



Praca dyplomowa

Zastosowanie sztucznych sieci neuronowych w określaniu rodzaju gruntu na podstawie badań sondą statyczną CPTu

Gliwice, czerwiec 2017

*Pragnę podziękować Jędrzejowi Wierzbickiemu
z Uniwersytetu Adama Mickiewicza w Poznaniu
za udostępnienie danych użytych w tej pracy.*

Spis treści

1. Wstęp.....	5
2. Cel pracy	7
3. Badanie sondą statyczną CPTu.....	9
3.1. RYS HISTORYCZNY.....	9
3.2. CHARAKTERYSTYKA BADANIA CPTu	11
3.3. ZALEŻNOŚĆ MIĘDZY PARAMETRAMI CPTu A RODZAJEM GRUNTU	14
3.4. WPŁYW GŁĘBOKOŚCI.....	14
3.5. ISTNIEJĄCE METODY KLASYFIKACJI GRUNTÓW	14
3.5.1. Klasyfikacja Schmertmanna	15
3.5.2. Klasyfikacja Douglasa i Olsena.....	15
3.5.3. Klasyfikacja Robertsona.....	16
3.5.4. Klasyfikacja Marre`a z PN-B-04452:2002	17
3.5.5. Klasyfikacja Robertsona z PN-B-04452:2002.....	18
3.5.6. Klasyfikacja z projektu badawczego Nr T07E 047 30	18
4. Ogólny zarys sztucznych sieci neuronowych	21
4.1. ZARYS HISTORYCZNY	21
4.2. BUDOWA I DZIAŁANIE SZTUCZNYCH SIECI NEURONOWYCH.....	22
4.2.1. Budowa sztucznego neuronu	24
4.2.2. Funkcja aktywacji.....	25
4.2.3. Sieć neuronów	26
4.2.4. Uczenie sieci.....	27
4.3. MODELOWANIE SZTUCZNYCH SIECI NEURONOWYCHA	28
4.4. PRZYKŁADOWE ZASTOSOWANIA SZTUCZNYCH SIECI NEURONOWYCH W GEOTECHNICE.....	29
5. Dane.....	31
5.1. STRUKTURA I FILTRACJA DANYCH.....	31
5.2. WIZUALIZACJA DANYCH.....	33
5.3. PODZIAŁ I NORMALIZACJA DANYCH	36
6. Budowa sztucznej sieci neuronowej	41
6.1. BUDOWA SIECI	41
6.2. PROCES UCZENIA SIECI	44
6.3. PROCES WERYFIKACJI SIECI	45
7. Prezentacja i analiza wyników	47
7.1. DOKŁADNOŚĆ SIECI	47
7.2. SKUTECZNOŚĆ WYBRANEJ SIECI NA TLE INNYCH METOD.....	49
7.3. PROPOZYCJA NOWEJ METODY OKREŚLANIA RODZAJU GRUNTU	52

8. Podsumowanie	53
8.1. PRZYDATNOŚĆ W ROZPOZNAWANIU GRUNTÓW	53
8.2. MOŻLIWE KIERUNKU ROZWOJU	53
9. BIBLIOGRAFIA	55
9.1. PUBLIKACJE.....	55
9.2. NORMY	56
9.3. MATERIAŁY INFORMACYJNE.....	56

1. WSTĘP

Coraz częściej rozpoznanie podłoża odbywa się przez stosowanie nowoczesnych metod badawczych *in situ*. Metody te zapewniają dużą szybkość i wysoką dokładność przeprowadzonych badań, jednak sporym problemem jest interpretacja wyników. Przykładem takiego badania jest sondowanie CPTu. Badanie to pozwala między innymi na określenie budowy geotechnicznej i geologicznej podłoża, określenie rodzaju gruntu czy oszacowanie jego parametrów. Istnieje wiele metod interpretacji badania, jednak tylko osoby z dużą wiedzą i doświadczeniem są w stanie prawidłowo zinterpretować wyniki badań.

Wiele czynników uniemożliwia stworzenie uniwersalnej i dokładnej metody opracowania wyników. Dotychczas stosowane metody opierają się głównie na badaniach doświadczeniach, na podstawie których opracowano zależności w postaci nomogramów i wykresów. W tym celu zastosowano statystyczne metody analizy danych zebranych podczas badań polowych. Jednak aktualnie bardzo szybko rozwija pewna odmiana statystycznej analizy danych: sztuczne sieci neuronowe.

Sztuczne sieci neuronowe nie są nową ideą, ich początki sięgają lat czterdziestych dwudziestego wieku. Jednak przez większość czasu trudno było wskazać ich praktyczne zastosowanie. Głównie ze względu na niską moc obliczeniową komputerów oraz brak dużych baz danych, które mogłyby służyć do ich uczenia. Dziś mamy do czynienia ze znacznym wzrostem zainteresowania tą metodą. Moc obliczeniowa komputerów podwaja się co 24 miesiące zgodnie z prawem Gordona Moora (Moore 1965). Organizacja SINTEF (The Foundation for Scientific and Industrial Research) w 2013 roku podała, że 90% danych wytworzonych na świecie powstała w przeciągu poprzednich dwóch lat.

Największy naukowy oraz komercyjny rozwój sztucznych sieci neuronowych ma właśnie miejsce. W marcu 2016 roku świat obiegła informacja o wygranej programu komputerowego z osiemnastokrotnym mistrzem świata w grze o wiele bardziej skomplikowanej niż szachy - Go. Dotychczas komputery w Go grały bardzo słabo. W tej grze za każdą rundą ilość możliwych ruchów wynosi od kilkudziesięciu do nawet kilkuset. W związku z czym ilość ruchów, które musi przeanalizować komputer, aby rozwiązać Go, wynosi 2×10^{170} . Przyjmując że komputer jest w stanie przeanalizować trylion (10^{18}) ruchów na sekundę (Deep Blue, komputer który pokonał mistrza szachowego Garri Kasparowa, analizował dwieście milionów ruchów na sekundę), takie obliczenia trwałyby około sześciu miliardów lat, czyli połowę wieku wszechświata. Program, który wygrał, działał inaczej. Jego algorytmy wykorzystują sztuczne sieci neuronowe w powiązaniu z metodą "Monte Carlo tree search" do wyboru kolejnego ruchu.

Wspomniana rozgrywka między koreańskim wielokrotnym mistrzem świata Lee Sedol a programem komputerowym AlphaGo stworzonym przez firmę Google zakończyła się zwycięstwem 4-1 dla AlphaGo. Zwycięstwo to zostało uznane przez “Science Magazine” jako jedno z dziesięciu “Breakthrough of the Year 2016” (wśród nagrodzonych odkryć była między innymi obserwacja fal grawitacyjnych). Chiński gracz Ke Jie, aktualny lider rankingu Go na świecie, po trzykrotnej porażce z komputerem, stwierdził: *„Ludzie rozwijali się w dziedzinie gier przez tysiące lat – ale teraz komputery mówią nam że się mylimy. Wydaje mi się, że nikt nie jest nawet blisko poznania podstaw Go”*.

Dzięki dużemu zainteresowaniu, dziś można podać wiele innych przykładów praktycznego zastosowania sztucznych sieci neuronowych. Między innymi rozpoznawanie ręcznego pisma, prowadzenie samochodów autonomicznych, tłumaczenie tekstów pisanych i mówionych, rozpoznawanie tekstu mówionego z ruchów warg czy rozpoznawanie twarzy. Sztuczne sieci neuronowe sprawdzają się więc w dziedzinach, w których dotychczasowe metody były nieefektywne lub w ogóle nie istniały. Wszystko dzięki jednej z największych zalet sieci neuronowych: zdolności do uczenia się wyłącznie na podstawie przykładów bez znajomości jakichkolwiek wzorów czy teorii.

W związku z tym rodzi się pytanie: w jakich jeszcze dziedzinach można je zastosować? Geotechnika z pewnością jest jedną z tych dziedzin. Dotychczas z sukcesem zostały zastosowane między innymi w przewidywaniu współczynnika prekonsolidacji OCR, zjawiska upłynnienia gruntu, osiadania fundamentów bezpośrenich czy określania nośności pali. Ta praca ma na celu odpowiedzieć na pytanie czy do powyższej listy można dodać identyfikację gruntów na podstawie sondowania CPTu.

Podziękowania należą się

2. CEL PRACY

Celem niniejszej pracy jest stworzenie i opisanie sieci neuronowej, która będzie w stanie określić rodzaj gruntu na podstawie badania sondą statyczną CPTu. W ramach pracy powstanie aplikacja, za pomocą której zostaną zbudowane i analizowane sieci neuronowe. Aplikacja będzie wykonana od podstaw w języku programowania Python korzystając z biblioteki programistycznej TensorFlow udostępnionej na licencji wolnego oprogramowania Apache 2.0. Do uczenia sieci zostanie wykorzystana baza danych powstała z wyników sondowań CPTu oraz skorelowanych z nimi wierceniami udostępniona przez Wierzbickiego z Uniwersytetu Adama Mickiewicza, na podstawie których określono rodzaju gruntu. Modyfikując budowę i rozmiar sieci wyznaczy się jej optymalna struktura o największej dokładności działania. Zakres pracy obejmuje również zebranie dostępnych analitycznych metod określania rodzaju gruntu na podstawie sondowań CPTu i porównanie ich dokładności z dokładnością sieci neuronowej.

Praca składa się z dwóch części: ogólnej i praktycznej. W pierwszej części – rozdziały 3. i 4. – zawarte są informacje o badaniach sondą CPTu i wybrane istniejące metody interpretacji tych badań oraz podstawowe informacje o tym, w jaki sposób sieci neuronowe uczą się, działają i jak można je modelować.

W drugiej części – rozdziały od 5. do 7. – zawarto informacje o bazie danych gruntów i jej pochodzeniu, strukturze opisywanej sieci neuronowej oraz wyniki jej testów.

W podsumowaniu zawarto porównanie dokładności między siecią neuronową a innymi metodami oraz możliwy kierunek rozwoju metody klasyfikacji gruntu opartej na sztucznych sieciach neuronowych.

3. BADANIE SONDĄ STATYCZNĄ CPTU

Do rozwiązywania zaawansowanych problemów geotechnicznych niezbędna jest dokładna identyfikacja charakterystyki podłoża. Dotychczas najczęstszą metodą określenia parametrów gruntów w podłożu było wykonanie wierceń i badań makroskopowych, na podstawie których określa się rodzaj i stan gruntu. Pozostałe parametry, w tym parametry wytrzymałościowe, określane są na podstawie badań laboratoryjnych i korelacji normowych. Takie podejście jest niewystarczające do rozwiązywania zaawansowanych problemów geotechnicznych. Niebędne jest zastosowanie innych metod rozpoznania podłoża. W nowej normie geotechnicznej (PN-EN 1997-2:2009) dotyczącej rozpoznania i badania podłoża gruntowego opisano dziewięć popularnych badań polowych. Na pierwszym miejscu, jako jedno z podstawowych, opisano badania sondą CPTu. Metoda ta cechuje się relatywnie niskim kosztem, prostotą wykonania, ciągłością oraz dużą dokładnością. Te zalety, wraz szerokim zakresem parametrów gruntu, jakie można określić, sprawiają że badanie to zyskuje dużą popularność na polskim rynku. Podstawowe informacje o tym badaniu, jak i sposoby klasyfikacji gruntu są przedmiotem tego rozdziału.

3.1. Rys historyczny

Pierwsze użycie zmormalizowanych penetrometrów do badań gruntu pochodzi z Holandii z 1932 roku. Pierwszym urządzeniem był aparat Barentsena, od nazwiska Pietera Barentsena, który jako pierwszy użył tej metody do testowania podłoża pod budowę drogi w pobliżu miasta Gouda. Aparat ten składał się z żerdzi zakończonej stożkową końcówką, która była umieszczona w zewnętrznej rurze obsadowej. Badanie było wykonywane poprzez wciskanie całego zestawu w grunt mierząc opór wciskania. W ten sposób uzyskując połączony opór wciskania stożka i opór tarcia gruntu o pobocznice rury obsadowej. Następnie, co określone zagłębienie, wciskano na niewielką głębokość wyłącznie stożkową końcówkę (bez rury obsadowej, która była unieruchomiona na ten czas) aby uzyskać wyłącznie opór pod stożkiem. Różnica między pomiarami była oporem tarcia rury obsadowej o grunt. Wciskanie tej sondy odbywało się ręcznie, natomiast balastem była tylko i wyłącznie waga operatora, przez co badanie miało zastosowanie tylko w gruntach słabych do niewielkich głębokości rzędu kilku metrów.

Aby zwiększyć głębokość badań należało zwiększyć siłę użytą do wciskania sondy – zwiększyć balast. Pierwsze takie badanie przeprowadzono już w 1935 roku, gdy uzyskano 10 ton siły wciskającej bez użycia hydrauliki. Balast zrealizowano przez wykonanie wykopu o wymiarach 2 m x 3 m i 1 m głębokości. Na jego dnie ułożono drewnianą konstrukcję, która następnie była zasypaana. Poprzez zakotwienie całego urządzenia do tej konstrukcji jak pokazano na rysunku 3-1 balastem było 6 m³ gruntu która na niej spoczywała. Przygotowanie

tego badania trwało nawet do 3 dni, jednak dzięki temu zabiegowi zwiększono głębokość penetracji i zakres objętych nimi gruntów, a samo badanie umożliwiło po raz pierwszy w historii oszacować nośności drewnianych pali w inny sposób niż przez próbne obciążenie.



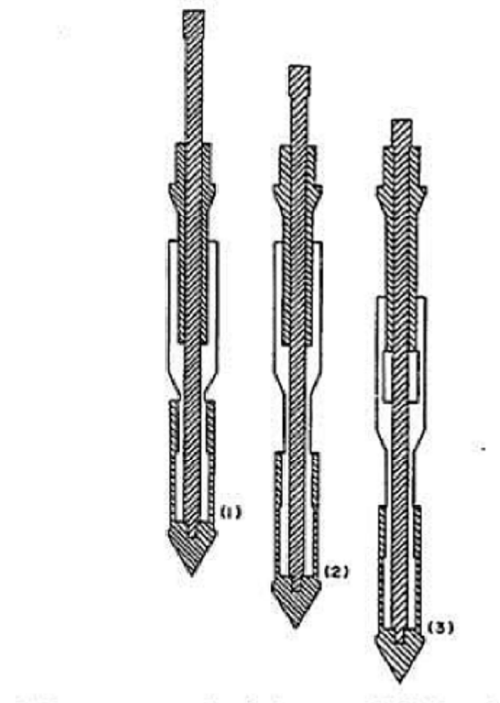
Rys. 3-1. Ręczne wciskanie sondy z zakotwieniem urządzenia w gruncie (www.gouda-geo.com)

Później ulepszono metodę stosując napęd hydrauliczny, a kotwienie realizowano poprzez wkręcanie w grunt świrdrów ślimakowych do głębokości kilku metrów bez naruszania struktury gruntu. Na początku lat 50-tych pojawiły się pierwsze urządzenia montowane na pojazdach ciężarowych o masie kilkunastu ton nie wymagające kotwienia, dzięki czemu znacznie skrócono czas wykonywania badania.

Inną ważną i rozwijaną kwestią był kształt i budowa wciskanej sondy. W 1950 roku Begemann wprowadził nową końcówkę (rys 3-2). Podstawową zaletą tej końcówki było wyeliminowanie błędu wynikającego z przedostawania się ziaren piasku między żerdź a rurą obsadową oraz umożliwianie oddzielnego pomiaru tarcia tulei ciernej i oporu pod stożkiem. Opracowana końcówka mierzyła opór tulei ciernej jedynie na krótkim odcinku ponad stożkiem. Begemann na podstawie swoich badań opublikował również wykresy, pozwalające określić nośność konkretnych rodzajów pali w zależności od oporu tarcia wynikającego z badania CPT.

W 1965 roku po raz pierwszy zastosowano sondę z końcówką elektroniczną. Jej powstanie wynikało z wad końcówki mechanicznej podczas badań na znaczną głębokość i dużych wartościach oporu gruntu. W takich warunkach w końcówce mechanicznej znacznie rosło tarcie między żerdzią a zewnętrzną rurą osłonową. Dodatkowo istniało ryzyko, że stożek końcówki może być wbijany nieosiowo. Mogło to doprowadzić do sytuacji, w której

rzeczywista głębokość badania była znacznie mniejsza od tej wynikającej z pomiarów. Czujniki do pomiaru ciśnienia wody w porach dodano około dziesięć lat po pojawianiu się końcówki elektrycznej.



Rys. 3-2. a) badanie sondą CPT z lat 50 wciskana ręcznie, b) końcówka Begemanna z tuleją cierną (www.gouda-geo.com)

W dzisiejszych czasach nowoczesne urządzenia montowane są na podwoziach samochodów ciężarowych, o masie całkowitej 40 ton. Dzięki temu nie ma konieczności jego kotwienia. Najczęściej stosowana jest końcówka elektryczna z pomiarem ciśnienia wody w porach, jednak wciąż używa się końcówek mechanicznych w sytuacji gdy zachodzi ryzyko jej uszkodzenia. Końcówki elektryczne używa się głównie w złożonych warunkach gruntowych, gdzie potrzebna jest duża dokładność pomiaru.

3.2. Charakterystyka badania CPTu

Badanie sondą statyczną CPTu polega na wciskaniu sondy pionowo w badany grunt ze stałą prędkością 20 mm/s. Podczas badania mierzone są: opór gruntu pod stożkiem q_c , opór tarcia gruntu o tuleję cierną f_s oraz ciśnienie wody w porach gruntu u . Dla badania z końcówkami elektrycznymi pomiary najczęściej prowadzone są w odstępach co kilka centymetrów. Na podstawie tych trzech parametrów mierzonych bezpośrednio, można obliczyć wielkości skorygowane (wzory 3-1 do 3-5), które uwzględniają budowę końcówki czy wzrost naprężenia pierwotnego z głębokością:

- skorygowany opór całkowity na stożku:

$$q_t = q_c + u_2 * (1 - a_n) \quad (3-1)$$

- współczynnik tarcia:

$$R_f = \frac{f_s}{q_c} * 100\% \quad (3-2)$$

- znormalizowany współczynnik tarcia:

$$F_r = \frac{f_s}{q_c - \sigma_{v0}} \quad (3-3)$$

- znormalizowany współczynnik ciśnienia wody w porach gruntu:

$$B_q = \frac{u_2 - u_0}{q_t - \sigma_{v0}} \quad (3-4)$$

- znormalizowany opór na stożku:

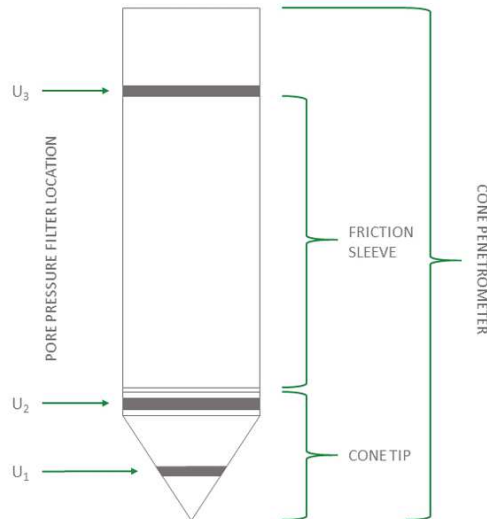
$$Q_t = \frac{q_t - \sigma_{v0}}{\sigma'_{v0}} \quad (3-5)$$

gdzie:

- a_n - współczynnik powierzchni stożka (zależy od konstrukcji sondy – podaje producent),
- u_0 – ciśnienie hydrostatyczne,
- σ_{v0} – składowa pionowa naprężenia pierwotnego całkowitego,
- $\sigma'_{v0} = \sigma_{v0} - u_0$ – składowa pionowa naprężenia pierwotnego efektywnego.

