Wstęp do Machine Learning (IML)

Informatyka – SI, inżynierskie stacjonarne, 5 sem., 30 godz. wykładu, 30 godz. laboratorium

Dr inż. Paweł Syty, psyty@pjwstk.edu.pl

Politechnika Gdańska, Instytut Fizyki i Informatyki Stosowanej – adiunkt

- relatywistyczna fizyka atomowa; inżynieria biomedyczna; inżynieria oprogramowania; uczenie maszynowe

Polsko-Japońska Akademia Technik Komputerowych – współpraca (MLR, dawniej TIN)

Radiato.ai, TITAN, MIDAS – Product Owner, AI Scientific Expert, webmaster

AiBay (Zatoka Sztucznej Inteligencji) – członek założyciel, webmaster

NlightniN Production – Software Engineer & Machine Learning Expert w projekcie NlightVR (EEG + VR) oraz AI MySalad

Dawniej: **Fido Intelligence** (uczenie maszynowe w lingwistyce); **Currenda** (kontekstowe rozpoznawania tekstów, w szczególności pozwów sądowych); **Adar** (problem transportowy); i wiele innych…

Forma zaliczenia przedmiotu:

- ocena indywidualnego projektu głównego (10 p.)
- ocena zadań cząstkowych, realizowanych na laboratorium (40 p.)

Program zajęć

- Historia i definicja uczenia maszynowego. Modele maszynowe. Omówienie zastosowań uczenia maszynowego.
- Rola danych i ich jakości w uczeniu maszynowym. Przygotowywanie danych do procesu uczenia.
- Hiperparametry. Metryki. Badanie korelacji. Istotność statystyczna.
- Podstawy biologiczne sztucznych sieci neuronowych.
- Historia i podstawy sztucznych sieci neuronowych. Podstawowe architektury. Funkcje aktywacji.
- Metody uczenia sieci neuronowych. Funkcje błędu. Zdolność uogólniania sieci neuronowej.
- Wybór optymalnej architektury sieci neuronowej.
- Rozpoznawanie obrazów. Sieci splotowe. Wstęp do uczenia głębokiego.
- Metody regularyzacji. Sprawdzian krzyżowy.
- Sieci autoasocjacyjne. Transfer learning.
- Sieci rekurencyjne.
- Elementy wyjaśnialnej sztucznej inteligencji.
- Inne metody uczenia maszynowego: k-najbliższych sąsiadów, maszyna wektorów nośnych (SVM), drzewo decyzyjne, las losowy, klasyfikator bayesowski, logika rozmyta, systemy ekspertowe, algorytmy genetyczne, automaty komórkowe, liniowe metody mieszane, metody zespołowe.
- Uczenie nienadzorowane – analiza skupień metodą centroidów (k-means). Analiza szeregów czasowych.
- Praktyczne zagadnienia uczenia maszynowego, np. uczenie na CPU vs GPU.
- Zastosowania – rzeczywiste przykłady wykorzystania uczenia maszynowego.

Narzędzia:

- Interpreter języka Python 3.x, biblioteki TensorFlow, TensorBoard, tf-explain, sci-kit learn, OpenCV

Przykładowa literatura

Podstawowa

- Daniel T. Larose „Metody i modele eksploracji danych”, PWN 2022
- Aurélien Géron, „Uczenie maszynowe z użyciem Scikit-Learn i TensorFlow”, Helion 2020
- Robert Johansson „Matematyczny Python. Obliczenia naukowe i analiza danych z użyciem NumPy, SciPy i Matplotlib”, Helion 2021
- Sebastian Raschka, Vahid Mirjalili „Python. Machine learning i deep learning. Biblioteki scikit-learn i TensorFlow 2”, Helion 2021
- Aileen Nielsen, „Szeregi czasowe. Praktyczna analiza i predykcja z wykorzystaniem statystyki i uczenia maszynowego”, Helion 2020

Uzupełniająca

- Joel Grus, „Data science od podstaw. Analiza danych w Pythonie”, Helion 2020
- Dokumentacja pakietów Keras i Tensorflow

