BÁO CÁO ĐỒ ÁN

linear regression

Thông tin cá nhân:

Họ và Tên: Lê Quốc Trung

MSSV: 20127369

Tài liệu tham khảo:
kiến thúc tổng quát về linear regression, ý tưởng và kiến thức cho câu 1c:
https://machinelearningcoban.com/2016/12/28/linearregression/
https://towardsdatascience.com/data-preparation-and-preprocessing-is-just-as-important-creating-the-actual-model-in-data-sciences-2c0562b65f62
phần k-fold validation:
https://github.com/kvmduc/applied-math/blob/main/Applied%20Math/Lab04/18127080.py
dẫn chứng cho đặc trưng tốt nhất:
https://thanhnien.vn/hoc-dai-hoc-giup-song-tho-hon-post931582.html
https://zingnews.vn/nguoi-hoc-dai-hoc-song-tho-hon-post1195843.html

I. Thư viện sử dụng

```
import pandas as pd
import numpy as np
from sklearn.model_selection import *
from operator import itemgetter

# Import thêm dữ thư viện nê´u cà`n
```

1. pandas dùng để đọc file csv
2. numpy để thực hiện các tính toán trên ma trận
3. sklearn dùng để sử dụng tách data theo k-fold được thư viện cung cấp sẵn
4. itemgetter để sử dụng enumerate()

II. Các hàm sử dụng

1. **class chính về linear regression để thực hiện fit data cũng như các bước predict**

```
class OLSLinearRegression:
    def fit(self, x, y):
        X_pinv = np.linalg.inv(x.T @ x) @ x.T
        self.w = X_pinv @ y
        return self

    def get_params(self):
        return self.w

    def predict(self, x):
        return np.sum(self.w.ravel() * x, axis=1)
```

2. **hàm tìm các independent variable của công thức hồi quy và tính RMSE:**

```
def LinearRegression(x_train, y_train, x_test, y_test):
```

- bằng cách nhận các tập dữ liệu bao gồm tập train và tập test đã được chia sẵn, sau đó thực hiện chuẩn hóa bằng numpy.array() để có thể thao tác trên ma trận

bằng thư viện numpy sau đó thực hiện fit data và dự đoán y_hat dựa trên tập test bằng công thức: A^T*Ax = A^T*B
- RMSE được tính dựa trên tập y_test cho sẵn và y_hat vừa dự đoán được
- hàm trả về các independent variable của công thức hồi quy và sai số độ đo được tính bằng công thức RMSE

3. **hàm thực thi k-fold cross-validation:**

```
def KFold_CrossValidation(data, feature, k=5):
```

- hàm được tùy chỉnh lại cho thực hiện đầy đủ các yêu cầu của đề bài bao gồm tìm đặc trưng tốt nhất và tìm các và các kết quả x_hat và RMSE của từng đặc trưng, do đề bài yêu cầu k = 5 nên mặc định sẽ là 5-fold cross-validation
- 5-fold cross-validation là phương pháp để đánh giá mô hình bằng cách chia tập dữ liệu ra thành 5 phần (đã được xáo trộn ngẫu nhiên) với mỗi phần sẽ sử dụng chính nó để đánh giá mô hình và sử dụng 4 phần còn lại để train mô hình sau đó sẽ hủy mô hình hiện tại và tiếp tục với các phần khác. phương pháp k-fold cross-validation dễ thực hiện và tốn ít chi phí nhưng cho độ tin cậy khá cao

4. **Các hàm process_data[1-7]:**

Tổng quan: dùng để tạo nên các tập data tùy biến theo ý muốn nhằm tìm ra mô hình tốt nhất