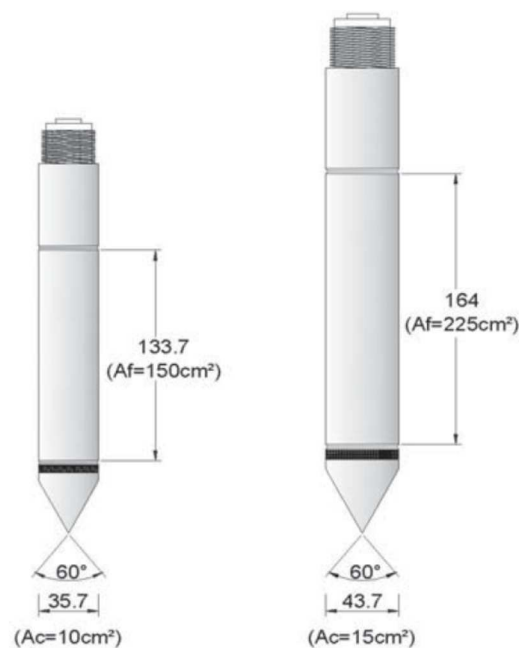
Wyniki przeprowadzonych badań przedstawiane są w formie wykresów w zależności od funkcji głębokości. Jak widać do wyznaczenia niektórych wartości skorygowanych (używanych w kilku metodach klasyfikacyjnych), niezbędna jest znajomość ciężaru objętościowego gruntu do wyznaczenia pionowego naprężenia pierwotnego. Robertson zaproponował przyjęcie ciężaru objętościowego poszczególnych warstw na podstawie wstępnej interpretacji innymi metodami nie wymagającymi posługiwania się tym parametrem.

Położenie czujników ciśnienia wody nie zostało dotychczas ustandaryzowane. Rozróżniamy trzy główne położenia: na stożku u_1 , między stożkiem a tuleją cierną u_2 (najczęściej spotykane) oraz za tuleją cierną u_3 . Ich położenie na końcówce pokazano na rysunku 3-3.



Rys. 3-3. Położenie punktów pomiaru ciśnienia wody w porach (www.lankelma.co.uk)

Najczęściej stosowanym rozmiarem końcówki są końcówki o powierzchni przekroju stożka netto 10 cm^2 (35,7 mm średnicy) oraz 15 cm^2 (średnica 43,7 mm), które pokazano na rysunku 3-4. Dostępne są także końcówki o powierzchniach od 5 cm^2 do 25 cm^2 . Większe końcówki w połączeniu z żerdziami o średnicy 36 mm są często stosowane do badań na większą głębokość dzięki znacznej redukcji tarcia na żerdziach. Należy być jednak świadomym wpływu rozmiaru stożka na wyniki pomiarów. Sprzęt użyty w badaniach CPTu powinien spełniać wymogi norm ISO 22476-1:2013 oraz PN-EN ISO 10012.



Rys. 3-4. Porównanie wymiarów końcówki o rozmiarze 10 cm^2 i 15 cm^2 (Bajda, Biliniak, Lech 2012)

3.3. Zależność między parametrami CPTu a rodzajem gruntu

Ze względu na różnicę zachowania się gruntów podczas wykonywania badania CPTu, mierzone parametry oraz charakterystyka krzywej dla różnych rodzajów gruntów są różne. Generalnie można stwierdzić, że dla gruntów niespoistych opór pod stożkiem jest znacznie większy w stosunku do tarcia na tulei ciernej (współczynnik tarcia R_f jest niski) oraz nadmiar ciśnienia wody w porach (w stosunku do ciśnienia hydrostatycznego) jest bardzo mały. Z kolei dla gruntów spoistych można zaobserwować znaczne wartości współczynnika tarcia R_f ze względu na małe wartości oporu stożka q_c . W przeciwieństwie do gruntów niespoistych obserwowany jest znaczny nadmiar ciśnienia wody w porach. Krzywe parametrów CPTu w piaskach i glinach piaszczystych mają charakter ząbkowany, postrzępiony. Dla glin i namulów krzywa jest gładka i ma lekko falisty charakter.

3.4. Wpływ głębokości

Głębokość ma wpływ na mierzone parametry. W niektórych metodach klasyfikacji gruntu uwzględnia się wpływ głębokości poprzez wprowadzenie wartości znormalizowanych, na przykład znormalizowany opór na stożku Q_t czy znormalizowany współczynnik tarcia F_r . Głębokość ma szczególnie wpływ na opór stożka i mierzone ciśnienie wody w porach gruntu. Gdy stożek wciskany jest w grunty piaszczyste opór stożka na początku rośnie wraz z głębokością aż uzyskana zostanie głębokość krytyczna, po której uzyskuje stabilną wartość. W gruntach spoistych nie obserwuje się takiej zależności. Charakterystyka krzywej dla tych gruntów jest lekko falista w związku z niejednorodnością gruntu, jednak w ogóle wartość ta rośnie w sposób liniowy wraz z głębokością.

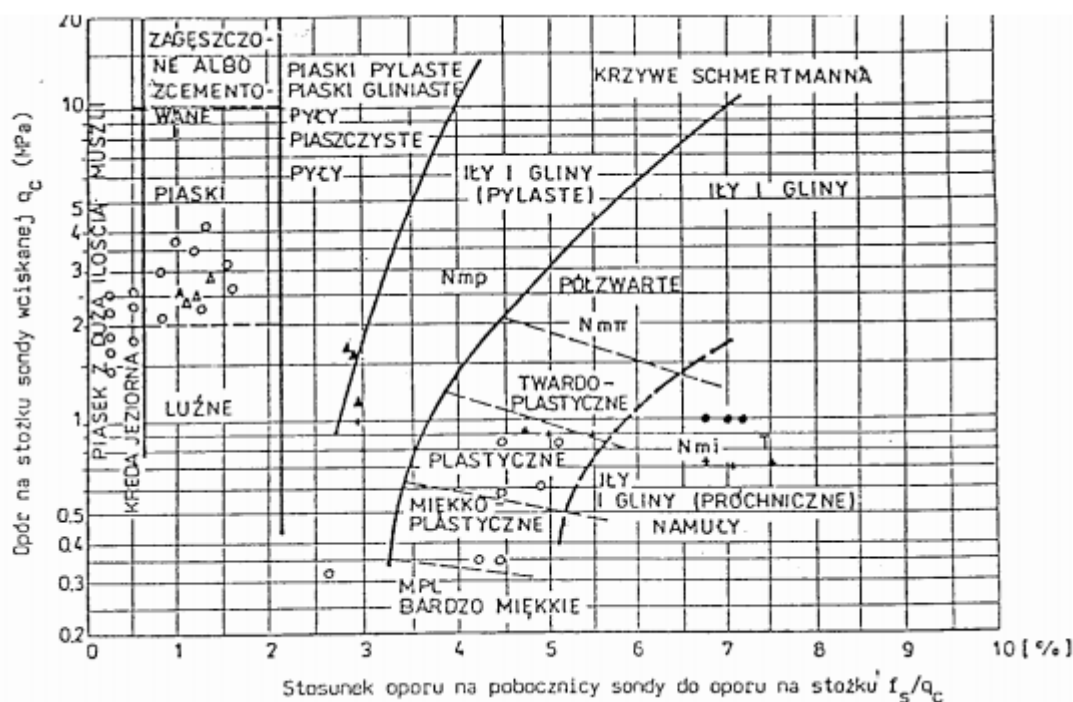
Wartość ciśnienia wody w porach jest złożeniem ciśnienia hydrostatycznego oraz nadmiarowego ciśnienia w porach generowanego przez wciskanie sondy. Ciśnienie porowe jest zależne od ścisłości oraz trwałości gruntu i rośnie wraz z głębokością. W gruntach piaszczystych i glinach piaszczystych wartości ciśnienia wody zmieniają się w znacznym stopniu, ale mają małe wartości. Natomiast w gruntach gliniastych ciśnienie rośnie w sposób liniowy wraz z głębokością.

3.5. Istniejące metody klasyfikacji gruntów

Spośród dostępnych metod klasyfikacyjnych najstarsze monogramy (Schmertmann, Marre oraz Douglas i Olsen) służą do interpretacji wyników badań sondą z końcówką mechaniczną. Najczęściej stosowanymi na świecie metodami są metody Robertsona i in. bazująca na wartościach q_t i R_f oraz Robertsona w postaci dwóch osobnych nomogramów opierająca się na wartościach Q_t , F_r oraz B_q . W Polsce najczęściej stosuje się dostosowane do polskich oznaczeń i podane w normie zmodyfikowane metody Marre'a oraz Robertsona i in. W wielu metodach (między innymi Robertsona) nie wyznacza się konkretnej nazwy gruntów, a jedynie „charakter zachowania” (SBT – *soil behaviour type*). Na kilku nomogramach można znaleźć również informacje dodatkowe, dotyczące na przykład kierunku wzrostu stopnia prekonsolidacji OCR.

3.5.1. Klasyfikacja Schmertmanna

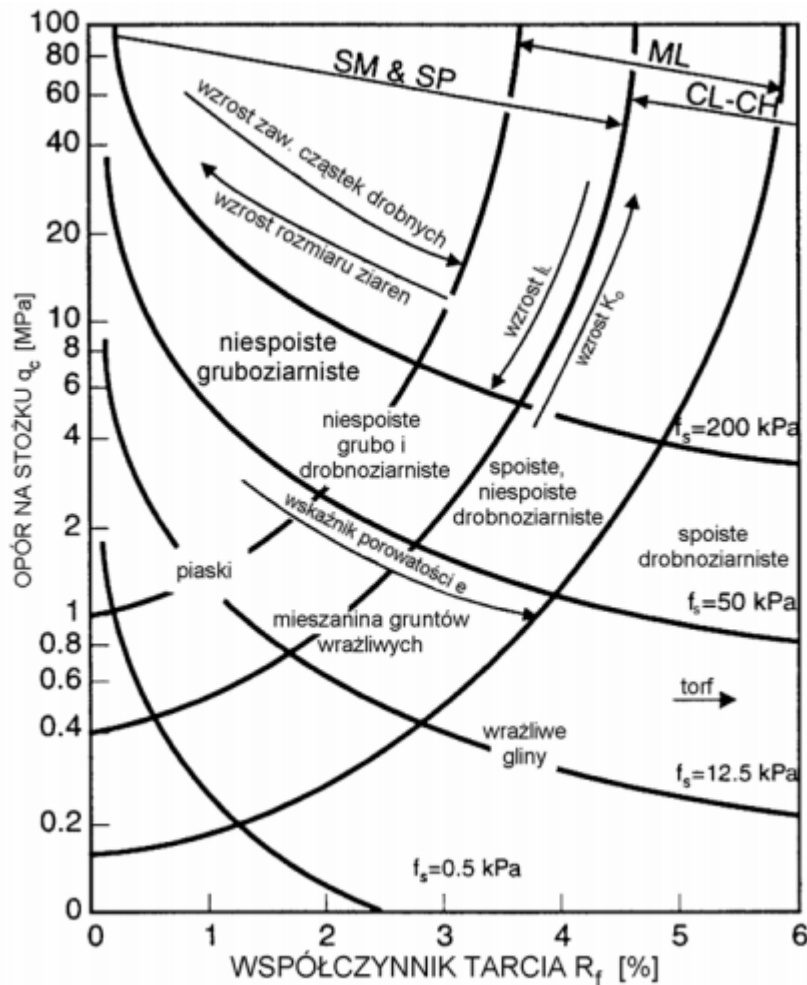
Klasyfikacja ta została zaproponowana przez Johna Schmertmanna w 1978 roku i dotyczy wyłącznie sondy z końcówką mechaniczną (kończówką Begemanna), nie uwzględnia wpływu ciśnienia wody w porach czy wpływu głębokości. Na osi pionowej (rysunek 3-5) znajduje się opór pod stożkiem w tonach na stopę kwadratową (lub w kilogramach na centymetr kwadratowy), na osi poziomej współczynnik tarcia w procentach. Klasyfikacja powstała na podstawie badań CPT w różnych miejscach w okolicach Luizjany, a także Kalifornii, Oklahomy, Utah, Arizony i Nevady. Oryginalna klasyfikacja pokazana na rysunku została później modyfikowana i kalibrowana na podstawie lokalnych doświadczeń w celu zwiększenia jej dokładność.



Rys. 3-5. Klasyfikacja Schmertmanna (Schmertmann 1978, tłumaczenie Sikora 2009)

3.5.2. Klasyfikacja Douglasa i Olsena

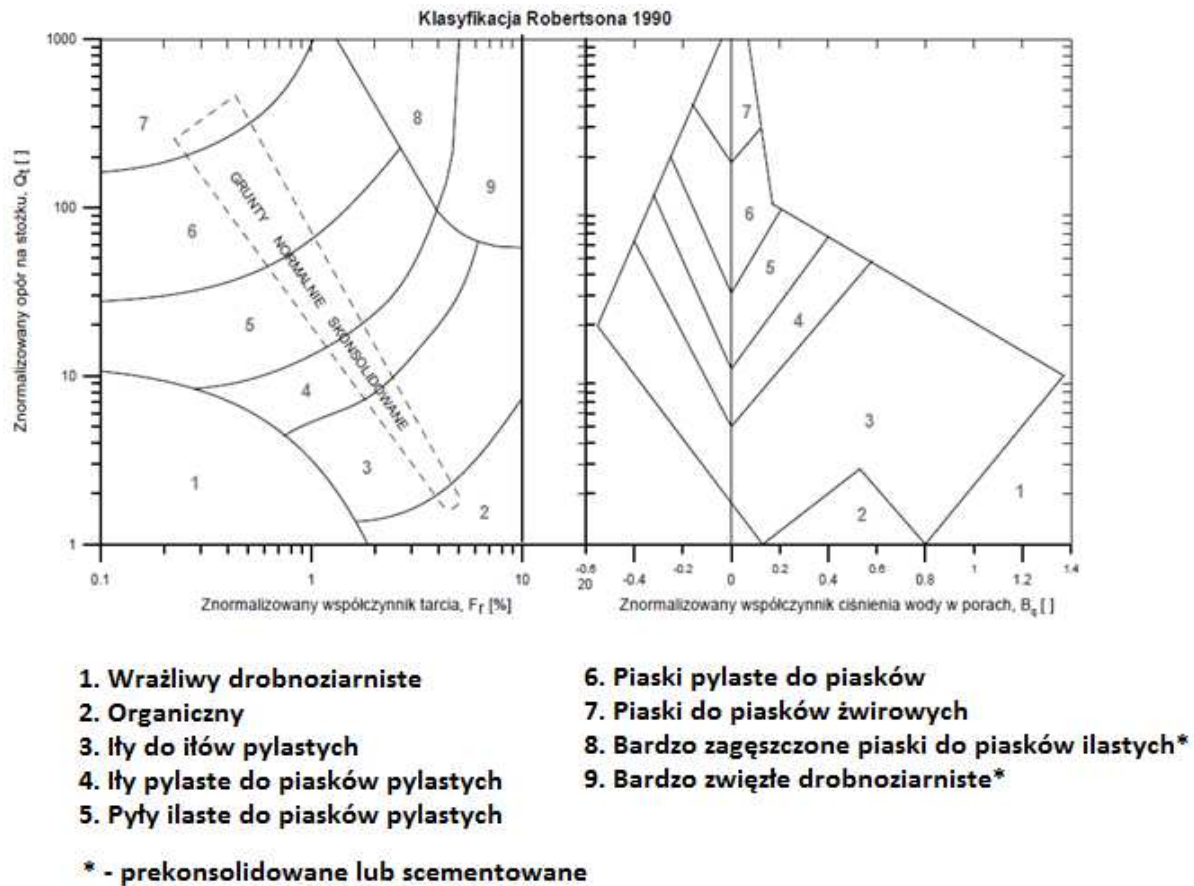
Klasyfikacja z 1981 roku uwzględniająca opór stożka, współczynnik tarcia jak i opór tarcia pobocznicy. Jest to klasyfikacja która nie wskazuje dokładnego rodzaju gruntu, jedynie jej charakter zachowania (*soil behaviour type* - SBT). Podobnie jak w przypadku klasyfikacji Schmertmanna, dane zostały zebrane z zachodniej części Stanów Zjednoczonych. Wykres (rysunek 3-6) wskazuje cztery odrębne regiony gruntów: spoistych drobnoziarnistych, spoistych i niespoistych drobnoziarnistych, niespoistych drobno- i gruboziarnistych oraz niespoistych gruboziarnistych. Klasyfikacja ta pokazuje, że wykresy klasyfikacyjne nie są w stanie precyzyjnie wskazać rodzaju gruntu, jedynie służą jako informacja o charakterze danego gruntu. Oznaczenia na wykresie: SM – piaski pylaste, SP – piaski równoziarniste, ML – ropy, CL – glina, CH - gliny twardeplastyczne, peat – torf.



Rys. 3-6. Klasyfikacja Douglasa i Olsena (Douglas, Olsen 1981, , tłumaczenie Sikora 2009)

3.5.3. Klasyfikacja Robertsona

Klasyfikacja zaproponowana przez Petera Robertsona w 1990 r., stanowi ulepszenie klasyfikacji Robertsona i in. z 1986 roku. Jest jedną z klasyfikacji, która uwzględnia ciśnienie wody w porach, typ i kształt sondy oraz jako jedną z pierwszych uwzględnia wpływ głębokości. Metoda została przedstawiona na dwóch wykresach ukazanych na rysunku 3-7, tworząc trójwymiarowy system klasyfikacji. Podczas wyznaczania wartości B_q niezbędna jest znajomość ciśnienia ustabilizowanego ciśnienia wody gruntowej u_0 . W przypadku braku informacji o ciśnieniu wody w porach, autor zaleca użycie wyłącznie lewego wykresu. Autor zaleca też, przyjmowanie ciężarów objętościowych na podstawie tablicy 3-1.