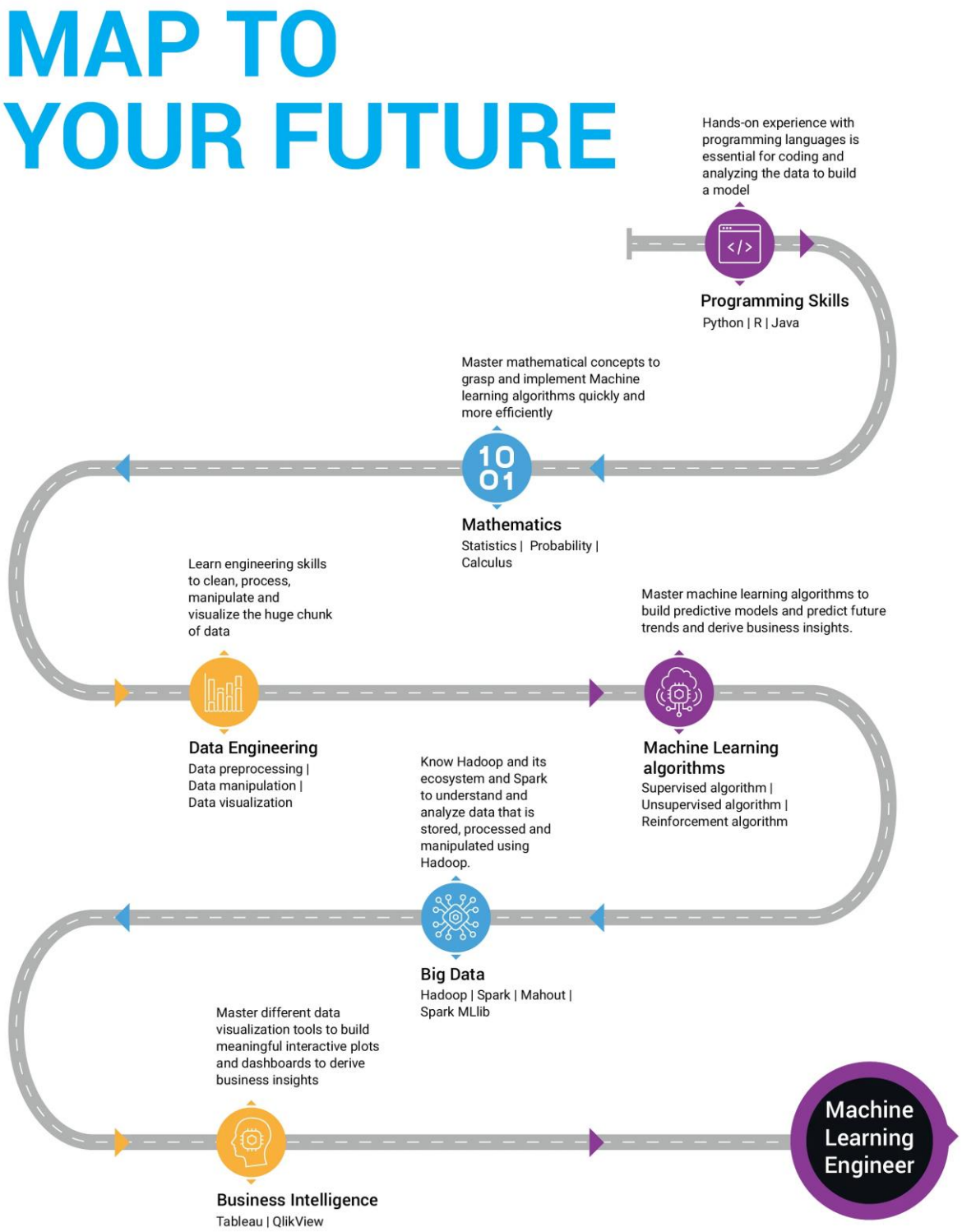
Efekty kształcenia

- Student zna i rozumie podstawowe zastosowania wybranych narzędzi uczenia maszynowego
- Student wie, że jakość danych ma kluczowe znaczenie w przypadku uczenia maszynowego
- Student zna i rozumie metody przechowywania danych dla uczenia maszynowego
- Student potrafi wybrać odpowiednie narzędzie do rozwiązania problemu zaistniałego w przedsiębiorstwie, a wymagającego użycia uczenia maszynowego
- Student potrafi poprawnie przygotować dane i stworzyć na ich podstawie zbiory uczące, walidujące i testowe
- Student potrafi dobrać właściwe parametry wejściowe, by zoptymalizować działanie wybranych algorytmów.
- Student potrafi przeprowadzić analizę otrzymanych wyników
- Student jest gotów do ciągłego podnoszenia swoich kompetencji zawodowych, osobistych i społecznych oraz zna możliwości dokształcania się przez całe życie

Wstęp do ML

Wykład 1.

Historia i definicja uczenia maszynowego. Omówienie zastosowań uczenia maszynowego. Rola danych i ich jakości w uczeniu maszynowym. Przygotowywanie danych do procesu uczenia.



Uczenie maszynowe

Dziedzina nauki, wchodząca w skład sztucznej inteligencji.

Sztuczna inteligencja – definicja wg. Komisji Europejskiej

> *Sztuczna inteligencja odnosi się do systemów zaprojektowanych przez ludzi, które ze względu na założony cel działają w świecie fizycznym lub cyfrowym, postrzegając swoje środowisko, interpretując zgromadzone ustrukturyzowane lub nieustrukturyzowane dane, wnioskując na podstawie wiedzy uzyskanej z tych danych i decydują o najlepszych działaniach / akcjach możliwych do podjęcia w kierunku osiągnięcia tego celu, zgodnie z predefiniowanymi parametrami.*

Uczenie maszynowe rozszerza tę definicję o **możliwość doskonalenia się sztucznego systemu** przy pomocy **zgromadzonego doświadczenia** (na podstawie dostępnych danych) i nabywania na tej podstawie **nowej wiedzy**.

Dziedzina interdyscyplinarna (głównie informatyka, robotyka, statystyka)

Arthur Samuel (1959; twórca programów szachowych, pionier SI): „*Uczenie maszynowe to dziedzina nauki, która daje komputerom możliwość uczenia się, lecz bez ich bezpośredniego programowania*”

Donald Michie (1991): "System uczący się wykorzystuje zewnętrzne dane empiryczne w celu tworzenia i aktualizacji podstaw dla udoskonalonego działania na podobnych danych w przyszłości oraz wyrażania tych podstaw w zrozumiałej i symbolicznej postaci"

Uwaga! Uczenie się „maszynowe” ma tu nieco inny kontekst, niż w pojęciu potocznym. Zadaniem procesu uczenia się systemu jest osiąganie rezultatów opartych na wiedzy cząstkowej, samodoskonalenie, tworzenie nowych pojęć, wnioskowanie indukcyjne

Kamienie milowe w rozwoju uczenia maszynowego

- programy do nauki szachów (Arthur Samuel, Stanford, USA, 1959)
- system ekspertowy Dendral (Stanford, USA, 1965) – identyfikacja molekuł związków organicznych
- program AM – Automated Mathematician (Douglas Lenat, Stanford, USA, 1977) – algorytmy heurystyczne do poszukiwania/udoskonalania twierdzeń
- TD-Gammon (Gerald Tesauro, IBM, USA, 1991) – program do gry w backgammona
- Deep Blue (IBM, USA, 1997) – program szachowy, pierwszy system komputerowy, który wygrał z aktualnym mistrzem świata
- Watson (IBM, USA, 2011) – system do gry w teleturnieju Jeopardy! (w Polsce – Va banque)

  Watson wygrał 77 141 $, eksperci (najlepsi zawodnicy w historii Jeopardy!): Ken – 24 000$, Brad – 21 600 $

