

## 1. Przedstaw sposoby agregacji danych.

### Wersja 1.

Agregacja danych polega na łączeniu danych z różnych źródeł w jedną całość. W efekcie agregacji powstaje nowa baza danych, która po spełnieniu przesłanek wskazanych w ustawie o ochronie baz danych może być również bazą danych w prawnym tego słowa znaczeniu. Sama agregacja danych odbywa się z wykorzystaniem odpowiedniego oprogramowania, które pobiera dane z ich źródła, a następnie scalą w jedną bazę danych. Agregacja danych polega na wyliczeniu jednej lub wielu statystyk, takich jak średnia arytmetyczna (najczęstszy przypadek), minimum, maksimum itp., dla grup obserwacji wyznaczonych przez kategorie zmiennych grupujących. W wyniku tej procedury powstaje nowa macierz danych, w której jedna obserwacja odpowiada jednej kategorii zmiennej grupującej, a wartości zmiennych są zastąpione wyliczonymi wartościami przyjętej statystyki dla poszczególnych grup.

Funkcje agregujące są również czasami nazywane funkcjami grupującymi!

- Funkcje agregujące mogą być użyte z dowolnymi prawidłowymi wyrażeniami np. COUNT(), MAX(), MIN(), z formatami liczbowymi jak i napisami czy datami
- Wartości NULL są ignorowane
- DISTINCT wyklucza powtarzające się wpisy
- Przykłady: AVG(x), COUNT(x), MIN(x), STDEV(x), SUM(x), VARIANCE(x)

### Używanie funkcji agregujących z grupami wierszy

Do funkcji agregującej można przesyłać bloki wierszy. Wykona ona obliczenia na grupie wierszy każdego bloku i wróci jedną wartość dla każdego bloku np. uzyskanie średniej ceny różnego typu produktów. Z klauzulą GROUP BY możemy używać dowolnej funkcji agregującej.

### Nieprawidłowe użycie funkcji agregującej

Jeżeli zapytanie zawiera funkcję agregującą i pobiera kolumny, które nie zostały w niej ujęte należy umieścić ją w klawiszach GROUP BY. Poza tym nie można używać funkcji agregujących do ograniczania wierszy za pomocą klawiszu WHERE.

#### Funkcje agregacji

- AVG(<exp>) — Średnia z wartością wyrażenia <exp>,
- COUNT(<exp>) — Liczba wierszy dla których <exp> nie jest NULL,
- COUNT(\*) — Liczba wszystkich wierszy,
- COUNT(DISTINCT <exp>) — Liczba wszystkich wierszy dla których wartość <exp> jest różna,
- MAX(<exp>) — Maksymalna wartość <exp>,
- MIN(<exp>) — Minimalna wartość <exp>,
- STDEV(<exp>) — Odchylenie standardowe dla <exp>,
- SUM(<exp>) — Suma wartości <exp>,
- VARIANCE(<exp>) — Wariancja dla <exp>.

### Wersja 2.

#### Funkcje agregujące

Agregacja to łączenie danych w jedną całość w wyniku czego powstaje nowa baza danych. To funkcje, które zwracają jedną wartość wyliczoną na podstawie wielu wierszy. Jest to sposób wyliczenia statystyk opisowych takich jak średnia arytmetyczna, min, max, odchylenie standardowe, suma, wariancja.

Jeśli użyjemy funkcji *distinct* to wyrażenie, które się powtarzają nie będą uwzględniane.

### **Group by**

Grupowanie polega na podzieleniu elementów zbioru w jeden zbiór, który ma wspólną cechę. Dokonuje się grupowania w celu użycia funkcji agregujących nie na całym zbiorze, ale na wybranych elementach zbioru, które mają wspólną cechę.

### **Klauzula having/where**

Klauzula where wykonuje się przed grupowaniem. Odpowiednikiem where jest klauzula having, która może być zastosowana po funkcji grupującej.

Przykład: *Select location\_id, count(\*) from departments where manager\_id is not null Group by location\_id having count(\*)>1*

### **Dodatek – przykłady**

## Funkcje grupowe - wprowadzenie

multi-row functions

avg, count, max, min, sum

zał: salary = 100, 200, 300, job\_id = ST\_CLERK

```
select avg(salary),max(salary),count(*),sum(salary),count(distinct job_id)
from employees;
```

200 300 3 600 1

```
select avg(salary)
from employees
where department_id=110;
```

## Funkcje grupowe – klauzula GROUP BY

```
select avg(salary), department_id
from employees
group by department_id;
```

Złożenia funkcji grupowych

```
select max(avg(salary))
from employees
group by department_id;
wartość największej średniej pensji w departamencie
```

## Funkcje grupowe – klauzula HAVING

```
select avg(salary), department_id
from employees
group by department_id
having count(*)>=3;
obliczenia dla departamentów zatrudniających >=3 osoby

where + nie f. grupowa
having + f.grupowa

select
from
where
group by
having
order by
```

## 2. Omów mechanizmy łączenia danych z wielu tabel.

### Joins

Join jest poleceniem, które łączy wiersze z dwóch lub więcej tabel lub widoków. Oracle Database wykonuje joina, kiedy tylko pojawią się kilka tabel w klauzuli FROM. SELECT

wybiera jakiekolwiek kolumny z tych tabel. Jeśli dwie z tabel mają kolumnę o takiej samej nazwie to należy zdefiniować, o której tabelę chodzi, aby uniknąć dwuznaczności.

### **Join Conditions**

Większość zapytań join ma przynajmniej jeden warunek, albo w klauzuli FROM albo w WHERE. Warunek JOIN porównuje dwie kolumny, każda z innej tabeli. Aby wykonać join, Oracle Database bierze pary wierszy, każda z innej tabeli, dla których warunek wynosi TRUE. Kolumny, które pojawiły się w warunku join, muszą się również pojawić w liście SELECT.

Aby wykonać join na trzech lub więcej tabelach, Oracle najpierw łączy dwie tabele opierając się na porównaniu kolumn i powstałego wcześniej joina. Oracle kontynuuje proces, dopóki wszystkie tabele są złączone w jeden zbiór. Można zoptymalizować kolejność, w której Oracle łączy tabele. Zależy to od warunków, indeksów, czy innych dostępnych statystyk.

Klauzulach WHERE, która posiada warunek join może również zawierać inne warunki, które odnoszą się do kolumn z jednej tabeli. Te warunki zwężają liczbę wierszy wyrzuconą przez zapytanie join.

### **Equijoins**

Equijoin jest joinem z warunkiem zawierającym znak równości. Equijoin łączy wiersze, które mają identyczne wartości dla danych kolumn. Zależnie od wewnętrznego algorytmu, wynik po optymalizacji ma określzoną wielkość.

### **Self Joins**

Self join jest łączeniem tabeli samej ze sobą. Tabela pojawia się dwa razy w klauzuli FROM. Aby wykonać self join, Oracle Database miesza i łączy wiersze z tabeli, która spełnia warunek join.

### **Cartesian Products**

Jeśli dwie tabele w kwerendzie join nie mają warunku join, to Oracle Database zwraca iloczyn kartezjański. Łączy każdy wiersz z każdym. Na przykład, jeśli łączy się dwie tabele z 100 wierszami każda, to iloczyn kartezjański będzie miał 10 000 wierszy.

### **Inner Joins**

Inner join jest joinem dwóch lub więcej tabel, który zwraca te wiersze, które spełniają warunek join.

### **Outer Joins**

Outer join rozszerza warunek inner join. Zawiera wiersze, które spełniają warunek i te wiersze z pierwszej tabeli, które nie mają odpowiadających sobie wierszy z drugiej.

Można użyć outer join, aby zapoelić luki w niekompletnych danych. Taki join się nazywa partitioned outer join i używa query\_partition\_clause ze składni join\_clause.

Oracle rekomenduje użycie FROM ze składni OUTER JOIN raczej niż operatora JOIN (+). Zapytania outer join, które korzystają z operatora (+) mają następujące ograniczenia, których nie ma użycie FROM z OUTER JOIN:

- Operator może się pojawić w WHERE i w FROM
- Jeśli A i B są połączone przez wiele warunków, należy użyć operatora dla wszystkich warunków. Jeśli się tego nie zrobi to Oracle Database wyrzuci wiersze odpowiadające inner join bez wyrzucenia błędu.
- Operator nie wyrzuci outer join, jeśli zdefiniuje się jedną tabelę w outer join, a drugą w inner join.

### **Antijoins**

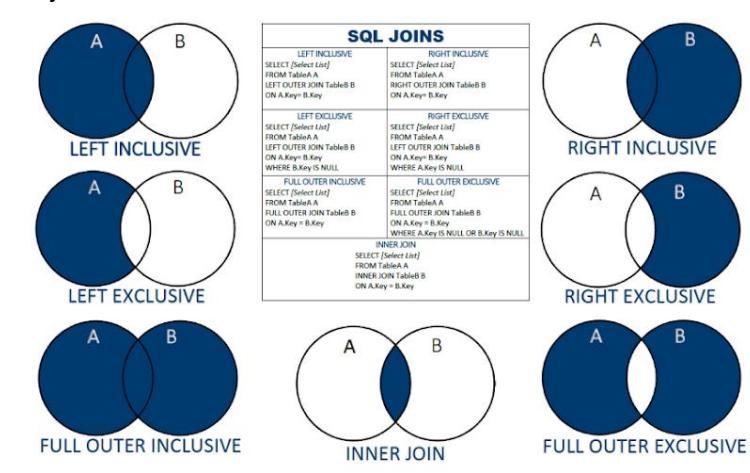
Antyjoin zwraca wiersze z tabeli z lewej strony polecenia, które nie mają odpowiedników w wierszach tabeli po prawej stronie polecenia. Zwraca wiersze, które się nie dopasowują (NOT IN).

### Semijoins

Semijoin zwraca wiersze, które spełniają EXIST, bez duplikowania wierszy z lewej strony warunku, kiedy wiele wierszy z prawej strony warunku spełnia kryteria.

Semijoin i antijoin nie zachodzi jeśli występuje OR w w klauzuli WHERE.

### Wersja 2



### Natural join

```
select department_name, city  
from departments natural join locations;  
Domyslnie łączenie wg wszystkich kolumn o identycznych nazwach i tego samego typu
```

### Using

```
select department_name, city  
from departments join locations  
using (location_id)  
łączenie wg wskazanych kolumn
```

### On

```
select department_name, city  
from departments d join locations l  
on (d.location_id=l.location_id)  
łączenie wg wskazanych kolumn, možliwość wyboru operatora warunku
```

### Cross join

```
select department_name, city  
from departments cross join locations  
łączenie każdego wiersza z każdym! Uwaga na obszerny rezultat.
```

### Łączenia rozszerzone

#### Lewostronnie

```
select last_name, department_name  
from employees e left outer join departments d  
on (e.department_id=d.department_id)  
Zobacz wszystkich pracowników
```

#### Prawostronnie

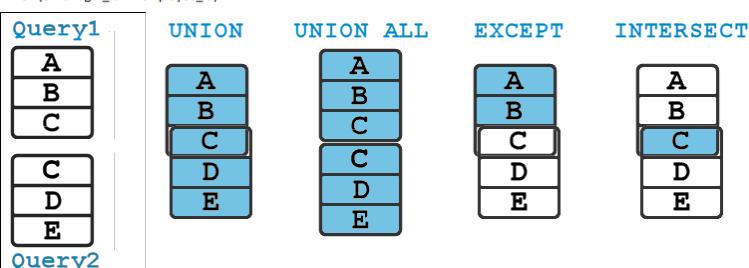
```
select last_name, department_name  
from employees e right outer join departments d  
on (e.department_id=d.department_id)  
Zobacz wszystkie departamenty
```

#### Pełne

```
select last_name, department_name  
from employees e full outer join departments d  
on (e.department_id=d.department_id)  
Zobacz wszystkich pracowników i departamenty
```

### Łączenie tabeli samej ze sobą

```
select e.last_name, e.manager_id, m.employee_id, m.last_name  
from employees e join employees m  
on (e.manager_id=m.employee_id)
```



**3.** Kiedy należy stosować funkcje działające na pojedynczych wierszach, a kiedy funkcje grupowe? Na jakich typach danych działają?

## **FUNKCJE DZIAŁAJĄCE NA POJEDYNCZYCH WIERSZACH**

Funkcje działające na pojedynczych wierszach zwracają pojedynczy wiersz wynikowy dla każdego wiersza tabeli lub widoku, którego dotyczy zapytanie. Funkcje te mogą pojawiać się na listach wyboru, klauzulach WHERE, klauzulach START WITH i CONNECT BY oraz klauzulach HAVING.

W zależności od typu danych, wyróżniamy kilka rodzajów funkcji działających na pojedynczych wierszach.

Rozróżniamy:

### **Funkcje numeryczne - działające na danych liczbowych**

Funkcje numeryczne akceptują wprowadzanie numeryczne i zwracają wartości liczbowe. Większość funkcji numerycznych zwraca wartości NUMBER z dokładnością do 38 cyfr dziesiętnych. Funkcje transcendentalne COS, COSH, EXP, LN, LOG, SIN, SINH, SQRT, TAN i TANH są dokładne z dokładnością do 36 cyfr dziesiętnych. Funkcje transcendentalne ACOS, ASIN, ATAN i ATAN2 mają dokładność do 30 cyfr dziesiętnych.

### **Funkcje znakowe zwracające wartości znakowe**

Zwracają one wartości następujących typów danych, chyba że udokumentowano inaczej:

- Jeśli argumentem wejściowym jest CHAR lub VARCHAR2, zwrócona wartość to VARCHAR2.
- Jeśli argumentem wejściowym jest NCHAR lub NVARCHAR2, zwrócona wartość to NVARCHAR2.

Długość wartości zwracanej przez funkcję jest ograniczona maksymalną długością zwracanego typu danych. W przypadku funkcji, które zwracają CHAR lub VARCHAR2, jeśli długość zwracanej wartości przekracza limit, wówczas baza danych Oracle obciną go i zwraca wynik bez komunikatu o błędzie. W przypadku funkcji zwracających wartości CLOB, jeśli długość zwracanych wartości przekracza limit, wówczas Oracle zgłasza błąd i nie zwraca danych.

### **Funkcje znakowe zwracające wartości liczbowe**

Funkcje znakowe zwracające wartości liczbowe mogą przyjmować jako argument dowolny typ danych znakowych. Funkcje znakowe zwracające wartości liczbowe to: ASCII, INSTR, LENGTH, REGEXP\_COUNT, REGEXP\_INSTR.

### **Funkcje zestawu znaków**

Funkcje zestawu znaków zwracają informacje o zestawie znaków. Funkcje zestawu znaków to: NLS\_CHARSET\_DECL\_LEN, NLS\_CHARSET\_ID, NLS\_CHARSET\_NAME.

### **Funkcje Datetime**

Działają na wartościach data (DATA), znacznik czasu (TIMESTAMP, TIMESTAMP Z STREFĄ CZASOWĄ i TIMESTAMP Z LOKALNĄ STREFĄ CZASOWĄ) oraz wartości przedziału (INTERWAŁ DZIEŃ DO DRUGIEJ, INTERWAŁ ROK NA MIESIĄC).

Niektóre funkcje datetime zostały zaprojektowane dla typu danych Oracle DATE (ADD\_MONTHS, CURRENT\_DATE, LAST\_DAY, NEW\_TIME i NEXT\_DAY). Jeśli podasz wartość znacznika czasu jako argument, baza danych Oracle konwertuje wewnętrznie typ danych wejściowych na wartość DATA i zwraca wartość DATA. Wyjątkami są funkcja MONTHS\_BETWEEN, która zwraca liczbę, oraz funkcje ROUND i TRUNC, które w ogóle nie akceptują znacznika czasu ani wartości interwału. Pozostałe funkcje daty i godziny zostały zaprojektowane tak, aby akceptować dowolny z trzech typów danych (data, znacznik czasu i interwał) i zwracać wartość jednego z tych typów. Wszystkie funkcje datetime, które zwracają bieżące systemowe informacje o czasie, takie jak SYSDATE, SYSTIMESTAMP,

**CURRENT\_TIMESTAMP** itd., Są oceniane raz dla każdej instrukcji SQL, niezależnie od tego, ile razy są do niej odniesienia.

#### **Funkcje XML**

#### **Funkcje JSON**

#### **Funkcje dużych obiektów**

#### **FUNKCJE GRUPOWE**

Funkcje, które zwracają jedną wartość obliczoną na podstawie przekazanego zbioru parametrów, nazywamy funkcjami grupującymi.

Czasami chcemy pogrupować wiersze tabeli i uzyskać jakieś informacje na temat tych grup wierszy. Na przykład możemy chcieć uzyskać średnie ceny różnych typów produktów z tabeli products.

#### **Funkcje agregujące**

Funkcje agregujące zwracają pojedynczy wiersz wyników na podstawie grup wierszy, a nie pojedynczych wierszy. Funkcje agregujące mogą pojawiać się na listach wyboru oraz w klauzulach ORDER BY i HAVING. Są one powszechnie używane z klauzulą GROUP BY w instrukcji SELECT, w której baza danych Oracle dzieli wiersze zapytanej tabeli lub widoku na grupy. W zapytaniu zawierającym klauzulę GROUP BY elementy listy wyboru mogą być funkcjami agregującymi, wyrażeniami GROUP BY, stałymi lub wyrażeniami obejmującymi jedną z nich. Oracle stosuje funkcje agregujące do każdej grupy wierszy i zwraca pojedynczy wiersz wyników dla każdej grupy. Jeśli pominięto klauzulę GROUP BY, Oracle stosuje funkcje agregujące na liście wyboru do wszystkich wierszy w zapytanej tabeli lub widoku. Używasz funkcji agregujących w klauzuli HAVING, aby wyeliminować grupy z danych wyjściowych na podstawie wyników funkcji agregujących, a nie na podstawie wartości poszczególnych wierszy zapytanej tabeli lub widoku.

#### **Funkcje analityczne**

Funkcje analityczne obliczają wartość zagregowaną na podstawie grupy wierszy. Różnią się od funkcji agregujących tym, że zwracają wiele wierszy dla każdej grupy. Grupa wierszy nazywana jest oknem i jest zdefiniowana przez klauzulę analityczną. Dla każdego wiersza zdefiniowane jest przesuwane okno wierszy. Okno określa zakres wierszy używanych do wykonywania obliczeń dla bieżącego wiersza. Rozmiary okien mogą być oparte albo na fizycznej liczbie wierszy, albo na logicznym interwale, takim jak czas. Funkcje analityczne to ostatni zestaw operacji wykonanych w zapytaniu, z wyjątkiem końcowej klauzuli ORDER BY. Wszystkie sprzężenia i wszystkie klauzule WHERE, GROUP BY i HAVING są wypełniane przed przetworzeniem funkcji analitycznych. Dlatego funkcje analityczne mogą pojawiać się tylko na liście wyboru lub klauzuli ORDER BY.

Funkcje analityczne są powszechnie używane do obliczania agregacji, przenoszenia, wyśrodkowania i raportowania agregatów.

#### **Funkcje odwołań do obiektów**

Manipulują wartościami REF, które są odniesieniami do obiektów określonych typów obiektów. Funkcje odwołania do obiektu to: DEREF, MAKE\_REF, REF, REFTOHEX, WARTOŚĆ.

#### **Funkcje OLAP**

Zwracają dane z obiektu wymiarowego w dwuwymiarowym formacie relacyjnym.

Funkcja OLAP to CUBE\_TABLE.

## **WERSJA 2**

Funkcje działające na pojedynczych wierszach są stosowane dla konkretnych zmiennych, w których zmiennymi mogą być zmienne liczbowe, znakowe, daty. Funkcje grupowe są stosowane dla całych zbiorów, jeśli interesuje nas informacja o całym zbiorze danym, a nie poszczególnych jednostkach.

**Funkcje działające na pojedynczych wierszach (obserwacjach):**

- Funkcje numeryczne
- Funkcje transcendentalne (cos, exp, ln, log, sin, tan)
- Funkcje znakowe (length)

**Funkcje grupowe:**

- Agregujące
- Analityczne (order by)

**DODATEK**

## Funkcje działające na pojedynczych wierszach

single-row functions

```
1. funkcje znakowe
Upper, lower, initcap

select *
from employees
where last_name='King';

select *
from employees
where initcap(last_name)='King';

SUBSTR('Ala ma kota',5,2)      → ma
LENGTH('Ala ma kota')          → 11
LPAD ('Ala ma kota', 20, '')   → *****Ala ma kota
REPLACE ('Ala ma kota','A','Jo') → Jola ma kota
```

Dr Danuta Wódz, SGH, 2016

## 1. SN-Budowa

```
2. funkcje liczbowe
round
select round(16.75,1)
from dual;
16.8
select round(16.75,0)
from dual;
17
select round(16.75)
from dual;
17
pominiecie 2. argumentu powoduje przyjęcie wartości domyślnej(0)

tabela dual - tabela pomocnicza do zapytań jednowierszowych
```

## 3. funkcje działające na datach

```
ADD_MONTHS (data1, liczba)
select last_name, hire_date, hire_date+90,add_months(hire_date,3)
from employees;

MONTHS_BETWEEN (data1, data2)
```

---

### 3. funkcje działające na datach

```
ADD_MONTHS (data1, liczba)
select last_name, hire_date, hire_date+90,add_months(hire_date,3)
from employees;

MONTHS_BETWEEN (data1, data2)
```

### 4. funkcje konwersji

```
char do number czasem
number do char zawsze
char do date czasem
date do char zawsze
```

#### Funkcja konwersji to\_char.

A. liczby w rozbudowanym formacie  
format podstawowy 123 → f. rozbudowany 123.00 \$  
to\_char(salary, '999999.99\$')

B. daty w rozbudowanym formacie  
f. podstawowy 01-Feb-2016 → f. rozbudowany 2016 february 1, tuesday, 8:42  
to\_char(sysdate, 'yyyy month dd, day, hh24:mi')

mm	2 cyfry miesiąca
dd	2 cyfry dnia w miesiącu
d	numer dnia w tygodniu
mon	3 literowy skrót nazwy miesiąca
month	pełna nazwa miesiąca
yy	2 ostatnie cyfry roku
yyy	rok pełny
day	nazwa dnia tyg.
hh24	godzina
mi	minuta
ss	sekunda

Dr Danuta Wódz, SGH, 2016

---

## 1. SN-Budowa i ekspl

#### C. standardowa konwersja

```
to_char(salary)
zastosowanie:
np. w funkcji nvl, gdzie jest wymagana zgodność typów argumentów
```

Funkcja konwersji to\_number  
to\_number('123') → 123

Funkcja konwersji to\_date  
przejście z f. rozbudowanego do standardowego  
to\_date('1410 JULY 15','yyyy MONTH dd')

O \_\_\_\_\_

C. standardowa konwersja  
to\_char(salary)  
zastosowanie:  
np. w funkcji nvl, gdzie jest wymagana zgodność typów argumentów

Funkcja konwersji to\_number

Funkcja konwersji to\_date  
przejście z f. rozbudowanego do standardowego  
`to_date('1410 JULY 15','yyyy MONTH dd')`

## 5. funkcje polimorficzne

`nvl(commission_pct,0) - wymaga zgodności typów argumentów  
nvl(commission_pct,'nie pobiera prowizji') → ERROR  
nvl(to_char(commission_pct),'nie pobiera prowizji')`

```
złożenie funkcji  
funkcja1(arg1,arg2)      f. prostą  
funkcja1(funkcja2(arg1),arg2) f.złożoną  
  
case job_id when 'ST_CLERK' then 'URZEZ'  
                when 'IT_PROG' then 'PROGRAMIST'  
                else 'INNY' end
```

+ funkcje grupowe wymienione wyżej na screenach

**4.** Omów klasyfikację funkcji działających na pojedynczych wierszach.

Ad. Pytanie wyżej.

Funkcje działające na pojedynczych wierszach zwracają jako swój wynik jeden rekord dla każdego wiersza z tabeli/widoku którego dotyczy zapytanie. Funkcje mogą znajdować się w klauzuli SELECT, warunkach WHERE, START WITH, CONNECT BY i HAVING.

## Funkcje działające na pojedynczych wierszach dzielimy na

**Funkcje znakowe** - zwracają wartości znakowe następujących typów danych i mogą je przekształcać, trzymając się tych założeń:

- Jeżeli argumentem wejściowym jest CHAR lub VARCHAR2, zwrócona wartość to VARCHAR2.
  - Jeżeli argumentem wejściowym jest NCHAR lub NVARCHAR2, zwrócona wartość to NVARCHAR2.
  - Długość wartości zwracanej przez funkcję jest ograniczona maksymalną długością zwracanego typu danych. chyba, że wskazano inaczej
  - W przypadku funkcji, które zwracają CHAR lub VARCHAR2, jeśli długość zwracanej wartości przekracza limit, wówczas baza danych Oracle obciną go i zwraca wynik bez komunikatu o błędzie.

## Przykładowe funkcje:

- **UPPER** – zwraca znaki CHAR jako wielkie litery
  - **LOWER** – zwraca znaki CHAR jako małe litery
  - **INITCAP** – zwraca znaki CHAR w konfiguracji pierwsza wielka litera i pozostałe małe
  - **PTTRIM** – ucinia od prawej strony CHAR znaki, zaczynając od zdefiniowanego w funkcji

**Funkcje liczbowe** – działają na wartościach liczbowych i zwracają w wyniku wartość liczbową.

Przykładowe funkcje:

- ROUND(n) - zaokrąglanie liczby n
- EXP – zwraca liczbę e poniesioną do potęgi n
- SIN – wyliczenie sinusa dla liczby n

**Funkcje znakowe zwracające wartości liczbowe** – działają na znakach ale zwracają wartości liczbowe. Przyjmują jako argument dowolny typ danych znakowych.

Przykładowa funkcja: LENGTH – funkcja zwraca liczbę znaków z jakiej składa się CHAR

#### **Funkcje działające na datach**

Działają na wartościach data (DATA), znacznik czasu (TIMESTAMP, TIMESTAMP Z STREFĄ CZASOWĄ I TIMESTAMP Z LOKALNĄ STREFĄ CZASOWĄ) oraz wartości przedziału (INTERWAŁ DZIEŃ DO DRUGIEJ, INTERWAŁ ROK NA MIESIĄC).

Niektóre funkcje datetime zostały zaprojektowane dla typu danych Oracle DATE (ADD\_MONTHS, CURRENT\_DATE, LAST\_DAY, NEW\_TIME i NEXT\_DAY). Jeśli podasz wartość znacznika czasu jako argument, baza danych Oracle konwertuje wewnętrznie typ danych wejściowych na wartość DATA i zwraca wartość DATA. Wyjątkami są funkcja MONTHS\_BETWEEN, która zwraca liczbę, oraz funkcje ROUND i TRUNC, które w ogóle nie akceptują znacznika czasu ani wartości interwału. Pozostałe funkcje daty i godzin zostały zaprojektowane tak, aby akceptować dowolny z trzech typów danych (data, znacznik czasu i interwał) i zwracać wartość jednego z tych typów. Wszystkie funkcje datetime, które zwracają bieżące systemowe informacje o czasie, takie jak SYSDATE, SYSTIMESTAMP, CURRENT\_TIMESTAMP itd., są oceniane raz dla każdej instrukcji SQL, niezależnie od tego, ile razy są do niej odniesienia.

Pozostałe funkcje daty i godzin zostały zaprojektowane tak, aby akceptować dowolny z trzech typów danych (data, znacznik czasu i interwał) i zwracać wartość jednego z tych typów. Wszystkie funkcje działające na datach, które zwracają bieżące systemowe informacje o czasie, takie jak SYSDATE, SYSTIMESTAMP, CURRENT\_TIMESTAMP itd., są oceniane dla każdej instrukcji SQL, niezależnie od tego, ile razy są do niej odniesienia.

Przykładowe funkcje:

- ADD\_MONTHS – zwraca w wyniku datę która jest wynikiem dodania do podanej daty zdefiniowanej liczby miesięcy
- MONTHS\_BETWEEN – zwraca liczbę miesięcy pomiędzy podanymi dwoma datami
- CURRENT\_DATE – zwraca bieżącą datę w strefie czasowej sesji
- SYSDATE – zwraca bieżącą datę i godzinę ustawioną dla systemu operacyjnego, w którym rezyduje serwer bazy danych.
- ROUND (date) – zwraca datę zaokrągloną do jednostki określonej przez model formatu fmt

**Funkcje konwersji** - konwertują wartość z jednego typu danych na inny.

Zasadniczo forma nazw funkcji jest zgodna z konwencją typu danych TO. Pierwszy typ danych to typ danych wejściowych. Drugi typ danych to wyjściowy typ danych.

Przykładowe funkcje konwersji:

- TO\_CHAR – zmienia liczby, daty, typy znakowe na typ danych VARCHAR2
- TO\_NUMBER - konwertuje wyrażenie na wartość typu danych NUMBER
- TO\_TIMESTAMP - konwertuje dane typu CHAR, VARCHAR2, NCHAR lub NVARCHAR2 na wartość typu TIMESTAMP
- CAST - konwertuje jeden zdefiniowany typ danych lub wartość w konkretnym typie na inny wbudowany typ danych lub wartość w zdefiniowanym typie.

## **5. Jakie znasz polecenia zmieniające zawartość tabeli? Jakie są ich skutki oraz zakres oddziaływania?**

DML – Data Modification Language – odtłam SQL zajmujący się manipulacją danych.

Za manipulację danymi odpowiadają polecenia z grupy DML, służące do umieszczania rekordów w bazie, kasowania, oraz dokonywania zmian na istniejących danych.

Poleceniami należącymi do tej grupy są:

- INSERT – umieszczenie danych w bazie, (warto dodać, że dodajemy nowe wiersze)
  - UPDATE – modyfikacja danych,
  - Truncate - Usunięcie wszystkiego z tabeli .
- DELETE – usunięcie danych z bazy, (warto dodać, że usuwamy wybrane wiersze)
- Zakres polecenia INSERT będzie adekwatny do ilości wprowadzanych danych, natomiast zakres oddziaływania polecień UPDATE czy DELETE definiuje w zapytaniu warunek filtrujący WHERE. Brak użycia tego warunku spowoduje zmiany bądź usunięcie wszystkich rekordów w danej tabeli.

(Podczas uruchamiania instrukcji DML baza danych zapewnia, że wiersze tabel zachowają spójność. Oznacza to, że zmiany wprowadzane w wierszach nie mogą wpływać na zależność klucza głównego i obcego modyfikowanych tabel. Klucze te bowiem muszą pozostać unikalne o czym świadczą więzy integralności)

Aby zmiany wykonane za pomocą polecen z grupy DML były widoczne w systemie muszą zostać zatwierdzone poleceniem commit. W przeciwnym razie wszelkie modyfikacje będą widoczne jedynie w ramach sesji w której zostały uruchomione tworząc tzw. transakcję, czyli grupę powiązanych ze sobą instrukcji SQL. Po użyciu polecenia commit zmieniona zawartość tabeli będzie widoczna dla wszystkich sesji oraz zapytań wykonanych po zatwierdzonej operacji.

Oprócz zatwierdzenia użytkownik wykonujący zmiany w zawartości tabeli ma również możliwość wycofania swoich operacji przy użyciu polecenia rollback, pod warunkiem, że nie został wcześniej wykonany commit.

### **DODATEK**

Tutaj warto dodać polecenie *alter* z zakresu DDL (data definition language), które pozwala na dodanie/usunięcie kolumny, modyfikację kolumn (zmiana typu danych).

### **PRZYKŁADY**

## 6. DML

insert - wstawianie wiersza  
update - aktualizacja  
delete - usunięcie

Uwaga! Pracujemy na kopiiach  
create table ecopy

as select \*  
from employees;

create table dcopy  
as select \*  
from departments;

### Insert

insert into dcopy  
values (120,'Operations',107,1400);

wartości do wszystkich, kolejność kolumn naturalna

insert into dcopy(location\_id,department\_id,department\_name)  
values (1400,120,'Operations');

insert into ecopy(employee\_id,last\_name,salary,department\_id)  
values (120,'JANES',4500,20);

emp\_history - dane byłych pracowników, struktura taka jak w ecopy

insert into emp\_history  
select \*  
from ecopy  
where employee\_id=202;

Odszedł pracownik – 202 przeszedł na emeryturę, jego dane zostały skopiowane.

### Delete

delete from ecopy  
where employee\_id=202;

Dane pracownika 202 usunięte z tabeli aktualnych pracowników.

### Update

update ecopy  
set salary=13000,  
 manager\_id=205,  
 job\_id='MK\_MAN'  
where employee\_id=103;

update ecopy  
set salary=salary\*1.1;

update ecopy  
set salary=(select salary  
 from employees  
 where employee\_id=205)  
where last\_name='Hunold'

---

### DDL:

#### Alter

alter table: add, modify, drop

alter table ecopy  
modify last\_name varchar2(40);

## 6. Jaką rolę pełni Data Dictionary (Słownik Danych) i jak się nim posługiwać?

### WERSJA 1

Jest to zbiór tabel, który przechowuje informacje o bazie danych (inaczej nazywane metadane).

Data Dictionary zawiera informacje o:

- elementach baz danych jak tabele, wiersze, kolumny
- wykorzystywanej pamięci
- ograniczeniach
- uprawnienia użytkowników
- aktualizacjach Jak się nim posługiwać?

Excel – żeby był opis tych danych (instrukcja) np. w ten sposób są kodowane takie wartości

**WERSJA 2** [https://docs.oracle.com/cd/B10501\\_01/server.920/a96524/c05dicti.htm](https://docs.oracle.com/cd/B10501_01/server.920/a96524/c05dicti.htm)

Jednym z najważniejszych elementów bazy danych Oracle jest Słownik Danych (Data Dictionary). Jest to zbiór tabel read-only, który przechowuje informacje o bazie danych.

Słownik Danych zawiera:

Definicje elementów bazy (tables, views, indexes, clusters, synonyms, sequences, - procedures, functions, packages, triggers, itp.)

- Informacje o ilości wykorzystywanej pamięci
- Domyślne wartości kolumn
- Informacje o ograniczeniach integralności
- Nazwy użytkowników Oracle oraz uprawnienia i role przyznawane użytkownikom
- Informacje o zmianach wykonanych przez użytkowników, np. o aktualizacjach
- Inne ogólne informacje o bazie danych

Struktura Słownika danych to tabele i widoki przechowywane w obszarze danych SYSTEM bazy danych. Ponieważ tabele Słownika danych są read-only, można wykorzystać jedynie zapytania SQL (SELECT), aby uzyskać do nich dostęp.

### 1. Struktura Słownika danych:

**Base Tables** – podstawowe tabele przechowujące informacje o relacjach w bazie danych. Użytkownicy rzadko uzyskują do nich bezpośredni dostęp, ponieważ są znormalizowane, a większość danych przechowywana jest w zaszyfrowanym formacie.

**User- Accessible Views** – widoki zawierające podsumowania i informacje z podstawowych tabel. Widoki te dekodują dane tabeli podstawowej na użyteczne informacje, takie jak nazwy użytkowników lub tabel za pomocą JOINów lub klauzuli WHERE w celu uproszczenia informacji. Większość użytkowników ma dostęp do tych widoków, a nie tabel podstawowych.

**SYS, Owner of the Data Dictionary** – użytkownik Oracle SYS posiada wszystkie tabele podstawowe i dostępne dla użytkownika widoki słownika danych. Żaden użytkownik nie powinien zmieniać (UPDATE, DELETE, INSERT) jakikolwiek wierszy lub obiektów schematu zawartych w schemacie SYS. Takie działanie może zagrozić integralności danych.

### 2. Korzystanie ze Słownika danych

Podstawowe zastosowania:

- Oracle uzyskuje dostęp do słownika, aby znaleźć informacje o użytkownikach, obiektach schematu i strukturach pamięci.
- Oracle modyfikuje słownik danych za każdym razem, gdy wydawana jest instrukcja języka definicji danych (DDL)
- Każdy użytkownik Oracle może używać słownika danych jako odwołania tylko do odczytu w celu uzyskania informacji o bazie danych.

### 3. Jak Oracle korzysta ze Słowników danych:

Dane w tabeli bazowej słowników są niezbędne dla funkcji Oracle. Dlatego tylko Oracle powinno zapisywać lub zmieniać informacje ze słownika danych. Oracle udostępnia skrypty do modyfikowania tabel słownika danych, gdy baza danych jest aktualizowana. Podczas operacji na bazie danych, Oracle czyta słownik danych, aby upewnić się, że istnieją obiekty schematu i że użytkownicy mają do nich odpowiedni dostęp. Oracle aktualizuje także słownik danych w sposób ciągły, aby odzwierciedlić zmiany w strukturach bazy danych, np. jeżeli użytkownik X utworzy tabelę o nazwie „PARTS”, wówczas do słownika danych zostaną dodane nowe wiersze, które odzwierciedlają nową tabelę, kolumny, segmenty, zakresy i uprawnienia, które X ma do tej tabeli. Te nowe informacje są następnie widoczne przy następnym zapytaniu o widoki słownika.

**Public Synonyms for Data Dictionary Views**- Oracle tworzy publiczne synonimy dla wielu widoków słownika danych, aby użytkownicy mieli do nich wygodny dostęp. Administrator Security również może utworzyć dodatkowe publiczne synonimy dla obiektów schematu, które są używane w całym systemie. Użytkownicy powinni unikać nazywania własnych obiektów schematów takimi samymi nazwami, jak te używane w synonimach publicznych.

**Cache the Data Dictionary for Fast Access**- wiele informacji ze słownika danych jest przechowywanych w SGA w pamięci podręcznej słownika, ponieważ Oracle stale uzyskuje dostęp do słownika danych podczas operacji na bazie danych, aby sprawdzić poprawność dostępu użytkownika i zweryfikować stan obiektów schematu. Wszystkie informacje są przechowywane w pamięci przy użyciu algorytmu ostatnio używanego (the least recently used (LRU)).

**Inne programy i słowniki danych**- mogą odwoływać się do istniejących widoków i tworzyć własne tabele słownika danych lub własne widoki. Twórcy aplikacji piszący programy odwołujące się do słownika danych powinni odwoływać się do publicznych synonimów, a nie do bazowych tabel, synonimy rzadziej zmieniają się między wersjami oprogramowania.

## 7. W jakim celu buduje się perspektywy? Omów możliwe klauzule polecenia do tworzenia perspektyw.

### Wersja 1

Perspektywy – to „nakładka” na tabele, która ogranicza informacje np. tylko do imienia i nazwiska bez podawania adresu, ID, email etc. Perspektywy redukują czas zapytania.

Perspektywy buduje się w celu:

- Ukrycie złożoności zapytania do tabeli przed użytkownikiem
- Nie umożliwia użytkownikom bezpośredniego konstruowania zapytań do tabeli
- Przyznaje dostęp użytkownikom tylko do widoków
- Przyznaje dostęp użytkownikom tylko do określonych wierszy

Klauzule:

```
create view nazwa_perspektywy  
drop view nazwa_perspektywy  
replace view nazwa_perspektywy  
create (or replace) view nazwa_perspektywy as select from
```

### WERSJA 2

To nie jest osobna struktura z danymi, tylko definicja. System pamięta definicję - ograniczającą zakres widzianych danych, można nadać uprawnienie do perspektywy - unikamy wielokrotnego pisania skomplikowanych poleceń - bezpieczniejsze rozwiązanie niż skrypty SQL pamiętane poza bazą

```
create view dept50 as select * from employees where department_id=50;
select * from dept50;
select last_name, salary*12 from dept50 where salary<8000 order by hire_date;
Perspektywa zachowuje się jak tabela, ta sama składnia select dla tabel i perspektyw
create view pracownicy (nazwisko,pensja_roczna,data_zatr) as select
last_name,salary*12,hire_date from employees where salary<10000
create view dane (nazwisko, nazwa_dep) as select last_name,department_name from
employees,departments where employees.department_id=departments.department_id;
drop view pracownicy;
create or replace view
```

#### **WERSJA 3**

Perspektywa jest predefiniowanym zapytaniem jednej lub wielu tabel (zwanych tabelami bazowymi). Pobieranie informacji z perspektywy odbywa się w taki sposób jak pobieranie informacji z tabeli. Wystarczy jedynie umieścić nazwę perspektywy w klauzuli FROM.

Cele budowania perspektyw:

- Umożliwienie umieszczenia złożonego zapytania w perspektywie i przyznania do niej dostępu użytkownikom. To pozwala ukryć złożoność przed użytkownikami.
- Pozwalają na uniemożliwienie użytkownikom bezpośredniego wysyłania zapytań do tabel bazy danych, przyznając im dostęp jedynie do widoków.
- Umożliwiają przyznanie perspektywie dostępu jedynie do określonych wierszy tabel bazowych, co pozwala na ukrywanie wierszy przed użytkownikami.

## **8. Operacje na zbiorach – omów składnię poleceń i znaczenie uzyskanych wyników.**

#### **WERSJA 1**

- Union all (suma zbiorów)
- Union (nie powtarzające się elementy)
- Intersect (część wspólna)
- Minus (odejmowanie)

#### **WERSJA 2**

Operują one zawsze, na wynikach całych kwerend (tabel wejściowych) i zwracają tabelę wynikową, będącą zbiorem identycznym jak określony jak pierwsza tabela wejściowa (liczba nazwy kolumn). Zawierają jednak elementy (wiersze), zgodne z arytmetyką zbiorów, określona przez operator :

**UNION ALL** – zwraca wszystkie wiersze pobrane przez zapytania, łącznie z tymi powtarzającymi się.

```
SELECT product_id,product_type_id, name FROM products UNION ALL SELECT prd_id,
prd_type_id, name FROM more_products;
```

**UNION** – zwraca jedynie niepowtarzające się wiersze zwrócone przez zapytania.

SELECT product\_id, product\_type\_id, name FROM products UNION SELECT prd\_id, prd\_type\_id, name FROM more\_products;  
**INTERSECT** – zwraca jedynie te wiersze, które zostały pobrane przez obydwa zapytania.  
SELECT product\_id, product\_type\_id, name FROM products INTERSECT SELECT prd\_id, prd\_type\_id, name FROM more\_products;  
**MINUS** – zwraca wiersze powstałe po odjęciu tych pobranych przez drugie zapytanie od tych pobranych przez pierwsze zapytanie.  
SELECT product\_id, product\_type\_id, name FROM products MINUS SELECT prd\_id, prd\_type\_id, name FROM more\_products;  
Jest kilka zasad, które muszą być spełnione. Warunkiem podstawowym, które regokolwiek ze sposobów operowania na zbiorach w sposób pionowy, jest podobna struktura tabel wejściowych. Liczba kolumn w każdym zbiorze (kwerendzie), musi być identyczna oraz typy danych poszczególnych kolumn, muszą do siebie pasować. Nazwy kolumn, nie mają znaczenia. W zbiorze wynikowym, atrybuty będą nazwane tak jak w pierwszej z kwerend.

## 9. Przedstaw podzapytania – typy, klauzule, w których mogą wystąpić, operatory.

W SQL zapytanie to coś ogólnego, a podzapytania są do wyciągania szczegółowych informacji.  
Podzapytanie to osadzenie jednej instrukcji w drugiej instrukcji (funkcja w funkcji). Można umieścić w klauzuli where, having, from czy instrukcji select. Zewnętrzne zapytanie czeka na wynik wewnętrznego zapytania.  
Rodzaje podzapytań:

- Jednowierszowe – zwracają do zewnętrznej instrukcji SQL zero lub jeden wiersz np. zwraca 1 użytkownika jako wynik
- Wielowierszowe – zwracają do zewnętrznej instrukcji SQL co najmniej jeden wiersz np. zwraca kilka osób jako wynik
- Wielokolumnowe – zwracają więcej niż jedną kolumnę
- Skorelowane – odwołują się do jednej lub kilka kolumn zewnętrznej instrukcji SQL.
- Zagnieżdżone – umieszczone są wewnętrznie innego podzapytania

### WERSJA 2

Podzapytanie (podkwerenda) to osadzenie jednej instrukcji w innej instrukcji. Możliwe jest łączenie zapytań bez względu na różny typ danych w funkcje zagnieżdżone. Podzapytanie może również zawierać inne podzapytanie (Oracle Database nie posiada ograniczeń na liczbę poziomów podzapytań w klauzuli FROM zapytania najwyższego poziomu, natomiast w klauzuli WHERE można zagnieździć do 255 poziomów).

Wyróżniamy dwa podstawowe rodzaje podzapytań:

- podzapytanie jednowierszowe – zwracają do zewnętrznej instrukcji SQL zero lub jeden wiersz. Podzapytanie możemy umieścić w klauzuli WHERE, klauzuli HAVING lub klauzuli FROM instrukcji SELECT. Istnieje specjalny przypadek jednowierszowego podzapytania, który zawiera dokładnie jedną kolumnę. Tego rodzaju podzapytanie nazywamy podzapytaniem skalarnym.
- podzapytanie wielowierszowe – zwracają do zewnętrznej instrukcji SQL co najmniej jeden wiersz. Podzapytania wielowierszowe zwracają jeden wiersz lub kilka wierszy do zapytania zewnętrznego. Zapytanie zewnętrzne może obsługiwać

podzapytania zawierające wiele wierszy za pomocą operatorów IN (sprawdza czy wartość znajduje się na konkretnej liście wartości, możliwe również zastosowanie NOT IN), ANY (stosuje do porównania wartości z każdą wartością na liście; należy przed nim umieścić operator porównania : =, <, >, <=, >=), ALL (stosuje do porównania wartości z każdą wartością na liście; należy przed nim umieścić operator porównania : =, <, >, <=, >=). Do sprawdzenia czy wartości znajduje się na liście pochodzącej z podzapytania skorelowanego możemy użyć operatora EXISTS.

Wyróżniamy ponadto trzy rodzaje podzapytań, które mogą zwrócić jeden wiersz lub wiele wierszy:

- podzapytania wielokolumnowe - zwracają do zewnętrznej instrukcji SQL więcej niż jedną kolumnę.
  - – odwołują się do jednej lub kilku kolumn zewnętrznej instrukcji SQL. Takie podzapytania nazywamy „skorelowanymi”, ponieważ są one powiązane z zewnętrzna instrukcją SQL poprzez te same kolumny. Używane gdy chcemy uzyskać odpowiedź na pytanie dotyczące wartości w każdym wierszu znajdującym się w zewnętrznym zapytaniu (np. chcemy sprawdzić czy występuje relacja między danymi ale nie interesuje nas ile wierszy zostało zwróconych przez podzapytanie). Podzapytanie skorelowane wykonywane jest raz dla każdego wiersza w zapytaniu zewnętrznym, co odróżnia je od podzapytania nieskorelowanego które jest uruchamiane jeden raz przed uruchomieniem zapytania zewnętrznego.

· podzapytania zagnieżdżone – są umieszczone wewnętrz innego podzapytania (można zagnieździć do 255 poziomów podkwerend).

Podzapytania w klauzuli WHERE – umieszczone w nawiasach okrągłych (); możemy wykorzystywać operator równości (=) lub operatory porównania (<, <=, >, >=).

Podzapytania w klauzuli HAVING – pozwala na filtrowanie grupy wierszy na podstawie wyników zwracanych przez podzapytanie;

Podzapytania w klauzuli FROM – z perspektywy klauzuli FROM zewnętrznego zapytania wyniki podzapytania są po prostu kolejnym źródłem danych.

Podzapytanie w klauzuli FROM instrukcji SELECT jest także nazywane widokiem wbudowanym, natomiast podzapytanie w klauzuli WHERE instrukcji SELECT – podzapytaniem zagnieżdżonym. Zastosowanie podzapytań:

- zdefiniowanie zestawu wierszy w celu wstawienia do tabeli docelowej (w instrukcjach INSERT lub CREATE TABLE)
  - zdefiniowanie zestawu wierszy, które mają zostać uwzględnione w widoku lub widoku zmaterializowanym (w instrukcjach CREATE VIEW lub CREATE MATERIALIZED VIEW)
  - zdefiniowanie wartości, które zostaną przypisane do istniejących wierszy w instrukcji UPDATE
  - podanie wartości dla warunków w klauzulach WHERE, HAVING lub START WITH instrukcji SELECT, UPDATE i DELETE
  - zdefiniowani tabeli, która będzie obsługiwana przez zapytanie zawierające; Wykonuje się to umieszczając podzapytanie w klauzuli FROM zapytania zawierającego(tak jak w przypadku nazwy tabeli). W ten sposób można używać podzapytań zamiast tabel w instrukcjach INSERT, UPDATE i DELETE.
- Wewnątrz klauzuli WITH można umieszczać podzapytania, do których odwoływać się można poza klauzulą WITH, co nazywa się przygotowaniem podzapytań.

**DODATEK:**

### **Podzapytanie zwraca jedną wartość**

Znajdź osoby, które zarabiają więcej od Fay

```
select *  
from employees  
where salary > (select salary  
                 from employees  
                 where last_name='Fay');
```

Znajdź osoby, które pracują na tym samym stanowisku co 178, ale zarabiają więcej od niego

```
select *  
from employees  
where job_id=(select job_id  
                  from employees  
                  where employee_id=178)  
and salary >(select salary  
                  from employees  
                  where employee_id=178);
```

Uwaga! select wewnętrzny musi zwrócić dokładnie jedną wartość.

Znajdź wszystkie osoby pracujące na tym samym stanowisku co Matos.

```
select *  
from employees  
where job_id=(select job_id  
                  from employees  
                  where last_name='Matos')  
and last_name<>'Matos';
```

### **Zniesienie ograniczeń związanych z f. grupowymi**

```
select max(salary)  
from employees;
```

```
select *  
from employees  
where salary=(select max(salary)  
                  from employees);
```

```
select department_id,avg(salary)  
from employees  
group by department_id  
having avg(salary)=(select max(avg(salary))  
                  from employees  
                  group by department_id);
```

## **Podzapytanie zwraca wiele wartości - inne operatory**

```
select *  
from employees  
where salary in (select min(salary)  
                  from employees  
                  group by department_id);
```

1. najmniejsze pensje w departamentach: 100, 200, 300
2. osoby o pensji równej albo 100 albo 200 albo 300

```
select *  
from employees  
where salary < any (select salary  
                     from employees  
                     where job_id='ST_CLERK');
```

---

## **10. Omów typowe rozwiązania Big Data w obszarze baz/repozytoriów danych.**

Rozwiązania Big Data w obszarze baz danych: HDFS, Hive, Cassandre, HBase, Druid, MongoDB, Redis.

**HDFS (Hadoop Distribute File System)** – jest to rozproszony system plików umieszczony na wielu serwerach. Cechuje się:

- Wysokim poziomem tolerancji na awarie (bo dużo serwerów)
- Przetwarzanie wsadowe – dostęp do danych przesyłanych strumieniowo
- Przechowywanie dużych zbiorów danych – wysoka przepustowość danych
- Język zapytań HQL (podobna składnia co SQL) HIVE – oprogramowanie, które ułatwia odczytywanie, zapisywanie, zarządzanie dużymi zbiorami danych znajdującymi się w magazynach rozproszonych.

**Cassandra** – kolumnowa baza danych (dane są przechowywane pionowo – kolumnami). Cechuje się:

- Krótkimi czasami odpowiedzi
- Posiada cechy bazy klucz – wartość (key – values -> najmniej skomplikowany sposób)
- Operuje na rodzinach kolumn, kluczach

**HBase** – rozproszona, kolumnowa baza danych. Charakteryzuje się:

- Nierelacyjna baza danych
- obsługuje zapytania w pamięci operacyjnej
- Dobrze sprawdza się do obsługi szybkich zapytań na duże zbiory
- Mniej radzi sobie z zapytaniami analitycznymi jak agregacje danych Druid – kolumnowa baza danych wspierająca analizy wielowymiarowe (OLAP)

**MongoDB** – nierelacyjna baza danych (noSQL działają w czasie rzeczywistym).

Cechuje się:

- Dużą skalowalnością i wydajnością
- Brak ściśle zdefiniowanej struktury
- Posiada możliwość składowania danych w pamięci operacyjnej

**Redis** – baza danych klasy in – memory. Cechuje się:

- Składowaniem danych w pamięci operacyjnej
- Oferuje bardzo szybki odczyt i zapis danych

## Wersja 2

- HDFS (Hadoop distributed file system)
  - Jest to rozproszony system plików przeznaczony do pracy na sprzęcie komputerowym
    - Używany jest do skalowania od pojedynczego klastra Apache Hadoop do nawet tysiąc wezłów
    - Zapewnia następujące funkcje:
      - Szybkie odtwarzanie po awarii sprzętu
      - Dostęp do danych przesyłanych strumieniowo
      - Przechowywanie dużych zbiorów danych
      - Spójność
      - Wydajność
      - Poręczność – kompatybilny z wieloma systemami operacyjnymi
- HIVE
  - Jest to oprogramowanie magazynu danych Apache Hive, które ułatwia odczytywanie, zapisywanie i zarządzanie dużymi zbiorami danych znajdującymi się w magazynach rozproszonych
    - Zapytania składamy w języku HiveSQL
    - Zapewnia następujące funkcje:
      - Umożliwia łatwy dostęp do danych
      - Mechanizm narzucania struktury na różne formaty danych
      - Dostęp do plików przechowywanych bezpośrednio w HDFS
      - Wbudowane złącze dla plików tekstowych z wartościami oddzielonymi przecinkami i tabulatorami o Hive nie jest przeznaczony do przetwarzania transakcji online
- Cassandra o Rozproszona baza danych napisana w Java
  - Obsługuje duże zbiorы danych, odpowiada na zapytania SQL
  - Współpraca z Hadoop i Spark, możliwość wykonywania algorytmów opartych na MapReduce
    - Automatyczna replikacja danych

## 11. Przedstaw specyfikę środowisk analitycznych stosowanych w Big Data.

Na środowiska analityczne w systemie Big Data składa się:

1. Źródło danych - stąd płynie strumień danych, które są dalej przetwarzane i analizowane. Z reguły dane pochodzą z systemów operacyjnych lub innych miejsc jak na przykład media społecznościowe.
2. Proces i miejsce ładowania danych - definiowanie miejsca do którego będziemy ładować dane z systemów źródłowych. Dane te będą tutaj tymczasowo przechowywane, sortowane i klasyfikowane w określone tematy, tak aby komponenty z kolejnych warstw mogły je konsumować w zależności od swoich własnych potrzeb i wymagań. Narzędzia, które mogą obsłużyć ten krok: Kafka, RabbitMQ.
3. Przechowywanie danych - to najniższa warstwa w części analitycznej. Narzędzia z tej warstwy pozwalają na składowanie danych o różnych formatach i różnym przeznaczeniu.

Podczas budowania naszej architektury do wyboru mamy tutaj: bazy relacyjne, bazy NoSQL, obiektowe pamięci masowe (S3), kolekcje indeksów, analityczne bazy danych i systemy plików. Warto wiedzieć, że wybór narzędzia należy zawsze dostosować do scenariusza wykorzystania. Na przykład, na potrzeby przechowywania zdjęć, filmów i opisów produktów w sklepie internetowym lepszą bazą będzie mongoDB. Z kolei do przechowywania logów lub danych z sensorów bardziej będzie nadawać się Casandra. HBase natomiast świetnie sprawdzi się w przypadku przechowywania danych niestrukturalnych dla środowiska transakcyjnego.

- a) W bazach NoSQL będziemy przechowywać dane niestrukturalne, które najczęściej będą wykorzystywane po stronie środowisk operacyjnych. Będą to te dane, dla których musimy zagwarantować między innymi krótkie czasy opóźnień oraz szybkie inserty. Narzędzia, które mogą obsłużyć ten krok: mongoDB, Casandra, HBase.
  - b) System plików HDFS wykorzystamy do budowy jeziora danych (data lake). Dane tutaj przechowujemy w postaci surowej (RAW) i używamy do przetwarzania wsadowych. Nie korzystamy z żadnego schematu/modelu, tak jak to ma miejsce w przypadku tradycyjnych hurtowni. Przykładowe narzędzia do wykorzystania w tym miejscu: Cloudera, Hortonworks, MapR.
  - c) Przestrzeń obiektowa S3 jest coraz częściej wykorzystywana w środowiskach big data. Jest to miejsce do składowania danych „chłodnych”, czyli takich, które nie są często używane. Dane przechowywane w tym miejscu mogą być opisane kontekstowo z wykorzystaniem metadanych, które są częścią obiektu i są przechowywane razem z danymi. Storage S3 jest to również dobre miejsce do przechowywania danych, które muszą spełniać określone regulacje prawne. Przed audytorami i regulatorami znakomicie „broni się” obiektowa pamięć masowa, która korzysta z technologii WORM. Przykładowe narzędzia do wykorzystania w tym miejscu: Hitachi Content Platform, Dell-EMC Elastic Cloud Storage, AWS S3.
4. Bazy danych - tutaj definiujemy schematy i modele dla danych, które są przechowywane poniżej w jeziorze danych. Na żądanie użytkowników biznesowych i na potrzeby analityki danych tworzymy tzw. data marty. Do wykorzystania mamy bardzo różne typy baz danych: MPP (Massively Parallel Processing), OLAP (OnLine Analytical Processing), bazy kolumnowe, a nawet bazy relacyjne. Przykładowe narzędzia do wykorzystania w tym miejscu: Teradata, Vertica, Netezza, SAP HANA, PostgreSQL.
5. Przetwarzanie danych:
- a) Przetwarzanie w czasie rzeczywistym – każde zdarzenie jest procesowane indywidualnie, nie jesteśmy świadomi zdarzeń i danych historycznych. Informacja zwrotna jest przekazywana natychmiast. Przykładowe narzędzia do wykorzystania w tym miejscu: Hitachi Operational Intelligence (HOI), Spark, Flink.
  - b) Przetwarzanie wsadowe – procesy są grupowane i przetwarzane jednocześnie. Wykorzystywane do analityki danych historycznych. Przykładowe narzędzia do wykorzystania w tym miejscu: Spark, MR (MapReduce).
  - c) Przetwarzanie mikro-wsadowe – jest czymś pośrednim pomiędzy tymi dwoma poprzednimi typami. Korzystamy tutaj z technik stosowanych w przetwarzaniu wsadowym, ale dla scenariuszy czasu rzeczywistego. Prawie każdy przypadek przetwarzania w czasie rzeczywistym można przeprocesować za pomocą technik wykorzystywanych w przetwarzaniu mikro-wsadowym. Przykładowe narzędzia do wykorzystania w tym miejscu: Spark.