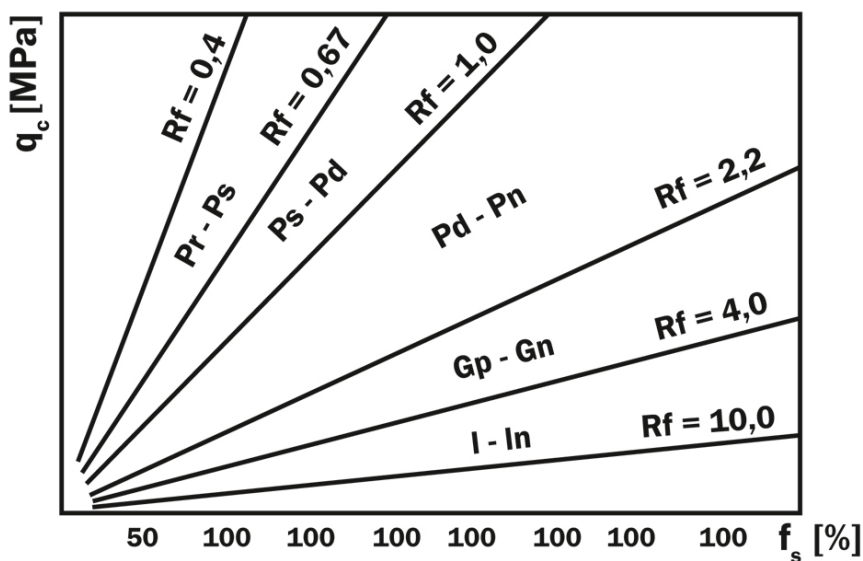
Rys. 3-7. Klasyfikacja Robertsona (Robertson 1990)

Tabela 3-1. Ciężary objętościowe przyjmowane do ustalenia naprężenia pierwotnego (Lunne, Robertson, Powel 1997)

Obszar SBT	1	2	3	4	5	6	7	8	9	10	11	12
γ (kN/m ³)	17,5	12,5	17,5	18,0	18,0	18,0	18,5	19,0	19,5	20,0	20,5	19,0

3.5.4. Klasyfikacja Marre`a z PN-B-04452:2002

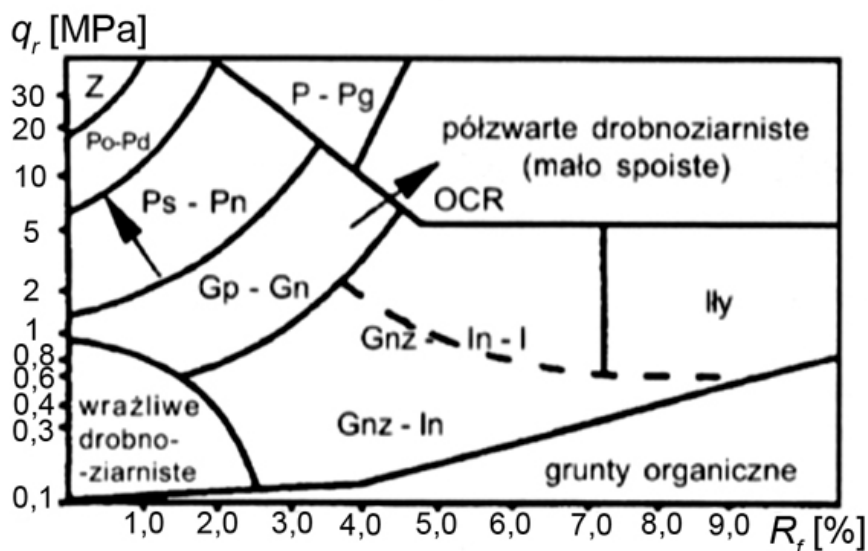
Klasyfikacja przedstawiona na rysunku 3-8 przeznaczona jest do badań sondą z końcówką mechaniczną. Powstała na podstawie klasyfikacji Marra z 1981 roku z uwzględnionymi poprawkami Zbigniewa Młynarka i zespołu z 1997 roku. Jednocześnie oznaczenia gruntu dostosowano do polskich. Na wykresie pojawił się również dodatkowy parametr R_f – współczynnik tarcia.



Rys. 3-8. Klasyfikacja Marra z PN-B-04452:2002 (PN-B-04452:2002)

3.5.5. Klasyfikacja Robertsona z PN-B-04452:2002

Klasyfikacja oparta na klasyfikacji Robertsona z 1986 roku, nie uwzględnia późniejszych zmian klasyfikacji z 1990 roku (patrz rysunek 3-9). Dostosowana do polskich nazw gruntów przez zespół Zbigniewa Młynarka w 1997 roku. W porównaniu do oryginalnej klasyfikacji Robertsona, w polskiej dodatkowo wydzielono łą i łą plastyczne od glin pylastych i piaszczystych. Klasyfikacja uwzględnia wpływ ciśnienia porowego wody, pomija wpływ głębokości.

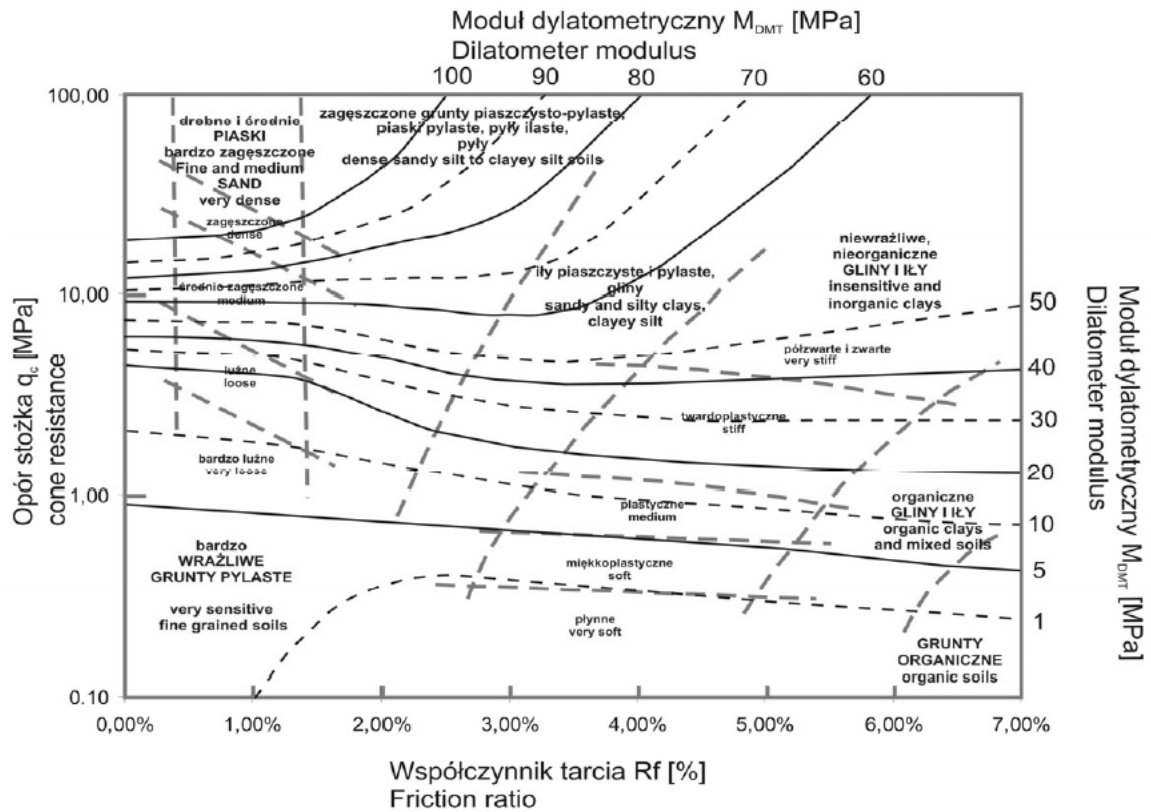


Rys. 3-9. Klasyfikacja Robertsona z PN-B-04452:2002 (PN-B-04452:2002)

3.5.6. Klasyfikacja z projektu badawczego Nr T07E 047 30

Nowa metoda klasyfikacji pokazana na rysunku 3-10 z 2008 roku, powstała w efekcie projektu badawczego Nr T07E 047 30 realizowanego w ITB. W ramach tego projektu na 30 wytypowanych poligonach badawczych na terenie Polski wykonano serię badań w celu

określenia zależności między wynikami badania CPTu i dylatometru a parametrami gruntu, w tym klasyfikacji gruntu. Dzięki tym badaniom oraz dostępnej literatury opracowano wykres klasyfikacyjny, uwzględniający warunki regionalne, w zależności od oporu pod stożkiem q_c i współczynnika tarcia R_f oraz modułu dylatometrycznego M_{DMT} .



Rys. 3-10. Klasyfikacja z projektu badawczego Nr T07E 047 30 (Godlewski 2009)

4. OGÓLNY ZARYS SZTUCZNYCH SIECI NEURONOWYCH

Sztuczne sieci neuronowe są ogólnym określeniem struktur matematycznych i programistycznych modeli służących do przetwarzania sygnału wejściowego poprzez wykonanie podstawowych operacji oraz określenia wartości sygnału wyjściowego. Są jednym ze sposobów uczenia maszynowego. Inspiracją do ich powstania był mózg człowieka. Ze względu na dużą różnorodność aktualnie trudno jest wskazać definicję sieci neuronowych. Ogólnie można ją określić jako zbiór podstawowych jednostek obliczeniowych przetwarzających informację, komunikujących się między sobą oraz pracujących równolegle. Z każdym połączeniem między podstawowymi jednostkami skojarzone są wagi, które są zmieniane podczas procesu uczenia.

4.1. *Zarys historyczny*

Początki sztucznych sieci neuronowych sięgają lat 40-tych dwudziestego wieku, gdy neurofizyk Warren McCulloch oraz matematyk Walter Pitts opublikowali pracę (McCulloch and Pitts 1943), opisującą działanie pojedynczego neuronu mózgu. W swojej pracy opracowali oni matematyczny model komórki nerwowej, który jest używany do dzisiaj. Model jest podzielony na dwie części: dodawanie ważonych wartości wejściowych, a następnie zastosowanie funkcji wyjściowej.

Następną ważną pracą w dziedzinie poznawania zasad działania mózgu jest praca Donalda O. Hebba (Hebb 1949), w której wykazano że połączenia neuronów są „wzmacniane” za każdym razem, gdy są używane. Idea ta jest podobna do sposobu uczenia człowieka - poprzez wielokrotne powtarzanie konkretnego zagadnienia jesteśmy w stanie zapamiętać więcej informacji na dłużej. Wraz z rozwojem zaawansowania komputerów w latach 50, w końcu było możliwe symulowanie sieci neuronowych. Pierwszy krok w tym celu wykonał Nathaniel Rochester z IBM. Niestety pierwsze podejście nie zakończyło się sukcesem. Z kolei w 1959 roku Bernard Widrow i Marcin Hoff z Uniwersytetu Stanford skutecznie rozwinęli modele nazwane „ADALINE” oraz „MADALINE” (Multiple Adaptive Linear Elements). ADALINE została stworzona, aby rozpoznawać wzory binarne w transmisji sygnału rozmowy telefonicznej, natomiast MADALINE służyła do eliminowania echa w rozmowach telefonicznych.

Powyższe modele nie powstałyby bez badań Franka Rosenblatta, który w 1957 roku jako pierwszy opisał perceptron. Perceptron zbudowany był z jednego lub kilku neuronów McCullocha-Pittsa. Algorytm ten wprowadził metodę uczenia nadzorowanego neuronów, można było stosować go do klasyfikacji zbiorów separowalnych liniowo. Później nastąpiła

pewna stagnacja w rozwoju sztucznych sieci neuronowych nazywana potocznie „*quiet years*” lub „*AI winter*”.

Sytuacji nie zmieniło nawet jedno z najważniejszych odkryć w historii metod uczenia sztucznych sieci neuronowych: zastosowanie algorytmu wstecznej propagacji błędów przez Paula Werbosa w 1974 roku. W swojej pracy doktorskiej na Uniwersytecie Harvarda jako pierwszy opisał proces wstecznej propagacji błędów w całej sieci. Metoda ta jest ciągle ulepszana i wykorzystywana do dziś. Punktem zwrotnym można uznać stworzenie metod uczenia, które mogą być stosowane do głębokich sieci neuronowych o wielu warstwach ukrytych. Cel ten został osiągnięty dopiero w 1989 roku, gdy naukowcy stworzyli niezbędne algorytmy, jednak czas uczenia sieci był zbyt długi, by metodę zastosować w praktyce. Sytuacja zmieniła się na początku XXI wieku, gdy określenie „*deep learning*” zyskało popularność po opublikowaniu artykułu przez Geoffreya Hinton i Rusłana Salakhutdinova (Hinton, Salakhutdinov 2006), który pokazał, że wielowarstwowe sieci neuronowe mogą być uczone warstwa za warstwą.

W kolejnych latach nastąpił znaczny wzrost rozwoju i zainteresowania, który trwa do dziś. Wiele ośrodków badawczych pracuje nad jej rozwojem, a w sieci dostępnych jest wiele informacji i gotowych przykładów zastosowania sztucznych sieci neuronowych.

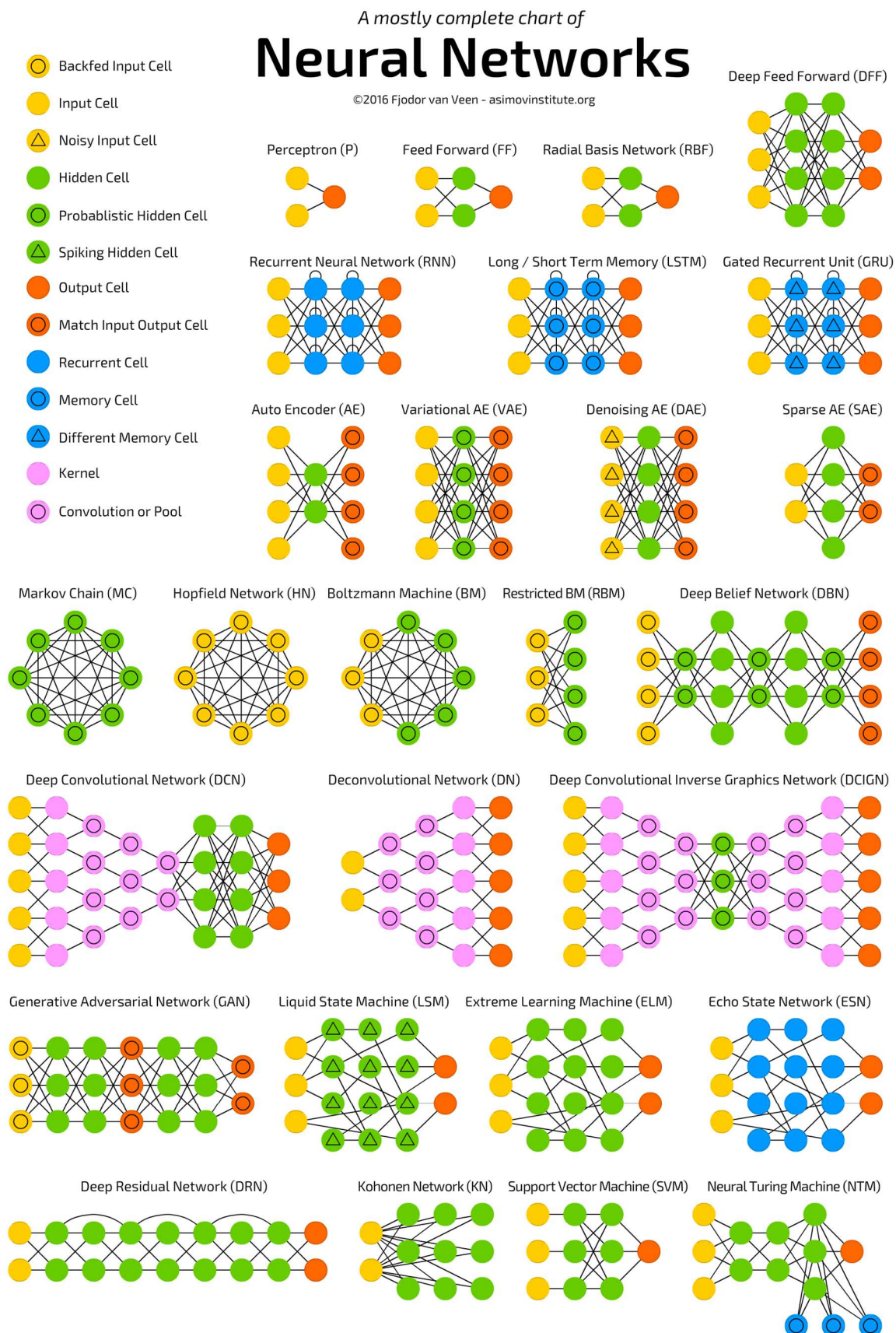
4.2. Budowa i działanie sztucznych sieci neuronowych

Jak już zostało wspomniane, sztuczna sieć neuronowa jest matematycznym modelem, który próbuje symulować strukturę i funkcjonalność mózgu. Dokładna zasada budowy i działania mózgu wciąż nie jest nam znana, jednak dzięki pracom nad próbami jego symulacji dziś możemy korzystać z sztucznych sieci neuronowych. Prace nad ich rozwojem doprowadziły do powstania różnych typów sieci neuronowych. Nie jest celem tej pracy omówienie ich wszystkich. Autor ograniczył się do opisanie tylko jednego typu, która została użyta w tej pracy. Pozostałe typy przedstawiono w formie graficznej. Na rysunku 4-1 przedstawiono graficznie za Fjodor van Veen, większość najczęściej używanych sieci neuronowych.

W pracy użyto jednej z najczęściej wykorzystywanych typów sieci: „*deep feed-forward neural network*”, czyli „głębokiej jednokierunkowej sieci neuronowej”. O sieci głębokiej mówimy wtedy, gdy sieć ta ma więcej niż jedną warstwę ukrytą. Natomiast jednokierunkowa oznacza, że sygnał w sieci przepływa wyłącznie w kierunku od węzłów wejściowych do wyjściowych. Innym typem mogłaby być sieć rekurencyjna, która charakteryzuje się sprzężeniem zwrotnym między wejściem a wyjściem.

Podstawową częścią składową, czy to mózgu czy sztucznej sieci jest neuron. W kolejnych podpunktach zostaną krótko opisane podstawowe elementy, które budują sztuczny neuron, zasady budowania z nich sieci oraz w jaki sposób przebiega proces uczenia.

