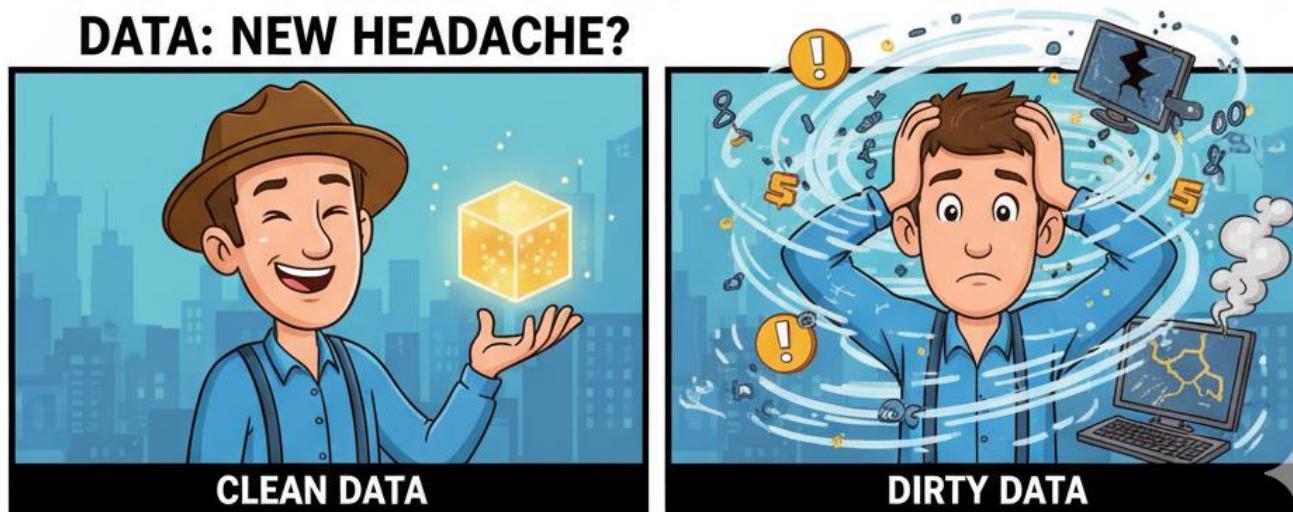
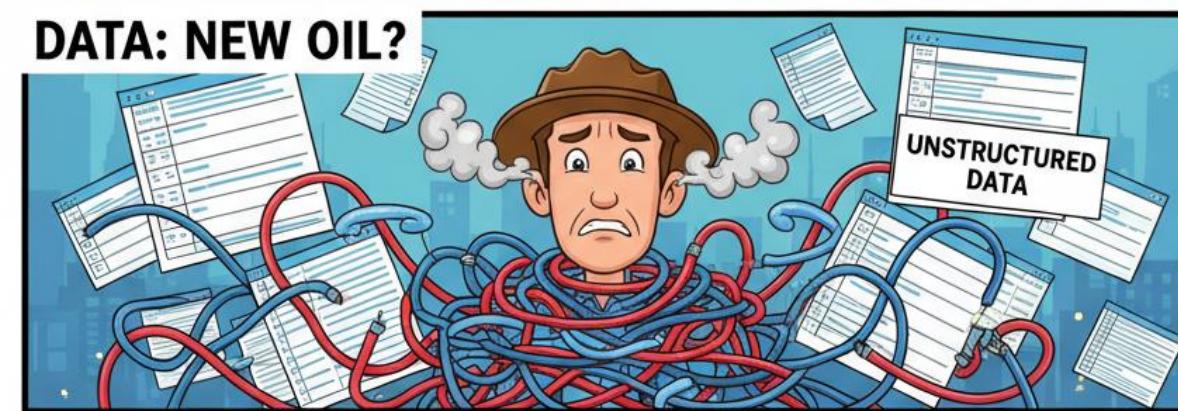


Krajobraz analizy danych

Paweł Gliwny

Nasz przedmiot: Eksploracja danych

- Dane: nowa ropa naftowa czy problem?
- Źródło: *Gemini*



Czym są dane?

„Jeśli mamy dane, przyjrzyjmy się danym.

Jeśli mamy tylko opinie, postępujmy według moich”.

— Jim Barksdale, były CEO firmy Netscape

Słownictwo nauki o danych (ang. *data science*)

- Ramki danych
- Obserwacje, rekordy, próby
- Cechy, pola, atrybuty, predyktory, zmienne
- Dane ustrukturyzowane i nieustrukturyzowane
- Statystki zbiorcze
- ...