  Reguły były niekorzystne dla maszyny, bo ludzie zgłaszają się słysząc pytanie natychmiast po jego zakończeniu, a maszyna ma opóźnienie zanim przeanalizuje pytanie i może się zgłosić

  Poprawność odpowiedzi to ok. 95%

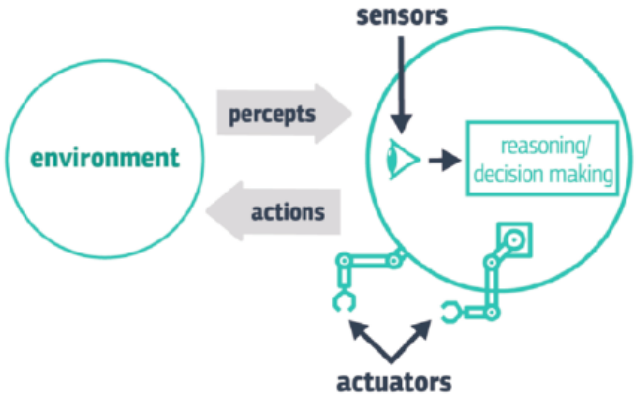
  Watson działa na superkomputerze IBM Blue Gene/P, 15 TB RAM, 2 880 rdzeni procesorów, 80 Tflop

  Baza danych: encyklopedie, słowniki, artykuły, bazy leksykalne, literaturę piękną. Użyto ok. 100 algorytmów do analizy tekstu

  Watson zaliczył parę nietypowych wpadek, np. kategoria: „Also On Your Computer Keys”, pytanie: „It's an abbreviation for Grand Prix auto racing”, prawidłowa odpowiedź: „What is F1?”. Watson jej nie udzielił

- Alphago (DeepMind – Google, 2017) – wygrana w GO
- ChatGPT (OpenAI, 2022) – interaktywny model uniwersalnego zastosowania, w formie chatbota

Główne cele i zastosowania uczenia maszynowego



- Zdobywanie nowej wiedzy dzięki interakcji z otoczeniem
- Uogólnianie zdobytej wiedzy
- Formułowanie reguł decyzyjnych
- Tworzenie nowych pojęć
- Wykrywanie prawidłowości w danych
- Modyfikowanie, uogólnianie i precyzowanie danych
- Użytkowane dużych baz danych, w szczególności poszukiwanie i analiza zależności między danymi
- Analiza, badanie i opracowywanie złożonych problemów naukowych i technicznych
- Prowadzenie działań w zmiennych warunkach (robotyka, sterowanie produkcją)

Przykładowe zastosowania praktyczne

- Rozpoznawanie mowy
  - rozpoznawanie mowy ludzkiej; automatyczne tłumaczenie; dyktowanie tekstów; interfejsy użytkownika sterowane głosem; automatyzacja głosem czynności domowych; interaktywne biura obsługi…
- Nawigacja i sterowanie
  - kierowanie pojazdem / robotem; odnajdywanie drogi w nieznanym środowisku; automatyzacja produkcji…
- Analiza i klasyfikacja danych
  - medycyna; rozpoznawanie pisma; aproksymacja danych; przewidywanie trendów; wykrywanie zależności; klasyfikacja obiektów; bankowość…

Wybrane metody uczenia maszynowego

- Wnioskowanie i uczenie z przykładów (proste wnioskowanie na podstawie posiadanych danych)
- Uczenie analityczne (uogólnianie na podstawie przykładów)
- Uczenie się zbioru reguł (np. klauzule Horna, reguły produkcji)
- Uczenie indukcyjne (empiryczne regularności -> hipotezy)
- Uczenie bayesowskie (wnioskowanie probabilistyczne)
- Uczenie przez wzmacnianie (kara / nagroda)

Inny podział metod

- Uczenie nadzorowane (ang. *supervised*)
    - Mamy dostęp do danych wejściowych jak i żądanych wyjściowych (zbiór treningowy)
- Uczenie nienadzorowane (ang. *unsupervised*)
    - Mamy dostęp do danych wejściowych, system sam dokonuje strukturyzacji danych
- Uczenie ze wzmocnieniem (ang. *reinforcement*)
    - Mamy dostęp do danych wejściowych i informację o tym, czy wykonane działanie przyniosło korzyść czy też nie

- Metody leniwe (ang. *lazy*) i gorliwe (ang. *eager*)
    - Te pierwsze nie wymagają wstępnego treningu

Federated learning (Google, 2017): Wykorzystanie urządzeń mobilnych do procesu uczenia

Modele

Zastosowanie wybranej metody uczenia maszynowego prowadzi do stworzenia modelu (inaczej – hipotezy lub ich zestawu). Model służy do dalszej predykcji na kolejnych danych.

Najczęściej stosowane modele:

- Sztuczne sieci neuronowe (w tym jako wynik tzw. uczenia głębokiego)
- Klasyfikator bayesowski
- Drzewo decyzyjne
- Las losowy
- k-najbliższych sąsiadów (kNN)
- Maszyna wektorów nośnych
- Algorytmy genetyczne
- Logika rozmyta
- Systemy ekspertowe
- Automaty komórkowe
- Liniowe modele mieszane
- Modele zespołowe (ang. *ensemble*), w tym modele wzmacniane (ang. *gradient boosting*)

Dane i ich rola w uczeniu maszynowym

"If you put into the machine the wrong figures, will the right answer come out?" – Charles Babbage (1864)

"Garbage in, garbage out" – The United States Internal Revenue Service (1963)

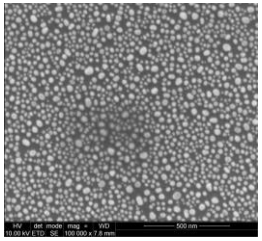
"There are no clean datasets." – Josh Sullivan, Booz Allen Hamilton VP, in Fortune (2015)