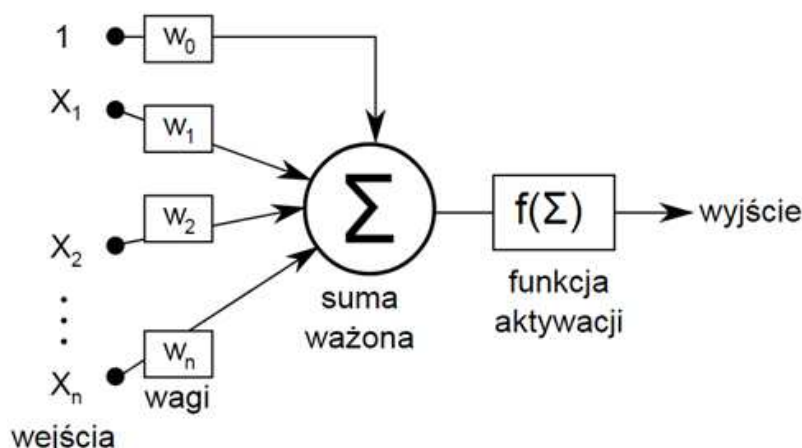
Tłumaczenie niektórych elementów do grafiki 4-1: Input Cell – węzeł wejściowy, Hidden Cell – węzeł warstwy ukrytej, Output Cell – węzeł warstwy wyjściowej, Recurrent Cell – węzeł rekurencyjny, Convolution – konwolucja, memory cell – węzeł pamięci.



Rys. 4-1. Sieci neuronowe i ich typy (<http://www.asimovinstitute.org>)

4.2.1. Budowa sztucznego neuronu

Model sztucznego neuronu praktycznie nie zmienił się od czasów jego powstania, czyli modelu neuronu McCullocha-Pittsa z roku 1943, zaprezentowanego na rysunku 4-2. W tym modelu wszystkie sygnały wejściowe x_n przemnażane są przez towarzyszące im wagi w_n będące liczbami rzeczywistymi, a następnie są sumowane. Następnie suma ta staje się argumentem funkcji aktywacji. Funkcje aktywacji omówione zostaną w kolejnych podpunktach. Funkcja aktywacji jest za razem ostatnią operacją w neuronie i stanowi jego wartość wyjściową.



Rys. 4-2. Model neuronu McCullocha-Pittsa (<http://pl.wikipedia.org>)

Oprócz podstawowych wartości wejściowych x_n , do sumatora doprowadzone jest dodatkowe wejście o stałej wartości 1 i wadze w_0 nazywane progiem lub wyrazem wolnym (ang. *bias*). Waga ta poddawana jest takiemu samemu procesowi uczenia, jak pozostałe wagi.

Cały algorytm można zaprezentować w postaci wzoru (4-1):

$$y = F\left(\sum_{i=1}^n w_i * x_i + w_0\right) \quad (4-1)$$

gdzie:

- x_i – wartości sygnału wejściowego „i”,
- w_i – wartości wagi połączenia „i”,
- w_0 – próg (ang. *bias*),
- $F()$ – funkcja aktywacji,
- y – odpowiedź neuronu (sygnał wyjściowy).

Można więc zauważyć, że pomijając funkcję aktywacji, operacje matematyczne, które wykonuje się w sztucznym neuronie, sprowadzają się do podstawowych operacji dodawania oraz mnożenia. Funkcja aktywacji i jej cel zostanie opisany w następnym podpunkcie.

4.2.2. Funkcja aktywacji

Największą niewiadomą we wzorze (4-1) jest funkcja aktywacji. Funkcja ta pełni bardzo ważną rolę w neuronie. To właśnie on definiuje właściwości sztucznego neuronu. Teoretycznie może być dowolną matematyczną funkcją, jednak w praktyce wykorzystuje się ich kilkanaście, w zależności od rodzaju i złożoności problemu, jaki rozwiązujemy. Można wyróżnić funkcje progowe, funkcje liniowe oraz funkcje nieliniowe (sigmoidalne). Najczęściej stosowane funkcje aktywacji, ich wykresy, równania i pochodne przedstawiono na rysunku 4-3.

Podstawowym celem funkcji aktywacji jest wprowadzenie nieliniowości do modelu. Gdyby nie wprowadzona nieliniowość kolejne warstwy sieci neuronowych nie wprowadziłyby żadnych korzyści. Dowolną liczbę warstw sieci neuronowej bez funkcji aktywacji można by zastąpić jedną warstwą, co kwestionowałoby zasadność używania więcej niż jednej warstwy. Kolejny powodem ich stosowania jest spłaszczenie wyniku działania neuronu do określonego przedziału (np. od zera do jedynki). Pozwala to zmniejszyć ryzyko niekontrolowanego wzrostu wartości wag, jak i przyspiesza proces uczenia.

Progowe funkcje aktywacji są funkcjami binarnymi, które mogą przyjąć tylko dwie wartości (np. zero i jeden). Oznacza to tyle, że jeżeli wartość wejściowa jest większa od wartości progowej to funkcja przyjmuje jedną wartość, a gdy jest mniejsza przyjmuje drugą wartość. Najczęściej stosowane są funkcje bipolarne i unipolarne.

Nazwa	Wykres	Równanie	Pochodna
Liniowa		$f(x) = x$	$f'(x) = 1$
Progowa unipolarna		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x \neq 0 \\ ? & \text{for } x = 0 \end{cases}$
Sigmoidalna unipolarna		$f(x) = \frac{1}{1 + e^{-x}}$	$f'(x) = f(x)(1 - f(x))$
Tangens hiperboliczny		$f(x) = \tanh(x) = \frac{2}{1 + e^{-2x}} - 1$	$f'(x) = 1 - f(x)^2$
Arcus tangens		$f(x) = \tan^{-1}(x)$	$f'(x) = \frac{1}{x^2 + 1}$
Obcięta funkcja liniowa		$f(x) = \begin{cases} 0 & \text{for } x < 0 \\ x & \text{for } x \geq 0 \end{cases}$	$f'(x) = \begin{cases} 0 & \text{for } x < 0 \\ 1 & \text{for } x \geq 0 \end{cases}$
Softmax		$f(\vec{x})_i = \frac{e^{x_i}}{\sum_{k=1}^K e^{x_k}} \quad \text{for } i = 1, \dots, K$	$\frac{\partial f(\vec{x})_i}{\partial x_j} = f(\vec{x})_i (\delta_{ij} - f(\vec{x})_j)$

Rys. 4-3. Wybrane funkcje aktywacji (<http://en.wikipedia.org>)

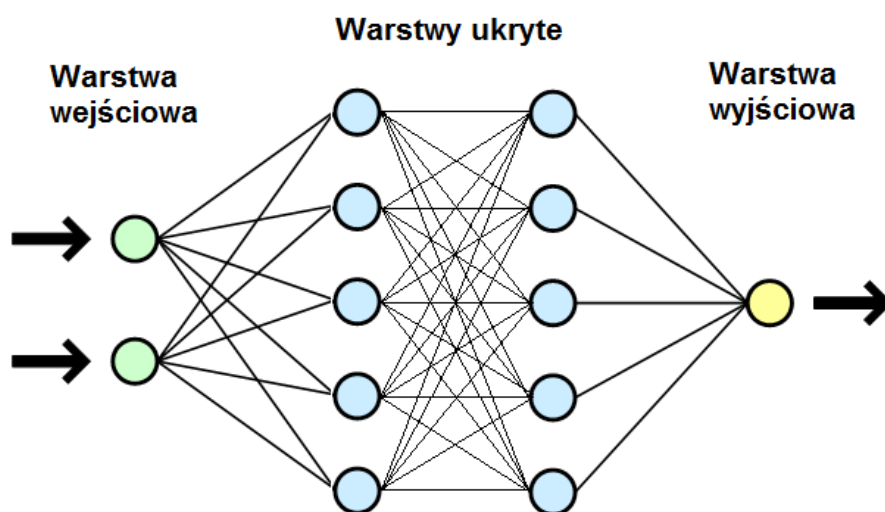
Funkcje nieliniowe (sigmoidalne) swoim kształtem przypominają literę s, ich wartości najczęściej ograniczone są do zakresu (0,1) lub (-1,1). Funkcje te doskonale sprawdzają się w wielowarstwowych sieciach, które rozwiązują zadania nieliniowe. Dzięki zastosowaniu funkcji nieliniowych już dwuwarstwowa sieć neuronowa jest uniwersalnym aproksymatorem funkcji (Cybenko, George 1989). W przeciwieństwie do funkcji progowych mają ciągłą pochodną, dzięki czemu można do nich stosować metody uczenia oparte na gradientach.

Obcięta funkcja progorowa jest funkcją powstałą z połączenia funkcji progowej i funkcji liniowej. Stała się bardzo popularna w ostatnich latach ze względu na znaczne przyspieszenie procesu uczenia w porównaniu do funkcji sigmoidalnych, jednocześnie mogącą modelować zadania nieliniowe.

Szczególnym typem funkcji nieliniowej jest funkcja softmax. Jest to funkcja wykładnicza, znormalizowana w ten sposób, aby suma aktywacji dla całej warstwy zawsze była równa 1. Dzięki temu wartości wyjściowe mogą być interpretowane jako pewnego rodzaju „prawdopodobieństwo” przynależności do danego sygnału wyjściowego. Praktycznie używana jest wyłącznie w warstwach wyjściowych, w sieciach, które rozwiązują problemy klasyfikacyjne.

4.2.3. Sieć neuronów

Gdy połączymy dwa neurony McCullocha-Pittsa ułożone warstwowo, uzyskamy najprostszy perceptron wielowarstwowy – czyli inaczej jedną z najprostszych sieci neuronowych. W każdej takiej warstwie neuronów może być od jednego neuronu w górę, ograniczeniem jest jedynie moc obliczeniowa komputerów.



Rys. 4-4. Struktura sieci neuronowej (za www.wikipedia.org)

Rozróżniamy trzy podstawowe typy warstw: warstwę wejściową, warstwy ukryte, warstwę wyjściową. Przykładową strukturę sieci neuronowej pokazano na rysunku 4-4.

Warstwa wejściowa składa się z węzłów, które nie są de facto neuronami. W tej warstwie nie są wykonywane żadne obliczenia, warstwa ta jedynie przekazuje dane wejściowe do sieci. Najczęściej każdy pojedynczy węzeł odpowiada jednej wprowadzanej do sieci wartości, co oznacza, że ich ilość w warstwie zależy od analizowanego problemu.

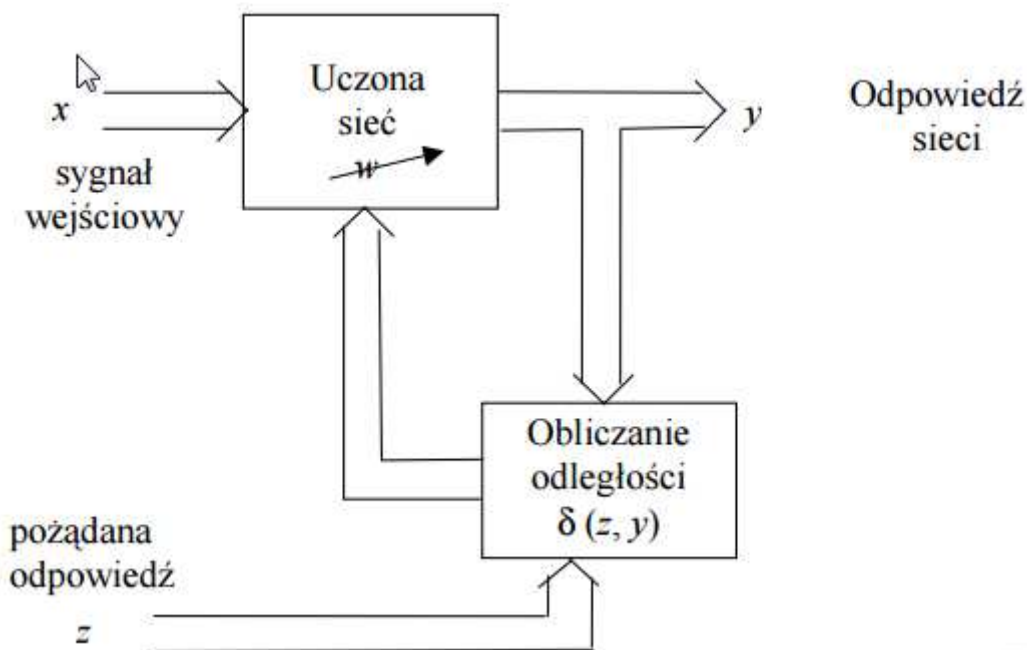
Najbardziej rozbudowaną częścią sieci są warstwy ukryte. Każdy pojedynczy węzeł jest pojedynczym neuronem, który połączony jest z każdym węzłem warstwy poprzedniej (warstwy wejściowej lub poprzedzającej warstwy ukrytej). W związku z czym wyjście z neuronów poprzednich warstw staje się wejściem każdego neuronu w warstwie kolejnej, i tak aż do warstwy wyjściowej. Podczas budowy sieci to właśnie ilość warstw, jak i ilość neuronów w każdej warstwie jest elementem, który ulega zmianie. Im bardziej skomplikowane zadanie jest modelowane tym więcej warstw i neuronów potrzeba.

Ostatnią warstwą jest warstwa wyjściowa. Działa ona jak warstwy ukryte, jednak wyjście neuronu nie jest wejściem dla kolejnej warstwy, jest odpowiedzią działania tej sieci w analizowanym problemie. Ich ilość, podobnie jak w warstwie wejściowej, bezpośrednio zależy od analizowanego problemu: każdemu neuronowi przyporządkowany jest jeden parametr.

4.2.4. Uczenie sieci

Zadaniem uczenia sieci jest takie dostrojenie wartości wag, aby odpowiedź sieci była jak najbliższa oczekiwanym. Odbywa się to w sposób iteracyjny na dwa możliwe sposoby: uczenie nadzorowane (uczenie z nauczycielem) oraz uczenie bez nadzoru (samouczenie). Do uczenia z nauczycielem wykorzystywane są przygotowane zestawy danych. Zestaw taki składa się z informacji o wartościach parametrów wejściowych oraz oczekiwanej odpowiedzi algorytmu na zadane dane wejściowe. Przeciwnieństwem tego sposobu jest samouczenie, gdzie nie ma informacji o oczekiwanej odpowiedzi sieci, jedynie informacje o danych wejściowych.

W tej pracy wykorzystano metodę uczenia nadzorowanego. Mając zestaw oczekiwanych odpowiedzi sieci oblicza się odległość (błąd) między nią a rzeczywistą odpowiedzią sieci. Na jego podstawie dokonuje się aktualizacji wag i ponownego obliczenia błędu (rysunek 4-5). Kolejne iteracje przeprowadzane są do momentu uzyskania satysfakcjonującej odległości (błędu).



Rys. 4-5. Schemat uczenia z nauczycielem (<http://www.neurosoft.edu.pl>)

Najprostszy algorytm, który można wykorzystać do uczenia jednowarstwowej sieci neuronowej, jest algorytmem wykorzystującym regułę delta. Dla sieci wielowarstwowych wykorzystywany jest algorytm wstecznej propagacji błędów (ang. *backpropagation*). To właśnie poznanie tej metody, dopiero w latach 80-tych, umożliwiło dalszy rozwój prac nad sieciami neuronowymi. Algorytm ten umożliwił uczenie sieci o więcej niż jednej warstwie, co znacznie zwiększyło ich możliwości. Dokładny algorytm uczenia został już bardzo dokładnie opisany w wielu pracach, między innymi w pierwszej pracy na ten temat: „*Learning representations by back-propagating errors*” (Rumelhart, Hinton, Williams 1986).

4.3. Modelowanie sztucznych sieci neuronowych

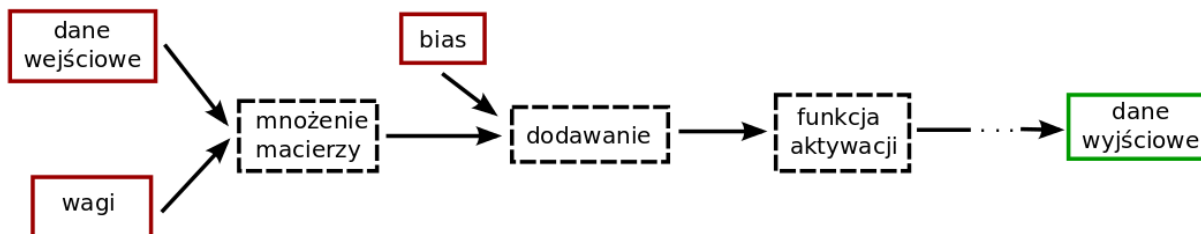
Rozwój sieci neuronowych nie byłby możliwy bez dostępu do oprogramowania wspomagającego ich tworzenie i badanie. Powstało wiele darmowych bibliotek i dodatków, na przykład Theano - rozwijane przez Uniwersytet w Montrealu, Caffé - stworzone przez Berkley Vision and Learning Center, czy bardzo popularny w ostatnim czasie TensorFlow - rozwijany przez Google Brain Team i udostępniony w listopadzie 2015 roku. Od tego momentu TensorFlow jest aktywnie rozwijany zarówno przez twórców jak i użytkowników, udostępniony na wiele systemów operacyjnych w tym system Windows.

Spośród wymienionych wyżej metod, w poniższej pracy wykorzystano bibliotekę TensorFlow. W tej bibliotece wszystkie obliczenia reprezentowane są jako grafy. Każdy węzeł w grafie obliczeniowym jest pojedynczą operacją, która pobiera dowolną, wskazaną ilość tensorów, wykonuje obliczenia i przekazuje ją dalej. W terminologii programu tensorem nazywamy wielowymiarową tablicę, będącą podstawowym nośnikiem danych. Biblioteka może służyć do wykonywania dowolnych obliczeń, nie tylko do modelowania sieci neuronowych, jednak to właśnie to one są jej głównym przeznaczeniem.

Programy pisane w TensorFlow składają się z dwóch zasadniczych faz: fazy budowy grafu oraz fazy realizacji grafu. Budowanie grafu odbywa się poprzez zdefiniowanie jego węzłów w określonej kolejności, to jest kolejności wykonywania operacji. Następnie graf jest uruchamiany w sesji, która zamienia go w operacje wykonywalne przez komputer (procesor CPU lub kartę graficzną GPU). Dopiero podczas tej fazy wprowadzane są dane do grafu, przez który „przepływają” (ang. *flow*) przez kolejne węzły, wyznaczając wynik operacji. Dzięki znajomości całego zadania (grafu) przed przystąpieniem do obliczeń, najważniejszą zaletą tego rozwiązania jest duża szybkość i efektywność wykonywanych działań, natomiast wadą jest konieczność uruchamiania całego grafu, nawet jeśli wykonuje się najprostsze obliczenia.

Sieci neuronowe w TensorFlow są przedstawione jako graf kolejnych operacji wykonywanych w sieci. Węzłami grafu mogą być na przykład mnożenie tensora wartości wejścia przez tensor wag, dodanie tensora biasu czy zastosowanie funkcji aktywacji. Na rysunku 4-6 przedstawiono przykładowy graf reprezentujący pojedynczy perceptron (lub całą ich warstwę). Budowę grafu kończy się definicją funkcji kosztu oraz wyboru metody uczenia spośród dostępnych w TensorFlow. Uczenie realizowane jest przez uruchomienie grafu w sesji, w której minimalizuje się funkcję kosztu używając zdefiniowanej metody uczenia. Różniczkowanie oraz propagacja błędów wykonywane są automatycznie dzięki znajomości

całego grafu, użytkownik dostarcza jedynie dane uczące. W programie można stworzyć od najprostszych jednowarstwowych sieci neuronowych po najbardziej zaawansowane sieci LSTM (*long-short term memory*).