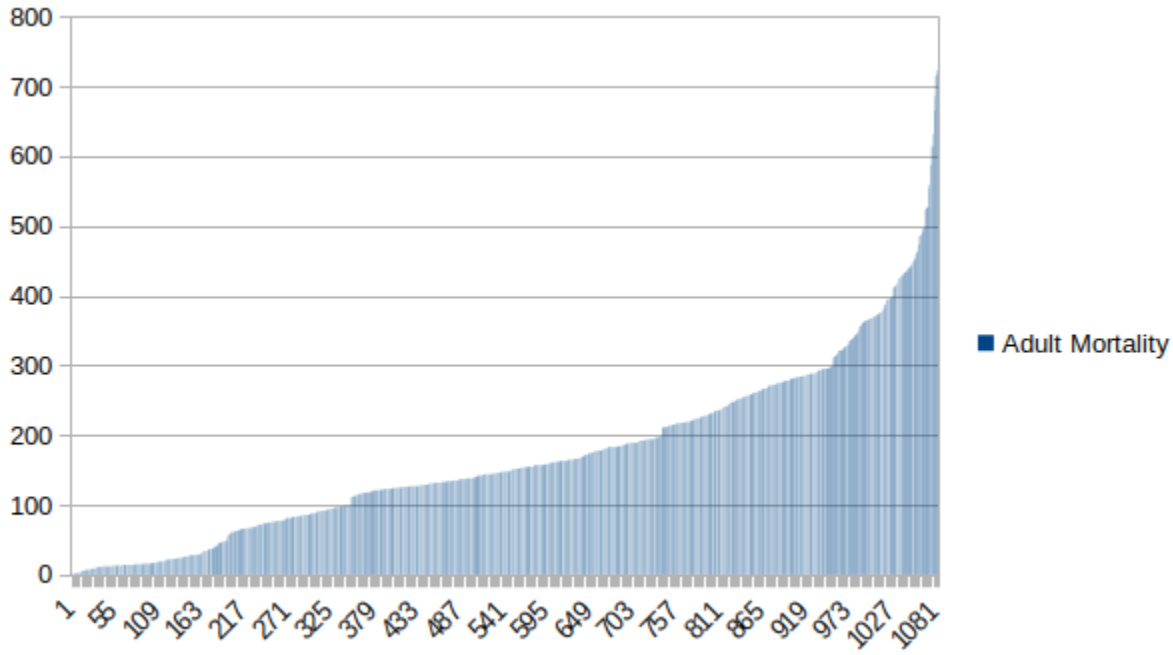
- process_data1: tổng tất cả các đặc trưng trong tập train
- Process_data2: tổng 3 đặc trưng có RMSE thấp nhất dựa trên 5-fold cross-validation được thực hiện bên trên
- Process_data3: sử dụng duy nhất 1 đặc trưng có RMSE cao nhất dựa trên 5-fold cross-validation được thực hiện bên trên
- Process_data1: lấy căn bật 2 của tập data3
- Process_data1: bình phương tập data3
- Process_data1: lấy căn bật 2 của tập best_feature
- process_data1: bình phương tập best_feature

III. Nhận xét mô hình

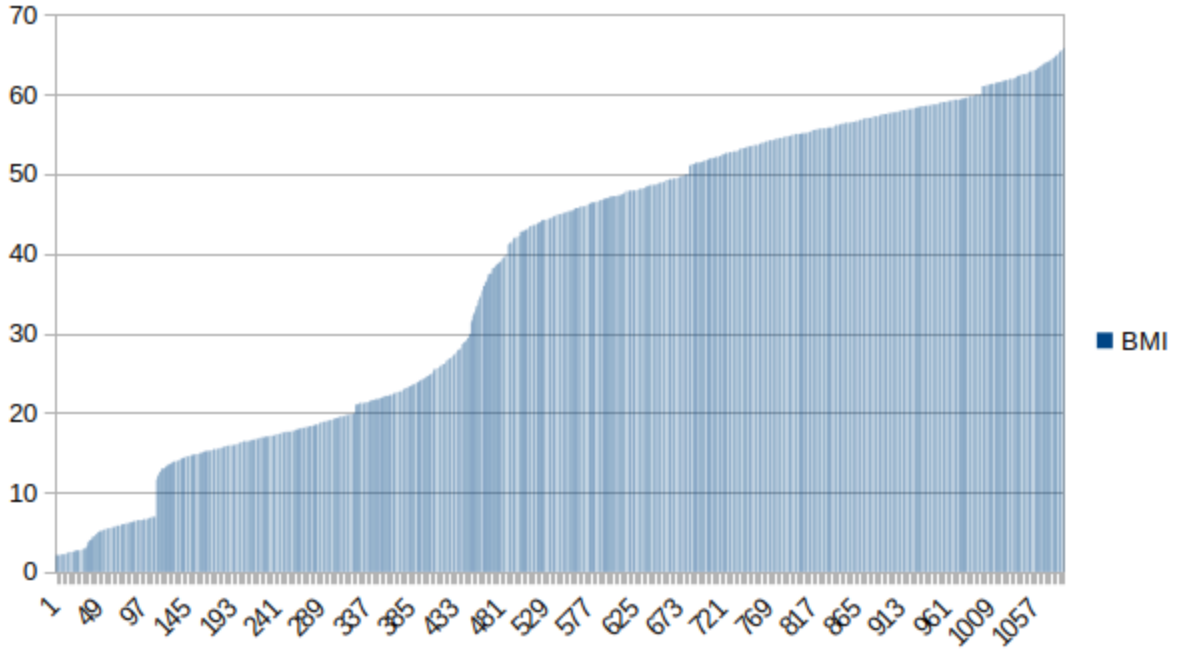
1. mô hình do đề bài cung cấp sử dụng toàn bộ 10 đặc trưng: mô hình cho kết quả độ đo tính theo RMSE khá tốt trong tất cả mô hình trong bài, chứng tỏ độ chính xác cao và có thể cho ra kết quả dự đoán tin tương được, nhưng do sử dụng toàn bộ 10 đặc trưng nên mô hình yêu cầu lượng dữ liệu nhiều (10*1080 dòng) nên sẽ tốn chi phí nhiều hơn các mô hình còn lại
2. **10 mô hình 5-fold cross-validation:**