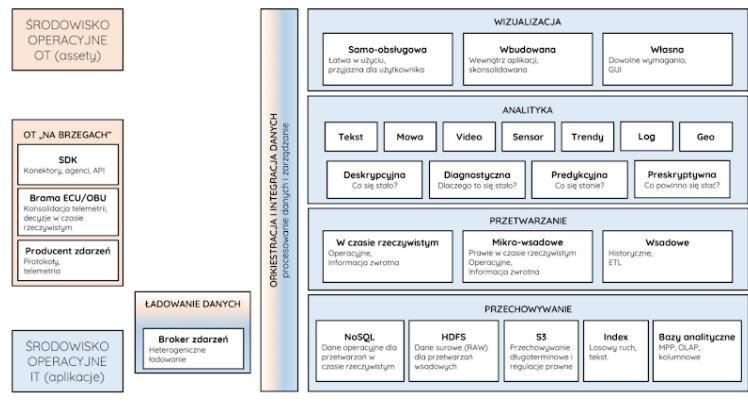
## 6. Analityka:

- deskrypcyjna – badamy co się dzieje lub co się stało,
- diagnostyczna (jest to specyficzny rodzaj tej poprzedniej) – szukamy odpowiedzi, dlaczego to się stało,
- predykcyjna – chcemy wiedzieć co się wydarzy,
- preskryptywna – interesuje nas odpowiedź na pytanie co się powinno wydarzyć (tutaj używamy analityki predykcyjnej i dodatkowo podpowiadamy akcję „co należy zrobić”).

Przykładowe narzędzia do wykorzystania w tym miejscu: Pentaho, Cognos, Microstrategy, Spark MLlib, Python, R.

## 7. Wizualizacja:

- wizualizacja samoobsługowa – zależy nam na tym, aby dostarczyć narzędzie łatwe w obsłudze, z przyjemnym interfejsem dla użytkownika biznesowego. Użytkownik ten będzie mógł samodzielnie tworzyć raporty oraz korzystać z paneli informacyjnych.
- wizualizacja wbudowana – chcemy, aby nasz silnik wizualizacji został wbudowany wewnątrz aplikacji biznesowej. Celem jest dostarczenie narzędzia, z którego będzie można korzystać bez konieczności wychodzenia z tej aplikacji (np. wizualizacja realizowana wewnątrz Salesforce'a).
- wizualizacja własna (custom) – dostarczamy dowolnych (własnych i „customizowanych”) paneli informacyjnych oraz raportów w odpowiedzi na każde zapotrzebowanie użytkownika. Przykładowe narzędzia do wykorzystania w tym miejscu: Tableau, QlickView, Power BI.



## 12. Omów wybrany algorytm stosowany w analityce Big Data.

Tutaj można wrzucić MapReduce, ale też inne.

- Algorytm – jest to skończony ciąg jasno zdefiniowanych czynności koniecznych do wykonania pewnego rodzaju zadań, sposób postępowania do rozwiązania problemu.
- Etapy algorytmu k-średnich:
  - Inicjalizacja: wybranie początkowego zbioru środka klastra
  - Przypisanie obserwacji do i-tego klastra, którego średnia jest najbliższa obserwacji.
  - Wyliczenie nowych średnich dla klastrów przy stałym przypisaniu obserwacji
  - Obliczenie SSE (sumy kwadratów błędów)
  - Powrót do kroku drugiego i zapętlenie, aż zostanie spełnione kryterium zbieżności
- Algorytm dąży do minimalizacji SSE co jest problematyczne, ponieważ algorytm może być optymalizowany w ekstremum lokalnym, a nie globalnym.
- Metoda jest bardzo wrażliwa na skalowanie zmiennych, należy używać danych o współmiernych jednostkach

Algorytm – skończony ciąg jasno zdefiniowanych czynności koniecznych do wykonania pewnego rodzaju zadań, sposób postępowania prowadzący do rozwiązania problemu.

Algorytm – jednoznaczny przepis obliczenia w skończonym czasie pewnych danych wejściowych do pewnych danych wynikowych.

Zazwyczaj przy analizowaniu bądź projektowaniu algorytmu zakłada się, że dostarczane dane wejściowe są poprawne, czasem istotną częścią algorytmu jest nie tylko przetwarzanie, ale i weryfikacja danych.

C4.5 jest algorytm używany do generowania drzewa decyzyjnego opracowane przez Ross Quinlan . C4.5 jest przedłużeniem wcześniejszego algorytmu Quinlan: ID3 . Drzewa decyzyjne wygenerowane przez C4.5 może być stosowany do klasyfikacji, i z tego powodu, C4.5 jest często określany jako klasyfikatora statystycznych .

C4.5 buduje drzewa decyzyjne ze zbioru danych treningowych w taki sam sposób jak ID3 , używając pojęcia informacji entropii. Dane treningowe to zestaw z już sklasyfikowanych próbek. Każda próbka składa się z wektora p-wymiarowej , w której przedstawiają wartości atrybutów i cechy próbki, jak również grupy, w których spada.  $S = \{s_{-1}, \{2\} s_{-2} \dots s_l\} | X_{-1} \{\{1, l\}\} x_{-2} \{\{2, l\}\}, \dots, \{\{x_{-P}, l\}\} x_{-l} \{s_l\}$

W każdym węźle drzewa, C4.5 wybiera atrybut danych, który najskuteczniej dzieli swój zestaw próbek na podzbiory wzbogacony w jednej klasie, lub inne. Kryterium ląpania jest znormalizowane informacje zysk (różnica w entropii) . Atrybut z najwyższą znormalizowanego zysku informacyjnego zostanie wybrany do podjęcia decyzji. Algorytm C4.5 następnie używa rekursji na spartycjonowanych sublistach klas.

*Rekurencja w informatyce jest sposób rozwiązywania problemu, gdy rozwiązanie zależy od rozwiązań mniejszych wystąpień tego samego problemu (w przeciwnieństwie do iteracji ). Podejście można zastosować do wielu rodzajów problemów i rekurencja jest jednym z centralnych idei informatyki.*

Algorytm ten ma kilka przypadków bazowych .

- Wszystkie próbki na liście należą do tej samej klasy. Gdy tak się stanie, po prostu tworzy nowy węzeł liścia na drzewie decyzyjnym.
- Żadna z cech nie dostarcza zysku informacyjnego. W tym przypadku, C4.5 tworzy węzeł decyzyjny wyżej drzewa przy użyciu oczekiwanej wartości klasy.
- Wystąpienie wcześniej niewidzianych klas. Ponownie, C4.5 tworzy węzeł decyzyjny wyżej drzewa przy użyciu wartości oczekiwanej.

## 13. Na czym polega MapReduce?

### WERSJA 1

**Map/mapping** – zmniejszenie obserwacji np. policzenie wystąpień danego słowa

**Shuffling** – druga faza mapowania służy konsolidacji = grupowania częstości występowania danego słowa

**Reducing** – podsumowanie, agregacja danych np. zliczenie powtarzających się słów Np. policzenie średniej dla dużej liczby obserwacji, rozdzielenie to na mniejszą ilość map, agregując i sumując (reduce).

Przykład:

Welcome to Hadoop Class

Hadoop is good

Hadoop is bad

**Mapping:** Welcome 1, to 1, Hadoop 1, Class 1, Hadoop 1, is 1, good 1, etc.

**Shuffling:** Hadoop 1, Hadoop 1, Hadoop 1, etc.

**Reducing:** Hadoop 3, is 2, etc.

### WERSJA 2

MapReduce jest to framework do łatwego tworzenia programów przetwarzających duże (kilka TB) zbiory danych, gdzie przetwarzanie musi być odporne na usterki pomimo wykorzystywania „niepewnych” węzłów.

MapReduce pomaga zwiększeniu wydajności dzięki przetwarzaniu strumieniowemu (brak wyszukiwan) oraz tworzeniu potoków (pipelining).

Operacje realizowane są podczas dwóch kroków:

- Krok „map” – węzeł nadzorczy (master node) pobiera dane z wejścia i dzieli je na mniejsze podproblemy, po czym przesyła je do węzłów roboczych (ang. worker nodes). Każdy z węzłów roboczych może albo dokonać kolejnego podziału na podproblemy, albo przetworzyć problem i zwrócić odpowiedź do głównego programu.
- Krok „reduce” – główny program bierze odpowiedzi na wszystkie podproblemy i łączy je w jeden wynik – odpowiedź na główny problem.

MapReduce dzieli dane wejściowe na niezależne „kawałki”, przetwarzane równolegle przez zadania map.

Wyjście z zadań map jest sortowane przez framework i podawane na wejście zadań reduce.

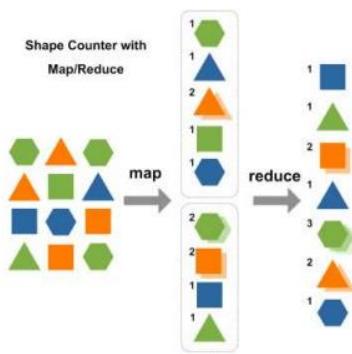
Framework zajmuje się szeregowaniem zadań, monitorowaniem i ponownym uruchamianiem w przypadku błędu.

Typowo, dane są przetwarzane na węzle, który je przechowuje – zwiększenie przepustowości.

Jeden JobTracker na węzele master i po jednym TaskTracker na węzłach slave.

### Cechy Map/Reduce:

- Małe elementarne zadania map i reduce: lepsze równoważenie obciążenia, szybszy „powrót” po błędzie/porażce.
- Automatyczne ponowne uruchamianie: niektóre węzły są zawsze powolne lub niestabilne.
- Standardowy scenariusz: te same maszyny realizują przechowywanie i przetwarzanie.
- Optymalizacja lokalna: zadania map są w miarę możliwości alokowane do maszyn przechowujących ich dane wejściowe.



## 14. Co to jest Deep Learning, podaj przykład.

### WERSJA 1

Deep Learning – obszar uczenia maszynowego, która jest obszarem sztucznej inteligencji. Technika deep learningu polega na tworzeniu sieci neuronowych, której zadaniem jest doskonalenie technik rozpoznawania głosu i obrazu. Struktura sieci neuronowych składa się z wielu warstw wejściowych, ukrytych i wyjściowych. Sieć głęboka (deep) składa się z dużej liczby neuronów, połączenie pomiędzy warstwami są bardziej skomplikowane i wymagana jest znacznie większa moc obliczeniowa.

Przykład: konwolucyjne sieci neuronowe CNN

### Wersja 2

- Deep learning stanowi część obszaru uczenia maszynowego, które z kolei jest częścią obszaru sztucznej inteligencji
- Jest to technika, która polega na tworzeniu sieci neuronowych, których głównym zadaniem jest doskonalenie technik rozpoznawania głosu i przetwarzania języka naturalnego
- Proces uczenia jest głęboki, ponieważ struktura sztucznych sieci neuronowych składa się z wielu warstw danych wejściowych, ukrytych i wyjściowych
- Przykład:
  - Konwolucyjne sieci neuronowe
  - Sprawdzają się bardzo dobrze przy rozpoznawaniu obrazów lub dźwięków

Proces uczenia jest głęboki, ponieważ struktura sztucznych sieci neuronowych składa się z wielu warstw danych wejściowych, wyjściowych i ukrytych. Każda warstwa zawiera jednostki, które przekształcają dane wejściowe w informacje, które następne warstwy mogą używać dla pewnego zadania predykcyjnego. Algorytmy uczenia głębokiego są ułożone hierarchicznie wedle rosnącej złożoności i abstrakcji. Dane muszą przejść przez kilka warstw przetwarzania.

Sieci głębokiego uczenia różnią się od „kanonicznych” sieci jednokierunkowych następującymi cechami:

- składają się z większej liczby neuronów,
- połączenia pomiędzy warstwami są bardziej skomplikowane,
- do przetrenowania wymagane są znacznie większe moce obliczeniowe,
- cechy wyodrębniane są automatyczne.

Stwierdzenie „większa liczba neuronów” oznacza, że liczba ta rośnie na przestrzeni lat, w miarę jak modele stawały się coraz bardziej skomplikowane. Warstwy również się zmieniały, od w pełni połączonych w sieciach wielowarstwowych, poprzez lokalne ciągi neuronów pomiędzy warstwami w sieciach konwolucyjnych, aż do zamkniętych połączeń z tym samym neuronem w sieciach rekurencyjnych (oprócz połączeń z poprzednią warstwą). Większa liczba połączeń oznacza, że trzeba optymalizować więcej parametrów, a to wymaga znacznie większych mocy obliczeniowych. Rozwój techniki umożliwił budowanie sieci nowej generacji, potrafiących samodzielnie, w bardziej inteligentny sposób wyodrębniać cechy. Dzięki temu za pomocą głębszych sieci można dzisiaj modelować bardzo skomplikowane procesy (np. zaawansowane rozpoznanie obrazów). Ponieważ wymagania przemysłu wciąż się zmieniają i rosną, sieci neuronowe musiały rozszerzyć swoje możliwości.

W przypadku tradycyjnego uczenia maszynowego potrzebne jest wspomaganie programisty, który musi bardzo precyzyjnie określić komputerowi, jakich cech powinien szukać przy rozpoznawaniu danego obiektu. Przewaga głębskiego uczenia polega na tym, że program samodzielnie buduje zestaw cech do rozpoznania. Nie tylko robi to szybciej, ale zazwyczaj bardziej dokładnie.

Zastosowania deep learning obejmują wszystkie rodzaje aplikacji analitycznych big data, a w szczególności te skoncentrowane na przetwarzaniu języka naturalnego (NLP – natural language processing), tłumaczeniu języków obcych, diagnostyce medycznej, transakcjach giełdowych, bezpieczeństwie sieci czy rozpoznawaniu obrazów oraz mowy. Xbox, Skype, Google Now i Siri firmy Apple® to tylko kilka przykładów marek wykorzystujących technologie głębskiego uczenia w swoich systemach do rozpoznawania ludzkiej mowy i wzorców głosowych.

#### **Przykład: Sieci CNN (Konwolucyjne sieci neuronowe)**

Przeznaczeniem sieci CNN jest wyodrębnianie z danych cech wyższego rzędu przez konwolucję. Sieci te dobrze nadają się do rozpoznawania obrazów obiektów i regularnie zwyciężają w zawodach w klasyfikowaniu obrazów. Są w stanie rozpoznawać twarze, osoby, znaki drogowe, zwierzęta i innego rodzaju obiekty. Zastosowanie sieci CNN można rozszerzyć o analizę tekstu dzięki ich możliwości rozpoznawania liter. Sprawdzają się również w analizie słów jako dyskretnych jednostek tekstowych oraz w analizie dźwięku. Skuteczność sieci CNN w rozpoznawaniu obrazów była jedną z głównych przyczyn docenienia przez świat potęgi głębskiego uczenia. Sieci CNN okazują się najbardziej przydatne w sytuacjach, gdy dane wejściowe mają określoną strukturę. Przykładem mogą być obrazy lub dźwięki zawierające powtarzające się sekwencje. Wartości w tego typu danych wejściowych tworzą przestrzenne relacje.

## **15. Jakimi cechami charakteryzują się typowe problemy Big Data?**

Ilość danych w organizacjach rośnie nieustannie. Obecne infrastruktury i architektury często niewystarczalne, by sprostać temu wyzwaniu. Informatycy są odpowiedzialni za dostarczenie odpowiedniej technologii do zarządzania technicznymi wymaganiami dla ogromnych strumieni danych. Biorąc pod uwagę cykl życia danych, można wyróżnić trzy grupy wyzwań (problemów) Big Data: problemy dot. danych, problemy dot. procesu danych i problemy dot. zarządzania danymi. W skrócie 1. grupa (problemy dot. danych) odnosi się do cech danych, 2. grupa (problemy dot. procesów danych) związana jest z technikami typu „w jaki sposób”, np. jak przechwytywać dane, jak je integrować, jak przekształcać, itp. 3. grupa obejmuje prywatność, bezpieczeństwo, zarządzanie i aspekty etyczne.

**Ww. 1. grupa wyzwań Big Data odnosi się do cech danych, do których zalicza się m.in.:**

- Ilość: ilość przechowywanych danych rośnie dramatycznie szybko. Zatem normalne jest posiadanie pojemności danych w kategoriach petabajtów (PB). Obecnie śledzimy i nagrywamy wszystko: dane środowiskowe, dane biznesowe, dane medyczne, dane z nadzoru, itp. Dlatego mamy ogromne ilości danych, których nie mogą być zarządzane przez obecne systemy tradycyjne. Dochodzi też do sytuacji, w której ilość danych w przedsiębiorstwie rośnie, a odsetek danych, które można przetwarzać, maleje. Co prowadzi do powstania „strefy ślepej”. Ta strefa wskazuje dane typu „nie wiem”, które mogą mieć ogromne znaczenie lub mogą być bezużyteczne.
- Różnorodność: ogromny przyptływ danych doprowadził również do kolejnego problemu, który stanowi dużą różnorodność danych pod względem formatów i typów. Ogromny wzrost zastosowań czujników, urządzeń inteligentnych, technologii współpracy społecznej spowodował, że oprócz danych o tradycyjnym typie występują również surowe dane, częściowo ustrukturyzowane i nieustrukturyzowane, dane zebrane ze stron internetowych, wiadomości e-mail, forów mediów społecznościowych, itd.
- Prędkość: Prędkość jest definiowana jako pomoc zdolności bieżącej aplikacji do obsługi i przetwarzania strumienia danych, który jest wytwarzany w sposób ciągły i w stałym tempie i musi zostać przeanalizowany w czasie zbliżonym do rzeczywistego. To stanowi nowe wyzwanie dla Big Data.
- Prawdziwość: Odnosi się to do stronniczości, niepewności, nieprawdy i brakujących wartości w danych. Ta funkcja mierzy precyzję danych i możliwość wykorzystania ich w analizie. Poziom poprawności zestawu danych ustali w jakim stopniu analizowane dane są ważne dla badanego problemu. Według niektórych badaczy ta cecha stanowi największe wyzwanie Big Data.
- Zmienność: Zmienność danych oznacza, jak długo dane są ważne oraz jak długo powinniśmy przechowywać je w naszych bazach danych. Obecnie świat polega na danych w czasie rzeczywistym. Określenie danych nieaktualnych i bezużytecznych w analizie jest pożądane a jednocześnie stanowi wyzwanie dla Big Data.
- Jakość: Mierzy czy dane są odpowiednie do wykorzystania w procesie podejmowania decyzji. Jakość danych może być niska lub wysoka, w zależności od parametrów: kompletności danych, poprawności danych, dostępności i ich terminowości.
- Dogmatyzm: Big Data dostarcza wielu cennych informacji, jednak nie należy polegać wyłącznie na cyfrach, zdrowy rozsądek i eksperckie porady są pożądane.

**2. grupa dotyczy problemów, jakie powstają podczas przetwarzania Big Data, rozpoczyna się krokiem przechwytywania, a kończy prezentowaniem wyników.**

**Wyzwania Big Data obejmują:**

- gromadzenie i rejestrowanie danych: Rosnąca ilość danych jest wytwarzana przez odpowiednie źródła. Ogromnym wyzwaniem Big Data jest właściwa filtracja danych, która wiąże się z zdefiniowaniem inteligentnych filtrów, które będą odróżniać to, co jest przydatne oraz to, co jest bezużyteczne. Drugim ważnym problemem Big Data jest automatyczne generowanie metadanych opisujących zarejestrowane dane oraz sposób ich rejestrowania i pomiaru. Istotne jest, by rozwijać systemy do generowania metadanych i weryfikować pochodzenie danych w różnych etapach analizy danych.
- wydobywanie i czyszczenie informacji: pozyskane dane są zazwyczaj w nieodpowiednim formacie. Stworzenie odpowiedniego procesu ekstrakcji do wydobywania właściwych informacji jest ogromnym wyzwaniem dla Big Data. Właściwe dane powinny być w standardowej i ustrukturyzowanej formie, gotowej do analizy. Ponadto Big Data może zawierać również nieprawidłowe informacje. Np. pacjenci mogą podawać błędne nazwy przyjmowanych leków, co doprowadzi do błędnej dokumentacji medycznej. W takich sytuacjach ważne jest zastosowanie technik czyszczenia danych, które służą do zweryfikowania błędów i zapewnienia jakości danych. Modele kontroli jakości są dużym wyzwaniem dla Big Data.
- integracja i agregacja danych: strumień Big Data jest heterogeniczny. Dlatego nie jest wystarczające zapisać go w repozytorium w formie zbioru zestawu danych, ponieważ znalezienie pożądanych danych i ujęcie ich w analizie byłoby bardzo utrudnione. Analiza danych to proces obejmujący znalezienie danych, ich identyfikację, zrozumienie i przywoływanie danych. Przeprowadzenie analizy danych na dużą skalę wymaga zautomatyzowania ww. kroków. Wiele zostało już zrobione, jednak wiele również zostało do zrobienia przy dane właściwie integrować.
- przetwarzanie zapytań, modelowanie i analiza danych: Big Data to dane, które są zazwyczaj chaotyczne, heterogeniczne, dynamiczne i powiązane ze sobą. Chaotyczne zbiory danych Big Data mogą być bardziej przydatne niż małe próbki danych, ponieważ ogólne statystyki mogą zostać wyodrębnione z powtarzających się wzorów. Ponadto Big Data tworzy dużą sieć połączeń heterogenicznej informacji, nadmiarowa informacja może zostać wykorzystana do wypełniania braków danych. Big Data umożliwia przeprowadzanie interaktywnych analiz danych w czasie rzeczywistym. Problemem jaki stoi przed Big Data jest opracowanie technik przetwarzania zapytań, które poradzą sobie ze złożonością skalowania terabajtów Big Data oraz umożliwią interaktywny czas reakcji.
- Interpretacja: Przeprowadzona analiza, która jest niezrozumiała jest bezwartościowa. Użytkownik końcowy powinien zrozumieć i zweryfikować wyniki wygenerowane przez system komputerowy, a ten z kolei powinien ułatwić ich odczyt dla użytkownika. Jednakże to z powodu złożoności Big Data, to stanowi wyzwanie. Podsumowując, podanie samych wyników analizy jest niewystarczające, ważne jest by opisać przeprowadzone kroki oraz użyte dane do analizy. Systemy BI stosujące rozbudowane techniki wizualizacyjne w prosty i zrozumiały sposób obrazują wyniki analizy, dzięki czemu użytkownik właściwie dokonuje interpretacji.

**3. grupa dotyczy problemów związanych z zarządzaniem:**

- Prywatność: Prywatność stanowi poważny problem, szczególnie w kontekście Big Data. Na przykład w sektorze zdrowia istnieją już przepisy które regulują prywatność pacjentów.

Obawa przed niewłaściwym wykorzystaniem danych rośnie nieustannie, a szczególnie w sytuacji, gdy dane są łączone z różnych źródeł.

- Bezpieczeństwo: Coraz więcej firm buduje duże środowiska do przechowywania danych, ich agregowania i analizowania. Wynika to z faktu, że Big Data pomaga biznesom dopasować swoje produkty i usługi do potrzeb klientów. W efekcie ilość dużych repozytoriów wzrosła i jest w zainteresowaniu grup przestępcoch dających do uzyskania dostępu do nich i osiągnięcia dużych korzyści.
- Rządy: zarządzanie dużymi repozytoriami danych jest ogromnym wyzwaniem, a jednocześnie dużą odpowiedzialnością. Big Data to poufne dane, do których powinny mieć dostęp tylko osoby uprawnione, a jednocześnie powinny zawierać dane jakościowe, które zostaną wykorzystane do przeprowadzenia analiz, na podstawie, których trafna interpretacja zostanie zdefiniowana.

## 16. Omów przykładowe techniki stosowane w rozpoznawaniu wzorców.

### Rozpoznawanie wzorców

- Statystyka opisowa
- Analiza skupień
- Sieci samoorganizujące się
- Analiza asocjacji i sekwencji

Grupowanie pod nadzorem	Grupowanie bez nadzoru
Dana jest jednoznacznie określona zmienna celu	Nie istnieje jednoznacznie określona zmienna celu. Algorytm eksploracji danych poszukuje wzorców i struktur wśród wszystkich zmiennych
<ul style="list-style-type: none"><li>• Sztuczne sieci nerowne</li></ul>	<ul style="list-style-type: none"><li>• Drzewa decyzyjne</li><li>• Regresja logistyczna</li><li>• Wielowarstwowy perceptron MLP</li><li>• Metoda <math>k</math>-najbliższych sąsiadów</li><li>• SVM</li></ul>

- 1. Metody hierarchiczne:**
  - Skupienia tworzą drzewa binarne i w ten sposób uzyskiwana jest hierarchia tj. jedne skupienia są zawarte w drugich.
  - Uwzględniając kryterium rozpoczęcia procesu grupowania wyróżniamy: metody aglomeracyjne i metody podziałowe.
  - Ze względu na sposób wyznaczania odległości między skupieniami najczęściej spotykane metody aglomeracyjne to: najbliższego sąsiedztwa, najbliższego sąsiedztwa, mediany, środka ciężkości, średniej odległości wewnętrz skupień, średniej odległości między skupieniami, minimalnej wariancji Warda.
- 2. Metody optymalizacyjno-iteracyjne:**
  - Wymagają wstępniego podziału zbioru obiektów na określona liczbę podzbiorów. Wybrany sposób podziału jest iteracyjnie modyfikowany. Np. metoda  $k$ -średnich.
- 3. Metody obszarowe:**
  - Przestrzeń grupowania jest dzielona na rozłączne obszary a obiekty znajdujące się w otrzymanych obszarach tworzą grupy.
- 4. Inne metody**

## Grupowanie – analiza skupień

**Grupa, skupienie (cluster)** – zbiór obiektów, które są podobne do siebie nawzajem i niepodobne do obiektów z innych grup.

**Grupowanie (clustering)** – oznacza grupowanie obserwacji (rekordów) w klasy (grupy) podobnych obiektów.

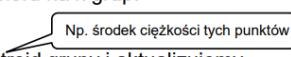
Zmienność pomiędzy grupami (*between-cluster variation, BCV*) ↑

Zmienność wewnętrz grupy (*within-cluster variation, WCV*) ↓

**Analiza skupień (cluster analysis)** – grupowanie bez nadzoru, ma za cel wykrycie w zbiorze obserwacji skupień, czyli grup.

Grupowanie często stanowi wstęp do dalszej analizy danych.

## Algorytm $k$ -średnich

1. Określamy na ile grup ( $k$ ) zbiór danych ma zostać podzielony.
2. Losowo wybieramy  $k$  rekordów (obserwacji) jako początkowe środki grup.
3. Dla każdego rekordu znajdujemy najbliższy środek grupy, wyznaczając w ten sposób podział zbioru na  $k$  grup.  


Np. środek ciężkości tych punktów
4. Dla każdej z  $k$  grup wyznaczamy centroid grupy i aktualizujemy położenie każdego środka grupy jako nową wartość centroidu.
5. Powtarzamy punkty od 3 do 5 aż do osiągnięcia określonego warunku stopu.

## Algorytm $k$ -średnich – warunek stopu

- Algorytm  $k$ -średnich kończy działanie, gdy centroydy już się nie zmieniają.
- Algorytm może skończyć działanie, gdy zostanie spełnione pewne kryterium zbieżności, np. brak istotnego zmniejszania sumarycznego błędu kwadratowego:

$$SSE = \sum_{i=1}^k \sum_{p \in C_i} d^2(p, m_i)$$

$C_i$  – grupy

$m_i$  – centroid i-tej grupy

$p \in C_i$  – punkt danych z i-tej grupy

## Metoda SOM – sieci Kohonena

**Sieci samoorganizujące** (*Self Organization Maps*, SOM) stanowią klasę sieci neuronowych bez warstwy ukrytej.

Szczególnym przypadkiem sieci samoorganizujących są **sieci Kohonena**.

Sieci samoorganizujące:

- Przekształcają złożone, wielowymiarowe sygnały wejściowe w prostsze, mniej wymiarowe przestrzenie ich cech charakterystycznych.
- Neurony położone bliżej siebie są bardziej podobne do siebie, niż do innych neuronów znajdujących się dalej.

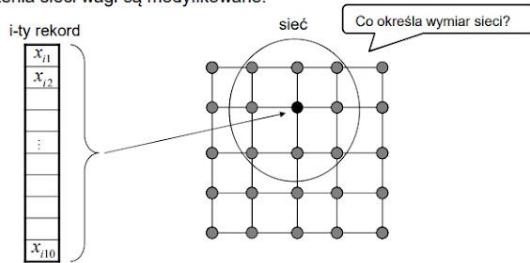
### Procesy sieci samoorganizujących

1. **Rywalizacja:** neurony wyjściowe rywalizują ze sobą, by uzyskać najlepszą wartość odległości np. euklidesowej. Węzeł wyjściowy z najmniejszą odlegością euklidesową pomiędzy danymi wejściowymi a wagami zostaje zwycięzcą.
2. **Współdziałanie:** pobudzany jest neuron wygrywający oraz inne neurony z nim sąsiadujące.
3. **Adaptacja:** wszystkie neurony z sąsiedztwa neuronu wygrywającego uczestniczą w adaptacji czyli uczeniu. Wagi ich są tak dopasowywane, aby neurony te miały większe szanse na ponowne wygranie rywalizacji w przypadku podobnego rekordu.

## Sieci Kohonena - przykład

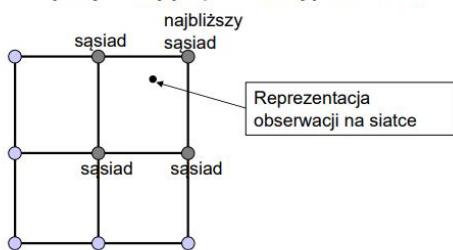
### Przykład 2

- Rozważmy zbiór danych, w którym każdy rekord (obserwacja) jest opisany przez 10 zmiennych, zatem mamy dziesięciowymiarowy wektor wejściowy.
- Załóżmy, że chcemy użyć sieci Kohonena o rozmiarze  $5 \times 5$ .
- Każdy neuron jest również opisany przez dziesięciowymiarowy wektor wag.
- W procesie uczenia sieci wagą są modyfikowane.



17

- Z wykorzystaniem odpowiedniego odwzorowania każda obserwacja otrzymuje swoją reprezentację w siatce neuronów (punktów).



- Ponieważ każda kratka w siatce odpowiada jednemu skupieniu, zatem liczba obserwacji (rekordów) musi być większa niż liczba zadeklarowanych węzłów w siatce.

## Sieci Kohonena

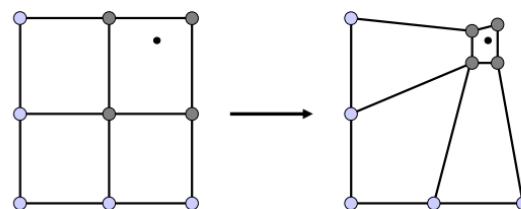
Każdy neuron jest zdefiniowany przez wektor wag i lokalizację w siatce.

W procesie budowy sieci wagi neuronów są modyfikowane.

Wektory wag podążają w stronę punktów centralnych skupień danych.

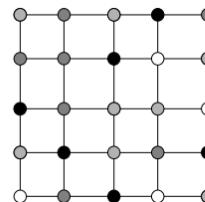
Obszar utworzony przez neurony, w którym znajduje się dana obserwacja jest następnie zacieśniany.

Proces jest kontynuowany dla wszystkich pozostałych obserwacji.



## Sieci Kohonena

- Ostatecznie otrzymujemy siatkę z neuronami odzwierciedlającymi skupienia.
- Sąsiadujące neurony reprezentują grupy rekordów o zbliżonych cechach.
- Na rysunku większe natężenie koloru odpowiada większej liczności danego skupienia.



## 17. Na czym polega przetwarzanie rozproszone?

### WERSJA 1

Przetwarzanie rozproszone to system, którego składniki znajdują się na różnych komputerach podłączonych do sieci. Główny problem to łączenie danych odczytywanych z różnych źródeł. Rozwiązaniem jest Hadoop – otwarta platforma przeznaczona do

rozproszonego składowania i przetwarzania wielkich zbiorów danych przy pomocy klastrów. Łączenie danych odbywa się przy pomocy MapReduce. Kilka serwerów na raz wykonuje operacje równolegle np. usunięcie profilu użytkownika na FB – będzie usuwał informację z wielu miejsc na raz, nie jest ważna kolejność wykonywania. Jest to zapytanie do wielu serwerów na raz. Innym przykładem są urządzenia mobilne.

Przetwarzanie rozproszone to dziedzina informatyki, która bada systemy rozproszone. System rozproszony to system, którego składniki znajdują się na różnych komputerach podłączonych do sieci (inaczej węzły – node), które komunikują się i koordynują swoje działania poprzez przekazywanie sobie nawzajem komunikatów. Komponenty te współpracują ze sobą w celu osiągnięcia wspólnego celu. Trzy istotne cechy systemów rozproszonych to: współprzeźność składowych, asynchroniczność i niezależna awaria składowych.

W systemach asynchronicznych, poszczególne węzły wykonują operacje z różnymi prędkościami w takt niezależnych zegarów, a czas transmisji wiadomości jest skończony lecz nieznany.

W obliczeniach rozproszonych każdy procesor ma swoją własną pamięć prywatną (pamięć rozproszoną). Wymiana informacji odbywa się poprzez przekazywanie komunikatów pomiędzy procesorami.

Głównym problemem jest łączenie danych odczytywanych z różnych źródeł. Wszystkie problemy są łatwo rozwijywane przez Hadoopa - otwartą platformę programistyczną napisaną w języku Java przeznaczoną do rozproszonego składowania i przetwarzania wielkich zbiorów danych przy pomocy klastrów komputerowych.

Problem przetwarzania danych jest obsługiwany przez system plików Hadoop Distributed File System (HDFS), a problem łączenia danych jest obsługiwany przez paradymat MapReduce. MapReduce zasadniczo zmniejsza problem odczytu i zapisu danych z dysku, dostarczając model programowania zajmujący się obliczaniem kluczów i wartości. Hadoop zapewnia w ten sposób: niezawodny, współdzielony system magazynowania i analizy. Magazynowanie jest obsługiwane przez HDFS, a analiza przez MapReduce.

Map Reduce jest to algorytm służący przetwarzaniu równoleglemu dużych zbiorów danych w rozproszonym środowisku. Podejście opracowane przez firmę Google. Algorytm składa się z dwóch głównych kroków:

- map – pobranie danych z wejścia i ich podział na podzbiory. Dekompozycja problemu na podproblemy.
- reduce – zgromadzenie odpowiedzi, ich połączenie i przekazanie wyniku

Programowanie rozproszone zazwyczaj zalicza się do jednej z kilku podstawowych architektur, np. peer-to-peer:

Peer-to-peer: architektura, w której nie ma specjalnych maszyn świadczących usługi lub zarządzających zasobami sieciowymi. Zamiast tego wszystkie zadania są jednolicie podzielone pomiędzy wszystkie maszyny, znane jako peer-to-peer i mogą służyć zarówno jako klienci, jak i serwery. Przykładami tej architektury są BitTorrent i sieć Bitcoin.

## **18. Omów wybraną metodykę opisującą sposób realizacji procesu twórczego modelu analitycznego.**

## ***The Cross-Industry Standard Process for Data Mining (CRISP-DM)***

### **Sześć faz CRISP-DM:**

- 1. Zrozumienie problemu biznesowego (*Business understanding phase*)** – określenie celów projektu, wyrażenie ich w języku problemów data mining, określenie wstępnej strategii osiągnięcia tych celów.
- 2. Poznanie danych (*Data understanding phase*)** – zbieranie danych, wykorzystanie prostych metod analizy danych do zapoznania się z danymi, ocena jakości danych, ewentualne wstępne określenie podzbiorów danych, które mogą zawierać informacje prowadzące do ważnych prawidłowości.
- 3. Przygotowanie danych (*Data preparation phase*)** – przygotowanie wstępnego oraz ostatecznego zbioru danych, wybór zmiennych i obiektów do analizy, ewentualna analiza niektórych zmiennych, czyszczenie danych.

## ***The Cross-Industry Standard Process for Data Mining (CRISP-DM)***

- 4. Modelowanie (*Modeling phase*)** – wybór technik modelowania, budowa modelu.
- 5. Ocena (*Evaluation phase*)** – ocena zbudowanych modeli pod względem poziomu dopasowania, efektywności, interpretowalności, użyteczności w realizacji celów projektu; określenie elementów, których znaczenie nie zostało uwzględnione, wstępne określenie możliwości wdrożenia wyników w praktyce.
- 6. Wdrożenie (*Deployment phase*)** – przygotowanie raportu, wykorzystanie modeli, zastosowanie modelu do podobnego zagadnienia lub innych obiektów, ocena efektów biznesowych.

## **Metodologia SEMMA**

Proces analizy danych składa się z kilku etapów:

- |                                       |                |
|---------------------------------------|----------------|
| • Próbkowanie                         | <b>SAMPLE</b>  |
| • Eksploracja danych                  | <b>EXPLORE</b> |
| • Modyfikacja danych                  | <b>MODIFY</b>  |
| • Budowa modelu                       | <b>MODEL</b>   |
| • Ocena skuteczności i jakości modelu | <b>ASSESS</b>  |

Pięć przedstawionych etapów składa się na metodologię SEMMA, zgodnie z którą zbudowane jest środowisko analityczne **SAS Enterprise Miner**.

- 19.** Wymień kluczowe założenia będące warunkami zastosowania modeli predykcyjnych do wspomagania procesów decyzyjnych.

Modele predykcyjne mają na celu poprawne przewidywanie przyszłych (nowych) obserwacji na podstawie modelu istniejącego. Dużą rolę odgrywa w nich wykorzystywanie prób uczących (budowa modelu) i prób testowych (predykcja nowych obserwacji).

Modele predykcyjne powinny cechować się wysokim poziomem konfirmacji i posiadać wysoką trafność przewidywania.

- mogą występować braki danych - drzewa decyzyjne; (możliwe jest uzupełnienie braków danych)
- dostępna ogromna ilość danych – setki tysięcy wierszy (obserwacji) oraz setki, a nawet tysiące kolumn (zmiennych). Jest możliwość pre-selekcji danych zmiennych jak jest dużo, i próbkowanie gdy bardzo dużo obserwacji. Segregacja danych.
- musi być odpowiednia ilość zmiennych objaśniających (nie może być jedna zmienna x i y, tylko powinno być kilka zmiennych x)
- najczęściej stosuje się zmienna ciągła przypominającą rozkład normalny (zakłada się że jak jest dużo zmiennych to one naturalne dążą do rozkładu normalnego)
- nie stosuje się wartości ujemne
- nie może być dużo wartości odstających

**Założenia:**

- 1) Rozkład danych rzeczywistych zbliżony do rozkładu danych treningowych – problem z brakami danych (nie można nagle odrzucić połowy obserwacji)
- 2) Relacja zachowana na przestrzeni czasu (dla przyszłych danych będzie zachodzić tak samo)
- 3) Odpowiednie określenie funkcji celu – w celu maksymalizacji zysku

Modele predykcyjne mają na celu przewidywanie zmiennych zależnych na podstawie zmiennych opisujących dany problem. Aby móc przystąpić do modelowania predykcyjnego należy spełnić założenia:

- o Zdefiniowane modelowanego problemu oraz zrozumienie go
- o Wymagana jest odpowiednio duża próba danych, aby móc na nich podstawie budować model
- o Dane włączane do modelu muszą być zapisane w odpowiedniej, ustrukturyzowanej formie

## **20. Jak mierzymy jakość modelu prognostycznego?**

W prognozowaniu najistotniejszym zagadnieniem jest skuteczność modelu, który powinien z możliwie jak najmniejszym błędem estymować zmienność celu. Jeżeli sprawdzamy błąd predykcji na zbiorze już zaobserwowanym, to mamy do czynienia z prognozą ex post. W ocenie ex post obliczamy różnicę pomiędzy wartością zaobserwowaną, a prognozowaną. Natomiast jeżeli liczymy błąd na próbkach spoza próbki uczącej mamy do czynienia z oceną ex ante.

#### Współczynnik determinacji R<sup>2</sup>

Jest to miara jakości dopasowania modelu do danych. Mówiąc o tym jaki procent zmienności zmiennej celu jest objaśniany poprzez predykatory (zmienne objaśniające). R<sup>2</sup> informuje nas jaka część wariancji zmiennej zależnej w próbie jest wyjaśniana przez zmienne zawarte w modelu. Miara ta wyraża się wzorem:

$$R^2 = \frac{\sum_{t=1}^n (\hat{y}_t - \bar{y})^2}{\sum_{t=1}^n (y_t - \bar{y})^2},$$

gdzie  $\hat{y}_t$  jest zaobserwowaną wartością zmiennej objaśnianej,  $\bar{y}$  jest estymowaną wartością zmiennej objaśnianej, a  $\bar{y}$  jest średnia arytmetyczna zaobserwowanych zmiennych celu.

Współczynnik ten przyjmuje wartości od 0 do 1, gdzie 0 oznacza brak dopasowania, a 1 dopasowanie idealne, które często świadczy o nadmiernym dopasowaniu i może być powodem przeuczenia modelu. W przypadku gdy mamy do czynienia z więcej niż tylko jednym predyktorem, stosuję się skorygowane R<sup>2</sup>. W przypadku gdy będziemy dodawać kolejne zmienne niezależne wartość R<sup>2</sup> rośnie zawsze, nawet gdy ten predyktor nie wyjaśnia wariancji zmiennej celu.

#### Błąd średniookwadratowy MSE (ang. Mean Squared Error):

$$MSE = \frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2$$

#### Średni błąd bezwzględny MAE (ang. Mean Absolute Error).

Mierzy średnią wielkość błędu dla predykcji bez uwzględnienia kierunku błędu. Dla próbki testowej jest to średnia arytmetyczna bezwzględnych różnic pomiędzy wartością zaobserwowaną, a wartością estymowaną, gdzie poszczególne różnice posiadają te same wagę. Miarę średniego błędu bezwzględnego można zapisać poniższym wzorem:

$$RMSE = \sqrt{\frac{1}{n} \sum_{t=1}^n (y_t - \hat{y}_t)^2}.$$

Obie miary MAE oraz RMSE pozwalają na wyznaczenie średniego błędu predykcji w jednostkach zmiennej celu. Wartości błędu są obojętne na kierunek oraz mieszczą się w zakresie od 0 do  $\infty$ . Dla obu miar w przypadku porównywania różnych modeli predykcyjnych istotne jest to, aby była jak najmniejsza.

Istotną różnicą pomiędzy tymi dwiema miarami jest to, że RMSE przydziela większe wagi dużym błędom ze względu na potęgowanie, które występuje we wzorze. RMSE jest szczególnie użyteczne w przypadku dużych błędów dla obserwacji odstających. Ze względu na to, że RMSE wzrasta wraz ze wzrostem wariancji rozkładu częstotliwości błędu, podczas gdy MAE jest stabilny. RMSE przeważnie jest większe niż MAE i ma tendencję do zwiększenia błędu w momencie zwiększania liczby obserwacji.

#### Średni bezwzględny błąd procentowy MAPE (ang. Mean Absolute Percentage Error)

Jest to miara, która opisuje procentową dokładność modelu progностycznego poprzez obliczenie sumy ilorazu wartości MAE i wartości zaobserwowanej. Podobnie jak MAE nie uwzględnia kierunku błędu. Miarę MAPE można zapisać poniższym wzorem:

$$MAPE = \frac{1}{n} \sum_{t=1}^n \left| \frac{y_t - \hat{y}_t}{y_t} \right| = \frac{1}{n} \sum_{t=1}^n \frac{MAE}{|y_t|}.$$

Wartość miliary średniego bezwzględnego błędu procentowego jest z zakresu od 0% do 100%. Czym mniejsza wartość tej miliary tym model lepiej przewiduje zmiennej celu.

#### Wyjaśniana wariancja (ang. Explained Variation).

Miara objaśnianej wariancji mierzy proporcję dla jakiej model predykcyjny uwzględnia zmienność danych. Miarę objaśnianej wariancji można zapisać jako różnica idealnej predykcji równej jeden i stosunku wariancji błędu (różnicy wartości oczekiwanej od estymowanej) do wariancji wartości oczekiwanej. Powyższą miarę można zapisać wzorem:

$$\text{Explained variance}(y, \hat{y}) = 1 - \frac{\text{Var}\{y - \hat{y}\}}{\text{Var}\{y\}}.$$

Jeżeli wartość objaśnianej wariancji jest bliższa jedności oznacza to, że model dobrze wyjaśnia zmienność zmiennej zależnej.

## 21. Jak mierzmy jakość modelu progностycznego?

W zależności od tego jaki mamy model możemy użyć inne miary do badania jakości modelu progностycznego.

Dla modeli klasyfikacyjnych, w których zmienną Y jest zmienna jakościowa to:

- Accuracy
- Specificity
- Sensitivity
- AUC (Area under the curve)
- Confusion matrix

Dla modeli regresyjnych, w których zmienną Y jest zmienna ilościowa to:

- Błąd średniokwadratowy (Mean square error)
- Średni błąd bezwzględny (Mean absolute error)
- Pierwiastek błędu kwadratowego (Root mean square error)
- Współczynnik determinancji R<sup>2</sup>

1) **Accuracy** (ułamek poprawnych przewidywań)

$$\text{Accuracy} = \frac{\text{liczba poprawnych predykcji}}{\text{wszystkie predykcje}} = \frac{TP + TN}{TP + TN + FP + FN}$$

2) **Sensitivity** (wrażliwość) – przewiduje wszystkie true positive

$$\text{Sensitivity} = \frac{TP}{TP + FN}$$

3) **Specificity** (specyficzność) – zdolność do przewidzenia true negatives

$$\text{Specificity} = \frac{TN}{TN + FP}$$

4) **AUC**

5) **Współczynnik determinancji R<sup>2</sup>** – jest to miara jakości dopasowania danych do modelu. Mówiąc o tym jaki % zmienności zmiennej celu jest wyjaśnianych poprzez zmienne objaśniające.

$$R^2 = \frac{\sum_{t=1}^n (\hat{y}_t - \bar{y}_t)^2}{\sum_{t=1}^n (y_t - \bar{y}_t)^2}$$

$y_t$  – wartość zmiennej objaśnianej

$\hat{y}_t$  – wartość estymowana zmiennej objaśnianej

$\bar{y}_t$  – średnia arytmetyczna zaobserwowanych zmiennych wyjaśnianych

Zakres od 0 do 1. Dla 0 brak dopasowania danych do modelu. Dla 1 idealne dopasowanie, model może być przeuczony.

## **21.** Omów w jaki sposób wykorzystanie systemu kontroli wersji wpływa na efektywność procesu twórczego rozwiązań analitycznych.

Korzystanie z systemu kontroli wersji (VCS ang. Version Control Systems) w projektach ma wiele zalet.

### **Kolaboracja**

Dzięki VCS wszyscy członkowie zespołu mogą pracować całkowicie swobodnie - na dowolnym pliku w dowolnym momencie. VCS pozwoli później scalić wszystkie zmiany we wspólną wersję. Nie ma wątpliwości, gdzie jest najnowsza wersja pliku lub całego projektu. Jest to wspólne, centralne miejsce: system kontroli wersji.

### **Przechowywanie wersji**

System kontroli wersji potwierdza, że istnieje tylko jeden projekt. Dlatego na dysku jest tylko jedna wersja, nad którą aktualnie trwa praca. Cała reszta - wszystkie poprzednie wersje i warianty - są starannie zapakowane w VCS. Gdy jest to potrzebne, można odtworzyć dowolną wersję w dowolnym momencie.

### **Przywracanie poprzednich wersji**

Jeśli ostatnio wprowadzone zmiany okażą się złe, można je po prostu cofnąć za pomocą kilku kliknięć. Wiedząc to, można pracować nad ważnymi elementami projektu bez obaw i stresu, że zmiany są nieodwracalne lub trudne do cofnięcia.

### **Zrozumienie zmian w projekcie**

Każda zmiana wymaga krótkiego opisu tego, co zostało zmienione. Dodatkowo (jeśli jest to kod / plik tekstowy), można zobaczyć, co dokładnie zostało zmienione w treści pliku. Pomaga to zrozumieć, w jaki sposób projekt ewoluował między wersjami.

### **Tworzenie kopii zapasowej**

Efektem ubocznym używania rozproszonego VCS, takiego jak Git, jest to, że może on działać jako kopia zapasowa; każdy członek zespołu ma na swoim dysku pełną wersję projektu - w tym pełną historię projektu. Jeśli serwer się zepsuje (a dyski zapasowe zawiodą), wszystko, czego potrzebujesz do odzyskania, to jedno z lokalnych repozytoriów Git.

## **22.** Wyjaśnij co to jest reprodukowalność procesu analitycznego i dlaczego jest ona ważna w praktyce gospodarczej.

### **Wysiął ją co to jest reprodukowalność procesu analitycznego**

"Reroduktywialność odnosi się do zdolności badacza do powielania wyników wcześniejszych badań z wykorzystaniem materiałów podobnych do tych, które zostały użyte przez pierwotnego badacza. Oznacza to, że drugi badacz może wykorzystać te same surowe dane do budowania tych samych plików analizy i wdrożenia tej samej analizy statystycznej w celu uzyskania takich samych wyników ... Powtarzalność jest minimalnym warunkiem koniecznym, aby odkrycie było wiarygodne i zawierało informacje." [Steven N. Goodman, 2016]

[National Science Foundation, 2015]

REPRODUKTYWALNOŚĆ odnosi się do zdolności badacza do powielania wyników wcześniejszych badań przy użyciu tych samych materiałów i procedur, które zostały zastosowane przez pierwotnego badacza

Np. Badacz wykorzystuje te same surowe dane, tworzy te same pliki analizy i te same procedury statystyczne, aby upewnić się, że takie same wyniki jak w opublikowanym badaniu

Różnice mogą wynikać z:

- Przetwarzania (np. brakujących danych) danych
- Zastosowania metody statystycznej (np. różnych wartości domyślnych)
- Przypadkowych błędów w pierwotnej analizie (lub analizie następcej)

Odtwarzalność jest minimalnym niezbędnym warunkiem, aby wniski były wiarygodne i zawierały informacje.

### **Dlaczego jest ona ważna w praktyce gospodarczej?**

Dzisiejsze nauki - zwłaszcza nauki społeczne - są trochę zawirowane. Wiele najważniejszych eksperymentów i ustaleń nie jest powtarzalnych. Ten „kryzys odtwarzalności” ma znaczące implikacje nie tylko dla przyszłości badań naukowych i rozwoju, ale dla każdej firmy oczekującej wzrostu zysków z inwestycji w innowacje, eksperymenty i analizę danych. Biznes musi uczyć się na błędach nauki.

Jako wiceprezes ds. Badań w Arnold Ventures mam ścisłą wiedzę na temat tego trwającego kryzysu, ponieważ sfinansowałem sporą z tych działań „drugiego spojrzenia”. Oto nieszczęśliwa próbka tego, co sfinansowaliśmy i znaleźliśmy:

W 2015 r. Science opublikowało wyniki największego projektu replikacji, jaki kiedykolwiek przeprowadzono: Reproducibility Project in Psychology, w którym setki badaczy z całego świata podjęły próbę powtórzenia 100 eksperymentów psychologicznych z najlepszych czasopism. Tylko około 40% wyników można było z powodzeniem powielić, podczas gdy pozostałe były niejednoznaczne lub ostatecznie nie zostały powtórzone.

**Reprodukowalność procesu** to możliwość powtórzenia procesu w celu uzyskania takich samych wyników na podstawie napisanego kodu i sposobu wykonywania obliczeń. Ważne, aby analiza danych mogła być reprodukowalna. Jest to zdolność badacza do powielenia wcześniejszych wyników na podstawie danych surowych (jeszcze nieprzetworzonych).

Jakie są zalety z powtarzalności?

- Wzrost prawdopodobieństwa, że badania zostały wykonane prawidłowo
- Łatwiej jest rozszerzyć badanie
- Wzrost dokładności badania

## **23. Omów podstawowe sposoby zapewnienia reprodukowalności procesu analitycznego.**

Wymagania reprodukowalności procesu:

- Dostęp do surowych danych (przed przetworzeniem, czyszczeniem i transformacją)
- Dostęp do zestawu instrukcji wyjaśniającej etapy przetwarzania danych i analizę danych + kod
- Informacje o systemie operacyjnym, pakietach i bibliotekach potrzebnych do obliczeń
- Ziarno analizy, jeśli polegam na algorytmie losowym – muszą być te same dane treningowe i testowe

To", co należy odtworzyć, to zazwyczaj:

Rzeczywiste wyniki, które obejmują:

- Tabele
- Wizualizacje / liczby / wykresy
- Wartości zgłoszone w tekście

Dowody statystyczne potwierdzające ustalenia (np. p-values, przedziały ufności, wiarygodne przedziały).

**Wymagania dotyczące wykazania reproduktywności**

Panuje powszechna zgoda co do tego, że badania można odtwarzać tylko wtedy, gdy:

- Dostępne są „surowe” dane, przy czym „surowe” odnosi się do danych przed jakąkolwiek manipulacją przez badacza (np. przed czyszczeniem i transformacją danych).
- Dostępny jest pełny zestaw instrukcji wyjaśniających wszystkie etapy przetwarzania i analizy danych.

W praktyce, gdy organizacje (np. Wydawcy czasopism) wymagają, aby badania były odtwarzalne, wprowadzą niektóre lub wszystkie z następujących dodatkowych wymagań:

- Dostarczony jest zestaw plików zawierający dane i kod. Możliwe jest tworzenie tabel i dowolnych wykresów / grafik / wizualizacji pochodzących z danych poprzez uruchomienie kodu.
- Szczegółowe informacje na temat systemu używanego do uruchomienia analizy: system operacyjny, lataki, nasiona liczb losowych, określone wersje wszystkich programów / pakietów / bibliotek.
- Kod jest napisany w sposób, który można łatwo zrozumieć.
- Otwartość / transparentność. Wszystkie dane i materiały są dostępne (w przeciwnieństwie do „dostępnych na żądanie”) - np. opublikowane na GitHub lub w międzynarodowym repozytorium danych.

To znaczy:

- Inną stroną (np. Recenzent) z powodzeniem odtworzyła wyniki i certyfikowała je jako takie.
- "Logs" pokazują, że kluczowe wyniki zostały pomyślnie utworzone na podstawie danych wejściowych.
- Kluczowe wyniki są powiązane z danymi i kodem, dzięki czemu można bezpośrednio sprawdzić relacje.

**24.** Wyjaśnij co to jest próg odcięcia w modelach klasifykacyjnych oraz omów od czego zależy jego optymalna wartość w przypadku wykorzystania takiego modelu do wspomagania podejmowania decyzji.

**Próg odcięcia stosuje się w modelach klasyfikacyjnych takich jak regresja logistyczna** gdzie wynikiem są wartości ciągłe do zdecydowania **czy dany wynik należy do pewnej klasy lub nie**. Jeśli wynikiem modelu jest prawdopodobieństwo przynależności do pewnej klasy to próg można zastosować do uzyskania dyskretnego (binarnego) klasyfikatora: jeśli wynik klasyfikatora jest powyżej ustalonego progu, klasyfikator wytwarza P, w przeciwnym razie F.

Wybór optymalny próg odcięcia będzie zależeć przed wszystkim od typu problemu który chcemy rozwiązać ponieważ sklasyfikowanie wyniku do niepoprawnej klasy może skutkować wysokim kosztem (np. nie zdiagnozowanie choroby) lub utratą potencjalnych korzyści (np. nie udzielenie kredytu). Typ problemu wpłynie w dużej mierze na wybraną metrykę do mierzenia błędów modelu i od tego jaka jest jego maksymalna akceptowana wartość błędu.