Dane, a informacje

- Informacje to wiedza pochodna czerpana z **mierzenia procesu**, myślenia o czymś nowym, debatowania.
- Od czujników w satelitach do neuronów aktywujących się w naszych mózgach nieustannie powstają **nowe informacje**.
- Komunikowanie i rejestrowanie tych informacji nie (zawsze) jest jednak proste.
- Staramy się przekazywać wiedzę i zapisywać to, czego się nauczyliśmy. Kiedy to robimy tworzymy dane.
- ***Dane są zatem zakodowanymi informacjami.***

Przykładowy zbiór danych

- Tabela danych jest nazywana zbiorem/ramką danych
- Każdy **wiersz** tabeli jest zmierzonaą instancją powiązanych informacji.
- Każda **kolumna** tabeli jest listą informacji, zakodowanych w jednakowym formacie

Data	Wydatki na reklamy	Sprzedaż w sztukach	Zysk	Medium
2021-01-01	2000	100	10 425	Prasa
2021-02-01	1000	150	15 349	Internet
2021-03-01	3000	200	25 095	Telewizja
2021-04-01	1000	175	12 433	Internet

Nazewnictwo w tabeli (ramce danych)

- Wiersze tabeli często nazywa się **obserwacjami, rekordami, krotkami lub próbami**.
- Kolumny zbiorów danych często określa się mianem **cech, pól, atrybutów, predyktorów lub zmiennych**.
- Punkt danych jest przecięciem obserwacji i cechy.
 - Na przykład 150 sztuk produktu sprzedanych dnia 2021-02-01 jest punktem danych

Znaj swoich rozmówców

- Dane studiuje się w wielu różnych dziedzinach, które mają własne żargony.
- Te same rzeczy często mają różne nazwy.
- Częścią bycia specem od danych jest umiejętność rozmawiania z **różnymi grupami** z uwzględnieniem ich preferencji.



Typy danych

- Dwa najczęstsze typy danych opisuje się jako **liczbowe** lub **kategoryczne**.
 - **Dane liczbowe** składają się głównie z liczb, ale mogą używać dodatkowych symboli do identyfikowania jednostek.
 - **Dane kategoryczne** składają się ze słów, symboli, fraz i (dezorientująco) czasem również liczb, jak w przypadku kodów pocztowych.
- Dane liczbowe i kategoryczne dzielą się na dalsze *podkategorie*.

Typy danych **liczbowych**

- **Dane ciągłe**, które mogą przyjmować dowolną wartość na osi liczbowej.
 - Reprezentują one fundamentalnie niepoliczalny zbiór wartości.
 - Temperatura powietrza, zmierzona i przekształcona w dane, reprezentowałaby zmienną ciągłą.
- **Dane dyskretne**, w przeciwieństwie do danych ciągłych, mają precyzję ograniczoną do wartości całkowitych.
 - Na przykład liczba samochodów, które posiadasz, może wynosić 0, 1, 2 lub więcej, ale nie 1,23.
 - Odzwierciedla to rzeczywistą naturę mierzonej wielkości

Typy danych kategorycznych

Dane uporządkowane (lub porządkowe) to dane kategoryczne z pewnym wewnętrznym porządkiem.

- Ankiety wykorzystują dane porządkowe, gdy proszą o ocenę wrażeń w skali od 1 do 10.
- Porządkowe dane kategoryczne nie muszą być zakodowane jako liczby.
- Na przykład rozmiary koszulek: mały, średni, duży, bardzo duży.

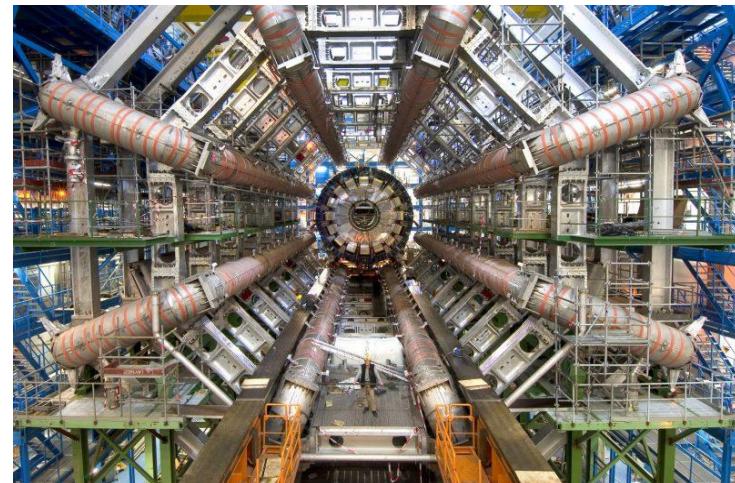
Typy danych kategorycznych

Dane nieuporządkowane (lub nominalne) nie mają żadnego wewnętrznego porządku.

- Cecha „Medium” o wartościach „Prasa”, „Internet”, „Telewizja”
- Odpowiedzi „Tak” lub „Nie” o przynależności do fanów Widzewa lub ŁKS.
- Kolejność ich prezentacji jest zawsze dowolna — nie można powiedzieć, że jedna kategoria jest „większa” od innej.

Jak gromadzi się dane i jaką mają strukturę ?

- Dane można też szerzej klasyfikować na podstawie tego, jak je zebrano i jaką mają strukturę.
- Dane można opisać jako **obserwacyjne** lub **eksperymentalne**, w zależności od tego, jak je zebrano



Dane obserwacyjne, a eksperymentalne

- **Dane obserwacyjne** gromadzi się zgodnie z tym, co widzą lub słyszą jakieś osoby lub komputery *pasywnie obserwujące jakiś proces* (zautomatyzowane systemy, takie jak sensory czy oprogramowanie monitorujące).
- **Dane eksperymentalne** gromadzi się zgodnie z metodą naukową z wykorzystaniem przepisanej metodyki.

Dane obserwacyjne

Większość danych w firmach (i na świecie) ma charakter obserwacyjny.