- tổng quan: 10 mô hình đều không có được độ chính xác cao như mô hình mẫu đề bài sử dụng cả 10 đặc trưng, do linear regression kết quả dự đoán rất dễ bị ảnh hưởng bởi các dữ liệu có tính ngoại lai hay dữ liệu bị nhiễu (sensitive to noise) nên 10 mô hình sử dụng riêng các đặc trưng sẽ càng thể hiện rõ hơn nếu dữ liệu của đặc trưng bị nhiễu. **Các đồ thị được vẽ dựa trên dữ liệu của duy nhất đặc trưng đó giúp ta dễ đánh giá được tính chất của dữ liệu hơn, không phải là mô hình hồi quy**
    - 1. mô hình với đặc trưng "Adult Mortality": mô hình cho ra kết quả RMSE cao (46.2) cho thấy tỉ lệ tử vong ở người trưởng thành không ảnh hưởng nhiều đến

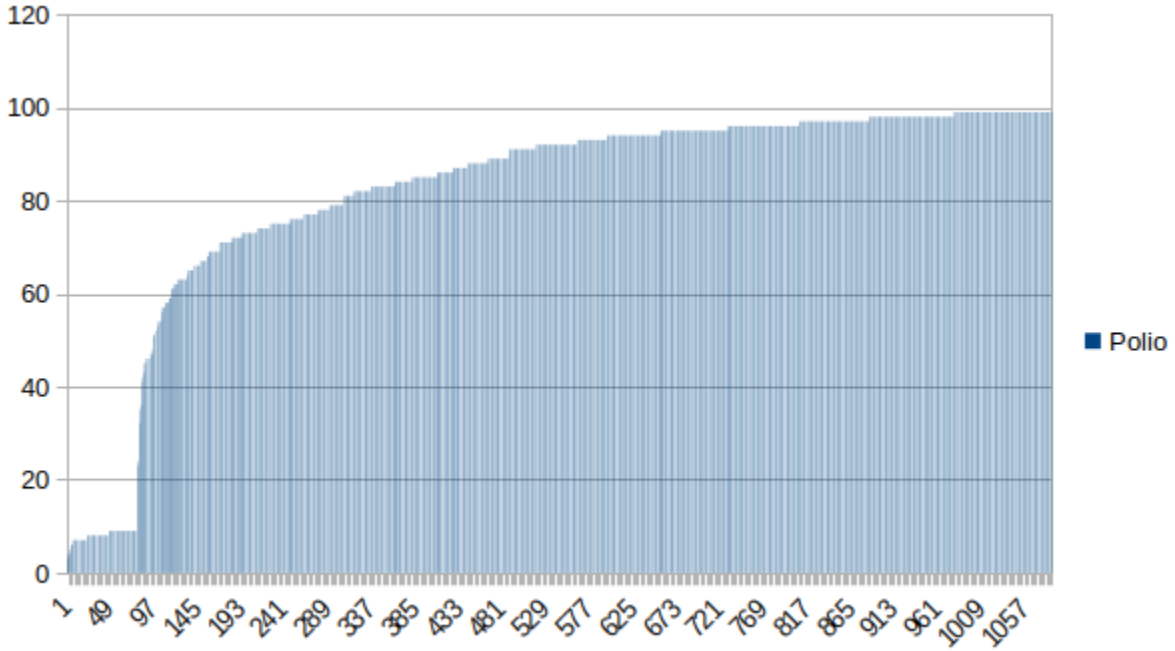
kết quả cần dự đoán, thống kê cho thấy lượng dữ liệu có giá trị lớn hơn nhiều so với phần còn lại của tập là tương đối cao (các data nằm bên phải)