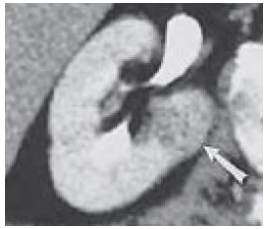
Problemy z danymi

- Wiedza może być niepewna, niepełna, niedokładna
  - niepewność: prawdziwość niektórych stwierdzeń nie jest pewna
  - niepełność: niektóre prawdziwe stwierdzenia nie są znane, lecz nie można z tego powodu zakładać ich nieprawdziwości
  - niedokładność: przynależność stwierdzenia do niektórych relacji nie jest znana dokładnie; dane mogą być sprzeczne
- Różne sposoby organizacji danych
  - Dane tekstowe

set_id	price	freight_length	load_date	unload_date	from_lat	from_lng	target_lat	target_lng
39	860.000000	717	"2014-03-03 20:00:00"	"2014-03-05 12:00:00"	52.4658000	6.7931000	47.4875000	5.0634000
39	500.000000	396	"2014-03-05 08:00:00"	"2014-03-06 08:00:00"	46.8856000	5.6597000	43.8768000	4.6303000
39	400.000000	496	"2014-03-06 16:00:00"	"2014-03-07 08:00:00"	43.7867000	5.0046000	44.6000000	0.7167000
39	1100.000000	1327	"2014-03-07 07:00:00"	"2014-03-10 08:00:00"	44.2354000	-0.9137000	45.9586000	11.4359000
39	2150.000000	1407	"2014-03-11 12:00:00"	"2014-03-13 12:00:00"	45.4928000	10.8849000	48.1149000	2.1030000

  - Dane binarne (dźwięki, obrazy)





- Klauzule Horna, formuły w postaci standardowej Skolema

$$L_0 \vee \neg L_1 \vee \cdots \vee \neg L_n$$

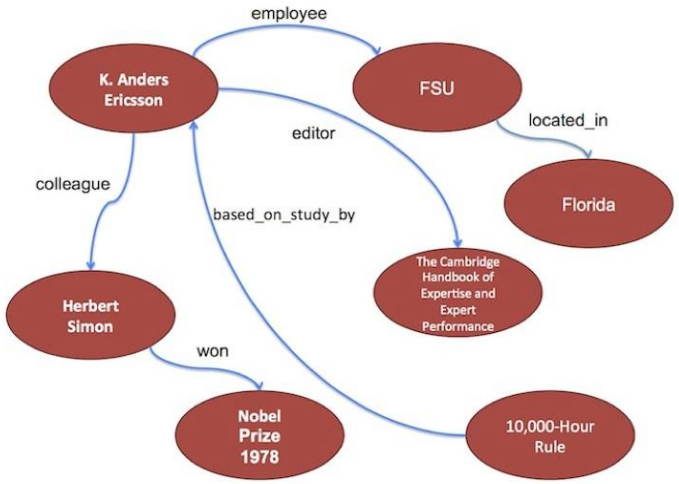
$$(\forall \ldots)(\forall \ldots) \ldots (\forall \ldots)\alpha.$$

$$(\forall x)(\forall v)(\neg P(x, h_1(x)) \vee P(h_2(x), b)) \wedge (\neg P(x, h_1(x)) \vee \neg R(v, b)) \wedge$$
$$\wedge (\neg Q(h_1(x), a) \vee P(h_2(x), b)) \wedge (\neg Q(h_1(x), a) \vee \neg R(v, b))$$

- Reguły produkcji

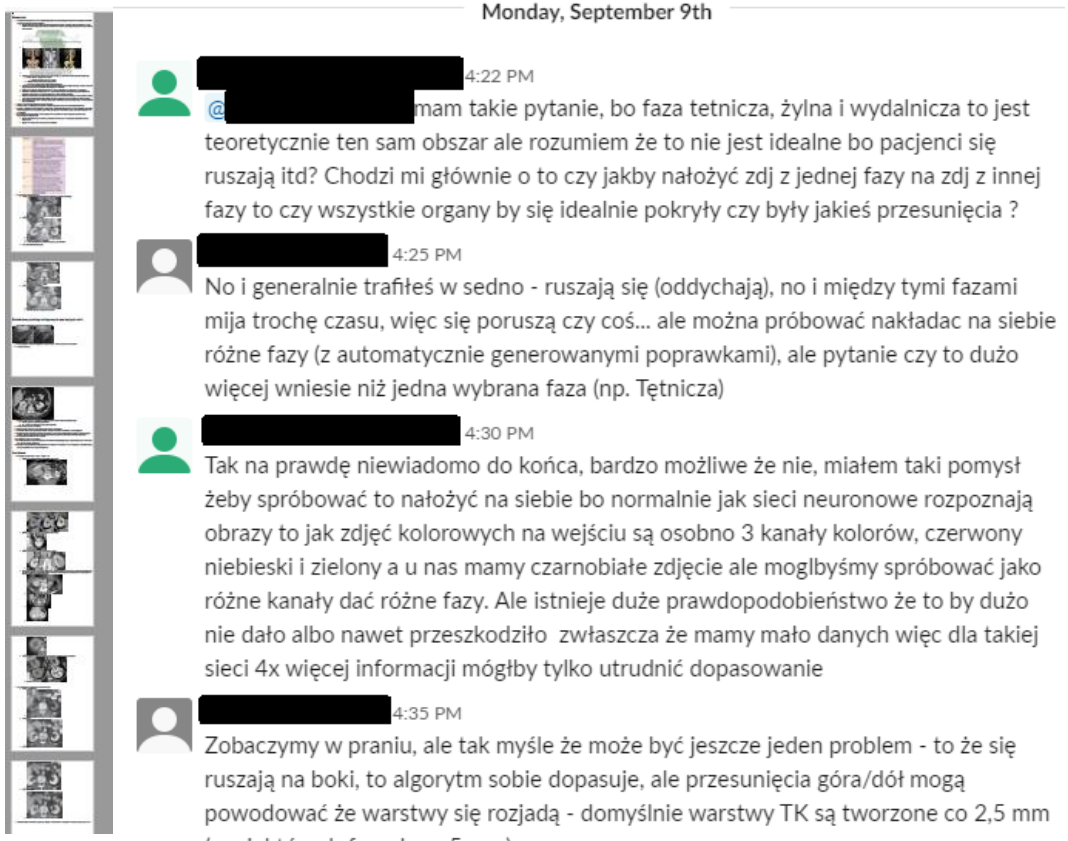
IF Kultura bakteryjna rozwinęła się we krwi
i odczyn jest gramopozytywny
i bakterie wniknęły przez jelito
i żołądek lub miednica są miejscem infekcji
THEN Istnieją silne poszlaki, że klasą bakterii, które są za to odpowiedzialne są Enterobacteriacae.

