28.09.2016r.

BIAI – projekt

Rok akademicki 2015/2016

Grupa Rybnik

Tytus Dragon

Marcin Andrzejewski

1. Temat projektu
   Tematem projektu była predykacja rodzaju przestępstwa popełnionego na terenie San Francisco na podstawie danych zaczerpniętych z kortotek policyjnych.
   Temat zaczęrpnięty z konkursów na stronie kaggle.com, stamtąd też pobrane były dane treningowe i testowe do realizacji zadania.

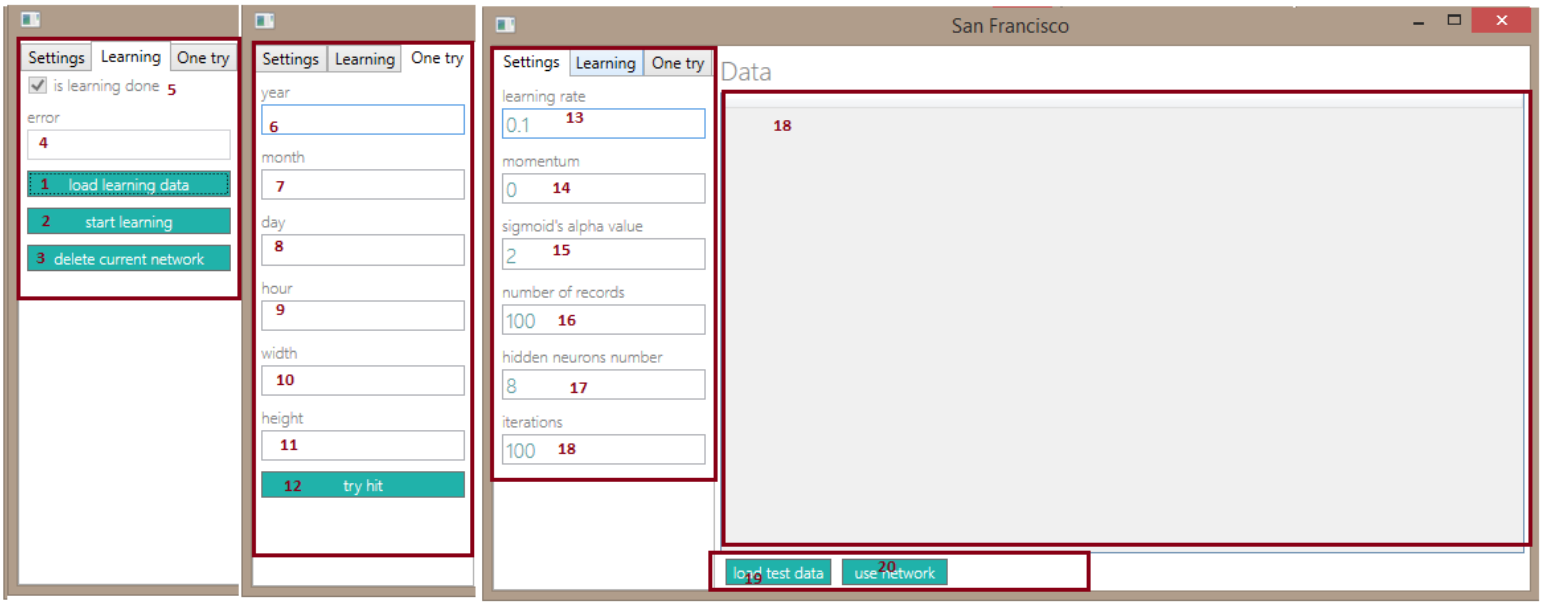
   Dates – data wydarzenia zarejestrowanego przez policję
   Category - kategoria tego wydarzenia, to co przewidujemy
   Descript – opis szczegółowy
   DayOfWeek – dzień tygodnia
   PdDistrict
   Resolution
   Address
   X - szerokośc goegr.
   Y – dł. Geogr.

   Do realizacji zadania postanowilismy korzystać z technologii C# oraz sugerowanej biblioteki aForge, która pozwala na tworzenie sieci neuronowej oraz uczenie jej algorytmem propagacji wstecznej.

   > *Propagacja wsteczna – podstawowy algorytm uczenia nadzorowanego wielowarstwowych, jednokierunkowych sieci neuronowych. Podaje on przepis na zmianę wag dowolnych połączeń elementów przetwarzających rozmieszczonych w sąsiednich warstwach sieci. Oparty jest on na minimalizacji sumy kwadratów błędów (lub innej funkcji błędu) uczenia z wykorzystaniem optymalizacyjnej metody największego spadku. Dzięki zastosowaniu specyficznego sposobu propagowania błędów uczenia sieci powstałych na jej wyjściu, tj. przesyłania ich od warstwy wyjściowej do wejściowej, algorytm propagacji wstecznej stał się jednym z najskuteczniejszych algorytmów uczenia sieci.*

   Nasza sieć posiada 6 neuronów warsty wejściowej i 39 w warstwie *output-u*.