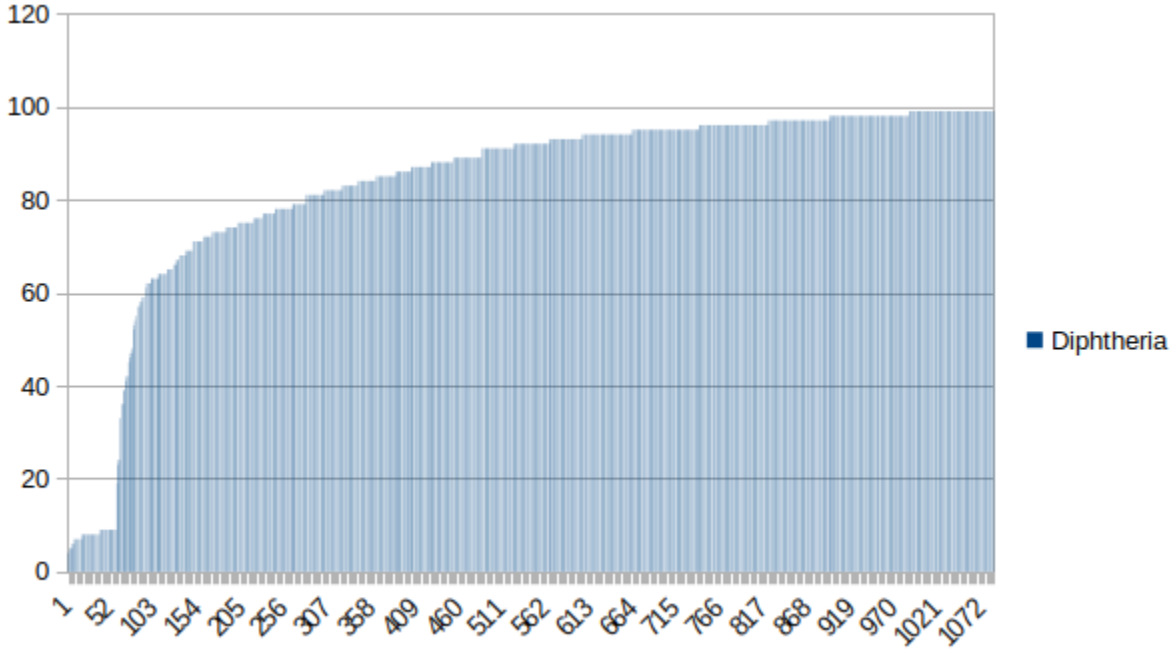
- 2. mô hình với đự trưng BMI": mô hình cho ra kết quả RMSE tương đối (27.9). Thống kê cho thấy phần lớn tập dữ liệu có giá trị tương đối bị nhiễu ít nên chỉ số BMI cho thấy cân nặng kh ảnh hưởng nhiều đến tuổi thọ



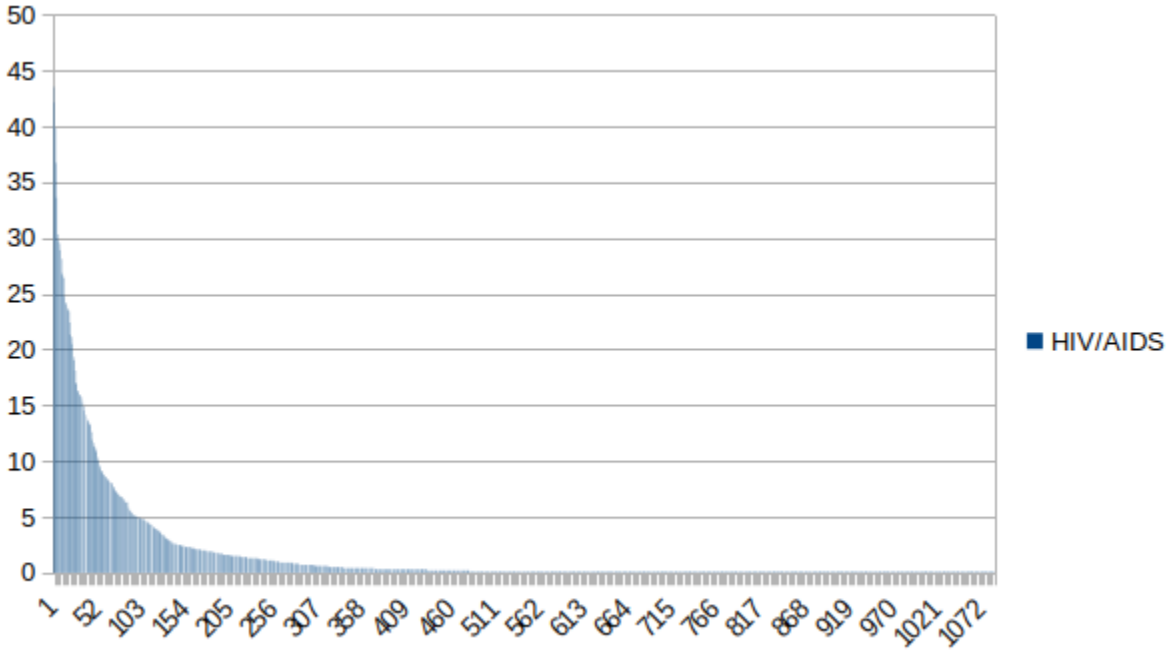
- 3. mô hình với đặc trưng "Polio": kết quả RMSE của mô hình khá tốt (17.8) cho thấy có sự liên quan nhiều giữa polio và tuổi tác. Dù theo thống kê thì dữ liệu của polio bị nhiễu khá nhiều khi phần lớn dữ liệu tập trung từ [40-100] nhưng cũng gần 10% dữ liệu chỉ có giá trị bằng khoảng 1/10 so với phần còn lại