Bardzo często do zdecydowania optymalnego progu odcięcia w środowisku uczenia maszynowego stosuje się krzywe ROC, po części ze względu na fakt, że precyza (accuracy) klasyfikacji jest często słabym miernikiem do pomiaru wydajności modelu. Krzywa ROC posiadają właściwości, które czynią ją szczególnie przydatnym w domenach o skośnym rozkładzie klas i nierównych kosztach błędu klasyfikacji. Cechy te stają się coraz ważniejsze w miarę kontynuowania badań w obszarach uczenia się wrażliwego na koszty i uczenia się w obecności niezrównoważonych klas.

Krzywa ROC pozwala nam na przedstawienie informacji zawartej w macierzy błędów (ang. Confusion Matrix) dla każdego progu odcięcia i składa się z:

		Klasa prawdziwa	
		1	0
Klasa prognozowana	1	True positive (TP)	False positive (FP)
	0	False negative (FN)	True negative (TN)

- oś Y wykresu wskazuje proporcje True Positive Rate (TPR, czułość) czyli obserwacje poprawnie zidentyfikowane jako pozytywne
- oś X wskazuje proporcje False Positive Rate (TNR, 1 - Specyficzność) obserwacje negatywne które zostały nie poprawnie sklasyfikowane

A więc wykres ROC przedstawia wzajemne kompromisy między korzyściami (True Positive Rate) i kosztami (False Positive Rate). Aby stworzyć krzywe ROC, musielibyśmy wielokrotnie ocenić model z różnymi programami odcięcia i połączyć otrzymane punkt.

Jakość klasyfikacji różnych modeli można porównać za pomocą krzywej ROC, wyliczając takie wskaźniki jak pole pod krzywą (AUC) (Area Under ROC Curve). AUC mierzy cały dwuwymiarowy obszar pod krzywą ROC i stanowi zagregowaną miarą wydajności wszystkich możliwych progów klasyfikacyjnych. Jednym ze sposobów interpretacji AUC jest prawdopodobieństwo, że model sklasyfikuje poprawnie przypadkowy przykład pozytywny jako pozytywny niż jako negatywny.

**Próg odcięcia (ang. cut – off)** – stosuje się do klasyfikowania obserwacji do odpowiednich klas wybór optymalnego progu odcięcia będzie zależeć od typu problemu.

Jaki jest koszt błędów?

Taki próg, który maksymalizuje funkcję zysku (maksymalizuje akceptowalną wartość błędu).

$$np \cdot \text{zysk}(p) = 100TP - 10FP$$

Próg odcięcia to wartość prawdopodobieństwa, powyżej którego model przypisze zajście zdarzenia, a poniżej którego przypisze brak wystąpienia zdarzenia. Jako punkt odcięcia często przyjmuje się liczbę równą 0,5 dla rozkładów równomiernych.

Do wyboru optymalnego progu odcięcia można stosować krzywe ROC, która przedstawia zależność pomiędzy sensitivity (wrażliwość), a specificity (specyficzność). AUC to pole pod krzywą ROC. Im większa wartość AUC tym lepiej jest dopasowany model.

## 25. Wyjaśnij do czego wykorzystywana jest regularyzacja w procesie budowy modeli predykcyjnych.

27. Wyjaśnij do czego jest wykorzystywana regularyzacja w procesie budowy modeli predykcyjnych. (+ podać może przykład)

**Regularyzacja** – nadanie ograniczeń wariancji modelu poprzez ograniczenie wartości współczynników modelu predykcyjnego. Regularyzacja to jeden ze sposobów zapobiegania przeuczenia modelu tzw. overfitting, czyli nadmierнемu dopasowaniu modelu.

Przykłady regularyzacji:

### lasso w regresji liniowej

- Zawiera sumę modułów wag jako składnik kary
- Wagi (bety) mogą się zerować przy odpowiednio dużych wartościach lambda
- Można pozbyć się nieistotnych parametrów z modelu
- Ograniczenie: bety nie mogą być większe od zadanej wartości

### ridge regression (regresja grzebietowa)

- Zawiera sumę kwadratów wag jako składnik kary
- Wielkość próby jest niewielka
- Wartości parametrów mogą zmierzać asymptotycznie do zera
- Zmniejszenie wrażliwości prognoz na danych treningowych

Niepoprawnie wytrenowana sieć może zostać przetrenowana (ang. over-fitting) gdy model jest nadmiernie dopasowany do szumu zawartego w danych treningowych. Jest to powszechny problem w uczeniu maszynowym i analizie danych, może doprowadzić do uzyskania słabych wyników predykcji i straty cennego czasu. Regularizacja sieci neuronowej to technika, która wprowadza niewielkie modyfikacje do procesu uczenia sieci, aby model uogólniał się lepiej i aby zachowywał się podobnie na danych na których nie był trenowany.

#### Ridge regression L2

Metoda stosowana, gdy wielkość próbki danych jest stosunkowo niewielka, Ridge regression może poprawić wyniki otrzymane na zbiorze walidacyjnym poprzez zmniejszenie uzyskanej wariancji predykcji, dzięki zmniejszeniu wrażliwości prognoz na dane treningowe.

#### Lasso regression L1

Bardzo podobny do Ridge regression z tą różnicą, że zamiast podnosić szacowane parametry modelu do kwadratu, obliczamy ich wartości bezwzględne:

$$\text{SSE} + \lambda | \text{szacowane parametry modelu} |$$

Różnica między regresją Ridge'a a regresją Lasso polega na tym, że w regresji Ridge'a wartości parametrów mogą jedynie asymptotycznie zmierzać do zera, natomiast w regresji lasso mogą być równe zero, co pozwala na całkowite pozbycie się niepotrzebnych parametrów modelu a zatem zmniejszając otrzymaną wariancję gdy parametry są nie istotne. W przeciwnieństwie regresja Ridge radzi sobie lepiej gdy wszystkie parametry modelu są istotne.

#### Elastic Net Regression

Metoda zalecana gdy istnieje korelacja między parametrami. Elastic Net Regression grupuje i redukuje parametry z skorelowanymi zmiennymi, pozostawiając je w równaniu lub usuwając je.

#### Learning rate

To hiperparametry modeli który reguluje zakres tempa w jakim dostosowywane są wagie sieci neuronowych do danych, czyli szybkość rytmu uczenia stosowanego przez algorytm optymalizacji. Należy uważać na dwie możliwe pułapki dotyczące parametr uczenia się. Pierwsza dotyczy zbyt dużej szybkości uczenia się, co prowadzi do pominięcia globalnego minimum algorytmu, być może nigdy nie zbliżając się do optymalnego rozwiązania. Drugim możliwym problemem jest wybranie zbyt małej szybkości uczenia się, co powoduje bardzo powolny proces treningu sieci powodując stratę czasu.

#### Drop-out

Technika drop-out polega na losowym pomijaniu neuronów (wraz z ich połączeniami) z sieci neuronowej podczas treningu. Dzięki temu zapobiega się zbyt dużej współpracy między neuronami., a także zwiększa się szybkość treowania sieci poprzez pomijanie niektórych neuronów. Proces powtarza się wiele razy otrzymując wiele sieci z mniejszą liczbą neuronów które następnie służą do oszacowania uśrednionej prognoz wszystkich „ciękników” sieci, tworząc pojedynczą sieć z wszystkimi neuronami i mniejszymi wagami

#### Wczesny stop (ang. Early stopping)

Zbyt mało treningu oznacza, że model nie będzie nie dotrenowany (ang. underfitted) osiągając słabe wyniki na zbiorze testowym treningowym. Zbyt dużo treningu oznacza, że model będzie pasował do zestawu danych treningowych i będzie miał słabą wydajność w zestawie testowym.

Kompromisem jest trenowanie na zestawie danych szkoleniowych, ale zatrzymanie treningu w momencie, gdy jakość predykcji na zbiorze walidacyjnym zaczyna się obniżać. To proste, skuteczne i szeroko stosowane podejście do szkolenia sieci neuronowych.

#### **Bagging**

To technika regularizacji która bazuje się na połączenie kilku modeli. Chodzi o to, aby trenować kilka różnych modeli osobno, a następnie wszystkie modele głosują na różne wynik dla przykładów testowych. Jest to przykład strategii uczenia maszynowego zwanej uśrednianiem modeli.

#### **Batch Normalization**

Normalizacja wsadowa (ang. Batch Normalization) znacznie przyspiesza proces treningu sieci neuronowych oraz pomaga przy starannym dostrojeniu sieci głębokiego uczenia w inicjalizacji wag i parametrów uczenia się. Normalizacja zmniejsza nadmiernego dopasowywania do danych treningowych ponieważ skalowanie danych wprowadza dodatkowy szum do ukrytych warstw sieci neuronowych.

## **26. Wyjaśnij różnicę, pomiędzy wnioskowaniem obserwacyjnym, interwencyjnym i kontrfaktycznym.**

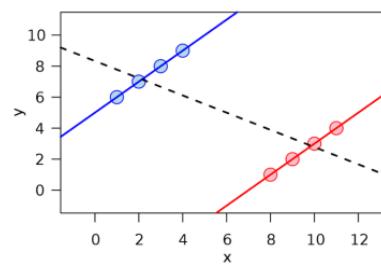
**Wnioskowanie obserwacyjne** polega na analizie danego zjawiska bez próby ingerowania w przyczyny, tzn. możemy sobie dopowiedzieć co mogło być przyczyną danego zdarzenia lub też jaki będzie jego skutek tylko obserwując dane zdarzenie. Czyli standardowa analiza danych, gdzie mamy do dyspozycji zestaw informacji, na które nie mamy wpływu.

**Wnioskowanie interwencyjne** polega na manipulowaniu poszczególnymi zdarzeniami i sprawdzaniem jakie będą skutki, np. jeśli manipulujemy zdarzeniem A i nic się nie dzieje, to A nie może być przyczyną zdarzenia B, ale jeśli manipulacja zdarzeniem B prowadzi do zmiany A, to wiemy, że B jest przyczyną A, chociaż mogą istnieć również inne przyczyny A. Przykładem mogą być drug trials, kiedy część populacji poddaje się leczeniu, a pozostała podaje się placebo, mamy wtedy bezpośredni wpływ na rozwój wydarzeń.

**Wnioskowanie kontrfaktyczne** polega na obserwacji danego zdarzenia i jego przyczyny i zastanowieniu się czy jeśli owa przyczyna nie zaszła by czy na pewno nie zaszeli by uzyskany skutek. Czyli mamy połączenie obserwacji (obserwujemy faktyczne zajście wydarzenia) z interwencją (co by było gdyby manipulować przyczyną – czyli co by było gdyby teoretyczna przyczyna nie istniała). Np. "gdyby nie ingerencja człowieka w klimat, nie byłoby teraz suszy".

## **27. Wyjaśnij na czym polega paradoks Simpsona.**

Paradoks Simpsona – interakcja (zależność) między obserwacjami w obrębie grup jest inna niż globalnie.



Paradoks Simpsona jest oznaczeniem zaskakującej sytuacji, która może wystąpić, gdy dwie populacje są do siebie porównywane w odniesieniu do występowania jakiejś cechy.

Jest to paradoks, w którym trend statystyczny wydaje się być obecny, gdy dane są podzielone na odrębne grupy, ale zanika lub odwraca się, gdy dane są rozpatrywane jako całość.

Paradoks ten jest związany z cechą danych zagregowanych, która może pojawić się w sytuacji, gdy przyczynowe wnioski są wyciągane na różnych poziomach wyjaśnień: od populacji do podgrup lub podgrup do jednostek.

Paradoks Simpsona został opisany przez E.H. Simpsona w 1951 roku.

Simpson wykazał, że statystyczny związek zaobserwowany w danej populacji - tj. zbiór podgrup lub jednostek - może zostać odwrócony we wszystkich podgrupach składających się na tę populację.

Ten paradoks ma istotne konsekwencje dla nauk medycznych i społecznych.

**Przykład:**

Leczenie, które wydaje się skuteczne na poziomie populacji, może w rzeczywistości mieć niekorzystne skutki w każdej z podgrup populacji.

Wyższe dawkowanie leku może być związane z wyższymi wskaźnikami wyzdrowienia na poziomie populacji; jednakże w podgrupach (np. zarówno dla mężczyzn jak i kobiet) wyższe dawkowanie może w rzeczywistości skutkować niższymi wskaźnikami wyzdrowienia. Nawet jeśli istnieje negatywna zależność między "dawką leczniczą" a "odzyskaniem sprawności" zarówno u mężczyzn, jak i kobiet, w przypadku połączenia tych grup pojawia się pozytywna tendencja (przerwana linia).

Tak więc, gdyby analizować te dane globalnie, sugerowałoby, że preferowane jest leczenie wyższą dawką, podczas gdy w rzeczywistości jest odwrotnie.

Pomimo tego, że istnieje negatywna zależność między dawką, a wyzdrowieniem zarówno u mężczyzn, jak i u kobiet, to w przypadku zgrupowania razem istnieje zależność pozytywna.

## 28. Przedstaw korzyści ekonomiczne z przetwarzania danych w chmurze.

Cloud computing stał się jednym z wiodących tematów rozostrań nad nowymi technologiami, które mogą obniżyć wydatki w przedsiębiorstwach, jednocześnie zwiększać ich efektywność działania. Współcześnie rozwiązania dostępne w chmurze obliczeniowej mogą w szerokim zakresie wspierać działalność i wpływać na konkurencyjność zarówno dużych, średnich jak i małych przedsiębiorstw.

#### Do najważniejszych korzyści związanych z zastosowaniem Cloud Computingu można zaliczyć:

- uniknięcie zakupu kosztownego sprzętu, oprogramowania oraz braku konieczności utrzymywania specjalistycznych pomieszczeń (przeznaczenie zaoszczędzonych środków finansowych na inne cele biznesowe),
- szybki dostęp do niezbędnych zasobów IT z dowolnego miejsca (wsparcie mobilności pracowników,
- wysoka skalowalność i wydajność udostępnianych zasobów IT. Klient w zależności od potrzeb ma możliwość dowolnie zwiększyć potencjał lub zrezygnować z części użytkowanych zasobów IT. Zmiana wielkości i zakresu użytkowanych produktów może dotyczyć dowolnego czasu i odbywa się automatycznie na żądanie usługobiorcy. Ta funkcjonalność z punktu widzenia optymalizacji kosztów przedsiębiorstwa jest bardzo korzystna, gdyż umożliwia regulowanie opłat za usługi informatyczne w systemie „płat za to, co wykorzystałeś”
- redukcja ryzyka inwestycyjnego w zakresie technologii IT
- relatywnie niższe koszty pozyskania, utrzymania i rozwoju zasobów IT,
- większa przewidywalność kosztów IT. Cloud computing jest usługą pozwalającą na precyzyjne określenie kosztów związanych z jej użytkowaniem. W zależności od zapisów umowy pomiędzy dostawcą a odbiorcą możliwe jest określenie płatności za czas użytkowania bądź za jednostkę udostępnionej mocy obliczeniowej lub pojemności dyskowej. Sprzyja to przewidywalności i przejrzystości kosztów danego rozwiązania. Dostawca dokonuje optymalizacji pracy posiadanych zasobów poprzez monitorowanie i mierzenie wykorzystania zasobów przez swoich klientów w celu jak najlepszego ich zagospodarowania.
- mniejsze zapotrzebowanie na kadry IT,
- przeniesienie odpowiedzialności za funkcjonowanie i rozwój zasobów IT na dostawcę

## 29. Omów technologie serveless w gromadzeniu i przetwarzaniu danych na potrzeby procesów analitycznych.

Serverless to, mówiąc krótko, model usług w chmurze, w którym programista/architekt skupia się wyłącznie na tworzeniu logiki biznesowej, a nie na infrastrukturze, na której ma ona być wykonywana. Termin serverless architecture może sugerować, że żadnych serwerów nie ma. Jest to oczywiście nieprawda - serwery fizyczne są. Są w tym, że programista tworząc rozwiązanie serverless nie musi się w ogóle zajmować stawianiem maszyn, aktualizacją systemów operacyjnych, konfiguracją sieci, skalowaniem aplikacji. Odpowiedzialność za to przejmuje dostawca danej usługi serverless, czyli np. Amazon, Microsoft, Google. Każdy z dużych dostawców chmury publicznej ma w swojej ofercie usługi, które możemy nazywać bezserwerowymi.

Jakie warunki usługa musi spełnić, aby zasłużyć na to miano? Moim zdaniem jest ich kilka:

- Zero administrowania infrastrukturą
- Automatyczne skalowanie usługi wraz z rosnącym obciążeniem
- Płatność za faktycznie wykorzystywane zasoby, brak konieczności ponoszenia kosztów „z góry”

Potencjalne zalety takich rozwiązań są widoczne na pierwszy rzut oka – ograniczenie kosztów operacyjnych, większa elastyczność i dynamika procesu tworzenia oprogramowania, a co za tym idzie szybsze dostarczanie wartości w postaci nowych funkcjonalności aplikacji/systemu.

<https://aws.amazon.com/serverless/> ← tu trochę o przykładach też :)

## **30.** Przedstaw metody przechowywania danych dużych rozmiarów w chmurze.

Jak skalować dane w chmurze? Wyróżnia się 3 metody skalowania.

- Wirtualne dyski – każdy plik ma swoją nazwę, tak jak na komputerze stacjonarnym, tylko że w chmurze.
- Key object storage – np. S3 w AWS, każdy element, który chcemy przechowywać jest obiektem i ma unikalny klucz (np. URL). Po tym URL można się dostać do obiektu. Nie można aktualizować pliku jak w wirtualnych dyskach, tylko tutaj plik się podmienia. S3 pozwala przechowywać aktualną wersję.
- Bazy danych SQL (relacyjne bazy danych) – każda tabela ma unikalny klucz i kolejne atrybuty w kolumnach użytkownik ma swoje ID i atrybuty jak: imię, nazwisko, email. Aby połączyć użytkownika z atrybutem trzeba w tabeli użytkownik wpisać jako klucz obcy klucz z tabeli adresy.
- Bazy noSQL (nierelacyjne bazy danych) – not only SQL. Nie musi być schematu tak jak w bazach relacyjnych, większa swoboda i szybsze w obsłudze.

#### **Składowanie danych w chmurze:**

- Virtualne dyski
- Key object storage
- Bazy danych SQL i noSQL

Najpopularniejsi dostawcy serwerów w chmurze: AWS S3, Google Cloud Storage, pCloud, F(x) Data Cloud, Azure Storage, hubiC

#### **Przykłady z wykorzystaniem AWS:**

- EBS – Elastic Block Storage - pamięć blokowa (wirtualny dysk w chmurze)
- EFS – Elastic File System - pamięć blokowa współdzielona pomiędzy innymi komputerami (Network File Storage)
- S3 – Simple Storage Service – magazyn typu klucz - wartość, pozwala na przechowywanie dowolnych danych binarnych

Są to kontenery (buckety) tworzone w ramach regionu. Dane nigdy nie opuszczają swojego regionu, w ramach regionu dane są replikowane do wszystkich stref dostępności (availability zones)

Dane cechuje trwałość (durability) na poziomie 99,9999% (prawdopodobieństwo przechowania obiektu przez okres jednego roku)

99,99% availability – bardzo łatwa dostępność danych przechowywanych w bucket'cie

#### **Główne zastosowania S3:**

- hostowanie stron www (od małych do ogromnych rozmiarów),
- krótko-, średnio- i długotrwała archiwizacja danych dowolnego rozmiaru (S3, S3-IAS, S3-Glacier)
- import fizycznych dysków do chmury (AWS Snowball)
- przechowywanie danych dla analistyki Big Data
- analityka danych w klastrach Hadoop / Spark – znacznie tańsza niż HDFS

#### **Zasady tworzenia S3:**

- należy wybrać region
- nadać bucketowi unikalną (dla całego AWS) nazwę
- w obrębie kontenera można tworzyć foldery i określać reguły dostępu

#### **Sposoby przenoszenia danych do S3:**

- AWS import / export - wysyłanie danych do AWS
- AWS Snowball - płatna usługa - zamówienie kuriera z seifem z dyskami, na które wgrywa się dane
- AWS Snowmobile - ciężarówka 'zbierająca' i 'przewożąca' dane
- AWS direct connect - bezpośrednie dedykowane połączenie naszego data center z AWS

#### **Bazy danych relacyjne (RDS, Reshift) oraz nierelacyjne (NoSQL – DynamoDB)**

##### **Relacyjne:**

- wartości pól oparte na typach prostych (liczba, tekst, wartość logiczna)
- dane zapisane w dwuwymiarowych tabelach (encjach / relacjach – wiersze (rekordy / krotki) i kolumny (atrybuty))

- systemy zarządzania bazami danych:
- DBMS – Database Management System
- RDS – Relational Database Management System

Właściwości relacyjnej bazy danych:

- wszystkie dane w bazie przedstawiane są w postaci dwuwymiarowych tabel
- dane z różnych kolumn mogą być porównywane
- kolejność wierszy w tabelach nie jest istotna, operacje są definiowane logicznie
- kolejność kolumn w tabelach nie jest istotna, kolumna jest określana przez nazwę
- odpowiadające sobie wiersze z wielu tabel można łączyć (klucz obcy)

RDS – zalety – ACID:

- Atomicity - atomowość transakcji - zbiór operacji połączony w transakcję wykoną się albo w całości albo wcale
- Consistency - spójność transakcji - baza danych umożliwia definiowanie zasad integralności danych i je kontroluje)
- Isolation - izolacja transakcji - niezatwierdzona transakcja nie wpływa na inne
- Durability - trwałość danych - zatwierdzona transakcja nie zostaje utracona

Przykład Google Cloud:

BigQuery – jest to hurtownia danych w chmurze, ładuje się do niej dane i query są wykonywane już na zasobach w chmurze, co zwiększa szybkość odczytu w przypadku dużych zbiorów danych

Tak jak w przypadku AWS dane są zbierane w obrębie regionów

Również tworzy się buckety z danymi, co pozwala zwiększyć bezpieczeństwo danych (dopiero po nadaniu uprawnień ma się dostęp do bucketa)

### **31.** Omów skalowanie dokumentowych baz danych typu noSQL w chmurze na przykładzie DynamoDB.

Skalowanie baz danych to rozbudowywanie serwera lub dodawanie nowych serwerów. Skalowanie pionowe polega na dokładaniu zasobów do istniejącego serwera, natomiast skalowanie poziome polega na dokładaniu serwerów, które przechowują kopie baz danych i obsługują część żądań.

Bazy danych typu noSQL (non SQL) to nierelacyjny typ baz danych, zawierający dane niestrukturyzowane. Silniki typu noSQL pozwalają przekazywać dowolne dane, bez uprzednio przygotowanych schematów, a także efektywnie korzystać z ich analizy. Dzięki temu bazy noSQL są znacznie bardziej elastyczne od baz relacyjnych, odpowiednie do obsługiwanego setek tysięcy użytkowników aplikacji jednocześnie.

Dynamo DB to usługa oferowana przez Amazon. Jest to trwała baza danych z wbudowanymi zabezpieczeniami, kopiami zapasowymi i możliwością przywracania danych. Zaprojektowana do obsługi wysokowydajnościowych aplikacji, jej główne cechy to:

#### **Wydajność na dużą skalę**

DynamoDB może obsługiwać tabele dowolnego rozmiaru dzięki skalowaniu poziomowemu. Może wykonać ponad 10 bilionów żądań dziennie, nawet przy 20 milionach żądań na sekundę. Obsługuje zarówno modele klucz-wartość, jak i dokumenty, dzięki czemu schemat DynamoDB jest na tyle elastyczny, że każdy wiersz danych może mieć dowolną liczbę kolumn w dowolnym momencie.

Dodatkowo, DynamoDB posiada pamięć podręczną DAX (DynamoDB Accelerator), zapewniającą wysoką wydajność odczytu tabel na dużą skalę: czas odczytu od milisekund do mikrosekund, nawet przy milionach żądań na sekundę.

Globalne tabele DynamoDB automatycznie replikują dane w wybranych regionach AWS i skalią pojemność, dostosowując ją do obciążenia, dzięki czemu czas odczytu i zapisu zostaje skrócony. Natomiast DynamoDB Streams przechwytyują uporządkowaną w czasie sekwencję modyfikacji i zapisują ją do 24 godzin, dzięki czemu aplikacje mogą korzystać z przechwytywania zmian.

#### **Architektura w chmurze**

DynamoDB jest bezserwerowy – automatycznie skaluje tabele w górę i w dół, aby dostosować pojemność i utrzymywać wydajność. Pozwala to na optymalizację kosztów, dzięki dostosowywaniu potrzebnych zasobów do obciążenia.

#### **Gotowość obsługi przedsiębiorstw**

DynamoDB obsługuje transakcje ACID (atomicity, consistency, isolation, durability) - umożliwia tworzenie aplikacji biznesowych na dużą skalę. Wszystkie dane są domyślnie szyfrowane, istnieje możliwość tworzenia pełnych kopii zapasowych setek terabajtów danych.

## **32. Omów skalowanie procesów analitycznych w chmurze.**

Skalowanie procesów analitycznych w chmurze polega na dynamicznym przydzielaniu zasobów w celu dopasowania do wymagań dotyczących wydajności procesu np. wiele użytkowników zacznie korzystać z aplikacji, to powoduje, że aplikacja działa wolniej.

- Skalowanie w pionie – oznacza zmianę pojemności zasobu np. przeniesienie aplikacji na wirtualną maszynę o większym rozmiarze (np. dodanie RAM'u pamięci)
- Skalowanie w poziomie – oznacza dodawanie lub usuwanie wystąpień zasobu (np. dodanie kilku maszyn, używanie kilku serwerów zamiast 1 – rozproszenie na kilku pracowników)
- Skalowanie automatyczne – to instrumentacja i monitorowanie systemów na poziomie aplikacji, usługi. Jest to logika podejmowania decyzji, która decyduje o tym czy wykonać dane zadanie względem wcześniej zdefiniowanych progów. Polega na ciągłym testowaniu i monitorowaniu strategii skalowania automatycznego. Ustawienie parametrów – serwer patrzy, że zwiększa się liczba zapytań, powoduje zwiększenie liczby serwerów. Gdy nie potrzeba jest takiej mocy obliczeniowej, wyłącza część serwerów.

Skalowanie procesów w chmurze polega na dynamicznym przydzielaniu zasobów w celu dopasowania do wymagań dotyczących wydajności. Gdy rośnie ilość pracy, aplikacja może potrzeować dodatkowych zasobów, aby utrzymać wymagane poziomy wydajności i spełniać warunki umów dotyczących poziomu usług (SLA). Kiedy zapotrzebowanie spada, dodatkowe zasoby nie są już potrzebne i można cofnąć ich przydział, aby zminimalizować koszty.

Skalowanie automatyczne wykorzystuje elastyczność środowisk hostowanych w chmurze, zmniejszając nakłady pracy związane z zarządzaniem oraz redukując potrzebę ciągłego monitorowania wydajności systemu przez operatora i podejmowania decyzji o dodaniu lub usunięciu zasobów.

Istnieją dwa sposoby skalowania aplikacji:

- Skalowanie w pionie (w górę i w dół) oznacza zmianę pojemności zasobu. Przykładowo można przenieść aplikację na maszynę wirtualną o większym rozmiarze. Skalowanie w pionie wymaga często tymczasowej niedostępności systemu podczas jego ponownego wdrażania. W związku z tym automatyzowanie skalowania w pionie jest rzadziej używane.
- Skalowanie w poziomie (na zewnątrz i do wewnątrz) oznacza dodawanie lub usuwanie wystąpień zasobu. Podczas aprowidowania nowych zasobów aplikacja będzie nadal działać bez przeszkód. Po zakończeniu procesu aprowidacji rozwiązanie będzie wdrożone w tych dodatkowych zasobach. Jeśli zapotrzebowanie spadnie, dodatkowe zasoby można bezproblemowo wyłączyć i cofnąć ich przydział.

Strategia skalowania automatycznego obejmuje:

- Instrumentację i monitorowanie systemów na poziomie aplikacji, usługi i infrastruktury.
- Logikę podejmowania decyzji, która ocenia te metryki względem wcześniej zdefiniowanych progów lub harmonogramów i decyduje o tym, czy wykonać skalowanie.
- Składniki, które skalują system.
- Testowanie, monitorowanie i dostosowanie strategii skalowania automatycznego, aby upewnić się, że działa zgodnie z oczekiwaniemi.

Warto zaznaczyć, że skalowanie automatyczne dotyczy przede wszystkim zasobów obliczeniowych. Mimo że istnieje możliwość skalowania w poziomie bazy danych lub kolejki komunikatów, wiąże się to zwykle z partycjonowaniem danych, które zazwyczaj nie jest zautomatyzowane.

### **33. Omów Function as a service - model przetwarzania oparty o architekturę Lambda.**

#### **Co to jest function as a service (FaaS)?**

FaaS (Function As A Service) należy do kategorii CCS (Cloud Computing Services), która zapewnia klientom platformę do tworzenia, uruchamiania i zarządzania aplikacjami. Robi to bez skomplikowanej konserwacji i budowy infrastruktury, która zwykle wiąże się z opracowaniem i uruchomieniem aplikacji. Budowanie aplikacji zgodnie z tym modelem jest sposobem na uzyskanie architektury „bezserwerowej”. Ten model jest najczęściej używany do budowania mikroserwisów (microservices).

#### **Zasady FaaS:**

- Pełna abstrakcja serwerów od dewelopera
- Fakturowanie na podstawie zużycia i wykonania, a nie wielkości instancji serwera
- Usługi sterowane zdarzeniami i natychmiast skalowalne

#### **Czy to jest AWS Lambda?**

AWS Lambda to usługa obliczeniowa, która pozwala uruchamiać kod bez obsługi administracyjnej lub zarządzania serwerami. AWS Lambda wykonuje kod tylko w razie potrzeby i skaliuje się automatycznie, od kilku żądań dziennie do tysięcy na sekundę. AWS Lambda uruchamia kod w infrastrukturze obliczeniowej o wysokiej dostępności i wykonuje całą administrację zasobami obliczeniowymi, w tym konserwację serwera i systemu operacyjnego, zapewnianie pojemności i automatyczne skalowanie, monitorowanie i rejestrowanie kodu.

AWS Lambda może zostać wykorzystany do uruchomienia kodu w odpowiedzi na zdarzenia, takie jak zmiany danych w segmencie Amazon S3 lub tabeli Amazon DynamoDB; do uruchamiania kodu w odpowiedzi na żądanie HTTP przy użyciu Amazon API Gateway; lub wywołania kodu za pomocą wywołań API wykonanych przy użyciu AWS SDK. Może zostać użyta również aby zbudować aplikacje bez serwera złożone z funkcji uruchamianych przez zdarzenia i automatycznie wdrażać je za pomocą CodePipeline i AWS CodeBuild. Aby uzyskać więcej informacji, zobacz Aplikacje AWS Lambda.

#### **Jak to działa?**

- Przesłanie kodu do AWS Lambda lub napisanie w edytorze kodów Lamdba.
- Konfiguracja kodu, aby uruchamiał się z innych usług AWS, punktów HTTP lub aktywności w aplikacji.
- Lambda uruchamia kod tylko kiedy zostanie do tego zmuszony(triggered), używając tylko potrzebnych zasobów obliczeniowych.
- Opłata jest tylko za wykorzystany czas obliczeniowy.

### **34. Omów tworzenie i zarządzanie bezpieczeństwem środowisk analitycznych dla języków Python i R w chmurze.**

Główne elementy zarządzania bezpieczeństwem dla chmury:

- 1) Zarządzanie danymi
- 2) Zarządzanie tożsamością i dostępem
- 3) Ochrona danych i prywatność
- 4) Bezpieczeństwo sieci
- 5) Bezpieczeństwo i integralność infrastruktury

1) Zarządzanie danymi:

- Klasifikacja danych (które dane wymagają znacznej ochrony)
- Odkrywanie danych (identyfikacja wrażliwych danych, środki ochrony takie jak maskowanie, redukcja, tokenizacja czy szyfrowanie danych)
- Tagowanie danych (zidentyfikować metody wprowadzania danych w klastrze)

2) Zarządzanie tożsamością i dostępem

- Uprawnienia użytkownika do danych
- Autoryzacja użytkownika na podstawie przypisanej roli

3) Ochrona danych i prywatność

- Kryptografia/tokenizacja danych
- Szyfrowanie przezroczyste
- Maskowanie danych osobowych uniemożliwia identyfikację użytkownika

4) Bezpieczeństwo sieci

- Ochrona danych podczas transportu
- Strefa bezpieczeństwa sieci (kontrola zabezpieczająca dostęp do danych poziomów)

5) Bezpieczeństwo i integralność infrastruktury

- Rejestrowanie/audyt (prowadzenie dzienników kontroli)

Struktura bezpieczeństwa projektów Big Data (na przykładzie hadoop).

Poniższa sekcja może zapewnić bezpieczeństwo docelowe architektury bezpieczeństwa platformy Big Data.

Główne elementy proponowanego bezpieczeństwa Big Data są następujące:

- Zarządzanie danymi
- Zarządzanie tożsamością i dostępem
- Ochrona danych i prywatność
- Bezpieczeństwo sieci
- Bezpieczeństwo i integralność infrastruktury

Powyższe „5 filarów” Struktury Bezpieczeństwa projektów Big Data można rozłożyć na 21 elementów, z których każdy ma kluczowe znaczenie dla zapewnienia bezpieczeństwa i łagodzenia skutków ryzyka.

#### 1. Zarządzanie danymi

Zarządzanie danymi możemy podzielić na trzy podstawowe podskładniki. Są to: klasyfikacja danych, odkrywanie danych i tagowanie danych.

##### 1.1 Klasyfikacja danych

Skuteczna klasyfikacja danych jest prawdopodobnie jednym z najważniejszych działań, które mogą doprowadzić do skutecznego wdrożenia kontroli bezpieczeństwa na platformie Big Data. Kiedy organizacje mają do czynienia z bardzo dużą ilością danych, są w stanie określić, które dane są ważne, co wymaga między innymi ochrony kryptograficznej i jakie pola należy najpierw ustalić jako priorytetowe dla ochrony.

##### 1.2 Odkrywanie danych

Brak świadomości sytuacyjnej w odniesieniu do wrażliwych danych może narazić organizację na znaczne ryzyko. Kluczem jest identyfikacja, czy wrażliwe dane są obecne, gdzie się znajdują, a następnie uruchomienie odpowiednich środków ochrony danych, takich jak maskowanie danych, redakcja danych, tokenizacja lub szyfrowanie.

##### 1.3 Tagowanie danych

Należy zidentyfikować wszystkie metody wprowadzania danych w klastrze. Obejmują one wszystkie metody ręczne (np. Administratorzy) lub metody automatyczne (np. Zadania ETL) lub te, które przechodzą przez niektóre meta-warstwy (np. Kopiowanie plików lub tworzenie + zapis).

#### 2. Zarządzanie tożsamością i dostępem

##### 2.1 Uprawnienie użytkownika + pomiar danych

Użytkownicy mają dostęp do danych poprzez centralne zarządzanie zasadami dostępu. Ważne jest, aby powiązać politykę z danymi, a nie z metodą dostępu

##### 2.2 Autoryzacja użytkowników na podstawie przypisanej do nich roli.

Zarządzanie dostępem do danych według roli (a nie użytkownika). Określenie relacji między użytkownikami i rolami przez grupy.

Ochrona kryptograficzna na poziomie aplikacji (taka jak szyfrowanie na poziomie pola / kolumny, tokenizacja danych oraz redakcja / maskowanie danych) zapewniają kolejny wymagany poziom bezpieczeństwa.

### 3.1 Kryptografia na poziomie aplikacji (tokenizacja, szyfrowanie na poziomie pola)

Choć szyfrowanie na poziomie pola / elementu może oferować szczegółowość zabezpieczeń i możliwości śledzenia audytu, wiąże się to z koniecznością ręcznej interwencji w celu ustalenia pól wymagających szyfrowania oraz miejsca i sposobu włączenia autoryzowanego deszyfrowania.

### 3.2 Szyfrowanie przezroczyste (warstwa dysku / HDFS)

Pełne szyfrowanie dysku uniemożliwia dostęp za pośrednictwem nośnika pamięci. Szyfrowanie plików może także chronić (uprzywilejowany) dostęp na poziomie systemu operacyjnego węzła.

### 3.3 Maskowanie danych / Redakcja danych

Maskowanie danych lub redakcja danych przed załadowaniem w typowym procesie ETL usuwa dane osobowe umożliwiające identyfikację użytkownika przed załadowaniem. Dlatego żadne poufne dane nie powinny być przechowywane.

### 4. Bezpieczeństwo sieci

Warstwa bezpieczeństwa sieci jest podzielona na cztery podskładniki. Stanowią ochronę danych podczas transportu oraz w strukturze sieciowej + elementy autoryzacji.

#### 4.1 Ochrona danych podczas transportu

Istnieje wiele scenariuszy zagrożeń, które z kolei nakazują konieczność https i zapobiegają ujawnieniu informacji lub podniesieniu kategorii zagrożeń uprzywilejowanych. Korzystanie z protokołu TLS w celu uwierzytelnienia i zapewnienia prywatności komunikacji między węzłami, serwerami nazw i aplikacjami.

#### 4.2 Strefy bezpieczeństwa sieci

Klastry muszą być podzielone na punkty dostawy z punktami pośrednimi, w których sieciowe listy kontroli dostępu ograniczają dozwolony ruch do zatwierdzonych poziomów.

#### 5. Bezpieczeństwo i integralność infrastruktury

Warstwa bezpieczeństwa i integralności infrastruktury jest podzielona na podstawowe podskładniki. Są to: rejestrowanie / audit, integralność plików + monitorowanie sabotażu danych oraz uprzywilejowane monitorowanie użytkowników i aktywności.

#### 5.1 Rejestrowanie / audit

Wszystkie zmiany systemowe / ekosystemowe muszą być kontrolowane przy zachowaniu ochrony dzienników kontroli.

Gdy dane nie są ograniczone do jednego z podstawowych komponentów, bezpieczeństwo danych ma wiele ruchomych części i wysoki procent fragmentacji. W związku z tym istnieje rozrzucona metadanych i dzienników kontroli we wszystkich fragmentach.

## **35. Omów zarządzanie bezpieczeństwem, użytkownikami i prawami dostępu w chmurze - użytkownicy, role, polityki i grupy.**

## **35. Omów zarządzanie bezpieczeństwem, użytkownikami i prawami dostępu w chmurze - użytkownicy, role, polityki i grupy.**

### **Użytkownicy**

Firma posiadając konto AWS może utworzyć konta dla swoim pracowników, którzy będą mogli korzystać ze wspólnych zasobów chmury obliczeniowej. Po utworzeniu konta dla pracownika, pracownik dostaje ID danego konta AWS, login w obrębie firmy i hasło. Drugim rodzajem użytkowników są użytkownicy z programistycznym dostępem do chmury. Przykładem takiego użytkownika może być aplikacja mobilna, w której korzystamy z zapisywania i odczytywania zdjęć za pomocą S3. W tej sytuacji po rejestracji użytkownika dostajemy access key ID i secret access key, które podajemy w konfiguracji aplikacji.

### **Polityki**

Określają zbiór pozwoleń na używanie poszczególnych usług AWS. Na przykład możliwe jest stworzenie polityki "Edycja zdjęć", w której możliwe będzie przeglądanie, dodawanie i edytowanie zdjęć, bez prawa do usuwania. Politykę taką można następnie przypisać konkretnej osobie, roli lub grupie. Stworzenie takiej polityki może odbywać się poprzez "wyklikanie" lub poprzez zdefiniowanie polityki w pliku o formacie json.

### **Rola**

Zawiera przynajmniej jedną Politykę. Za pomocą ról możliwe jest pośrednie dodawanie polityk do konta użytkownika. Przykładowo można utworzyć rolę "Administracja aplikacją XXX", która będzie zawierała 5 polityk. Dodając nowego użytkownika do projektu aplikacji mobilnej łatwiejsze z zarządzaniem jest nadać mu jedną rolę niż 5 Polityk.

\*Możliwe jest również utworzenie roli z politykami dla użytkowników aplikacji mobilnej logujących się np poprzez Google lub Facebooka. Nie jest możliwe przypisywanie każdemu takiemu użytkownikowi loginu i hasła do AWS.

### **Grupa**

W obrębie konta AWS utworzyć można przykładowo dwie grupy: "Administratorzy", "Programiści" i nadać tym grupom odpowiednie role. Gdy do firmy dołączy nowy programista wystarczy umieścić go w grupie "Programiści" i dzięki temu zamiast przypisywać mu kilkańście potrzebnych ról, wszystkie role przypisane do tej grupy są od teraz rolami nowego programisty. Możliwe jest również umieszczenie programisty w obydwu grupach.

**36.** Przedstaw systemy zarządzania relacyjną bazą danych w chmurze i ich zastosowania w analityce danych.

Główymi dostawcami RDBMS (ang. Relational Database Management System – system zarządzania relacyjną bazą danych) w chmurze są Amazon, Microsoft, Oracle. Oferują oni usługi na swoich platformach: Amazon Web Services (AWS), Microsoft Azure i Oracle Cloud.

Systemy zarządzania relacyjną bazą danych w chmurze, nie różnią się wiele od tych dostępnych na prywatnych stacjach roboczych. Posiadają one natomiast przewagi jakie gwarantuje użycie chmury obliczeniowej, a więc dostępność, elastyczność oraz płatności za faktyczne zużycie.

Amazon oferuje Amazon Relational Database Service, która pozwala na łatwiejsze stworzenie, zarządzanie, skalowanie relacyjnej bazy danych w chmurze AWS. Usługa ta dostarcza wydajny kosztowo oraz możliwy do dopasowywania rozmiarem zasób dla stworzenia relacyjnej bazy danych oraz zarządza popularnymi zadaniami administracji baz danych. Z wykorzystaniem Amazon RDS użytkownik może korzystać z narzędzi, które już zna, czyli MySQL, MariaDB, PostgreSQL, Oracle, Microsoft SQL Server. Platforma dodatkowo zapewnia kopie zapasowe tworzonego systemu. Użytkownik może założyć wymaganą liczbę baz danych i zarządzać kilkoma jednocześnie, w ramach jednej usługi. Silnik do obsługi relacyjnych baz danych, który oferowany jest przez Amazon nazywa się Aurora.

Praca z relacyjną bazą danych w chmurze (np. Amazon) pozwala nie tylko na jej tworzenie i zarządzanie, ale także na wykorzystanie jej do analizy danych. AWS dostarcza rozwiązania, które wykorzystywane są do dokładnego analizowania danych, które przechowywane są w bazach danych, które znajdują się na platformie. Narzędziami do analizy danych w AWS Aurora jest np. Amazon Aurora Parallel Query. Narzędzie to pozwala na szybsze przetwarzanie zapytań na posiadanych danych, ponieważ nie potrzebne jest ich kopowanie do innego systemu, są one automatycznie zaczynane z Aurora. Innymi narzędziami do analizy czy wizualizacji danych od Amazon są Athena czy QuickSight.

## Systemy zarządzania bazą danych (DBMS)

### ❑ Oprogramowanie w architekturze klient-serwer

- Silnik bazy danych (ang. engine) – serwer
- Oprogramowanie warstwy klienta (ang. front-end), narzędzia 4GL
- Ewentualnie oprogramowanie pośrednie (ang. middleware)

### ❑ Różni dostawcy

- Firmy komercyjne: **Oracle**, IBM (DB2) , Sybase (Adaptive Sever), Teradata **Microsoft (MS SQL Server)**
- Open Source : **MySQL**, **PostgreSQL**, **Firebird**, **HSQLDB**

+OLAP++HURTOWNIE+OLTP

## **37. Przedstaw modele przetwarzania danych w chmurze: IaaS (Infrastructure-as-aService), PaaS (Platform-as-a-Service) oraz SaaS (Software-as-a-Service).**

### **Infrastruktura jako usługa (IaaS)**

Infrastruktura jako usługa, czasami w skrócie IaaS, zawiera podstawowe elementy składowe IT w chmurze i zazwyczaj zapewnia dostęp do funkcji sieciowych, komputerów (wirtualnych lub na dedykowanym sprzęcie) i przestrzeni do przechowywania danych. Infrastruktura jako usługa zapewnia najwyższy poziom elastyczności i kontroli zarządzania zasobami IT i jest najbardziej podobna do istniejących zasobów IT, które jest znane wielu działom IT i programistom.

### **Platorma jako usługa (PaaS)**

Platformy jako usługa eliminują potrzebę zarządzania przez infrastrukturę bazową (zwykle sprzętem i systemami operacyjnymi) i pozwalają skupić się na wdrażaniu aplikacji i zarządzaniu nimi. Pomaga to być bardziej wydajnym, ponieważ nie musisz martwić się o zakup zasobów, planowanie wydajności, konserwację oprogramowania, łatanie lub inne niezróżnicowane podnoszenie ciężarów związane z uruchomieniem aplikacji.

### **Oprogramowanie jako usługa (SaaS)**

Oprogramowanie jako usługa zapewnia gotowy produkt, który jest uruchamiany i zarządzany przez usługodawcę. W większości przypadków osoby określające Oprogramowanie jako Usługę odnoszą się do aplikacji użytkowników końcowych. Dzięki ofercie SaaS nie musisz myśleć o tym, jak usługa jest utrzymywana ani jak zarządzana jest podstawowa infrastruktura; musisz tylko pomyśleć o tym, jak będziesz używać tego konkretnego oprogramowania. Typowym przykładem aplikacji SaaS jest internetowa poczta e-mail, w której można wysyłać i odbierać wiadomości e-mail bez konieczności zarządzania dodatkami do produktu e-mail lub utrzymywania serwerów i systemów operacyjnych, na których działa program pocztowy.

### **SaaS (Software as a service) – oprogramowanie hostowe/z sieci**

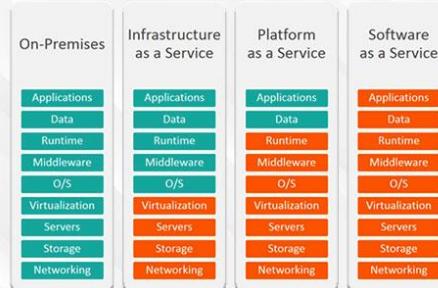
Całość aplikacji i danych przechowywana jest zewnętrznie na serwerach dostawcy usługi, a użytkownicy mają do niej dostęp przez Internet. Na komputerze użytkowników nie jest instalowane oprogramowanie.

Dostarczenie klientowi określonych funkcji oprogramowania. Użytkownik otrzymuje dostęp tylko do potrzebnych narzędzi, które nie muszą być połączone za pomocą jednego interfejsu. Klient plací jednorazowo za skorzystanie z usługi/oprogramowania np. poczta e-mail. Dział IT zajmuje się wyłącznie zapewnieniem dostępu do aplikacji użytkownikom i zapewnia rozwiązywanie ich problemów. Jest to ograniczona możliwość dostosowania funkcjonalności do własnych potrzeb. Można zmieniać opcje oprogramowani, ale kształt aplikacji pozostaje taki sam.

## Typy usług oferowanych w chmurze

- Infrastructure-as-a-service (IaaS)
  - Zwirtualizowana maszyna (server) (virtual computer)
    - Linux
    - Windows
- Platform-as-a-service (PaaS)
  - zarządzana baza danych w chmurze (relacyjna Oracle, MS SQL Server, PostgreSQL, no sql - DynamoDB)
  - Hosting WWW w chmurze
- Software-as-a-service (SaaS)
  - Oprogramowanie zarządzane w chmurze
    - MS Office 365
    - GMail
    - Google docs

## IAAS VS. PAAS VS. SAAS



FIELDFORCECONNECT

You Manage

Other Manages

**38.** Omów kwestie etyczne związane z Big Data.

Technologia big data dostarcza wyzwań z ochroną prywatności przetwarzanych danych. Następuje wybór pomiędzy skupieniem się na przydatności danych lub na ich prywatności.

Same duże zbiory danych, jak każda technologia, są etycznie neutralne, ich wykorzystanie może jednak neutralne nie być. Choć etyka jest pojęciem abstrakcyjnym, może mieć bardzo realne implikacje. Celem jest opracowanie lepszych sposobów i środków angażowania się w celowe dochodzenie etyczne w celu informowania i dostosowywania naszych działań do naszych wartości.

Możliwym wyjściem z tej sytuacji jest próba anonimizacji danych.

Inteligentna ocena gigantycznych ilości danych pochodzących z różnych źródeł pozwala firmom uzyskać wgląd w zainteresowania i życie użytkowników. Firmy są w stanie wyśledzić trendy lub wzory statystyczne, zasady lub korelacje między indywidualnymi cechami. Dla przykładu, mogą one przewidzieć przyszłe zachowania, zestawiając i oceniaciąc przeszłe i obecne wzory działania. Firmy, organizacje, państwa są bardzo zainteresowane tymi prognozami z wielu powodów: mogą je wykorzystywać do systemów wczesnego ostrzegania przed potencjalnymi zagrożeniami, minimalizowania ryzyka, oszczędzania czasu i osiągnięcia zysku. Mogą również wykorzystać je do sprawowania kontroli i użycia siły.

Słедzenie (tracking) i ocenianie (scoring) to najważniejsze metody wykorzystywane do zbierania i analizowania danych osobowych. Obie są używane do przewidywania przyszłych zachowań przez tworzenie profilu osoby lub grupy. Zawiera on ich zainteresowania, zwyczaje konsumpcyjne, miejsce pobytu, kontakty społeczne, wypłacalność, zdolność kredytową, zachowania i informacje o zdrowiu.

Firmy wykorzystują nasze dane do mierzenia, oceny i klasyfikowania nas, a nawet tworzenia złożonych profiliów. Mogą nas podzielić na dobrych i złych klientów, ustalać indywidualne ceny lub premie, oceniać naszą wypłacalność, przewidywać nasze potrzeby czy wzory zachowania, odmówić nam ubezpieczenia produktu lub zaoferować go nam na gorszych warunkach. Nasze dane mogą być wykorzystane do ustalenia poglądów politycznych i religijnych, stanu zdrowia, preferencji seksualnych, a nawet emocji i nastrojów. W rezultacie firmy i organizacje, które dysponują tymi danymi, mają też wiele możliwości manipulacji, dyskryminacji, kontroli społecznej i nadzoru. Podczas gdy ich możliwości zwiększą się, nasze się zmniejszą, gdyż doświadczamy większych ograniczeń naszej wolności w działaniu i podejmowaniu decyzji. Serwisy internetowe dzięki zebranych danych o preferencjach użytkowników są w stanie przeprowadzić kampanię reklamową dopasowaną do zainteresowań użytkowników.

Dane medyczne, publikowane przez ludzi między innymi przez aplikacje fitnessowe, smartbandy mogą zostać wykorzystane przez firmy ubezpieczeniowe i jeśli ciągle publikowanie informacji o zdrowiu miałyby stać się normą mogłyby one na tym wiele zaoszczędzić.

Big data ocenia ludzi na podstawie ich przewidywanych preferencji, a nie faktycznego zachowania. Ogranicza to możliwości zachowywania się w sposób spontaniczny i swobodnego kształtowania przyszłości. Nasze poprzednie działania nigdy nie są zapomniane - są wykorzystywane do tworzenia prognoz i nigdy nie możemy uciec od przeszłości. Co ważniejsze, analiza prognostyczna dokonywana na Big data zagraża naszej wolności niezależnego działania i podejmowania decyzji.

Anonimizacja danych nie oferuje już odpowiedniej ochrony, gdyż Big data ma coraz większą możliwość reidentyfikacji osoby powiązanej z anonimowymi informacjami przez wykorzystanie coraz to bardziej zróżnicowanych danych. Nawet najbardziej niewinne informacje mogą zdradzić tożsamość osoby, jeśli analizujący system zebrał odpowiednią liczbę danych.

Zbiory danych nie są ani dobre, ani złe. Jednak obecne wydarzenia w cyfrowej sieci, technologiiach nadzoru i w sferze bezpieczeństwa pokazują dobrinie, że Big data oznacza przede wszystkim dużą siłę i duży biznes. Gdziekolwiek to możliwe, firmy, rządy i organizacje publiczne powinny działać zgodnie z zasadą proporcjonalności (przeznaczenia), jakości informacji i sprawiedliwego dostępu. Powinny także jasno zaznaczyć jakich algorytmów używają i „stale sprawdzać oraz potwierdzać wybór i jakość wprowadzanych danych”.

## **39.** Omów cechy danych istotne w procesie analizy danych.

Przed przystąpieniem do analizy danych należy surowe dane sprawdzić pod kątem

- Prawdziwości danych
- Jednoznaczności
- Identyfikowalne
- Kompletne
- Porównywalności
- Aktualności danych
- Z odpowiednio długiego okresu dane

Z danymi należy je odpowiednio:

- Wyczyścić
- Zintegrować
- Przetransformować
- Zredukować wymiar

Dane to surowe fakty, liczby lub inne szczegóły dotyczące pewnych zdarzeń. Najczęściej są zapisywane i przechowywane w postaci sformalizowanych zapisów na nośniku danych. W zależności od źródła pochodzenia mogą charakteryzować się różną jakością. Przed przystąpieniem do dalszej analizy należy ocenić je pod kątem:

- prawdziwości
- jednoznaczności
- identyfikowalności zjawiska przez zmienną/zmienne
- kompletności
- aktualności w przyszłości
- kosztu zbierania i opracowywania
- porównywalności

W wyniku ich przeglądu do dalszej analizy powinny zostać wybrane te dane, które są dokładne, kompletne, indywidualne, zgodne z rzeczywistością, możliwe jak najczystsze i najnowsze, a także dostępne z odpowiednio długiego okresu. Ważne również, aby zawierały informacje istotne dla badanego problemu, odzwierciedlały jego istotę, a ponadto były jednoznacznie oznaczone. W celu wykorzystania odpowiednich narzędzi służących do ich analizy warto, aby dane miały postać płaskiej tabeli lub widoku.

Jeżeli jakość danych jest wątpliwa, to na wynikach też nie będzie można polegać. Z tego względu, aby wydobyć z nich użyteczną wiedzę, w procesie analizy stosuje się techniki, które mają na celu poprawę ich jakości. Do pożądanych praktyk służących poprawie istotnych ich cech zalicza się:

- Czyszczenie danych
- Integrację danych
- Transformację danych (normalizacja)
- Redukcję wymiaru

## **40.** Przedstaw, na czym polega zmienność danych i jak ją uwzględnić w wizualizacji danych.

Posiadanie określonego zbioru danych opisującego badane zjawisko nie zawsze oznacza, że jesteśmy w stanie dowiedzieć się całkowitej prawdy na temat tych zdarzeń. Badane zjawiska mogą być złożone i cechować się dużą zmiennością w czasie. Odpowiednie ujęcie danych i przedstawienie ich w określonym kontekście powoduje, że wnioski z badanego zjawiska nabierają większego sensu. Przedstawianie ilości, średnich i innych zagregowanych pomiarów może być interesujące, jednak fluktuacje danych i ich zmienność mogą okazać się najbardziej cenną informacją posiadanego zbioru.

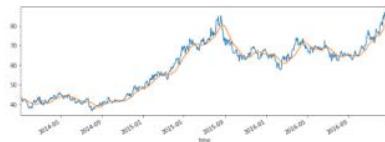
Neodpowiednia wizualizacja może doprowadzić do przeoczenia najważniejszych cech zjawiska. W książce podany jest przykład wypadków samochodowych w USA, które zostały przedstawione poprzez punkty na mapie. Z tak przedstawionego grafu można bardziej dowiedzieć się, jak wygląda sieć dróg w Ameryce, a mało można wyczytać informacji istotnych dla badanego problemu, jakim są wypadki drogowe. W książce przedstawiono inne grafy z tego samego zbioru danych, które przedstawiają, jak zmienia się ilość wypadków drogowych w określonych okresach czasowych.

Przykładowo, zestawienie ilości wypadków z roku na rok pozwoliło przedstawić, że w latach 2006-2010 nastąpił znaczny ich spadek. Przedstawienie ilości wypadków z miesiąca na miesiąc podkreślało sezonowość zjawiska. Dzięki takiemu przedstawieniu danych można było zobaczyć, że najczęściej wypadków występuje w sierpniu, a w następnym miesiącu następuje niewielki wzajemny spadek. Tylko w latach 2005-2007 lipiec miał najczęściej wypadków. Dzięki takiej wizualizacji możemy dowiedzieć się, co mogło spowodować takie zmiany akurat w tych latach lub odpowiednio ukierunkować się do dalszych badań.

