Podprojekt indywidualny - raport

Regresja logistyczna

Michał Romaszkin

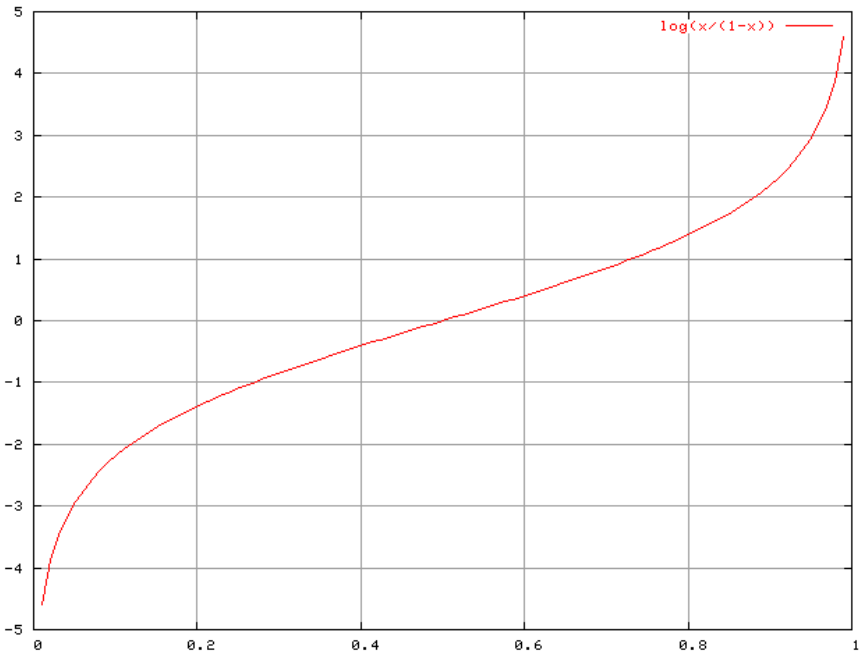
1. Regresja logistyczna

Regresja logistyczna jest szczególnym przypadkiem modelu regresji liniowej. Używa jej się w statystyce. Opiera się ona na wyrażaniu prawdopodobieństwa, tzw. „szansy” następująco:

$$Odds = \frac{p}{1-p}$$

gdzie *p* oznacza sukces. Te prawdopodobieństwo przekształca się następnie na logarytm szans za pomocą funkcji logit:

$$logit(p) = \ln\frac{p}{1-p} = \ln(\mathrm{p}) - \ln(1-p)$$



2. Implementacja

W celu użycia regresji logistycznej do nauczania maszynowego agenta w symulacji użyłem biblioteki *scikit-learn* dla języka programowania Python. W modelu zostały użyte domyślne parametry (np.: użyty algorytm do optymalizacji problemu), z wyjątkiem wybrania problemu jako „wielomianowy”.

3. Dane wejściowe

Dane znajdują się w pliku *logistic-input.txt*, który jest wygenerowany przez funkcję, która po każdym kroku agenta zapisuje do pliku aktualne otoczenie śmieciarki oraz kolejną zmianę stanu. Każdy krok jest przedstawiony w następujący sposób:

$$x_1, x_2, x_3, x_4, x_5, x_6, x_7, x_8, y$$

gdzie x_i oznacza otocznie śmieciarki („5” – ulica, „6” – dom, „7” – trawa i inne), a y oznacza zmianę stanu („0” – ruch w górę, „1” – w lewo, „2” – w dół, „3” – w prawo, „4” – brak ruchu). Dane te zostały uzyskane za pomocą algorytmu *Breadth First Search*.

4. Model

Model po otrzymaniu danych dzieli je za pomocą funkcji *train_test_split*, gdzie 25% danych użytych jest do testów, a 75% do trenowania. Każdy krok jest wybierany losowo. Trenowanie odbywa się za pomocą metody *fit()*.

5. Dane wyjściowe

W pliku *logistic-data.txt* znajduje się tabelka z przewidywanym wyborem algorytmu oraz z faktycznym wyborem. W pliku *logistic-score.txt* znajduje się celność algorytmu wywołana metodą *.score()*. Po 7 uruchomieniach wynik był w granicach 44%.