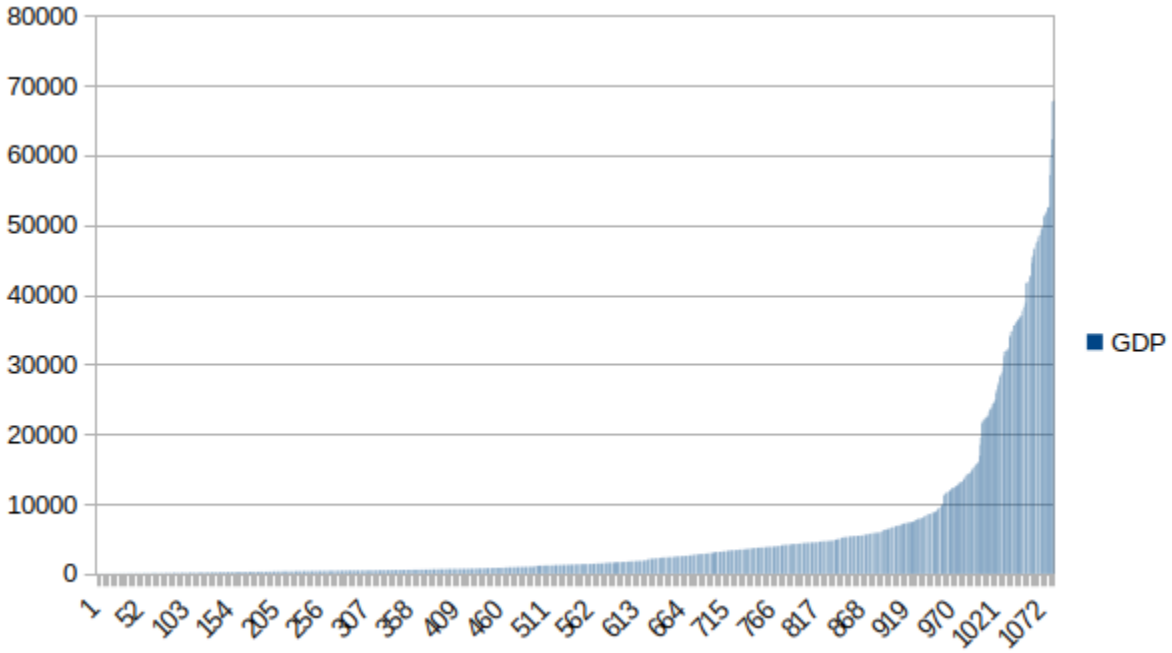
- 4. mô hình với đặc trưng "Diphtheria": kết quả RMSE của mô hình tốt hơn chút đỉnh (16.03) so với “Polio" cho thấy sự ảnh hưởng giữa "Diphtheria" và độ tuổi. Ta có thể thấy sự tương đồng giữa dữ liệu thống kê của hai đặc trưng "Diphtheria" và “Polio" nhưng "Diphtheria" có vẻ phân phối được tập trung hơn