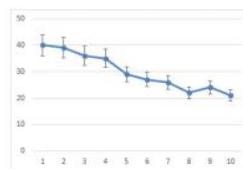
Podkreśla się jednak, że zejście do zbyt dalekiego poziomu szczegółowości może również spowodować nierozumienie badanego zjawiska. Korzystając dalej z przykładu wypadków drogowych zestawienie ilości wypadków drogowych w każdej godzinie w przeciągu roku nie pozwala wyciągnąć klarownych wniosków. Przydatna okazuje się tu odpowiednia agregacja danych i przedstawienie danych osobno dla każdego miesiąca. Dzięki temu widać, że wypadki drogowe zdarzają się najczęściej w godzinach powrotu z pracy, a zestawienie osobne dla każdego miesiąca pozwoliło ominąć problem sezonowości, który mógł znieksztalcić wyciągane wnioski.

W prezentacji danych chodzi przede wszystkim o to, że warto spojrzeć na dane nie tylko w kontekście średniej, mediany lub sumy, ponieważ pomiary te opisują tylko jedną część problemu. Często zdarza się, że te wartości przedstawiają tylko, gdzie jest środek dystrybucji i ukrywają interesujące szczegóły, na których powinno się skupić, zarówno przy podejmowaniu decyzji, jak i prezentowaniu wniosków. Przykładowo może zdarzyć się tak, że wartości odstające przedstawiają coś, co trzeba szybko naprawić lub zwrócić na to szczególną uwagę. Tak samo zmienność danych w czasie może być sygnałem, że coś dobrego (lub złego) dzieje się w danej organizacji.

- 1) Szeregi czasowe - wykres na osi X to czas. Współczynnik zmienności przedstawiony na osi Y to przeskalowana wariancja.



2) Zmienność jako błąd standardowy. Uwzględnienie na wykresach słupków błędu.



## 41. Przedstaw, na czym polega niepewność w analizie danych i jak można wpływać na jej wielkość.

- 1) Brak dostępu do wystarczającej ilości obserwacji – za mała próba, aby uogólnić problem.
- 2) Przy modelu regresji – model jest tworzony na danych historycznych. Nie ma pewności, że model będzie tak samo działał w przyszłości.

Jak wpływać na niepewność w analizie danych?

Poprzez zwiększenie doboru próby, zwiększenie obserwacji.

Niepewność w analizie danych oznacza sytuację, w której dane wartości poszczególnych zmiennych mogą spowodować różne wartości innych zmiennych (np. zmiennej, której wartość próbujemy przewidzieć na ich podstawie) i nie znamy prawdopodobieństwa wystąpienia poszczególnych wyników. Czyli każda decyzja jest obarczona błędem, ponieważ jest podejmowana przez model w warunkach niepewności.

Wpływanie na wielkość niepewności zależy od zadanego problemu, posiadanych cech i używanych algorytmów. Nie ma więc jednej metody jej redukcji.

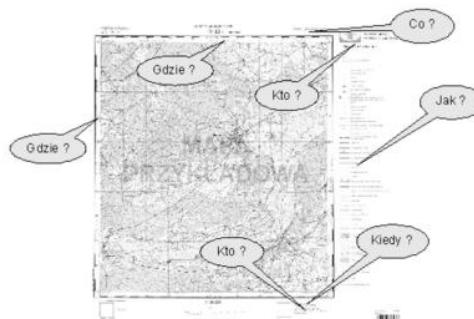
Istnieją dwa główne źródła niepewności:

1 - Błędy losowe - niewynikające z czynników powtarzalnych. To na przykład zaburzenia w rozkładach wartości obserwacji w próbce tzw. obserwacje odstające. Jeżeli jest ich mało, należy rozważyć usunięcie takich obserwacji z analizowanego zbioru. Jeśli usunięcie nie jest pożądane, wówczas warto znormalizować takie wartości. Przy wykrywaniu takich obserwacji pomocne są wizualizacje danych. Sam proces nazywa się czyszczeniem danych.

2 - Błędy systematyczne - wynikające z zastosowanej przez analityka metody. Na przykład źle dobrana próba do analizy. Skrajnym przykładem może być np. brak osób powyżej 40 roku życia w próbce danych na temat klientów księgarskich, dla których wiek klienta ma rozkład normalny. Stąd należy uważnie wybierać dane do analizy tak, aby były jak najbardziej reprezentatywne.

## 42. Jakie znaczenie mają metadane w analizie danych.

**Metadane to informacje o danych.**



Rys. 1. Marginalia mapy jako przykład metadanych, odpowiedź na pytania: kto, kiedy, co, jak i gdzie?

**Metadane można podzielić na:**

- **Opisowe** (dostarczają informacji niezbędnych do odszukania/identyfikacji zbioru np. tytuł, streszczenie, autor, słowa kluczowe)
- **Strukturalne** (ulatwiają interpretację danych i ich praktyczne zastosowanie np. opisanie zależności pomiędzy zbiorami lub elementami zbioru)
- **Administracyjne** (to między innymi sposób utworzenia danych, datę, typ, informację o dostępie)

Jakie są korzyści ze stosowania metadanych?

- o Szybkie uzyskanie informacji o danych
- o Ułatwienie znalezienia informacji o danych
- o Możliwość identyfikacji i zarządzania danymi

Metadane to inaczej "dane o danych", lub też "informacja o informacji". Są to określenia umożliwiające wyszukanie pożąanej informacji wraz z odpowiedzią w jakiej relacji pozostaje ona do innych informacji. Opisując więc zbiory danych przestrzennych powinny zawierać informację o rodzaju obiektów, ich położeniu, pochodzeniu a także dotyczące ich szczegółowości, standardów, praw własności i praw autorskich, cen jak również sposobach uzyskania dostępu do danych oraz warunkach użycia ich w określonym celu. Najczęstszym podawanym przykładem metadanych jest katalog biblioteczny.

Dla poprawnego zarządzania metadanymi ważne jest aby były jednoznaczne w swej zawartości bez względu na to przez kogo zostały utworzone.

#### Rodzaje metadanych:

**Metadane wyszukiwania** - służą do wyboru zbiorów, mogą stać się przedmiotem zainteresowań danej osoby o konkretnych wymaganiach, obejmują one:

- nazwę i opis zbioru danych,
- podstawowe przeznaczenie i zakres stosowania danych,
- datę pozyskania danych i ich aktualizacji,
- producenta, dostawcę i głównych użytkowników danych,
- obszar, dla którego dane się odnoszą (współrzędne);
- nazwy geograficzne lub jednostki podziału administracyjnego;
- strukturę zbiorów i sposób dostępu do danych".

**Metadane rozpoznania**-zawierają więcej szczegółów o zbiorze, umożliwiają:

- ocenę jakości danych,
- określenie przydatności zbioru danych pod względem wymagań użytkowników,
- nawiązanie kontaktu z dyponentem danych celem uzyskania dalszych informacji, w szczególności informacji na temat warunków korzystania z danych".

**Metadane stosowania** - zawierają szczegóły zbioru, które gwarantują:"

- odczytania danych oraz ich transferu,
- interpretacji danych i praktycznego korzystania z nich w aplikacji użytkownika".

**Metadane konservatorskie**- jest to połączenie pozostałych rodzajów metadanych. Wspierają one długoterminowe przechowywanie materiałów cyfrowych.

#### Przykłady metadanych:

- data i czas utworzenia pliku;
- adres lub położenie geograficzne miejsca utworzenia pliku;
- imię i nazwisko, nazwa firmy, nazwa komputera lub adres IP;
- nazwy wszystkich współtwórców dokumentu lub dodane komentarze;
- typ użytego aparatu i jego ustawienia podczas robienia zdjęcia;
- typ użytego urządzenia rejestrującego audio lub wideo i jego ustawienia podczas nagrywania;
- marka, model i operator smartfona".

Wskaźniki określające jakość metadanych":

- kompletność (ang. completeness) dotyczy zarówno ilości i rozkładu elementów metadanych w rekordach z punktu widzenia zawartości pojedynczych rekordów, jak i rozkładu danych w bazie danych (poziom semantyczny). Analiza kompletności określa stopień realizacji głównych funkcji metadanych w systemie, w tym głównie identyfikowalności opisywanego źródła.

- poprawność (ang. correctness), która bywa traktowana synonymicznie z dokładnością (ang. accuracy), jest wskaźnikiem najtrudniejszym do oceny, przez co analizy oparte na tym wskaźniku nie poddają się automatyzacji (poziom pragmatyczny)." Przede wszystkim analizuje poprawność rekordów, wartości metadanych, pisownię (gramatyka, znaki specjalne bądź interpunkcje).
- spójność obejmuje pojedyncze bazy danych oraz współpracujących bibliotek cyfrowych. "Można mówić o spójności pragmatycznej zapisu elementu danych, tworzonych linkach między zasobami powiązanymi relacjami, identyfikatorów i identyfikacji, prezentacji wyników wyszukiwania (wyświetlania metadanych) oraz syntaktyki metadanych".
- analiza zdublowanych rekordów, które powstają w działaniach lokalnych oraz podczas wymiany/integracji danych.

**Zastosowanie metadanych:**

- lokalizacje danych;
- definicje danych związane z obsługiwanymi bazami danych i relacje;
- przepływy danych i związane z nimi procesy ETL;
- numery wersji metadanych oraz informacje o modyfikacjach;
- statystyki użycia danych;
- uprawnienia dostępu do danych.

**Korzyści stosowania metadanych:**

- szybsze uzyskanie informacji na temat zbiorów danych, dostępnych dla interesującego nas obszaru,
- łatwiejsze zarządzanie zasobami danych w obrębie organizacji,
- możliwość lepszego zaplanowania działań dotyczących aktualizacji danych,
- zwiększenie prawdopodobieństwa uniknięcia budowy zbiorów danych zgromadzonych już przez inne organizacje,
- poszerzenie kregu użytkowników danych

**43.** Wymień i omów układy współrzędnych stosowane przy wizualizacji danych.

**Układ współrzędnych kartezjańskich (prostokątny)** – prostoliniowy układ współrzędnych mający dwie prostopadłe osie.

Układem współrzędnych kartezjańskich w przestrzeni n-wymiarowej nazywa się układ współrzędnych, w którym zadane są:

- punkt zwany początkiem układu współrzędnych, którego wszystkie współrzędne są równe zeru
- ciąg n parami prostopadłych osi liczbowych zwanych osiami układu współrzędnych. Dwie pierwsze osie często oznaczane są jako:
  - (pierwsza os, zwana osią odciętych),
  - (druga, zwana osią rzędnych),
- Liczba osi układu współrzędnych wyznacza wymiar przestrzeni.

**Układ współrzędnych biegunowych** (układ współrzędnych polarnych) – układ współrzędnych na płaszczyźnie wyznaczony przez pewien punkt zwany biegunem oraz półprostą OS o początku w punkcie O zwaną osią biegunową.

Każdemu punktowi P płaszczyzny przypisujemy jego współrzędne bieguno we, jak następuje:

- promień wodzący punktu P do jego odległość  $|OP|$  od bieguna,
- amplituda punktu P to wartość kąta skierowanego pomiędzy półprostą OS a wektorem  $OP \rightarrow$

Dla jednoznaczności przyjmuje się, że współrzędne bieguna O są równe  $(0,0)$ . O amplitudzie możemy zakładać, że  $0 <= \phi <= 2\pi$  (niektórzy autorzy przyjmują  $-\pi <= \phi <= \pi$ ).

Ten układ jest używany rzadziej niż układ kartezjański, ale może być użyteczny w przypadkach, w których kąt lub kierunek jest ważny.

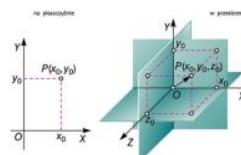
#### **Układ współrzędnych geograficzny**

Dane lokalizacji mają dodatkową zaletę połączenia ze światem fizycznym, co z kolei nadaje natychmiastowy kontekst i związek konkretnym punktom. Dzięki geograficznemu układowi współrzędnych możliwe jest zmapowanie tych punktów.

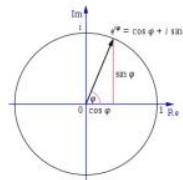
Dane o lokalizacji występują w wielu formach, ale najczęściej opisuje się je jako szerokość i długość geograficzną, które są kątami odpowiednio względem równika i południka zerowego. Czasami uwzględnia się również wysokość.

( Współrzędne geograficzne – szerokość i długość geograficzna mierzone w stopniach, minutach i sekundach kątowych. )

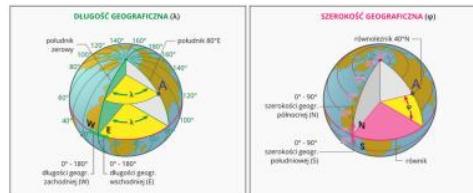
- Układ współrzędnych kartezjańskich



- Układ współrzędnych biegunowych – układ współrzędnych na płaszczyźnie wyznaczony przez pewien punkt O zwany biegunem i półprostą OS o początku w punkcie O zwaną osią biegunową.



- Układ współrzędnych geograficznych



DODATKOWE:

Źródło: <http://www2.agroparistech.fr/ufr-info/membres/cornuejols/Teaching/AGRO/UC-1A-explorer-data/Data-Points-Visualization-That-Means-Something.pdf>

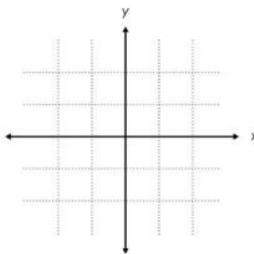
Istnieją trzy typy współrzędnych, które pokrywają większość zagadnień wizualizacji danych. Przedstawiono je na obrazie:

## Coordinate systems

There are a variety of them, from cylindrical to spherical, but these three will cover most of your bases.

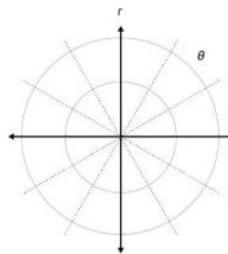
### Cartesian

If you've ever made a graph, the x- and y-coordinate system will look familiar to you.



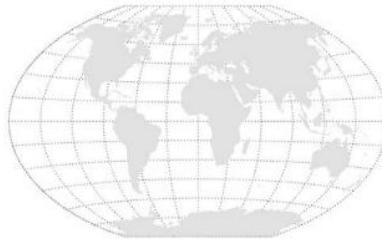
### Polar

Pie charts use this system. Coordinates are placed based on radius  $r$  and angle  $\theta$ .



### Geographic

Latitude and longitude are used to identify locations in the world. Because the planet is round, there are multiple projections to display geographic data in two dimensions. This one is the Winkel tripel.



---

### Układ kartezjański

Najczęściej używany układ w przypadku wykresów. Opierają się o niego np. wykresy słupkowe czy kropkowe. Pozwala na wizualizację nie tylko punktów  $(x,y)$ , ale także  $(x,y,z)$ . Warto pamiętać, że początek układu zaczyna się na przecięciu wszystkich osi (punkt  $(0,0)$  lub  $(0,0,0)$ ). Łatwy do wykorzystania ze względu na tak prostą możliwość umiejscowienia punktów, między którymi można wyznaczyć odległość następującym wzorem:

$$distance = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2}$$

---

### Układ polarny/Współrzędnych biegunowych

Wykorzystywany na przykład do wykresów kołowych. Składa się z okrągłej siatki, gdzie punkt po prawo to 0 stopni. Im większy kąt tym większe zwrócenie punktu w kierunku przeciwnym do ruchu wskazówek zegara.

Wykorzystywany rzadziej niż kartezjański, ale jest bardzo intuicyjny, gdy trzeba wskazać kąt lub kierunek.

### Układ geograficzny

Dane lokalizacyjne mają tą zaletę, że przekazują informację w przełożeniu na świat fizyczny (mapę). Dane lokalizacyjne przyjmują wiele form, jednak najpopularniejsza to

dane oparte o szerokość i długość geograficzną (lat,long), stanowiących kąty, które tworzone są przez punkt z równikiem lub południkiem zerowym. Czasem dodaje się trzeci wymiar, stanowiący elewację. W porównaniu ze współrzędnymi kartezjańskimi szerokość geograficzna jest jak oś pozioma, a długość geograficzna jest jak oś pionowa. To znaczy, jeśli używasz płaskiej projekcji. Trudną częścią mapowania powierzchni Ziemi jest to, że jest ona owinięta wokół kulistej masy, ale zwykle trzeba ją wyświetlić na dwuwymiarowej powierzchni, takiej jak ekran komputera. Różnorodność sposobów na to nazywa się projekcjami i jak pokazano na rysunku 3-14, każda z nich ma swoje zalety i wady. Kiedy rzutujesz coś, co jest trójwymiarowe na płaszczyznę dwuwymiarową, niektóre informacje są tracone, podczas gdy inne informacje są zachowywane.

Na przykład odwzorowanie Mercatora zachowuje kąty w lokalnych regionach. Został stworzony w XVI wieku przez kartografa Geradusa Mercatora głównie do nawigacji na morzach i nadal jest najczęściej używanym rzutem do wyszukiwania kierunku online. Z kolei rzut Albersa zachowuje obszar, ale zmienia kształta. Więc projekcja, którą wybierzesz, zależy od tego na czym chcesz się skoncentrować.

## Map projections

### Equirectangular

Typically used for thematic mapping, but doesn't preserve area or angle.



### Albers

Scale and shape not preserved; angle distortion is minimal.



### Mercator

Preserves angles and shapes in small areas, making it good for directions.



### Lambert conformal conic

Better for showing smaller areas and often used for aeronautical maps.



### Sinusoidal

Preserves area; useful for areas near the prime meridian.



### Polyconic

Was often used to show US in the mid-1900s; little distortion in small areas near meridian.



### Winkel Tripel

Minimized area, angle, and distance distortion; good choice for world map.



### Robinson

A compromise between preserving areas and angles; good to show world map.



### Orthographic

Representing a 3-D object in 2-D, need to rotate to area of interest.



**44.** Wymień i omów metody wizualizacji proporcji.

Dane dotyczące proporcji są pogrupowane, ale nie według czasu, ale według kategorii, podkategorii i populacji. Przez populację rozumiemy nie tylko populację ludzką, ale także wszystkie możliwe wybory lub wyniki. W ankiecie ludzie mogą zostać zapytani o to czy zgadzają się z pewnym stwierdzeniem, czy też nie, bądź też nie mają zdania na dany temat. Każda z kategorii reprezentuje coś z osobna, a suma części reprezentuje całość.

W przypadku proporcji zwykłe szukamy trzech rzeczy: maksimum, minimum oraz ogólnego rozkładu. W celu uzyskania dwóch pierwszych wystarczy jedynie posortować dane od największych do najmniejszych oraz wybrać dwa końce, które reprezentować będą największą oraz najmniejszą wartość. W przypadku ankiety może to dotyczyć najpopularniejszych i najmniej popularnych odpowiedzi udzielanych przez respondentów. W przypadku danych sprzedażowych będzie to najwyższa oraz najniższa wartość sprzedaży, itd. Rozkład proporcji również jest interesujący, ponieważ może pokazać jak różni się wybór jednej odpowiedzi od innych? Czy kalorie rozkładają się równomiernie na tłuszcz, białko i węglowodany, czy dominuje jedna grupa? Poniższe wykresy mogą okazać się podobne przy wizualizacji tego typu danych:

**1) Część całości** - Są to proporcje w najprostszej formie. Zestaw proporcji, które sumują się do 1 lub zestawu wartości procentowych, które sumują się do 100 procent. Wykres umożliwia jednocześnie pokazanie poszczególnych części w stosunku do innych, ale zachowuje też całkowitą wartość.

**2) Wykres słupkowy** - Każdy słupek na wykresie reprezentuje kategorię, a im dłuższy jest słupek, tym większa jest jego wartość. To, czy wyższa wartość jest lepsza czy gorsza, może oczywiście różnić się w zależności od zestawu danych i punktu widzenia.

**3) Wykres kołowy** - są bardzo stare i bardzo popularne, najczęściej widywane podczas prezentacji proporcji: zarówno w prezentacjach biznesowych, jak i w mediach. Pierwszy znany wykres kołowy został opublikowany przez Williama Playfaira, który wynalazł również wykres liniowy i wykres słupkowy w 1801 roku. Koło reprezentuje całość, a poszczególne wycinki koła, przypominające kawałki ciasta (stąd angielska nazwa pie chart) reprezentują części całości. W przypadku wartości procentowych, wszystkie części powinny sumować się do 100 procent. Wykresy kołowe mają swoich zwolenników jak i przeciwników - niektórzy zarzucają im, że nie są tak dokładne jak wykresy słupkowe więc niektórzy uważają, że powinni się ich całkowicie unikać. Nie znaczy to jednak, że nie można ich używać, jednak warto pamiętać o dobrych praktykach przy używaniu tego typu wizualizacji. Im więcej kategorii należy pokazać na wykresie, tym wykres kołowy staje się coraz mniej czytelny i w pewnym momencie może okazać się koniecznym rezygnacją z tego typu wizualizacji.

Aby uczynić wykres kołowy bardziej czytelnym, należy stosować praktyki podobne jak podczas tworzenie innych wizualizacji: dodanie tytułu w celu wyjaśnienia czytelnikom na co patrą oraz zmiana kolorów na przejrzyste i łatwo rozróżnialne ulepszając wizualnie wykres, podobnie jak posortowanie kawałków wykresu w rosnącej lub malejącej kolejności, zgodnie z ruchem wskaźnika zegara oraz nałożenie etykiet z wartościami na wykres. Należy pamiętać, że kolor może odgrywać ważną rolę w tym, jak ludzie czytają wykres. Nie jest tylko elementem estetycznym - może być wizualną wskazówką, np. do posortowanych wartości można zastosować kolory od najciemniejszych (największe wartości) do najjaśniejszych (najmniejsze wartości).

**4) Wykres z wyciętym kołem (donut chart)** - podobny do wykresu kołowego, ale z wyciętym otworem pośrodku, dzięki czemu wygląda jak pączek. Ponieważ w środku jest dziura, wartości nie oceniamy już pod kątem, ale używamy do tego długości łuku. Podobnie jak przy wykresie kołowym, wykres może być nieczytelny przy dużej ilości kategorii, ale w przypadkach z mniejszą liczbą kategorii pączka nadal może się przydać. Ten typ wykresu, dzięki wyciętemu kołu umożliwia umieszczenie w środku wykresu dodatkowej informacji, np. Sumę całosci, bądź też datę.

**5) Skumulowany wykres słupkowy z kategoriami** - ten typ wykresu można używać m. in. Do wizualizacji szeregow czasowych, ale świetnie sprawdzi się także do pokazania proporcji dla poszczególnych kategorii z podkategoriami, np. Umożliwi na uzyskanie informacji, który segment produktowy sprzedaje się najlepiej oraz która z podkategorii danego segmentu wpływa najbardziej na wynik (np. Elektronika sprzedaje się lepiej niż

Narzędzia domowe, a na wynik elektroniki najbardziej wpływa sprzedaż komputerów i telefonów). Podobnie jak w przypadku innych typów wizualizacji, warto zadać o dodanie tytułu wykresu, etykiet oraz dobranie odpowiednich kolorów.

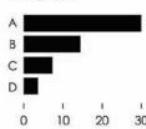
**6) Mapa drzewa** - Jest wykresem wykorzystywanym do prezentacji dużych zbiorów danych uporządkowanych hierarchicznie lub skategoryzowanych w sposób wymagający bardzo mało miejsca. Mapa drzewa wypełnia całkowicie dostępną przestrzeń przylegającymi prostokątami, których rozmiar reprezentuje zmienną liczbową. Hierarchie i kategorie są prezentowane jako prostokąty zawarte w innych prostokątach. Dane mogą być prezentowane nie tylko jako rozmiar prostokątów, ale także jako kolor, stanowiąc druga zmienna i wzmacniając tym samym obraz. Zewnętrzne prostokąty reprezentują kategorie nadrzędne, a prostokąty w obrębie kategorii nadrzędnej są podkategoriami (kategoria rodzic-dziecko).

**7) Wykres mozaikowy** - jest to dwuwymiarowy wykres złożony z prostokątów, podobny do mapy drzewa. Bywa inaczej nazywany wykresem macierzowym. Jego zaletą jest prezentacja dodatkowych danych liczbowych w trzecim wymiarze. Służy do prezentowania kilku poziomów hierarchii i złożonych danych. Cały prostokąt wykresu traktuje się jako populację, którą dzieli się na segmenty prezentujące w różnych wymiarach różne aspekty informacji. Samo wykonanie wykresu mozaikowego nie jest zbyt proste, ale jego wielką zaletą jest wielowymiarowa prezentacja danych.

## Categories

When your data is straightforward, with a value for each category, these are easy to read and create.

### Bar graph



With length as visual cue, useful for straightforward comparisons

### Symbol plot



Can be used in place of bars, but can be hard to see small differences

## Parts of a whole

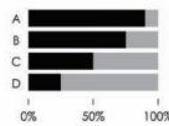
The categorical breakdown within a population can be interesting, and you might want to keep the groups together, although often not essential.

### Pie chart



Parts add to 100 percent, typically sorted clockwise for readability

### Stacked bar chart



Often used to show poll results and can also be used for raw counts

## Subcategories

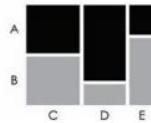
Data can have a hierarchical structure, which can be important in data interpretation and it often allows for different points of view.

### Treemap



Shows hierarchical structure in a compact space, area often combined with color

### Mosaic plot



Allows comparison across multiple categories in one view

**45.** Wymień i omów metody wizualizacji relacji.

Przy wizualizacji zmiennych można wykorzystać ich korelację lub rozkłady i na ich podstawie stworzyć m.in.:

**Wykres punktowy (scatter plot)** – przydatny przy wizualizacji niewielkiej liczby (dwóch lub trzech) zmiennych niezależnych. Wówczas dwie zmienne opisane są na osiach X i Y, a trzecią zmienną można przedstawić za pomocą koloru lub powierzchni ‘punktów’ wykresu (wykres bąbelkowy, bubble chart), jednak przy dużej liczbie obserwacji wykres może stać się nieczytelny, a powierzchnie trudne do porównania. W przypadku większej liczby zmiennych można zestawić je w parach, w postaci siatki (macierzy), wówczas możliwe jest sprawdzanie ich korelacji.

#### Histogram

**Diagram lodygowo-listkowy (stem-and-leaf plot, stemplot)** – tak samo jak histogram używany jest do wizualizacji kształtu rozkładu. Tworzy się go dla danych uporządkowanych, ta metoda sprawdza się lepiej, gdy obserwacji nie jest zbyt dużo.

**Mapa cieplna (heat map)** – przedstawienie danych w postaci zestawu kolorów; Każdej obserwacji odpowiada wiersz, a komórki przyjmują odpowiedni kolor w zależności od wartości cechy (np. im ciemniejszy kolor, tym wyższa wartość zmiennej).

**Wykres współrzędnych równoległych (parallel coordinates plot)** – również przedstawia dane poziomo, jednak w przeciwieństwie do map cieplnych, do ich porównania używa położenia, a nie koloru. Każda pionowa oś reprezentuje zakres jednej zmiennej (najwyższa wartość na górze, a najniższa na dole), a jedna linia odpowiada jednej obserwacji. Jeśli linie przebiegają równolegle, pomiędzy zmiennymi występuje pozytywna korelacja, a jeżeli linie przecinają się, można domniemywać korelację negatywną.

**Wykres radarowy (star plot, star chart, radar chart, spider chart)** – przedstawianie danych poprzez modyfikowanie długości osi o wspólnym początku. Każda oś reprezentuje jedną zmienną z minimalną wartością zmiennej w centrum, a wartościami najwyższymi na końcach (podobnie jak w wykresie współrzędnych równoległych).

Ja bym tu napisał:

- Wykres punktowy (scatter plot) i wszystko podobne (czyli liniowy etc)
- Heatmap
- Współrzędnych równoległych
- Bąbelkowy

Nie wiem co histogram ma do relacji, kiedy on pokazuje rozkłady.

## 46. Wymień i omów metody wizualizacji danych geolokalizacyjnych.

#### **Narzędzia do wizualizacji danych geolokalizacyjnych:**

Wraz ze wszystkimi danymi geograficznymi przedostającymi się do domeny publicznej pojawiło się także wiele narzędzi do mapowania tych danych. Niektóre wymagają tylko odrobinę programowania, aby coś uruchomić, podczas gdy inne wymagają nieco więcej pracy. Istnieje również kilka innych rozwiązań, które nie wymagają programowania.

##### **Mapy Google, Yahoo i Microsoft**

To najłatwiejsze rozwiązanie online; chociaż wymaga to trochę programowania. Im lepiej kodujesz, tym więcej możesz zrobić dzięki interfejsom API mapowania oferowanym przez Google, Yahoo i Microsoft.

##### **ArcGIS**

Wspomniane wcześniej usługi mapowania online są dość podstawowe w tym, co mogą zrobić. Jeśli chcesz bardziej zaawansowanego mapowania, najprawdopodobniej musisz samodzielnie zaimplementować tę funkcję. ArcGIS jest inny. To ogromny program, który umożliwia mapowanie dużej ilości danych i wykonywanie wielu różnych czynności, takich jak wygładzanie i przetwarzanie.

##### **Modest maps**

Modest Maps to biblioteka Flash i ActionScript do map opartych na kafelkach, pozwala na obsługę języka Python. Zabawną rzeczą w Skromnych Mapach jest to, że jest bardziej frameworkm niż interfejsem API, takim jak ten oferowany przez Google. Zapewnia absolutne minimum potrzebnych do utworzenia mapy online, a następnie pozwala na implementację tego, co chcesz.

#### **Metody wizualizacji danych:**

##### **Znajdź szerokość i długość geograficzną**

Przed wykonaniem jakiegokolwiek mapowania weź pod uwagę dostępne dane i dane, których faktycznie potrzebujesz. W większości praktycznych zastosowań potrzebujesz szerokości i długości geograficznej do mapowania punktów, a większość zestawów danych nie jest taka. Zamiast tego najprawdopodobniej będziesz mieć listę adresów. Nie możesz po prostu podłączyć nazw ulic i kodów pocztowych i oczekwać ładnej mapy. Najpierw musisz uzyskać szerokość i długość geograficzną, a następnie przejść do geokodowania. Zasadniczo bierzesz adres, przekazujesz go do usługi, usługa sprawdza bazę danych w celu dopasowania adresów, a następnie uzyskujesz szerokość i długość geograficzną w miejscu, w którym usługa myśla, że twój adres znajduje się na świecie. Jeśli chodzi o to, z której usługi skorzystać, jest ich wiele.

##### **Mapa z kropkami**

Teraz, gdy masz punkty o szerokości i długości geograficznej, możesz je mapować. Prostą drogą jest zrobienie komputerowego odpowiednika umieszczenia pinezek na papierowej mapie na tablicy. Jak pokazano w schemacie na ryc. 8-1, umieszasz znacznik dla każdej lokalizacji na mapie.

Chociaż jest to prosta koncepcja, w danych można zobaczyć takie funkcje, jak grupowanie, rozkładanie(spread) i wartości odstające.

W niektórych przypadkach przydatne może być połączenie kropek na mapie, jeśli kolejność punktów ma jakiekolwiek znaczenie. Wraz z rosnącą popularnością internetowych usług lokalizacyjnych, takich jak Foursquare, ślad lokalizacji nie są wcale takie rzadkie.

#### Skalowane punkty

Bajczęściel w danych nie znajduje się tylko lokalizacja. Istnieją także inne wartości przypisane do lokalizacji, takie jak sprzedaż dla firmy lub populacja miasta. Nadal możesz mapować za pomocą punktów, ale możesz wziąć zasady wykresu bąbelkowego i użyć go na mapie.

#### Regiony

Mapowanie punktów mogą doprowadzić Cię tylko do tego miejsca, ponieważ reprezentują tylko pojedyncze lokalizacje. Powiaty, stany, kraje i kontynenty to całe regiony z granicami, a dane geograficzne są zwykle agregowane w ten sposób. Na przykład znacznie łatwiej jest znaleźć dane dotyczące zdrowia obywateli dla stanu lub kraju niż dla poszczególnych pacjentów lub szpitali. Zwykle odbywa się to w celu zachowania prywatności, podczas gdy w innych przypadkach dane zagregowane są po prostu łatwiejsze do rozpowszechnienia. W każdym przypadku wizualizacji relacji przestrzennych zazwyczaj wykorzystasz swoje dane dokładnie w ten sposób.

#### Locations

A direct translation of latitude and longitude to two-dimensional space is straightforward and intuitive, but can pose challenges when there are a lot of locations.

##### Location map



Points represent locations and can be scaled by metric

##### Connections



Points can be connected to show relationships between locations

#### Regions

Oftentimes the density of individual points across regions is more informative than points on a map that can overlap.

##### Choropleth map



Defined regions colored by data and meaning can change based on scale

##### Contour map

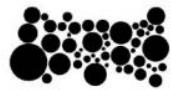


Lines show data continuously over geography, using density

#### Cartograms

Choropleth maps give large regions more visual attention, regardless of the data, so cartograms instead size regions by the data and ignore physical area.

##### Circular cartogram



Entire regions sized by data instead of physical area using shapes

##### Diffusion-based cartogram



Regions sized by data but boundaries stay connected

**47.** Wymień obiekty bazy danych i omów ich przeznaczenie.

<https://www.sqlpedia.pl/obiekty-w-bazach-danych/>

## Obiekty w bazie danych

- Schematy
- Tabele
- Widoki
- Ograniczenia (*constraints*)
- Sekwencje
- Procedury składowane
- Role
- Użytkownicy
- Wyzwalacze (*triggers*)
- Inne

Bazy danych, w szczególności w systemie Oracle rozpoznają obiekty zawarte w ramach danego schematu (schema), który sam w sobie często wymieniany jest jako osobna struktura w nomenklaturze bazodanowej. Każdy schemat jest własnością użytkownika bazy danych i najczęściej posiada taką samą nazwę jak użytkownik. W ramach schematu wyróżniamy między innymi następujące obiekty:

**Tabela** – jest to podstawowa jednostka służąca do przechowywania danych w bazie danych. Dane przechowywane są w rzędach i kolumnach, zaś sama tabela definiowana jest poprzez jej nazwę jak i poszczególne kolumny, te zaś definiujemy poprzez nazwę kolumny, jej typ oraz wielkość.

Przy tworzeniu obiektu tabeli określamy także jej ograniczenia, jak i potencjalne szyfrowanie danych w niej zawartych. Po stworzeniu obiektu możemy wypełnić go danymi, które tworzą kolejne rzędy tabeli przy użyciu kwerend w języku SQL.

**Perspektywa / widok** – jest to sztucznie stworzona prezentacja danych zawartych w jednej, bądź wielu tabeli. Perspektywa traktuje wynik danej kwerendy jako pełnoprawną tabelę, przez co często perspektywy traktuje się jako wirtualną tabelę, bądź zapisaną kwerendą. Perspektywy są szczególnie wykorzystywane w przypadkach potrzeby ograniczenia ilości informacji zawartych w tabeli, czy też w celu łatwego dojścia do danych zawartych w wielu tabelach, z pominięciem nieistotnych dla użytkownika.

Z racji bicia pochodnymi tabel, perspektywy posiadają wiele podobieństw z tabelami - możliwe jest ograniczenie wyświetlanego rekordów, tworzenie osobnych podzapytań do danej perspektywy oraz manipulacja danymi zawartymi w danej perspektywie.

**Widok zmaterializowany** – jest to obiekt służący do podsumowania, replikowania i przetwarzania danych. Ważną cechą widoków zmaterializowanych jest ich możliwość odświeżania. Stosuje się je najczęściej przy wyliczaniu agregatów - tam gdzie wyliczenie wyniku zajmuje dużo czasu, a dane źródłowe nie zmieniają się zbyt często. Wykorzystywane są one szczególnie w hurtowniach danych, środowiskach rozproszonych oraz przy wspieraniu decyzji w oparciu o dane.

W hurtowniach danych często nazywane są także podsumowaniami, gdyż stosuje się je głównie do przechowywania zagregowanych czy usrednionych danych.

W środowiskach rozproszonych służą do replikowania danych z rozproszonych źródeł przy jednoczesnej synchronizacji i update-owaniu danych. Tym samym umożliwiają lokalny dostęp do danych w innym przypadku znajdujących się jedynie w zdalnym dostępie.

**Synonim** – jest to innymi słowy alias nadawany innym obiektom i elementom baz danych, takim jak tabela, perspektywa, procedura, fukcja czy obiektom stworzonym przez użytkownika. Jako że synonim jest to jedynie alias, nie wymagają one żadnej alokacji pamięci poza ich definicją w słowniku danych.

Synonimy wykorzystywane są głównie w celach bezpieczeństwa i wygody. Dzięki nim możemy zamaskować właściciela danego obiektu, uprościć nazewnictwo wymagane w ramach zapytań SQL czy też ukazać lokalizacje poszczególnych obiektów w przypadku rozproszonych baz danych.

## 48. Wymień i omów metody wizualizacji szeregów czasowych.

Podczas wizualizacji danych szeregow czasowych celem jest sprawdzenie co już się wydarzyło, co się zmieniło i o ile oraz co pozostało takie samo. Na przykład, czy w porównaniu do ubiegłego roku jest mniej czy więcej? Jakie są możliwe wyjaśnienia wzrostu, spadku lub też braku zmiany. Dane czasowe można podzielić na dyskretne lub ciągłe. Znajomość kategorii, do której należą Twoje dane, może pomóc ci zdecydować, jak je wizualizować. W danych dyskretnych, wartości pochodzą z określonych punktów lub bloków czasu i istnieje skończona liczba możliwych wartości. Na przykład odsetek osób, które zdają test każdego roku, jest dyskretny. Ludzie przystępują do testu i to wszystko. Ich wyniki się już nie zmieniają, a test jest przeprowadzany w określonym dniu. Coś takiego jak temperatura jest jednak ciągłe. Można ją mierzyć o dowolnej porze dnia i dowolnym okresie i ciągle się zmienia.

Wizualizacja szeregow czasowych umożliwia także weryfikację czy powtarzający się na przestrzeni czasu schemat jest pozytywny czy negatywny, oczekiwany bądź też nieoczekiwany.

Podobnie jak w przypadku danych dotyczących proporcji, **wykres słupkowy** może posłużyć jako prosty sposób wizualizacji danych w czasie, z wyjątkiem tego, że zamiast kategorii na jednej z osi, umieszcza się czas. Do przykładowych wizualizacji szeregov czasowych można zaliczyć na przykład zmianę stopy bezrobocia na przestrzeni kilkunastu lat. Wizualną wskazówką na takim wykresie wysokość słupka. Im niższa wartość, tym krótszy będzie słupek. Im większa wartość, tym będzie wyższy.

Do wizualizacji szeregov czasowych można też użyć **skumulowanych wykresów słupkowych**, które przypominają wykresy słupkowe. Różnica polega na tym, że prostokąty są ułożone jeden na drugim. Stosuje się skumulowane wykresy słupkowe, gdy istnieją podkategorie, a ich suma jest znacząca.

Częściej bardziej interesująca jest zmiana wartości niż to, ile wynosi w danym punkcie czasu wartość wizualizowanej zmiennej, jednak w zależności od kontekstu biznesowego przydatne może okazać się analizowanie obu. Nabylenie zmian można jeszcze łatwiej zauważać używając do wizualizacji wykresu liniowego.

**Wykres kropkowy** także można zastosować w taki sam sposób jak wykres liniowy, gdyż dane i osie są takie same, zmienia się jednak wizualny odbiór wykresu. Czasami bardziej sensowne jest używanie punktów zamiast wykresów słupkowych czy też liniowych, ponieważ zajmują mniej miejsca. Na takim wykresie może być jednak trudniej zauważać trendy.

Innym typem wykresu, który również może posłużyć jako wizualizacja szeregov czasowych jest **wykres schodkowy**, to wykres liniowy składający się z tylko pionowych i poziomych linii, które łącząc się przypominają schodki. Pionowe linie wykresu obrazują wielkość powstałych zmian, natomiast poziome okres ich trwania. Sprawdźmy jak na przykładzie poniższych danych, wykres schodkowy różni się od liniowego.

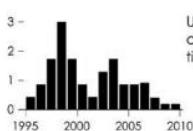
Wizualizacja danych ciągłych szeregow czasowych jest podobna do wizualizacji danych dyskretnych. Nadal istnieje dyskretna liczba punktów danych, nawet jeśli zestaw danych jest ciągły. Struktura ciągła i dyskretna jest taka sama. Różnica między nimi polega na tym, co reprezentują w świecie fizycznym. Jak poprzednio omówiono, ciągłe dane reprezentują stałe zmieniające się zjawiska. Gdy występuje dużo danych w zbiorze lub dane są zasumione, może być trudno dostrzec trendy i wzorce. Aby to ułatwić, można oszacować linię trendu. Linia trendu to linia, która przechodzi przez jak największej punktów i minimalizuje zsumowaną odległość od punktów do dopasowanej linii. Najłatwiejszą drogą jest stworzenie prostej linii za pomocą podstawowego równania nachylenia.

W niektórych przypadkach można zastosować mniej popularne typy wykresów. Na przykład, analizując dane dotyczące lotów i mając do dyspozycji dane dzienne, można analizować różnego rodzaju cykle poprzez użycie kalendarzowej mapy ciepła. Na takim wykresie bardzo łatwo wylapać obserwacje odstające, które będą oznaczały się najsilniejszym bądź najciemniejszym odcieniem wybranego koloru.

### Time series

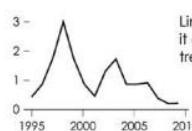
There are a variety of ways to see patterns over time, using cues such as length, direction, and position.

#### Bar graph



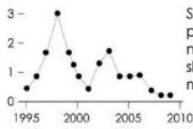
Useful for discrete points in time

#### Line chart



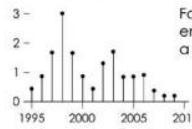
Lines can make it easier to see trends

#### Dot plot



Shows distinct points but might need line to show trend if not much data

#### Dot-bar graph

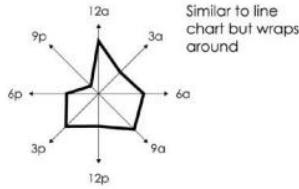


Focuses more on endpoints than a bar graph

### Cycles

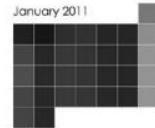
Time of day, day of the week, and month of the year repeat themselves, so it is often beneficial to align the segments in time.

#### Radial plot



Similar to line chart but wraps around

#### Calendar



Patterns for days of week seen more easily than views above

FIGURE 4-15 Visualizing time series data

**49.** Przedstaw, na czym polega uwzględnienie kontekstu a analizie danych.

Dwa podstawowe aspekty uwzględnienia kontekstu w analizie danych:

- Znaczenie danych – dane bez odpowiedniej warstwy kontekstu nie mają realnego znaczenia (np. Liczba 42 jako punkt w badanym zbiorze bez żadnego opisu i znajomości kontekstu ma nieskończonie wiele potencjalnych znaczeń). Z tego względu analiza wymaga zrozumienia genezy badanych danych, procesu który je generuje, przyjętych konwencji, a także sposobu zbierania danych (np. dane przekrojowe, panelowe). Dopiero po uwzględnieniu tych elementów analityk może próbować dokonywać interpretacji analizowanych danych. W celu przechowywania informacji o kontekście danych stosuje się tzw. Metadane (np. W formie nazw i opisów zmiennych), które mają pomóc zrozumieć przedmiot analizy.
- Organizacyjny – analiza danych zawsze odbywa się w pewnym otoczeniu, które warunkuje dostępne zasoby, środowisko danych, ale także cele powiązane z projektem. Rozpoznanie otoczenia organizacyjnego pod tym kątem pozwala określić jakie elementy wejściowe (dane, zasoby ludzkie, zasoby techniczne) są dostępne w ramach projektu analitycznego oraz to jakich elementów wyjściowych, czyli przede wszystkim nowej wiedzy czy też rezultatów biznesowych, od niej oczekujemy.

Uwzględnienie danych w podejściu procesowym: CRISP-DM

Eksploracja danych powinna być procesem niezawodnym i powtarzalnym, w którym obok analityków uczestniczą także menedżerowie, nie koniecznie posiadający wykształcenie z zakresu analizy danych.

Dlatego też w latach 90. opracowano standard procesu data mining, znany jako The Cross-Industry Standard Process for Data Mining CRISP-DM, który jest rezultatem wielu prac badawczych z udziałem ponad 300 organizacji. Model ten stanowi specyfikację działań, wyznacza wskaźniki i dobre praktyki w zakresie eksploracji danych. Według modelu CRISP-DM odkrywanie wiedzy z danych jest procesem cyklicznym składającym się z 6 etapów.

W ramach każdego z etapów zdefiniowane są zadania, które należy zrealizować. Model ten zakłada, że zbieranie i analiza danych nie mogą być prowadzone bez zrozumienia kontekstu funkcjonowania organizacji. Wskazuje także, że ostatecznym celem analizy danych jest poyskanie wiedzy pozwalającej na wdrożenie praktycznych rozwiązań. Założeniem modelu jest także płynność przechodzenia pomiędzy etapami. Poniżej rysunek przedstawia schemat modelu CRISP-DM, na którym uwzględniono współpracę zespołów biznesowego i analitycznego.

Pierwszym etapem wyróżnionym w modelu CRISP-DM jest zrozumienie potrzeb biznesowych. W trakcie tego etapu podejmowane są kluczowe decyzje mające wpływ na dalsze działania. Wyznaczanie zakresu projektu (project scoping) jest typem analizy ex-ante, która pozwala ocenić projekt zanim zostanie formalnie uruchomiony, ustalić na jakie pytania ma odpowiadać model i kto będzie odbiorcą analiz. Na tym etapie podjęta jest już decyzja o realizacji projektu, jednak należy zdecydować, jak projekt będzie zorganizowany, zwłaszcza jaki jest jego cel (na jakie pytana ma odpowiadać model i kto będzie jego odbiorcą), skala działania i kluczowe założenia (na przykład objęcie analizą określonych segmentów rynku, grupy klientów a wykluczenie innych), jakie są możliwe czynniki ryzyka niepowodzenia. Ważnym zadaniem jest także określenie zapotrzebowania na zasoby do realizacji projektu, w przypadku analizy danych jest to dobór źródeł danych, zewnętrznych bądź wewnętrznych a także zasoby ludzkie, czyli dobór członków zespołu.

## 50. Wyjaśnij co to jest system kontroli wersji na przykładzie systemu Git i zaproponuj typowy workflow.

**System kontroli wersji (ang. version control system)** – oprogramowanie służące do śledzenia zmian głównie w kodzie źródłowym oraz pomocy programistom w łączeniu zmian dokonywanych w plikach przez wiele osób w różnym czasie.

**Architektura:**

- scentralizowane (oparte na architekturze klient – serwer np. CVS – centralize version control system) – jedno centrum (każdy pobiera z centrum pliki). Wszystko może być stracone, bo wszystko w 1 pliku.
- rozproszone (oparte na architekturze P2P np. BitKeeper)

Git – rozproszony system kontroli wersji, który pozwala na prowadzenie równoprawnych, niezależnych gałęzi, które można dobrowolnie synchronizować ze sobą. Stworzony przez Linux. Pobieram całą historię kodu, robi się snapshot – zrzut aktualnego stanu (tworzenie plików) commit. Tworzenie kolejnych zdjęć programu (pobieranie u siebie lokalnie).

Cechy Git:

- wsparcie dla rozgałęzionego procesu
- praca off – line
- efektywna praca z dużymi projektami

**Rozwiązywanie workflow:**

Master and Develop – utworzenie 2 gałęzi, które powinny istnieć przez cały process wytwarzania oprogramowania.

Master – przechowuje wersję końcową

Develop – wersja developerska, taka do której wprowadzane są zmiany

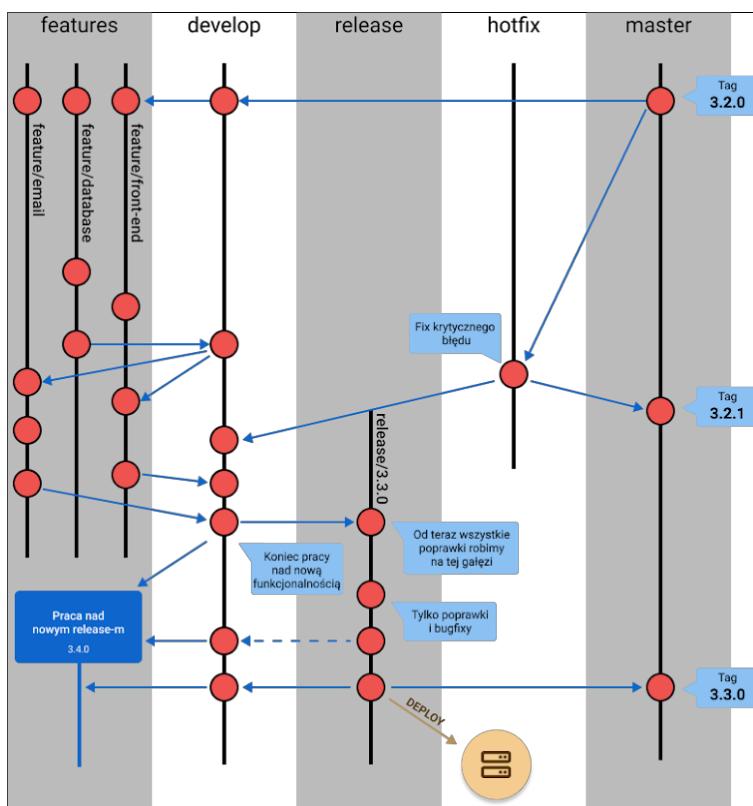
Po utworzeniu dwóch głównych linii rozwojowych, utworzenie linii pobocznych takich jak Feature, Hotfix, Release.

Feature – przechowywane są wszystkie nowe funkcjonalności oraz elementy, które mają być przetestowane przed wdrożeniem na produkcję.

Hotfix – wykonywane są w niej zadania, które związane są z naprawą błędów w aktualnej wersji.

Release – dodawanie notatek, aktualizowanie dokumentacji.

Po skończonej pracy należy scalić release z master i develop.

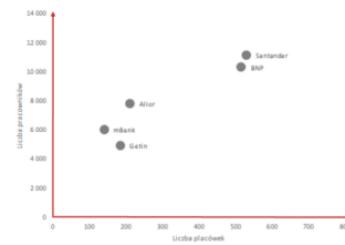


**51.** Omów wybraną technikę redukcji wymiaru danych, jej zalety i wady.

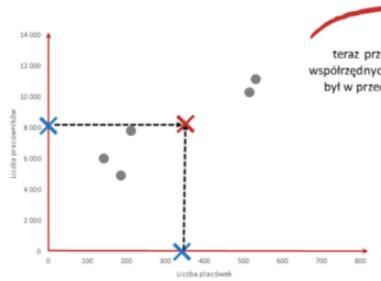
### Analiza głównych składowych

- polega na przekształceniu zmiennych wejściowych na nieskorelowane główne zmienne składowe (czynniki)
- każda z głównych składowych jest liniową zależnością zmiennych wejściowych
- główne składowe są tak uporządkowane, aby wariancja kolejnych głównych składowych była coraz mniejsza

	Allor	mBank	Getin	BNP	Santander
Liczba placówek	214	143	188	517	532
Liczba etatów	7 785	5 993	4 891	10 278	11 113



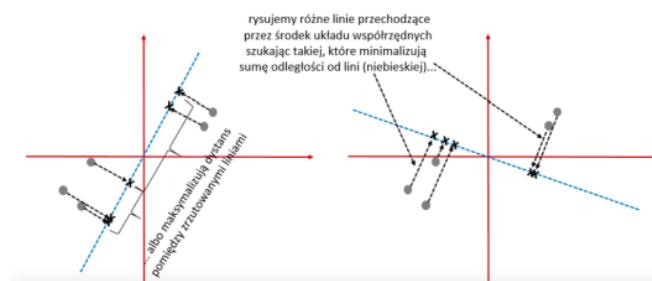
Wyliczamy średnie wartości dla x i y i zaznaczamy punkt wspólny, który musi być zawarty w prostej.



Minimalizujemy odległość pomiędzy punktami (znalezienie najlepszej prostej).

Minimalizujemy odległość pomiędzy punktami (znalezienie najlepszej prostej).

Teraz szukamy najlepszego rzutu na płaszczyznę.



Wady:

- Utrata danych
- Wartości odstające mogą znacznie zaburzać wyniki

## 52. Omów pojęcie obliczeń równoległych i podstawowe problemy, które pojawiają się przy obliczeniach równoległych.

Obliczenia równoległe są formą obliczeń, w której wiele instrukcji jest wykonywanych jednocześnie. To oznacza, że dwa lub więcej procesów współpracuje jednocześnie w celu rozwiązania pojedynczego problemu. Obliczenia równoległe stały się dominującym wzorcem w architekturze komputerowej, głównie za sprawą upowszechnienia procesów wielordzeniowych. Ze względu na skalę można wyróżnić obliczenia na poziomie:

- Bitów
- Instrukcji
- Danych

- Zadań

Jakie są problemy wynikające z obliczeń równoległych?

- Duży koszt
- Czekanie w nieskończoność (tzw. dead lock) aby dwie osoby nie czekaly na zrobienie tego samego zadania – rozwiązaniem może być ograniczenie czasowe lub wykonanie wpierw pierwszego procesu

Dodatek:

Zalety:

- Duże możliwości przyspieszenia wykonywania danych procesów.
- Wiele problemów rozwiązywanych w tym samym czasie.
- Odpowiednia implementacja zarówno ze strony software oraz hardware pozwala na optymalizację kosztów, gdyż maszyny zdolne do obliczeń równoległych są tworzone na bazie powszechnie dostępnych komponentów.
- Nowe możliwości w przetwarzaniu, w przypadku obliczeń sekwencyjnych nie ma możliwości rozwiązania niektórych problemów.

Wady:

- Konieczność pisania odpowiedniego oprogramowania, które umożliwia przetwarzanie równolegle.
- Dodatkowy czas poświęcany na transfer danych, synchronizację, komunikację etc.
- Technika: lepsze chłodzenie dla maszyn umożliwiających takie obliczenia, wysoki pobór prądu.
- Rozwiązania oparte o zrównoleglenie są trudniejsze w implementacji, debugowaniu, a także ciężko udowodnić poprawność ich działania.

### 53. Omów pojęcie estymatora odpornego na wybranych przykładzie.

**Estymator** – statystyka służąca do szacowania wartości parametru rozkładu nienormalnego. Rolę estymatora może pełnić mediana. Z założenia estymatory zakładają pewien rozkład normalny zmiennych. W praktyce rozkład zawiera dane odstające, które zaburzają wyniki obliczeń. Stąd jednym z odpornych estymatorów na dane odstające jest **średnia ucincana**. Bazuje ona na usunięciu danych odstających i policzeniu średniej z otrzymanych wartości w zbiorze.

Ogólnie rzecz biorąc, estymatory wyliczane są najczęściej na podstawie średniej. Działa to dobrze w przypadku, gdy estymatory są wyliczane dla danych o rozkładzie normalnym lub do niego zbliżonym. Jednakże dane nigdy nie są idealne, także zawierają wartości odstające, czy też ich rozkład nie jest nawet zbliżony do normalnego. Powoduje to, że zwykłe, nieodporne estymatory nie spełniają swojej roli. Estymatory odchyleń standardowych czy współczynniki regresji liniowej zwykle mają tę cechę, że obserwacje bardziej oddalone od średniej są bardziej wpływowne, czyli wchodzą z większą wagą do wyniku. W ten sposób nawet jedna obserwacja może zakłócić wynik analizy.

Średnia, mediana, odchylenie standardowe i rozstęp międzykwartylowy to przykładowe statystyki, które szacują odpowiadające im wartości populacji. Idealnie, wartości próbki będą stosunkowo zbliżone do wartości populacji i nie będą systematycznie zbyt wysokie ani zbyt niskie (tj. bezstronne).

Niestety, wartości odstające i wartości ekstremalne w długim ogonie rozkładu skośnego mogą spowodować, że niektóre statystyki próbek staną się stronnicze, oszacowania niskiej jakości. Co to znaczy? Statystyki próbki będą systematycznie zbyt wysokie lub zbyt niskie i oddalają się od prawidłowej wartości.

Zatem statystyka odpornościowa będzie efektywna, będzie mieć tylko niewielkie obciążenie i będzie asymptotycznie nieobciążona wraz ze wzrostem próby kiedy są w niej wartości odstające i wartości ekstremalne w długich ogonach.