- Przykłady: liczba wizyt na stronie, dzienna sprzedaż, liczba e-maili
- Mogą być **zapisywane celowo lub przypadkowo.**
- Powstają jako produkt uboczny (transakcje, płatności, social media)
- Często przechowywane w bazach danych „na później”
- **Zbierane także świadomie** (np. ankiety, sondaże)

Dane eksperymentalne

- Gromadzi się je celowo i metodycznie w celu odpowiedzi na konkretne pytania.
- „Złoty standard” danych dla statystyków i badaczy.
- Stosuje się procedurę do losowo dobranych osób lub rzeczy.
- Przykład: *badania kliniczne lekarstw lub testy A/B*.



Metodyka eksperymentów A/B

- Stosowana w wielu branżach, np. badania leków, marketing.
- W marketingu cyfrowym:
 - Projektanci witryn eksperymentują z układami i reklamami.
 - „Rzut monetą” decyduje o wyświetleniu reklamy A lub B.
 - Po odwiedzinach tysięcy użytkowników analizuje się klikalność.
 - **Randomizacja** zapewnia równowagę innych cech (pora dnia, typ internauty).

Zastosowanie i korzyści eksperymentów A/B

Zastosowanie:

- Testy A/B pozwalają określić, która opcja generuje więcej **kliknięć**.
- Przykład: reklamy A i B wyświetlane losowo, wyniki analizowane po zebraniu danych.

Korzyści:

- **Obiektywna ocena skuteczności** różnych wersji (np. stron internetowych, reklam).
- Minimalizacja wpływu zmiennych zakłócających dzięki randomizacji.

Dane ustrukturyzowane i nieustrukturyzowane

- Dane **ustrukturyzowane** wyglądają tak jak dane w arkuszach kalkulacyjnych, bazach danych
 - Są uporządkowane i mają strukturę w postaci wierszy oraz kolumn.
- Dane **nieustrukturyzowane** to na przykład tekst z recenzji na Amazonie, zdjęcia na Facebooku, filmy na YouTubie albo pliki dźwiękowe.
 - Wymagają pomysłowych technik w celu przekształcenia ich w dane ustrukturyzowane na potrzeby metod analitycznych

Dane ustrukturyzowane (ang. Structured Data)

Informacje, które mają **ściśle zdefiniowany**, z góry narzucony **model danych**.

- Struktura danych (nazwy kolumn, typy danych, relacje) jest określona *przed* ich wprowadzeniem.
- Dzięki regularnej budowie można je **łatwo przeszukiwać, sortować i filtrować** przy użyciu standardowych zapytań, np. SQL.
- **Główne dane ilościowe:** często składają się z liczb, dat, czy krótkich, zdefiniowanych ciągów tekstowych.
- Zazwyczaj przechowuje się je w **relacyjnych bazach danych** (np. MySQL) lub hurtowniach danych.

Dane nieustrukturyzowane (ang. *Unstructured Data*)

- To wszystkie informacje, które **nie posiadają z góry narzuconego schematu** i istnieją w swojej "surowej", natywnej formie.
- Stanowią one **zdecydowaną większość** (szacuje się, że 80-90%) danych generowanych na świecie.
- Ich analiza jest znacznie trudniejsza, ponieważ najpierw trzeba nadać im jakąś strukturę lub wyciągnąć z nich istotne cechy.

Cechy dane nieustrukturyzowane

- **Brak schematu:** nie mają predefiniowanej struktury tabelarycznej.
- **Trudność w przeszukiwaniu:** standardowe narzędzia bazodanowe sobie z nimi nie radzą. Wyszukiwanie wymaga zaawansowanych algorytmów.
- **Główne dane jakościowe:** dominują tekst, obraz, dźwięk i wideo.
- **Przechowywanie:** Używa się do tego **nierelacyjnych baz danych** (NoSQL), **jezior danych** (Data Lakes) czy systemów plików (np. Amazon S3, HDFS)

Dane nieustrukturyzowane: tekst

- **Tekst:** E-maile, posty w mediach społecznościowych, recenzje produktów, artykuły, dokumenty PDF.
- **Analiza:** Przetwarzanie języka naturalnego (**NLP**), np. do analizy sentymetu (czy recenzja jest pozytywna?), ekstrakcji słów kluczowych czy klasyfikacji tematów.

```
# pip install -U spacy
# python -m spacy download en_core_web_sm
import spacy

# Load English tokenizer, tagger, parser and NER
nlp = spacy.load("en_core_web_sm")

# Process whole documents
text = ("When Sebastian Thrun started working on self-driving cars at "
        "Google in 2007, few people outside of the company took him "
        "seriously. "I can tell you very senior CEOs of major American "
        "car companies would shake my hand and turn away because I wasn't "
        "worth talking to," said Thrun, in an interview with Recode earlier "
        "this week.")
doc = nlp(text)

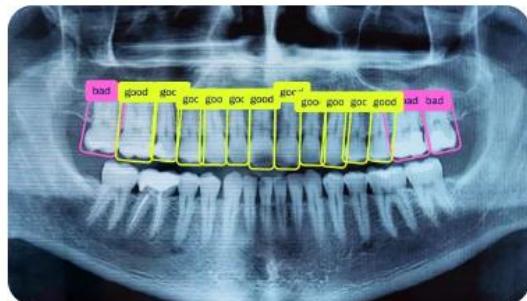
# Analyze syntax
print("Noun phrases:", [chunk.text for chunk in doc.noun_chunks])
print("Verbs:", [token.lemma_ for token in doc if token.pos_ == "VERB"])

# Find named entities, phrases and concepts
for entity in doc.ents:
    print(entity.text, entity.label_)
```