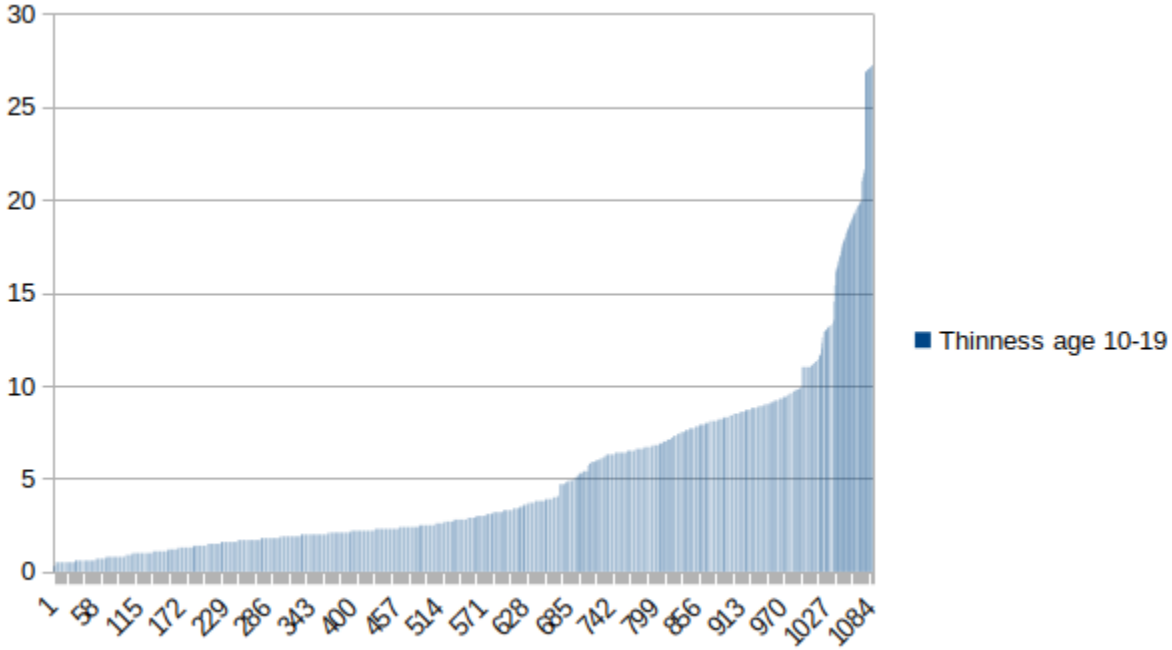
- 5. mô hình với đặc trưng "HIV/AIDS": đây là mô hình cho kết quả RMSE cao nhất (67.1) cho thấy rằng đặc trưng này gần như kh ảnh hưởng đến kết quả dự đoán.



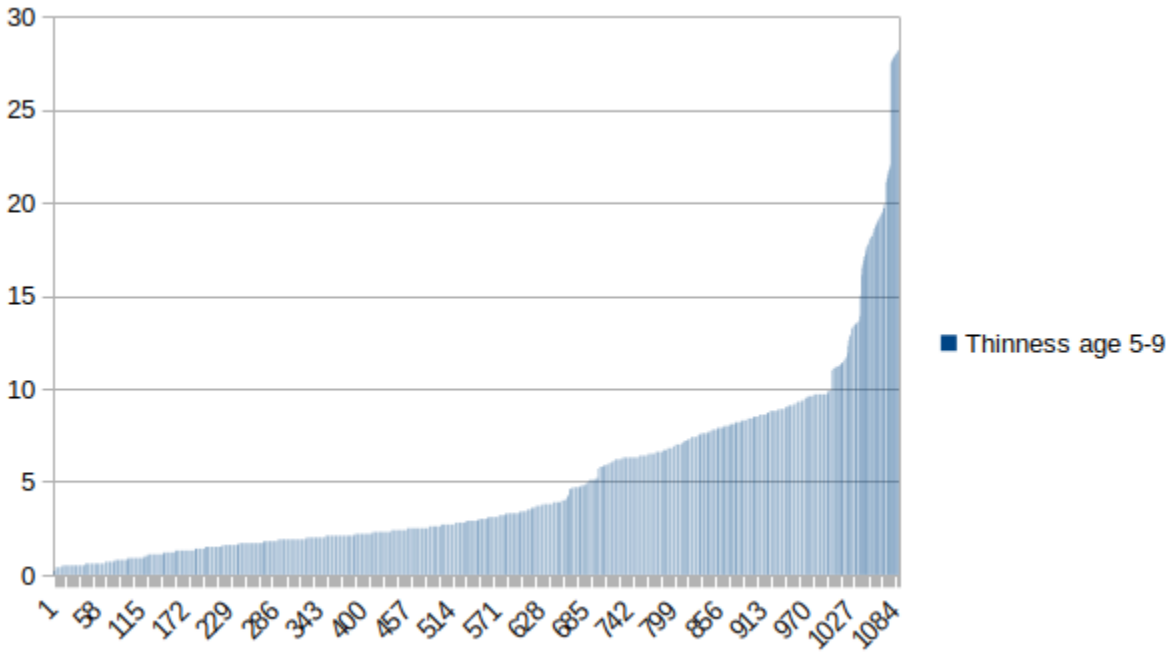
- 6. mô hình với đặc trưng “GDP": kết quả sai số RMSE của mô hình này cũng vô cùng cao (60.2) tương tự như đặc trưng số 5 thể hiện mối tương quan ít giữa đặc trưng và kết quả đích