- Grafy wiedzy



- Różne formaty przechowywania danych
  - Pliki tekstowe ustrukturyzowane (CSV, JSON, XML, HTML) albo i nie…
  - Relacyjne (MySQL, MSSQL, PostgreSQL, Oracle…) i nierelacyjne bazy danych (MongoDB…)
  - Arkusze kalkulacyjne (Excel i inne)
  - Pliki PDF
  - Pliki HTML (Web scraping)
  - Obrazy skompresowane lub nie
  - Muzyka skompresowana lub nie
  - Tysiące innych formatów, otwartych / zamkniętych
- Różne kodowanie znaków
  - Unicode vs reszta świata
- Różne znaki końca wiersza
  - \n kontra \r\n
- Typy danych
  - Liczby i daty/godziny w różnych formatach
  - Teksty sformatowane i niesformatowane; białe znaki
  - Typy logiczne
  - Zbiory i enumeratory
  - Dane binarne

- Różny sposób dostępu do danych
  - Plik
  - Interaktywny frontend do systemu
  - Strumień danych
  - API
  - Systemy schyłkowe



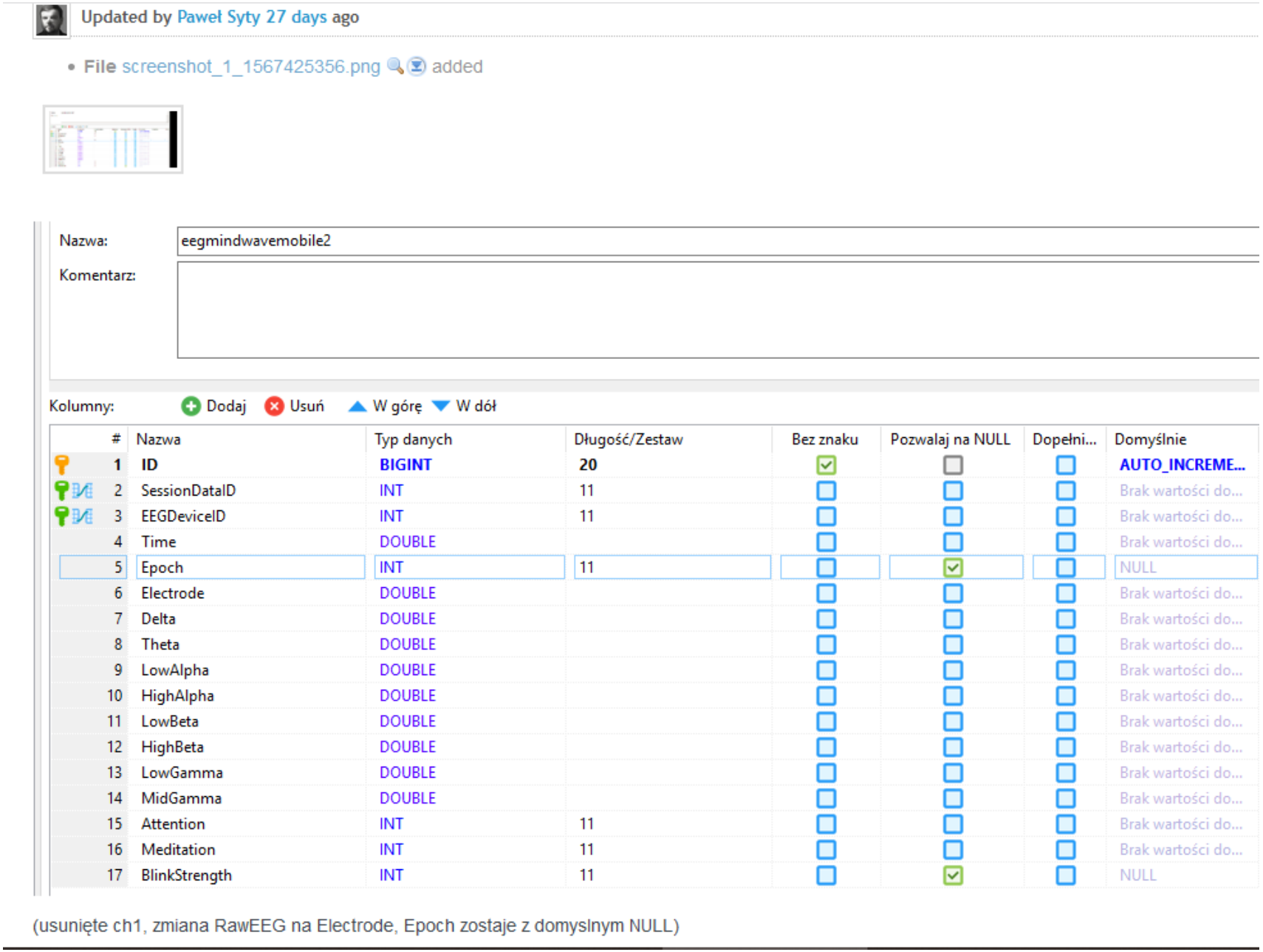
Proces obróbki danych

- Analiza i zrozumienie problemu
- Zebranie i przechowywanie danych
- Czyszczenie danych
- Uczenie maszynowe
- Reprezentacja wyników i ich wizualizacja
- Podsumowanie – czy udało się rozwiązać problem, jakie są ograniczenia rozwiązania, co można by zrobić inaczej, jakie są następne kroki

Uwaga techniczna

Każdy etap należy odpowiednio dokumentować! Na przykład w pracy w zespole SCRUM-owym: tworzyć zadanie w tablicy sprintów (np. „Import pliku HTML z danymi giełdowymi”) i opisywać wykonane w ramach zadania czynności (np. „W pliku wejściowym wystąpiły błędy formatowania, takie jak niezamknięte cudzysłowy i znaczniki HTML. Błędy te zostały poprawione za pomocą skryptu w Pythonie: correctHTML.py”).

