



AGH

AKADEMIA GÓRNICZO-HUTNICZA IM. STANISŁAWA STASZICA W KRAKOWIE

**WYDZIAŁ ELEKTROTECHNIKI, AUTOMATYKI,
INFORMATYKI I INŻYNIERII BIOMEDYCZNEJ**

KATEDRA INFORMATYKI STOSOWANEJ

Praca dyplomowa inżynierska

Sztuczna inteligencja na przykładzie symulacji komputerowej
Artificial intelligence in computer simulation

Autor:

Bartłomiej Konieczny

Kierunek studiów:

Informatyka

Opiekun pracy:

dr inż. Mirosław Gajer

Kraków, 2015

Oświadczam, świadomy(-a) odpowiedzialności karnej za poświadczenie nieprawdy, że niniejszą pracę dyplomową wykonałem(-am) osobiście i samodzielnie i nie korzystałem(-am) ze źródeł innych niż wymienione w pracy.

Serdecznie dziękuję ... tu ciąg dalszych podziękowań np. dla promotora, żony, sąsiada itp.

Spis treści

1. Wstęp	7
1.1. Cele pracy	7
2. Uczenie maszynowe - rozdział teoretyczny	9
2.1. Podejścia do uczenia maszynowego	9
2.1.1. Uczenie nadzorowane	9
2.1.2. Uczenie nienadzorowane	10
2.1.3. Uczenie ze wzmocnieniem	12
2.2. Podsumowanie	12
3. Implementacja	13
3.1. Opis implementacji	13
3.1.1. Wybór algorytmów	13
3.2. Symulacja graficzna	13
3.3. Wynik działania	13
4. Podsumowanie	15

1. Wstęp

W ostatniej dekadzie można zauważyć zwiększone zainteresowanie rozwiązaniami z dziedziny sztucznej inteligencji. Innowacyjne pomysły z użyciem tych algorytmów pozwalają nie tylko na interpretację ogromnych ilości danych, których człowiek nie jest w stanie przetworzyć ale również, między innymi, na rozwój autonomicznych pojazdów, jeżdżących bez nadzoru kierowcy.

Według [1] sztuczną inteligencją możemy nazwać “badanie i rozwój inteligentnych maszyn, w szczególności programów komputerowych”.

Inteligentne zachowanie agenta możemy zdefiniować, gdy agent[2]

- dostosowuje swoje zachowanie do aktualnych warunków i celów,
- ma zdolność zmiany otoczenia i celów,
- uczy się z doświadczenia,
- wykonuje odpowiednie do swoich ograniczeń akcję.

Wykorzystując powyższe definicję, sztuczną inteligencję określamy jako dziedzinę naukową zajmującą się badaniem, rozwojem i implementacją programów i maszyn wykazujących cechy inteligencji, tzn. takie które uczą się z doświadczenia, będąc zmiennym w stosunku do otaczającego ich otoczenia i warunków dążą do wykonania swoich celów uwzględniając obowiązujące je ograniczenia.

1.1. Cele pracy

Celem pracy jest opis i implementacja inteligentnego agenta. Agent wykorzystując algorytmy uczenia ze wzmocnieniem wyciąga wnioski z podejmowanych akcji i dostosowuje swoje zachowanie. Aby lepiej zaprezentować wyniki jego działania, stworzone zostanie proste środowisko graficzne, które będzie przedstawiać wyniki przeprowadzanych przez robota wyborów.

W pracy zostanie wyjaśnione pojęcie uczenia maszynowego i podstawowe podejścia do rozwiązania problemów z tej dziedziny. Dzięki temu możliwe będzie podsumowanie różnic między uczeniem ze wzmocnieniem, a pozostałymi podejściami do uczenia maszynowego.

2. Uczenie maszynowe - rozdział teoretyczny

Jednym z najistotniejszych zagadnień z dziedziny sztucznej inteligencji jest uczenie maszynowe. Uczenie maszynowe jest metodą analizy danych, która automatyzuje budowę modelu analitycznego na podstawie nauki z danych. W wielu zastosowaniach ich użycie jest znacznie bardziej efektywne od manualnego programowania, w wyniku czego uczenie maszynowe znalazło szerokie zastosowanie w informatyce i innych dziedzinach. W ostatniej dekadzie można zauważyć zwiększone użycie metod uczenia maszynowego[3].