Rys. 4-6. Przykładowy graf reprezentujący pojedynczy perceptron w TensorFlow

TensorFlow można wykorzystać w kilku językach programistycznych, między innymi GoLang, C czy C++. Jednak najczęściej używa się go w połączeniu z językiem Python.

4.4. Przykładowe zastosowania sztucznych sieci neuronowych w geotechnice

Sieci neuronowe znalazły już zastosowanie w szeroko pojętej geotechnice. Ich zastosowanie może być zalecane szczególnie w przypadkach, gdy modelowanie teoretyczne nie daje rezultatów zgodnych z badaniami doświadczeniami lub istnieją trudności w zastosowaniu innych metod statystycznych. Spośród wszystkich zastosowań, w literaturze (Sulewska 2011) podaje się ich skuteczne zastosowanie między innymi do niżej zaprezentowanych przykładów.

Pierwszym z nich jest przewidywanie parametru prekonsolidacji gruntu OCR. Na podstawie 195 wyników badań glin sondą PCPT zbudowano sieć neuronową o budowie 5-8-1 (liczba węzłów w warstwie wejściowej, ukrytej i wyjściowej). Wynikiem działania sieci była wartość współczynnika prekonsolidacji OCR. Wyniki te były później porównane do wyników otrzymanych za pomocą innych dostępnych metod (modeli empirycznych i teoretycznych) określania współczynnika OCR oraz wyników badań edometrycznych, na podstawie których określono rzeczywisty współczynnik OCR. Na tle innych metod sieć neuronowa najlepiej spisywała się do określania współczynnika OCR glin w stanie nienaruszonym. Z mniejszą dokładnością określała OCR glin spękanych, jednak jej dokładność była równa lub większa niż najlepsza dostępna metoda.

Kolejnym przykładem jest przewidywanie upłynnienia gruntu podczas trzęsień ziemi, głównie w luźnych i nawodnionych piaskach. Do uczenia wykorzystano 85 przypadków, spośród których w 42 doszło do upłynnienia. Na podstawie tych danych sieć była w stanie określić czy przy danych parametrach gruntu może dojść do upłynnienia czy nie. Błąd na podstawie danych testowych wyniósł 7,7%.

Sztuczne sieci neuronowe posłużyły również do przewidywania parametrów zagęszczalności gruntów niespoistych. Sieć służyła do przewidywania wilgotności optymalnej w_{opt} oraz gęstości objętościowej szkieletu gruntowego przy najściślejszym ułożeniu ziaren ρ_{dmax} na podstawie parametrów krzywej przesiewu gruntu. Błąd sieci neuronowej w stosunku

do wyniku badania aparatem Proctora wynosił odpowiednio 30% w stosunku do wilgotności optymalnej i 8% w stosunku do maksymalnej gęstości objętościowej gruntu.

Wszystkich zastosowań są dziesiątki, z których większość można znaleźć w literaturze zarówno angielskiej i polskiej. Spośród wszystkich zastosowań w geotechnice największym problemem z jakim często można się spotkać była mała ilość danych uczących. Jednak mimo to ilość zastosowań w geotechnice rośnie i ze względu na ciągły rozwój zarówno geotechniki, jak i sieci neuronowych ilość ich zastosowań będzie coraz większa.

5. DANE

Na potrzeby tej pracy niezbędne było stworzenie bazy danych gruntów, które posłużą do uczenia sztucznej sieci neuronowej. Ze względu na charakter działania tych sieci, niezbędna jest duża liczba rekordów danych, przy czym im większa, tym większą rzeczywistą dokładność można osiągnąć. Badania CPTu dostarczają ich wystarczającą ilość: w każdym badaniu są ich po 100 na każdy metr (pomiar odbywa się co każdy centymetr), w pojedynczym sondowaniu, które ma średnio powyżej 10 m, może ich być ponad tysiąc.

Dzięki uprzejmości Jędrzeja Wierzbickiego z Uniwersytetu Adama Mickiewicza autor pracy dysponował 34 sondowaniami CPTu wraz ze skorelowanymi profilami podłoża gruntowego, na którym zaznaczone były informacje o rodzaju gruntu (Wierzbicki J. Materiały własne).

Dane z każdego sondowania statycznego zapisane były w plikach tekstowych, umożliwiających odczytanie ich zarówno za pomocą podstawowych programów komputerowych takich jak Notatnik czy Excel, ale również za pomocą języka skryptowego Python. Do tego celu stworzona została osobna aplikacja, która wczytuje, filtruje a następnie normalizuje dostępne dane przed przystąpieniem do uczenia SSN. Wykorzystuje ona dostępną na licencji wolnego oprogramowania bibliotekę Pandas, która umożliwia wczytywanie, budowę i modyfikację dużych baz danych.

5.1. *Struktura i filtracja danych*

Każdy pojedynczy plik z sondowania CPTu składał się z dwóch części. W pierwszej z nich zawarte były podstawowe informacje o miejscu i dacie przeprowadzonych badań, współczynniki powierzchni stożka czy numerze i głębokości badań. W drugiej części zawarte były wyniki sondowań. Przykładowy wydruk części informacji z pojedynczego pliku pokazano na rysunku 5-1. W formie tabelarycznej zawarte były zmieniające się wraz z głębokością podstawowe informacje mierzone w badaniu, między innymi:

- aktualna głębokość w cm,
- opór stożka Q_c w MPa,
- opór pobocznicy F_s w kPa,
- ciśnienie wody w porach U_2 w kPa,
- odchylenie wciskanej sondy od pionu w stopniach,
- szybkość wciskania sondy w cm/s,
- ciśnienie hydrostatyczne U_0 w kPa.

	Qc	Fs	Speed	U2	Tilt	a Factor						
Factors:	223780	23516		10	24054	141139	0.58					
Zeros:	2339	4869		0	9300	3004						

Depth [cm]	Qc [MPa]	Fs [KPa]	U2 [KPa]	Rf [%]	U2/Qc [%]	Qc-U2 [MPa]	Tilt [°]	Dist [cm]	Speed [cm/sec]	Qt [MPa]	U2-U0 [KPa]
1	1.0144	4.5076	11.848341	0.444363	1.168027	1.002541	0.6061	0.0106	0.8	1.019365	11.848341
2	2.2165	4.6777	6.11125	0.211042	0.275721	2.210351	0.5165	0.0196	2	2.219029	6.11125
3	1.8322	2.5515	-2.702253	0.13926	-0.14749	1.834859	0.9542	0.0362	0.4	1.831022	-2.702253
4	2.3014	3.4445	-1.912364	0.14967	-0.0831	2.30328	0.9542	0.0529	1.9	2.300564	-1.912364
5	3.8073	5.188	-2.286522	0.136263	-0.06006	3.809597	1.2723	0.0751	1.9	3.80635	-2.286522
6	5.5635	6.6763	-1.371913	0.120002	-0.02466	5.564872	1.718	0.1051	1.8	5.562924	-1.371913
7	6.4081	6.166	-0.623597	0.096223	-0.00973	6.408703	1.718	0.1351	2.1	6.407817	-0.623597
8	6.9935	8.7175	0.748316	0.124651	0.0107	6.992727	1.783	0.1662	1.9	6.99379	0.748316
9	7.7934	12.29	4.65619	0.157692	0.059746	7.788712	1.783	0.1973	2	7.795324	4.65619
10	8.0928	15.266	4.489898	0.18864	0.05548	8.08828	1.783	0.2284	2	8.094655	4.489898
11	8.3475	16.329	-2.951692	0.19562	-0.03536	8.350436	1.783	0.2595	2	8.346244	-2.951692
12	8.1866	23.176	-9.894404	0.283093	-0.12086	8.196506	1.9083	0.2928	2	8.182456	-9.894404
13	8.1732	30.83	-3.824728	0.377209	-0.0468	8.177031	1.783	0.3239	2	8.171599	-3.824728
14	8.066	30.958	-8.023614	0.383806	-0.09948	8.073981	1.9083	0.3572	2	8.062588	-8.023614
15	7.9185	29.809	-3.949447	0.376454	-0.04988	7.922441	1.783	0.3884	2	7.916833	-3.949447
16	7.9855	31.298	-0.997755	0.391932	-0.0125	7.986519	1.783	0.4195	2	7.985102	-0.997755
17	8.0749	36.996	8.855076	0.458162	0.109662	8.06604	1.783	0.4506	2	8.078614	8.855076

Rys. 5-1. Wydruk przykładowych danych z badania CPTu (Wierzbicki J. Materiały własne)

Rodzaj świda		Średnica rur i głębokość zarurowania [m]		Głębokość zwierciadła wody gruntowej [m p.p.t.]		Głębokość poboru próby gruntu [m p.p.t.]		Skala pionowa		Profil litologiczny		Przeloty warstw [m]		Opis makroskopowy					Stratygrafia									
														Rodzaj gruntu					Wilgotność		Stan gruntu		Ilość walczków		Zawartość CaCO3		Numer warstwy geotechnicznej	
1	2	3	4	5	6	7	8	9	10	11	12	13	14															
					Gb		gleba; ciemno-szara	w	-	-	-	Gb																
					Pd	0,3	piasek drobny; szary	w	-	-	-	-																
			1,5	1,0	Gp//Pd	0,6	głina piaszczysta przew. piaskiem drobnym; brązowa	w	tpl	1/1	I	IVc																
			3,0	2,0	Pg//Pd	1,8	piasek gliniasty przew. piaskiem drobnym; jasno-brązowy	w	pl	1/2nw	I	IVb																
			4,5	3,0	Gp//Pd	3,6	głina piaszczysta przew. piaskiem drobnym; brązowo-szara	w	tpl	1/2	IV	IVc																
			6,0	4,0	Gp+Ż	4,7	głina piaszczysta ze żwirem; szaro-brązowa	w	tpl	1/2	IV	IVc																
			7,5	5,0		6,8																						
			10,5	6,0	Gp		głina piaszczysta; szara	w	tpl	1/2	IV	VIb																
				7,0																								
				8,0																								
				9,0																								
				10,0																								
				11,0																								
				12,0																								
				13,0																								
				14,0																								

Rys. 5-2. Wydruk przykładowego profilu podłoża gruntowego (Wierzbicki J. Materiały własne)

Z każdym pojedynczym sondowaniem skorelowane są profile gruntowe (rysunek 5-2) uzyskane dzięki wierceniom. Na podstawie tych profili uzyskano informacje o rodzaju gruntu jaki jest przypisany do każdego pojedynczego rekordu z badania CPTu. Informacje te zostały zamienione na postać możliwą do odczytania przez komputer (pokazany na rysunku 5-3) i wczytane do bazy danych gruntów.

Ostatecznie uzyskano bazę kilkunastu tysięcy rekordów danych gruntów, w których zawarte są zarówno informacje o pomiarach z badania CPTu, jak i informacje o gruncie. Przed przystąpieniem do dalszych prac, dane zostały przefiltrowane. Usunięto wszystkie grunty, które zawierały jakiegokolwiek domieszki innych gruntów czy kamienie. Nie analizowano również gruntów leżących blisko powierzchni terenu (mniej niż 0,5 m poniżej poziomu terenu) oraz warstwy o grubości mniejszej niż 1 m w przypadku gruntów niespoistych i 1,5 m w przypadku gruntów spoistych. Z każdej pozostałej warstwy wyeliminowano grunty do głębokości 0,5 m poniżej stropu i powyżej spągu każdej warstwy niespoistej i 0,75 m dla gruntów spoistych. Z bazy usunięto również wszystkie rekordy które uzyskano w momencie w którym sonda podczas wciskania odchyliła się od pionu o więcej niż 4 stopnie.

Gleb.	Grunt1	Grunt2
0,3	Gb	0
2,8	Gp	Pd
3,2	Gp	Pg
4,5	Gp	0
5,3	Pd	G
5,6	Pd	0
13,4	Gp	0
14,0	Gp	0
25,0	Gp	0

Rys. 5-3. Informacje o profilu gruntowym w formie wczytywanym do programu.

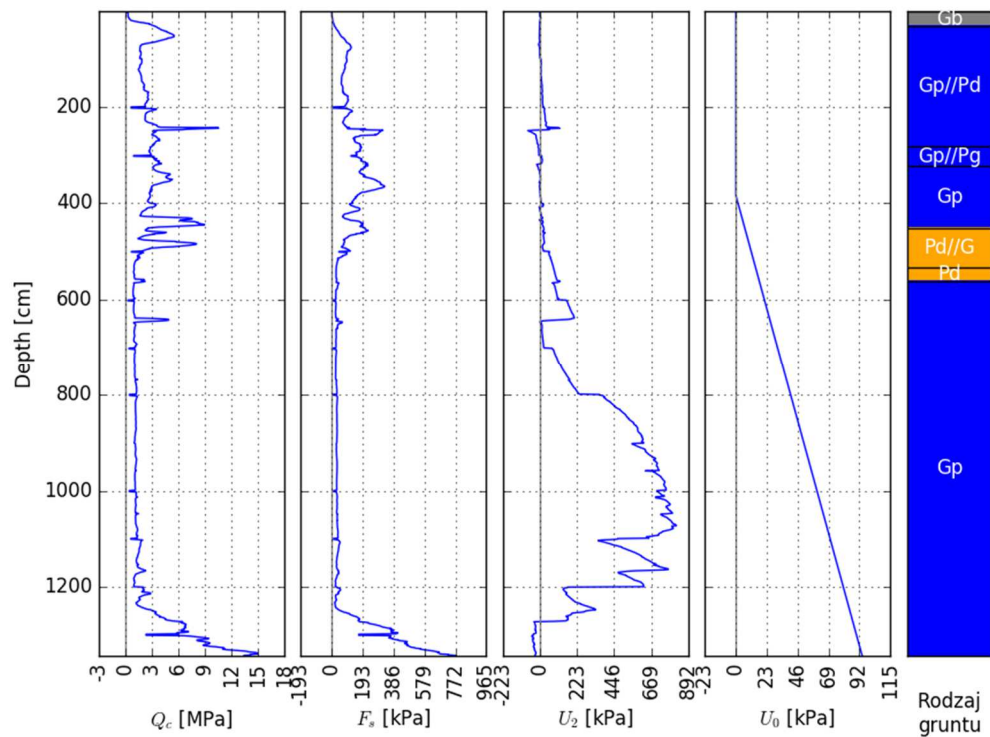
Dzięki takiemu procesowi filtracji danych uniknięto wpływu małej głębokości badania, niejednorodności gruntu, zbyt małej grubości warstw oraz wpływu warstw poprzednich na pomierzone wartości w aktualnej warstwie. Po filtracji, z kilkunastu tysięcy rekordów, pozostało prawie dziesięć tysięcy różnorodnych rekordów danych, które zostały poddane dalszej obróbce.

5.2. Wizualizacja danych

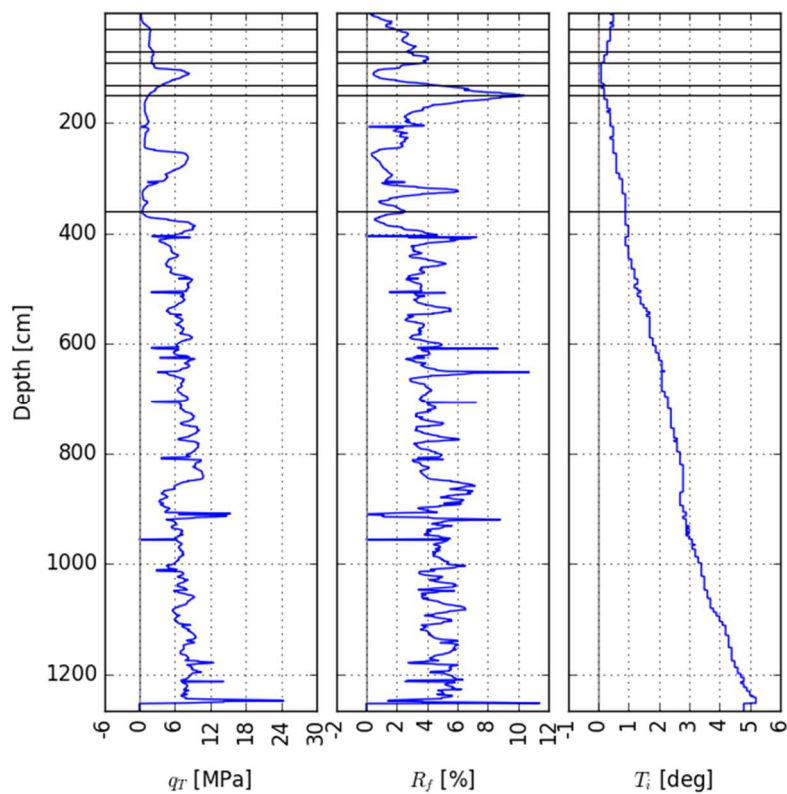
Po wczytaniu każdego profilu oraz na każdym etapie obróbki danych, przechowywanie całej bazy danych w formie „dataframe-u” Pandas w prosty sposób umożliwia ich wizualizację. Do tego celu wykorzystano kolejną bibliotekę dostępną w języku skrytowym Python na zasadach wolnego oprogramowania: „Matplotlib”. Stworzony na jej podstawie skrypt umożliwia zbudowanie wykresów z każdego sondowania w dowolnej, zadanej przez użytkownika, konfiguracji. Po wpisaniu wymaganych wartości powstają interpretacje graficzne takie jak na rysunkach 5-4 i 5-5, gdzie:

- Q_c – opór wciskania stożka,
- F_s – opór tarcia pobocznicy,
- U_2 – ciśnienie wody w porach gruntu,

- U_0 – ciśnienie hydrostatyczne wody w gruncie,
- q_T – znormalizowany opór na stożku,
- R_f – współczynnik tarcia,
- T_i – odchylenie wciskania stożka od pionu,