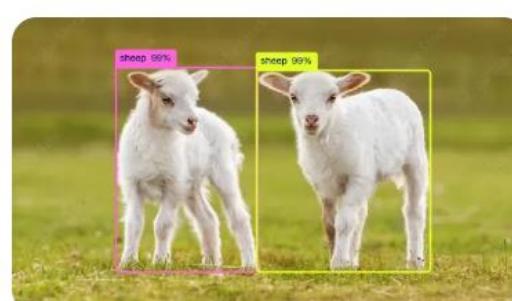
Dane niestrukturyzowane: obrazy

- **Obrazy:** zdjęcia na Instagramie, skany dokumentów, zdjęcia satelitarne, obrazy z monitoringu.
 - **Analiza:** widzenie komputerowe (**Computer Vision**), np. do rozpoznawania obiektów na zdjęciach, identyfikacji twarzy czy analizy obrazów medycznych.

Ultralytics YOLO applications in agriculture



OBB object detection



Object detection



Instance segmentation



Image classification

Dane nieustrukturyzowane: wideo i dźwięk

Wideo: Filmy na YouTube, nagrania z wideokonferencji, monitoring miejski.

Analiza: Podobnie jak przy obrazach, ale z uwzględnieniem wymiaru czasu, np. do śledzenia ruchu obiektów.

Dźwięk: Pliki MP3, nagrania rozmów w call center, podcasty.

Analiza: Przetwarzanie sygnałów audio, np. do transkrypcji mowy na tekst (Speech-to-Text) czy identyfikacji mówcy.

A co pomiędzy? Dane Semistrukturalne (ang. *Semi-structured Data*)

- **Trzecia** kategoria
- Dane semistrukturalne nie pasują do sztywnego schematu relacyjnej bazy danych
 - zawierają **znaczniki lub metadane**, które pozwalają na pewną organizację i hierarchizację.
- Pliki **JSON** i **XML**.
 - Mają strukturę opartą na znacznikach i zagnieżdżeniach
 - Elastyczne, ale jednocześnie łatwiejszymi do maszynowego przetworzenia niż czysty tekst.

```
{  
  "uzytkownik": {  
    "imie": "Anna",  
    "nazwisko": "Kowalska",  
    "email": anna.k@example.com,  
    "zainteresowania": ["analiza danych", "AI",  
    "rower"]  
  }  
}
```

Podsumowanie typy danych

Cecha	Dane Ustrukturyzowane	Dane Nieustrukturyzowane	Dane Semistrukturalne
Struktura	Z góry narzucony schemat	Brak schematu	Elastyczny schemat (znaczniki)
Model	Relacyjny (tabele, wiersze)	Nierelacyjny (natywna forma)	Hierarchiczny (drzewo, graf)
Przykłady	Arkusze Excel, bazy SQL	Pliki tekstowe, wideo, audio	Pliki JSON, XML
Przechowywanie	Relacyjne bazy danych (SQL)	Bazy NoSQL, jeziora danych	Bazy NoSQL (np. dokumentowe)
Analiza	Łatwa, głównie SQL	Złożona (NLP, CV), wymaga AI/ML	Umiarkowanie złożona

Dlaczego to jest ważne?

- **Narzędzia i techniki analityczne, których użyjesz, zależą całkowicie od rodzaju danych.**
 - Nie da się zastosować zapytania SQL do analizy tysięcy recenzji na Amazonie.
 - Nie ma sensu używać zaawansowanych modeli NLP do analizy danych sprzedażowych w tabeli

Współczesna analytyka danych

- Polega w dużej mierze na umiejętności pracy z każdym rodzajem danych,
- Szczególnie na przekształcaniu danych nieustrukturyzowanych i semistrukturalnych w formę **ustrukturyzowaną**, na której **mogą działać algorytmy uczenia maszynowego**.

Gigantyczne zbiory danych we współczesnym świecie

YouTube:

- Szacuje się, że na platformę trafia ponad **500 godzin nowego materiału wideo na minutę**.
- Całkowita objętość danych przechowywanych przez YouTube jest trudna do dokładnego oszacowania, ale z pewnością liczy się w setkach eksabajtów (EB).
- źródło: [wiki/YouTube](#)

Trening dużych modeli językowych (LLM)

- Modele takie jak GPT są trenowane na niewyobrażalnie wielkich zbiorach danych.
- Zbiór danych open-source "**The Pile**" ma rozmiar ponad **800 GB**.
- Szacuje się, że komercyjne modele, takie jak GPT-4, były trenowane na zbiorach danych o łącznej wielkości sięgającej **dziesiątek terabajtów (TB)**.
- Źródła: [EleutherAI - The Pile](#)

