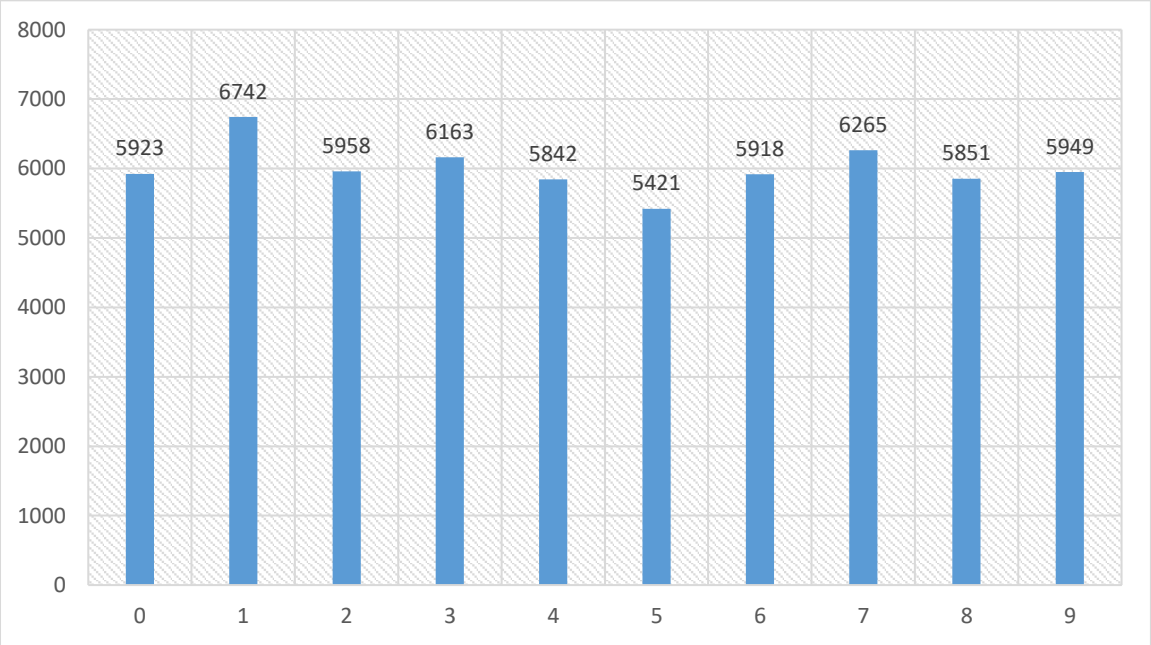
Rozpoznawanie odręcznie pisanych liczb

1) Opis problemu

Celem projektu jest stworzenie sieci neuronowej, interpretującej obrazy przedstawiające odręcznie napisane cyfry. Sieć ma za zadanie prawidłowe rozpoznanie zapisanej cyfry. W tym celu wykorzystano dataset MNIST zawierający 70000 obrazów w rozdzielczości 28x28 pixeli. Z czego 60000 obiektów stanowi zbiór treningowy a pozostałe 10000 zbiór testowy. W zbiorze każdemu obrazowi przypisana jest odpowiadająca mu cyfra.

Histogram przedstawiający ilość wystąpień każdej z cyfr



2) Wykorzystane technologie i algorytmy

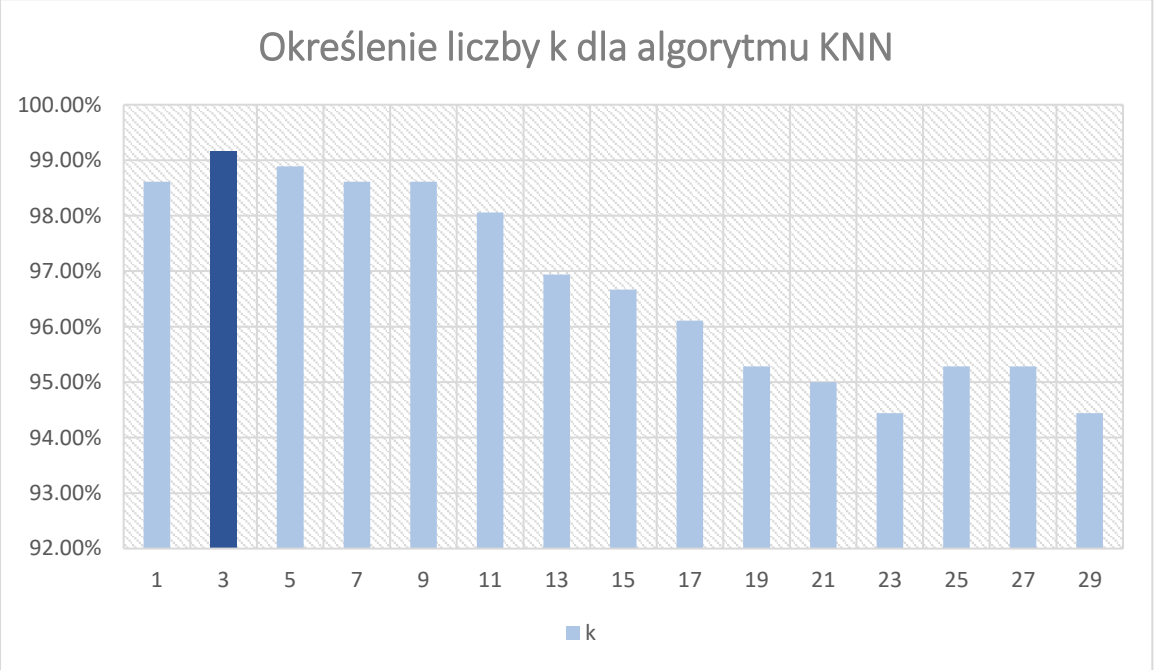
Implementacja programu napisana jest w języku Python z wykorzystaniem biblioteki Scikit-Learn. Rozpoznawanie obrazów odbywa się z wykorzystaniem algorytmu KNN (K-Nearest-Neighbours), który należy do grupy nadzorowanych algorytmów uczenia maszynowego. Jest nieparametrycznym algorytmem uczenia, co oznacza, że nie zakłada niczego na temat podstawowych danych. Algorytm opiera się na prostej idei przewidywania nieznanych wartości poprzez dopasowanie ich do najbardziej podobnych znanych wartości.