Inny przykład dokumentacji zadania (modyfikacja tabeli w bazie danych):



Updated by Paweł Syty 27 days ago

- File screenshot_1_1567425356.png added

Nazwa: eegmindwavemobile2

Komentarz:

Kolumny: Dodaj Usuń W górę W dół

#	Nazwa	Typ danych	Długość/Zestaw	Bez znaku	Pozwalaj na NULL	Dopełni...	Domyślnie
1	**ID**	**BIGINT**	**20**	☑	☐	☐	**AUTO_INCREME...**
2	SessionDataID	INT	11	☐	☐	☐	Brak wartości do...
3	EEGDeviceID	INT	11	☐	☐	☐	Brak wartości do...
4	Time	DOUBLE		☐	☐	☐	Brak wartości do...
5	Epoch	INT	11	☐	☑	☐	NULL
6	Electrode	DOUBLE		☐	☐	☐	Brak wartości do...
7	Delta	DOUBLE		☐	☐	☐	Brak wartości do...
8	Theta	DOUBLE		☐	☐	☐	Brak wartości do...
9	LowAlpha	DOUBLE		☐	☐	☐	Brak wartości do...
10	HighAlpha	DOUBLE		☐	☐	☐	Brak wartości do...
11	LowBeta	DOUBLE		☐	☐	☐	Brak wartości do...
12	HighBeta	DOUBLE		☐	☐	☐	Brak wartości do...
13	LowGamma	DOUBLE		☐	☐	☐	Brak wartości do...
14	MidGamma	DOUBLE		☐	☐	☐	Brak wartości do...
15	Attention	INT	11	☐	☐	☐	Brak wartości do...
16	Meditation	INT	11	☐	☐	☐	Brak wartości do...
17	BlinkStrength	INT	11	☐	☑	☐	NULL

(usunięte ch1, zmiana RawEEG na Electrode, Epoch zostaje z domyslnym NULL)