Słynny cytat byłego CEO Google w nowym kontekście

- Słowa Erica Schmidta z 2010 roku, że "*co dwa dni tworzymy tyle informacji (5 eksabajtów), ile stworzyliśmy od zarania cywilizacji do 2003 roku*", **doskonale oddają tempo wzrostu.**
- Dziś, w erze AI, IoT i mediów społecznościowych, to tempo jest jeszcze bardziej zawrotnie
- Podobną ilość danych generujemy prawdopodobnie w ciągu **kilku godzin.**



Dane w eksperymenty naukowych

Obserwatorium Very C. Rubin (Legacy Survey of Space and Time - LSST)

- Ten rewolucyjny teleskop będzie generował około **20 TB danych surowych każdej nocy**.
- W ciągu dziesięciu lat trwania projektu da to katalog danych o wielkości około **60 PB**, a pełne przetworzone archiwum osiągnie **setki petabajtów**.
- Źródło: [Rubin Observatory - Data Management](#)

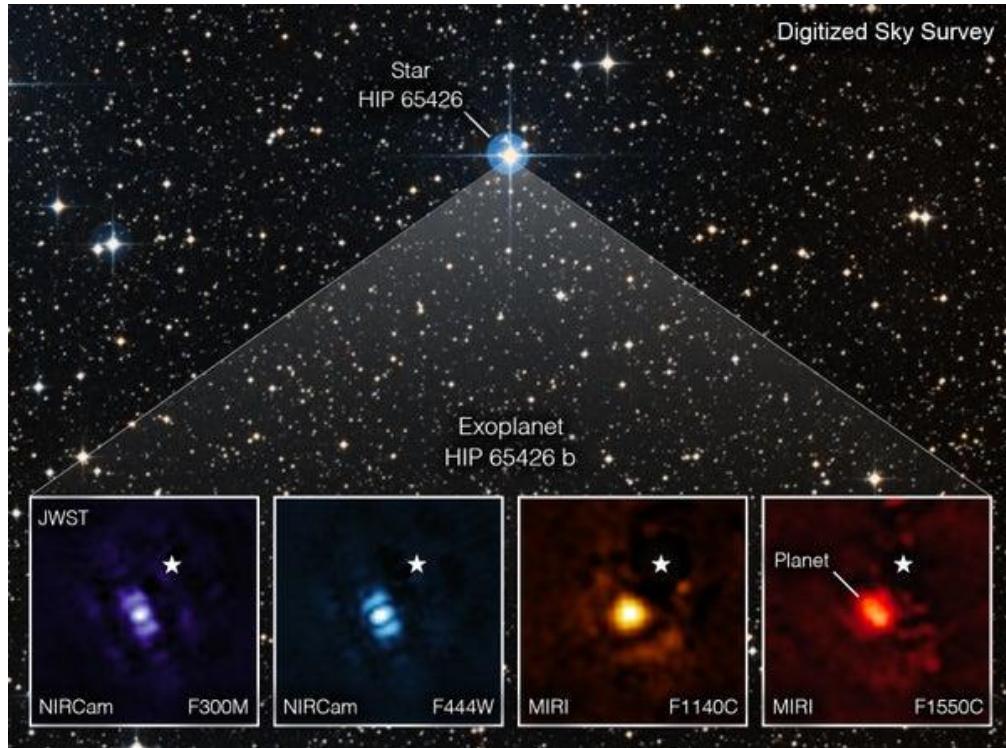


Dane w eksperymenty naukowych

Kosmiczny Teleskop Jamesa Webba (JWST)



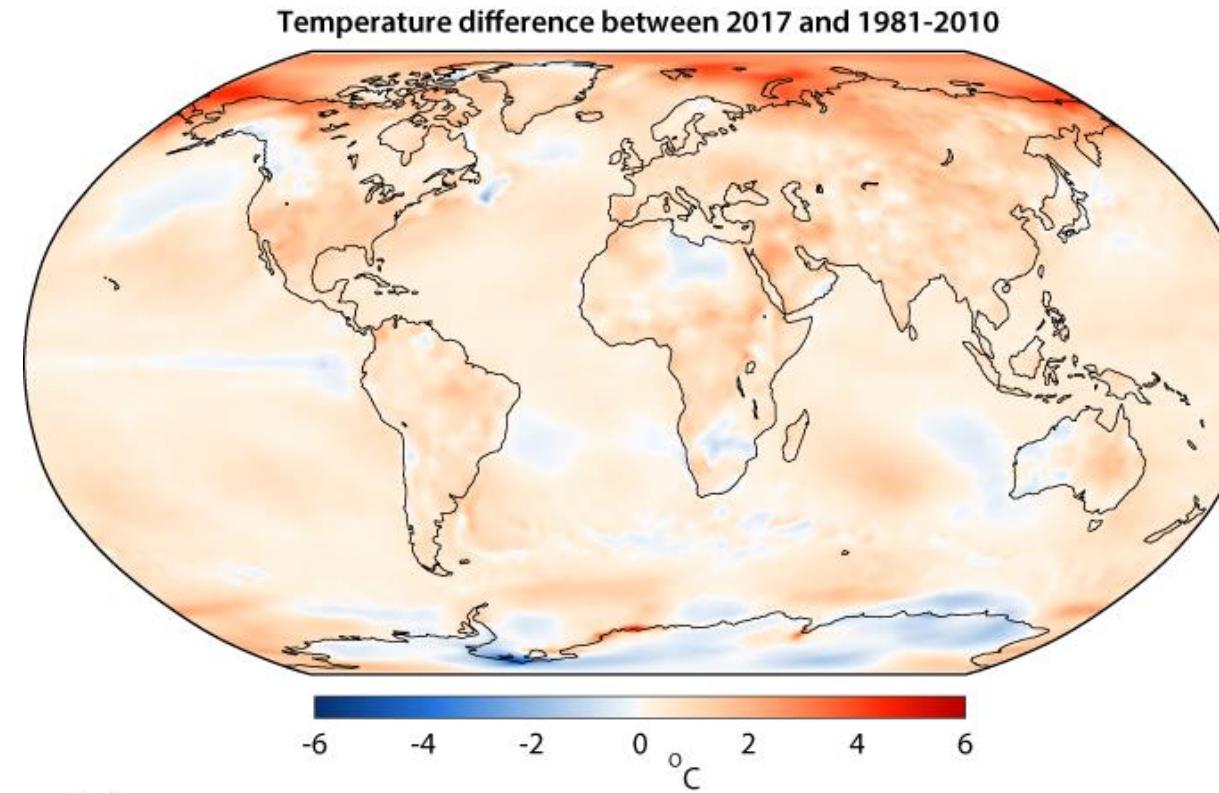
- Webb przesyła na Ziemię około **57 GB danych dziennie**, dostarczając niezwykle złożonych i cennych informacji, które rewolucjonizują astronomię.
- Źródło: [DPReview - JWST SSD](#)