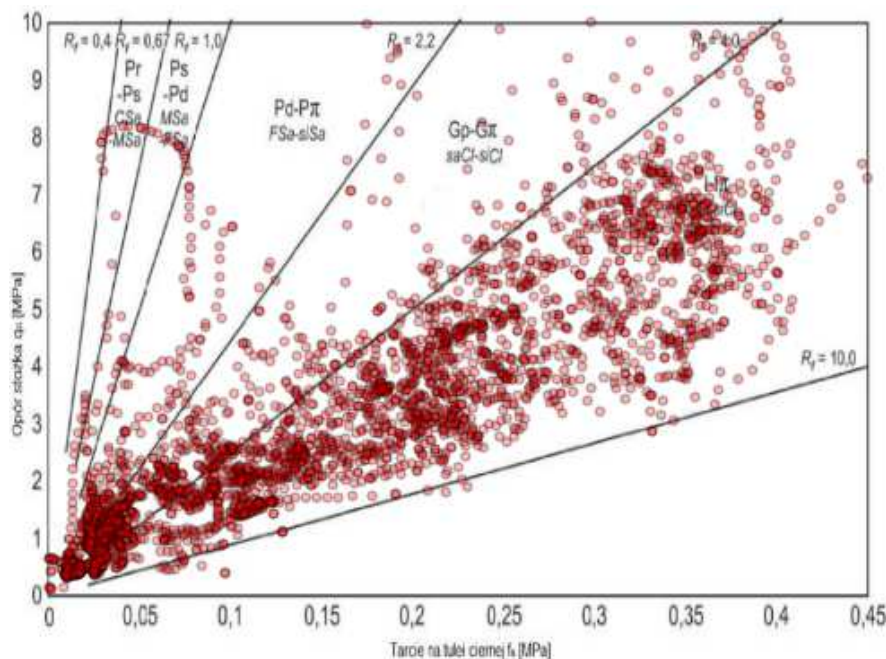


Rys. 5-4. Przykładowy wykres wygenerowany z programu,

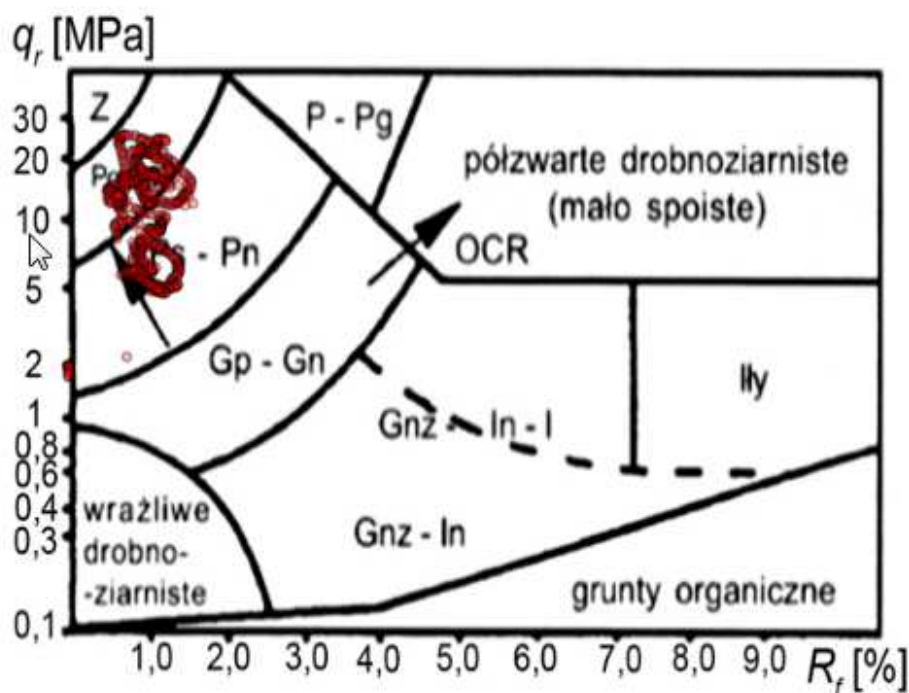


Rys. 5-5. Przykładowy wykres wygenerowany z programu

Jednym z celów pracy jest porównanie możliwości SSN z istniejącymi metodami klasyfikacji. Aby umożliwić te porównanie niezbędne było stworzenie programu nanoszącego każdy rekord gruntu na wykresy omawiane w punkcie 3.5. W tym celu, również z użyciem biblioteki „Matplotlib”, stworzono skrypt, który to umożliwia. Po wprowadzeniu informacji o gruntach i rodzaju wykresu (do wyboru są wykresy Marre’a z PN-B, Robertsona z PN-B, Robertsona z 1990r, Schmertmanna oraz Dągłasa & Olufsen) na jaki należy nanieść informację użytkownik otrzymuje wydruk z wykresem i naniesionymi na nim gruntami. Przykłady takich wydruków pokazano na rysunkach 5-6 oraz 5-7.



Rys. 5-6. Nomogram Marre'a z PN-B z naniesionymi danymi z sondowania (PN-B-04452:2002)



Rys. 5-7. Nomogram Robertsona z PN-B z naniesionymi danymi z sondowania (PN-B-04452:2002)

5.3. Podział i normalizacja danych

Dane wyfiltrowane na podstawie punktu 5.1 (przykładowe 20 rekordów danych przedstawiono w tabeli 5-1) zostały przeskalowane lub znormalizowane. Proces ten jest wykonywany w celu wyrównania wartości w warstwie wejściowej w sieci. Wyrównuje ona „ważność” każdej wartości wejściowej i jej wpływ na ostateczną odpowiedź sieci. To sama sieć na podstawie procesu uczenia, zmieniając wagi połączeń, decyduje, która wartość wejściowa jest ważna w danym przypadku. Dzięki normalizacji znacznie szybciej postępuje proces uczenia, a również możliwe jest osiągnięcie większej dokładności ostatecznej.

Tabela 5-1. Przykładowe dane przed normalizacją

No.	Głębokość [cm]	Qc [MPa]	Fs [kPa]	U0 [kPa]	U2 [kPa]	SBT [-]
0	326	0.457	13.72	0.98	22.27	2
1	1594	3.786	28.6	149.5	94.37	5
2	326	6.786	88.59	23.35	12.96	5
3	705	2.089	77.56	54.25	-40.51	4
4	906	1.13	28.84	51.21	582.25	2
5	980	1.105	30.39	58.47	750.79	2
6	421	2.37	163.66	20.99	174.85	2
7	562	4.914	45.6	44.83	58.18	5
8	329	3.198	222.24	32.27	345.14	2
9	362	5.756	48.51	27.86	32.72	5
10	753	9.765	322.29	58.96	-33.67	4
11	517	0.879	30.58	50.72	15.78	2
12	275	6.14	52.52	0	12.1	5
13	275	7.732	74.36	12.07	19.12	2
14	716	0.251	5.75	63.37	142.24	1
15	222	2.249	140.64	0	117.85	2
16	903	4.624	185.74	73.67	-33.5	4
17	452	2.4	123.59	0.98	41.39	2
18	266	5.408	57.69	15.7	27.47	5
19	648	3.394	296.32	49.74	40.86	2
20	349	2.938	47.82	25.6	11.88	5

Tabela 5-2. Parametry danych przed normalizacją

	Głębokość [cm]	Qc [MPa]	Fs [kPa]	U0 [kPa]	U2 [kPa]	SBT [-]
Odchylenie standardowe	328.3	5.2	102.7	32.8	181.8	-
Wartość średnia	571.5	6.2	126.3	43.2	96.0	-
Wartość minimalna	41	0	0	0	-100	2
Wartość maksymalna	1916	28.6	488.5	181.1	1353.9	6

Jak widać na tabeli 5-2, przed normalizacją wartości danych różniły się między sobą dość znacząco, szczególnie opór pod stożkiem Q_c ma znacznie mniejsze wartości średnie od pozostałych wartości.

Normalizację głębokości przeprowadzono za pomocą metody Min-Max (wzór 5-1) do wartości z zakresu od 0 do 1, przy czym 0 to głębokość 0, a 1 to maksymalna głębokość równa 1916 cm. Normalizację pozostałych parametrów przeprowadzono za pomocą metody Z-Score (wzór 5-2). Stosowana jest ona w przypadku danych, których rozkład przypomina rozkład naturalny, tak jak ma to miejsce w tym przypadku. Dzięki tej metodzie dane po normalizacji mają średnią równą zero i odchylenie standardowe równe jeden. SBT (*soil behaviour type*) jako jedyne nie była normalizowana.

Po procesie normalizacji, dane (tab. 5-3, 5-4) mają postać znacznie mniej zrozumiałą dla człowieka, jednak bardzo dobrze zrozumiałą dla komputerów.

$$x' = \frac{x}{max} \quad (5-1)$$

$$x' = \frac{x - \mu}{\sigma} \quad (5-2)$$

gdzie:

- x – wartość zmiennej przed normalizacją,
- x' – wartość zmiennej po normalizacji,
- max – wartość maksymalna zestawu danych,
- μ – wartość oczekiwana,
- σ – odchylenie standardowe.

Tabela 5-3. Parametry danych po normalizacji

	Głębokość [-]	Q_c [-]	F_s [-]	U_0 [-]	U_2 [-]	SBT [-]
Metoda normalizacji	Min-Max	Z-Score	Z-Score	Z-Score	Z-Score	-
Odchylenie standardowe	0.2	1.0	1.0	1.0	1.0	-
Wartość średnia	0.298	0	0	0	0	-
Wartość minimalna	0.0214	-1.2102	-1.2298	-1.3183	-1.0784	1
Wartość maksymalna	1	4.3463	3.5258	4.2015	6.9214	5

Tabela 5-4. Przykładowe dane po normalizacji

No.	Głębokość [-]	Qc [-]	Fs [-]	U0 [-]	U2 [-]	SBT [-]
0	0.170146138	-1.121542	-1.03518724	-1.288381	-0.40565	2
1	0.831941545	-0.475878	-0.89881027	3.2385958	-0.00893	5
2	0.170146138	0.105976	-0.34899477	-0.60653	-0.45687	5
3	0.367954071	-0.805014	-0.45008603	0.3353198	-0.75108	4
4	0.472860125	-0.991013	-0.89661064	0.2426588	2.675508	2
5	0.511482255	-0.995862	-0.8824047	0.4639479	3.602858	2
6	0.219728601	-0.750513	0.33903073	-0.678465	0.433888	2
7	0.293319415	-0.257101	-0.74300324	0.0481926	-0.20806	5
8	0.1717119	-0.589921	0.87592341	-0.334644	1.370867	2
9	0.188935282	-0.093794	-0.71633274	-0.469063	-0.34815	5
10	0.393006263	0.6837571	1.79289358	0.4788834	-0.71344	4
11	0.269832985	-1.039695	-0.88066333	0.2277233	-0.44136	2
12	0.143528184	-0.019317	-0.67958061	-1.318252	-0.4616	5
13	0.143528184	0.289454	-0.47941441	-0.950351	-0.42298	2
14	0.373695198	-1.161496	-1.10823324	0.6133027	0.254459	1
15	0.115866388	-0.773981	0.12804969	-1.318252	0.120259	2
16	0.471294363	-0.313347	0.54139656	0.9272528	-0.71251	4
17	0.235908142	-0.744695	-0.0282156	-1.288381	-0.30044	2
18	0.138830898	-0.161289	-0.63219695	-0.839707	-0.37704	5
19	0.338204593	-0.551907	1.55487544	0.1978523	-0.30336	2
20	0.182150313	-0.640349	-0.72265667	-0.537949	-0.46282	5

W następnym kroku dane zostały losowo potasowane, aby wyeliminować wpływ kolejności wprowadzania danych na proces uczenia oraz zostały podzielone na dwa sposoby. Po pierwsze każdemu rodzajowi gruntu została przyporządkowana SBT (*Soil behaviour type*) zgodnie z tabelą 5-5. Podziału tego dokonano na podstawie podziału zaproponowanego przez Robertsona w 1990 roku.

Niestety ze względu na ograniczoną ilość dostępnych danych z badań CPTu, nie było możliwe stworzenie pełnej, obejmującej wszystkie SBT, bazy danych. Niektóre typy SBT, pomimo że występują, stanowią mały procent całości (na przykład SBT typ 2 – grunty organiczne stanowią jedynie 2% całości), co będzie wpływało w znaczny sposób na realną dokładność w tym obszarze. Należy pamiętać, że sieć jest dokładna jedynie w zakresie, który jest objęty przez dane uczące. Gdy te dane są niepełne lub wybrakowane, tak jak mamy do czynienia w tym wypadku, sztuczna sieć neuronowa w tym zakresie będzie niedokładna, mimo że za każdym razem da jakiś wynik swojego działania.

Tabela 5-5. Podział danych ze względu na SBT wg Robertsona

Obszar SBT	Opis (za klasyfikacją Robertsona 1990)	Przydzielony rodzaj gruntu
1	<i>Wrażliwe drobnoziarniste</i>	-
2	<i>Organiczne</i>	Nm, T
3	<i>Iły do iłów pylastych</i>	I, Ip, G, Gz, Gpz, Gp
4	<i>Iły pylaste do pyłów ilastych</i>	Iπ, π, Gπz, Gπ
5	<i>Pyły ilaste do piasków pylastych</i>	πp, Pg
6	<i>Piaski pylaste do piasków</i>	Pπ, Pd, Ps
7	<i>Piaski do piasków zwirowych</i>	Pr, Pog, Po, Ż, Żg
8	<i>Bardzo zagęszczone piaski do piasków ilastych*</i>	-
9	<i>Bardzo zwięzłe drobnoziarniste*</i>	-

* - prekonsolidowane lub scementowane

Ostatecznie wszystkie dane zostały podzielone trzy części: dane uczące, dane walidacyjne oraz dane testowe zgodnie z tabelą 5-6. Dane uczące są największą grupą składającą się z prawie siedmiu tysięcy rekordów i są podstawą do uczenia sieci neuronowej. Dane walidacyjne są mniejszą, złożoną z prawie dwóch tysięcy rekordów danych grupą, które służą do określania dokładności sieci na każdym etapie jej uczenia czy określania rozmiarów sieci. Dane walidacyjne nigdy nie służą uczeniu sieci, jedynie określają jej chwilową dokładność. Ostatnią grupą są dane testowe, o rozmiarze około jednego tysiąca rekordów danych. Dane te wykorzystane są tylko raz: na samym końcu pracy do określania ostatecznej dokładności najlepszej sieci o optymalnej budowie. Podobnie jak w przypadku danych walidacyjnych, dane te nie służą do uczenia, co oznacza że sieć nigdy wcześniej ich nie „widziała”. Dzięki temu upewniamy się, że dana sieć została wybrana prawidłowo oraz działa poprawnie, jak i mamy pewność że posiada zdolności do uogólniania.

Tabela 5-6. Podział danych na uczące, walidujące i testowe

Obszar SBT	Liczba rekordów			
	Dane łącznie	Dane uczące	Dane walidujące	Dane testowe
1	0	0	0	0
2	173	117	37	19
3	3991	2786	820	385
4	412	271	88	53
5	983	688	198	97
6	4234	2989	818	427
7	0	0	0	0
8	0	0	0	0
9	0	0	0	0
Suma	9793	6851	1961	981
Procent całości	100.0%	70.0%	20.0%	10.0%

6. BUDOWA SZTUCZNEJ SIECI NEURONOWEJ

Sztuczne sieci neuronowe można modelować na wiele sposobów. Najczęściej wykorzystuje się w tym celu języki programistyczne lub skryptowe, rzadziej gotowe programy stworzone przez ich producentów. W tej pracy do zamodelowania sieci neuronowej wykorzystany został język skryptowy Python w połączeniu z biblioteką „Tensorflow” dostępną na zasadzie licencji wolnego oprogramowania Apache 2.0.

W omawianym przypadku zamodelowano jednokierunkową sztuczną sieć neuronową o kilku warstwach ukrytych. Do uczenia sieci wykorzystano algorytm o nazwie „Adam” (Kingma et. al., 2014), która jest rozwinięciem metody stochastycznego algorytmu spadku gradientu (ang. *stochastic gradient descent*). Aby niedopuścić do efektu przeuczenia (ang. *overfitting*), zastosowano technikę nazywaną „dropout”. Wyłącza ona losowe neurony w danej warstwie podczas procesu uczenia, co zwiększa czas uczenia, jednak zapobiega efektowi przeuczenia.

Aby uzyskać jak najlepsze rezultaty zamodelowano sieci neuronowe o różnej ilości warstw ukrytych i różnej ilości neuronów w tych warstwach. W tabeli 6-1 przedstawiono analizowane sieci i ich budowa (budowę sieci przedstawiono w postaci X-Y-Y-Z, gdzie X – liczba neuronów w warstwie wejściowej, Y – liczba neuronów w warstwach ukrytych, Z – liczba neuronów w warstwie wyjściowej) oraz liczbę epok uczących każdej sieci. W kilku przypadkach zmieniono również funkcję aktywacji z ReLU na Sigmoidalną, aby sprawdzić zachowanie sieci przy innych funkcjach aktywacji. Ilość epok każdej sieci dobierano na podstawie obserwacji procesu uczenia, jednak ta ilość była ograniczana, tak aby nie doprowadzić do przeuczenia sieci.

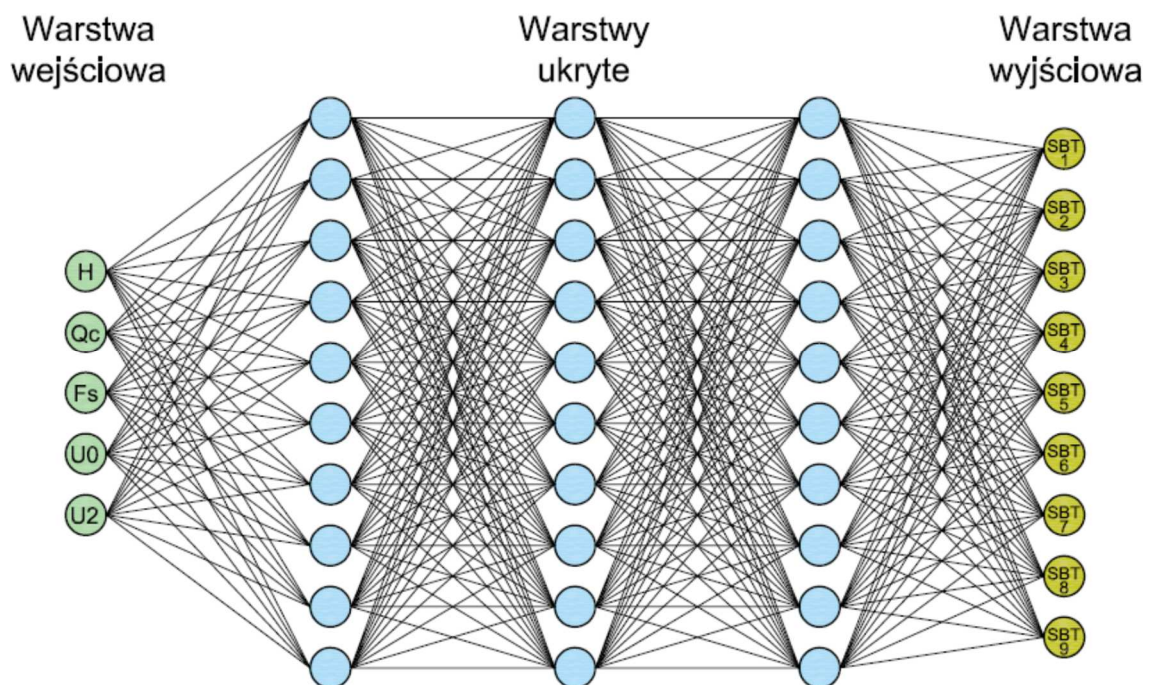
Wyniki działania tych sieci zaprezentowano w pkt 7, natomiast w kolejnych podpunktach opisano sposób w jaki każda z tych sieci była modelowana.