Więcej o czyszczeniu danych

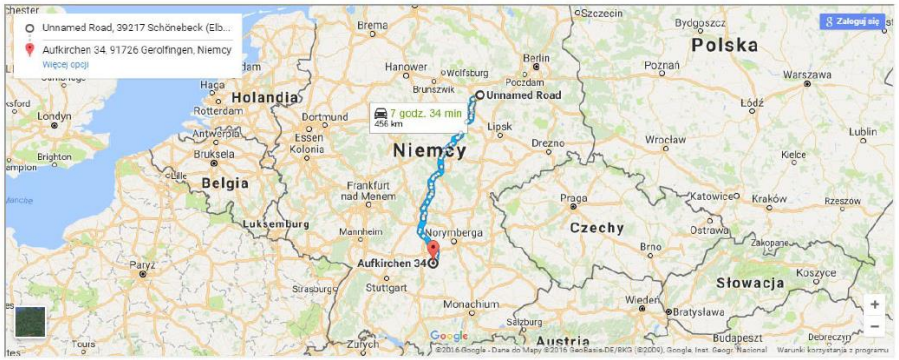
- Wizualne sprawdzenie danych

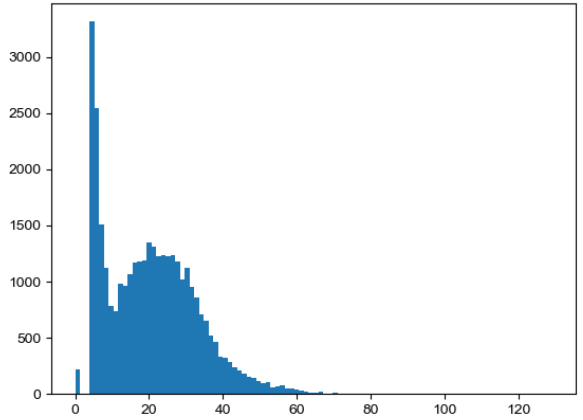
timestamp	rejestracja	VehicleOrDeviceName	lat	lon	licznik	ignition	speed	rpm	event	fuel	fuelpercentage	BatteryVoltage	Sterownik
1536867185		133129			0								pulson.133129
1536867185	XXX01ML	Kia XXX01ML	54.346984	18.635423	59208.6	false	0	0	28	2.57	5	12.59	albatross.59540
1536867185	XXX10ML	Kia XXX10ML	54.355079	18.65553	44766.8	false	0	0	12	3.758	7	12.94	albatross.59632
1536867185	XXX07ML	Kia XXX07ML	50.242721	19.128997	43118.3	false	0	0	32	3.366	6	11.94	albatross.59512
1536867185	XXX09ML	Kia XXX09ML	54.326324	18.321937	46733	false	0	0	3	4.203	8	12.62	albatross.78078
1536867185	XXX13ML	Kia XXX13ML	54.348262	18.671075	48846.1	false	0	0	3	0.408	1	12.65	albatross.59489
1536867185	XXX12ML	Kia XXX12ML	54.382152	18.28701	53559.2	false	0	0	3	1.394	3	12.81	albatross.78085
1536867185	XXX11ML	Kia XXX11ML	54.493053	18.438251	44072	false	0	0	3	0.355	1	12.94	albatross.59481
1536867185	XXX08ML	Kia XXX08ML	54.339691	17.889526	74831.6	false	0	0	3	2.125	4	12.81	albatross.78091
1536867185	XXX14ML	Kia XXX14ML	54.202232	16.180894	62530.3	false	0	0	3	2.38	4	12.62	albatross.78079
1536867185	XXX15ML	Kia XXX15ML	54.535095	17.741743	49293.9	false	0	0	28	0.61	1	13	albatross.59491
1536867185	XXX05ML	Partner XXX05ML	53.368419	20.407041	84346.2	false	0	0	3	132	220	12.84	albatross.78108
1536867185	XXX03ML	Partner XXX03ML	53.363834	20.426134	35788.8	false	0	0	3	127.059	212	12.75	albatross.78076
1536867185	XXX04ML	Boxer XXX04ML	54.321681	18.249071	50728	false	0	0	3	58.5	65	12.97	albatross.78080
1536867185	XXX06ML	Boxer XXX06ML	53.989288	20.411775	49789	false	0	0	3	86.4	96	12.62	albatross.78086
1536867185	XXX27MP	Kia XXX27MP	54.258892	18.631664	61599	false	0	0	3	0	0	12.71	albatross.78103
1536867185	XXX63MP	Kia XXX63MP	52.868949	20.606254	62320.4	false	0	0	32	1.77	3	12.84	albatross.59587
1536867185	XXX67MP	Kia XXX67MP	54.263526	18.656728	66361.3	false	0	0	3	3.387	6	12.68	albatross.78092
1536867185	XXX65MP	Kia XXX65MP	52.412635	16.989149	107577.8	false	0	0	28	1.68	3	12.84	albatross.78074
1536867185	XXX70MP	Kia XXX70MP	54.298584	18.61638	48915.6	false	0	0	32	0.816	2	12.71	albatross.59506
1536867185	XXX72MP	Kia XXX72MP	50.057506	22.134044	95296.8	false	0	0	3	3.673	7	12.68	albatross.59478
1536867185	XXX71MP	Kia XXX71MP	50.081901	20.000398	68362.1	false	0	0	3	0	0	12.91	albatross.59510
1536867185	XXX75MP	Crafter XXX75MP	54.419834	18.252807	29675.3	false	0	0	3	0	0	12.68	albatross.78088
1536867185	XXX74MP	Crafter XXX74MP	52.731365	19.69662	74437.8	false	0	0	28	-375	-500	13.2	albatross.78077
1536867185	XXX79MP	Crafter XXX79MP	52.291603	21.054407	92826.2	false	0	0	3	-362.5	-483	12.88	albatross.78087
1536867185	XXX64MP	Crafter XXX64MP	51.461696	19.214952	78263.3	false	0	0	3	-487.5	-650	12.91	albatross.78371
1536867185	XXX78MP	Crafter XXX78MP	51.853984	17.936836	74142.6	false	0	0	3	-225	-300	12.88	albatross.78357
1536867185	XXX76MP	Crafter XXX76MP	53.344074	18.193109	84148.2	false	0	0	3	-50	-67	12.71	albatross.78366
1536867185	XXX47MP	Kia XXX47MP	50.112693	18.540325	65651.4	false	0	0	32	3.456	7	12.65	albatross.59504
1536867185	XXX45MP	Kia XXX45MP	51.418785	21.963575	91425.9	false	0	0	3	1.394	3	13.07	albatross.78083

- Zastosowanie metod statystycznych
  - Policzenie i analiza średnich, median, odchyleń standardowych, percentyli i na tej podstawie identyfikacja anomalii w danych

```
MIN: 0.0
MAX: 429496729.5
ŚREDNIA: 199080.4651410694
MEDIANA: 19.8
ZAKRES: 429496729.5
ODCHYLENIE STANDARDOWE: 9244239.58149461
WARIANCJA: 85455965440071.64
PERCENTYL 90%: 36.6
```

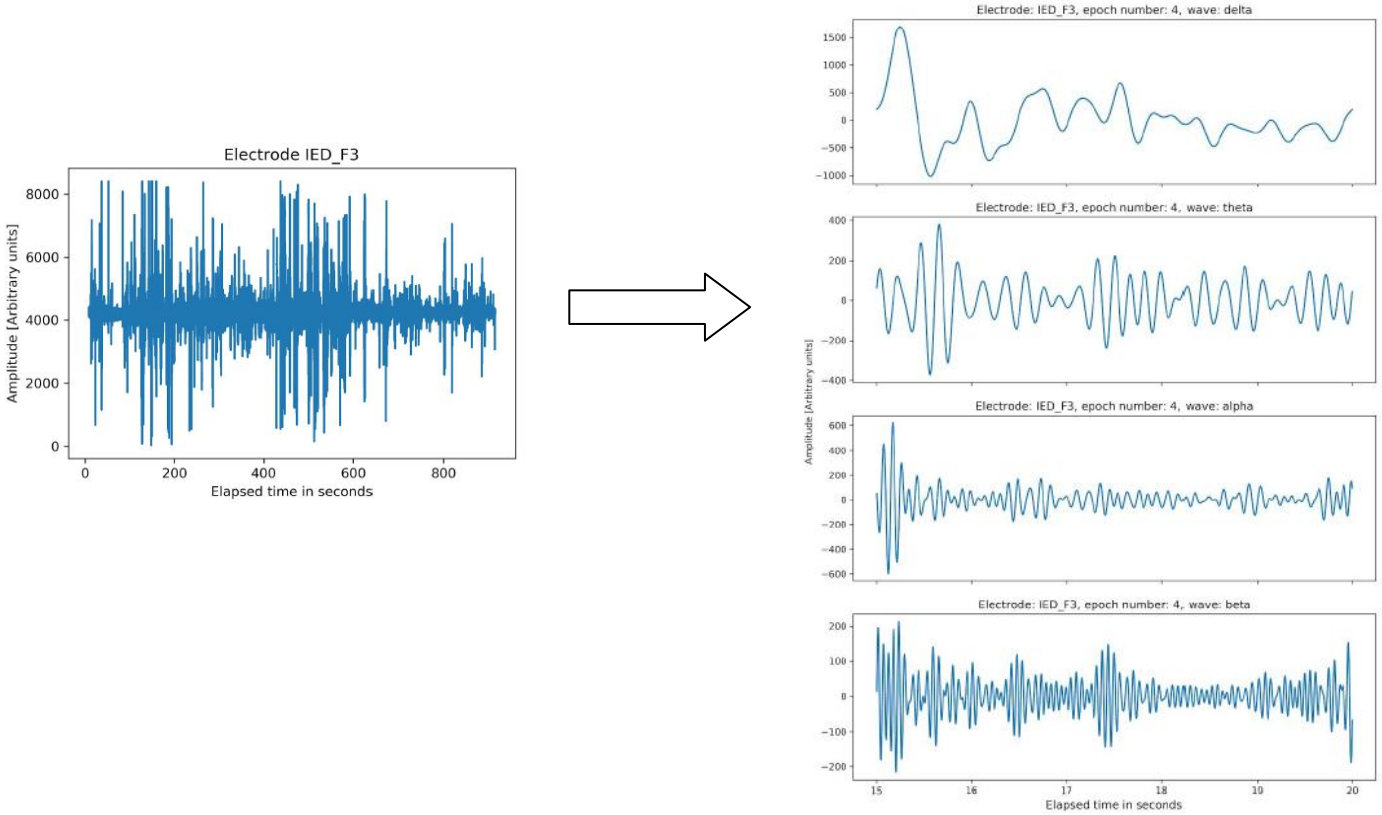
- Tymczasowe posortowanie danych
- Zliczenie częstości występowania wybranych danych
- Analiza graficzna danych (np. histogramy, mapy)





