

POLITECHNIKA WROCŁAWSKA
WYDZIAŁ INFORMATYKI I TELEKOMUNIKACJI



Przewidywania zdobytego złota i obrażeń na podstawie statystyk z gry League of Legends

Sprawozdanie z laboratorium

AUTOR

Jan Skibiński

nr albumu: **260460**

kierunek: **Informatyka stosowana**

13 czerwiec 2022

Streszczenie

Praca przedstawia system, który pozwala przewidzieć zdobyte złoto/zadane obrażenia w meczu gry League of Legends na podstawie statystyk (K/D/A, czas gry, zabite miniony). Dataset został pobrany z użyciem Riot API. Pobrane dane zostały następnie oczyszczone oraz użyto modelu do predykcji zdobytego w meczu złota na podstawie statystyk gracza.

1 Wstęp – sformułowanie problemu

Autor potrzebuje przewidzieć przewagę złota swoich przeciwników, względem siebie i swojej drużyny w obecnie trwającym meczu. Informacja ta nie jest wyświetlana, dopóki mecz się nie zakończy, a jest to użyteczna informacja, pozwalająca na skuteczniejsze wybieranie celu ataku.

2 Opis danych

Wielkość datasetu 100 wierszy. (Zbiór wartości jest specyficzny dla datasetu, potencjalnie wszystkie są $<0, \infty$)

Kolumna "kills" - zmienna int, określa ona ilość zabójstw zdobytych przez danego gracza w meczu. Zbiór wartości: $<1, 22>$

Kolumna "deaths" - zmienna int, określa ona ilość zgonów danego gracza w meczu.

Zbiór wartości: $<1, 13>$

Kolumna "assists" - zmienna int, określa ona ilość asyst zdobytych przez danego gracza w meczu.

Zbiór wartości: $<1, 24>$

Kolumna "minions" - zmienna int, określa ona liczbę zabitych minionów przez danego gracza w meczu.

Zbiór wartości: $<7, 215>$

Kolumna "duration" - zmienna float, określa ona czas trwania meczu (w minutach).

Zbiór wartości: $<11, 31>$

Kolumna "gold" - zmienna int, określa ona ilość złota zdobytego przez danego gracza w meczu.

Zbiór wartości: $<6638, 24300>$

Kolumna "damage" - zmienna int, określa ona ilość zadanych obrażeń wrogim bohaterom, przez danego gracza w meczu. Zbiór wartości: $<3254, 75641>$

3 Opis rozwiązania

Dane do datasetu zostaną pobrane ze strony <https://eun1.api.riotgames.com>. Dostęp do danych uzyskano za pomocą API. Baza danych została zapisana w postaci ramki danych biblioteki **Pandas**. Zawiera ona informacje o 100 ostatnich meczach gry League of Legends (tryb arurf) rozegranych przeze mnie.

Używając metody *Multiple Linear Regression* (MLR) na danych uzyskano model pozwalający na określenie złota zdobytego przez gracza w obecnie trwającej grze. Jest to dokonywanie przy pomocy modelu, który bierze jako zmienne decyzyjne: zabójstwa, śmierci, asysty, miniony, czas gry i na ich podstawie próbuje przewidzieć, ile złota gracz zdobył w tej grze. Jest to możliwe, dzięki mocnej korelacji naszych danych, wszystkie zmienne decyzyjne wpływają pośrednio lub bezpośrednio na to, ile dany gracz zdobył złota.

4 Rezultaty obliczeń

4.1 Plan badań

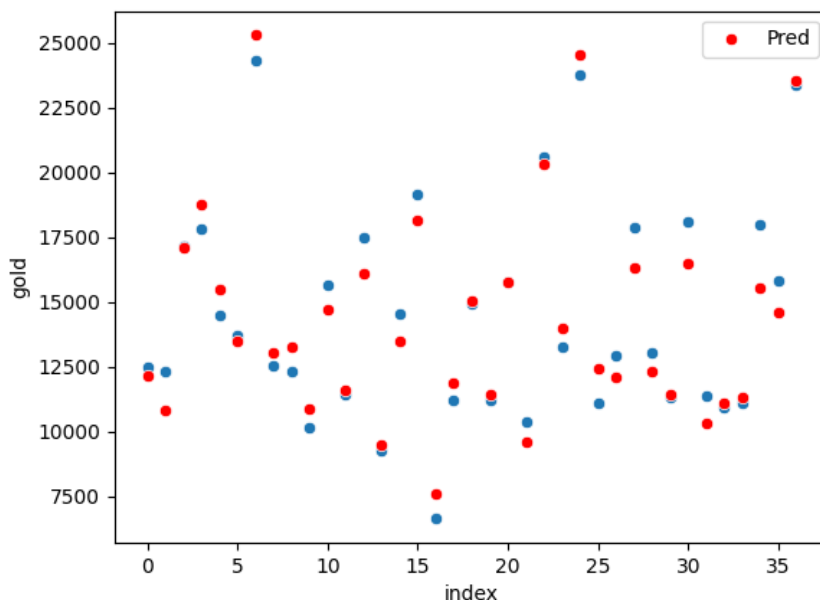
Zbiór danych zostanie podzielony na dwie części: treningową i testową w stosunku, który zwróci najlepsze wyniki (prosta funkcja decyduje o cut poincie, funkcja ta jako parametry przyjmuje minimalny rozmiar danych testowych oraz treningowych). Przed zdecydowaniem cut pointu usuniemy z danych ekstremalne przypadki (bardzo krótkie mecze)

4.2 Wyniki obliczeń

Model zdobytego złota można określić następującym wzorem:

$$\det Gold = a * kills + b * deaths + c * assists + d * minions + e * duration + f \quad (1)$$

Na rys. 1 pokazany jest wykres dopasowania modelu do danych testowych parametry (MIN-TEST=16, MINTRAINING=60, MINGAMELENGTH=11).



Rysunek 1: Dopasanie modelu do danych

Parametry wyliczone przez curvefit: $a = 415.46962063$ $b = -26.42115756$ $c = 140.22010774$ $d = 34.71285853$ $e = 377.52246582$

Oznacza to, że zgony wpływają negatywnie na zdobyte złoto, natomiast najważniejszymi parametrami są zabójstwa oraz czas gry, zabite miniony będą jako 3 najważniejsze, natomiast asysty nie mają aż tak dużego wpływu.

Podsumowując, symulacje ujawniają że wg. (1) wyliczenie złota jaki posiada nasz przeciwnik powinno być relatywnie proste. Starczy zwracać uwagę na czas gry i zabójstwa, oraz zabite miniony, w obliczeniach spokojnie można pominąć asysty i śmierci (chyba, że ma ich bardzo dużo)

Do oceny modelu użyto metryk mean-absolute-error, która wyniosła 756.72 oraz mean-absolute-percentage-error, która wyniosła 5.43%

5 Wnioski

Przedstawiona metoda pozwala na przygotowanie modelu i ten model jest wytrzymały na nowe dane. Uzyskany model daje szczególnie ciekawe dane dla gier o ekstremalnej długości (to znaczy bardzo krótkiej, albo bardzo długiej). Pozwala on zauważyć pewne znaczące zależności od tego, jak statystyki naprawdę wpływają na naszą przewagę w grze. Między innymi pozwala zauważyć, że zgon nie wpływa w znaczący sposób na naszą przewagę, natomiast wpływa on na przewagę przeciwnika.

A Dodatek

Kody źródłowe(utrzymane w konwencji języka Python wraz z instrukcjami uruchomienia) umieszczone zostały w repozytorium github:

https://github.com/Pandoriam/msid_project.