 đặc trưng mới là căn bậc 4 2 đặc trưng tốt nhất<br>Bỏ 2 đặc trưng cũ | 3.977142 |
| 4 | … + $(\sqrt[4]{Schooling} + \sqrt[4]{Income\ composition\ of\ resources})$ | Đặc trưng mới là tổng của căn 4 2 đặc trưng tốt nhất<br>Bỏ 2 đặc trưng cũ | 4.699223 |

| | | | |
|---|---|---|---|
| 5 | $\sqrt{Schooling} + \sqrt{Income\ composition\ of\ resources}$ | Chỉ sử dụng tổng căn 2 đặc trưng tốt nhất | 6.177053 |

- Mô hình hiệu quả nhất : mô hình 3
- Công thức hồi quy :
- Life expectancy = w0*Adult Mortality + w1*BMI + w2*Polio + w3*Diphtheria + w4*(HIV/AIDS) + w5*GDP + w6*Thinness age 10-19 + w7*(Thinness age 5-9) + w8* $\sqrt[4]{Income\ composition\ of\ resources}$ + w9* $\sqrt[4]{Schooling}$
- Sai số ước lượng RMSE : 3.870607190322188
- Nhận xét :

  + RMSE của mô hình 3 tốt hơn rất nhiều so với mô hình gốc ban đầu (3.87 > 7.064 )

# V. Tài liệu tham khảo


[1] sklearn.metrics.mean_squared_error — scikit-learn 1.1.1 documentation

[2] sklearn.model_selection.KFold — scikit-learn 1.1.1 documentation
Giới thiệu về k-fold cross-validation - Trí tuê nhân tao (trituenhantao.io)

Tài liệu Lab04