Gdy wartości odstające i długie ogony rozkładu są obecne, statystyka odpornościowa będzie dość zbliżona do poprawnej wartości, biorąc pod uwagę rozmiar próby i nie będzie

*systematycznie zawyjać lub zaniżać wartości dla populacji. Wraz ze wzrostem próby, podejście statystyczne stają się w pełni nieobciążone.*

*Statystyki odpornościowe opierają się wpływowi długich ogonów i wartości odstających. Działają dobrze w wielu różnych rozkładach prawdopodobieństwa, zwłaszcza nie-normalnych rozkładach.*

*Statystyką odpornościową jest na przykład mediana, która ma punkt przełamania na poziomie 50%. Oznacza to, że można zmodyfikować aż do 50% złamanych wartości nim wartość mediany będzie przekłamana. Dla porównania średnia ma punkt przełamania na poziomie 0%.*

*W przypadku estymatorów odpornych dla wariancji, nie można używać odchylenia standardowego. To ten sam przypadek co średnia, zastępując nawet jedną wartość w próbie, wartość tej miary się zmienia. W tym przypadku można zastanowić się nad użyciem IQR. Rozstęp międzykwartylowy (IQR) to środkowa połowa zbioru danych. Jest podobny do mediany pod tym względem, że można zastąpić wiele wartości bez zmiany IQR. Ma punkt załamania 25%. W konsekwencji, spośród tych trzech miar, rozstęp międzykwartylowy jest najbardziej solidną statystyką.*

*Solidne analizy statystyczne mogą dawać prawidłowe wyniki nawet wtedy, gdy nie istnieją idealne warunki dla danych ze świata rzeczywistego. Analizy te działają dobrze, gdy przykładowe dane mają różne rozkłady i mają nietypowe wartości. Innymi słowy, możesz ufać wynikom nawet wtedy, gdy założenia nie są w pełni spełnione.*

*Na przykład testy hipotez parametrycznych, które oceniają średnią, takie jak testy t i ANOVA, zakładają, że dane mają rozkład normalny. Jednak dzięki centralnemu twierdzeniu granicznemu testy te są odporne na odchylenia od rozkładu normalnego, gdy wielkość próbki na grupę jest wystarczająco duża.*

*Pamiętaj, aby wiedzieć, dla których właściwości każda analiza statystyczna jest odporna. Na przykład, podczas gdy tradycyjne testy t i ANOVA radzą sobie z naruszeniami założenia o normalności, nie są w stanie oprzeć się skutkom wartości odstających. Testy nieparametryczne nie wymagają określonego rozkładu, ale różne grupy w analizie muszą mieć taki sam rozrzuć. Dlatego testy nieparametryczne nie są odporne na naruszenie założenia równych wariancji.*

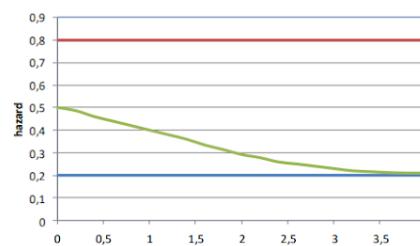
*W przypadku zagadnień ahz istnieje możliwość użycia estymatorów kanapkowych, które mogą być wykorzystywane w momencie, gdy występują zdarzenia powtarzalne. W przypadku istnienia takich zdarzeń w badaniu konieczna jest korekta błędów oszacowań, statystyk testowych, współczynników czy też funkcji hazardu. Do korekty błędów oszacowań wykorzystuje się odporne błędy standardowe - tzw. estymatory kanapkowe ABA, gdzie A jest odwróconą obserwowałą macierzą informacji. (Nazwa kanapka pochodzi od wzoru macierzowego dla solidnego estymatora wariancji-kowariancji, który ma ogólną postać ABA, gdzie A jest odwrotnością obserwowanej macierzy informacji).*

*Technika ta jest czasami określana jako metoda uśredniania populacji. Wykorzystywane to jest np. w przypadku procedury PHREG, estymującej modele proporcjonalnych szans Cox'a. Stosując i nie stosując owych estymatorów, można zauważać nalożenie korekty na błędy standardowe, zatem też na test chi-kwadrat i p-value dla konkretnej zmiennej. Nie nakłada ona jednak korekty na błędy systematyczne we współczynnikach, które wynikają z nieobserwowej heterogeniczności. Zwykle wartości testów chi-kwadrat będą mniejsze niż w przypadku braku zastosowania takich estymatorów. Zatem p-value w tym przypadku będzie wyższe.*

Konieczność zastosowania takich estymatorów wynika z nieobserwowej heterogeniczności. W przypadku regresji liniowej, heterogeniczność jest zawarta w składniku resztowym, jednak dla modelu Coxa nie jest to możliwe. Heterogeniczność ta prowadzi błędnych funkcji hazardu i współczynników przy zmiennych, a w przypadku zdarzeń powtarzalnych prowadzi także do korelacji pomiędzy zdarzeniami (obserwacjami), co w efekcie prowadzi do zniekształconych błędów oszacowań i statystyk testowych.

## Nieobserwowa heterogeniczność

- W ekstremalnej sytuacji nieobserwowa heterogeniczność może doprowadzić do sytuacji kiedy estymowany hazard w populacji będzie malejący pomimo że prawdziwy hazard w populacji nie będzie malejący dla żadnej z obserwacji.



## 54. Omów technikę regularyzacji na wybranym przykładzie, np. regresji LASSO.

**Regularyzacja** – nadanie ograniczeń wariancji modelu poprzez ograniczenie wartości współczynników modelu predyktacyjnego. Regularyzacja to jeden ze sposobów zapobiegania przeuczenia modelu tzw. overfitting, czyli nadmierнемu dopasowaniu modelu.

**Lasso w regresji liniowej** – kryterium zawiera sumę modułów wag jako składnik kary. W tej metodzie wag mogą się zerować przy odpowiednio dużych wartościach lambda, całkowite pozbycie się nieistotnych parametrów. W modelu parametry są liczone z wartości bezwzględnej.

1. Lasso umożliwia (relatywnie często przypadek) usuwanie poszczególnych pojedynczych zmiennych z modelu (tj.  $\beta_i = 0$ ).
2. W przypadku regresji grzbietowej, wartości parametrów są rzadko kiedy redukowane do dokładnie 0 (ale mogą do niskich wartości).
3. Lasso umożliwia bardziej intuicyjną interpretację parametrów modeli.
4. W przypadku regresji grzbietowej, można zaobserwować zmianę znaku.
5. W lasso, zmiany wielkości parametrów są liniowe względem zmiany parametru kary.
6. Regresja grzbietowa nakłada większą karę na większe wartości parametrów modelu.

7. W przypadku równych wartości parametrów (lub bardzo skorelowanych) regresja grzbietowa zwróci równe wagę, a lasso zwróci parametry o tych samych znakach, ale nie dokładnie tej samej wartości (suma wartości parametrów będzie stała) – w lasso może pojawić się problem identyfikowalności zmiennych.

8. Obie metody umożliwiają redukcję wariancji parametrów modelu.

LASSO, znane także jako L1 albo *Least Absolute Shrinkage and Selection Operator* jest najczęściej stosowaną metodą regularizacji regresji liniowej, w której wprowadzenie członu kary w miejsce dużych wartości bezwzględnych parametrów skutkuje uzyskaniem estymatorów o mniejszej wariancji, kosztem ich obciążenia.

$$LASSO = \left( \sum_{i=1}^n y'_i - \beta'_0 - \sum_{j=1}^p \beta'_j x'_{ij} \right) + \lambda \sum_{j=1}^p |\beta'_j| \quad (1)$$

$$\sum_{j=1}^p |\beta'_j| \leq \lambda \quad (2)$$

gdzie  $y'_i$  oznacza standaryzowaną  $i$ -tą zmienną objaśnianą,  $\beta'_0$  – standaryzowany wyraz wolny,  $\beta'_j$  – standaryzowane współczynniki regresji dla standaryzowanej kowarianty  $j$ ,  $x'_{ij}$  oznacza  $i$ -tą standaryzowaną zmienną dla  $j$ -tej standaryzowanej kowarianty,  $n$  – liczba obserwacji,  $p$  – ilość kowariant,  $\lambda$  – parametr penalizujący.

Wartość /Lambda dostrajana jest z poziomu danych w ten sposób, że im większa wariancja w danych, tym większa kara. Główna trudność leży właśnie w ustaleniu odpowiedniego parametru kary. Stosuje się tu ocenę błędu predykcji przez walidację krzyżową lub kryteria informacyjne. Duże wartości lambd powodują, że zmniejszają się wartości bezwzględne współczynników i więcej z nich zmierza do zera, zmniejszając tym samym wariancję kosztem obciążenia. Wartość współczynników jest ściągana do zera z siłą użależnioną od wartości lambda.

Kary nakładane na parametry są stałe i nie zależą od oszacowanej wartości parametru. Dzięki temu LASSO zachowuje relację pomiędzy zmiennymi, która jest wynikiem oszacowania przy pomocy KMNK.

Regularyzacja ta ma jednak wady. W jej wyniku możliwe jest wygenerowanie modelu, który jest całkowicie pozbawiony sensu i odrzuca zmienne, które są ważne w badanym problemie. W szczególności w przypadku w którym w modelu występują dwie (lub więcej) zmienne silnie współliniowe regresja LASSO losowo wybierze tylko jedną z nich, nie jest też dobrą metodą do oszacowania modelu, gdy jest mniej obserwacji niż zmiennych.

## 55. Co oznacza określenie 3V oraz 5V w kontekście problematyki Big Data?

Koncepcja zjawiska przedstawiona została na początku lat 2000 przez Douga Laneya. Pierwotnie, według tej koncepcji dane masowe podlegały zasadzie trzech „V” :

- **Volume** – ilość danych. Dane, pozyskiwane są z wielu źródeł, takich jak media społecznościowe, sensory, urządzenia mobilne, strony internetowe. Dane analizowane są w czasie rzeczywistym, zapisywane są wyłącznie informacje kluczowe. Co do wielkości zbioru danych jaka kwalifikuje go do analizy big data, nie ma jednoznacznej definicji. Big data to duże zbiory cyfrowych danych, których celem przetwarzania jest zdobycie nowych informacji lub wiedzy. Nowoczesne technologie upraszczają magazynowanie danych tej wielkości.
- **Velocity** – szybkość przepływu danych. Wzrostowi ilości danych towarzyszy przyrost szybkości tworzenia danych oraz ich wykorzystania. Wymusza to obsługę danych z odpowiednim reżimem czasowym. Nastuchiwany jest strumień danych i w przypadku określonych zdarzeń podejmowane są niezwłoczne działania. Podstawą biznesową wykorzystania technologii big data jest korzystanie z najbardziej aktualnych informacji. Duża prędkość sprawia, że dane są bardzo nietrwałe, kolejne aktualizacje skutkują przedawnianiem się poprzednich wersji danej informacji.
- **Variety** – różnorodność. Dane przesypane są w różnych formatach, ustrukturyzowanych i nieustrukturyzowanych. Do danych ustrukturyzowanych zaliczamy dane numeryczne, klasyczne bazy danych, a do nieustrukturyzowanych wiadomości e-mail, pliki audio, video i tekstowe, a także dane transakcji finansowych.

Z czasem rozszerzono koncepcję o kolejne dwa "V":

- **Veracity** (wiarygodność) Konsekwencją pracy z dużą ilością danych jest ryzyko przekłamań danych. Duża ilość spływających danych negatywnie wpływa na ich jakość, powstaje szum informacyjny. Big data odpowiada za zarządzanie wiarygodnością danych dla ich użytkowników.
- **Value** (wartość), określa cel gromadzenia tak dużej ilości danych, a także znalezienie powiązań jawnych bądź ukrytych dla budowania nowej zebranych danych. Na podstawowym poziomie dane nie posiadają żadnej wartości wewnętrznej. Stają się pożyteczne tylko gdy służą rozwiązaniu określonego problemu lub zaspokojeniu określonej potrzeby.

## 56. Wyjaśnij pojęcia danych ustrukturyzowanych i nieustrukturyzowanych.

- **Ustrukturyzowane** (dane numeryczne, relacyjne bazy danych, tekst, html, strumienie danych)
- **Nieustrukturyzowane** (pliki audio, video, mail)

Wszystkie algorytmy uczenia maszynowego wymagają danych ustrukturyzowanych zapisanych w tabelarycznej postaci. Zorganizowane są one w kolumnach cech charakteryzujących każdą obserwację (wiersze). Przykładem mogą być takie cechy jak: płeć, wzrost czy ilość posiadanych samochodów, na podstawie których można przewidywać czy klient będzie spłacał kredyt czy też nie. Takie przewidywanie również oznaczać jest jako cecha. Zmienne te dobrane są tak, by łatwo można je było pozyskać. Dzięki tak otrzymanym tabelom cech możemy stosować algorytmy XGBoost lub regresji logistycznej w celu wyznaczenia odpowiedniej kombinacji zmiennych wpływających na prawdopodobieństwo dobrego albo złego klienta.

Dane nieustrukturyzowane to takie, które nie są ułożone w tabelarycznej postaci. Przykładem może być dźwięk, obraz czy tekst. W procesie przetwarzania zawsze

przetworzone zostają one na jakąś formę wektorową. Jednak poszczególne litery, częstotliwości - czy piksele nie niosą ze sobą żadnych informacji. Nie tworzą osobnych cech, co jest kluczowe dla odróżnienia ich od danych ustrukturyzowanych.

Ustrukturyzowane dane zazwyczaj składają się z jasno zdefiniowanych informacji (takich jak tekst i cyfry), które można z łatwością wyszukiwać i utrzymywać lub śledzić w zorganizowanych tabelach lub bazach danych. W międzyczasie nieustrukturyzowane dane są umieszczone w różnych plikach lub formatach nośników, które nie są wewnętrznie odpowiednio pogrupowane lub sklasyfikowane.

Dane zawarte w bazach danych, dokumentach, wiadomościach e-mail i innych plikach danych do analizy predykcyjnej można sklasyfikować jako dane strukturalne lub nieustrukturyzowane. Zbudowany dane są dobrze zorganizowane, mają spójną kolejność, są stosunkowo łatwe do przeszukiwania i wyszukiwania oraz mogą być łatwo dostępne i zrozumiałe dla osoby lub programu komputerowego.

Klasycznym przykładem uporządkowanych danych jest arkusz kalkulacyjny programu Excel z etykietami kolumn. Takie uporządkowane dane są spójne; nagłówki kolumn - zwykle krótkie, dokładne opisy treści w każdej kolumnie - mówią dokładnie, jakiego rodzaju treści możesz się spodziewać. Dane strukturalne są zwykle przechowywane w dobrze zdefiniowanych schematach, takich jak bazy danych. Zwykle jest tabelaryczny z kolumnami i wierszami, które jasno określają jego atrybuty.

Brak struktury z drugiej strony dane są zwykle nieformalne, nie tabelaryczne, rozproszone i niełatwe do odzyskania; takie dane wymagają celowej interwencji, aby nadać im sens.

Różne wiadomości e-mail, dokumenty, strony internetowe i pliki (tekstowe, dźwiękowe i / lub wideo) w rozproszonych lokalizacjach są przykładami danych nieustrukturyzowanych. Trudno jest skategoryzować zawartość nieustrukturyzowanych danych. Zwykle składa się głównie z tekstu, zwykle tworzy go mieszanina dowolnych stylów, a znalezienie atrybutów, których można użyć do opisania lub grupowania, nie jest łatwym zadaniem.

Ogólnie rzecz biorąc, na świecie występuje wyższy odsetek danych nieustrukturyzowanych niż danych strukturalnych. Dane nieustrukturyzowane wymagają więcej pracy, aby były użyteczne, dlatego przyciągają więcej uwagi - w związku z tym zwykle pochłaniają więcej czasu.

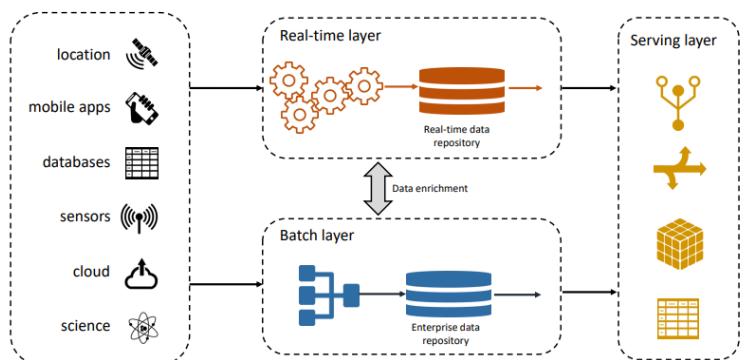
Porównajmy	dane strukturalne	nieustrukturyzowane.
Charakterystyka	Zbudowany	Brak struktury
Stwarzyszenie	Zorganizowany	Rozproszone i rozproszone
Wygląd	Formalnie zdefiniowane	Dowolna forma
Dostępność	Łatwy dostęp i zapytania	Trudny dostęp i zapytania
Dostępność	Procentowo niższe	Procentowo wyżej
Analiza	Skuteczna analiza	Wymagane jest przetwarzanie
wstępne		

Dane nieustrukturyzowane nie są całkowicie pozbawione struktury - wystarczy je znaleźć. Nawet tekst w plikach cyfrowych nadal ma pewną strukturę, która jest z nim związana, często widoczna w metadanych - na przykład tytuły dokumentów, daty ostatniej modyfikacji plików i nazwiska ich autorów.

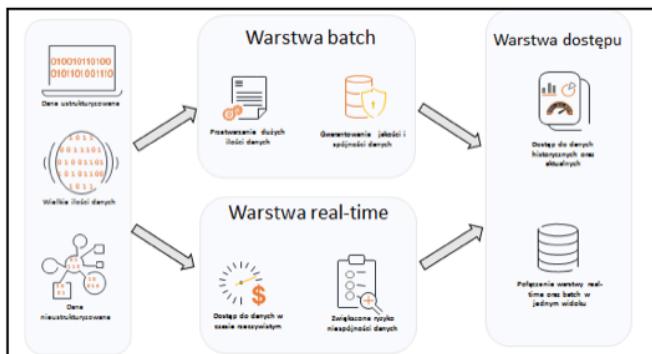
To samo dotyczy wiadomości e-mail: zawartość może być nieustrukturyzowana, ale są z nią powiązane dane ustrukturyzowane - na przykład data i godzina wysłania, nazwy nadawców i odbiorców, czy zawierają załączniki.

## 57. Przedstaw architektury: Lambda i Kappa.

### Architektura lambda



### Architektura Lambda



### Architektura lambda

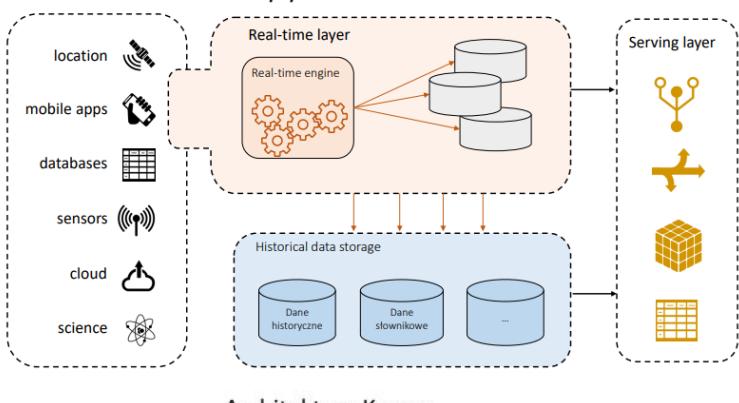
Założenie lambda polega na stworzeniu dwóch osobnych przepływu, gdzie jeden jest odpowiedzialny za przetwarzanie w trybie wsadowym, a drugi za dostęp do nich w czasie rzeczywistym. Stale napływający strumień danych jest kierowany do obu warstw.

1. **Warstwa batch** – Obliczenia wykorzystywane na całym zbiorze danych. Odbiera się to kosztem czasu, ale otrzymane w zamian dane zawierają pełną historię i wysoką jakość. Zakłada się, że zbiór danych znajdujący się w warstwie batch ma formę niepodzielną, którą należy jedynie rozszerzać, aniżeli usuwać z niej dane. W ten sposób można zapewnić pełną historyzację i spójność danych.

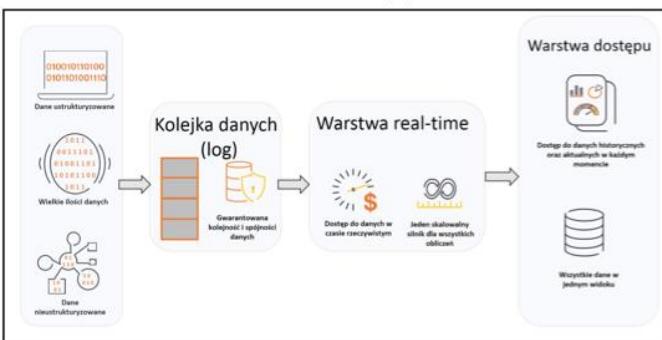
2. **Warstwa real-time** przetwarza napływające dane w trybie rzeczywistym. Oferuje ona niski czas dostępu do danych, co przekłada się na możliwość szybszego pozyskania informacji. Niestety, brak dostępu do danych historycznych sprawia, że nie wszystkie obliczenia są możliwe do wykonania. Często jakość oraz wiarygodność danych pochodzących z warstwy real time nie jest tak wysoka, jak z warstwy batch. Te drugie należy uważać za bardziej wiarygodne, ale ze względu na dłuższy czas potrzebny na ich załadowanie, warstwa real-time okazuje się nieocenioną pomocą, chcąc zagwarantować możliwość przetwarzania danych w trybie rzeczywistym.
3. **Warstwa dostępu** jest miejscem, w którym tworzone są widoki na podstawie warstwy batch oraz real-time. Dane są agregowane w taki sposób, aby końcowy użytkownik widział je jako jedną, spójną całość, aniżeli dwa niezależne, całkiem odrębne systemy. Widoki powinny być przygotowane w taki sposób, aby zapewnić możliwość wykonywania wszelkiego rodzaju analiz ad-hoc, ciesząc się przy tym szybkim dostępem do danych.

Koncepcja architektury Lambda zapewnia wiele zalet, przede wszystkim doskonale kompromis między przetwarzaniem wsadowym i real-time. Największą i najczęściej wspominaną wadą jest konieczność utrzymywania dwóch niezależnych aplikacji – jednej do zasilania warstwy batch, natomiast drugiej do warstwy real-time. Narzędzia wykorzystywane w poszczególnych warstwach różnią się między sobą, więc ciężko jest dobrać jedno, które może być wykorzystane do dwóch celów. Jest to możliwe w przypadku Apache Spark, gdzie po zdefiniowaniu logiki, według której dane mają być przetwarzane, można je wywołać w trybie batchowym, jak i w trybie Spark Streaming, który ze strumienia danych wejściowych tworzy tzw. micro-batch i pozwala na przetwarzanie ich w czasie niemal rzeczywistym

## Architektura kappa



Architektura Kappa



Architektura kappa – odpowiedź na krytykę Lambda

W roku 2014 Jay Kreps w swoim blogu wprowadził termin architektury Kappa jako odpowiedź na krytykę związaną z implementacją oraz utrzymaniem systemów opartych o architekturę Lambda. Nowa architektura została oparta o cztery główne założenia:

**Wszystko jest strumieniem** – strumień stanowi nieskończoną ilość skończonych paczek danych (batchów). Stąd każdy rodzaj zasilienia danymi może być uważany za strumień.

**Dane są niezmienne** (immutable) – dane surowe są persystowane w oryginalnej postaci i nie zmieniają się, dzięki czemu w każdej chwili można ich użyć ponownie.

**Reguła KISS** – Keep it short and simple. W tym przypadku przez użycie wyłącznie jednego silnika do analizy danych zamiast kilku, jak w przypadku architektury Lambda.

**Możliwość odnowienia stanu danych** – kalkulacje i ich rezultaty mogą być odświeżane przez odtwarzanie danych historycznych i aktualnych bezpośrednio z tego samego strumienia danych w każdym momencie.

Krytyczne dla powyższych reguł jest zapewnienie niezmiennej i oryginalnej kolejności danych w strumieniu. Bez spełnienia tego warunku nie jest możliwe uzyskanie konsystentnych (deterministycznych) wyników obliczeń. Podobnie jak w architekturze Lambda mamy warstwę Real-Time i warstwę Dostępu, które pełnią tutaj te same funkcje. Natomiast brak jest warstwy Batch, która stała się zbędna ponieważ historię można w każdej chwili odtworzyć ze strumienia danych w warstwie Dostępu przy pomocy identycznego silnika przetwarzania danych.

## 58. Przedstaw kluczowe cechy uczenia i predykcji w trybie wsadowym (offline learning) i przyrostowym (online learning).

Podział ze względu na możliwość trenowania przyrostowego przy użyciu strumienia nadsyłanych danych:

**Uczenie wsadowe** - systemy wykorzystujące wszystkie zapisane dane. Zajmuje dużo czasu i zasobów. Często nazywane przetwarzaniem w trybie off-line. System taki jest wpierw uczyony a następnie wdrażany do cyklu produkcyjnego i już więcej nie jest trenowany! . Wymiana modelu odbywa się po wytrenowaniu nowego modelu dla wszystkich danych - proces ten łatwo zautomatyzować.

**Uczenie przyrostowe** nazywane również procesem on-line model jest trenowany dla sekwencyjnie dodawanych (nowych) danych. Co zrobić gdy napływanające dane przestają być prawidłowe? (detekcja anomalii).

	Online machine learning	Offline machine learning
<b>Complexity</b>	More complex because the model keeps evolving over time as more data becomes available.	Less complex because the model is fed with more consistent data sets periodically.
<b>Computational power</b>	More computational power is required because of the continuous feed of data that leads to continuous refinement.	Fewer computational power is needed because data is delivered in batches; the model isn't continuously refining itself.
<b>Use in production</b>	Harder to implement and control because the production model changes in real-time according to its data feed.	Easier to implement because offline learning provides engineers with more time to perfect the model before deployment.
<b>Applications</b>	Used in applications where new data patterns are constantly required (e.g., weather prediction tools)	Used in applications where data patterns remain constant and don't have sudden concept drifts (e.g., image classification)

## 59. Podaj przykład i omów w jakich sytuacjach wskazane jest zastosowanie modelu przetwarzania OLTP.

Ad 60.

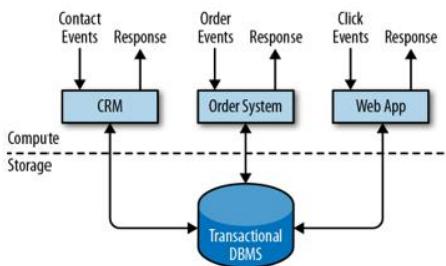
## 60. Podaj przykład i omów w jakich sytuacjach wskazane jest zastosowanie modelu przetwarzania OLAP.

OLAP (On – line Analytical Processing, przetwarzanie analityczne)

OLTP (On – line Transactional Processing, przetwarzanie transakcyjne)

OLTP	OLAP
przetwarzanie bazy danych bieżącej	tworzenie raportów
maksymalna wydajność	-
wymagany jest dostęp do aktualnych danych	dane mogą być dostępne z opóźnieniem
tworzenie prostych zapytań w dużych ilościach	niewielka liczba zapytań
np. systemy bankowe obsługujące salda rachunków klientów	np. raporty dynamiki sprzedaży produktów

**Model tradycyjny** - przetwarzanie transakcyjne w trybie on-line, **OLTP** (on-line transaction processing). Świeśnie sprawdza się w przypadku obsługi bieżącej np. obsługa klienta, rejestr zamówień, obsługa sprzedaży itp. Wykorzystywany w systemach Enterprise Resource Planning (ERP) Systems, Customer Relationship Management (CRM) software, and web-based applications.



Model ten dostarcza efektywnych rozwiązań do:

- efektywne i bezpieczne przechowywanie danych,
- transakcyjne odtwarzanie danych po awarii,
- optymalizacja dostępu do danych,
- zarządzanie wspólnienością,
- przetwarzanie zdarzeń -> odczyt -> zapis

Aktualnie wiele działających aplikacji (nawet w jednym obszarze) realizuje się jako mikroservisy, czyli małe i niezależne aplikacje (filozofia programowania LINUX - rób mało ale dobrze).

A co w przypadku gdy mamy do czynienia z:

- agregacjami danych z wielu systemów (np. dla wielu sklepów),
- wspomaganie analizy danych,
- raportowanie i podsumowanie danych,

- optymalizacja złożonych zapytań,
- wspomaganie decyzji biznesowych.

Badania nad tego typu zagadnieniami doprowadziły do sformułowania nowego modelu przetwarzania danych oraz nowego typu baz danych - Hurtownie Danych (Data warehouse).

**Przetwarzanie analityczne on-line OLAP (on-line analytic processing).**

Wspieranie procesów analizy i dostarczanie narzędzi umożliwiających analizę wielowymiarową (czas, miejsce, produkt).

Proces rzucający danych z różnych systemów do jednej bazy nazywamy Extract-Transform-Load (ETL) (normalizacja i encoding and schema transaction).

Analiza danych z hurtowni to przede wszystkim obliczanie **agregatów** (podsumowań) dotyczących wymiarów hurtowni. Proces ten jest całkowicie sterowany przez użytkownika.

#### Przykład

Załóżmy, że mamy dostęp do hurtowni danych gdzie przechowywane są informacje dotyczące sprzedaży produktów w supermarketie. Jak przeanalizować zapytania:

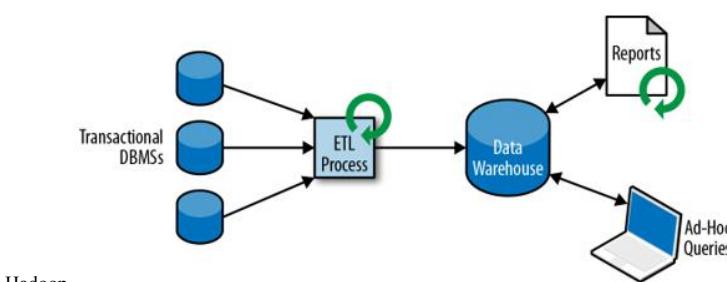
1. Jaka jest łączna sprzedaż produktów w kolejnych kwartałach, miesiącach, tygodniach ?
2. Jaka jest sprzedaż produktów z podziałem na rodzaje produktów ?
3. Jaka jest sprzedaż produktów z podziałem na oddziały supermarketu ?

Odpowiedzi na te pytania pozwalają określić wąskie gardła sprzedaży produktów przynoszących deficyty, zaplanować zapasy w magazynach czy porównać sprzedaż różnych grup w różnych oddziałach supermarketu.

W ramach Hurtowni Danych najczęściej wykonuje się dwa rodzaje zapytań:

1. Wykonywane okresowo w czasie zapytania raportowe obliczające biznesowe statystyki
2. Wykonywane ad-hoc zapytania wspomagające krytyczne decyzje biznesowe.

Oba wykonywane w trybie batchowym. Dziś ściśle wykonywane z użyciem technologii



Hadoop.

**61.** Wyjaśnij pojęcie i zastosowania biznesowe hurtowni danych.

Systemy Big data mogą być częścią (źródłem) dla **Hurtowni danych** (np. Data Lake, Enterprise Data Hub)

Ale **Hurtownie danych** nie są systemami Big Data!

1. **Hurtownie danych**

- przetrzymywanie danych wysoko strukturyzowanych
- skupione na analizach i procesie raportowania
- 100% accuracy

• **Hurtownie danych** - przechowuje historię zdarzeń w celu późniejszej analizy (Tabela Faktów).

• „**Hurtownia danych jest zbiorem danych**

- zorientowanych tematycznie,
- zintegrowanych,
- przeznaczonych tylko do odczytu,
- wersjonowanych czasem,
- zorganizowanych dla wspierania celów zarządczych.”

- William H. Inmon

- Business Intelligence czerpie wiedzę z systemów funkcjonujących w przedsiębiorstwie, następnie wykorzystuje tą wiedzę do wspomagania decyzji
- Hurtownia danych jest miejscem przechowującym dane z systemów funkcjonujących w przedsiębiorstwie
- Business Intelligence nie musi korzystać z Hurtowni Danych
- Hurtownia Danych jest zawsze elementem Business Intelligence

- **Dane muszą być zorientowane tematycznie**

- dane są zorganizowane według tematów mających kluczowe znaczenie dla organizacji, a nie według funkcjonalności, czy też podziału organizacyjnego
- wynika to z faktu, że analiza funkcjonowania organizacji i jej otoczenia wymaga globalnego spojrzenia na dane

- **Dane muszą być zintegrowane**

- dane gromadzone w hurtowni danych integrują informacje o poszczególnych tematach pochodzące z wielu źródeł, tak aby dawać w miarę pełny opis sytuacji

- **Dane muszą być nieulotne**

- dane przechowywane w hurtowni są przeznaczone tylko do odczytu
- nie usuwamy danych historycznych

- **Dane muszą być wersjonowane wg czasu powstania**

- zawartość hurtowni danych stanowią nie tylko dane obrazujące obecną sytuację, ale również - lub przede wszystkim - dane historyczne
- gromadzenie danych historycznych umożliwia badanie nie tylko sytuacji obecnej, ale również trendów, zmian otoczenia, pozwala odpowiedzieć na pytanie „dlaczego?” oraz dokonywać prognoz

- **Dane muszą być zorganizowane pod kątem działalności zarządczej, analitycznej**

- działalność zarządcza i analityczna polega między innymi na analizowaniu sytuacji, trendów, zmian, wzorców oraz prognozowaniu

Architektura powstała jako odpowiedź na wady „klasycznych” hurtowni danych:

- HD odpowiadają tylko na pytania, które były znane wcześniej
- Hurtownie danych i data marty posiadają dane o określonej szczegółowości. Nie można jej zwiększyć
- HD opierają się na zdefiniowanych źródłach danych

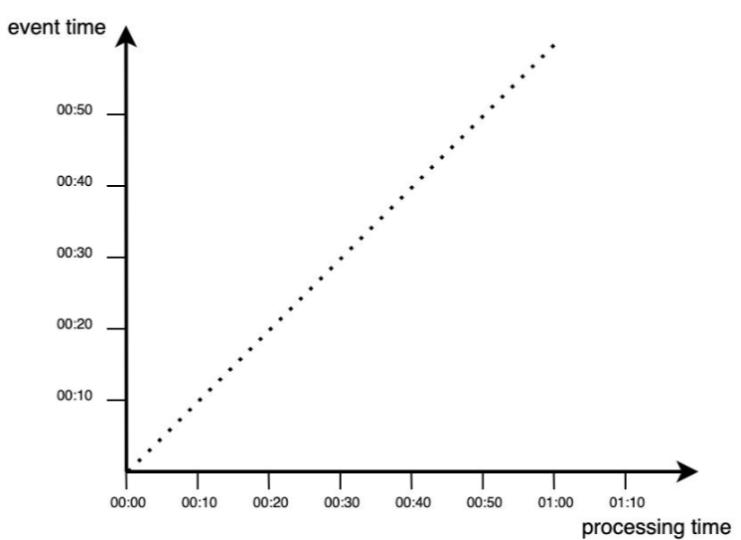
## **62. Omów problem czasu w strumieniowym przetwarzaniu danych, czym jest Watermark.**

W przypadku przetwarzania wsadowego przetwarzamy dane historyczne i czas uruchomienia procesu przetwarzania nie ma nic wspólnego z czasem występowania analizowanych zdarzeń.

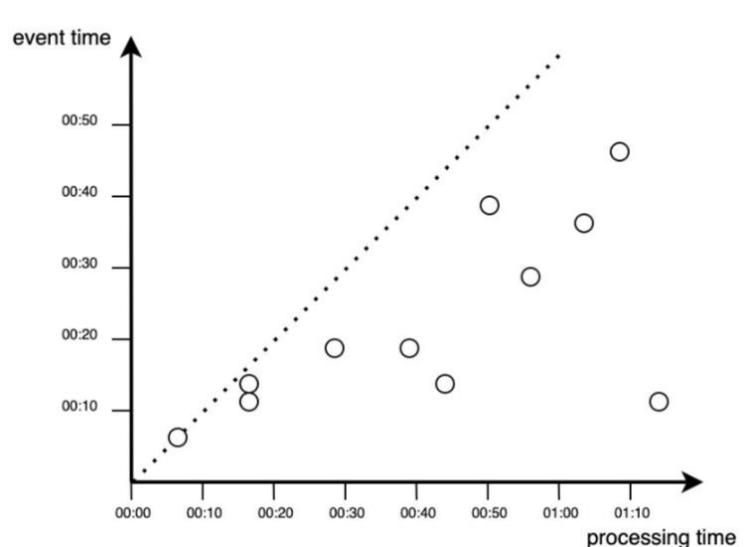
Dla danych strumieniowych mamy dwie koncepcje czasu:

1. czas zdarzenie (event time) - czas w którym zdarzenie się wydarzyło.
2. czas przetwarzania (processing time) - czas w którym system przetwarza zdarzenie.

W przypadku idealnej sytuacji:



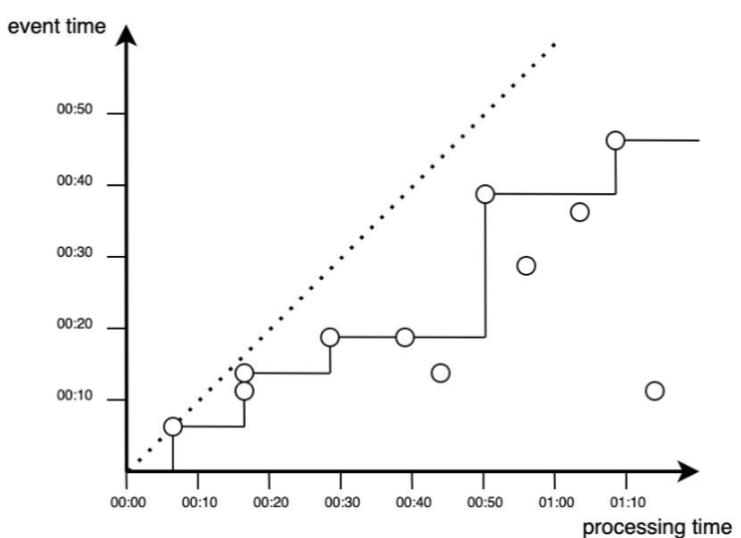
W rzeczywistości przetwarzanie danych zawsze odbywa się z pewnym opóźnieniem, co reprezentowane jest przez punkty pojawiające się poniżej funkcji dla sytuacji idealnej (poniżej diagonalnej).



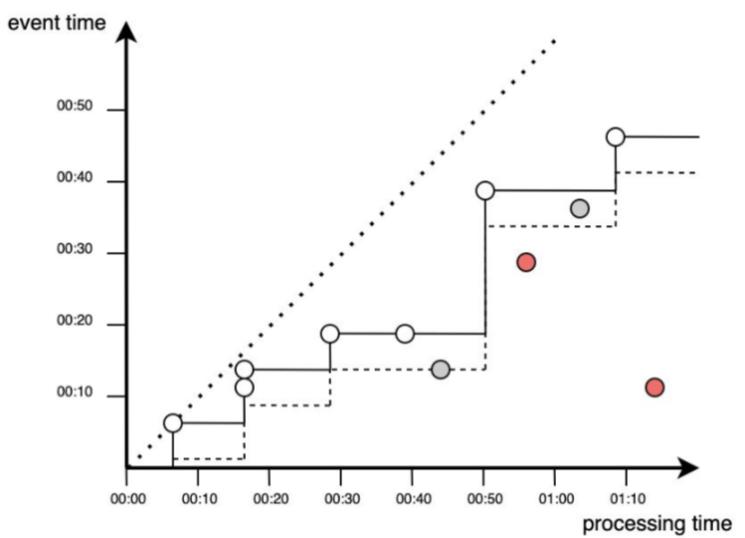
W aplikacjach przetwarzania strumieniowego istotne okazują się różnice między czasem powstania zdarzenia i jego procesowania. Do najczęstszych przyczyn opóźnienia wyszczególnia się przesyłanie danych przez sieć czy brak komunikacji między urządzeniem a siecią. Prostym przykładem jest tu przejazd samochodem przez tunel i śledzenie położenia przez aplikację GPS.

Możesz oczywiście zliczać ilość takich pominiętych zdarzeń i uruchomić alarm w sytuacji gdy takich odrzutów będzie za dużo. Drugim (chyba częściej) wykorzystywanym sposobem jest zastosowanie korekty z wykorzystaniem tzw. watermarkingu.

Proces przetwarzania zdarzeń w czasie rzeczywistym można przedstawić w postaci funkcji schodkowej, reprezentowanej na rysunku:



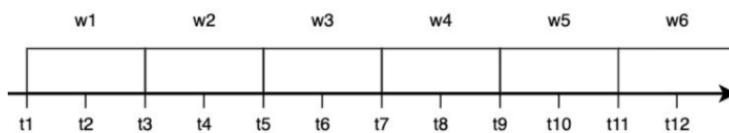
Jak można zauważyć nie wszystkie zdarzenia wnoszą wkład do analizy i przetwarzania. Realizację procesu przetwarzania wraz z uwzględnieniem dodatkowego czasu na pojawienie się zdarzeń (watermark) można przedstawić jako proces obejmujący wszystkie zdarzenia powyżej przerywanej linii. Dodatkowy czas pozwolił na przetworzenie dodatkowych zdarzeń, natomiast nadal mogą zdarzyć się punkty, które nie będą brane pod uwagę.



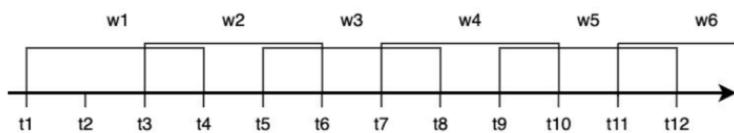
Przedstawione na wykresach sytuacje jawnie wskazują dlaczego pojęcie czasu jest istotnym czynnikiem i wymaga ścisłego określenia już na poziomie definiowania potrzeb biznesowych. Przypisywanie znaczników czasu do danych (zdarzeń) to trudne zadanie.  
*Streaming Watermark of a stateful streaming query is how long to wait for late and possibly out-of-order events until a streaming state can be considered final and not to change.*  
*Streaming watermark is used to mark events (modeled as a row in the streaming Dataset) that are older than the threshold as "too late", and not "interesting" to update partial non-final state.*

## okna czasowe

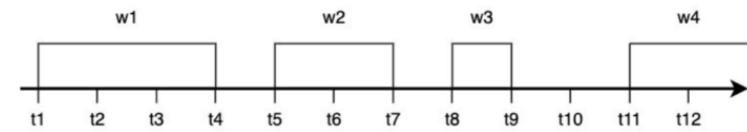
**Okno rozłączne** (ang. *tumbling window*) czyli okno o stałej długości. Jego cechą charakterystyczną jest to, iż każde zdarzenie należy tylko do jednego okna.



**Okno przesuwne** (ang. *sliding window*) obejmuje wszystkie zdarzenia następujące w określonej długości między sobą.



**Okno skokowe** (ang. *hopping window*) tak jak okno rozłączne ma stałą długość, ale pozwala się w nim na zachodzenie jednych okien na inne. Stosowane zazwyczaj do wygładzania danych.



### 63. Przedstaw różnicę pomiędzy wsadowym i strumieniowym sposobem przetwarzania danych.

Analityka strumieniowa pozwala na szybkie wyłapywanie trendów i wyprzedzenie konkurencji używającej tylko przetwarzania wsadowego. Reakcja na nagle zdarzenia na giełdzie czy w sieciach społecznościowych w celu zapobiegania bądź minimalizowania strat. Rodzaj danych

Batch = Duże, historyczne zbiory

Stream = Strumień danych, on line, przesyłane w trybie ciągłym

Czas uruchomienia przetwarzania

Batch = minuty, godziny, dni (patrz Hurtownie danych)

Stream = Real-time/near-real-time

Ponowne przetwarzanie

Batch = możliwe i stosowane bardzo często

Stream = „niemożliwe”

#### 5. Przetwarzanie danych:

- a) Przetwarzanie w czasie rzeczywistym – każde zdarzenie jest procesowane indywidualnie, nie jesteśmy świadomi zdarzeń i danych historycznych. Informacja zwrotna jest przekazywana natychmiast. Przykładowe narzędzia do wykorzystania w tym miejscu: Hitachi Operational Intelligence (HOI), Spark, Flink.
- b) Przetwarzanie **wsadowe** – procesy są grupowane i przetwarzane jednocześnie. Wykorzystywane do analistyki danych historycznych. Przykładowe narzędzia do wykorzystania w tym miejscu: Spark, MR (MapReduce).

Analityka strumieniowa pozwala na szybkie wyłapywanie trendów i wyprzedzenie konkurencji używającej tylko przetwarzania **wsadowego**. Reakcja na nagle zdarzenia na giełdzie czy w sieciach społecznościowych w celu zapobiegania bądź minimalizowania strat.

- który pojazd firmowej floty ma prawie pusty bak i gdzie wysłać prowadzącego pojazd do tankowania.
- Który pojazd floty zużywa najwięcej paliwa i dlaczego?
- Które urządzenia w zakładzie czy fabryce mogą ulec awarii w ciągu najbliższych dni?
- Jakie części zamienne trzeba będzie wymienić i w których maszynach w najbliższym czasie?
- Ile klientów aktualnie robi zakupy w sklepie i czy można im coś zaproponować?
- Czy klient dzwoni w celu zerwania umowy?
- i wiele wiele innych.

**Przedsiębiorstwo to organizacja, która generuje i odpowiada na ciągły strumień zdarzeń.**

Metody przetwarzania danych obejmują między innymi:

- **Przetwarzanie wsadowe:** przetwarzanie wsadowe zakłada podział danych na grupy lub partie, które można przetwarzać wraz z pojawiającą się dostępnością zasobów. Podczas przetwarzania wsadowego partie danych są przetwarzane kolejno, jedna po drugiej. Choć przetwarzanie wsadowe jest skuteczne w procesie przetwarzania dużych ilości danych, zazwyczaj najlepiej się sprawdza w przypadku danych, które nie wymagają natychmiastowego użycia.
- **Przetwarzanie strumieniowe:** przetwarzanie strumieniowe występuje w przypadku danych przetwarzanych ciągle, od momentu ich pojawienia się w potoku danych. Ten rodzaj przetwarzania zapewnia szybszą analizę mniejszych ilości danych w porównaniu z przetwarzaniem wsadowym.

Zazwyczaj jest wykorzystywane do przetwarzania danych, które wymagają szybkiego podjęcia działania.

## 64. Opisz dwa biznesowe zastosowania analizy danych w czasie rzeczywistym.

Detekcja anomalii

**Wartość odstająca** (ang. outlier) to obserwacja (wiersz w tabeli danych), która jest względnie odległa od pozostałych elementów próby. Wyraża ona przekonanie, iż związek między zmiennymi niezależnymi i zależnymi dla danej obserwacji może być inny niż dla pozostałych obserwacji. Dla pojedynczych zmiennych wartości odstające można określić wykorzystując wykres pudełkowy. Realizuje on wartości kwartyli, gdzie pierwszy i trzeci wyznaczają boki, natomiast wewnętrz umieszczana jest linia podziału realizująca drugi kwartyl (czyli medianę). Jego generowanie pozwala nam opisać rozkład oraz zaznaczyć wartości nietypowe jako spełniające  $x_{out} < Q1/4 - 3Q$  lub  $x_{out} > Q3/4 + 3Q$  gdzie  $Q = \text{frack}Q_3/4 - Q_1/4$ ,  $\text{frack}Q = \text{frack}Q_3/4 - Q_1/4$ .

Pojęcie wartości odstającej jest dość intuicyjne. Bolid formuły 1, jeśli chodzi o prędkość, to samochodowy outlier pośród samochodów używanych na co dzień.

W przypadku zagadnień z uczeniem nadzorowanym usuwanie anomalnych danych często poprawia jakość otrzymanego modelu. Poszukiwanie wartości odstających wykorzystywane jest również w procesie **detekcji anomalii**.

Polega on na wyszukiwaniu „dziwnych” przypadków w zbiorze danych.

Przetwarzając zbiór danych transakcji kredytowych technika ta może być wykorzystana do określenia transakcji fraudowych. Ma ona również zastosowanie w:

- wyszukiwaniu intruzów w sieci internetowej poprzez analizę zachowań jej użytkowników,
- monitorowaniu danych medycznych
- wyszukiwaniu wadliwych elementów poprzez analizę obrazu.

Metody wyszukiwania anomalii możemy podzielić ze względu na wykorzystywane modele uczenia maszynowego, na nadzorowane i nienadzorowane.

1. Do nadzorowanych metod można zaliczyć:

- sieci neuronowe,
- algorytm K-najbliższych sąsiadów
- sieci Bayesowskie.

2. W przypadku metod nienadzorowanych najczęściej bazuje się na założeniu, że większość napływających danych jest prawidłowa i tylko bardzo niewielki procent to dane odstające. Wykorzystuje się tutaj takie metody jak

- klasteryzacja metodą K–średnich,
- autoenkodery
- metody wykorzystujące testowanie hipotez.

Oczywiście metod wyszukiwania anomalii można znaleźć dużo więcej niż tylko wymienione wyżej.

Metoda klasyczna

Jej zasadniczym elementem pozwalającym stwierdzić czy dany przypadek jest anomalią to prawdopodobieństwo pojawienia się danego wiersza danych  $p(x)p(x)$ . Jeśli jest ono małe  $p(x) < \epsilon$  to można taką wartość traktować jako anomalię.

Jak policzyć prawdopodobieństwo otrzymania danego punktu? - określ rozkład.

Postać rozkładu musimy założyć.

Najprościej jest przyjąć, iż rozkład ten jest rozkładem normalnym  $N(\mu, \sigma)N(\mu, \sigma)$ . Jeśli nasze dane zawierają więcej niż jedną cechę możemy zastosować natwne założenie, iż zmienne te są od siebie niezależne. Wartości parametrów rozkładu wyestymować możemy z próby zgodnie z wyborem odpowiednich estymatorów.

Wartość oczekiwana estymowana jest przez średnią z próby, natomiast wariancję możemy wyestymować jako wariancję z próby.

Po otrzymaniu tych wartości dla każdej zmiennej odczytujemy, np. z tablic rozkładu normalnego, prawdopodobieństwo pojawienia się konkretnej wartości x<sub>i</sub>.

Ewaluacje wybranej metodologii można przeprowadzić poprzez wygenerowanie modelu klasyfikacji binarnej dla naszych danych.

Wskazany przykład można uznać za realizację modelowania anomalii z wykorzystaniem uczenia nadzorowanego.

#### Isolation Forest

Isolation Forest to algorytm oparty na algorytmie drzewa decyzyjnego, który identyfikuje wartości odstające (anomalie) poprzez izolację.

Zaproponowany został przez Fei Tony Liu, Kai Ming Ting oraz Zhi-Hua Zhou w 2008 roku. Algorytm ten ma liniową złożoność obliczeniową i pozwala na szybką detekcję anomalii z wykorzystaniem mniejszej ilości pamięci oraz działa dobrze w przypadku wielowymiarowych danych.

Izolacja wartości odstających następuje poprzez losowy wybór zmiennej z całego zbioru zmiennych i losowego wyboru punktu podziału (pomiędzy wartością minimalną i maksymalną). Losowy podział generuje mniejszą drogę w drzewie w przypadku anomalnych wartości (znajdują się one bliżej korzenia drzewa) niż dla wartości typowych.

Odległość tą można wyznaczyć generując dużą liczbę drzew, a następnie obliczając średnią odległość od korzenia, gdyż każdy punkt przechodzący przez drzewo generuje wynik na jakiejś głębokości drzewa.

Do tego może coś o giełdzie? np.

Obliczenia wykonywane na podstawie danych giełdowych pozwalają na dogłębną analizę zachowania rynku, w różnym stopniu szczegółowości oraz w różnych ramach czasowych.

Pozwala to na łatwiejsze podejmowanie decyzji w ramach tego rynku, zarówno dla inwestorów jak i dla algorytmów zajmujących się takimi zadaniami w sposób automatyczny.

- Brick-and-mortar businesses can track pieces of data from customers' mobile devices—for example, their locations—and send targeted incentives when would-be shoppers are nearby.
- Financial institutions can see stock market fluctuations in real-time and rebalance portfolios based on accurately computed, up-to-the-minute risk assessments. Or, they could offer this same capability (at a fee, of course) for clients who want more control over their investments.
- Web companies can examine clickstream records across multiple websites, combine that with at-rest data like demographic information, and automatically determine what subsets of viewers best relate to particular campaigns or even content placement.
- Ecommerce companies or other financial companies can watch machine-driven algorithms and find patterns that might be suspicious, helping to detect fraud the moment it happens.

## 65. Wymień i omów metodyki procesu eksploracji danych.

UWAGA: TU MOŻE CHODZIĆ TAKŻE O CRISP-DM I SEMMA.

### **Klasyfikacja**

Metoda ta polega na tworzeniu modelu, który używany jest do klasyfikowania nowych obiektów bazy. Występuje tu tak zwany zbiór danych treningowych a odkryte modele klasyfikacji są później używane do klasyfikacji nowych obiektów o nieznanej klasyfikacji.

Przykład zastosowania metody klasyfikacji:

- Wykrywanie nadużyć i oszustw finansowych korzystając ze zbioru danych treningowych zawierającego przykłady nadużyć i przykłady operacji uczciwych.
- Diagnostyka chorób na podstawie wcześniejszej klasyfikacji schorzeń.
- Obejmuje wiele technik: m.in.: statystyka, drzewa decyzyjne, sieci neuronowe

### **Grupowanie**

Polega na tworzeniu skończonych podzbiorów (klas, grup) obiektów posiadających podobne cechy. Metody te grupują obiekty w klasę w taki sposób, aby maksymalizować podobieństwo wewnętrzklasowe obiektów i minimalizować podobieństwo pomiędzy klasami obiektów.

Przykłady zastosowania metody grupowania :

- grupowanie dokumentów
- grupowanie klientów
- segmentacja rynku

### **Odkrywanie sekwencji**

Jest to odkrywanie wzorców zachowań na podstawie analizy danych zawierających informacje o zdarzeniach, które wystąpiły w określonym przedziale czasu, w celu znalezienia zależności pomiędzy występowaniem określonych zdarzeń w czasie.

Przykłady zastosowania metody odkrywania wzorców sekwencji:

- Odkrywanie wzorców zachowań użytkowników korzystających z Internetu.
- Badanie notowań akcji i odkrywanie wzorców w celu ustalenia modelu decyzyjnego dla strategii inwestycyjnych.

### **Odkrywanie asocjacji**

Jest to najszerza klasa metod eksploracji danych. Umożliwia znajdowanie nieznanych zależności i/lub reguł (asocjacji) pomiędzy występującymi elementami w zbiorach danych.

Przykłady zastosowania metody wyszukiwania asocjacji:

- Analiza koszyka kupionych produktów przez klienta w celu planowania

rozmieszczenia produktów w supermarketach. Odkrywane reguły mogą przykładowo wyglądać następująco:

„Jeżeli klienci kupują chleb i mleko, to kupują również masło”.

### **Wykrywanie zmian i odchyлеń**

Analiza danych zmieniających się w przedziale czasu i znajdowanie różnic pomiędzy aktualnymi a oczekiwanymi wartościami danych.

- Sygnalizowanie awarii lub włamania do systemów sieciowych.
- Wykrywanie oszustw podatkowych lub wyludzeń ubezpieczeniowych.

Wstępna eksploracja:

- wyliczanie podstawowych statystyk:min,max,avg,std, miary pozycyjne etc.
- poznawanie rozkładów zmiennych
- wizualizacja
- Analiza jakości danych
- Miary korelacji

Tutaj ogólnie należy opisać techniki uczenia bez nadzoru. Służy to eksploracji danych w celu uzyskania interesujących i istotnych informacji z danych, a nie szuka odpowiedzi na konkretne pytania, co jest przeciwnieństwem uczenia nadzorowanego. Należy tu wyróżnić:

- Statystykę opisową
- Analizę skupień - np. segmentacja klientów przy użyciu k-means.
- Sieci samoorganizujące się (SOM Kohonen)
- Analiza asocjacji i sekwenacji.

## **66. Omów dwie główne grupy metod eksploracji danych.**

Autor artykułu wymienił cztery grupy metod:

- techniki predykcyjne;
- techniki deskrypcyjne;
- techniki uczenia nadzorowanego;
- techniki uczenia bez nadzoru.

Dodatkowo, autor podkreśla, że kategorie nie są ścisłe tj. technika predykcyjna może posługiwać się technikami z zakresu uczenia nadzorowanego i na odwrót.

Strzelam, że najprawdopodobniej poprawną odpowiedzią na to pytanie jest zarówno podział na metody predykcyjne i deskrypcyjne jak i na techniki uczenia nadzorowanego i techniki uczenia bez nadzoru.

#### **Techniki predykcyjne (ang. predictive techniques)**

inaczej nazywane technikami lub modelami przewidywania, starają się na podstawie odkrytych wzorców dokonać uogólnienia i przewidywania wartości danej zmiennej. Pozwalają na przewidywanie wartości zmiennej wynikowej na podstawie wartości pozostałych zmiennych (badawczych lub przewidujących). Techniki te w SWD wykorzystywane są do przewidywania i szacowania np. zasobów (sprzętu/ludzi) do rozwiązywania postawionego problemu.