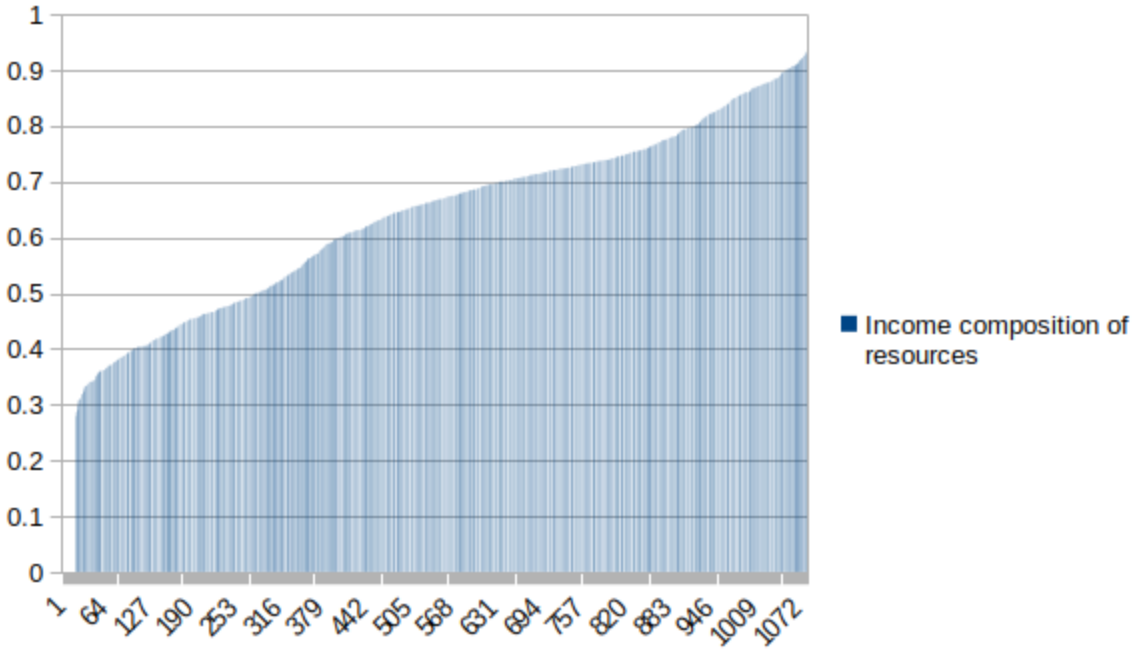
- 7. mô hình với đặc trưng “Thinness age 10-19": mô hình cho chỉ số RMSE cao (51.88)



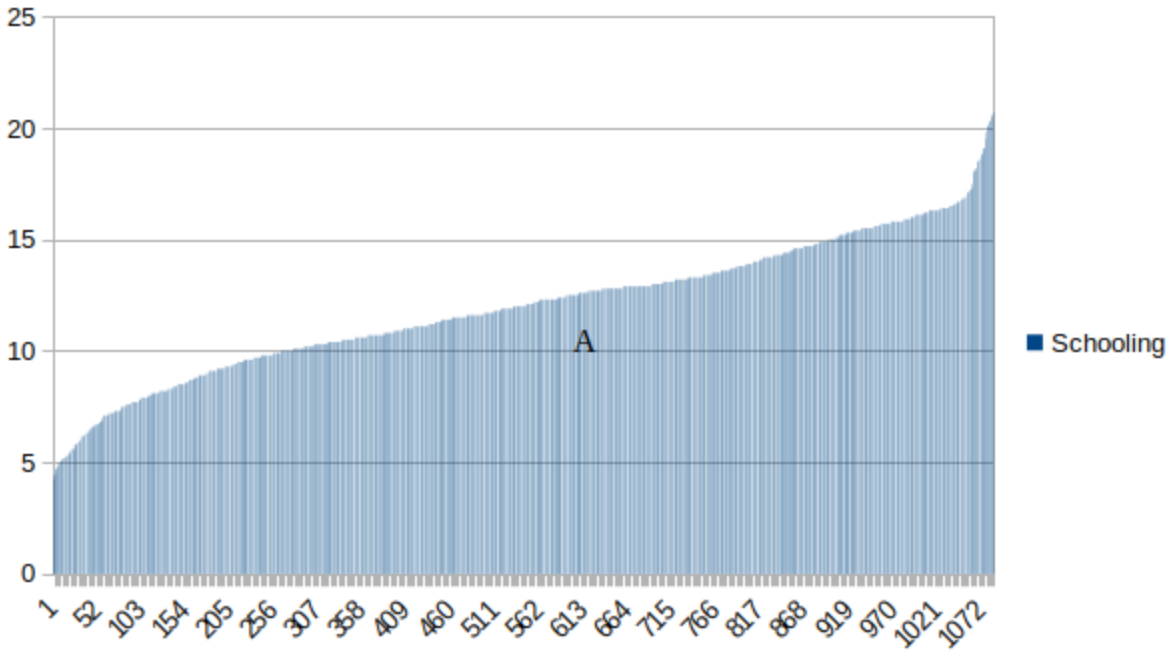
- 8. mô hình với đặc trưng "Thinness age 5-9": mô hình này với mô hình 7 có thể xem là cùng một đặc trưng do không có sự khác nhau nhiều trong dữ liệu cũng như kết quả sai số RMSE (51.83)