3) Przebieg procesu rozpoznawania

W naszym modelu początkowo wybieramy proporcje testowanych danych względem walidowanych – na początek przyjeliśmy, że 75% będzie do uczenia i 25% do testów.

Następnie w algorytmie KNN, trzeba ustalić k – ustalona z góry liczba najbliższych do zbioru uczącego obserwacji. Ustalamy ją za pomocą metody bilbioteki sklearn KNeighborsClassifier oraz wybierając ze zbioru nieparzystych liczb całkowitych w przedzial [1:30]. Najbliższe obserwacje sprowadzają się do minimalizacji pewnej metryki , mierzącej odległość pomiędzy wektorami zmiennych objaśniających dwóch obserwacji.

Wynik prezentuje się następująco



Jak widać na histogramie największą dokładność 99.17% miało k o wartości 3.

Biblioteka sklearn pozwala nam również wygenerować raport klasyfikacyjny, który służy do pomiaru jakości prognoz z algorytmu klasyfikacji. Metoda prezentuje 4 podstawowe współczynniki pomiaru

- Precision – dokładność pozytywnych prognoz
- Recall – ułamek pozytywnych obrazów, które zostały poprawnie zidentyfikowane
- F1 score - ważona średnią harmoniczną precyzji i przywołania, tak że najlepszy wynik to 1.0, a najgorszy 0.0

Wyniki prezentują się następująco

Digit	Precision	Recall	F1 score
0	1.00	1.00	1.00
1	0.97	1.00	0.98
2	1.00	1.00	1.00
3	1.00	1.00	1.00
4	0.98	1.00	0.99
5	0.98	0.98	0.98
6	1.00	1.00	1.00
7	1.00	0.97	0.99
8	1.00	0.97	0.99
9	0.95	0.95	0.95

4) Testowanie wyuczonego modelu

Uruchomiliśmy nasz wyuczony model na wszystkich testowych obrazach ze zbioru MNIST (10000) i otrzymany wynik dał nam całościową precyzje wynoszącą 98.75%.

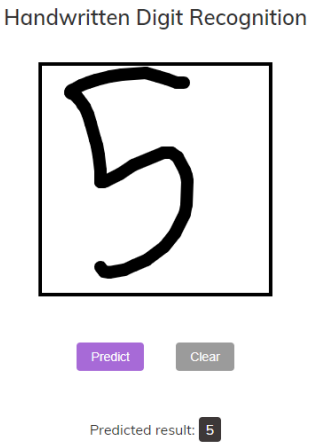
Poniższa tabelka przedstawia precyzje dla poszczególnych cyfr

Digit	Wrongly Predicted	Test amount	Precision
0	0	949	100%
1	0	797	100%
2	0	919	100%
3	0	953	100%
4	0	1227	100%
5	27	1252	97.84%
6	0	920	100%
7	24	974	97.53%
8	31	847	96.34%
9	43	1162	96.29%

5) Stworzenie wersji webowej

Na koniec stworzyliśmy wersje web'ową, która pozwala narysować odręcznie cyfrę, a wyuczony przez nas model spróbuje ją rozpoznać.

Wygląd aplikacji



Aplikacja dostępna jest pod adresem https://flask-hdr.herokuapp.com/

Wykonali

Piotr Wadycki, Kamil Wereszko