

Critérios de teste utilizados

Para testar o projeto, um arquivo de teste foi utilizado, rodado usando pytest.

O arquivo test_fis.py contém os testes para os 4 módulos, e foi criado usando uma liga de NBA Fantasy como exemplo.

Os critérios utilizados para os testes tiveram o intuito de conferir se cada módulo rodou corretamente, ou seja, o setup, a extração dos dados, a organização dos dados e a visualização dos dados foram executados corretamente com os dados da liga de exemplo.

Módulo SetUp:

O SetUp é inicializado a partir de informações para o tipo de esporte, o ano e o identificador de uma liga específica.

Para verificar se o objeto da classe SetUp foi inicializado corretamente foram usados 3 testes:

- Test_setup_league_name
- Test_setup_number_teams
- Test_setup_league_type

test_setup_league_name

Verifica se o setup foi inicializado com o nome da liga correspondente ao código passado.

test_setup_number_teams

Verifica se o setup foi inicializado com o número de times correspondente ao código passado.

test_setup_league_type

Verifica se o setup foi inicializado com o tipo de liga correspondente ao código passado.

Módulo Extract_Data

Os objetos criados aqui são das sub-classes de Extract_Data, e são inicializados a partir de um objeto da classe SetUp, os dados correspondentes são extraídos por cada objeto criado, os dados das semanas da liga correspondente, pelo Extract_Week_Data e os dados de jogadores pelo Extract_Player_Season_Data.

Para verificar se o objeto de Extract_Data foi inicializado corretamente, 3 testes foram realizados:

- test_week_extraction_player_count
- test_week_extraction_week_1
- test_player_extraction_2017_season

test_extraction_player_count

Verifica se o objeto criado da classe Extract_Week_Data contém o atributo "players_per_team" com o valor correspondente à configuração da liga usada para teste.

test_extraction_week_1

Carrega o arquivo .json com os dados extraídos pela classe `Extract_Week_Data`, usando o caminho para o diretório contido como atributo do objeto, e verifica se os dados extraídos pelo objeto estão corretos. Essa checagem é feita ao observar um elemento específico do conjunto de dados extraídos, para ver se o seu valor bate com o valor correspondente para a liga.

test_player_extraction_2017_season

Carrega o arquivo .json com os dados extraídos pela classe `Extract_Player_Season_Data`, usando o caminho para o diretório contido como atributo do objeto, e verifica se os dados extraídos pelo objeto estão corretos. Essa checagem é feita ao observar um elemento específico do conjunto de dados extraídos, para ver se o seu valor bate com o valor correspondente para a liga.

Módulo Data_Organizer

O `Data_Organizer` é inicializado a partir do mesmo objeto da classe `SetUp`, lê os dados extraídos para a liga correspondente e organiza os mesmos em um objeto da classe `DataFrame` da biblioteca `pandas`.

Para verificar se o objeto de `Data_Organizer` (`Teams_Weekly_Data_Organizer`) foi inicializado corretamente, 4 testes foram realizados:

- `test_week_organizer_current_week`
- `test_week_organizer_data_output`
- `test_week_organizer_week_1`
- `test_player_organizer_2017_season_data`

test_organizer_current_week

Verifica se o objeto criado contém o atributo “`current_week`” com o valor correspondente ao objeto da classe `SetUp` usado para inicializar a classe.

test_organizer_data_output

Verifica se os atributos contendo os outputs da classe são do tipo correto, um objeto da classe `DataFrame`, da biblioteca `pandas`.

test_organizer_week_1

Verifica se o `DataFrame` criado para organizar os dados está com os valores corretos. Essa checagem é feita ao observar um elemento específico do conjunto de dados extraídos, para ver se o seu valor bate com o valor correspondente para a liga.

test_player_organizer_2017_season_data

Verifica se o `DataFrame` criado para organizar os dados está com os valores corretos. Essa checagem é feita ao observar um elemento específico do conjunto de dados extraídos, para ver se o seu valor bate com o valor correspondente para a liga.

Módulo Data_Visualization

O objeto de Data_Visualization é inicializado a partir do objeto da classe Data_Organzier com os dados estruturados, usa os dados vindos de um DataFrame da biblioteca pandas e cria visualizações a partir deles.

Para verificar se o objeto de Data_Visualization, duas etapas foram feitas, uma para a sub-classe Visualize_Teams_Week_Data e outro para a sub-classe Visualize_Teams_Season_Data.

Para verificar se o objeto da classe Visualize_Teams_Week_Data foi inicializado corretamente, 4 testes foram realizados, todos ao gerar visualizações com os dados da primeira semana:

- test_visualizer_week_1_table
- test_visualizer_week_1_table_correctedness
- test_visualizer_week_1_score
- test_visualizer_week_1_score_correctedness

test_visualizer_week_1_table

Verifica se se o atributo da classe contendo o output week_table é do tipo correto, um objeto da classe Styler, da biblioteca pandas.

test_visualizer_week_1_table_correctedness

Verifica se o output week_table foi criado com os valores corretos. Essa checagem é feita ao observar o mesmo elemento específico utilizado anteriormente do conjunto de dados extraídos, para ver se o seu valor bate com o valor correspondente para a liga.

test_visualizer_week_1_score

Verifica se se o atributo da classe contendo o output week_score é do tipo correto, um objeto da classe Styler, da biblioteca pandas.

test_visualizer_week_1_score_correctedness

Verifica se o output week_score foi criado com os valores corretos. Essa checagem é feita ao observar o mesmo elemento específico utilizado anteriormente do conjunto de dados extraídos, para ver se o seu valor bate com o valor correspondente para a liga.

Para verificar se o objeto da classe Visualize_Teams_Season_Data foi inicializado corretamente, 2 testes foram realizados:

- test_visualizer_season_table
- test_visualizer_season_score

test_visualizer_season_table

Verifica se se o atributo da classe contendo o output season_table é do tipo correto, um objeto da classe Styler, da biblioteca pandas.

test_visualizer_season_score

Verifica se se o atributo da classe contendo o output season_table é do tipo correto, um objeto da classe AxesSubplot, da biblioteca matplotlib.

Módulo Data_Analyzer

O objeto de Data_Analyzer é inicializado a partir do objeto da classe Data_Organzier com os dados estruturados, usa os dados vindos de um DataFrame da biblioteca pandas e gera as análises em cima deles.

Para verificar se o objeto da classe Analyze_Players_Data foi inicializado corretamente, 2 testes foram realizados, todos ao gerar as previsões para a temporada de 2019.

- test_analyzer_ranking_prediction
- test_analyzer_final_predictions

test_analyzer_ranking_prediction

Verifica se o atributo ranking_3 foi criado com os valores corretos. Essa checagem é feita ao observar um elemento específico da previsão, verificando se o melhor jogador previsto para a temporada de 2019 está correto.

test_analyzer_final_predictions

Verifica se se o atributo da classe contendo o output final_predictions é do tipo correto, um objeto da classe Styler, da biblioteca pandas.