6.1. Budowa sieci

Pierwszym etapem budowy sieci neuronowej używając „Tensorflow” jest zdefiniowanie jej struktury. Na strukturę sieci neuronowej składają się połączone ze sobą trzy elementy: warstwa wejściowa, połączone między sobą warstwy ukryte oraz warstwa wyjściowa. Przykładowa struktura sieci, która została użyta w pracy, przedstawiona jest na rysunku 6-1.

Tabela 6-1. Badane sieci neuronowe

Sieć		Liczba epok uczących
Budowa	Uwagi	
5-15-15-9		500
5-20-20-9		500
5-25-25-9		500
5-30-30-9		500
5-40-40-9		500
5-10-10-10-9		500
5-15-15-15-9		500
5-15-20-15-9		500
5-20-20-20-9		500
5-20-30-20-9		500
5-25-25-25-9	F. Akt.: Sigmoidalna	500
5-25-40-25-9		500
5-25-40-25-9		500
5-30-30-30-9		500
5-15-20-20-15-9		1000
5-20-30-30-20-9		1000
5-25-40-40-25-9		1000
5-25-40-40-25-9	F. Akt.: Sigmoidalna	1000
5-25-40-40-25-9		3000



Rys. 6-1. Przykładowa struktura sieci neuronowej użytej w tej pracy o budowie 5-10-10-10-9

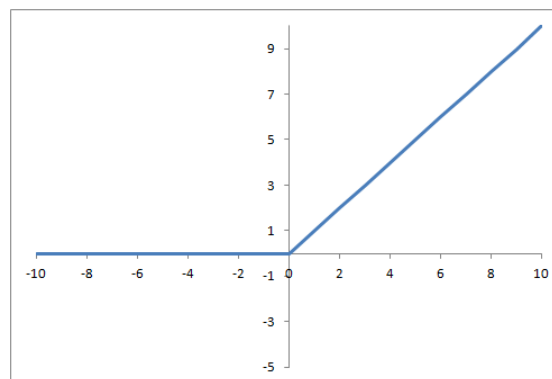
Warstwa wejściowa w omawianej sieci składała się z pięciu węzłów. Każdy z węzłów odpowiada jednemu z analizowanych parametrów z sondowania statycznego CPTu. Wartości tych parametrów zostały znormalizowane zgodnie z pkt. 5.4 tej pracy. Węzły warstwy wejściowej, w kolejności wystąpienia w tej warstwie, to:

1. głębokość pomiaru -H
2. opór wciskania stożka - Q_c
3. tarcie wciskania tuleji ciernej - F_s
4. ciśnienie hydrostatyczne – U_0
5. ciśnienie wody w porach gruntu – U_2

Każdy z węzłów z warstwy wejściowej został połączony z każdym z węzłów (neuronów) z pierwszej warstwy ukrytej. Każdy neuron warstwy ukrytej posiada dodatkowe wejście nazywane biasem. Wartość biasu zawsze wynosi 1, wagi jego połączenia podlegają normalnemu procesowi uczenia. W warstwach ukrytych funkcją aktywacji każdego neuronu jest obciążona funkcja liniowa (ang. *Rectified Linear Unit* - ReLU), przedstawiona wzorem 6-1, wykres funkcji pokazano na rysunku 6-2.

$$y(x) = \max(0, x) \quad (6-1)$$

W kilku przypadkach zastosowano funkcję aktywacji Sigmoidalną, aby sprawdzić jak inne funkcje aktywacji sprawdzają się w tym zadaniu. Po każdej warstwie ukrytej zastosowano technikę „Dropout”, przy prawdopodobieństwie zachowania neuronu ustawioną na 0,9 (10% neuronów w warstwie zostanie losowo wyłączonych podczas każdej iteracji procesu uczenia).



Rys. 6-2. Wykres funkcji aktywacji ReLU

Warstwa wyjściowa składała się z dziewięciu neuronów połączonych z każdym neuronem ostatniej warstwy ukrytej. Każdy węzeł warstwy wyjściowej odpowiada jednemu z typów SBT (*soil behaviour type*). Podobnie jak warstwy ukryte, neurony warstw wyjściowych posiadają dodatkowe wejście – *bias*. Dla warstwy wyjściowej zastosowano specjalną funkcję aktywacji „Softmax”, opisaną w punkcie 4.2.2. Dzięki tej funkcji na wyjściu warstwy ukrytej otrzymujemy wartości, które odpowiadają swoistemu „prawdopodobieństwu” przypisanemu danemu typowi SBT. Dzięki tej funkcji suma wartości wszystkich węzłów wyjściowych zawsze jest równa 1.

6.2. Proces uczenia sieci

Po zdefiniowaniu budowy sieci kolejnym etapem tworzenia sztucznej sieci neuronowej był proces uczenia. Do uczenia zastosowano algorytm Adam (*Adaptive Moment Estimation*). Algorytm ten jest jedną z metod, która na bieżąco dostosowuje swoje parametry podczas procesu uczenia. Zaletą tego algorytmu w stosunku do zwykłego algorytmu SGD (*stochastic gradient descent*) jest znaczne przyspieszenie procesu uczenia, lepsza zbieżność rozwiązania oraz większa ostateczną dokładność modelu.

Sam algorytm (rysunek 6-3) zaimplementowany w bibliotece Tensorflow, wymaga jedynie zadania początkowych parametrów. W omawianym przypadku parametry te są parametrami domyślnymi w Tensorflow i przyjmują wartości:

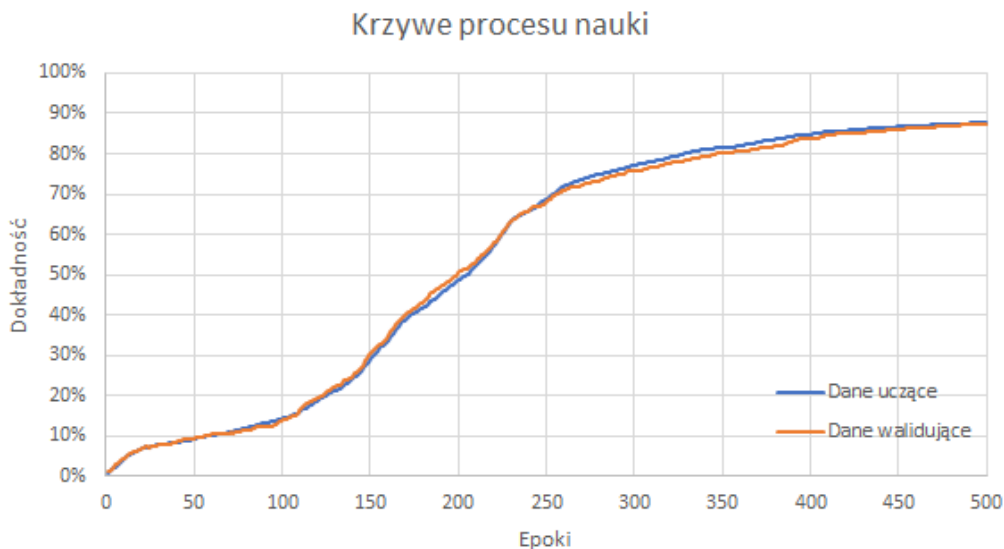
- współczynnik uczenia alfa: **0,001**
- beta1: **0,9**
- beta2: **0,99**
- epsilon: **10^{-8}**

Algorithm 1: *Adam*, our proposed algorithm for stochastic optimization. See section 2 for details, and for a slightly more efficient (but less clear) order of computation. g_t^2 indicates the elementwise square $g_t \odot g_t$. Good default settings for the tested machine learning problems are $\alpha = 0.001$, $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$. All operations on vectors are element-wise. With β_1^t and β_2^t we denote β_1 and β_2 to the power t .

Require: α : Stepsize
Require: $\beta_1, \beta_2 \in [0, 1)$: Exponential decay rates for the moment estimates
Require: $f(\theta)$: Stochastic objective function with parameters θ
Require: θ_0 : Initial parameter vector
 $m_0 \leftarrow 0$ (Initialize 1st moment vector)
 $v_0 \leftarrow 0$ (Initialize 2nd moment vector)
 $t \leftarrow 0$ (Initialize timestep)
while θ_t not converged **do**
 $t \leftarrow t + 1$
 $g_t \leftarrow \nabla_{\theta} f_t(\theta_{t-1})$ (Get gradients w.r.t. stochastic objective at timestep t)
 $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$ (Update biased first moment estimate)
 $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$ (Update biased second raw moment estimate)
 $\hat{m}_t \leftarrow m_t / (1 - \beta_1^t)$ (Compute bias-corrected first moment estimate)
 $\hat{v}_t \leftarrow v_t / (1 - \beta_2^t)$ (Compute bias-corrected second raw moment estimate)
 $\theta_t \leftarrow \theta_{t-1} - \alpha \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon)$ (Update parameters)
end while
return θ_t (Resulting parameters)

Rys. 6-3. Algorytm Adam (Kingma et. al., 2014)

Proces uczenia przeprowadzono wykorzystując znormalizowany, losowo potasowanych zestaw danych uczących. Uczenie podzielone jest na epoki, podczas jednej epoki algorytm uczący dokonuje uaktualnienia wartości wag wszystkich połączeń między neuronami na podstawie zestawu danych uczących. Po każdej takiej epoce określana jest dokładność sieci na podstawie danych walidacyjnych, a następnie rozpoczyna się kolejna epoka prowadząca do kolejnego uaktualnienia wartości wag, aż do uzyskania zadawalającej dokładności sieci na podstawie danych walidacyjnych. Na wykresie 6-4 przedstawiono przykładowy proces uczenia sieci. Wraz z kolejnymi epokami dokładność sieci na zestawie danych uczących, jak i walidujących rośnie aż do pewnego poziomu, gdzie się stabilizuje.



Rys. 6-4. Dokładność sieci zmieniająca się wraz z kolejnymi epokami

Dane walidacyjne nie uczestniczą w procesie uczenia, co oznacza że sieć nigdy nie „widziała” tych danych. Dokładność taka jest więc dobrą informacją w jaki sposób przebiega sam proces uczenia i jakie są zdolności sieci do uogólniania. Obserwując tę dokładność można zatrzymać proces uczenia, gdy dokładność przestaje rosnąć lub nawet spadać (efekt przeuczenia sieci – *overfitting*).

6.3. Proces weryfikacji sieci

Jak już zostało wspomniane, na każdym etapie procesu uczenia (zasadniczo po każdej epoce) oraz po zakończeniu procesu uczenia, określana była aktualna dokładność sieci na podstawie danych walidacyjnych.

Każdą sieć analizowaną w tej pracy, poddawano trzykrotnemu procesowi uczenia. Początkowe wartości wag przed procesem uczenia inicjowane są w sposób losowy. Prowadzi to do sytuacji, gdzie powtórzenie procesu uczenia od nowa daje minimalnie różniące się rezultaty – różnice w wartości wag i ostatecznej dokładności po tej samej ilości epok uczących. Dzięki kilkukrotnemu przeprowadzeniu procesu uczenia można określić średnią i maksymalną dokładność danej budowy sieci, na podstawie których są one później porównywane, a następnie z nich wybierana jest sieć o największej dokładności.

Po wybraniu najlepszej sieci (o najwyższej dokładności na danych walidujących) jest ona następnie weryfikowana na podstawie danych testowych. I to dokładność sieci na podstawie danych testowych jest ostateczną informacją o możliwościach zastosowania sieci w rozpatrywanym zagadnieniu.

7. PREZENTACJA I ANALIZA WYNIKÓW

Wynikiem procesu uczenia sieci są odpowiednio dobrane wagi połączeń między neuronami. Ze względu na dość duży rozmiar sieci neuronowej, dla największej sieci ilość tych wag (stopni swobody) przekracza cztery tysiące. Z tego względu pominięto ich prezentację. Przedstawione zostaną jedynie wyniki po procesie uczenia (dokładności sieci na danych walidacyjnych) oraz dokładność najlepszej sieci na danych testowych i porównanie jej do metod analitycznych rozpoznawania gruntu.

Uczenie sieci zostało wykonane na komputerze przenośnym Asus S43SN z procesorem Intel Core i5-2410M 2.3 GHz z systemem operacyjnym Windows 7. Oprogramowanie użyte do stworzenia tej sieci to między innymi:

- Python 3.5.2,
- Tensorflow 1.1.0,
- Pandas 0.19.2,
- Matplotlib 1.5.3,
- Numpy 1.12.1,
- Sublime Text.

7.1. Dokładność sieci

Na każdej z sieci analizowanej w tej pracy trzykrotnie został przeprowadzony proces uczenia. Następnie określono maksymalną i średnią dokładność każdej z sieci, dzięki czemu możliwe było porównanie ich między sobą. Dokładne wyniki zawarto w tabeli 7-1, gdzie posortowano je po średniej dokładności – sieci o najwyższej średniej dokładności znajdują się najwyżej.

Co łatwo zauważyć, w omawianym zadaniu najlepiej spisują się sieci o większej ilości neuronów w warstwach ukrytych. Z analizowanych sieci najlepszą średnią, jak i najlepszą maksymalną dokładność uzyskała sieć o budowie 5-25-40-40-25-9, tj. średnią dokładność 98.2% i maksymalną 98.8% przy trzech tysiącach epok uczących. Pozostałe sieci uzyskały odrobinę niższą dokładność, jednak różnice nie są bardzo duże.

Jedynie najlepszą sieć sprawdzono przy 3000 epok uczących. Pozostałe sieci były sprawdzane dla ilości epok zależną od stopnia skomplikowania sieci. W najmniejszych sieciach wzrost ilości epok uczących mógł doprowadzić do przeuczenia, a na pewno nie prowadził do wzrostu dokładności sieci. Jedynie największa sieć 5-25-40-40-25-9, ze względu na swój rozmiar, została poddana tak długiemu procesowi uczenia. Jak widać wpłynęło to jedynie minimalnie na średnią dokładność (z 97% do 98,2%, tj. około jeden

punkt procentowy różnicy) wydłużając proces uczenia trzykrotnie (z 1000 epok uczących trwających 1 minutę i 15 sekund do 3000 epok uczących trwających 3 minut i 10 sekund).

Tabela 7-1. Dokładności każdej analizowanej sieci

Sieć		Liczba epok uczących	Maksymalna dokładność	Średnia dokładność	Dokładność poszczególnych procesów uczenia		
Budowa	Uwagi				1	2	3
5-25-40-40-25-9	czas uczenia: 3m10s	3000	98.8%	98.2%	98.7%	98.8%	97.1%
5-25-40-40-25-9	czas uczenia: 1m15s	1000	97.5%	97.0%	97.5%	96.2%	97.4%
5-20-30-30-20-9		1000	96.3%	95.3%	94.0%	96.3%	95.7%
5-25-40-40-25-9	F. Akt.: Sigmoidalna	1000	95.7%	95.1%	94.5%	95.0%	95.7%
5-20-20-20-9		500	93.8%	92.3%	91.7%	91.3%	93.8%
5-25-40-25-9		500	93.6%	92.1%	93.6%	90.9%	91.8%
5-25-25-25-9		500	93.5%	91.7%	92.7%	93.5%	89.0%
5-30-30-9		500	92.3%	91.7%	91.9%	92.3%	91.0%
5-15-20-20-15-9		1000	94.2%	91.5%	89.4%	90.8%	94.2%
5-40-40-9		500	92.6%	91.3%	92.6%	90.9%	90.5%
5-30-30-30-9		500	93.3%	90.9%	93.3%	91.6%	87.9%
5-20-20-9		500	91.6%	90.0%	88.9%	91.6%	89.5%
5-20-30-20-9		500	91.4%	89.0%	87.6%	91.4%	88.0%
5-25-40-25-9	F. Akt.: Sigmoidalna	500	91.6%	88.8%	85.6%	91.6%	89.3%
5-25-25-9		500	91.1%	87.9%	89.8%	91.1%	82.8%
5-15-20-15-9		500	89.9%	87.5%	89.9%	85.9%	86.7%
5-15-15-15-9		500	88.4%	86.5%	87.4%	83.7%	88.4%
5-10-10-10-9		500	88.0%	85.7%	88.0%	85.9%	83.3%
5-15-15-9		500	88.4%	84.3%	84.5%	80.0%	88.4%

Dokładność sieci nie rośnie proporcjonalnie do ilość neuronów. Najmniejsza sieć i za razem najłabsza posiada 470 stopni swobody (połączeń między neuronami w sieci), w porównaniu do 4085 stopni swobody najlepszej sieci. Dziewięciokrotny wzrost ilości wag i skomplikowania sieci wiąże się jedynie ze wzorstem dokładności sieci o około dziesięć punktów procentowych. Prawdopodobnie kolejne zwiększenie ilości neuronów i połączeń mogłoby doprowadzić do kolejnego, lecz bardzo małego, zwiększenia dokładności. Wiązałoby się to jednak ze znacznym wzrostem czasu uczenia sieci, a nie gwarantowałoby znacznego zwiększenia dokładności.

Dwie sieci neuronowe zostały dodatkowo sprawdzone pod kątem wpływu funkcji aktywacji na proces uczenia. W tych sieciach zamiast funkcji ReLU zastosowano funkcję sigmoidalną. W obu przypadkach negatywnie wpłynęło to na dokładność sieci, więc dla pozostałych sieci pozostawiono funkcję aktywacji jako funkcję ReLU.

Dobór budowy sieci jest najtrudniejszym elementem pracy nad sieciami neuronowymi. Wymaga to albo doświadczenia, albo analizowania wielu różnych rozmiarów sieci neuronowych, aby uzyskać tą najlepszą. Są pewne ogólne zasady, które często pojawiają się w literaturze, jednak są to jedynie zalecenia. Jednym z takich zaleceń jest zasada aby ilość stopni swobody (wag - połączeń neuronów) była mniejsza od ilości danych uczących. I ten warunek został spełniony dla każdej sieci. (4085 stopnie swobody największej sieci do 6851 danych uczących). Kolejne zwiększanie budowy mogłoby doprowadzić do złamania tej

zasady, a co ważniejsze doprowadzić do przeuczenia sieci. W związku z czym sieć o budowie 5-25-40-40-25-9 (zaznaczona w tabeli na zielono) uznana została jako najlepszą możliwą siecią w danym przypadku i dla posiadanych danych.

7.2. Skuteczność wybranej sieci na tle innych metod

Wybraną sieć neuronową można porównać do metod analitycznych opisywanych w poprzednich punktach tej pracy. Do porównania tych metod posłużą dane testowe, czyli takie, których sieć nie widziała podczas procesu uczenia czy walidacji.