#### **Techniki deskrypcyjne (ang. description techniques)**

nazywane także technikami bądź modelami opisowymi, służą do formułowania uogólnień na temat badanych danych w celu uchwycenia ogólnych cech opisywanych obiektów oraz ich najważniejszych aspektów. Techniki te w SWD stosuje się do odkrywania grup i podgrup podobnych zdarzeń lub identyfikacji zdarzeń.

LUB

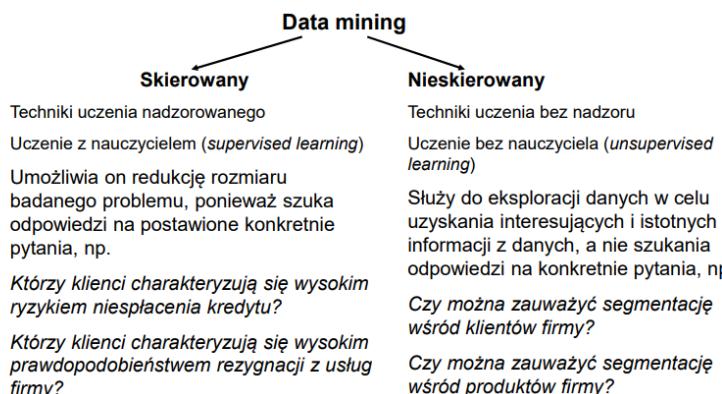
#### **Techniki uczenia nadzorowanego (ang. supervised learning)**

wykorzystują zbiory danych w których każdy obiekt posiada etykietę przypisującą go do jednej z przeddefiniowanych klas. Na podstawie zbioru uczącego budowany jest model, za pomocą którego można odróżnić obiekty należące do różnych klas. Technikami z zakresu uczenia nadzorowanego są techniki klasyfikacji do których należą drzewa decyzyjne, algorytmy najbliższych sąsiadów, sieci neuronowe, statystyka baysejowska, algorytmy maszyny wektorów wspierających SVM (ang. support vector machine) oraz techniki regresji.

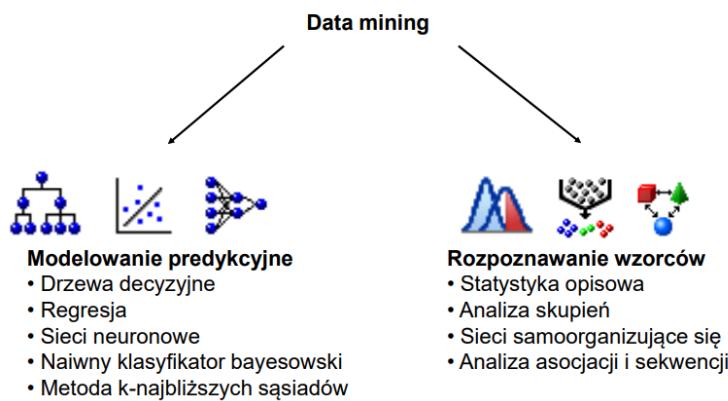
#### **Techniki uczenia bez nadzoru (ang. unsupervised learning)**

W przypadku tych technik brak jest etykiet obiektów, nie ma także zbioru uczącego. Techniki te starają się sformułować model (modele) wiedzy najlepiej pasujące do obserwowanych danych. Technikami z zakresu uczenia bez nadzoru są: techniki analizy skupień, klastrowania (ang. clustering), samoorganizujące się mapy (ang. self-organization map), algorytmy aproksymacji wartości oczekiwanej (ang. expectation-maximization) czy też zbiory przybliżone.

## Dwa nurty data mining



## Dwa nurty data mining



**67.** Omów metody selekcji zmiennych i obserwacji do modelowania data mining.

Ogólnie (metody selekcji zmiennych):

- Zmienne o wariancji bliskiej zeru – usuwamy takie zmienne
- Zmienne o bardzo dużej liczbie braków danych – usuwamy
- Zmienne które są istotne statystycznie ( $p - values$ ) – przy założeniu  $p - values$  na poziomie 0.05, wartości poniżej tego założenia wchodzą do modelu

Dla przykładu regresji (metody selekcji):

- Selekcja do przodu (ang. forward) – do modelu są wprowadzane kolejne zmienne istotne i wprowadzane są do momentu, aż model będzie wystarczająco satysfakcyjny. Nie można cofnąć wprowadzenia zmiennej do modelu.
- Selekcja do tyłu (ang. backward) – model zawiera wszystkie zmienne, a następnie usuwa się kolejno te, które są nicistotne.
- Selekcja krokowa (ang. stepwise) – kombinacja forward i backward selection. Podobna metoda jak forward, ale można usunąć zmienną z modelu jak nie pasuje.

Jak dzielimy obserwacje? Wykład u W. Grzendy (slajd).

Information value mierzy predykcyjną siłę zmiennej

Dzielimy zmienną na 10 binów i sprawdzamy % target w każdym binie i jeśli bardzo mocno się różnią, to może wskazywać, że zmienna ma dużą moc predykcyjną.

Jeśli wartość Information value jest:

- <0.02 --> zmienna bezużyteczna
- 0.02-0.1 --> słaba moc predykcyjna zmiennej
- 0.1- 0.3 --> umiarkowana
- 0.3- 0.5 --> duża siła predykcyjna zmienne
- >0.5 --> sprawdź zmienną, bo podejrzenie za dużo.

**Selekcja na podstawie Chi^2**

Selekcja na podstawie statystycznego testu niezależności Chi^2. Sprawdzamy niezależność zmiennej celu i badanej zmiennej.

- Proces data mining wymaga danych historycznych z odpowiednio długiego okresu czasu.
- Dostępna ogromna ilość danych – setki tysięcy wierszy (obserwacji) oraz setki, a nawet tysiące kolumn (zmiennych).

Zbyt wiele zmiennych	→ preselekcja zmiennych, agregacje, kombinacje zmiennych, transformacje liniowe i nieliniowe zbioru danych
Zbyt wiele obserwacji	→ próbkowanie (wybór losowy), poszukiwanie rekordów, które są szczególnie ważne z punktu widzenia, np. konstrukcji granic decyzyjnych przy klasyfikacji

Wybrane metody wstępnej selekcji zmiennych:

Metody statystyczne:

- Test niezależności chi-kwadrat (współczynnik zbieżności V-Cramera)
- Współczynniki korelacji (liniowej Pearsona, rangowej Spearmana/Kendalla)

Metody modelowania:

- Drzewa decyzyjne
- Regresja (dołączania, eliminacji, krokowa)

Inne metody:

- Analiza głównych składowych (PCA - *principal component analysis*)
- Grupowanie zmiennych

- W modelu powinny się znaleźć zmienne objaśniające silnie skorelowane ze zmienną celu oraz słabo skorelowane między sobą.

- Prostota modelu - zasada brzytwy Ockhama.

- Wybrany podzbiór danych powinien dać możliwość uzyskania takiego samego celu, jaki można by osiągnąć przy użyciu pełnego zbioru danych.
  - Idealnym wynikiem selekcji powinien być podzbiór, na którym jakość zbudowanego modelu będzie identyczna jak na całym zbiorze.
  - Idealny minimalny podzbiór powinien być niezależny od modeli na nim budowanych.
- 
- Selekcja rekordów ma na celu:
    - poprawę wydajności modelowania;
    - umożliwienie zastosowania danych do różnych metod analizy;
    - poprawę jakości danych (usunięcia rekordów nadmiarowych i obarczonych błędami).

Próbkowanie jest procesem wybierania podzbioru z danego zbioru.

Próbkowanie stosujemy, gdy:

- **nie możemy analizować wszystkich posiadanych danych;**
- **chcemy podzielić zbiór na kilka części.**

**Próba** – podzbiór elementów populacji generalnej podlegający badaniu.  
**Losowy dobór próby** musi spełniać następujące warunki:

1. każda jednostka populacji generalnej ma dodatnie znane prawdopodobieństwo znalezienia się w próbie;
2. istnieje możliwość ustalenia prawdopodobieństwa znalezienia się w próbie dla każdego zespołu elementów populacji.

W przypadku wyboru losowego można stosować następujące techniki losowania:

- **Losowanie niezależne (losowanie ze zwracaniem)** – po każdym losowaniu jednostka wraca do zbiorowości generalnej.
- **Losowanie zależne (losowanie bez zwracania)** – po każdym losowaniu element nie bierze już udziału w dalszym losowaniu.

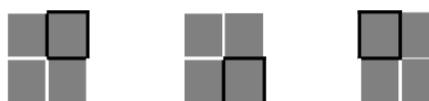
## **Losowanie warstwowe**

Przed przystąpieniem do losowania są tworzone warstwy:

- każdy element populacji jest zaliczony do jednej i tylko jednej warstwy;
- nie ma elementów pozostających poza warstwami;
- jednostki w danej warstwie muszą być jak najbardziej podobne;
- warstwy mają jak najbardziej różnić się między sobą.

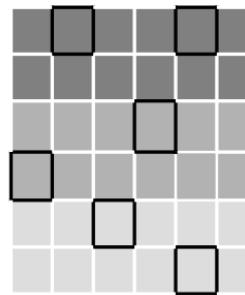
Losowania określonej liczby elementów dokonuje się z każdej warstwy w sposób niezależny.

Próbę stanowią elementy wylosowane ze wszystkich warstw.



## **Losowanie warstwowe proporcjonalne**

Dzielimy zbiór na warstwy i próbujemy z każdej warstwy według określonego schematu, aby próba była reprezentacyjna dla całego zbioru, rozmiar próbki w każdej warstwie powinien być proporcjonalny do jej liczności.



## **Przepróbkowanie - oversampling**

Losowanie warstwowe stosujemy, kiedy warstw jest niewiele, a istnieją istotne różnice w ich liczebnościach. Wówczas taki schemat losowania umożliwia otrzymanie próbki lepiej reprezentującej dany zbiór.

Jeśli warstwy interesujące nas pod kątem danego zjawiska są małe w porównaniu z pozostałymi stosuje się tzw. **przepróbkowanie (oversampling)** – losując tak samo dużą próbę z każdej warstwy, niezależnie od jej rozmiaru. Taki schemat losowania stosowany jest często przy analizach typu *credit scoring* lub przy badaniu zjawiska *churn*.

W SAS EM konstrując model na próbie powstałej w wyniku przepróbkowania, należy określić rzeczywiste odsetki wartości zmiennej objaśnianej. Te odsetki nazywane są prawdopodobieństwami *a priori*. Umożliwi to skorygowanie wartości ocen takich jak odsetek błędnych klasyfikacji, czy też różnych miar zysku.

## **Podział zbiorów w analizach data mining**

- **Zbiór uczący (treningowy)** – służy do „nauczenia” modelu, czyli znalezienia szukanej zależności panujące w zbiorze, w celu stworzenia modelu opisującego dane (około 50% całego zbioru).
- **Zbiór walidacyjny** – część zbioru służąca do porównania skuteczności otrzymanych modeli (około 25% całego zbioru).
- **Zbiór testowy** – część zbioru służąca do ostatecznej oceny skuteczności modelu, który najlepiej wypadł w części porównawczej (około 25% całego zbioru).

## **68. Metody klasyfikacji danych - przedstaw różnice i podobieństwa pomiędzy nimi.**

<b>modele</b>	<b>regresji</b>	<b>drzewa decyzyjnego</b>	<b>sieci neuronowej</b>
do czego służy?	interpretacja	prognoza	prognoza
liczba danych	mało	duże	duże
odporność na obserwacje odstające	nie	tak	tak (poradzą sobie, ale nie lubią) – stworzenie odpowiedniej liczby neuronów
moc obliczeniowa (złożoność metody)	szymbka metoda	szymbka metoda	duża moc obliczeniowa, dugo się liczy, wolna metoda
zalożenia	zakłada konkretną zależność (np. liniowa)	same znajdują zależność (metody elastyczne)	same znajdują zależność (metody elastyczne)
interpretacja wyników	prosta	trudna	trudna

Klasyfikacja danych – metoda eksploracji danych polegająca na znajdowaniu odwzorowania danych w zbiór predefiniowanych klas. Na podstawie zawartości bazy danych budowany jest model (np. Drzewo decyzyjne, które służy do klasyfikowania nowych obiektów w bazie danych)

Dane wejściowe - zbiór krotek/ Dane wyjściowe - model (klasyfikator).

Celem jest skojarzenie obiektu na podstawie jego cech z pewną kategorią

Kryteria porównawcze modeli klasyfikacji to:

- Dokładność predykci (zdolność do poprawnej predykci)
- Efektywność (koszt obliczeniowy)
- Odporność modelu (zdolność do predykci klas w przypadku braku lub niepełnych danych)
- Skalowalność (zdolność do konstrukcji klasyfikatora dla dowolnie dużych wolumenów)
- Interpretowalność (jak konstrukcja wpływa na zrozumienie danych)

#### Modele logitowe

- Podobna do regresji liniowej
- Prosty i często stosowany algorytm uczenia
- Łatwa do implementacji
- Szacuje związek między jedną zależną zmienną binarną i zmiennymi niezależnymi

#### Modele drzew decyzyjnych

- Popularna
- Algorytm uczenia nadzorowanego
- Intuicyjne zrozumiałe dla człowieka
- Łatwa skalowalność dla dużych zbiorów
- Dobra dokładność
- Stosunkowa prostota w tworzeniu
- Wadą niemożliwość łatwego wychwytycia korelacji między atrybutami
- Odporność na obserwacje odstające
- Mała odporność na szum danych
- Wadą np. Wymóg dużej próby uczącej

#### Modele sztucznych sieci neuronowych

- Duża odporność na niepełne lub błędne informacje
- Stosowana w przypadkach gdy nie jest znany sposób rozwiązania problemu

#### Maszyna wektorów wspierających (Support Vector Machines - SVM)

- Oferuje dużą dokładność
- Można wykorzystywać dodanych nieliniowych
- Może obsługiwać wiele zmiennych ciągłych i kategorycznych

#### Metoda Bayesa

- Naiwny, bo zakłada wzajemną niezależność zmiennych niezależnych
- Wymaga stosunkowo niewielkiej ilości danych
- Może wymagać nirealistycznych założeń (np. Wyniki muszą się wzajemnie wykluczać)

- Model w którym wiedza jest przedstawiona a priori

**Klasyfikatory najbliższych sąsiadów**

- Wyznaczenie k sąsiadów w n-wymiarowej przestrzeni do której badany punkt ma najbliższe

**Metoda K-średnich**

- Popularna
- Metoda analizy skupień/klastrowania
- Polega na dzieleniu populacji na klasy z jak najmniejszą wariancją w klasie

**69.** Przedstaw model drzewa decyzyjnego.

Ad 70

**70.** Omów modele lasów losowych.

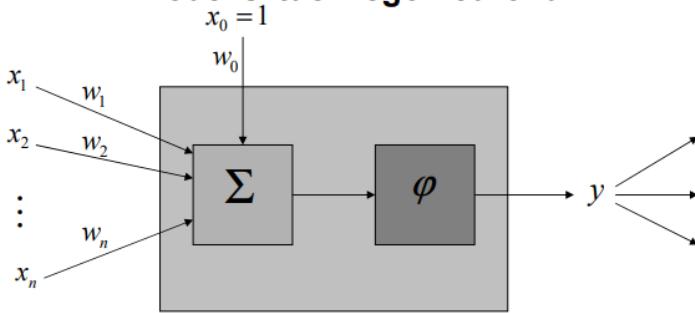


Zajecia\_5 (3).pdf

**71.** Przedstaw modele sztucznych sieci neuronowych.

Sieci neuronowe zostały zainspirowane biologią, a dokładniej, stanowią próbę uczenia nieliniiowego występującego w biologicznych systemach nerwowych.

## Model sztucznego neuronu



W modelu sztucznego neuronu:

- Zbierane są sygnały wejściowe ( $x_i$ ) z poprzedzających go neuronów.
- Otrzymane sygnały są sumowane z odpowiednimi wagami ( $w_i$ ) i tworzona jest z nich jedna wartość (np. za pomocą funkcji  $\emptyset$ )
- Otrzymany wynik jest wejściem dla funkcji aktywacji ( $\emptyset$ )
- Z wykorzystaniem funkcji aktywacji otrzymywany jest sygnał wyjściowy (dane wyjściowe) ( $y$ ).
- Sygnał wyjściowy przekazywany jest do następnych neuronów.

Sztuczny neuron:

- stanowi jednostkę przetwarzającą  $n$  sygnałów wejściowych;
- sygnały przetwarzane są z wykorzystaniem wag  $w_i$ ,  $i=1,\dots,n$ ;
- Wyznaczana jest w nim wartość sygnału wyjściowego wg według wzoru:

$$u = w_0 + \sum_{i=1}^n w_i x_i \quad y = \varphi(u)$$

wzoru:

Jednym z pierwszych, którzy zaproponowali model sztucznego neuronu był McCulloch-Pittsa. W przypadku tego modelu funkcja  $\varphi$ , będąca funkcją aktywacji ma postać funkcji skokowej:

$$\varphi(u) = \begin{cases} 1 & \text{dla } u \geq u_0, \\ 0 & \text{dla } u \leq u_0. \end{cases}$$

.

Sztuczna sieć neuronowa - definicje:

- Sieć komórek zdolnych do zdobywania, przechowywania i wykorzystywania wiedzy wynikającej z dotychczasowego działania.
- Procesor o masowej równoległości, zbudowany z prostych jednostek posiadający zdolność składowania wiedzy pochodzącej z doświadczenia i wykorzystujący ją.
- Układ neuronów w których wyjścia każdego neuronu są połączone, poprzez wagę, z wejściami wszystkich neuronów, w tym także z jego własnym wejściem.

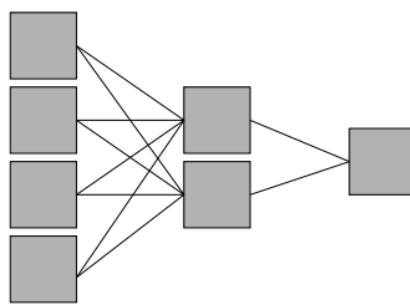
Sztuczna sieć neuronowa definiowana jest poprzez: zadanie modelu sztucznego neuronu, podanie topologii, zdefiniowanie reguł uczenia sieci.

Sieć neuronowa jest zbiorem prostych jednostek (neuronów) połączonych w określony sposób.

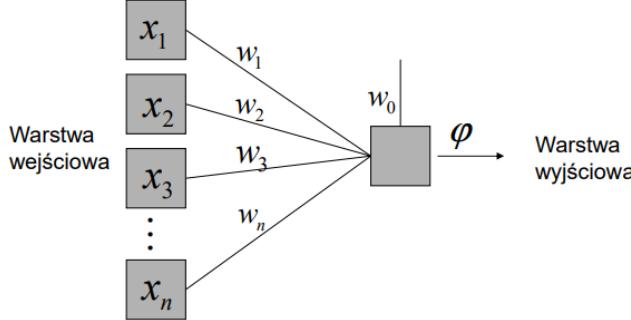
Sygnały (wartości) wejściowe sieci mają przypisane wagi, których wartości mogą podlegać zmianom w trakcie uczenia.

Sygnały wyjściowe sieci powstające jako odpowiedzi na sygnały wejściowe wyzaczają rozwiązań stawianego sieci zadania.

**Warstwa wejściowa    Warstwa ukryta    Warstwa wyjściowa**



### Model regresji liniowej – przykład prostej sieci neuronowej



$$E(y) = \varphi(w_0 + \sum_{i=1}^n w_i x_i) \quad \varphi^{-1}(E(y)) = w_0 + \sum_{i=1}^n w_i x_i$$

## Kodowanie sygnałów wejściowych oraz wyjściowych

Przed rozpoczęciem uczenia zaleca się kodowanie danych wejściowych tak, aby przyjmowały wartości z przedziału od 0 do 1.

- Zmienne ciągłe – normalizacja min-max

$$X^* = \frac{X - \min(X)}{\text{zakres}(X)} = \frac{X - \min(X)}{\max(X) - \min(X)}$$

- Zmienne jakościowe – znaczniki (flagi) informujące o wartości atrybutu.  
Zmienne jakościowe o  $k$  wartościach zamieniamy na  $k-1$  znaczników.

Neurony wyjściowe najczęściej dają na wyjściu wartości z przedziału od 0 do 1.

## Sieci neuronowe – klasyfikacja

- **Dwudzielny problem klasyfikacyjny:**

„Czy klient splaci kredyt?”

„Czy klient zrezygnuje z usług firmy?”

Przypadek pojedynczego neuronu wyjściowego z uprzednio ustawioną wartością progową oddzielającą dwie klasy.

- **Wielodzielny problem klasyfikacyjny:**

„Jakiej jakości jest produkt: zlej, średniej, dobrzej?”

Przypadek pojedynczego neuronu wyjściowego dla kilku kategorii jednoznacznie uporządkowanych.

- **Wielodzielny problem klasyfikacyjny:**

„Od jakiego producenta pochodzi produkt: XX, YY, ZZ”

Przypadek wielu neuronów wyjściowych – kodowanie 1 z  $n$  (każdej kategorii przypisywany jest dokładnie jeden neuron wyjściowy).

## Sieci neuronowe – szacowanie, przewidywanie

- Dane wejściowe → kodowanie np. normalizacja min-max
- Dane wyjściowe → denormalizacja

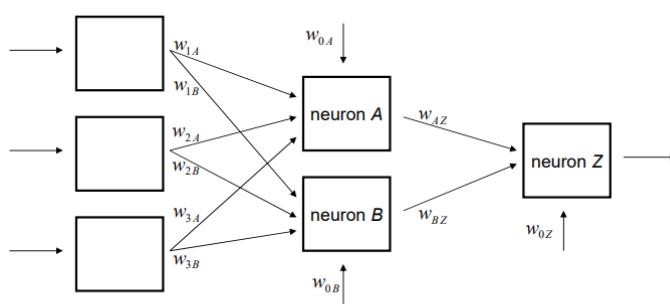
$$\text{wartość przewidywana} = \text{wynik} * \text{zakres} + \text{minimum}$$

wynik – wartość zwracana przez sieć

zakres – zakres wartości początkowych atrybutu przed skalowaniem  
minimum – minimalna wartość atrybutu przed skalowaniem

### Prosta sieć neuronowa – przykład

Warstwa wejściowa                    Warstwa ukryta                    Warstwa wyjściowa



Sieć warstwowa, jednokierunkowa i pełna.

Sieć pełna – każdy neuron jest połączony tylko z wszystkimi neuronami warstwy następnej i nie jest połączony z neuronami ze swojej warstwy.

## Prosta sieć neuronowa

Funkcja **łącząca (kombinacji)** służy do obliczania kombinacji liniowej sygnałów wejściowych z wykorzystaniem odpowiednich wag połączeń. Funkcja liniowa (**LINEAR**) dla węzła  $j$ :

$$net_j = \sum_i w_{ij} x_{ij} = w_{0j} x_{0j} + w_{1j} x_{1j} + \dots + w_{lj} x_{lj}$$

$x_{ij}$  – sygnał przekazywany z  $i$ -tego wejścia do  $j$ -tego neuronu

$w_{ij}$  – waga połączenia pomiędzy  $i$ -tym wejściem a  $j$ -tym neuronem  
( $j$ -ty neuron ma  $l+1$  wejść)

$x_i$  – sygnały wejściowe pochodzące z warstwy poprzedniej

$x_0$  – stałe w czasie wymuszenie zewnętrzne

### Przykład c.d.

Dla neuronu A w warstwie ukrytej:

$$net_A = \sum_i w_{iA} x_{iA} = w_{0A}(1) + w_{1A} x_{1A} + w_{2A} x_{2A} + w_{3A} x_{3A}$$

### Inne funkcje kombinacji

Niech  $x_1, x_2, \dots, x_p$  będą wartościami standaryzowanymi, a  $w_{ij}, b, b_j, a_j$  będą obliczane w sposób interaktywny, wówczas:

#### ADD

$$net_j = \sum_{i=1}^p x_i$$

#### EQSLOPES

$$net_j = w_{01j} + \sum_{i=1}^p w_{il} x_i$$

#### EQRADIAL

$$net_j = -b^2 \sum_{i=1}^p (w_{ij} - x_i)^2$$

#### EHRADIAL

$$net_j = -b_j^2 \sum_{i=1}^p (w_{ij} - x_i)^2$$

## Inne funkcje kombinacji

Niech  $f$  oznacza liczbę połączeń z neuronem, w *Enterprise Miner*  $f$  nazywane jest *fan-in*.

### EWRADIAL

$$net_j = f \log(\text{abs}(a_j)) - b^2 \sum_i^p (w_{ij} - x_i)^2$$

### EVRADIAL

$$net_j = f \log(\text{abs}(b_j)) - b_j^2 \sum_i^p (w_{ij} - x_i)^2$$

### XRADIAL

$$net_j = f \log(\text{abs}(a_j)) - b_j^2 \sum_i^p (w_{ij} - x_i)^2$$

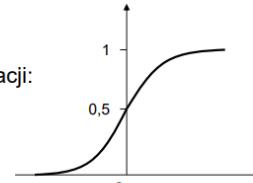
## Prosta sieć neuronowa – funkcja aktywacji

- Wyznaczona kombinacja liniowa  $net_j$  jest następnie wejściem dla funkcji aktywacji.

Funkcja **aktywacji** służy do przekształcania wyjścia z funkcji kombinacji i przesyłania tej wartości dalej.

Przykładowa sigmoidalna funkcja aktywacji:

$$\varphi(x) = \frac{1}{1 + e^{-x}}$$

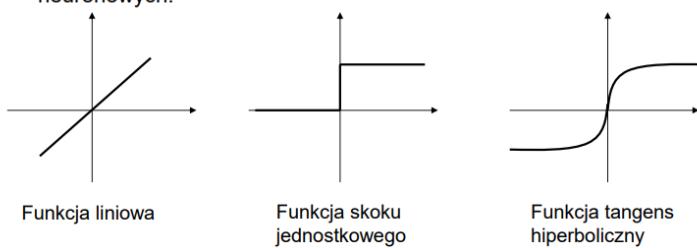


- Typowe algorytmy uczenia sieci neuronowych typu wielowarstwowy perceptron bazują na metodach gradientowych. W związku z tym metody te wymagają różniczkowalnych funkcji aktywacji. Szczególnie popularne jest wykorzystanie przedstawionej funkcji sigmoidalnej.

## Inne funkcje aktywacji

Funkcjami aktywacji mogą być dowolne funkcje ciągłe generujące w neuronie wyjściowym wartość z odpowiedniej skali.

Trzy inne podstawowe rodzaje funkcji aktywacji stosowane w sieciach neuronowych:



## Inne funkcje aktywacji

### **Arc Tan**

$$H_j = (2/\pi)\tan^{-1}(net_j) \quad j=1,2,3$$

### **Elliot**

$$H_j = \frac{net_j}{1+|net_j|}$$

### **Hyperbolic Tangent**

$$H_j = \tanh(net_j)$$

### **Logistic**

$$H_j = \frac{1}{1+\exp(-net_j)}$$

### **Gauss**

$$H_j = \exp(-0.5net_j^2)$$

### **Sine**

$$H_j = \sin(net_j)$$

### **Cosine**

$$H_j = \cos(net_j)$$

### **Exponential**

$$H_j = \exp(net_j)$$

### **Square**

$$H_j = net_j^2$$

### **Reciprocal**

$$H_j = \frac{1}{net_j}$$

### **Softmax**

$$H_j = \frac{\exp(net_j)}{\sum_{k=1}^3 \exp(net_k)} \quad j=1,2,3$$

## Prosta sieć neuronowa

### Przykład c.d.

Dla neuronu A funkcja aktywacji przyjmuje jako wejście  $net_A$  i wyznaczana jest wartość wyjściowa neuronu A:

$$y_A = \varphi(net_A) = \frac{1}{1 + \exp(-net_A)}$$

Analogicznie wyznaczana jest wartość wyjściowa dla neuronu B:

$$y_B = \varphi(net_B) = \frac{1}{1 + \exp(-net_B)}$$

W neuronie Z łączone są wartości wyjściowe pochodzące z neuronów A i B.

Wyznacza jest ważona suma sygnałów, oznaczając  $x_{BZ} = y_B - x_{AZ} = y_A$  mamy:

$$net_Z = \sum_i w_{iZ} x_{iZ} = w_{0Z}(1) + w_{AZ}x_{AZ} + w_{BZ}x_{BZ}$$

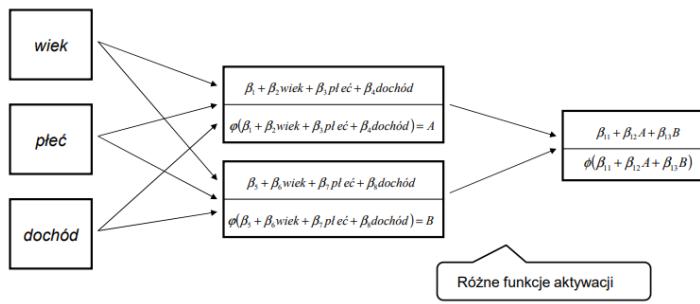
Wartość funkcji aktywacji neuronu Z wyznaczana jest w następujący sposób:

$$y_Z = \varphi(net_Z) = \frac{1}{1 + \exp(-net_Z)}$$

20

### Przepływ informacji w sieci neuronowej

Warstwa wejściowa      Warstwa ukryta      Warstwa wyjściowa



### **Zalety sieci neuronowych**

- Otrzymane wyniki są zmienną ciągłą
- Analiza wielu zmiennych jednocześnie
- Możliwości stworzenia modelu gdy rozwiązanie ma dużą złożoność
- Odporność na zaszumione dane

### **Wady sieci neuronowych**

- Trudności w ustaleniu parametrów architektury sieci
- Wpadanie w minima lokalne
- Potencjalnie długi czas uczenia sieci
- Problemy braków danych i obserwacji odstających
- Brak jasnej interpretacji

**72.** Omów metody grupowania danych.

<b>Grupowanie pod nadzorem</b>	<b>Grupowanie bez nadzoru</b>
Dana jest jednoznacznie określona zmienna celu	Nie istnieje jednoznacznie określona zmienna celu. Algorytm eksploracji danych poszukuje wzorców i struktur wśród wszystkich zmiennych
<ul style="list-style-type: none"> <li>• Sztuczne sieci nerонowe</li> </ul>	<ul style="list-style-type: none"> <li>• Drzewa decyzyjne</li> <li>• Regresja logistyczna</li> <li>• Wielowarstwowy perceptron MLP</li> <li>• Metoda <math>k</math>-najbliższych sąsiadów</li> <li>• SVM</li> </ul> <ul style="list-style-type: none"> <li>• Metoda <math>k</math>-śrenich</li> <li>• Metoda SOM – Kohonen</li> </ul>

## **Metody grupowania – inny podział**

### **1. Metody hierarchiczne:**

- Skupienia tworzą drzewa binarne i w ten sposób uzyskiwana jest hier tj. jedne skupienia są zawarte w drugich.
- Uwzględniając kryterium rozpoczęcia procesu grupowania wyróżniamy metody aglomeracyjne i metody podziałowe.
- Ze względu na sposób wyznaczania odległości między skupieniami najczęściej spotykane metody aglomeracyjne to: najbliższego sąsiada; najbliższego sąsiedztwa, mediany, średnia ciężkości, średniej odległości wewnętrz skupień, średniej odległości między skupieniami, minimalne wariancji Warda.

### **2. Metody optymalizacyjno-iteracyjne:**

- Wymagają wstępniego podziału zbioru obiektów na określona liczbę podzbiorów. Wybrany sposób podziału jest iteracyjnie modyfikowany. Np. metoda k-srednich.

### **3. Metody obszarowe:**

- Przestrzeń grupowania jest dzielona na rozłączne obszary a obiekty znajdujące się w otrzymanych obszarach tworzą grupy.

### **4. Inne metody**

**73.**



**Zajecia\_12 (2).pdf**

Omów metody analizy danych transakcyjnych.

Dane używane przez modele reguł asocjacyjnych mogą mieć format transakcyjny. Dane transakcyjne są zapisywane w postaci osobnego rekordu dla każdej transakcji lub pozycji. Jeśli klient dokonuje kilku zakupów, każdy będzie zapisany w osobnym folderze, wraz z powiązonymi elementami dowiązanymi na podstawie id. klienta.

Głównymi metodami analizy danych transakcyjnych jest analiza asocjacji. Metody te polegają na identyfikacji współzależności cech. Umożliwiają wykrycie logicznych reguł wiążących zmienne w zbiorze danych przez identyfikację pozycji, które występują razem.

Dzięki odkryciu współzależności między cechami można tworzyć reguły postaci: jeżeli cecha A towarzyszy określному zdarzeniu, to cecha B towarzyszy temu zdarzeniu z określonym prawdopodobieństwem. Reguły asocjacyjne mają postać: „Jeżeli A, to B”, czyli: „Jeżeli [poprzednik], to [następnik]”. Jeżeli transakcja pasuje do reguły, tzn. spełnione są warunki poprzednika i następnika, to mówimy, że reguła zawiera określona transakcję, albo że transakcja wspiera określoną regułę asocjacyjną. Regułę asocjacyjną jest implikacja danego zbioru na inny. Jeżeli transakcja pasuje do reguły, tzn. spełnione są warunki poprzednika i następnika, to mówimy, że reguła zawiera określoną transakcję, albo że transakcja wspiera określoną regułę asocjacyjną.

Wprowadza się wielkości pozwalające opisać właściwości reguł, tzw. charakterystyki reguł asocjacyjnych:

(2) ufność reguły (*confidence*):

$$\frac{n(A \cap B)}{n(A)} = P(A | B),$$

(1) wsparcie reguły (*support*):

$$\frac{n(A \cap B)}{N} = P(A \cap B)$$

(3) przyrost (*lift*):

$$\frac{\text{ufność}}{P(B)} = \frac{P(B | A)}{P(B)},$$

Tworzenie reguł asocjacyjnych odbywa się w dwóch etapach:

1. znajdujemy wszystkie zbiory częste
2. na podstawie częstych zdarzeń tworzymy reguły asocjacyjne, które spełniają warunek minimalnego wsparcia i poziomu ufności

## 74. Czy standardowy model regresji logistycznej należy do klasy uogólnionych modeli liniowych? Odpowiedź uzasadnij.

Na samym początku warto nadmienić czym właściwie jest uogólniona postać modelu liniowego.

W swojej najprostszej postaci model liniowy określa (liniowe) powiązanie pomiędzy zmienną zależną (lub odpowiednią) Y, a zbiorem predyktorów (zmiennych objaśniających) X, posiada on m.in. założenia o tym że:

- każda zmienna ma rozkład normalny
- każda zmienna posiada taką samą wariancję
- Występuje liniowa zależność między zmienną objaśnianą i objaśniającą.

Uogólniona postać modelu liniowego rozszerza ogólny model liniowy w taki sposób, że zmienna zależna jest liniowo powiązana z czynnikami i współzmiennymi za pośrednictwem określonej funkcji wiążącej.(Funkcja przybiera różną formę w zależności od rozkładu zmiennej y: identycznościowa, logarytmiczna, potęgowa, wiążąca Log-log, uogólniony Logit)

Model pozwala więc by zmienna zależna nie posiadała rozkładu normalnego. (Dzięki bardo ogólnej postaci wzoru modelu obejmując on wiele modeli statystycznych, takich jak takich jak regresja liniowa dla odpowiedzi o rozkładzie normalnym, modele logistyczne dla danych binarnych, modele logarytmiczno-liniowe dla danych o liczebności).

Uogólniony model liniowy definiowany jest więc poprzez rozkład zmiennej objaśnianej (z rodzin wykładniczych rozkładów) i funkcję łączącą , która opisuje związek wartości oczekiwanej zmiennej objaśnianej i kombinacji liniowej zmiennych objaśniających.

Dwumianowy model regresji logistycznej jest wykorzystywany do objaśniania zmiennej dychotomicznej(1|0) Y w zależności od poziomu egzogenicznych zmiennych Xn (jakościowych/ilościowych). Jest on szczególnym przypadkiem uogólnionego modelu liniowego, wykorzystującym wcześniej wspomnianą funkcję wiążącą Logit:

$$g(\mu) = \beta_0 + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_k X_k,$$

Gdzie beta0(wyraz wolny) i bety tworzą wektor oceny parametrów(współczynników regresji. Sama „g” jest funkcją wiążącą określającą związek średniej wartości zmiennej objaśnianej (μ) z liniową kombinacją predyktorów. Funkcja wiążąca w modelu logitowym ma postać logitu(Funkcja przekształcająca prawdopodobieństwo na logarytm szansy)

Dodatkowo w modelu regresji logistycznej zmienna objaśniająca przyjmuje wartości binarne. Naturalnym rozkładem prawdopodobieństwa do modelowania warunkowego rozkładu jest rozkład Bernoulliego z parametrem p (p jest prawdopodobieństwem tego, że y=1). Rozkład ten należy do rodzin rozkładów wykładniczych.

Powyższe dwa warunki potwierdzają przynależność regresji logistycznej do uogólnionych modeli liniowych.

Uogólnione modele liniowe to klasa modeli, w których zmienna objaśniana ma rozkład należący do rodzin wykładniczej. Rozkład dwumianowy i wielomianowy, opisujące odpowiednio rozkłady zmiennych binarnych oraz wielomianowych, należą do tej rodziny, więc można zastosować ten typ modeli w analizie takich zmiennych. W uogólnionych modelach liniowych zakłada się, że wartość oczekiwana jest różniczkowalną funkcją (g) kombinacji liniowej zmiennych objaśnianych (xB)  $g(u)=xB \rightarrow$  funkcja łącząca

Występuje to w modelu logistycznym

Model regresji **logistycznej** estymuje prawdopodobieństwo wystąpienia zdarzenia A.

$$P(A) = P(Y = 1) = \mu - wartość\ oczekiwana\ zmiennej\ Y = 1$$

$$\begin{aligned} P(Y = 1) &= \frac{1}{1 - e^{-z}} = \frac{1}{1 - e^{-(\alpha + \sum \beta_i X_i)}} \\ 1 - P(Y = 1) &= 1 - \frac{1}{1 - e^{-z}} \end{aligned}$$

W regresji logistycznej zamiast wprost modelować zmienną Y, to estymujemy iloraz szans.

$$OR = \frac{P(Y = 1)}{1 - P(Y = 1)}$$

$$OR = \frac{\text{prawdopodobieństwo wystąpienia zdarzenia } A}{\text{prawdopodobieństwo wystąpienia zdarzenia przeciwnego do zdarzenia } A}$$

$$\begin{aligned} OR &= \frac{P(Y = 1)}{1 - P(Y = 1)} = e^z = e^{\alpha + \sum \beta_i X_i} \quad | \log \\ \log(OR) &= \alpha + \sum \beta_i X_i = \alpha + \beta_1 X_1 + \beta_2 X_2 + \dots + \beta_n X_n - \text{funkcja logit} \end{aligned}$$

Ostatecznie logit z ilorazu szans modeluje funkcję liniową.

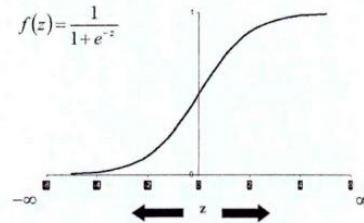
**75.** Przedstaw metody estymacji parametrów modelu regresji logistycznej.

$$\log(OR) = \alpha + \sum \beta_i X_i = \alpha + \beta_1 X_1 + \beta_2 X_2 + \cdots + \beta_n X_n - \text{funkcja logit}$$

Parametry w równaniu szacuje się metodą największej wiarygodności – polega ona na szukaniu takich nieznanych wartości parametrów, dla których funkcja przyjmuje wartości maksymalne.

#### 2.1. Funkcja logistyczna

$$f(z) = \frac{1}{1+e^{-z}}$$



Wartości maksymalne funkcji można wyznaczyć za pomocą pochodnych funkcji. Wylicza się pochodne funkcji dla każdego parametru  $\beta$  i przyrównuje się je do zera. Tworzy się układ k – równań. Aby rozwiązać taki układ k – równań należy wykorzystać metodę iteracyjną algorytmu Newtona – Raphsona. W SAS jest to procedura **logistic**.

Na wstępnie należy zaznaczyć, że w modelu regresji logistycznej estymujemy bezpośrednio prawdopodobieństwo

wystąpienia zdarzenia A:  $P(A) = P(Y = 1) = \mu$  gdzie  $\mu$  jest wartością oczekiwana zmiennej Y. Prawdopodobieństwo jest funkcją zależną od zmiennych

Zależność  $P(A)$  od zmiennych  $X_1 \dots X_k$  jest nieliniowa. Parametry równania logistycznego **szacuje się metodą największej wiarygodności(MNW)**. Jest to metoda iteracyjna.

Kierunek zmian  $P$  w zależności od zmiennej zależy od znaku współczynnika występującego przy tej zmiennej jeżeli  $Beta(i) > 0$  to wraz ze wzrostem  $X(i)$  wartość prawdopodobieństwa  $P$  wzrasta. Mówimy wtedy, że czynnik opisywany przez zmienną  $X(i)$  działa stymulująco zdarzenie A. W sytuacji odwrotnej powoduje spadek wartości  $P$ .

Metoda największej wiarygodności polega na szukaniu takich wartości nieznanego parametrów dla których funkcja przyjmuje wartość maksymalną. Bierze się to założenia, że w wyniku wylosowania próby powinno się zrealizować się zdarzenie o największym prawdopodobieństwie. Funkcja  $L$  osiąga maksimum w tych samych punktach co jej logarytm( $lnL$ ) w praktyce wyznacza się maksimum funkcji  $lnL$ .

(maksymalizujemy funkcję  $ln(L)$  a nie bezpośrednio  $L$ , ponieważ jeśli zastosujemy tutaj logarytm to z iloczynu będziemy przehodzić na sumę i łatwo nam wyznaczyć pochodną a szukanie maksimum funkcji sprawdza się do przyrównania pierwszej pochodnej do 0).

W metodzie Największej Wiarygodności wyróżni się dwa podejścia **bezwarunkowe i warunkowe**. Metoda bezwarunkowa to standardowa, właściwsza i ile liczba parametrów jest stosunkowo mała w porównaniu do liczebności próby. Metoda warunkowa jest właściwsza o ile liczba parametrów jest stosunkowo duża w porównaniu do liczebności próby. Jeśli mamy wątpliwości lepiej skorzystać z podejścia warunkowego.

Po wyznaczeniu wartości estymatorów należy obliczyć ich średnie błędy szacunku. Średnie błędy szacunku wyznacza się na podstawie macierzy kowariancji.

Nieznanego parametry beta szacujemy na podstawie próby losowej. Próba powinna być reprezentatywna dla populacji. Z powodu braku informacji(brak udzielonej odpowiedzi w ankiecie) stosuje się wagę mające na celu zapewnić zgodność próby i populacji. Suma wag musi być równa liczbie obserwacji.

Podstawowe zrozumienie metody największej wiarygodności dla modelu logitowego może pomóc w rozjaśnieniu większości sekretów tej techniki. Może też pomóc w zrozumieniu jak i dlaczego czasami powstają błędy. Zacznijmy od oznaczeń i zatożeń. Posiadamy dane dotyczące w jednostek ( $i=1, \dots, n$ ), co do których zakładamy, że są statystycznie niezależne. Dla każdej jednostki  $i$ , dane składają się z  $y_i$  i  $x_i$ , gdzie  $y_i$  jest zmienną losową z możliwymi wartościami 0 i 1, a  $x_i = [1 \ x_{i1} \ \dots \ x_{ik}]'$  jest wektorem zmiennych objaśniających (1 jest dla wyrazu wolnego). Dla uproszczenia traktujemy  $x_i$  raczej jako wektor ustalonych wartości zmiennych niż jako wektor zmiennych losowych. Przyjmując, że  $p_i$  jest prawdopodobieństwem, że  $y_i=1$ , zakładamy, że dane są generowane przez model logitowy, który pokazuje, że

$$p_i = \frac{1}{1 + e^{-\beta x_i}}$$

Teraz konstruujemy funkcję wiarygodności, która wyraża prawdopodobieństwo obserwacji naszych danych jako funkcji nieznanego parametrów. Prawdopodobieństwo zaobserwowania wartości  $y$  dla wszystkich obserwacji można zapisać jako

$$L = \Pr(y_1, y_2, \dots, y_n)$$

Ponieważ zakładamy, że obserwacje są niezależne, ogólne prawdopodobieństwo zaobserwowania wszystkich  $y_i$  może być wyrażone jako iloczyn indywidualnych prawdopodobieństw:

$$L = \Pr(y_1) \Pr(y_2) \dots \Pr(y_n) = \prod \Pr(y_i),$$

gdzie  $\prod$  oznacza powtarzane mnożenie.

Z definicji  $\Pr(y_i=1)=p_i$  a  $\Pr(y_i=0)=1-p_i$ . To oznacza, że możemy zapisać:

$$\Pr(y_i) = p_i^{y_i} (1-p_i)^{1-y_i}$$

W tym równaniu  $y_i$  działa jako przełącznik włączając i wyłączając części równania. Kiedy  $y_i=1$ ,  $p_i$  podniesione do potęgi  $y_i$  wynosi po prostu  $p_i$ . Ale  $1-y_i$  wynosi wtedy 0, a  $(1-p_i)$  podniesione do potęgi 0 wynosi 1. Odwrotnie jest, jeśli  $y_i=0$ . Podstawiając wyrażenie i wykonując działania algebraiczne, otrzymujemy:

$$L = \prod_{i=1}^n p_i^{y_i} (1-p_i)^{1-y_i} = \prod_{i=1}^n \left( \frac{p_i}{1-p_i} \right)^{y_i} (1-p_i).$$

Logarytmując obie strony równania, otrzymujemy:

$$\log L = \sum_i y_i \log \left( \frac{p_i}{1-p_i} \right) + \sum_i \log(1-p_i)$$

Ogólnie jest łatwiej pracować z logarymem funkcji wiarygodności, ponieważ iloczyn przekształca się w sumę, a wykładniki potęg stają się współczynnikami. Ponieważ logarytm jest funkcją rosnącą, cokolwiek maksymalizuje logarytm, będzie także maksymalizować pierwotną funkcję.

Podstawiając nasze wyrażenie dla modelu logitowego otrzymujemy

$$\log L = \sum_i \beta x_i y_i - \sum_i \log(1 + e^{\beta x_i}),$$

co daje najbardziej uproszczoną postać  $\ln$  funkcji wiarygodności możliwą do uzyskania. To nas prowadzi do kroku 2, wyboru wektora wartości  $\beta$ , który maksymalizuje  $\ln$  funkcji wiarygodności. Istnieje wiele różnych metod maksymalizacji takich funkcji. Jednym z popularnych podejść jest znalezienie pochodnych funkcji w odniesieniu do  $\beta$ , porównanie pochodnych do 0 i następnie rozwiązanie dla  $\beta$ . Wyznaczenie pochodnych  $\ln L$  i porównanie ich do 0 daje:

$$\begin{aligned} \frac{\partial \log L}{\partial \beta} &= \sum_i x_i y_i - \sum_i x_i (1 + e^{-\beta x_i})^{-1} \\ &= \sum_i x_i y_i - \sum_i x_i \hat{y}_i = 0 \end{aligned}$$

gdzie

$$\hat{y}_i = \frac{1}{1 + e^{-\beta x_i}}$$

przewidywane prawdopodobieństwo, że  $Y=1$  dla danego wektora wartości  $x_i$ . Ponieważ  $x_i$  jest wektorem, równanie jest w rzeczywistości systemem  $k+1$  równań, po jednym równaniu dla każdego elementu wektora  $\beta$ .

Osoby zaznajomione z teorią OLS mogą rozpoznać drugi wiersz równania jako identyczny do normalnych równań dla modelu liniowego. Różnica jest taka, że  $\hat{y}$  jest liniową funkcją  $\beta$  w modelu liniowym, ale nieliniową funkcją  $\beta$  w modelu logitowym. W konsekwencji, z wyjątkiem szczególnych przypadków, jak np. pojedyncza zmienność dychotomiczna, nie ma jawnego rozwiązania układu  $k+1$  równań. Zamiast tego, musimy polegać na metodach iteracyjnych, które prowadzą do kolejnych przybliżeń do rozwiązania do momentu, kiedy przybliżenia 'zbiegną' się (będą dostatecznie bliskie) do właściwej wartości. Ponownie, jest wiele różnych metod iteracyjnych. Wszystkie dają takie samo rozwiązanie, ale różnią się takimi czynnikami jak szybkość konwergencji, wrażliwość na wartości początkowe i trudności obliczania dla każdej iteracji.

Jedną z szerszej znanych metod iteracyjnych jest metoda Newton-Raphson'a, która może być opisana następująco:

Niech  $U(\beta)$  będzie wektorem pierwszych pochodnych logarytmu L w odniesieniu do  $\beta$  i niech  $I(\beta)$  będzie macierzą drugich pochodnych log L w odniesieniu do  $\beta$ . To jest,

$$U(\beta) = \frac{\partial \log L}{\partial \beta} = \sum_i x_i y_i - \sum_i x_i \hat{y}_i$$

$$I(\beta) = \frac{\partial^2 \log L}{\partial \beta \partial \beta'} = -\sum_i x_i x_i' \hat{y}_i (1 - \hat{y}_i)$$

Wektor pierwszych pochodnych  $U(\beta)$  jest czasem nazywany gradientem lub oceną. Macierz drugich pochodnych  $I(\beta)$  jest nazywana Hessianem. Zatem algorytm Newton-Raphson'a to

$$\beta_{j+1} = \beta_j - I^{-1}(\beta_j) U(\beta_j),$$

gdzie  $I^{-1}$  jest odwrotnością macierzy  $I$ .

W praktyce, potrzebny jest zbiór startowych wartości  $\beta_0$ . PROC LOGISTIC po prostu rozpoczyna ze wszystkimi współczynnikami równymi 0. Te wartości startowe są podstawione do prawej strony równania, co daje wynik pierwszej iteracji  $\beta_1$ . Następnie wartości są z powrotem podstawiane do prawej strony równania, pierwsze i drugie pochodne są przeliczane, a wynikiem jest  $\beta_2$ . Proces ten jest powtarzany do momentu, kiedy maksymalna zmiana w każdym oszacowaniu parametrycznym w kolejnym kroku jest mniejsza niż pewne kryterium. Jeśli wartość absolutna ostatniego oszacowania parametrycznego  $\beta_j$  jest mniejsza lub równa 0,01, standardowe kryterium konwergencji jest określone jako:

$$|\beta_{j+1} - \beta_j| < .0001$$

Jeśli ostatnie oszacowanie parametryczne jest większe niż 0,01 (wielkość absolutna), kryterium standardowe określa

$$\left| \frac{\beta_{j+1} - \beta_j}{\beta_j} \right| < .0001$$

Po znalezieniu rozwiązania  $\hat{\beta}$ , produktem ubocznym algorytmu Newtona-Raphson'a jest oszacowanie macierzy kowariancji estymatorów współczynników, czyli po prostu  $-I^{-1}(\hat{\beta})$ .

Macierz ta, którą można wydrukować naciskając COBV jako opcję w instrukcji MODEL (dla GENMOD lub LOGISTIC), jest często użyteczna przy konstruowaniu testów hipotez o liniowej kombinacji współczynników. Oszacowania błędów standardowych estymatorów współczynników uzyskuje się z pierwiastków kwadratowych elementów głównej przekątnej tej macierzy.

## 76. Interpretacja wyników oszacowań parametrów modelu regresji logistycznej.

Model regresji logistycznej jest szczególnym przypadkiem uogólnionego modelu liniowego. Znajduje zastosowanie, gdy zmienność zależna jest dychotomiczna, to znaczy przyjmuje tylko dwie wartości takie jak na przykład sukces lub porażka, wystąpienie lub brak pewnej jednostki chorobowej, kobieta lub mężczyzna. W zapisie matematycznym wartości te reprezentowane są jako 1 i 0.

### Interpretacja współczynników

$$\frac{P(x)}{1 - P(x)} e^{\alpha + \beta x}$$

Wynika z tego, że wraz ze wzrostem wartości  $x$  o 1 jednostkę szansa wzrasta  $e^\beta$  razy. Zauważmy, że dla  $\beta = 0$  prawdopodobieństwo przyjęcia przez zmenną wynikową Y wartości 1 nie zależy od zmiennej X (jest stałe). Jeśli  $\beta > 0$ , to  $e^\beta > 1$ , więc przyrost wartości  $x$  wiąże się ze wzrostem prawdopodobieństwa sukcesu. Dla  $\beta < 0$  wiąże się ze spadkiem.

Interpretację współczynników dla dychotomicznej zmiennej objaśniającej ułatwia nam pojęcie ilorazu szans (ang. odds ratio). Przypuśćmy, że badamy występowanie pewnego zjawiska w dwóch grupach. Niech dla przykładu prawdopodobieństwo wygrania w pewnej grze wynosi 0,8 dla kobiet i 0,5 dla mężczyzn. Zgodnie z przedstawioną wcześniej formułą szanse wynoszą odpowiednio  $0,8/0,2 = 4$  oraz  $0,5/0,5$ . Iloraz szans w naszym przypadku wynosi więc 4/1. Możemy powiedzieć, że szansa wygranej jest 4 razy większa u kobiet niż u mężczyzn. W ogólności wartości powyżej jedynki oznaczają, że prawdopodobieństwo zajścia danego zjawiska jest większe w pierwszej z porównywanych grup.

## Interpretacja modelu regresji logistycznej

**Szansa (odds)** – iloraz prawdopodobieństwa, że wydarzenie nastąpi przez prawdopodobieństwo, że wydarzenie nie nastąpi.

### Przykład 3 c.d.

Dla klienta w wieku lat 20 otrzymaliśmy, oszacowanie prawdopodobieństwa, że klient odpowie pozytywnie na kampanię marketingową: 0,62, że negatywnie: 0,38. Zatem szansa dla tego klienta  $0,62/0,38=1,63$ .

Wartość szansy dla binarnej regresji logistycznej ze zmienną objaśniającą o dwóch wartościach:

$$\frac{P(y=1|x=1)}{1-P(y=1|x=1)} = \frac{e^{\beta_0+\beta_1}/(1+e^{\beta_0+\beta_1})}{1/(1+e^{\beta_0+\beta_1})} = e^{\beta_0+\beta_1}$$

$$\frac{P(y=1|x=0)}{1-P(y=1|x=0)} = \frac{e^{\beta_0}/(1+e^{\beta_0})}{1/(1+e^{\beta_0})} = e^{\beta_0}$$

24

## Interpretacja modelu regresji logistycznej

### Iloraz szans (odds ratio, OR)

$$OR = \frac{P(y=1|x=1)}{1-P(y=1|x=1)} / \frac{P(y=1|x=0)}{1-P(y=1|x=0)} = \frac{e^{\beta_0+\beta_1}}{e^{\beta_0}} = e^{\beta_1}$$

### Rzyko względne (relative risk)

$$\frac{P(y=1|x=1)}{P(y=1|x=0)}$$

Do interpretacji wyników estymacji oprócz ilorazów szans wykorzystuje się również **efekty krańcowe**. W modelu logitowym efekt krańcowej zmiany  $X_i$  na wartość  $p_i$  wynosi:

$$\frac{\partial p_i}{\partial X_{ji}} = \beta_j \frac{\exp(\mathbf{x}_i \boldsymbol{\beta})}{(1 + \exp(\mathbf{x}_i \boldsymbol{\beta}))^2} = \beta_j p_i (1 - p_i).$$

Efekty krańcowe są najczęściej liczone dla średnich wartości zmiennych objaśnianych i dotyczą zmian wartości prawdopodobieństwa.

25

#### Przykład 4.1

Analiza spłaty zobowiązań klientów banku.

Model binarnej regresji logistycznej z jakościową zmienną objaśniającą o dwóch wartościach.

Zmienna objaśniana *default*:  $y$  (0 – klient spłacił kredyt, 1 – klient nie dotrzymał zobowiązań).

Zmienna objaśniająca *inne zobowiązania*:  $x_{inne}$  (0 – nie, 1 – tak).

Parametry oszacowano na podstawie 1500 elementowej próby:

$$\hat{P}(y=1|x) = \frac{e^{-2,996+1,744x_{inne}}}{1+e^{-2,996+1,744x_{inne}}} = \frac{1}{1+e^{-(2,996+1,744x_{inne})}}$$

- Oszacowane prawdopodobieństwo, że klient mając inne zobowiązania ( $x_{inne}=1$ ) nie spłaci kredytu wynosi: 0,222.
- Oszacowane prawdopodobieństwo, że klient nie mając innych zobowiązań ( $x_{inne}=0$ ) nie spłaci kredytu wynosi: 0,048.