2.1. Podejścia do uczenia maszynowego

- uczenie nadzorowane,
- uczenie nienadzorowane,
- uczenie ze wzmocnieniem,

2.1.1. Uczenie nadzorowane

Uczenie nadzorowane polega na wnioskowaniu funkcji z określonych danych treningowych. Wykorzystując dostarczone przykłady algorytmy potrafią estymować wartości danych, które mogą nie występować w podanym zbiorze wejściowym. Dzięki generalizowaniu z przykładów, metody uczenia nadzorowanego są w stanie wyznaczać przewidywane wartości na podstawie danych trenujących.

Ważną cechą danych trenujących w uczeniu nadzorowanym jest konieczność ich oznaczenia. Algorytm, aby móc szacować pożądane wartości funkcji, musi posiadać wiedzę o ich cechach.

Przykładem zastosowania algorytmów uczenia nadzorowanego jest system rozpoznawania niechcianych wiadomości w klientach pocztowych. Danymi wejściowymi są w tym przypadku kategoryzowane na pożądane lub niepożądane wiadomości e-mail. System generalizując podane mu przykłady jest w stanie zidentyfikować kolejne wiadomości i wykonać odpowiednią akcję, zależnie od preferencji użytkownika (może to być na przykład usunięcie lub przeniesienie do zdefiniowanego folderu).

Wiele różnych algorytmów uczenia nadzorowanego zostało wykorzystanych by rozwiązać problem klasyfikacji wiadomości e-mail. Użyto między innymi algorytmów k-nearest neighbor[4], Naive Bayes[5][6] czy Random Forest[7], jednak wiąże się to z istotnymi wadami[8]:

- **Wymagane oznaczenie danych testowych.** Metody uczenia nadzorowanego wymagają, aby dane trenujące były oznaczone. W przypadku klasyfikacji wiadomości e-mail, konieczne jest ich oznaczenie w zależności od tego czy są szkodliwe czy nie. Problem stwarza tutaj wielkość danych. Ilość wiadomości, która jest wymieniana w sieci jest bardzo duża. W związku z czym, żeby klasyfikacja miała sens, wymagane też jest oznaczenie sporej ilości przykładów, co nie zawsze jest możliwe i opłacalne do zrealizowania.
- **Mała liczba danych testowych.** W związku z niewielką (w stosunku do wszystkich możliwych) ilością danych trenujących, algorytm jest mało odporny na modyfikowane dane. Osoby roszylające niechciane wiadomości bardzo często będą zmieniać ich treść i strukturę, na taką, która nigdy nie pojawiła się wśród danych trenujących. Może mieć to negatywny wpływ na wynik działania algorytmu.

2.1.2. Uczenie nienadzorowane

Podobnie jak w uczeniu nadzorowanym, algorytmy uczenia nienadzorowanego wyznaczają funkcje na podstawie danych wejściowych, jednak są w stanie odkryć niewidoczne zależności między nimi. Konsekwencją wynikającą z charakterystyki algorytmów uczenia nadzorowanego jest niemożność określenia błędu lub poprawności rozwiązania. Celem działania algorytmu może być na przykład kategoryzowanie informacji (klasteryzacja).

W odróżnieniu od uczenia nadzorowanego, metody uczenia nienadzorowanego są w stanie wykryć ukryte wzorce w danych wejściowych.