[jwsts-first-exoplanet-images-forecast-a-bright-future](#)

Dane Klimatyczne

- Nowoczesne centra danych klimatycznych przechowują setki petabajtów informacji.
- Na przykład, zbiór danych **ERA5**, będący globalną reanalizą klimatu, ma rozmiar około **5 PB**
- Jest zarządzany przez *Europejskie Centrum Prognoz Średnioterminowych* w ramach programu *Copernicus*.



[ecmwf-era5-reanalysis-soon-extend-back-1979](#)

Dane z meczu piłki nożnej

- **Respo Vision:**
 - Polska firma z branży **deep tech**, która zmienia analitykę sportową za pomocą AI.
- **Przełomowa technologia:** Generuje precyzyjne, trójwymiarowe (3D) dane o ruchu zawodników i piłki na podstawie **nagrania z tylko jednej, standardowej kamery**.
- Ogromna skala danych: System zbiera aż **150 milionów punktów danych** z jednego meczu
- Źródło: meet-the-startups-respo-vision



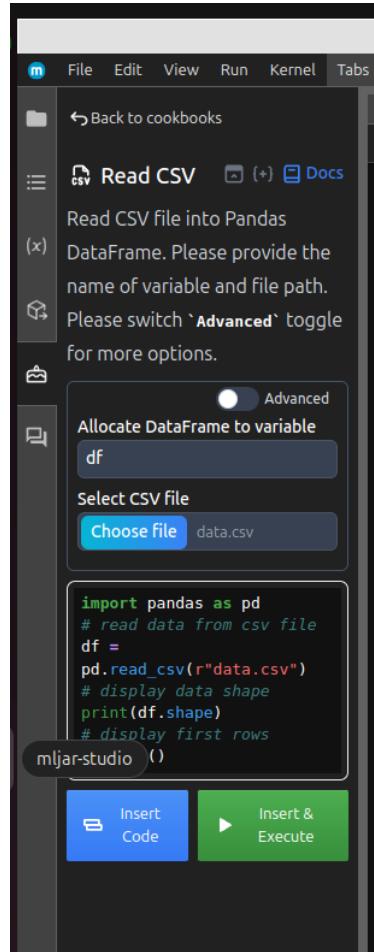
Narzędzia

- Językiem nauki o danych jest python, ..., R, js.
- Dashboardy
- Notebooki



MLJAR Studio

- Polski produkt
- *MLJAR Studio is a desktop app that will help you analyze your data. No coding experience required.*
- Link: <https://mljar.com/>



The screenshot shows the MLJAR Studio interface. On the left, there's a sidebar with a 'Read CSV' tool, which prompts the user to provide the name of a variable and a file path. It also includes an 'Advanced' toggle, a 'Select CSV file' button, and a 'Choose file' button. Below this is a code editor window containing Python code for reading a CSV file into a Pandas DataFrame. The code includes imports for numpy and pandas, reads the CSV file, prints its shape, and displays the first few rows. At the bottom of the code editor are two buttons: 'Insert Code' and 'Insert & Execute'. The main workspace on the right contains several tabs: 'pilka_nozna.ipynb', 'Untitled.ipynb', 'wstep_do_pandas.ipynb' (which is currently active), and 'wstep_do_wizualizacji.ipynb'. The active tab, 'wstep_do_pandas.ipynb', has a title 'Wstęp do pandas' and a section titled 'Series'. It contains text about the pandas library and its relationship to NumPy, followed by a code cell demonstrating the creation of a Series object from a list of values. The output of this cell shows the resulting Series object with index 0 and value 4.

```
File Edit View Run Kernel Tabs Settings Help  
pilka_nozna.ipynb Untitled.ipynb wstep_do_pandas.ipynb wstep_do_wizualizacji.ipynb  
Wstęp do pandas  
Biblioteka pandas zawiera struktury danych i narzędzia do manipulacji danymi, zaprojektowane tak, aby:

- Jest często używana w połączeniu z narzędziami do obliczeń numerycznych, takimi jak NumPy i SciPy, np. matplotlib.
- pandas adaptuje znaczącą część stylu NumPy w zakresie obliczeń na tablicach, w szczególności funkcji.
- Największą różnicą w stosunku do NumPy jest to, że pandas został zaprojektowany do pracy z danymi jednorodnymi (homogenicznymi) danymi numerycznymi w formie tablic.

[2]:  
import numpy as np  
import pandas as pd  
  
from pandas import Series, DataFrame  
  
Wprowadzenie do struktur danych pandas  
Series  
Obiekt Series to jednowymiarowy, podobny do tablicy obiekt, zawierający sekwencję wartości (o typach)  
[3]:  
obj = pd.Series([4, 7, -5, 3])  
obj  
[3]: 0    4  
      1    7  
      2   -5  
      3    3
```