- Zignorowanie, uzupełnienie lub usunięcie błędnych danych?
  - Czasem błędne dane można zignorować, jeżeli oczywistym jest że nie wpłyną na wynik uczenia
  - Jeżeli błędnych danych jest mało (<5%), zwykle można je usunąć bez szkody dla procesu uczenia
  - Błędne dane można poprawić wykorzystując np. medianę, średnią z pozostałych danych, czasem wstawić zero
  - NULL vs EMPTY vs 0

- Czasem dane wymagają wstępnej obróbki przed przystąpieniem do uczenia maszynowego



- W danych mogą występować zakłócenia, związane z daną dziedziną bądź środowiskiem pomiarowym
  - okresowe
  - przypadkowe

  Dane w takim przypadku należy odfiltrować przy wykorzystaniu metod matematycznych / statystycznych

Do zapamiętania: bardzo rzadko dane są wystarczającej jakości, żeby dało się je wykorzystać bezpośredniego; zwykle należy je oczyścić i wstępnie przetworzyć.

Python

- Zaprojektowany w 1991 roku jako język ogólnego przeznaczenia, pierwsza stabilna wersja została wydana w 1994 r., wersja (gałąź) 2 w r. 2000 a wersja (gałąź) 3 – w 2008.
- Interpretowany, obiektowy.
- Stopniowo podbił społeczność naukową i wyrósł na dojrzały ekosystem specjalistycznych pakietów do przetwarzania i analizy danych.
- Pozwala na eksperymenty i szybką, łatwą implementację teorii i szybkie wdrożenia aplikacji naukowych.
- Inne zastosowania: strony WWW (framework Django, CMS Plone), systemy automatyki domowej (Home Assistant), mikrokontrolery (MicroPython), skrypty do administracji systemami komputerowymi i wiele innych.

Zalety

- Jest bardzo uniwersalny. Można programować w różnych stylach (obiektowo lub proceduralnie), niezależnie od poziomu umiejętności.
- Jest wieloplatformowy – działa płynnie na systemach operacyjnych Windows, Linux i Mac.
- Chociaż interpretowany, jest niewątpliwie szybki w porównaniu do innych języków analizy danych takich jak R czy MATLAB (choć oczywiście sporo wolniejszy od C czy Javy). Wersja 3.11 z 2022 r. stanowi duży skok wydajnościowy.
- Są też szybkie implementacje, porównywalne z językami kompilowanymi – np. darmowy Intel Python.
- Można też pisać w języku Cython, który jest nadzbiorem Pythona i zapewnia wydajność jak w języku C.
- Może pracować na dużych danych umieszczonych w pamięci, ze względu na jej minimalne zużycie i doskonałe nią zarządzanie. Posiada efektywny odśmiecacz (ang. *garbage collector*).

- Jest dość prosty do nauczenia i używania.
- Bez problemu przetwarza bardzo duże liczby oraz liczby w systemie dwójkowym, ósemkowym, szesnastkowym.
- Łatwo przetwarza nowoczesne języki formalne do reprezentowania danych (np. YAML).
- Ma bogatą bibliotekę standardową i wiele dodatkowych modułów, z różnych dziedzin nauki i techniki.
- Istnieją specjalizowane dystrybucje (czyli sam język + wygodny edytor + gotowe do wykorzystania biblioteki) do analizy i przetwarzania danych, np. Anaconda.

Wady

- Dwie gałęzie, 2.x, 3.x, niekompatybilne ze sobą (gałąź 2.x nie jest już wspierana).
- Częsta niekompatybilność pakietów między sobą.
- Oznaczanie bloków programu za pomocą wcięć może być uciążliwe przy większych programach.
- Niekompatybilność tabulatorów i spacji przy wcięciach.
- Czasem (ale mimo wszystko wyjątkowo) niespójna składnia.

Instalacja, zarządzanie

- Podstawowa instalacja: https://www.python.org/ albo kompletna platforma Anaconda: https://www.anaconda.com/
- Program zarządzający: `pip` (w Anacondzie: `conda`), np. `python -m pip install tensorflow`
- Edytory / IDE : Idle (domyślny), Spyder (https://www.spyder-ide.org/), PyCharm, Jupyter (https://jupyter.org/)