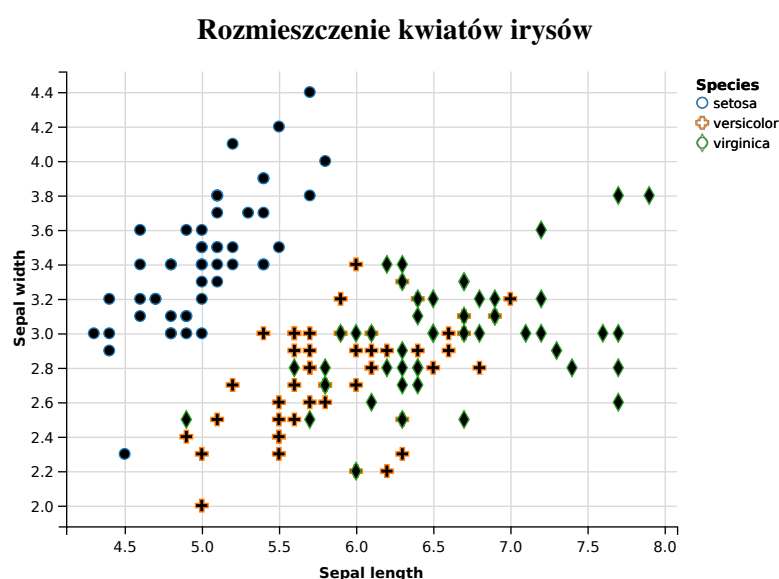
Używając jako wejścia informacji dotyczących kwiatów irysów w postaci przedstawionej na listingu 2.1, algorytmy uczenia nienadzorowanego mogą być zastosowane w celu wywnioskowania gatunku kwiatu (*setosa*, *versicolor*, *virginica*) na podstawie długości i szerokości płatków (*sepal*) i listka kielicha (*petal*).

	Sepal.Length	Sepal.Width	Petal.Length	Petal.Width	Species
1	5.1	3.5	1.4	0.2	setosa
2	4.9	3.0	1.4	0.2	setosa
3	4.7	3.2	1.3	0.2	setosa
4	4.6	3.1	1.5	0.2	setosa
53	6.9	3.1	4.9	1.5	versicolor
54	5.5	2.3	4.0	1.3	versicolor
55	6.5	2.8	4.6	1.5	versicolor
56	5.7	2.8	4.5	1.3	versicolor
101	6.3	3.3	6.0	2.5	virginica
102	5.8	2.7	5.1	1.9	virginica
103	7.1	3.0	5.9	2.1	virginica
104	6.3	2.9	5.6	1.8	virginica

Listing 2.1. Przykład danych o kwiatach irysów

Na rys. 2.1, przedstawione zostało rozmieszczenie gatunków kwiatów w zależności od długości i szerokości płatka kwiatu. Wyraźnie widać podział na dwa podstawowe klastry

- gatunek setosa,
- gatunek versicolor i virginica.



Rys. 2.1. Populacja kwiatów irysów w zależności od szerokości i długości płatka kwiatu. Źródło: Opracowanie własne

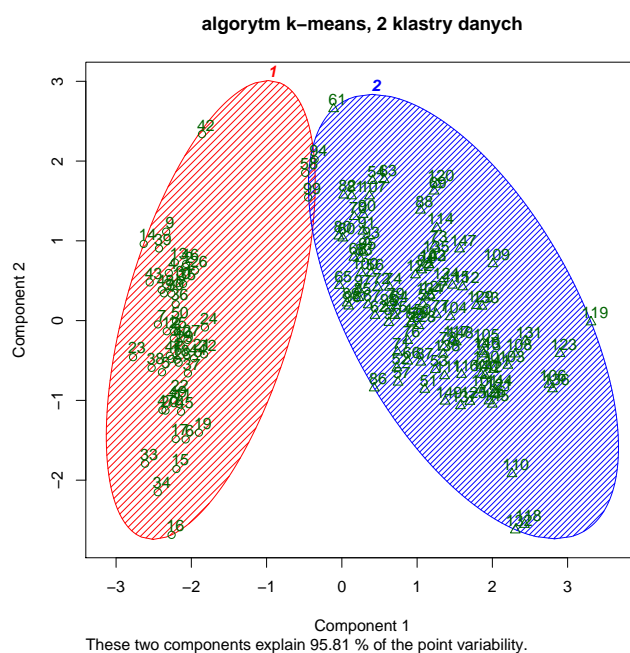
Stosując algorytmy uczenia nienadzorowanego, nie wiemy w jaki sposób klasyfikować dane trenujące. W tym celu możemy wykorzystać algorytm k-means[9], który na podstawie określonych cech grupuje dane w podaną liczbę klastrów. Poddając przedstawione dane działaniu algorytmu k-means, uzyskujemy wynik przedstawiony na rys. 2.2 oraz rys. 2.3.

Na rys. 2.2 można zaobserwować, że algorytm zadziała z dużą skutecznością rozróżniając pomiędzy dwoma gatunkami dla dwóch klastrów, tj. pomiędzy gatunkiem setosa, a gatunkami versicolor i virginica, jednak algorytm będzie miał problem rozróżnić pomiędzy gatunkami versicolor i virginica. W tym celu można zbiór podzielić na trzy klastry jak na rys.2.3, ale nie gwarantuje to wystarczającej skuteczności przydziału.