Różne podejścia do analizy danych

Statystyka opisowa i wnioskowanie

Statystyka

- Dostarcza podsumowania zbioru danych za pomocą metryk takich jak średnia, mediana i odchylenie standardowe.
- Jest to pierwszy krok do zrozumienia, co dane pokazują i gdzie może być potrzebna głębsza analiza.

Wnioskowanie statystyczne

- Pozwala na **wyciąganie wniosków wykraczających poza bezpośredni zbiór danych.**
- Używając próbek do generalizacji na temat większych populacji, narzędzia takie jak przedziały ufności i testy hipotez kwantyfikują niepewność.

Prawdopodobieństwo

- Stanowi podstawę zarówno wnioskowania statystycznego, jak i wielu modeli ilościowych.
- Umożliwia kwantyfikację niepewności i ocenę prawdopodobieństwa przyszłych wyników na podstawie bieżących informacji.

Optymalizacja i symulacje

Optymalizacja: Koncentruje się na znalezieniu najlepszego możliwego rozwiązania przy określonych ograniczeniach.

- Przykładem jest programowanie liniowe, często używane do alokacji zasobów lub maksymalizacji wydajności.

Symulacja: Metody takie jak symulacje Monte Carlo modelują złożone lub niepewne systemy poprzez generowanie tysięcy potencjalnych wyników.

- Technika ta jest szeroko stosowana w finansach, operacjach i analizie ryzyka.

Tym się nie będziemy zajmować

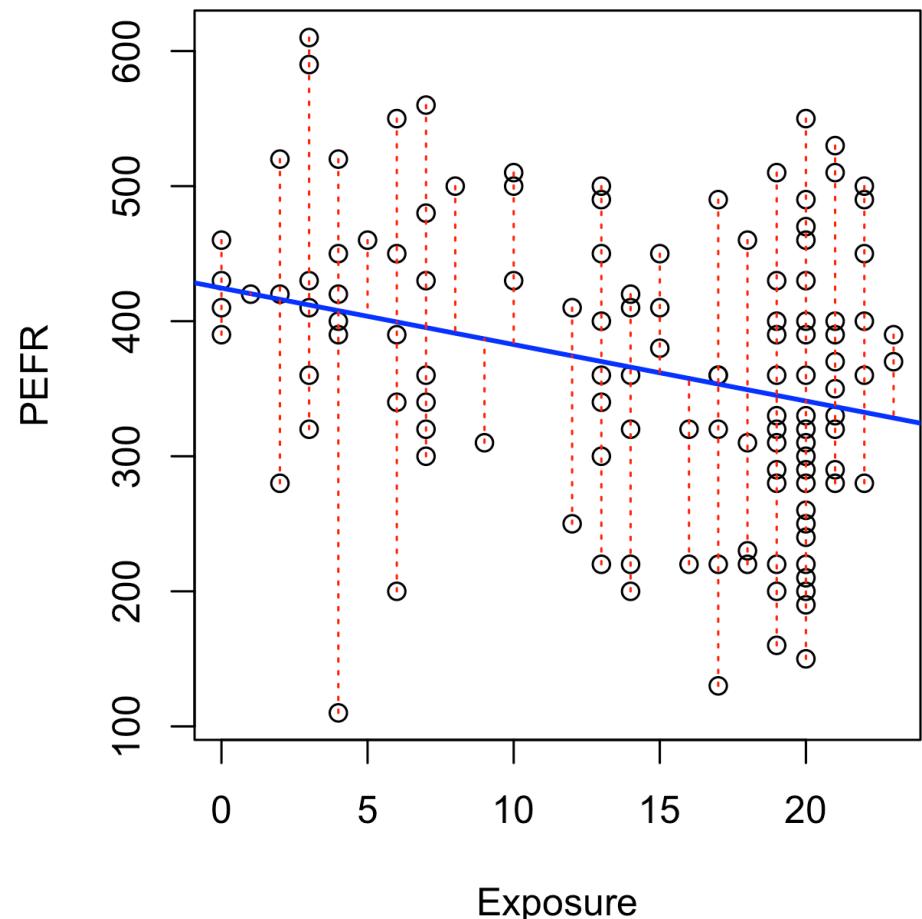
Nowoczesne podejścia: uczenie maszynowe (ang. *Machine learning (ML)*)

- Opiera się na wielu zasadach statystycznych.
- Wykorzystując algorytmy, które uczą się na podstawie danych, aby dokonywać coraz dokładniejszych **prognoz** lub **klasyfikacji**.
- ML wykracza poza tradycyjne modele statystyczne,
 - Ale ma te same korzenie i opiera się na kluczowych ideach, takich jak **prawdopodobieństwo** i **optymalizacja**.