2. Obsługa programu



1. Przycisk uruchamiający proces wczytywania danych testowych do programu, do wyświetlenia ich w *DataGrid* .
2. Przycisk uruchamiający uczenie sieci na podstawie wczytanych danych uczenia.
3. Przycisk służący do usunięcia istniejącej sieci
4. Liczony aktualny błąd uczenia.
5. Pole pokazujące, czy sieć istnieje – tj. Czy uczenie miało miejsce i sieć znajduje się w pliku.
6. Pierwsze pole typu *TextBox* służce do podania danych dla pojedynczego sprawdzenia działania sieci, w tym wypadku pole Roku.
7. Pole pojedynczego sprawdzenia – miesiąc.
8. Pole pojedynczego sprawdzenia – dzień.
9. Pole pojedynczego sprawdzenia – godzina.
10. Pole pojedynczego sprawdzenia – szerokość goegr.
11. Pole pojedynczego sprawdzenia – długość goegr.
12. Przysick wykonania przewidzenia kategori przestępstwa dla pojedynczych danych dla istniejacej sieci. Lista kategori przestępstw poniżej:
    1) ARSON
    2) ASSAULT
    3) BADCHECKS
    4) BRIBERY
    5) BURGLARY
    6) DISORDERLYCONDUCT
    7) DRIVINGUNDERTHEINFLUENCE
    8) DRUGNARCOTIC
    9) DRUNKENNESS
    10) EMBEZZLEMENT
    11) EXTORTION
    12) FAMILYOFFENSES
    13) FORGERYCOUNTERFEITING
    14) FRAUD
    15) GAMBLING
    16) KIDNAPPING
    17) LARCENYTHEFT
    18) LIQUORLAWS
    19) LOITERING
    20) MISSINGPERSON
    21) NONCRIMINAL
    22) OTHEROFFENSES
    23) PORNOGRAPHYOBSCENEMAT
    24) PROSTITUTION
    25) RECOVEREDVEHICLE
    26) ROBBERY
    27) RUNAWAY
    28) SECONDARYCODES
    29) SEXOFFENSESFORCIBLE
    30) SEXOFFENSESNONFORCIBLE
    31) STOLENPROPERTY
    32) SUICIDE
    33) SUSPICIOUSOCC
    34) TREA
    35) TRESPASS
    36) VANDALISM
    37) VEHICLETHEFT
    38) WARRANTS
    39) WEAPONLAWS

13. Pole współczynnika uczenia, jeden z parametrów do uczenia sieci.

    Algorytm wstecznej propagacji błędu wykazuje silną zależność pomiędzy przebiegiem procesu trenowania sieci, a wartością współczynnika uczenia. Zbyt mała jego wartość powoduje zwiększenie się liczby potrzebnych iteracji i tym samym czasu wymaganego na zakończenie treningu. Algorytm ma ponadto w takim przypadku tendencję to wygasania w minimach lokalnych funkcji celu. Ustalenie z kolei zbyt dużej wartości kroku grozi wystąpieniem oscylacji wokół poszukiwanego minimum, prowadzącej ostatecznie do zatrzymania procesu uczenia bez osiągniecie wymaganej wartości funkcji celu.

14. Współczynnik bezwłaności uczenia (momentum).
    Podobnie jak w przypadku współczynnika uczenia, także współczynnik momentu w najprostszej postaci algorytmu wprowadzany jest jako wartość stała, niezmienna podczas trwania treningu sieci neuronowej. Rozwiązanie takie jest oczywiście krytykowane za nieefektywność, ponieważ optymalna wartość współczynnika momentu jest także uzależniona od kształtu, jaki hiperpłaszczyzna funkcji celu przybiera w otoczeniu punktu wyznaczonego przez wektor bieżących wartości wag. Dla obszarów płaskich, gdzie proces uczenia wyraźnie zwalnia, duża wartość momentu pozwala mu nabrać rozpędu i zmniejszyć liczbę wykonywanych iteracji, unikając jednoczenie stopowania w napotykanych po drodze płytkich minimach lokalnych. Z kolei w przypadku wąwozu stosunkowo niewielka, ale znacząca wartość może spełniać rolę filtru tłumiącego oscylacje.
15. Pole wspólczynnika nachylenia sigmoidy użytej jako funkcja aktywacji.
16. Pole odpowadające za ilość badanych rekordów.
17. Liczba neuronów w warstwie ukrytej sieci nueronowej.
18. Ilość iteracji jakie sieć ma wykonać podczas cyklu.
19. Wprowadzenie danych testowych.
20. Przycisk umożliwiajacy przetestowanie danych zawartych w *DataGrid*.

3. Algorytm
    - Pierwszy etap to wczytanie z pliku csv. danych treningowych.
    - Jeśli plik deifniujący sieć neuronoą nie został wcześniej stworzony wykonuje się proces uczenia z wykorzystaniem algorytmu propagacji wstecznej z parametrami podanymi w zakładce *settings*, a następnie sieć jest zapisywana na dysku.
    - Szerokośc, długośc geograficzna oraz dzień tygodnia, godzina, dzień miesiąca, numer miesiąca trafiają do 6 neuronów wejściowych.
    - Na podstawie ustalonych wag zostaje wytypowany rodzaj przestępstwa przedstawiony w postaci liczby (np.napad w zbiorze wyjscowym o postaci Włamanie-napad-kradzież -> miałby postać 0-1-0, kradzież 0-0-1)
    - Wygenerowana sieć jest zapisywana do pliku *siec.bin*

4. Wnioski i obserwacje

    Najmniejszy błąd jaki udało nam się uzykać (~0,41) uzyskaliśmy przy przetwarzaniu dużej ilości danych. Pozwoliło to na 38/100 poprawnych wyników z zestawu, na którym testowaliśmy sieć.

    Neurony w warstwie ukrytej (ich liczba) nie miały większego znaczenia, ponieważ błąd był zawsze zbliżony to średniej.