Tym samym można zaobserwować, że gatunek versicolor i virginica nie są rozróżnialne, używając przedstawionych danych wejściowych.

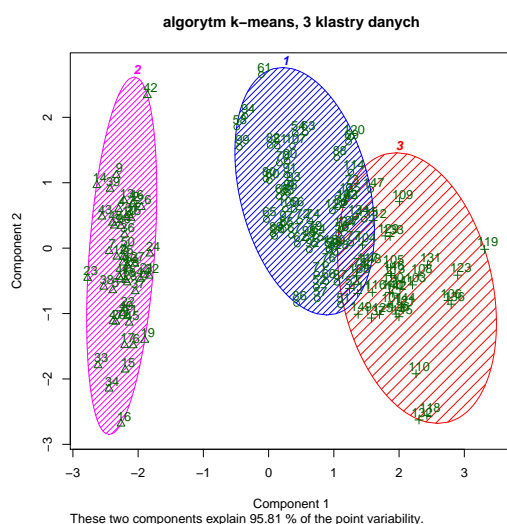
Rozważając sytuację, w której dodana by została duża liczba dodatkowych wpisów, różnych od aktualnych danych, można by było zaobserwować pojawienie się kolejnego klastra danych, co oznacza pojawienie się nowego gatunku kwiatu.

Wynik działania algorytmu k-means



Rys. 2.2. Wynik działania algorytmu klasyfikacji k-means dla 2 klastrów kwiatów irysów w zależności od szerokości i długości płatków kwiatu. Źródło: Opracowanie własne

Wynik działania algorytmu k-means



Rys. 2.3. Wynik działania algorytmu klasyfikacji k-means dla 3 klastrów kwiatów irysów w zależności od szerokości i długości płatków kwiatu. Źródło: Opracowanie własne

2.1.3. Uczenie ze wzmocnieniem

2.2. Podsumowanie

3. Implementacja

Implementacja

3.1. Opis implementacji

3.1.1. Wybór algorytmów

3.1.1.1. Q-learning

3.1.1.2. SARSA

3.2. Symulacja graficzna

3.3. Wynik działania

4. Podsumowanie

Bibliografia

- [1] John McCarthy. „What is artificial intelligence”. W: *URL: <http://www-formal.stanford.edu/jmc/whatisai.html>* (2007), s. 38.
- [2] Poole David L. i Mackworth Alan K. *Artificial Intelligence: Foundations of Computational Agents*. New York, NY, USA: Cambridge University Press, 2010. ISBN: 0521519004, 9780521519007.
- [3] Pedro Domingos. „A few useful things to know about machine learning”. W: *Communications of the ACM* 55.10 (2012), s. 78–87.
- [4] Loredana Firte, Camelia Lemnaru i Rodica Potolea. „Spam detection filter using KNN algorithm and resampling”. W: *Proceedings of the 2010 IEEE 6th International Conference on Intelligent Computer Communication and Processing*. IEEE. 2010, s. 27–33.
- [5] Muhammad N Marsono, M Watheq El-Kharashi i Fayez Gebali. „Binary LNS-based naïve Bayes inference engine for spam control: noise analysis and FPGA implementation”. W: *Computers & Digital Techniques, IET* 2.1 (2008), s. 56–62.
- [6] R Deepa Lakshmi i N Radha. „Spam classification using supervised learning techniques”. W: *Proceedings of the 1st Amrita ACM-W Celebration on Women in Computing in India*. ACM. 2010, s. 66.
- [7] Irena Koprinska i in. „Learning to classify e-mail”. W: *Information Sciences* 177.10 (2007), s. 2167–2187.
- [8] Wenjuan Li i in. „Towards designing an email classification system using multi-view based semi-supervised learning”. W: *Trust, Security and Privacy in Computing and Communications (Trust-Com), 2014 IEEE 13th International Conference on*. IEEE. 2014, s. 174–181.
- [9] Adam Coates i Andrew Y Ng. „Learning feature representations with k-means”. W: *Neural Networks: Tricks of the Trade*. Springer, 2012, s. 561–580.