Học Viện Công Nghệ Bưu Chính Viễn Thông

Báo Cáo bài tập lớn python

Sinh viên	Vũ Dũng
Mã Sinh Viên	B23DCVT110
Lớp	D23CQCE04-B
Nhóm	04

Hà Nội - 04/05/2025

Bài 1

File code: Bai_1.py

Ý tưởng:

1. **Khởi tạo và cấu hình**
    - Tạo một session HTTP với header giả trình duyệt để tránh bị chặn.
    - Đặt URL gốc (base_url) và các thông số delay ngẫu nhiên giữa các request.
2. **Lấy danh sách đội**
    - Tải trang chính (/en/), tìm bảng kết quả mùa giải và trích các link đến từng đội.
3. **Xử lý dữ liệu từng đội**
    - Với mỗi đội, tải trang của đội đó rồi song song (multi-thread) phân tích các bảng thống kê:
        - stats_standard_9: thông tin cơ bản (tên, tuổi, vị trí, số phút, bàn, kiến tạo…)
        - stats_keeper_9: thủ môn
        - stats_shooting_9: dứt điểm
        - stats_passing_9: chuyền bóng
        - stats_gca_9: tạo cơ hội
        - stats_defense_9: phòng ngự
        - stats_possession_9: kiểm soát bóng
        - stats_misc_9: các chỉ số khác
4. **Lưu trữ dữ liệu**
    - Dành cho mỗi cầu thủ một cấu trúc dữ liệu nhiều cấp (dictionary of dicts) để gom nhóm theo loại chỉ số.
    - Bỏ cầu thủ không đủ 90 phút thi đấu.
5. **Xuất CSV**
    - Sắp xếp theo tên cầu thủ, rồi ghi ra file results.csv.

Bài 2

File code: Bai_2_top_bottom.py, Bai_2_median_mean_std.py, Bai_2_Histogram.py, Bai_2_the_best_team.py

Ý tưởng:

- Bai_2_top_bottom.py:

1. **Đọc và làm sạch dữ liệu**
    - Dữ liệu được load từ file CSV (results.csv), thay thế các giá trị "N/a" bằng NaN.
2. **Chuyển đổi tuổi**
    - Cột Age ở dạng "năm–ngày" (ví dụ "25-120") được parse sang số thực (số năm cộng phần ngày/365.25) để dùng cho tính toán.
3. **Phân loại cột**
    - Xác định các cột "nhận dạng" (Name, Team, Nation, Position, AgeString) và các cột số liệu thống kê còn lại.
4. **Chuyển đổi kiểu dữ liệu**
    - Chuyển các cột thống kê thành số (float), với lỗi ép kiểu thành NaN.
5. **Tạo báo cáo Top/Bottom 3**
    - Với mỗi chỉ số (metric), lọc bỏ dòng thiếu dữ liệu, tìm 3 giá trị lớn nhất (TOP 3) và nhỏ nhất (BOTTOM 3).
    - Định dạng mỗi khối đầu ra: header, danh sách TOP 3, danh sách BOTTOM 3, rồi ngăn cách.
6. **Lưu file**
    - Ghi toàn bộ nội dung báo cáo ra file text (top_3.txt).

- Bai_2_median_mean_std.py:

1. **Chuyển đổi và làm sạch dữ liệu**
    - Đọc file CSV đầu vào vào DataFrame, thay thế các giá trị 'N/A' bằng NaN.
    - Chuyển cột Age từ chuỗi "năm–ngày" (ví dụ "25-120") sang số năm thực (năm + ngày/365.25).
2. **Phân loại cột**
    - Xác định danh sách các cột không phải số liệu thống kê (Name, Nation, Team, Position).
    - Lấy danh sách các cột số liệu thống kê (kiểu numeric).
3. **Tính toán các chỉ số tổng hợp**
    - Với toàn bộ dữ liệu (group_col=None): tính mean, median, std cho mỗi cột thống kê.
    - Với phân nhóm theo đội (group_col="Team"): tính các chỉ số tương tự cho từng đội.
4. **Xuất báo cáo**
    - Kết hợp thống kê chung (global) và theo đội (team) thành một

DataFrame.
- Ghi kết quả ra file CSV (results2.csv).

- Bai_2_Histogram.py

1. **Đọc dữ liệu**
   - Load bảng results.csv vào DataFrame df.
2. **Chọn các chỉ số**
   - Định nghĩa hai nhóm chỉ số: tấn công (goals, assists, sca) và phòng ngự (tackles, interceptions, recoveries), rồi gom vào selected_stats.
3. **Tạo thư mục lưu ảnh**
   - Tạo (nếu chưa có) thư mục team_histograms để lưu biểu đồ.
4. **Vẽ histogram cho tất cả cầu thủ**
   - Dùng matplotlib vẽ 6 biểu đồ histogram (2 hàng × 3 cột) tương ứng mỗi chỉ số trong selected_stats, với 20 bins, lưu ra file all_players_hist.png.
5. **Vẽ histogram cho từng đội**
   - Lặp qua từng giá trị khác biệt ở cột Team.
   - Cho mỗi đội, vẽ 6 biểu đồ histogram tương tự (nhưng 10 bins, màu khác), lưu thành file riêng đặt tên theo đội (thay khoảng trắng, ký tự đặc biệt).

- Bai_2_the_best_team.py

1. **Đọc và kiểm tra dữ liệu**
   - Load file CSV vào DataFrame, báo lỗi nếu không tìm thấy.
2. **Tính toán và in kết quả theo chỉ số**
   - Lấy các cột "Mean of …" (giá trị trung bình của từng chỉ số).
   - Với mỗi chỉ số: xác định đội có giá trị trung bình cao nhất, in ra console với màu sắc và biểu tượng, đồng thời cộng vào "Weighted Total Score" với trọng số định sẵn (Goals 1.2, Possession 0.8…).
3. **Xác định đội xuất sắc toàn diện**
   - Tính "Weighted Total Score" cho mỗi đội, chọn đội cao nhất và in kết quả tổng hợp.
4. **Lưu báo cáo chi tiết**
   - Gộp bảng phân tích từng chỉ số với thông tin đội, xuất ra file CSV Detailed_Premier_League_Analysis.csv.

Bài 3

File code: Bai_3.py

Ý tưởng:

1. **Chuẩn bị dữ liệu**
    - Đọc file results.csv rồi chỉ giữ các cột số (df_numeric).
    - Dùng SimpleImputer thay thế giá trị thiếu bằng giá trị trung bình của cột.
    - Chuẩn hóa dữ liệu (zero mean, unit variance) với StandardScaler.
2. **Xác định số cụm tối ưu**
    - Lặp tới k từ 2 đến 10: chạy KMeans, lưu giá trị **Inertia** (độ "khít" trong Elbow Method) và **Silhouette Score** (đánh giá chất lượng phân cụm).
    - Vẽ hai biểu đồ Elbow và Silhouette để chọn k phù hợp.
3. **Phân cụm với k đã chọn**
    - Chọn best_k = 3, chạy lại KMeans để gán mỗi cầu thủ vào 3 cụm.
4. **Giảm chiều và trực quan hóa**
    - Áp dụng PCA xuống 2 thành phần chính để dễ vẽ.
    - Vẽ scatter plot theo PC1 và PC2, tô màu theo nhãn cụm, và lưu ảnh.