- 9. mô hình với đặc trưng "Income composition of resources": mô hình này cho kết quả nổi trội so với các mô hình từng được huấn luyện cho thấy mối liên hệ mật thiết giữa đặc trưng và kết quả cần dự đoán. Biểu đồ thống kê cho thấy dữ liệu này được phân bố rất đồng đều



- **10. mô hình với đặc trưng "Schooling"**: đây chính là mô hình đạt kết quả tốt nhất với độ đo RMSE = 11.8, nhỏ hơn nhiều lần so với một vài mô hình bên trên. Biểu đồ thống kê cho thấy dữ liệu được phân bổ khá đồng đều. Đặc trưng "Schooling" được kết quả tốt như vậy cho thấy tầm quan trọng của việc học, trình độ học vấn đến tuổi thọ. Bởi ta có thể thấy học vấn ảnh hưởng rất nhiều đến nhiều khía cạnh khác trong cuộc sống của chúng ta như thu nhập, nhận thức, và các kiến thức khác về sức khỏe là một tác động gián tiếp rất lớn lên đời sống của chúng ta từ đó thay đổi tuổi thọ



3. **mô hình my_best_feature**: đây là mô hình huấn luyện lại của đặc trưng được cho là tốt nhất dựa theo phương pháp 5-fold cross-validation. Ta có thể thấy với chỉ số RMSE chỉ là 10.2 mặc dù lớn hơn nhưng không đáng kể so với mô hình sử dụng toàn bộ 10 đặc trưng (xét về tỉ lệ) mặc dù mô hình my_best_feature sử dụng lượng dữ liệu ít hơn nhiều so với mô hình 10 đặc trưng cho thấy sự chính xác của phương pháp k-fold cross-validation cũng như tầm quan trong của việc lựa chọn đặc trưng (feature engineering)
4. **7 mô hình tự thiết kế:**
    - 1. với đặc trưng là tổng của tất cả 10 đặc trưng: cho ra kết quả rất tệ chứng tỏ cách sử dụng dữ liệu không hợp lý
    - 2. với đặc trưng là tổng của 3 đặc trưng có kết quả tốt nhất ở câu 1b: cũng là cộng các đặc trưng nhưng cho kết quả tốt hơn rất nhiều so với mô hình trên. mục đích của mô hình này là muốn kiểm tra sự tương quan giữa các đặc trưng với nhau
    - 3. sử dụng đặc trưng có RMSE cao nhất trong câu 1b: đúng với dự đoán cho kết quả rất tệ nhưng có cùng hiệu quả (RMSE gần bằng nhau) với mô hình 1
    - 4. lấy căn bật 2 tất cả dữ liệu trong mô hình 3 nhằm giảm thiểu độ lệch chuẩn của cả tập dữ liệu của đặc trưng đó. Kết quả tốt hơn đáng kể so với chỉ sử dụng dữ liệu thô
    - 5. bình phương tất cả dữ liệu trong mô hình 3 nhằm tăng độ lệch chuẩn của cả tập dữ liệu của đặc trưng đó. Ta có thể thấy kết quả ảnh hưởng một phần nhiều đến dự đoán
    - **6. lấy căn bậc 2 đặc trưng tốt nhất**: đây là mô hình my_best_model: kết hợp giữa đặc trưng tốt nhất và sau khi nhận ra sự ảnh hưởng của việc giảm độ lệch chuẩn của dữ liệu lên kết quả dự đoán. Mô hình này cũng

như là kiểm chứng cho sự ảnh hưởng của độ lệch chuẩn Sau khi lấy căn thi cho ra kết quả tốt hơn nhiều so với trước đó.

- 7. bình phương đặt trưng tốt nhất: lấy đặc chưng tốt nhất và tệ nhất để kiệm nghiệm việc bình phương dữ liệu sẽ ảnh hưởng như thế nào đến kết quả đích. RMSE của mô hình này cho thấy dù là đặc trưng tốt nhất nhưng RMSE so với trước khi bình phương đã tăng hơn gấp đôi và tăng gấp 4 lần so với lấy căn bậc 2 như ở mô hình 6

5. **My_best_model:** Sử dụng ít dữ liệu và tiêu tốn ít tài nguyên huấn luyện hơn nhưng cho kết quả tốt hơn. Mô hình giúp chứng minh cho tầm quan trọng của việc chọn đặc trưng và sử lý dữ liệu ảnh hưởng rất lớn đến kết quả. my_best_model dù chỉ sử dụng 1 đặc trưng nhưng cho RMSE tốt hơn đến 40% so với mô hình sử dụng tất cả 10 đặc trưng.