Dokładność sieci neuronowej na tych danych z podziałem na każdy typ SBT przedstawiono w tabeli 7-2. Trudno wskazać typ SBT, z którym sieć ma problem rozpoznać. Dokładność utrzymuje się na średnim poziomie ponad 98%. Dla całości danych testowych dokładność wynosi 98.9%.

Tabela 7-2. Dokładności sieci neuronowej dla każdego SBT

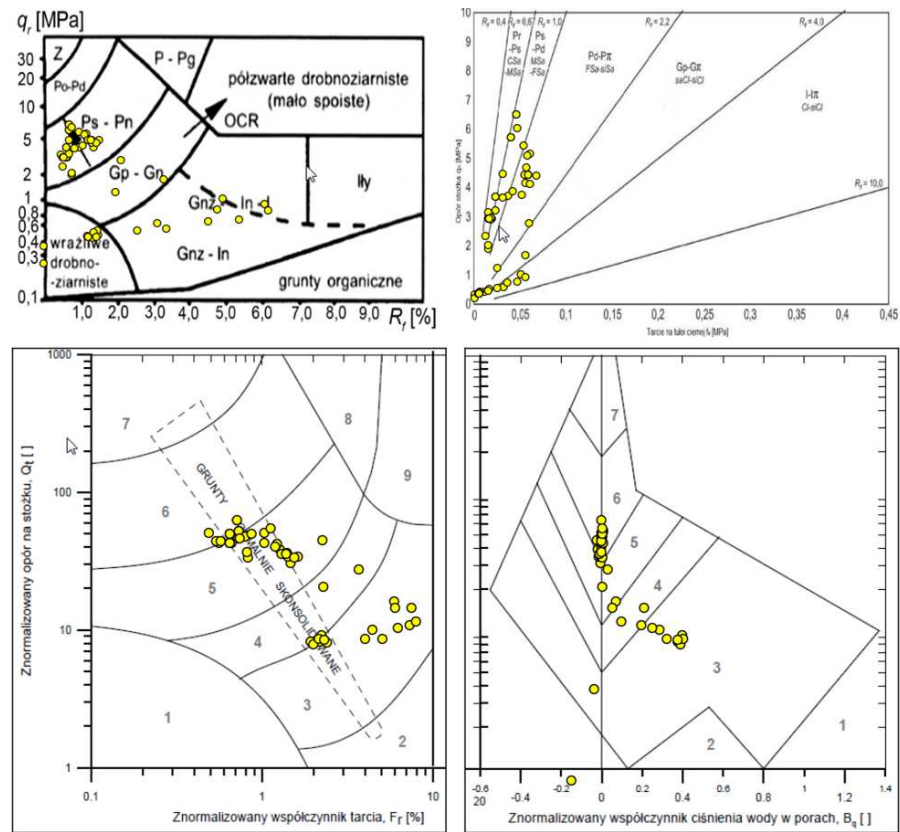
SBT	Liczba przypadków testowych	Dokładność sieci dla danego SBT
1	0	-
2	19	100.0%
3	385	99.2%
4	53	98.1%
5	97	97.9%
6	427	98.8%
7	0	-
8	0	-
9	0	-

Kolejnym krokiem jest sprawdzenie jak sprawdzają się inne metody. W tym celu na nomogramy Robertsona z PN-B, Marre`a z PN-B oraz Robertsona z 1990 r naniesione zostały dane testowe (rysunki 7-1 do 7-5), zachowując podział na SBT jak przy podziale danych do sieci neuronowych (tab. 5-5)

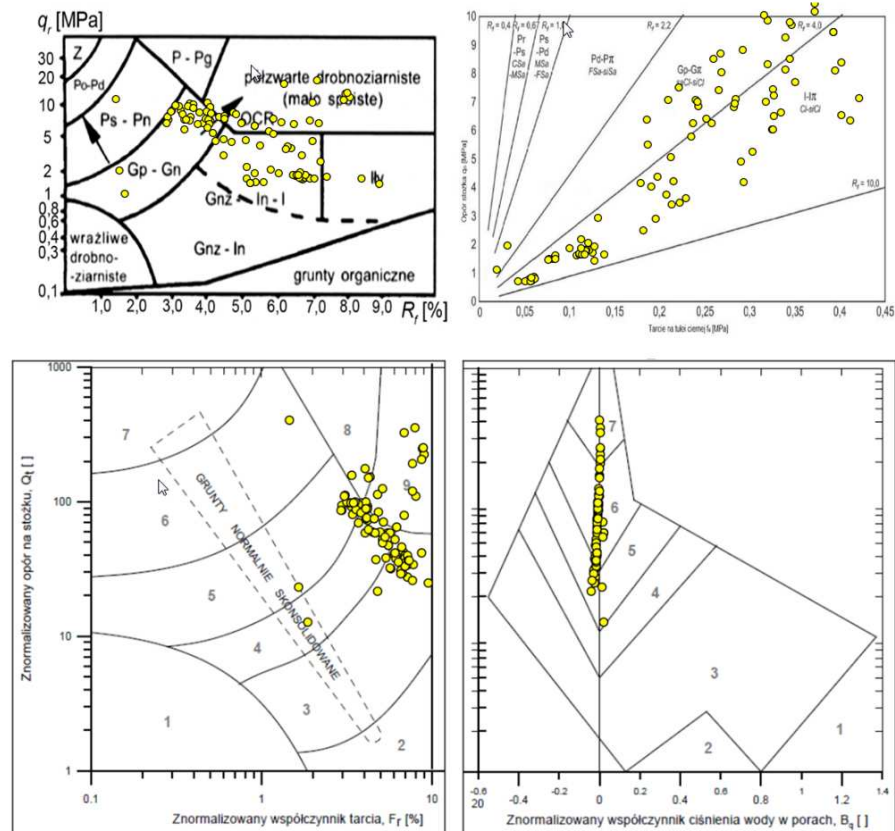
Na podstawie analizy tych wykresów można stwierdzić, że trudno na ich podstawie jednoznacznie określić typ SBT, a na pewno niemożliwe jest wyznaczenie rodzaju gruntu. Szczególnie jest to widoczne dla gruntów organicznych – SBT 2, gdzie dokładność rozpoznania wynosi praktycznie zero (rys 7.1).

Znacznie lepiej ma się sytuacja dla SBT 3 (rys 7-2). Na wszystkich trzech wykresach można odczytać, że dane gunty są w większości gruntami spoistymi, z pogranicza glin i iłów. Według wykresu Robertsona z 1990 roku można przyporządkować typ SBT jako 3 lub 4.

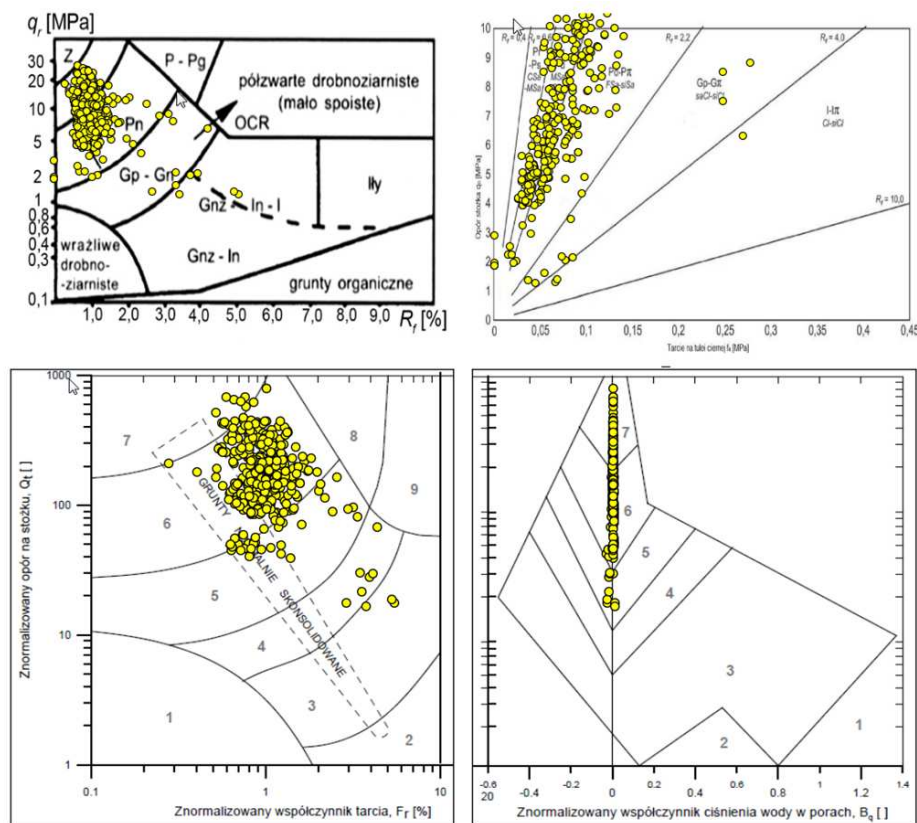
Trudno jest dobrze przeanalizować kolejne dwa typy SBT ze względu na niedużą liczbę dostępnych danych. Zasadniczo trudno jest wskazać różnicę między SBT 5 (rys 7-5) a SBT 3 (rys. 7-2). Wynika to prawdopodobnie z zaproponowanego w tej pracy przydziału gruntu do danego typu SBT. Zaproponowanie innego podziału być może zwiększyłoby dokładność rozpoznania na wykresach, a nie wpłynęło by na dokładność sieci neuronowej.



Rys. 7-3. Wykresy z naniesionymi gruntami danych testowych – SBT 4



Rys. 7-4. Wykresy z naniesionymi gruntami danych testowych – SBT 5



Rys. 7-5. Wykresy z naniesionymi gruntami danych testowych – SBT 6

7.3. Propozycja nowej metody określania rodzaju gruntu

Jak widać stosowanie nomogramów daje przybliżone informacje na temat rodzaju gruntu, jaki znajduje się w podłożu. Są to jednak informacje obarczone dużym błędem, a uzyskanie dokładnych informacji wymaga wykonania wierceń.

Z kolei sieci neuronowe mogą być bardzo dobrym narzędziem o bardzo dużej dokładności przewidywania. Gdyby zebrać wystarczającą ilość danych, pokrywającą w równomierny sposób wszystkie grunty lub typy SBT, to można by zbudować sieć, która w bardzo efektywny sposób będzie w stanie je rozpoznać. Wymaga to jednak dalszych analiz. Zaprezentowany tu efekt prac jest zaledwie pierwszym krokiem w dłuższym procesie stworzenia takiej metody. Niezbędne będzie odpowiednie zebranie, podzielenie i przefiltrowanie danych, niekoniecznie w zaprezentowany sposób. Następnie testowanie tak zbudowanej sieci na wielu przypadkach testowych. Praca ta miała jedynie na celu zweryfikowanie przydatności sztucznych sieci neuronowych w tym zagadnieniu.

8. PODSUMOWANIE

Sztuczne sieci neuronowe są tematem ciągłych prac i rozwojów. Tylko podczas okresu pisania tej pracy wydarzyło się wiele, co doprowadziło do zwiększenia możliwości SSN. Kolejne lata za pewne doprowadzą do kolejnych odkryć, jak i co raz ciekawszych jej zastosowań. Po głębszym poznaniu ich możliwości, można zdecydowanie stwierdzić, że geotechnika może być jedną z dziedzin, w której sieci neuronowe pokażą swoje mocne strony. Nowoczesne metody badawcze dostarczają ogromne ilości informacji, których nie jesteśmy w stanie przeanalizować w sposób tradycyjny. Stanowi to doskonałą przestrzeń do zastosowania SSN w przeróżnych zagadnieniach związanych z geotechniką

8.1. *Przydatność w rozpoznawaniu gruntów*

Po porównaniu możliwości sieci do pozostałych metod określania rodzaju gruntu, oczywistym jest stwierdzenie, że sztuczne sieci neuronowe bardzo dobrze nadają się do rozpatrywanego zagadnienia. Dane zagadnienie w takiej formie może być w bardzo prosty sposób zmodelowane jako sieć neuronowa. Podczas tworzenia tej pracy praktycznie za każdym razem wyniki były bardzo dobre bez względu na budowę sieci czy analizowane parametry. Utwierdza to w przekonaniu, że możliwy jest dalszy rozwój tej metody.

8.2. *Możliwe kierunki rozwoju*

Jak już zostało wspomniane, efekty tej pracy są dopiero pierwszym krokiem. Następnymi krokami, oprócz oczywistego zwiększenia zakresu danych, może być zastosowanie innego typu sieci neuronowej.

Ciekawym zadaniem wydaje się być analizowanie pojedynczego sondowania jako całości, pomiar po pomiarze wprowadzanym do sieci neuronowej. Do tego celu należałoby zastosować sieci rekurencyjne, być może jedne z najciekawszych sieci typu LSTM (ang. *long-short term memory*).

Kolejną możliwością jest zastosowanie innego podziału gruntów na SBT, być może nawet rezygnacja z takiego podziału na SBT i zastosowanie innego, na przykład wyrażającego procentową zawartość ilu, pyłu i piasku. W tej pracy nie brano pod uwagę tych warstw gruntów, które miały w sobie domieszki innych. Można by spróbować zaaplikować również i te dane do procesu uczenia sieci. W końcu jedną z zalet SSN jest pewna odporność na „zaszumienie” danych.

Można także się zastanowić czy inne parametry wprowadzane do sieci nie mają jakiegoś wpływu na działanie sieci. Być może kąt obrotu stożka podczas wciskania zwiększyłby dokładność tej sieci. Samą sieć można również zastosować do przewidywania

nie tylko rodzaju gruntu (czy SBT), ale również do innych parameterów. Ciekawym byłoby zastosowanie jej do przewidywania spójności i kąta tarcia wewnętrznego, również na podstawie badań CPTu lub innych.

Podsumowując, możliwości wykonania kolejnego kroku w badaniach nad przydatnością sieci neuronowych w geotechnice jest wiele. Aktualnie największym ograniczeniem jest dostępność do zbioru dużej ilości danych do analizy. Jeśli jednak przeskoczy się ten problem, to wszystkie wymienione propozycje, jak i wiele innych, stają się możliwymi do wykonania.

9. BIBLIOGRAFIA

9.1. Publikacje

- Bagińska I. (2012): Analiza oceny rodzaju gruntu ustalonego na podstawie badań CPTU, *Geoinżynieria drogi mosty tunele*, Vol. 37, No. 2, p. 38-45
- Bagińska I. (2015): Panaceum na rozpoznanie podłoża gruntowego: CPTM czy CPTU?, *Geoinżynieria drogi mosty tunele*, Vol. 50, No. 1, p. 28-32
- Bajda M., Biliniak M., Lech M. (2012): Wykorzystanie sondy CPTU 15 cm² do badań podłoża gruntowego, *Acta Scientiarum Polonorum. Architectura*, Vol. 12, No. 3, p. 61-72
- Douglas B.J., Olsen R.S. (1981): Soil classification using electric cone penetrometer, *Symposium on Cone Penetration Testing and Experience*, p. 209-227
- Godlewski T. (2013): Interpretacja badań polowych a Eurokod 7, *Acta Scientiarum Polonorum. Architectura*, Vol. 11, No. 2, p. 25-33
- Hebb D.O. (1949): *The Organization of Behavior*, Wiley
- Hinton G.E., Salakhutdinov R.R. (2006): Reducing the dimensionality of data with neural networks, *Science*, Vol. 313, No. 5786, p. 504-507
- Kowalska M. (2014): Identyfikacja rodzaju gruntu oraz parametrów wytrzymałościowych na podstawie wyników badań sondą statyczną CPTU, *Przegląd komunikacyjny*, No. 3, p. 91-103
- Lunne T., Robertson P., Powell J. (1997): *Cone Penetration Testing in Geotechnical Practice*, Spon Press
- McCulloch, W.S., Pitts W.H. (1943): A Logical Calculus of the Ideas Immanent in Nervous Activity, *Bulletin of Mathematical Biophysics*, Vol 7, p. 115-133
- Moore E.G. (1965): Cramming more components onto integrated circuits, *Electronics Magazine*, Vol. 38, No. 8, p. 114-117
- Projekt badawczy Nr 4 T07E 047 30 (2008): *Zależności regionalne parametrów geotechnicznych na podstawie sondowań, badań laboratoryjnych i pomiarów osiadań*, ITB
- Robertson P.K. (1990): Soil classification using the cone penetration test, *Canadian Geotechnical Journal*, Vol. 27, No. 1, p. 151-158
- Rosenblatt F. (1957): *The perceptron, a perceiving and recognizing automaton*, Cornell Aeronautical Laboratory
- Rumelhart D.E., Hinton G.E., Williams R.J (1986): Learning representations by back-propagating errors, *Nature*, Vol. 323, p. 533-536

- Schmertmann J.H. (1978): *Guidelines for Cone Penetration Test. (Performance and Design)*, Federal Highway Administration
- Sikora Z. (2009): *Symulacje komputerowe w geotechnice. Problem szacowania parametrów materiałowych i ważności rozwiązań numerycznych*, XXIV Ogólnopolskie warsztaty pracy projektanta konstrukcji, p. 337-394
- Sulewska M.J. (2011): Applying Artificial Neural Networks for analysis of geotechnical problems, *Computer Assisted Mechanics and Engineering Sciences*, Vol. 18, p. 231-241
- Tumay M.T., Karasulu Y.H., Młynarek Z., Wierzbicki J. (2011) Effectiveness of CPT-based classification charts for identification of subsoil stratigraphy. *Proceedings of the 15th European Conference on Soil Mechanics and Geotechnical Engineering*. p. 91-98.
- Widrow B. (1990): 30 years of adaptive neural networks: perceptron, Madaline, and backpropagation, *Proceedings of the IEEE*, Vol. 78, No. 9, p. 1415-1442

9.2. Normy

- PN-B-04452:2002. *Geotechnika. Badania Polowe.*, PKN
- PN-EN 1997-2:2009. *Eurokod 7. Projektowanie geotechniczne. Część 2: Rozpoznanie i badanie podłoża gruntowego*, PKN

9.3. Materiały informacyjne

- Asimov Institute. *The neural network Zoo*, <http://www.asimovinstitute.org/neural-network-zoo> (2017)
- Gouda. *History of Cone Penetration Testing (CPT)*, <https://www.gouda-geo.com/products/cpt-equipment/background-information/history-of-cone-penetration-testing-cpt> (2017)
- Lankelma. *Piezocone*, <http://www.lankelma.co.uk/cone-penetration-testing/cone-penetrometers/piezocone> (2017)
- Neurosoft. *Sztuczna inteligencja – algorytm uczenia sztucznych sieci neuronowych*, http://www.neurosoft.edu.pl/media/pdf/tkwater/sztuczna_inteligencja/2_alg_ucz_ssn.pdf (2017)
- Wikipedia. *Funkcja aktywacji*, https://pl.wikipedia.org/wiki/Funkcja_aktywacji (2017)
- Wikipedia. *Sieć neuronowa*, https://pl.wikipedia.org/wiki/Sieć_neuronowa (2017)