26

### Modele regresji logistycznej – przykłady

#### Przykład 4.1 c.d.

	$x_{inne} = 0$	$x_{inne} = 1$	Suma
$y = 0$	1000	350	1350
$y = 1$	50	100	150
Suma	1050	450	1500

- Szansa dla tych, co nie spłaciły zobowiązań i mieli inne zobowiązania:

$$100/350 = 0,286$$

- Szansa dla tych, co nie spłaciły zobowiązań i nie mieli innych zobowiązań:

$$50/1000 = 0,05$$

$$OR = \frac{100/350}{50/1000} = 5,72$$

- Ilość szans równa 5,72 oznacza, że jest przeszło pięć razy bardziej prawdopodobne, że klienci, którzy mają inne zobowiązania nie spłacą kredytu, niż klienci, którzy innych zobowiązań nie mają.

Dla regresji linowej:

$$y = 0,2x + 1$$

Wzrost zmiennej x o 1 jednostkę powoduje wzrost zmiennej y o 1,2 razy.

Dla regresji logistycznej:

$$\log(OR) = 0,2x$$

Wzrost zmiennej x o 1 jednostkę powoduje wzrost logarytmu z ilorazu szans o 1,2 razy (o 20%).

Nie można przejść z OR na prawdopodobieństwo tylko dla konkretnego przykładu.

### Modele regresji logistycznej bez transformacji zmiennych – wyniki i ich interpretacja

Effect	Odds Ratio Estimates	Point Estimate
D2_31 0 vs 4	<0.001	
D2_31 1 vs 4	<0.001	
D2_31 2 vs 4	0.001	
D2_31 3 vs 4	0.145	
D4_11 1 vs 8	1.200	
D4_11 2 vs 8	0.955	
D4_11 3 vs 8	0.798	
D4_11 4 vs 8	0.686	
D4_11 5 vs 8	0.655	
D4_11 6 vs 8	2.985	
D4_11 7 vs 8	0.364	
D5_4 0 vs 2	1.567	
D5_4 1 vs 2	1.607	
WIEK	1.053	0.953
w01_ 1 vs 12	0.230	
w01_ 2 vs 12	0.271	
w01_ 3 vs 12	0.222	
w01_ 4 vs 12	0.344	
w01_ 5 vs 12	0.184	
w01_ 6 vs 12	0.517	
w01_ 8 vs 12	0.242	
w01_ 9 vs 12	0.145	
w01_ 10 vs 12	0.260	
w01_ 11 vs 12	0.241	

Uzyskany wynik oznacza, że szansa przyjęcia przez zmienną objaśnianą kom wartości 1 dla respondentów z wykształceniem wyższym (D4\_11=1) jest o 20% większe niż dla respondentów bez wykształcenia (D4\_11=8).

Natomiast szansa dla osób z wykształceniem gimnazjalnym (D4\_11=6) jest 2.985 razy większa niż dla osób bez wykształcenia.

OR dla wieku (wyrażonego w latach) wynosi 0.953, co oznacza, że wraz ze wzrostem wieku o jedną jednostkę (1 rok), szanse na posiadanie komputera maleją o  $(1-0.953)*100\% = 4.7\%$ .

EXP(PARAM)

**77.** Weryfikacja istotności oszacowań parametrów regresji logistycznej.

Literatura opisuje dwa podejścia do testowania **istotności** współczynników regresji logistycznej:

- test wskaźnika wiarygodności ( likelihood ratio test): statystyka chi-kwadrat używająca  $-2 \ln L^*$
- test Wald'a: test Z używający błędów standardowych wyszczególnionych dla każdej zmiennej.

W przypadku dużych prób obydwie procedury dadzą w przybliżeniu takie same wyniki. Dla małych prób (lub umiarkowanych rozmiarów) możliwe są różne wyniki. Zalecany jest wtedy test wskaźnika wiarygodności.

#### Test wskaźnika wiarygodności

Stosując iloraz wiarygodności, odpowiadamy na pytanie, czy model zawierający zmienną (zmiennie) niezależne da nam lepsze przewidywanie wyników (czyli np. zachowania badanego) niż model niezawierający tej (tych) zmiennej(ych). Obliczanie tego współczynnika oznacza za każdym razem porównanie dwóch wartości statystyki wiarygodności, a konkretnie jej szczególnej postaci, czyli zlogarytmowanej wartości statystyki wiarygodności pomnożonej przez wartość  $-2$ .

Rozkład wartości ilorazu wiarygodności jest zgodny z rozkładem chi-kwadrat z tymi stopniami swobody, iloma zmiennymi różny się model pełny od modelu zredukowanego.

Wysokie wartości logarytmu wiarygodności oznaczają słabo dopasowany model regresyjny, gdyż im wyższa jego wartość, tym więcej zmienności zmiennej zależnej pozostałe niewyjaśnionej. Obliczając iloraz szans, porównujemy logarytm wiarygodności dla modelu zredukowanego (mniejszy model, zawierający mniejszą liczbę zmiennych niezależnych) z logarytmem wiarygodności dla modelu pełnego (większy model, zawierający więcej zmiennych niezależnych). Zwyczaje porównujemy modele różniące się od siebie jedną zmienią niezależną, po to, by sprawdzić, czy dodana zmienność zwiększa trafność przewidywań modelu.

#### Test Wald'a

Inną metodą testowania hipotez w regresji logistycznej jest współczynnik Walda (Z). Stosowany jest do testowania hipotez zerowych dla współczynników regresji logistycznej każdej zmiennej w modelu (hipotez o zerowej wartości współczynnika regresji, czyli o braku wpływu predyktora na zmiennej wynikowej  $H_0: \beta_i = 0$ ). Rozkład współczynnika Walda jest w przybliżeniu zgodny z rozkładem normalnym w dużych próbach. Natomiast rozkład współczynnika Walda podniesionego do kwadratu ( $Z^2$ ) zgodny jest z rozkładem chi-kwadrat z jednym stopniem swobody. W większości programów statystycznych współczynnik Walda przedstawiony jest w formie podniesionej do kwadratu, stąd często mówi się o współczynniku chi-kwadrat Walda. Wzór obliczeniowy dla współczynnika Walda opiera się na już wyestymowanych współczynnikach regresji oraz ich błędach standardowych.

#### Test wskaźnika wiarygodności

Jest aproksymowany rozkładem chi-kwadrat przy założeniu prawdziwości hipotezy zerowej, że porównywane modele nie różnią się od siebie.

$df = \text{stopnie swobody: różnica w liczbie parametrów w porównywanych modelach}$   
 $LR = -2\ln L_1 - (-2\ln L_2) = -2\ln(L_1/L_2)$

Statystyka LR porównuje dwa modele - pełny i zredukowany.

$H_0$ : parametry w pełnym modelu (dodatkowe względem modelu zredukowanego) są równe 0

#### Test Wald'a

Koncentracja na jednym parametrze, np:  $H_0: B=0$ .

Statystyka Walda:  $Z = \text{Beta}/\text{błąd parametru beta}$

Aproksymowane rozkładem  $N(0,1)$  przy założeniu prawdziwości  $H_0$ .

Może być także aproksymowane chi-kwadrat =  $Z^2$  z 1 st. swobody przy założeniu prawdziwości hipotezy 0.

#### Wnioskowanie na podstawie przedziałów ufności ocen ilorazów szans

Jeśli taki przedział zawiera 1, wówczas nieistotne na poziomie istotności zadanym dla przedziału.

## 78. Metody oceny dopasowania modelu regresji logistycznej do danych empirycznych.

Badanie dobroci dopasowania modelu możemy rozpoczęć, gdy jesteśmy usatysfakcjonowani z wstępnych wyników budowy modelu, to znaczy, uważamy, że model zawiera odpowiednie zmienne oraz ze zostały one włączone w odpowiedniej dla siebie formie funkcyjnej. Pod pojęciem dobroci dopasowania rozumie my stopień efektywności, w jakim model opisuje zmienne zależną. W przypadku regresji liniowej statystyki dopasowania opierają się na funkcjach bazujących na resztach pomiędzy wartościami zaobserwowanymi a przewidywanymi przez model. W regresji logistycznej istnieje kilka sposobów pomiaru różnic między obserwacjami a przewidywaniami.

**Przedstawienie miar dobroci dopasowania rozpoczęmy od dwóch rodzajów reszt: Persona i dewiancji.**

### Statystyka chi-kwadrat Persona i dewiancja

Do badania różnic pomiędzy wartościami zaobserwowanymi a teoretycznymi służą reszty Pearsona i dewiancji oraz obliczone na ich podstawie: statystyka chi-kwadrat Pearsona i dewiancja.

A więc statystyka chi-kwadrat Pearsona jest sumą kwadratów reszt Pearsona, dewiancja jest sumą kwadratów reszt dewiancji. Obie z tych statystyk przyjmują rozkład chi-kwadrat z liczbą stopni swobody równą  $J-(p+1)$ , gdzie  $j$  – liczba unikalnych kombinacji zmiennych niezależnych,  $p$  – liczba zmiennych niezależnych. Hipoteza zerowa zakłada dobrą dopasowaniego modelu do danych. Duża liczba profili w stosunku do liczby obserwacji, świadczy o tym, że w wielu komórkach tablicy kontyngencji znalazły się pojedyncze jednostki schematu odpowiedzi na poszczególne pytania, stanowiące zmienne wyjaśniające w modelu. Bez wątpienia ma to ogromny wpływ na jakość predykcji. W przypadku niewielkiej liczby m zaleca się pogrupowanie obserwacji w taki sposób, aby uzyskać m-asymptotyczne własności rozkładów powyższych statystyk.

### Test Hosmera i Lemeszowa

Test Hosmera i Lemeszowa bazuje na grupowaniu obserwacji ze względu na wartości wyestymowanych prawdopodobieństw. Najczęściej wyróżnia się dwie strategie grupowania obserwacji. Pierwsza polega na podziale wszystkich obserwacji na  $g=10$  grup, z czego pierwsza zawiera  $n_1=n/10$  obserwacji, charakteryzujących się najniższymi wartościami wyestymowanego prawdopodobieństwa, ostatnia zaś  $n_{10}=n/10$  obserwacji z najwyższymi wartościami. Inny podział polega na uporządkowaniu obserwacji po wcześniejszym ustaleniu punktów granicznych zdefiniowanych jako  $k/10$ , gdzie  $k=1,2,3,\dots,9$ . Do danej grupy wpadają obserwacje, które odznaczają się wartością wyestymowanego prawdopodobieństwa zawierającą się między wyznaczonymi punktami granicznymi. Po dokonaniu grupowania wycalane są wartości: dla  $y=1$  estymacje wartości oczekiwanych uzyskuje się poprzez sumowanie oszacowanych prawdopodobieństw obserwacji z grupy, dla  $y=0$  estymacje wartości oczekiwanych uzyskuje się przez odjęcie od 1 takiego oszacowanego prawdopodobieństwa.

### Test Score (Punktowy)

Inna miara badająca dobroć dopasowania jest statystyka zaproponowana przez Tsiatis a nazwana testem Score. Ogólna idea polega także na grupowaniu obserwacji, ale sam podział opiera się o arbitralnie ustalone wartości. Dla tak skonstruowanych g przedziałów tworzy się zmienne binarne, które określają, do którego przedziału prawdopodobieństwa wpada dana obserwacja. Nowe zmienne wprowadza się do modelu i za pomocą testu Score (o g-1 st. swobody) bada ich statystyczną istotność.

W sumie tu można dorzucić CHYBA macierz klasyfikacji, accuracy, sensitivity etc + krzywa roc + AUC.

## 79. Metody identyfikacji obserwacji odstających i wpływowych w regresji logistycznej.

+ Allison P. D., Logistic Regression Using SAS: Theory and Application, Second Edition. Cary, NC: SAS Institute Inc., 2012

Dla PROC LOGISTIC w SAS istnieje kilka sposobów na rozpoznanie wartości odstających i wpływowych. Są to metody inne niż np. reszty Pearsona i dewiancji, które opierają się także o wartość zmiennej przyjmującej wartość 0 lub 1. Istnieją specjalne statystyki pozwalające na poznanie wpływu konkretnych obserwacji:

### Obserwacje wpływowe , reszty wiarygodności

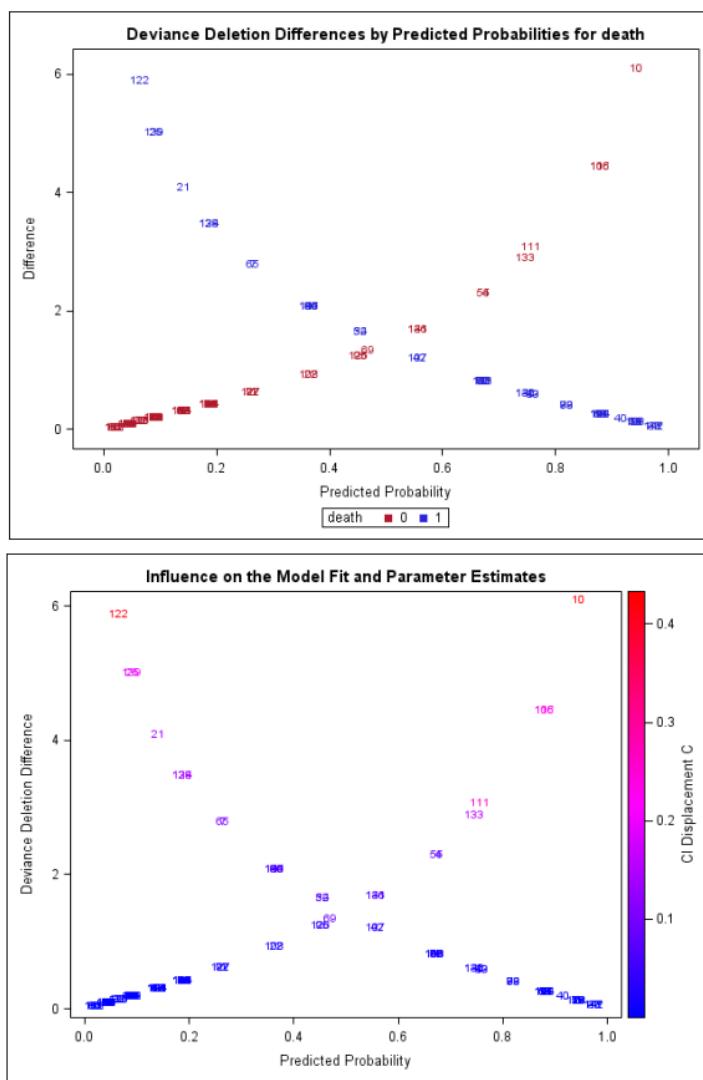
Obserwacje **wpływowe** zmieniają postać regresji, bo „przyciągają” do siebie równanie regresji. Obserwacje odstające osłabiają stabilność modelu, ponieważ zwiększą wartość składnika resztowego i co za tym idzie odchylenia standardowe oszacowań.

Wartości **wpływowe**: Np. diagonalna macierz kapeluszowa, wartości Cooka, czy DF Beta pozwalają na zidentyfikowanie wartości **wpływowych**, a następnie ich usunięcie. Df beta mówi o ile zmieni się dana beta po usunięciu konkretnej obserwacji. Duże zmiany bety są podstawą do usunięcia obserwacji.///

Potencjalną wartością **wpływową** nazywamy obserwację, dla której wartość zmiennej objaśniającej X znacząco odbiega od typowych wartości tej zmiennej. Obserwacja jest **wpływowa**, jeśli model istotnie zmienia się w zależności od obecności lub nieobecności tej obserwacji w zbiorze danych. Wartości **wpływowe** służą do wykrywania obserwacji **wpływowych**. W tym celu wykorzystuje Df Beta(jak się zmieni ocena parametru wyrazu wolnego po usunięciu danej obserwacji), diagonalną macierz kapeluszową, statystyki przesunięcia przedziału ufności C i CBar, reszty wiarygodności. ///

Reszty wiarygodności: Jest uśrednieniem reszty Pearsona i reszty Dewiancji (w pewnym sensie średnią ważoną standaryzowanych reszt Pearsona i dewiancji), służy do identyfikacji wartości odstających oraz **wpływowych**. Reszty Pearsona i Dewiancji mówią w jakim stopniu zmieniają dewiancję i reszty pearsona po usunięciu danej obserwacji.

- DFBETAS - pokazuje jak zmienią się współczynniki regresji w przypadku, gdy dana obserwacja zostanie usunięte ze zbioru danych. Rzeczywista zmiana jest dzielona przez błąd standardowy współczynnika.
- DIFDEV - zmiana w dewiancji po usunięciu danej obserwacji
- DIFCHISQ - miana Chi-kwadratu Pearsona wraz z usunięciem obserwacji
- C and CBAR - mierzy zmianę we współczynnikach regresji, podobne do odległości Cooka w regresji liniowej
- LEVERAGE - mierzy ekstermalność obserwacji w przestrzeni zmiennych objaśniających. Miara ta stanowi przekątną macierzy kapeluszowej.



80. Omów model wielomianowej regresji logistycznej.

## Wstęp

Binarna regresja logistyczna jest idealna, gdy zmienna zależna ma dwie kategorie, jednak co jeśli ma trzy lub więcej? W niektórych przypadkach uzasadnione może być zwiniecie kategorii do dwóch, ale ta strategia wiąże się z pewną utratą informacji. W innych przypadkach ograniczenie do dwóch kategorii może przysłonić to co chcemy zbadać. Przykładowo założmy, że chcemy oszacować model przewidujący, czy nowo zarejestrowani wyborcy zdecydują się głosować na demokratów, republikanów albo kandydatów niezależnych. Łączenie dowolnych dwóch kategorii może prowadzić do bardzo mylących wniosków.

## Przykład

Przeprowadzono ankietę na temat 195 studentów w celu badania wpływu stylów rodzicielskich na zachowania altruistyczne. Jedno z pytań brzmiało: "Jeśli znalazłeś portfel na ulicy, czy

- (1) zatrzymasz portfel i pieniądze
- (2) zatrzymasz pieniądze i zwróciisz portfel
- (3) zwróciisz portfel i pieniądze

Rozkład odpowiedzi dla zmiennej PORTFEL był następujący:

- (1) 24 zatrzyma oba,
- (2) 50 zatrzyma pieniądze,
- (3) 121 zwróci oba

Możliwe zmienne objaśniające to:

PLEĆ	1 = mężczyzna, 2 = kobieta
BIZNES	1 = uczestnik zajęć z biznesu 0 = w przeciwnym przypadku
KARA	W 1 = ukarany fizycznie przez rodziców w szkole podstawowej, ale nie gimnazjum lub liceum 2 = ukarany w szkole podstawowej i gimnazjum, ale nie w liceum 3 = ukarany na wszystkich 3 poziomach
EXPLAIN	"Gdy zostałeś ukarany, czy rodzice wyjaśnili dlaczego to, co zrobileś była zle?" 1 = prawie zawsze 0 = czasami lub nigdy

## SAS

Poniżej kod SASowy do oszacowania wielomianowego modelu logit dla danych z powyższego przykładu z portfelem:

```
PROC LOGISTIC DATA=wallet;  
  MODEL wallet = male business punish explain / LINK=GLOGIT;  
  RUN;
```

Nieuporządkowany model wielomianowy jest wywołany przez opcję LINK=GLOGIT. Jeśli ta opcja zostanie pominięta, to procedura LOGISTIC oszacuje skumulowany logit, który zakłada, że poziomy odpowiedzi są uporządkowane.

W regresji wielomianowej jest więcej niż 2 kategorię zmiennych. Mogą to być zmienne nominalne np. 3 różne kolory oczu. Nie są to zmienne porządkowe (które mają jakiś zachowany porządek – wtedy regresja porządkowa).

W SAS nieuporządkowany model wielomianowy jest wywoływany przez opcję LINK = GLOGIT.

### Model – zmienne – parametry

Uporządkowany	
Zmienne	Parametry
Intercept	$\alpha_1, \alpha_2, \dots, \alpha_{G-1}$
$X_1$	$\beta_1$
O postaci wielomianu	
Zmienne	Parametry
Intercept	$\alpha_1, \alpha_2, \dots, \alpha_{G-1}$
$X_1$	$\beta_1, \beta_2, \dots, \beta_{(G-1)}$

Jest Intercept dla każdego z G-1 porównań, ale dla danego predyktora dla każdego z G-1 poziomów jest jeden parametr  $\beta$

Odds ratios są niezmienne

### **Model dychotomiczny (0,1) vs. model o postaci wielomianu: Odds vs “odds-like”**

$$\text{logit } P(\mathbf{X}) = \ln \left[ \frac{P(D=1|\mathbf{X})}{P(D=0|\mathbf{X})} \right] = \alpha + \sum_{i=1}^p \beta_i X_i$$

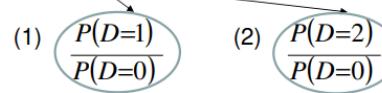
Odds dla stanu choroby o postaci poniżej jest wskaźnikiem prawdopodobieństwa (odds jako ratio of probabilities)

#### **Wynik dychotomiczny:**

$$odds = \frac{P(D=1)}{1-P(D=1)} = \frac{P(D=1)}{P(D=0)}$$

#### **Wynik dla trzech kategorii:**

Wykorzystanie “odds-like” wyrażenia dla dwóch porównań



Forma logitowa modelu wykorzystuje  $\ln$  z wyrażeń “odds-like”

$$(1) \quad \ln \left[ \frac{P(D=1)}{P(D=0)} \right] \quad (2) \quad \ln \left[ \frac{P(D=2)}{P(D=0)} \right]$$

Jeśli są 3 kategorie wynikowe, suma prawdopodobieństw powinna dać 1.

$$P(D=0) + P(D=1) + P(D=2) = 1$$

Ponieważ każde porównanie dotyczy dwóch prawdopodobieństw, prawdopodobieństwo we wskaźniku (probability in the ratio) nie sumuje się do 1. Tak więc dwa „odds-like” nie są prawdziwymi odds (true odds).

ALE  $P(D=1) + P(D=0) \neq 1$        $P(D=2) + P(D=0) \neq 1$

**DLATEGO TEŻ:**  $\frac{P(D=1)}{P(D=0)}$     i     $\frac{P(D=2)}{P(D=0)}$

„odds-like” lecz “nie prawdziwy” odds  
 (jakkolwiek analizy ograniczone do dwóch kategorii)

## Model dla trzech kategorii, jeden predyktor ( $X_1$ ) - Wiek:

$$\ln \left[ \frac{P(D=1|X_1)}{P(D=0|X_1)} \right] = \alpha_1 + \beta_{11} X_1 \quad \ln \left[ \frac{P(D=2|X_1)}{P(D=0|X_1)} \right] = \alpha_2 + \beta_{21} X_1$$

$\alpha_1 \longrightarrow 1 \text{ vs. } 0 \longleftarrow \beta_{11}$

$\alpha_2 \longrightarrow 2 \text{ vs. } 0 \longleftarrow \beta_{21}$

↑  
Prawdopodobieństwo, że wynik choroby jest w kategorii 2 dzielone przez prawdopodobieństwo, że wynik jest w kategorii 0.

Oba  $\alpha$  i  $\beta$  mają informację (subskrypty), czego dotyczą (jakich porównań dotyczą)

## Dwa wskaźniki Odds

$OR_1$  ( cat.1 vs. Cat.0 ) – Adenosquamous vs. Adenocarcinoma

$OR_2$  ( cat.2 vs. Cat.0 ) - Inna vs. Adenocarcinoma

$$OR_1 = \frac{P(D=1|X=1) / P(D=0|X=1)}{P(D=1|X=0) / P(D=0|X=0)} \quad OR_2 = \frac{P(D=2|X=1) / P(D=0|X=1)}{P(D=2|X=0) / P(D=0|X=0)}$$

Każda wartość odds jest liczona według logiki wykorzystywanej w SLR

Wyznaczając  $\log OR_1$  i  $OR_2$  i wstawiając 2 wartości dla zmiennej exposure otrzymujemy, że:

### Adenosquamous vs Adenocarinorma

$$OR_1 = \frac{\exp[\alpha_1 + \beta_{11}(1)]}{\exp[\alpha_1 + \beta_{11}(0)]} = e^{\beta_{11}}$$

Odds ratio dla pierwszego porównania jest równe  $e^{\beta_{11}}$

### Inna vs Adenocarcinoma

$$OR_2 = \frac{\exp[\alpha_2 + \beta_{21}(1)]}{\exp[\alpha_2 + \beta_{21}(0)]} = e^{\beta_{21}}$$

...dla drugiego odpowiednio  $e^{\beta_{21}}$

$$OR_1 = e^{\beta_{11}} \quad \text{są różne!} \quad OR_2 = e^{\beta_{21}}$$

Zatem otrzymujemy dwa różne wyrażenia wykorzystujące  $\beta_{11}$  i  $\beta_{21}$ . Zatem kwantyfikacja powiązania pomiędzy exposure i wynikami zleży od tego, który poziom wyniku jest porównywany.

#### Ogólny przypadek dla jednego predyktora

$$OR_g = \exp[\beta_{g1}(X_1^{**} - X_1^*)]$$

Jeśli mamy dwie wartości predyktora  $X(1)$ , tj.  $X_1^{**}$  i  $X_1^*$ , wówczas postać formuły g oznacza kategorię zmiennej chorobowej (1;2), które porównuje się z kategorią odniesienia (0).

Tu każda kategoria (z pominięciem referencyjnej) ma swój własny model o innym wyrazie wolnym i innych współczynnikach beta.

Interpretacja dla interceptu dla kat dobre (przykład zakłada istnienie 4 poziomów:

b.dobre, dobre, średnie, złe lub bardzo złe(ref)) = 4,171 - gdy wszystkie zmienne objaśniające przyjmują wartość 0, szanse dobrej oceny zdrowia, a nie oceny złej i bardzo złej są jak 65 do 1. Z dwóch osób różniących się tylko wiekiem, o jeden rok starsza z nich ma szansę o 2.6% mniejsze by ocenić zdrowie jako średnie, a nie jako złe lub bardziej złe niż osoba od niej o rok młodsza.

Osoby z wykształceniem podstawowym i niższym mają o 70% niższe szanse ocenić swoje zdrowie jako bardzo dobre i o 78% niższe jako dobre, a nie złe lub bardzo złe(kat ref), niż osoby z wykształceniem policealnym i wyższym (kat ref).

Parameter	zdrowie	DF	Estimate	Standard		Wald	
				Error	Chi-Square	Pr>	Exp ChiSq (Est)
Intercept	b. dobre	1	4.0698	0.9175	19.6657	<.0001	58.485
Intercept	dobre	1	4.1707	0.6042	47.6483	<.0001	64.758
Intercept	średnie	1	1.3381	0.5703	5.5055	.0190	3.812
wiek	b. dobre	1	-0.1172	0.00901	169.4467	<.0001	0.089

wiek	dobre	1	-0.0705	0.00670	110.6289	<.0001
wiek	średnie	1	-0.0262	0.00632	17.1969	<.0001
plec K	b. dobre	1	-0.5680	0.2313	6.0309	0.0141
plec K	dobre	1	-0.5680	0.2313	0.7409	0.3894
plec K	średnie	1	-0.0243	0.1811	0.0182	0.002
edu podstawowe	b. dobre	1	-1.2039	0.4392	7.2146	0.0061
edu podstawowe	dobre	1	-1.5050	0.3669	16.7133	<.0001
edu podstawowe	średnie	1	-0.1790	0.3721	0.2097	0.6470
edu średnie	b. dobre	1	-0.1062	0.5329	0.0397	0.8420
edu średnie	dobre	1	-0.2779	0.4980	0.3115	0.5768
edu średnie	średnie	1	0.5938	0.5084	1.3641	0.2428
edu zawodowe	b. dobre	1	-0.4730	0.3333	1.3080	0.2395
edu zawodowe	dobre	1	-0.3988	0.3500	1.0867	0.2000
edu zawodowe	średnie	1	0.4973	0.3667	1.8396	0.1750
dochod cięcko	b. dobre	1	1.8558	0.7789	5.6772	0.0172
dochod cięcko	dobre	1	0.9081	0.4118	4.8631	0.0274
dochod cięcko	średnie	1	0.8477	0.3395	6.2347	0.0125
dochod radziny	b. dobre	1	0.9064	0.2535	12.7793	0.0004
dochod radziny	dobre	1	1.0998	0.2016	29.7344	<.0001
dochod radziny	średnie	1	1.4046	0.1682	6.4849	0.0109
dochod dostatnio	b. dobre	1	2.4674	1.1025	4.8775	0.0272
dochod dostatnio	dobre	1	1.9951	1.1473	3.0237	0.0821
dochod dostatnio	średnie	1	1.3340	1.1961	1.3314	0.2466

Odds Ratio Estimates						
Effect		zdrojowic	Point	95% Wald		Wald
			Estimate	Confidence	Limits	
wiek		b. dobre	0.889	0.874	0.905	
wiek		dobre	0.932	0.920	0.944	
wiek		średnie	0.974	0.962	0.986	
plec K vs M		b. dobre	0.567	0.360	0.692	
plec K vs M		dobre	0.848	0.582	1.235	
plec K vs M		średnie	1.025	0.719	1.461	
edu podstawowe vs polic. i wyższe		b. dobre	0.308	0.127	0.717	
edu podstawowe vs polic. i wyższe		dobre	0.340	0.109	0.438	
edu podstawowe vs polic. i wyższe		średnie	0.843	0.497	1.176	
edu średnie	vs polic. i wyższe	b. dobre	0.899	0.316	2.516	
edu średnie	vs polic. i wyższe	dobre	0.757	0.285	2.010	
edu średnie	vs polic. i wyższe	średnie	1.811	0.669	4.905	
edu zawodowe	vs polic. i wyższe	b. dobre	0.630	0.291	1.361	
edu zawodowe	vs polic. i wyższe	dobre	0.698	0.351	1.386	
edu zawodowe	vs polic. i wyższe	średnie	1.644	0.801	3.374	

## 81. Omów model proporcjonalnych szans.

Model proporcjonalnych szans, czyli regresja logitowa ze zmiennymi porządkowymi.

**Autorzy:** wprowadzony przez Claytona (1974) w medycynie. Kolejne badania przeprowadzone m.in. przez Kirmaniego i Guptę (2001).

**Zastosowanie:** gdy zmienna wynikowa ma więcej niż dwie wartości (kategorie/klasy) i wartości te występują w skali porządkowej.

Oznaczmy kategorie jako  $j = 1, 2, \dots, g$

$$\log \frac{Pr(y \leq j | \mathbf{x})}{1 - Pr(y \leq j | \mathbf{x})} = \alpha_j - \beta' \mathbf{x},$$

gdzie  $\mathbf{x} = (x_1, \dots, x_p)'$  jest wektorem predyktorów.

Model ten jest serią modeli logitowych, uporządkowanych według stopnia narastania intensywności cechy wynikowej. Dla ustalonego  $j$  model jest modelem logistycznej regresji dla odpowiedzi binarnej 1 gdy  $y \leq j$ , i 0 gdy  $y > j$ .

Np. gdy cecha wynikowa Y przyjmuje wartości uporządkowane: mały, średni, duży, olbrzymi, to modele logitowe byłyby utworzone według narastających poziomów dychotomicznych:

- mały & więcej niż mały;
- co najwyżej średni & więcej niż średni;
- co najwyżej duży & olbrzymi

Dla  $g = 2$  otrzymujemy model regresji logistycznej.

Proporcjonalność szans (proportional odds) polega na tym, że wszystkie te modele tworzą równolegle hiperplaszczyzny regresji. Oznacza to taki sam wpływ zmiennych objaśniających w każdej klasie intensywności cechy wynikowej. Zmiany prawdopodobieństw cechy wynikowej w tych klasach są niezależne od cech objaśniających.

Procedury: GENMOD w SAS i polr w R.

Estymacja: przy użyciu metody największej wiarygodności.

Zmienne	Estymator	S.E.
Intercept1	-1.7388	0.1765
Intercept2	-0.0089	0.1368
RACE	0.7555	0.2466

Dwa wyrazy wolne (Intercept)- jeden dla każdego porównania, ale jest jeden parametr  $\beta$  dla pomiaru efektu rasy!

Odds ratio dla rasy jest równe  $e^{\beta_1}$

$$OR = \exp(0.7555) = 2,13$$

## Interpretacja wyrazów wolnych ( $\alpha_g$ )

Interpretacja wyrazów wolnych jest podobna do interpretacji w innych modelach regresji logistycznej

$\alpha_g = \log \text{odds } D \geq g$ , gdzie wszystkie zmienne niezależne są równe 0;  
 $g = 1, 2, 3, \dots, G-1$

W modelu proporcjonalnych odds modelujemy log odds kilku nierówności, co w konsekwencji daje kilka wyrazów wolnych, każdy wyraz wolny koresponduje z inną nierównością, czyli wartością g. Ponadto, zakładając, że kategorie g nie są puste, log odds  $D \geq g$  jest większe niż log odds dla  $D \geq (g+1)$ . To oznacza, że

$$\begin{array}{c} \alpha_g > \alpha_{g+1} \\ \Downarrow \\ \alpha_1 > \alpha_2 > \dots > \alpha_{G-1} \end{array}$$

Wyniki dla kobiet z inwazją nowotworową wskazują, że u KOBIET CZARNYCH są ponad dwukrotnie (tj. 2,13) większe szanse na zdiagnozowanie guza w stadium Poorly Differentiated niż w stadium Moderately lub Well Differentiated niż u KOBIET BIAŁYCH oraz ponad dwukrotnie większe szanse na zdiagnozowanie guza w stadium Poorly lub Moderately Differentiated niż w stadium Well Differentiated w porównaniu do BIAŁYCH KOBIET.

Podsumowując, w badanej kohortie Kobiety CZARNE w porównaniu do kobiet BIAŁYCH ze zdiagnozowanym guzem endometriozy są ponad dwukrotnie bardziej narażone na bardziej ostrą (zaawansowaną) odmianę guza.

Ponieważ:

$$\hat{OR}(D \geq 2) = \hat{OR}(D \geq 1) = 2,13$$

Kategorie zmiennych porządkowych można podzielić na dwie grupy, stawiając linie podziału między dowolnymi dwiema sąsiadującymi kategoriami. mając te podziały, można oszacować model podobny do modelu wielomianowego. Zwykle zatrudnia się, że parametry przy zmiennych objaśniających są takie same, niezależnie od tego gdzie postawiono linie podziału, a różnią się jedynie stałe odpowiednich kombinacji liniowych (wyrazy wolne). Założenie nazywa się to założeniem proporcjonalnych szans. Funkcja łączająca opiera się o logit i jest zwana skumulowanym logitem.

Weryfikacja założenia - Score test - porównanie do ogólnego modelu porządkowego, który różni się od modelu porządkowego tym, że wszystkie elementy wektorów Beta<sub>k</sub> są

różne. W modelu nieogólnionym wektory te różniły się tylko stałą. H0 mówi o równości odpowiednich elementów elementów Beta, z wyjątkiem stałej. Jeśli 0 prawdziwa to ogólniony sprowadza się do standardowego i jest spełnione założenie proporcjonalności. Statystyka testowa opiera się o szacowanie pochodnej funkcji wiarygodności w punkcie wyznaczonym przez H0. Ma rozkład chi-kwadrat z df=ilczba zmiennych objaśniających \*(liczba poziomów zmiennych-1).

Dla G kategorii  $\Rightarrow G-1$  sposobów na zdefiniowanie zmiennej dychotomicznej, która jest zmienną wynikową:

$D \geq 1$  vs.  $D < 1$ ;

$D \geq 2$  vs.  $D < 2, \dots$ ;

$D \geq G-1$  vs.  $D < G-1$ ;

$$\text{odds } (D \geq g) = \frac{P_{\text{rawdopodobieństwo}}(D \geq g)}{P_{\text{rawdopodobieństwo}}(D < g)} \quad \text{gdzie } g = 1, 2, 3, \dots, G-1$$

**Założenie o proporcjonalności odds jest ważnym założeniem w modelowaniu**

Przy założeniu proporcjonalności odds, Odds Ratio szacujący efekt wpływu wystawienia na ryzyko (zmiennej E, Exposure) dla dokonanych porównań będzie równy, niezależnie od tego, gdzie będzie punkt cięcia (cut-point), czyli gdzie zostanie postawiona linia (łączenia kategorii).

Zakładając, że mamy zmienną wynikową z 5-kategoriami i jedną zmienną E z kategoriami [E=0, E=1]. Przyjmując założenie o proporcjonalności odds, oznacza że odds ratio, które porównuje kategorie  $\geq 1$  i  $< 1$  jest taki sam jak odds ratio, które porównuje kategorie  $\geq 4$  i  $< 4$ .

Innymi słowy: Odds ratio jest niezmienny, niezależnie od tego w jaki sposób przeprowadza się dychotomizację zmiennej wynikowej (umiejscawia cut-point).

**Przykład:**  $OR(D \geq 1) = OR(D \geq 4)$

Porównując dwie grupy wystawienia na ryzyko:

E=1 vs. E=0, gdzie:

$$OR(D \geq 1) = \frac{\text{odds}[(D \geq 1) | E=1]}{\text{odds}[(D \geq 1) | E=0]} \quad OR(D \geq 4) = \frac{\text{odds}[(D \geq 4) | E=1]}{\text{odds}[(D \geq 4) | E=0]} \quad 6$$

### Ilustracja

$$\alpha_1 = \log \text{odds } D \geq 1$$

0	1	2	3	4
---	---	---	---	---

$$\alpha_1 = \log \text{odds } D \geq 1$$

$$\alpha_2 = \log \text{odds } D \geq 2$$

0	1	2	3	4
---	---	---	---	---

$$\alpha_2 = \log \text{odds } D \geq 2$$

$$\alpha_3 = \log \text{odds } D \geq 3$$

0	1	2	3	4
---	---	---	---	---

$$\alpha_4 = \log \text{odds } D \geq 4$$

0	1	2	3	4
---	---	---	---	---

$$\alpha_3 = \log \text{odds } D \geq 3$$

0	1	2	3	4
---	---	---	---	---

$$\alpha_4 = \log \text{odds } D \geq 4$$

$$\log \text{odds}(D \geq 1) \geq \log \text{odds}(D \geq 2) \geq \log \text{odds}(D \geq 3) \geq \log \text{odds}(D \geq 4)$$

Analysis of Maximum Likelihood Estimates						
Parameter		DF	Estimate	Standard Error	Chi-Square	Pr > ChiSq
Intercept bardzo dobre	1	-0.2330	0.3200	0.5303	0.4665	0.792
Intercept dobre	1	2.4857	0.3283	57.3437	<.0001	12.010
Intercept średnie	1	4.4340	0.3111	198.0011	<.0001	128.222
wiek	1	-0.0582	0.0035	0.6526	<.0001	0.943
płeć K	1	-0.3033	0.0963	9.9204	0.0016	0.738
edu podstawowe i niżej	1	-0.9704	0.1715	32.6347	<.0001	0.379
edu średnie ogólnokształcące	1	-0.2794	0.1748	2.5567	0.1098	0.756
edu zawodowe	1	-0.4701	0.1451	10.5007	0.0012	0.625
dochód ciętko	1	0.7993	0.2656	9.0564	0.0026	2.224
dochód radziemy sobie	1	0.5797	0.1067	29.5435	<.0001	1.786
dochód dostateczny	1	0.7440	0.2064	12.9893	0.0003	2.104

Odds Ratio Estimates						
Effect		Point Estimate		95% Wald Confidence Limits		Wald
		Estimate	Lower	Upper		
wiek		0.943	0.937	0.950		
płeć	K vs M	0.738	0.611	0.852		
edu	podstawowe i niżej vs policealne i wyższe	0.379	0.271	0.530		
edu	średnie ogólnokształcące vs policealne i wyższe	0.756	0.537	1.065		
edu	zawodowe vs policealne i wyższe	0.625	0.470	0.830		

Tutaj:

Gdyby wszystkie wartości zmiennych = 0, dla kat dobre następująca interpretacja:

- szanse na co najmniej dobrą ocenę, a nie średnią, lub złą lub bardzo złą, są jak 12 do 1.

Dla kat średnie:

- szanse na co najmniej średnią ocenę, a nie złą lub bardzo złą są jak 129:1

Dla wieku:

- Jednoroczną różnicą w wieku powoduje, że osoba starsza ma o 6% niższe szanse na lepszą, a nie gorszą ocenę stanu zdrowia niż osoba młodsza.

Wyrażenie **lepsza, nie gorsza** odnosi się do pewnej ustalonej kategorii. Na mocy założenia proporcjonalnych szans, iloraz jest taki sam, niezależnie od tego, która to kategoria.

Dla płci:

- kobiety mają szanse o 22% niższe niż mężczyźni na ocenę swojego stanu zdrowia jako lepszego, niż gorszego.

Ważny jest tu score test, zakładający proporcjonalność szans dla H0. Jeśli jednak H1, wtedy należy użyć modelu wielomianowego.

### 82. Metody doboru zmiennych objaśniających w modelach regresji.

**KROKI BUDOWY MODELU** - Podstawowe kroki budowy modelu zakładają: wstępne określenie modelu (jego zmiennych objaśniających), dobór zmiennych za pomocą jednej z metod (zastosowanie iteracyjnej procedury) i ewentualne przerwanie procedury w momencie, gdy nie jest ona możliwa do przeprowadzenia przy danym kryterium krokowym lub gdy wyczerpana zostaje założona wstępnie maksymalna liczba kroków. Istnieją trzy wyróżniane metody selekcji iteracyjnej:

- **Metoda eliminacji (backward)** – polega na początkowym włączeniu do modelu wszystkich zakładanych zmiennych objaśniających i kolejnym eliminowaniu tych, które nie spełniają założonych kryteriów (nie wykazuje statystycznej istotności). Wykluczenie zaczyna się od tej zmiennej, która charakteryzuje najwyższą wartość prawdopodobieństwa  $p$  w teście istotności przekraczającą założone  $\alpha$ . Po usunięciu takiej zmiennej ponownie oblicza się odpowiednie miary i testy istotności dla pozostałych zmiennych i powtarza się krok eliminacji. Procedura kończy się, gdy w modelu nie występują już zmienne statystycznie nieistotne lub gdy wyczerpana została maksymalna założona liczba kroków.
- **Metoda dołączania (forward)** – polega na początkowym włączeniu do modelu jedynie stałej i kolejnym „dokładaniu” zmiennych objaśniających (maks. po jednej zmiennej w każdym kroku), których wartość prawdopodobieństwa  $p$  w testu istotności nie przekracza założonego  $\alpha$ . Dołączanie zaczyna się od tej zmiennej, która charakteryzuje najniższe prawdopodobieństwo  $p$ , to znaczy od zmiennej, która jest najsilniej związana ze zmienną objaśnianą. Procedura kończy się, gdy wśród potencjalnych zmiennych niezależnych nie włączonych do modelu nie ma już takiej, która wykazywałaby statystyczną istotność po włączeniu lub gdy wyczerpana została maksymalna założona liczba kroków.
- **Metoda selekcji krokowej (stepwise)** – łączy ze sobą zarówno metodę eliminacji jak i dołączania, ponieważ zakłada, że na każdym etapie można wykluczyć lub dodać nową zmienną do modelu. Założony poziom istotności dla włączenia bądź wykluczenia zmiennej może być różny ( $\alpha_1 \neq \alpha_2$ ).

### 83. Jakość danych w analizach biznesowych. Znaczenie i metody oceny.

Na jakość danych składa się m. in poprawność i kompletność. Niska jakość danych może prowadzić przede wszystkim do błędnych wniosków analizy, jest związana ze zwiększym kosztem za poprawę analizy, zwiększa poświęcony czas przy badaniu problemu.

Do oceny danych:

- Procentowy udział braków danych w całym zbiorze danych
- Obserwacje odstające
- Zastosowanie reguły 3 sigm

Jakość danych = wiarygodność danych.

**Dane wyróżnia:**

- poprawność
- kompletność

**Dwa podstawowe rodzaje danych, które określają sposób kontroli jakości danych:**

- Dane eksperymentalne - konfirmacyjna analiza statystyczna (wgłębny w schemat doboru próby, przegląd kwestionariusza)
- Dane obserwacyjne - (transakcje, zachowanie na stronie www) - eksploracyjna analiza statystyczna (ocena procesu zapisu danych)

**Dlaczego jest to ważne? Dane o niskiej jakości powodują:**

- Zwiększenie czasu i nakładów na projekt
- Zmniejszają czas na analizę danych
- Obniżają zaufanie do wyników analiz
- Spowalniają innowacje i badania
- Prowadzą do błędnych, obciążonych, nieaktualnych lub opóźnionych decyzji
- demotywację wśród analityków, a także ich ucieczkę z firmy.

**Właściwości danych:**

- Dostępność danych (zapis tradycyjny lub elektroniczny, możliwość ich wykorzystania)
- Wielkość danych (liczba obserwacji, cech, długość zapisu w kolumnach, zbiory ustrukturyzowane vs zbiory danych tekstowych)
- Kompletność danych (braki danych: staranność w gromadzeniu danych, odmowy odpowiedzi)
- Poprawność danych (dokładność i błąd pomiaru, metoda zbierania np. wizualna skala analogowa dla ocen, problem z integracją i transferem danych)

**Role podczas pracy nad jakością danych w rozwiązaniach biznesowych:**

- System operacyjny: poprawność danych klienta, czas reakcji itd.
- System analityczny: umiejętność wyciągania ważnych wniosków z modelowania, np. prognozowanie popytu.

**W systemie analitycznym występują dwa kluczowi użytkownicy danych:**

- Data Management: zbieranie, transfer, **weryfikacja, zapytania**
- Analityk: analiza, raportowanie, interpretacja, **weryfikacja, zapytania (część wspólna z Data Management)**

Statystyka pozwala nie tylko na wyciąganie wniosków przez modelowanie, ale także na poprawę jakości danych poprzez wykrywanie outlierów czy uzupełnianie danych.

#### **Jakość danych w zaawansowanej analityce:**

- Rozmiar próby uwzględniający wskaźnik odpowiedzi w danych historycznych
- Nacisk na zmienne objaśniające w modelowaniu predykcyjnym
- Odpowiednia częstość obserwacji w szeregu czasowym
- Kodowanie brakujących wartości, wpływ braków danych na modelowanie
- Ocena zgodności rozkładów z założeniami modelu

Dobre praktyki w procesie gromadzenia danych:

- Dane z zaufanego źródła
- W trakcie całego procesu biznesowego nie ma możliwości zmiany danych bez śledzenia zmian.
- Pochodzenie i poprawność danych można prześledzić w systemach źródłowych
- Proces zbierania danych jest wystandardyzowany (ośrodki biorące udział w badaniu działają w sposób wystandardyzowany)
- System wprowadzania i pobierania danych wymusza wprowadzanie poprawności danych
- Dane wykorzystywane w modelowaniu odzwierciedlają dane źródłowe
- Pobieranie danych jest częścią restrykcyjnego procesu. W procesie gromadzenia danych powinny być wyznaczone ścisłe, jednolite zasady.
- Sposób przetwarzania danych jest udokumentowany i znany
- Zmiany wprowadzane z czasem w procesie zbierania danych są znane, na przykład: Zmiany w definicji pól
  - Data odpowiadająca pierwszej rejestracji w systemie
  - Metoda doboru próby jest udokumentowana.

#### **Skale pomiarowe w kontekście badania jakości danych:**

- Nominalna
- Porządkowa
- Przedziałowa
- Ilorazowa

#### **Wybrane błędy w danych:**

- Ukryte znaki
- Niewystandardowane wartości
- Nietypowe wyniki
- Nieoczekiwany rozkład wyników
- Częściowe wartości daty i czasu, systematyczne zmiany w danych, odstępstwa od założonego procesu
- Braki danych
- Duplikaty

#### **Rozwiązań:**

- Czyszczenie danych
- Identyfikacja nietypowych obserwacji (odstających)
- Badania rozkładów

- Uporanie się z brakami danych (rozpoznanie typów braków danych, usunięcie lub dobranie rozwiązania do imputacji)

#### **84. Imputacja danych. Istota i znaczenie.**

Imputacja danych – sztuczne wstawianie pewnych wartości do tabeli danych. Na ogół imputacja jest wykonywana w celu usunięcia tzw. braków danych, czyli wartości nieznanych. Wiele metod statystycznych nie akceptuje bowiem obserwacji z brakami danych.

Braki danych są jednym z praktycznych problemów, związanych z prowadzeniem badań statystycznych. Są powodowane jakością procesu gromadzenia danych, ale także postawą respondentów, co może bezpośrednio przekładać się na odpowiedź na pytanie badawcze. Ogólnie biorąc, braki danych możemy rozpatrywać jako wynik całkowicie losowego procesu (np. problem z dotarciem do respondenta) lub systematycznego procesu. W pierwszym przypadku mówimyśmy o problemie związanym z jakością danych, w drugim o brakach odpowiedzi. W analityce biznesowej, w zależności od natury problemu, rozwiązaniem byłoby wprowadzanie kontroli jakości danych oraz zastosowanie metodyki analizy braków danych.

Braki danych powodowane są niedoskonałością procesu gromadzenia danych. Szukane wartości istnieją, nie możemy jednak do nich dostrzec ze względu na jakość badania, technologię, bądź postawę respondentów. Wyróżniamy braki danych:

- na poziomie badanej jednostki (unit nonresponse)
- na poziomie cechy (item nonresponse)

#### **Problemy i wyzwania związane z brakami danych**

- Brak kontroli własności oszacowań. Wyniki mogą być obciążone.
- Wzrost wariancji estymatorów (mniejsza precyja oszacowań).
- Niepewność co do właściwości ocen ma przełożenie na zdolność do odpowiedzi na pytanie badawcze – braki danych mogą prowadzić do niedoszacowania lub przeszacowania badanego efektu. Tradycyjne metody nie umożliwiają kontroli obciążenia oszacowań.
- Braki danych wiążą się ze wzrostem złożoności analiz.

#### **Cele analizy braków danych**

- minimalizacja obciążenia estymatorów
- ocena niepewności (korekta błędów standardowych)
- maksymalne wykorzystanie dostępnych informacji

#### **Metody imputacji danych:**

- metody polegające na usunięciu obserwacji z brakami danych
- metody wykorzystujące wagę (weighting methods)
- metody imputacji danych (imputation-based methods)
- metody oparte na modelach (model-based procedures)

#### **Metody współczesne:**

- metody bazujące na rozkładach prawdopodobieństw
- imputacja wielokrotna (multiple imputation) - podstawowym problemem przy stosowaniu tej metody jest wybór modelu, za pomocą którego uzupełniane będą braki danych. Wybór modelu uzależniony jest od typu danych. Dwa ogólne typy modeli dotyczą odpowiednio cech ciągłych i skokowych.
- metody wykorzystujące wagę (kalibracja)

Imputacja danych polega na zastąpienie braków danych poprzez wstawienie wartości szacowanych.

Trzy powody, dla których występują braki danych -> potem są różne strategie do niwelacji braków danych.

## **WSTĘPNIAK - BRAKI DANYCH**

Braki danych powodowane są niedoskonałością procesu gromadzenia danych. Szukane wartości istnieją, nie możemy jednak do nich dotrzeć ze względu na jakość badania, technologię, bądź postawę respondentów. Rozróżnia się ze względu na:

- całkowity brak danych dla jednostki wylosowanej do próby
- brak wartości wybranej cechy

### **Problemy związane z brakami danych:**

- Własności oszacowań dla pełnej próby nie są odpowiednie. Możliwe obciążenie oszacowań.
- Wzrost wariancji estymatorów
- Wykorzystanie metod imputacji danych wiąże się ze wzrostem złożoności analiz. Zwykle rozpatruje się kilka możliwych scenariuszy.

### **Wzorce brakujących danych**

Wzorzec braków danych nazywamy monotonicznym, gdy możliwe jest uporządkowanie zmiennych w taki sposób, że obserwacje brakujące dla zmiennej  $Y_i$  stanowią także braki dla kolejnych zmiennych  $Y_{i+1}, Y_{i+2}, \dots, Y_k$ .

### **Mechanizmy braków danych**

Określają one zależność pomiędzy prawdopodobieństwem wystąpienia braku danych a obserwowanymi lub nieobserwowanymi wartościami cech. Wyróżnia się:

- **MCAR** - missing completely at random - braki danych są całkowicie losowe, gdy prawdopodobieństwo braku dla  $Y_i$  nie zależy od wartości  $Y_i$ , ani też od wartości pozostałych cech. Przy założeniu istnienia takich braków, analizy dla ograniczonego zbioru danych prowadzą do nieobciążonych oszacowań szukanych parametrów.
- **MAR** - missing at random - braki danych są losowe, gdy prawdopodobieństwo braku dla  $Y_i$  nie zależy od wartości  $Y_i$ , ale zależy od wartości innych cech. Np. osoby starsze są bardziej skłonne do udzielania informacji o ich ocenie produktu. Weryfikacja założenia wymaga oceny eksperckiej, jednak przy jego prawdziwości i wykorzystaniu odpowiednich technik można otrzymać nieobciążone oszacowania szukanych parametrów.
- **MNAR** - not missing at random - braki danych są nielosowe, gdy prawdopodobieństwo braku dla  $Y_i$  zależy od wartości  $Y_i$ . Np. osoby o skrajnych wartościach dochodu nie są skłonne do ich ujawniania. Wymaga analizy wrażliwości. Jeden ze scenariuszy mógłby stanowić, że w przykładzie dochód może być po prostu równy 0.

Relacja pomiędzy mechanizmami: **MCAR ⊂ MAR ⊂ MNAR**

### **Ocena losowości mechanizmu powstawania braków danych:**

- **Metoda I** - podział zbioru danych zgodnie z wektorem indykatorem braków danych na część znaną dla  $Y_i$  i część brakującą. Następnie testowane różnice w średnich między tymi dwiema grupami.
- **Metoda II** - ocena zależności pomiędzy wektorem indykatorem braków danych oraz wybranymi zmiennymi skokowymi, za pomocą miar asocjacji.

Brak statystycznie istotnych różnic wskazuje na mechanizm MCAR. Istotne różnice wskazują na MAR lub MNAR. Nie da się jednak empirycznie stwierdzić który.

## **IMPUTACJA**

### **Metody analiz niekompletnych zbiorów danych:**

- Usunięcie obserwacji z brakami danych
- Wykorzystujące wagę
- Imputacja danych
- Oparte na modelach

### **Metody tradycyjne imputacji:**

- **Analiza zbiorów kompletnych** - pominięcie wszystkich obserwacji z brakami danych
  - Wykorzystywane tylko gdy braków jest mało, a braki występują dla niewielu zmiennych.
  - Metody proste w użyciu
  - Przy założeniu MCAR dostarczają nieobciążonych oszacowań szukanych parametrów
  - Może prowadzić do utraty dużej liczby obserwacji
  - Obciążenie estymatorów dla MNAR i MAR.
  - Nieefektywne dla wnioskowania dla podpopulacji
- **Usuwanie wierszy parami** - available-case- ograniczenie zbioru danych do wartości zaobserwowanej dla danego etapu analizy. Do oszacowania średnich wykorzystywane wszystkie obserwacje bez braków, jeśli jest to możliwe.
  - Łatwe do zastosowania
  - Wykorzystanie maksimum dostępnych informacji
  - Obciążenie oszacowań dla MNAR i MAR
  - niewłaściwe oceny błędów standardowych - zmienna liczebność próby
  - Nie zawsze możliwa do zastosowania.
- **wykorzystanie sztucznych zmiennych**
- **imputacja pojedyncza:**
  - **średnia, mediana**
  - **regresyjna** - opracowanie modelu regresji na podstawie dostępnych, pełnych obserwacji w celu przewidywania dla zmiennej z brakami. Prowadzi do zaniesienia wariancji średniej badanej cechy. Stosowane przy monotonicznym wzorcu braków danych.
  - **stochastyczna** (regresja + reszta losowa) - to samo co regresyjna, ale dodany składnik losowy dany rozkładem reszt w modelu regresji. Nie uwzględnia niepewności związanej z imputacją.
  - **hot deck** - substytucja w ramach zbioru danych na zasadzie podobieństw między obiektami. Wykorzystywana metoda najbliższego sąsiedztwa. Imputuje się wartości pochodzące z obiektów najbardziej podobnych do obiektu z brakami danych.
  - **cold deck** - imputacja ze źródeł zewnętrznych
- **Plusy i minusy imputacji pojedynczej:**
  - Wykorzystanie maksimum informacji
  - Zmienna postać rozkładu w próbie - błędne wnioskowanie
  - Brak oceny niepewności związanej z imputacją - wartości imputowane traktowane jako obserwowane

- wariancje ocen szukanych parametrów są obciążone systematycznym błędem
- Imputowane wartości mogą wykrocza poza zakres zmienności badanej cechy

Metody pojedynczej imputacji nie uwzględniają niepewności związanej z imputacją. Stąd metody imputacji pojedynczej znajdują zastosowanie wyłącznie w ograniczonym zakresie - systematyczne niedoszacowanie błędów standardowych obciąża oceny przedziałowe.

Przejście od korekty ocen punktowych do korekty ocen przedziałowych wymaga zastosowania metod, które będą uwzględniać niepewność związaną z imputacją. **Metody uwzględniające niepewność związaną z estymacją to:**

- Metody bazujące na wielokrotnym losowaniu podprób z danej próby (resampling)
- Imputacja wielokrotna - na podstawie modelu uwzględniającego losowość.