Analiza zależności

Analiza regresji:

- Bada relacje między zmiennymi – jak jedna zmienna zmienia się w odpowiedzi na drugą.
- Podstawowe narzędzie do modelowania trendów i prognozowania przyszłych wyników.



Techniki uczenia maszynowego

- Obejmuje znane narzędzia, jak regresja liniowa
- Rozszerza je o elastyczne metody modelowania, takie jak **drzewa decyzyjne i lasy losowe**, które sprawdzają się w złożonych środowiskach.
- Źródło: [scikit-learn](#)

scikit-learn
Machine Learning in Python

Getting Started | Release Highlights for 1.7

- Simple and efficient tools for predictive data analysis
- Accessible to everybody, and reusable in various contexts
- Built on NumPy, SciPy, and matplotlib
- Open source, commercially usable - BSD license

Classification

Identifying which category an object belongs to.

Applications: Spam detection, image recognition.

Algorithms: Gradient boosting, nearest neighbors, random forest, logistic regression, and more...

Examples

Regression

Predicting a continuous-valued attribute associated with an object.

Applications: Drug response, stock prices.

Algorithms: Gradient boosting, nearest neighbors, random forest, ridge, and more...

Examples

Clustering

Automatic grouping of similar objects into sets.

Applications: Customer segmentation, grouping experiment outcomes.

Algorithms: k-Means, HDBSCAN, hierarchical clustering, and more...

Examples

Dimensionality reduction

Reducing the number of random variables to consider.

Applications: Visualization, increased efficiency.

Algorithms: PCA, feature selection, non-negative matrix factorization, and more...

Examples

Model selection

Comparing, validating and choosing parameters and models.

Applications: Improved accuracy via parameter tuning.

Algorithms: Grid search, cross validation, metrics, and more...

Examples

Preprocessing

Feature extraction and normalization.

Applications: Transforming input data such as text for use with machine learning algorithms.

Algorithms: Preprocessing, feature extraction, and more...

Examples

Techniki uczenia głębokiego

- Opiera się na architekturach sieci neuronowych
- Rozwija koncepcje znane z klasycznego uczenia maszynowego
- Wprowadzając **wielowarstwowe modele** (np. sieci konwolucyjne i rekurencyjne)
- Zdolne do **automatycznego wykrywania złożonych wzorców** w danych, takich jak obrazy czy tekst.

The screenshot shows the official Keras website. At the top, there's a navigation bar with links for 'GET STARTED', 'GUIDES', 'API', 'EXAMPLES', 'KERAS TUNER', 'KERAS RS', and 'KERAS HUB'. A search bar is located on the far right. The main header features a large red 'K' logo followed by the word 'Keras'. Below the header, a prominent red banner with white text reads 'A superpower for ML developers'. A subtext below the banner states: 'Keras is a deep learning API designed for human beings, not machines. Keras focuses on debugging speed, code elegance & conciseness, maintainability, and deployability. When you choose Keras, your codebase is smaller, more readable, easier to iterate on.' Three buttons labeled 'API DOCS', 'GUIDES', and 'EXAMPLES' are visible. To the right of the banner is a large graphic element consisting of overlapping red and white triangles forming a stylized 'X' shape. At the bottom of the page, there's a code snippet in Python showing a neural network architecture, followed by logos for JAX, TensorFlow, PyTorch, OpenVINO, and Intel, and a statement about multi-framework support.

```
inputs = keras.Input(shape=(32, 32, 3))
x = layers.Conv2D(32, 3, activation="relu")(inputs)
x = layers.Conv2D(64, 3, activation="relu")(x)
residual = x = layers.MaxPooling2D(3)(x)

x = layers.Conv2D(64, 3, padding="same")(x)
x = layers.Activation("relu")(x)
x = layers.Conv2D(64, 3, padding="same")(x)
x = layers.Activation("relu")(x)
x = x + residual

x = layers.Conv2D(64, 3, activation="relu")(x)
x = layers.GlobalAveragePooling2D()(x)
outputs = layers.Dense(10, activation="softmax")(x)
```

Welcome to multi-framework machine learning

With its multi-backend approach, Keras gives you the freedom to work with JAX, TensorFlow, and PyTorch. Build models that can move seamlessly across these frameworks and leverage the strengths of each ecosystem.

Zaliczenie

- Z **teorii** egzamin pisemny
- **Ćwiczenia**, z każdych zajęć zrobione zadania w postaci **notebook** wrzucamy **na swój github**
 - Wszystkie zadania na github: **3**
 - Na ocenę **4 i 5** będzie do zrobienia mini projekt z zbudowania dashboardu

Źródła

- **Książka: *Analityk danych. Przewodnik po data science, statystyce i uczeniu maszynowym*** by Alex J. Gutman, Jordan Goldmeier