#### **Metody współczesne imputacji:**

- bazujące na rozkładach prawdopodobieństw
- imputacja wielokrotna (omówiona w 85).
- metody wykorzystujące wagę (kalibracja)

#### **85. Imputacja wielokrotna: opis metody, wybór modelu do imputacji oraz estymacja parametrów.**

Metoda polegająca na wielokrotnym uzupełnianiu danych, aktualnie jedna z najsilniej rozwijających się gałęzi nauk o imputacji. Pozwala na otrzymanie estymatorów o dobrych właściwościach i znajduje ona zastosowanie dla złożonych wzorców braków danych.

Podstawowy problem to dobór modelu pozwalającego na imputację. Uzależniony jest on od typu danych.

#### **Imputacja wielokrotna przebiega w oparciu o proces zawierający dwa źródła losowości:**

- losowy zestaw parametrów  $\theta$ ,
- losowe wartości przy zadanym zestawie parametrów.

Zestaw parametrów losowany jest z rozkładów *a posteriori* przy danych wartościach znanych  $\mathbf{Y}$  oraz przy uwzględnieniu wiedzy *a priori* szukanych parametrów. Rozkłady *a posteriori* wyznaczane są w oparciu o metodę bayesowską.

Każdy zestaw parametrów oddaje jeden z możliwych wyników estymacji, co pozwala na ocenę tego, w jakim zakresie braki wpływają na oceny parametrów.

Uzupełnianie danych wartościami losowymi pozwala lepiej przybliżyć zróżnicowanie, co jest istotne dla obliczenia błędów standardowych oszacowań.

Punkt wyjścia do wyznaczenia rozkładów *a posteriori* jest określenie łącznego rozkładu prawdopodobieństwa. Do tego celu możemy wykorzystać funkcję wiarygodności.

Przyjmując, że funkcja prawdopodobieństwa (zm skokowa) lub funkcja gęstości (zm ciągła) zależy od pewnych parametrów, prawdopodobieństwo realizacji określonej próby możemy przedstawić jako funkcję obserwacji oraz parametrów opisujących odpowiednią funkcję rozkładu. Funkcję tą nazywa się funkcją wiarygodności.

Rozkład *a posteriori* szukanego parametru wyznaczamy zgodnie z regułą wynikającą z twierdzenia Bayesa.

W celu wyznaczenia rozkładu *a posteriori* należy ustalić rozkłady *a priori*. Założmy, że parametr  $\theta$  może przyjąć dowolną wartość rzeczywistą z takim samym

prawdopodobieństwem. W istocie stosujemy jednostajny rozkład a priori. Funkcja gęstości  $f(\theta) = c$ , gdzie  $c$  jest wartością stałą większą od zera.

#### **Przykład - regresja liniowa:**

Uzupełnianie na podstawie regresji liniowej postaci:  $y=XB+\text{reszta}$

W pierwszym kroku, na podstawie znanych wartości z próby wyznaczane są rozkłady a posteriori parametrów modelu. Z wyznaczonych rozkładów losowane są następnie współczynniki regresji oraz wariancja reszt. Na podstawie wylosowanych wartości parametrów uzupełniane są brakujące dane. Szczegółowy schemat postępowania jest następujący.

Estymowane są parametry Beta i  $\sigma$  modelu, opierając się na próbie  $r$  obserwacji. Mając dane oszacowane  $\sigma^2$ , losujemy wartość wariancji reszt z rozkładu. Mając dane  $\beta^+$  i wartość losowa  $\sigma^2$  losowany jest wektor parametrów  $\beta^-$ . Mając dane nt.  $\beta^-$  i  $\sigma$  uzupełniane są braki danych przez losowanie n-r wartości z rozkładu. Losowanie powtarzane jest m-krotnie w wyniku czego otrzymywane jest m uzupełnionych zbiorów danych.

#### **Procedura analiz z wykorzystaniem imputacji wielokrotnej jest następująca:**

- Utwórz m kompletnych zbiorów danych stosując wybraną metodę imputacji. Metoda powinna uwzględniać odchylenia losowe.
- Przeprowadź standardową procedurę dla każdego z otrzymanych zbiorów danych. W wyniku uzyskujemy m oszacowań szukanego parametru,
- Ocenę punktową i przedziałową MI otrzymujemy uśredniając wartości otrzymane dla przeprowadzonych imputacji.

#### **Istnieje kilka metod stosowanych w imputacji wielokrotnej:**

- **MONOTONE** - założenie o monotonicznych brakach
  - Wykorzystanie modelu regresji dla braków w wartościach ciągłych
  - Wykorzystanie modelu logistycznego dla braków w wartościach dyskretnych i porządkowych
- **FCS** - conditional specification methods - imputacja przy założeniu istnienia łącznego rozkładu badanych cech. Opiera się o zastosowanie odrębnych modeli imputacji dla każdej ze zmiennych. Wartości imputowane są stosowane do predykcji wartości brakujących na innych zmiennych. Możliwość wykorzystania modelu liniowego (ciągła), logistycznego (skokowa) i regresji Poissona (zmienna o określonej liczebności).
- **MCMC** - imputacja dla ogólnego wzorca braków danych za pomocą algorytmu Monte Carlo wykorzystującego Łąniczki Markowa przy założeniu wielowymiarowego rozkładu normalnego badanych cech. Wymaga określenia łącznego rozkładu wszystkich zmiennych. Wartości imputowane nie są stosowane do predykcji wartości brakujących na innych zmiennych. Nie wymaga specyfikacji modelu.

#### **86. Metody i modele analizy danych wzdużnych: opis i zastosowania w analityce biznesowej.**

Obserwacja wzdłużna

Modele z efektami stałymi i efektami losowymi

Analiza szeregów czasowych

Metody i modele dla danych wzdużnych ( $t \geq 2$ ) pozwalają, oceniać efekty względem czasu zarówno na podstawie danych eksperymentalnych, jak i obserwacyjnych. Porównanie odbywa się w oparciu o różnice obserwowane w ramach jednostek, które następnie uśredniamy.

Cechą charakterystyczną badań wzdużnych jest to, że pozwalają na studiowanie zmian w czasie. Podstawą do oceny zmian jest pomiar odniesiony do tej samej jednostki badania powtarzany kilkakrotnie w określonych odstępach czasu.

W najprostszym schemacie badania wzdużnego jednostki poddane są obserwacji w dwóch momentach. Próby tego typu określamy jako zależne. Analiza odbywa się w oparciu o statystykę będącą średnią zmianą obserwowaną w próbie.

Wyróżniamy dwa podstawowe typy modeli:

- $\Delta$  α stale - modele z efektami stałymi
- $\Delta$  α losowe - modele z efektami losowymi

Modele zawierające efekty stałe i efekty losowe określamy jako modele efektami mieszanymi.

Badania wzdużne pozwalają ocenić dynamikę zjawisk. W analizach biznesowych dane wzdużne wykorzystuje się w celu:

- Oceny relacji z klientami - jak zmienia się zakres i sposób wykorzystania usług świadczonych klientom
- Oceny poziomu parametrów w czasie przed i po podaniu leku
- Oceny wielkości sprzedaży
- Oceny zachowania klientów, którzy zrezygnowali z usług, w celu wychwycenia wskaźników, które mogą sygnalizować przyszłe zerwanie umowy
- Oceny poziomu strat w procesie produkcyjnym

**Jednowymiarową i wielowymiarową analizę wariancji**, ANOVA, MANO-VA (Univariate and Multivariate Analysis of Variance) do analizy danych wzdużnych. Metody te są dobrze opisane w literaturze i stosunkowo łatwe w użyciu. Stosowanie obu modeli wymaga, aby dane do analizy zwierały informacje o po-miarkach w określonych odstępach czasu, zakładają błędę o rozkładzie normalnym i ich homogeniczność na przestrzeni różnych grup.

**Modele regresji** – to olbrzymia gama narzędzi statystycznych stosowanych w analizie danych wzdużnych. W tym obszarze analizy obserwuje się bardzo szybki rozwój zarówno w zakresie doskonalenia klasy modeli, jak i metod estymacji i co za tym idzie aplikacji.

**Uogólnione równania estymujące** (Generalized Estimating Equations – GEE) jest to kolejne podejście z zakresu analizy regresji wykorzystywane do wnioskowania na podstawie danych longitudinalnych. Są one rozwinięciem uogólnionego modelu liniowego (GLM) wykorzystywanym do analizy longitudinalnej przy pomocy estymacji opartej na funkcji quasi-wiarygodności.

Metody i modele dla danych wzdużnych ( $t \geq 2$ ) pozwalają oceniać efekty pod względem czasu zarówno na podstawie danych eksperymentalnych, jak i obserwacyjnych. Porównanie odbywa się na podstawie uśrednionych różnic obserwowanych w ramach jednostek. Modele tego typu uwzględniają heterogeniczność badanych jednostek, która może wynikać z wielu czynników o charakterze osobniczym, środowiskowym, społecznym i behawioralnym.

W najprostszym wariancie badania wzdużnego jednostki poddane są obserwacji w dwóch momentach. Próby tego typu określane są jako zależne. Analiza odbywa się w oparciu o średnią zmianą obserwowaną w próbie.

### Znaczenie danych i analiz wzdużnych dla biznesu

Badanie wzdużne pozwala ocenić dynamikę zjawisk. W analizach biznesowych dane wzdużne wykorzystuje się do celu:

- oceny relacji z klientami - jak zmienia się zakres i sposób wykorzystania usług świadczonych klientom
- oceny poziomu parametrów w czasie przed i po podaniu leku
- oceny wielkości sprzedaży, szczególnie istotne dla nowych produktów
- oceny zachowania klientów, którzy zrezygnowali z usług, w celu wychwycenia wskaźników, które mogą sygnalizować przyszłe zerwanie umowy
- oceny poziomu strat w procesie produkcyjnym - optymalizacja procesowa

**Z punktu widzenia analiz biznesowych zagadnienia podejmowane w kontekście danych wzdużnych i szeregów czasowych to:**

- wizualizacja danych, wyodrębnienie trendu poprzez wygładzenie szeregu czasowego, a tym samym oczyszczenie go z wahań okresowych i nieregularnych;
- wykrywanie obserwacji odstających i zmian strukturalnych w badanym procesie. Wiedza o zmianach mających charakter skokowy może służyć do poprawy predykcji, co będzie odbywało się poprzez wprowadzenie do modelu predykcyjnego zmiennej indywidualowej identyfikującej przedziały, dla których możemy oczekwać wystąpienia takich skokowych zmian;
- analiza zmiany wariancji w czasie, m.in. badanie wariancji cen w czasie może służyć do oceny uwarunkowań rynkowych.

#### **Test istotności średniej dla prób zależnych**

Badaniu poddano pewną cechę przed i po wystawieniu na działanie czynnika. W wyniku tego otrzymano n par obserwacji. Średnią z różnic oznaczono przez  $\mu_d$ . Jako hipotezę 0 przyjęto, że jest ona równa 0. Testowanie odbywa się poprzez wyznaczenie różnic między efektami dla jednostki dla t=1 i t=2. Następnie oblicza się średnią oraz odchylenie dla tych wartości. Następnie obliczana jest statystyka testująca t=(średnia/odchylenie)\*sqrt(n). Jeżeli H0 prawdziwa to t ma rozkład t-studenta z n-1 df.

Jeżeli test wykaże istotność różnicy między pomiarami, wiemy, że wykryte różnice nie są wynikiem oddziaływania pewnych stałych cech jednostek uczestniczących w badaniu (płeć, rasa, grupa etniczna).

Dane wzdużne można przybliżać regresją liniową z efektami stałymi. Nie przynosi to jednak dobrych efektów. Istnieje możliwość przybliżenia przy pomocy modeli z efektami losowymi.

#### **Typy modeli**

Załóżmy model:  $y_i = X_i \beta + \epsilon_i$ ,

gdzie (tylko przedstawienie niejasnych parametrów):

- $\beta$  - dany rozkładem  $N(0, V)$ , gdzie  $V$  jest macierzą wariancji-kowariancji opisującą zależności pomiędzy pomiarami w czasie. W modelowaniu należy określić jej postać (compound symmetry, proces autoregresyjny)

Podany model można rozszerzyć tak, by uwzględnił przebieg zjawiska właściwy dla

konkretniej jednostki:  $y_{ij} = \tilde{\beta}_{i0} + \tilde{\beta}_{i1}x_{ij} + \epsilon_{ij}$ , gdzie:

$y_{ij}$  -  $j$ -ta wartość cechy objaśnianej ( $j = 1, 2, \dots, T$ ) dla  $i$ -tej jednostki ( $i = 1, 2, \dots, n$ ),

$\tilde{\beta}_{i0}$  - wyraz wolny dla  $i$ -tej jednostki,

$\tilde{\beta}_{i1}$  - współczynnik regresji dla  $i$ -tej jednostki, oraz

$\epsilon_i \sim N(0, \sigma^2)$ .

Jednostki są wylosowane z populacji. Podobnie parametry strukturalne właściwe dla jednostki są próbą z populacji współczynników regresji:

$\beta_i \sim N(\beta, D)$

Model ten można przedstawić także jako:

$$y_{ij} = (\beta_0 + b_{i0}) + (\beta_1 + b_{i1})x_{ij} + \epsilon_{ij}$$

gdzie:

$\beta_0$  i  $\beta_1$  określamy jako **efekty stałe**, natomiast

$b_{i0}$  i  $b_{i1}$  określamy jako **efekty losowe**.

**Interpretacja parametrów:**

- $\beta_1$  określa zmianę średniej  $y$ , związaną z jednostkowym wzrostem  $x_i$
- $b_{ii}$  interpretujemy jako ocenę odchylenia parametrów strukturalnych od populacji dla  $i$ -tej jednostki.

Należy także wspomnieć o imputacji koniecznej w takich badaniach, gdyż metody te są stosowane często w badaniach klinicznych o ograniczonej liczebności, a w trakcie badania, pewne jednostki mogą wypadnąć z badania.

## Znaczenie macierzy wariancji-kowariancji w modelowaniu danych wzdużnych

- ▶ wariancja pomiarów nie jest stała w czasie: *elementy na diagonali macierzy wariancji-kowariancji*.
- ▶ zależność pomiędzy kolejnymi obserwacjami w ramach jednostki: *elementy poza diagonałą macierzy wariancji-kowariancji*.
- ▶ struktura macierzy wariancji-kowariancji pozwala oddać charakter zależności pomiędzy kolejnymi obserwacjami  $Y_i$ , np. większe odstępy pomiędzy pomiarami będą charakteryzować się mniejszą zależnością.

## Wyglądzanie szeregów czasowych

Wykorzystanie średniej ruchomej do wygładzania szeregu czas

## Charakter i źródła korelacji w danych wzdużnych

Zależności obserwowane na zmiennej zależnej w danych wzdużnych możemy scharakteryzować następująco:

- ▶ obserwacje są dodatnio skorelowane
- ▶ siła korelacji często maleje wraz z zwiększeniem odstępu pomiędzy obserwacjami
- ▶ korelacje rzadko zbiegają do zera (całkowicie "wygasają") nawet dla dużych odstępów pomiędzy obserwacjami
- ▶ korelacja pomiędzy bliskimi sobie obserwacjami rzadko osiąga poziom bliski jedności

Źródła korelacji:

- ▶ zróżnicowanie pomiędzy jednostkami (np. różna siła odpowiedzi na lek) (efekty losowe w modelach - losowe wyrazy wolne i współczynniki regresji)
- ▶ zróżnicowanie w ramach obserwacji dla jednostki (np. zmiany wynikające z efektów sezonowych, wpływ czynników zewnętrznych)
- ▶ błąd pomiaru (zakłócanie efektu)

owego. Pozwala to na oczyszczenie szeregu z wahań okresowych. Możliwe także wygładzanie wykładnicze.

Pozwala na wykrywanie zmian przebiegu zjawiska w czasie:

- zmiana strukturalna - znaczny wzrost lub spadek tempa rozwoju badanego zjawiska.
- Zmiana skokowa - przesunięcie średniego poziomu
- Obserwacje odstające - nieregularne odstępstwa od tendencji centralnej.

### Interpretacja funkcjami sklejonymi (regresja adaptacyjna)

Modele regresji adaptacyjnej opierają się na interpolacji liniowymi funkcjami w ramach wyodrębnionych fragmentów szeregu czasowego. Momenty, dla których obserwujemy zmianę nachylenia dopasowanej funkcji wskazują na zmianę strukturalną w szeregu.

Cechy interpolacji funkcjami sklejonymi:

- metoda nieparametryczna
- nie ma wymogu formułowania postaci modelu a priori
- węzły ilustrujące zmiany strukturalne nie są identyfikowane na podstawie testów istotności.

## Regresja adaptacyjna - zapis modelu i estymacja parametrów

Estymowany model jest postaci przy  $l$  współczynników regresji:

$$\begin{aligned}y_i = & \beta_0 + \beta_1 \max\{x_i - x_1^*, 0\} + \beta_2 \max\{0, x_1^* - x_i\} \\& + \beta_3 \max\{x_i - x_2^*, 0\} + \beta_4 \max\{0, x_2^* - x_i\} \\& + \dots \\& + \beta_{l-1} \max\{x_i - x_r^*, 0\} + \beta_l \max\{0, x_r^* - x_i\},\end{aligned}\quad (130)$$

gdzie:

$y_i$  - wartość zmiennej celu dla  $i$ -tej jednostki

$x_i$  - wartość zmiennej objaśniającej dla  $i$ -tej jednostki

$\beta_0$  - wyraz wolny

$\beta_1, \beta_2, \dots, \beta_l$  - współczynniki regresji

$x_1^*, x_2^*, \dots, x_r^*$  - punkty węzłowe, dla których minimalizowana jest funkcja niedopasowania LoF (Lack-of-Fit). Znajdowanie minimum funkcji niedopasowania LoF: [S08AdaptiveRegIntro01.sas](#) [[Link](#)].

## Funkcja niedopasowania w regresji adaptacyjnej

Jako funkcję niedopasowania możemy przyjąć sumę kwadratów różnic wartości empirycznych i przewidywanych:

$$RSS = \sum_{i=1}^n (y_i - \hat{y}_i)^2, \quad (131)$$

gdzie:

$y_i$  - wartość zmiennej celu dla  $i$ -tej jednostki,

$\hat{y}_i$  - wartość przewidywana  $Y$  dla  $i$ -tej jednostki.

Algorytm rozpoczyna od wyboru punktu węzłowego, dla którego minimalizowana jest funkcja niedopasowania  $RSS$  [131]. Następnie dobierane są kolejne punkty minimalizujące tę funkcję przy zadanych wartościach z poprzednich iteracji, zgodnie z przyjętą maksymalną liczbą punktów węzłowych. Ostateczna postać modelu [130] będzie obejmować współczynniki istotne z punktu widzenia dopasowania oraz predykcji  $Y$  (Opis algorytmu: [Proc AdaptiveReg](#)).

Procedura w SAS wypisuje nawet informacje o zmianach strukturalnych.

Przykłady wykorzystania:

- Obserwacja załamań, punktów istotnych dla danych dotyczących liczby lotów w danym miesiącu (w przykładzie weźle na 9/11).

**87. Porównaj modele z efektami stałymi oraz modele z efektami losowymi. Przedstaw podstawowe różnice i zastosowania obu typów modeli.**

Tu jest problem bo wykład Korczyńskiego nie mówi o tym wiele, a nawet mieszka pojęcia  $xD$  gośc przedstawia model mieszany, twierdzi, że stały efekt to co innego niż irl.

$Y = X\beta + \epsilon$  model efektów stałych,

$Y = Z\gamma + \epsilon$  model efektów losowych,

$Y = X\beta + Z\gamma + \epsilon$  model mieszany,

gdzie poszczególne składowe modeli oznaczają:

$Y$  – wektor obserwowanych wartości zmiennej zależnej,

$X$  – znana macierz układu oparta na specyfikacji modelu,

$\beta$  – wektor nieznanych parametrów efektów stałych,

$Z$  – znana macierz układu dla efektów losowych,

$\gamma$  – wektor nieznanych parametrów efektów losowych,

$\epsilon$  – wektor błędów losowych.

Dzienn za to, ja jeszcze tu postaram się trochę internetu przepisać ☺

**Źródło: WSTĘP DO EKONOMETRII DANYCH PANELOWYCH Spis treści**

Modele te wykorzystywane są do danych panelowych, gdy zwykła regresja liniowa nie jest w stanie przynieść dobrego efektu.

**Model z efektami stałymi** używany jest, gdy zakłada się, że jednostki różnią się, a różnice między jednostkami są stałe w czasie.

Wówczas można wyróżnić:

$$y_{it} = X_{it}\beta' + \alpha_i + \epsilon_{it}$$

gdzie:

$\alpha_i$  - jest stałym w czasie efektem indywidualnym dla obserwacji  $i$ .

O parametrze  $a$ , można myśleć jak o indywidualnym dla każdej jednostki wyrazie wolnym w modelu. Oszacowanie zawierać będzie wpływ wszystkich charakterystyk niezawartych w wektorze zmiennych obserwacyjnych  $X$ . Można taki model szacować przy pomocy KMNK.

**Gruszczyński, Mikroekonometria:** Każdy efekt indywidualny obejmuje wszystkie stałe w czasie charakterystyki danej jednostki, które mają wpływ na zmiennej objaśnianą, a jednocześnie nie zostały uwzględnione explicite w wektorze  $x_{ib}$ , najczęściej dlatego, że mają charakter niekwantyfikowalny lub trudny do zmierzenia. W ten sposób (tu z przykładu) można uwzględnić w modelu efekt zdolności wybranego menadżera, który istnieje, jednak

nie jest znany. Należy zauważyć, że o ile istnieją inne, stałe w czasie zmienne, nieuwzględnione w wektorze  $X_{it}$  charakterystyki firm mające wpływ na zmienną objaśnianą, to one również będą ujęte w tym samym  $a_i$ , a ich odseparowanie od domniemanego efektu umiejętności menedżera nie jest możliwe.

Rozkład składnika losowego w tym modelu mają zerową wartość oczekiwana i stałą wariancję. Ponadto rozkłady te są niezależne, więc kowariancja takowych jest równa 0, gdy nie mówimy o tych samych jednostkach oraz/lub tych samych okresach. W tym podejściu, o efektach indywidualnych można myśleć jak o różnych wyrazach wolnych w szeregach czasowych dla poszczególnych jednostek: mają charakter stałego w czasie, deterministycznego parametru, który różni się dla poszczególnych  $i$ . Alternatywnie więc można równanie tego modelu zapisać jako:

$$y_{it} = \mathbf{x}'_{it}\beta + \sum_{j=1}^N \alpha_j d_{ij} + \varepsilon_{it}, \quad i = 1, \dots, N, \quad t = 1, \dots, T,$$

gdzie  $d_{ij}$  są zmiennymi binarnymi takimi, że  $d_{ii}=1$  dla  $i=j$ , zaś 0 w pozostałych przypadkach. Tutaj też pojawia się uszczegółowienie estymacji przy pomocy KMNK, zwanej LSDV - least squares dummy variables, pochodzącej od wykorzystania sztucznych zmiennych binarnych. Przedstawiona postać modelu ukazuje, że niemożliwe jest uwzględnienie wśród zmiennych objaśniających zmiennych stałych w czasie, w szczególności zaś w modelu nie ma wyrazu wolnego. Należy zauważyć, że wprowadzenie do zbioru jakiejkolwiek zmiennej, która byłaby stała w czasie dla każdej z jednostek (jako przykład można potraktować wyraz wolny), prowadziłoby do współliniowości ze zmiennymi  $d_{ij}$  i uniemożliwiłoby identyfikację modelu. Istnieje także opcja estymacji przy użyciu estymatora wewnętrzgrupowego, który jest optymalniejszy niż LSDV.

**Przykład użycia** - oszacowanie wpływu jakości intensywności nawożenia ziemi na wielkość plonów uzyskiwanych przez rolnika. W tym celu należy oszacować model, w którym  $y_{it}$  określa wielkość plonów zebranych z hektara ziemi przez  $i$ -tego rolnika w roku  $t$ , w wektorze  $x_{it}$  znajduje się zmienna określająca ilość zużytych nawozów ( $i$ , potencjalnie, inne kwantyfikowane czynniki wpływające na wielkość plonów), składnik losowy zawiera wpływ czynników losowych, jak na przykład warunków atmosferycznych, ewentualnych katastrof naturalnych. Ponadto jednak znaczącym czynnikiem wpływającym na  $y_{it}$  jest jakość gleby, jaką dysponuje dany rolnik. Jakość ta z reguły nie jest dobrze skwantyfikowana, ponadto nie ulega większym zmianom w czasie. Tym samym czynnik jakości gleby może zostać ujęty w efekcie indywidualnym. Zauważmy, jak potencjalnie dużym błędem byłoby ograniczenie analizy poprzez wykorzystanie danych przekrojowych: nie możliwość uwzględnienia efektów indywidualnych, a jednocześnie brak prawidłowej kwantyfikacji jakości gleby uniemożliwiąby uwzględnienie tego czynnika. Można domieścić, iż rolnik mający glebę słabej jakości jest zmuszony nawozić ją bardziej intensywnie, a i tak uzyskuje przy tym mniejsze plony od rolnika, który dysponuje glebą wyższej jakości, za to mniej nawożoną. Interpretacyjnie więc, intensywność nawożenia i wielkość plonów z jednostki powierzchni będą ze sobą skorelowane ujemnie, co często prowadzi do otrzymania ujemnej oceny parametru przy odpowiedniej składowej  $x_{it}$  i kuriozalnego wniosku o negatywnym wpływie ilości wykorzystywanych nawozów na wielkość plonów.

### Model z efektami losowymi

Tu każdej jednostce przypisywana jest pewna zmienna losowa, której realizacja odpowiada za efekt indywidualny w danym okresie. W modelu z efektami losowymi efekty indywidualne nie są jednakowe w kolejnych okresach. W rezultacie nie traktujemy efektów indywidualnych jak parametrów i nie szacujemy ich wartości. O ile w modelu z efektami stałymi efekty indywidualne mogliśmy interpretować jako indywidualny wyraz wolny, inny dla każdej jednostki, ale stały w czasie, to w modelu z efektem losowym efekty indywidualne możemy interpretować jako indywidualne składniki losowe. Do modelu można wprowadzić wyraz wolny.

$$y_{it} = \gamma + X_{it}\beta' + \alpha_i + \varepsilon_{it} \quad (8)$$

Do modelu (8) wprowadzamy łączny składnik losowy  $v_i = \alpha_i + \varepsilon_{it}$ , czyli zmienną losową stanowiącą sumę indywidualnego składnika losowego  $\alpha_i$  i białego szumu  $\varepsilon_{it}$ :

$$y_{it} = \gamma + X_{it}\beta' + v_i \quad (9)$$

gdzie pierwszy wyraz po prawej stronie równania oznacza wyraz wolny. Właśnie obecność tego wyrazu, nieobecnego w typowym modelu z efektami stałymi stanowi dużą różnicę. Wynika to z charakteru efektów w modelu z efektami losowymi. Tu są one traktowane jako losowe efekty. Nie stanowią one dodatkowych, potencjalnie szacowanych parametrów, lecz powiększają część stochastyczną modelu.

Pełny zapis modelu prezentuje się następująco:

$$\begin{aligned} y_{it} &= \mu + x'_{it}\beta + v_{it}, \quad i = 1, \dots, N, \quad t = 1, \dots, T. \\ v_{it} &= \alpha_i + \varepsilon_{it} \end{aligned}$$

Zmienną losową  $v_{it}$  określamy mianem łącznego składnika losowego - sumy efektów indywidualnych oraz białego szumu, którym jest składnik losowy epsilon. Nie jest tu problemem użycie zmiennych stałych w czasie dla każdej z jednostek. Nie dochodzi do ściślej współliniowości ze zmiennymi binarnymi, co uniemożliwiło ich uwzględnienie w modelu typu stałego.

Model ten wymaga przyjęcia dodatkowych założeń ze względu na przeniesienie efektów indywidualnych do części stochastycznej.

Zakłada się także niezależność zmiennych objaśniających i efektów indywidualnych. Do estymacji - nie można KMNK bo estymatory nieefektywne. Należy użyć uogólnionej metody najmniejszych kwadratów.

Wskazówki ogólne doboru metody (Źródło: [Dynamiczne modele panelowe w badaniach ekonomicznych](#)):

- Zależność od liczby okresów i liczbą obiektów
  - jeśli dużo okresów, a obiektów mało to można wprowadzić dodatkowe zmienne sztuczne odpowiadające poszczególnym obiektom. Wówczas można skorzystać z Fixed Effects.
  - Jeżeli bardzo mało okresów, a bardzo dużo obserwacji to bez sensu jest dodawanie zmiennych sztucznych, dodatkowo możliwa utrata zgodności estymatora. Wówczas wykorzystać Random Effects.
- Zależność od natury analizowanych obiektów:

- jeśli bada się obiekty tego samego rodzaju (państwa, bardzo duże firmy, gałęzie przemysłu) i istotne jest oszacowanie efektów grupowych dla tych konkretnych obiektów to właściwszy jest model FE.
- Jeśli rozważane są obiekty losowo wybrane z pewnej populacji, oszacowanie konkretnych efektów grupowych dla tych obiektów jest mniej istotne, gdyż badanie zazwyczaj koncentruje się na charakterystykach całej populacji. Wówczas RE.
- Jeśli efekty grupowe są skorelowane ze zmiennymi objaśniającymi to także składnik losowy, którego efekt  $\alpha_i$  jest składową, jest skorelowany ze zmienną objaśniającą. Wówczas estymator uogólnionej metody najmniejszych kwadratów traci zgodność, co kieruje analityka do użycia estymatora wewnętrzgrupowego. Do sprawdzenia czy ta korelacja występuje korzysta się z **testu Hausmana**.

Omawiany test jest najpopularniejszym testem statystycznym, stosowanym do rozstrzygania o tym, który z estymatorów:  $\hat{\beta}_{UMNK}$  czy  $\hat{\beta}_W$  posiada własności dobrego estymatora, a więc do dokonania wyboru pomiędzy modelem RE a modelem FE. Test został zaproponowany przez J. A. Hausmana [1978].

Idea konstrukcji tego testu opiera się na porównaniu własności dwóch estymatorów, z których jeden jest zgodny, niezależnie od tego, która z hipotez jest prawdziwa, a drugi – tylko jeśli prawdziwa jest  $H_0$ . Hipoteza zerowa zakłada, że efekty grupowe  $\alpha_i$  są nieskorelowane ze zmiennymi objaśniającymi:

$$H_0: E(u_{it} | \mathbf{x}_{it}) = \mathbf{0},$$

wobec

$$H_1: E(u_{it} | \mathbf{x}_{it}) \neq \mathbf{0}.$$

Przy założeniu prawdziwości  $H_0$ , oba estymatory:  $\hat{\beta}_W$  i  $\hat{\beta}_{UMNK}$  są zgodne, ale  $\hat{\beta}_W$  jest nieefektywny. Jeśli  $H_0$  jest nieprawdziwa, to  $\hat{\beta}_W$  jest zgodny, gdyż przekształcenie wewnętrzgrupowe  $\mathbf{Q}$  usuwa efekty grupowe, będące przyczyną korelacji; estymator  $\hat{\beta}_{UMNK}$  jest natomiast niezgodny. Innymi słowy, hipoteza zerowa gosi, że poprawną specyfikacją jest model RE, a hipoteza alternatywna – że poprawna jest specyfikacja modelu FE.

## 88. Regresja kwantylowa: opis i zastosowania w analityce biznesowej.

Regresja kwantylowa jest metodą estymacji zależności całego rozkładu zmiennej objaśnianej od zmiennych objaśniających. Regresja kwantylowa modeluje relację między zbiorem predyktorów (zmiennych niezależnych) i konkretnych centylami (lub kwantylami) zmiennej przewidywanej (zależnej), najczęściej medianę.

W regresji kwantylowej chcemy zobaczyć, jak różne kwantyle zmiennej objaśnianej zależą od wybranych zmiennych objaśniających.

Estymacja poszczególnych kwantylów pozwala na pełniejszy opis sytuacji zarówno w punkcie centralnym, jak i „ogonach” rozkładu, co jest szczególnie przydatne, gdy warunkowa dystrybuanta jest różnorodna i nie ma „standardowego” kształtu. Estymacja regresji na kwantylach jest semiparametryczna, a więc nie przyjmuje się założenia o typie rozkładu dla losowego wektora reszt w modelu.

Regresji kwantowej znalazło zastosowanie w oszacowaniu wartości zagrożonej (VaR).

Instytucje finansowe i ich organy regulacyjne stosują VaR jako standardową miarę ryzyka rynkowego. Miara ta ma prostą konstrukcję i może być stosowana dla szerokiego spektrum inwestycji finansowych. Wartość VaR mierzy ryzyko rynkowe na podstawie tego, ile portfel może stracić w danym okresie czasu, przy określonym poziomie ufności. Wartość zagrożona w czasie  $t$  (oznaczona przez VaR $t$ ) jest warunkowym kwantylem przyszłych wartości portfela.

Regresja kwantylowa pozwala szacować wpływ określonych cech na kwantyle rozkładu zmiennej losowej np. kwartyle, kwintyle, medianę. Jest to model zbliżony do modelu regresji liniowej z tym tylko, że regresja kwantylowa dostarcza zbioru oszacowań, podczas gdy regresja liniowa pozwala wyłącznie na ocenę tendencji centralnej.

Specyfikacja interpretacja modelu RK jest podobna do tej, którą znamy z modelu regresji liniowej. Jednakże w przeciwieństwie do regresji liniowej, w modelu regresji kwantylowej otrzymujemy rodzinę linii, a w ten sposób możemy skupić się na wybranym fragmencie rozkładu badanej zmiennej. Jako główne zalety modelu regresji kwantylowej wymienić możemy:

- odporność na obserwacje odstające
- wgląd w wybrane fragmenty warunkowego rozkładu zmiennej zależnej

## Zadanie optymalizacyjne w metodzie najmniejszych kwadratów

- ▶ Średnia jest parametrem, który minimalizuje sumę kwadratów odchyлеń wartości empirycznych od parametru:

$$\mu = \min_{\mu \in R} \sum_{i=1}^n (y_i - \mu)^2 \quad (119)$$

zgodnie z czym otrzymujemy zapis w postaci rozkładu warunkowego średniej  $\mu(x) = \mathbf{x}_i^T \boldsymbol{\beta}$  w MNK:

$$\boldsymbol{\beta} = \min_{\boldsymbol{\beta} \in R^p} \sum_{i=1}^n (y_i - \mathbf{x}_i^T \boldsymbol{\beta})^2 \quad (120)$$

## Zadanie optymalizacyjne w regresji kwantylowej

- ▶ Mediana jest parametrem, który minimalizuje sumę bezwzględnych odchyłeń wartości empirycznych od tego parametru:

$$Me = \min_{\zeta \in R} \sum_{i=1}^n |y_i - \zeta| \quad (121)$$

- ▶ Dla kwantyla  $\tau$  możemy zapisać:

$$\theta(\tau) = \min_{\zeta \in R} \sum_{i=1}^n \rho_\tau(y_i - \zeta) \quad (122)$$

gdzie:

$$\rho_\tau(y_i - \zeta) = (\tau I[y_i > \zeta] + (1 - \tau)I[y_i \leq \zeta]) |y_i - \zeta|$$

- ▶ Dla  $Q(\tau|x) = \mathbf{x}_i^T \boldsymbol{\beta}(\tau)$  zadanie optymalizacyjne polega na znalezieniu takiego wektora  $\boldsymbol{\beta}$ , który będzie minimalizować następującą sumę:

$$\boldsymbol{\beta}(\tau) = \min_{\boldsymbol{\beta} \in R^p} \sum_{i=1}^n \rho_\tau(y_i - \mathbf{x}_i^T \boldsymbol{\beta}) \quad (123)$$

## Regresja kwantylowa w ocenie błędu prognozy

Zasadniczym elementem oceny modelu predykcyjnego jest jego zdolność do trafnego prognozowania zmiennej celu. korzystając z regresji kwantylowej możemy dokonać takiej oceny, a w szczególności odpowiedzieć na pytania:

- jakie zmienne mają istotny wpływ na poprawę najtrajniejszych prognoz (kwantyla pierwszego)?
- jakie czynniki wpływają na prognozy, które obarczone są największym błędem. Czy możemy zidentyfikować wyniki które istotnie zmniejszą błąd prognozy dla tych jednostek, dla których prognozy nie były do tej pory trafne?

Przykłady zastosowań:

- Analiza błędu prognozy popytu - znalezienie błędów w prognozach innych modeli, wskazanie na czynniki wpływające na błędy w danych kwartylach błędów.
- Oszacowanie wagi urodzeniowej

Regresja kwantylowa jest odporna na nieregularność rozkładu zmiennej - wynika to z właściwości statystyk pozycyjnych.

## 89. Regresja adaptacyjna: model, opis estymacji i zastosowania w analityce biznesowej. + Opisane wyżej

Metoda MARSplines (ang. Multivariate Adaptive Regression Splines) należy do grupy metod wykorzystujących uczenie nieukierunkowane. W przypadku takiego typu uczenia poszukiwany jest model najlepiej pasujący do danych bez uwzględnienia mechanizmu jaki te dane wygenerował. Poszukiwany model nie ma służyć opisowi relacji, związków przyczynowych, a tylko prognozowaniu lub rozpoznawaniu. Poza uwzględnieniem wpływu predyktorów (tak jak w klasycznym modelu regresji) analizowane są wszystkie obserwacje danej zmiennej objaśniającej i obszar jej zmienności dzielony jest na przedziały, w których ma ona różny wpływ na badane zjawisko.

Do określania granic przedziałów stosowane są tzw. węzły (ang. Knots) stanowiące wartości progowe. Dzięki zastosowaniu takiego podejścia przeprowadzane jest porównanie wartości zmiennej objaśniającej z wartością progową, czego efektem może być przyjęcie przez tę zmienną pewnej wartości wag i różnego znaku. Wartość progowa określana jest jako  $t_i$ , a do rozróżnienia wartości zmiennej objaśniającej stosowana jest funkcja bazowa (ang. Basis Function) określana jako:  $\text{Bimax}(0; X-t)$ . Liczba dopuszczalnych funkcji bazowych wynosi  $2Nk$  (gdzie:  $N$  – liczba obserwacji,  $k$  – liczba predyktorów,  $2$  – konsekwencja zastosowania znaku  $-/+$ )

Algorytm MARSplines przeszukuje przestrzeń wszystkich wartości wejściowych i predykcyjnych (położenie węzłów  $t_j$ ), jak i interakcji między zmiennymi. Do modelu dodawane są wtedy kolejne funkcje bazowe (wybierane ze zbioru wszystkich dopuszczalnych funkcji) w taki sposób, by maksymalizować ogólny poziom dopasowania (wg minimum sumy kwadratów). Wynikiem tej operacji jest znalezienie najważniejszych zmiennych niezależnych, oraz najważniejszych ich interakcji.

Cechy interpolacji funkcjami sklejonymi:

- metoda nieparametryczna - brak wymagania co do założeń nt. zależności pomiędzy zmiennymi niezależnymi i zmiennymi zależnymi (zależność liniowa, logistyczna itp.);
- nie ma wymogu formułowania postaci modelu a priori
- węzły ilustrujące zmiany strukturalne nie są identyfikowane na podstawie testów istotności

Algorytm rozpoczyna od wyboru punktu węzłowego  $i$ , dla którego minimalizowana jest funkcja niedopasowania RSS. Następnie dobierane się kolejne punkty minimalizujące tę funkcję przy zadanych wartościach z poprzednich iteracji, zgodnie z przyjętą maksymalną liczbą punktów węzłowych. Ostateczna postać modelu będzie obejmować współczynniki istotne z punktu widzenia dopasowania oraz predykcji Y

Generalnie, modele nieparametryczne dobrze dostosowują się do danych, są bardzo elastyczne, co może prowadzić do niekorzystnego zjawiska nadmiernego dopasowania (przeuczenia, overfitting). Do zwalczania problemu, w MARSplines korzysta się z techniki redukcji (pruning), analogicznej do przycinania (w drzewach klasifikacyjnych), ograniczającej złożoność modelu przez redukowanie liczby funkcji bazowych.

Algorytm może zostać użyty, do predykcji cen akcji, badań przeżywalności dla nieregularnych grup, wyboru zmiennych w modelach scoringowych. Częstym zastosowaniem jest wybór odpowiednich zmiennych a następnie zastosowanie innego modelu np. Sieci neuronowych.

## 90. Metoda k-średnich i jej zastosowanie w ocenie wartości klienta w czasie CLV.

## Co to wartość klienta w czasie?

Wartość klienta w czasie (ang. Customer Lifetime Value - CLV) to zdyskontowana suma przyszłych wpływów, które można przypisać do relacji z klientem. Pozwala oszacować zysk, jaki firma osiągnie w przyszłości dzięki klientowi. Maksymalizacja CLV jest jednym z głównych celów organizacji nastawionych na osiąganie zysku.

$$CLV = \sum_{t=1}^{\infty} \frac{E(V_t)}{(1 + d)^{t-1}}$$

$V_t$  - wpływ pieniężny netto od klienta w okresie t

d – stopa dyskontowa

## Wartość klienta w czasie

Jednym z zastosowań CLV jest wspieranie decyzji dotyczących ustalania wydatków na przyciąganie nowych klientów. Koszty akwizycji są uzasadnione, jeśli wynoszą one mniej niż CLV klienta. Celem jest identyfikacja potencjalnych klientów z wysoką wartością CLV i unikanie tych o niskiej wartości w czasie.

(np. dostawcy mediów, operatorzy telekomunikacyjni)

Wartość klienta w czasie stanowi najlepsze podstawy do wyznaczania zasobów marketingowych: organizacje powinny inwestować zasoby przeznaczone na marketing jedynie w te działania, które zwiększą CLV, tak, aby CLV było wyższe niż koszty.

3 sposoby na zwiększenie CLV obecnych klientów:

- Dłuższe ich utrzymanie
- Zwiększenie wpływów od klientów
- Zmniejszenie kosztów obsługi klientów, marketingu lub obu

Wysokość środków ponoszonych na te taktiki jest wyznaczana przez zmianę CLV, którą wywołają.

## Modelowanie CLV

Modelowanie CLV stosowane jest szczególnie w branżach, w których występuje kontraktowa relacja z klientem, tzn. kiedy można określić dokładną datę końca relacji firma-klient.

Przykłady:

- Klienci operatora komórkowego kończą obowiązującą ich umowę i przestają opłacać rachunki
- Subskrypcje treści medialnych kończą się, kiedy użytkownicy rezygnują z nich lub nie decydują się na ich odnowienie
- Członkostwo w programie zdrowotno-sportowym wygasza w określonym terminie

## Rodzaje modeli CLV

Wyróżnia się dwa typy modeli CLV:

### 1. Modele typu „gone-for-good”

- zakładają, że klienci, którzy zrezygnowali z usługi, już nie wrócą
- w tym przypadku najważniejsze jest zatrzymanie klienta przez jak najdłuższy czas
- do analizy czasu do odejścia klienta stosuje się modele przeżycia
- przykłady: prosty i ogólny model retencji

### 2. Modele typu „always-a-share”

- nie zakładają, że brak aktywności ze strony klienta oznacza jego stałe odejście
- klient, który nie dokonał zakupu w bieżącym miesiącu, może wrócić w następnym
- przykłady: model migracji i podejście data mining do wartości w czasie

Moim zdaniem: Nie mieliśmy omawianych metod, które miałyby typowo oceniać CLV przy pomocy k-means. Nie wiem jak interpretować to pytanie. Poniżej przykład od Wołowiec i trochę z neta.

### **Prezentacja segmentacji A. Wołowiec**

Klienci mają różne pragnienia, potrzeby etc. Heterogeniczność ta pozwala na osiągnięcie przewagi konkurencyjnej przez firmy poprzez jej rozpoznanie i dostosowanie się do niej. Nie wszystkie potrzeby klientów mogą zostać zaspokojone przez jedną ofertę. Jedną z odpowiedzi na to zjawisko jest segmentacja, która pozwala na odnalezienie grup klientów o podobnych cechach.

### **Zastosowanie biznesowe**

1. Segmentacja rynku - dla każdego segmentu można dobrać strategię.

- Customizacja i personalizacja w obrębie podsegmentów - dalsza customizacja wewnątrz odnalezionych segmentów, gdzie dalej występują zróżnicowane potrzeby.

Jak dokonać segmentacji? - K-średnich

#### Następstwa wspólnej dla wszystkich klastrów wariancji błędu (założenie modelu):

- rozkład punktów należących do jednego klastra jest okrągły lub sferyczny
- wszystkie klastry mają taki sam kształt i dyspersję

Jeśli zmienne nie wpadają w okrągły obszarowej równej wielkości, w metodzie K-średnich może wystąpić błędów (np. obszary mają eliptyczny kształt). Podejście oparte na modelach mieszanych dopuszcza inne kształty i zmienne wielkości klastrów. Innym sposobem na doprowadzenie klastrów do postaci bardziej sferycznej jest transformacja danych przed estymacją modelu. Nie rozwiązuje to jednak kwestii równych wielkości.

Metoda K-średnich daje lepsze wyniki dla zmiennych ciągłych. Stosuje się dwie metody, które służą wprowadzeniu do analizy K-średnich zmiennych kategoryzujących:

- partycjonowanie – poza modelem – obserwacji z wykorzystaniem ważnych zmiennych kategoryzujących i zastosowanie modelu na każdej z partycji
- utworzenie sztucznych zmiennych zerojedynkowych i zastosowanie metod ważenia zmiennych

- Inicjalizacja.** Wybierz początkowy zbiór środków klastrów  $\hat{\mu}_{jk}^0$ , gdzie  $k = 1, \dots, K$ , a oznaczenie górnego 0 wskazuje numer iteracji. Zapoczątkuj licznik pętli  $h = 1$ .
- Przypisanie do klastrów.** Przypisz obserwację  $i$  do klastra, którego średnia jest najbliższa obserwacji. Odległość Euklidesa pomiędzy punktem  $x_{ij}$  a środkiem klastra  $\hat{\mu}_k^{h-1}$  podana jest wzorem  $d_{ik}^h = \sqrt{\sum_{j=1}^p (x_{ij} - \hat{\mu}_{jk}^{h-1})^2}$ . Obserwacja  $i$  jest przypisana do klastra, który jest jej najbliższy, tzn.  $g_i^h = \operatorname{argmin}_k d_{ik}^h$ .  $\operatorname{argmin}_k$  wskazuje, że  $g_i^h$  jest równe  $k$  minimalizując wartość  $d_{ik}^h$ .
- Wylatowanie średnich dla klastrów.** Przy stałym przypisaniu do klastrów, wylatuj ich średnie. Są one nazywane również centrami lub centroidami klastrów  $\hat{\mu}_{jk}^h = \operatorname{average}\{x_{ij}: g_i^h = k\}$ . Innymi słowy, nowe oszacowanie klastrowej średniej  $k$  jest prostą średnią wszystkich obserwacji przypisanych do danego klastra ( $g_i^h = k$ ).
- Obliczenie SSE.** Jest to suma kwadratów błędów (ang. sum of squared errors, SSE) lub całkowita wariancja wewnętrzna klastrów.  $SSE = \sum_{i=1}^n \sum_{j=1}^p (x_{ij} - \hat{\mu}_{g_i^h})^2 = \sum_{i=1}^n d_{ig_i^h}^2$ . Błędy są w rzeczywistości odległościami pomiędzy każdą obserwacją a najbliższym jej środkiem klastra.
- Pętla.** Niech  $h = h + 1$  i powróć do kroku 2, aż spełnione zostanie kryterium zbieżności lub przekroczone zostanie maksymalna dopuszczalna liczba iteracji.

Algorytm dąży do minimalizacji wartości SSE, co stanowi kombinatoryczny problem optymalizacyjny i jest bardzo trudne pod względem obliczeniowym. W szczególności, nie ma pewności czy algorytm będzie zbiegał do rozwiązania optymalnego, a różne punkty startowe mogą dawać różne rozwiązania. Ze względu na te kwestie, estymacja K-średnich może być generowana wielokrotnie dla różnych punktów startowych, a z wielu uzyskanych rozwiązań wybiera się to, które zapewnia najmniejszą wartość SSE.

Algorytm dąży do minimalizacji wartości SSE, co stanowi kombinatoryczny problem optymalizacyjny i jest bardzo trudne pod względem obliczeniowym. W szczególności, nie ma pewności czy algorytm będzie zbiegał do rozwiązania optymalnego, a różne punkty startowe mogą dawać różne rozwiązania. Ze względu na te kwestie, estymacja K-średnich może być generowana wielokrotnie dla różnych punktów startowych, a z wielu uzyskanych rozwiązań wybiera się to, które zapewnia najmniejszą wartość SSE.

**Wewnętrzkalstrowe odchylenie std. dla klastra  $k$  (RMS Std Deviation) =  $\sqrt{\frac{1}{p(n_k-1)} \sum_{g_i=k} \sum_{j=1}^p (x_{ij} - \hat{\mu}_{jk})^2}$ ,**

gdzie  $\hat{\mu}_{jk}$  to końcowa średnia zmiennej  $j$  dla klastra  $k$

**Całkowite odchylenie std. dla zmiennej  $j$  (Total STD) =  $\sqrt{\frac{1}{(n-1)} \sum_{i=1}^n (x_{ij} - \hat{\mu}_j)^2}$ ,**

gdzie  $\hat{\mu}_j$  to ogólna średnia zmiennej  $j$

**Wewnętrzkalstrowe odchylenie std. dla zmiennej  $j$  (Within STD) =  $\sqrt{\frac{1}{(n-K)} \sum_{i=1}^n (x_{ij} - \hat{\mu}_{jg_i})^2}$ ,**

gdzie  $\hat{\mu}_{jg_i}$  to ogólna średnia zmiennej  $j$  dla klastra  $k$ , do którego przypisana została obserwacja  $i$

**R-kwadrat (R-square) =  $1 - \left(\frac{\text{Within STD}}{\text{Total STD}}\right)^2$**

## **INTERNET + WŁASNE PRZEMYŚLENIA:**

Segmentacja pozwala na odnalezienie potrzeb klientów, ale umożliwia także wykorzystanie wydzielonych segmentów w połączeniu z obliczaniem CLV. Zamiast wyliczać uśrednione CLV dla całej populacji klientów można policzyć to na segmentach co pozwala na skuteczniejsze działanie. Pokażę to jak należy alokować zasoby dla wybranych segmentów.

Odpowiednia segmentacja i dostosowanie przekazu do każdego z segmentów wpływa także na wzrost CLV w targetowanych segmentach.

## **91. Wymień i omów zastosowania biznesowe modeli oceny wartości klienta w czasie CLV.**

Modelowanie CLV stosowane jest szczególnie w branżach, w których występuje kontraktowa relacja z klientem, tzn. kiedy można określić dokładną datę końca relacji firma-klient. Dwa typy modeli:

- **Gone-for-good** (klienci, którzy zrezygnowali już nie wrócią)
  - Prosty model retencji (mogą przedterminowo odstąpić od umowy, stopa retencji jest stała w czasie)
  - Ogólny model retencji (stopa retencji może się zmieniać w czasie, a wysokość opłat zależy od czasu odejścia)
- **Always-a-share** (nie zakładają, że brak aktywności ze strony klienta oznacza jego stałe odejście)
  - Model migracji (model buy, no-buy; model recency, model recency-frequency)

Poza tym występuje jeszcze model renty klienckiej (klienci podpisują umowę na określoną liczbę okresów rozliczeniowych i nie mają możliwości jej przedwczesnego rozwiązania).

Wartość klienta w czasie (ang. Customer Lifetime Value - CLV) to zdyskontowana suma przyszłych wpływów, które można przypisać do relacji z klientem. Pozwala oszacować zysk, jaki firma osiągnie w przyszłości dzięki klientowi. Maksymalizacja CLV jest jednym z głównych celów organizacji nastawionych na osiąganie zysku. Jednym z zastosowań CLV jest wspieranie decyzji dotyczących ustalania wydatków na przyciąganie nowych klientów. Celem jest identyfikacja potencjalnych klientów z wysoką wartością CLV i unikanie tych o niskiej wartości w czasie. Modelowanie CLV stosowane jest szczególnie w branżach, w których występuje kontraktowa relacja z klientem, tzn. kiedy można określić dokładną datę końca relacji firma-klient.

#### Przykłady:

- Klienci operatora komórkowego kończą obowiązującą ich umowę i przestają opłacać rachunki
- Subskrypcje treści medialnych kończą się, kiedy użytkownicy rezygnują z nich lub nie decydują się na ich odnowienie
- Członkostwo w programie zdrowotno-sportowym wygasza w określonym terminie

#### MODELE SEGMENTACJI:

- segmentacja rynku
- customizacja i personalizacja w obrębie podsegmentów

#### MODELE RETENCIJ:

- jak dużego zysku może oczekwać firma z relacji z klientem, aż do czasu jej zakończenia
- jak zmieniająca się w czasie stopa retencji wpływa na przyszłe zyski

#### MODEL MIGRACJI

- dla usług podróżniczych, finansowych, samochodowych czy sprzedaży detalicznej
- celem jest estymacja przyszłej wartości klienta
- ilu aktywnych klientów będzie po  $n$  okresach i jaki zysk zapewnią

## DODATEK

### Modele migracji

Stosowane do estymacji CLV, gdy między klientem a firmą niekoniecznie występuje relacja kontraktowa.

Zakładamy, że takie organizacje przyciągają klientów, którzy generują lub nie zysk dla firmy podczas dyskretnych okresów o równej długości, np. miesiąc czy rok.

Brak aktywności nie jest tu oznaką końca relacji.

Celem jest estymacja przyszłej wartości klientów i uzyskanie odpowiedzi na następujące pytania:

- Ile wynosi CLV klienta? (Odpowiedź na to pytanie wspomaga decyzje marketingowe dotyczące działań akwizycyjnych czy utrzymaniowych)
- Ilu aktywnych klientów będzie po  $n$  okresach i jaki zysk zapewnią?
- Jeśli zwiększone zostaną inwestycje w podniesienie retencji klientów (zatem w zwiększenie stopy retencji), jak wpłynie to na wielkość i zyskowność bazy klientów?
- Przyjmując pewien poziom retencji, ilu klientów należy zdobyć, aby osiągnąć pewien strategiczny cel jak na przykład: utrzymanie obecnego poziomu zyskowności lub zwiększenie liczby aktywnych klientów o 20% przez kolejne dwa lata?
- Modele retencji nie uwzględniają klientów, którzy ponownie się uaktywnili. Jak firmy mogą poprawnie uwzględnić ten typ klientów?

Zakładane jest, że klient podczas analizowanego okresu znajduje się w określonym stanie. Przykładem stanu może być dokonanie zakupu w poprzednim okresie. Klienci generują zysk w zależności od ich stanów oraz migrują pomiędzy nimi przez kolejne okresy z pewnym prawdopodobieństwem przejścia.

Po więcej do prezki trzeba iść bo to cały wykład o tym.

#### **Model retencji**

### **Modele estymujące CLV**

#### **1. Model renty klienckiej (ang. customer annuity model)**

- Klienci podpisują umowę na określona liczbę okresów rozliczeniowych i nie mają możliwości jej przedwczesnego rozwiązania

#### **2. Prosty model retencji (ang. simple retention model (SRM))**

- Zakłada, że klienci mogą przedterminowo odstąpić od umowy, stopa retencji jest stała w czasie i wśród klientów, a wpływy pieniężne są niezależne od czasu odejścia

#### **3. Uogólniony model retencji (ang. general retention model (GRM))**

- Zakłada, że stopa retencji może się zmieniać w czasie, a wysokości opłat zależą od czasu odejścia

**Tu też do prezki bo tego jest po prostu dużo.**

92. Przedstaw metody łączenia tabel w SAS i SQL.

## SQL

W Języku SQL tabele możemy łączyć podając więcej niż jedną tabelę w klausuli WHERE . Niezależnie od wybranego typułączenia, w wyniku przetwarzania FROM, otrzymujemy zawsze zbiór elementów (virtual table VT), opisany za pomocą wszystkich kolumn tabel wejściowych. Nie ma znaczenia czyłączysz dwie czy więcej tabel połączeniem wewnętrzny, zewnętrzny. Elementy (rekordy, wiersze) tabeli wynikowej, będą określone zawsze przez wszystkie atrybuty (kolumny) łączonych zbiorów. Łączenie wielu zbiorów (trzech i więcej) sprawdza się do wielokrotnego wykonania operacji łączenia dwóch tabel.

### Typy łączenia

- **INNER JOIN** - łączenie wewnętrzne, tabela zawiera te obserwacje, dla których warunek wynikający z klucza jest prawdziwy
- **LEFT OUTER JOIN (LEFT JOIN)** - łączenie zewnętrzne, tabela zawierająca obserwacje, które mają parę (Spełniają warunek wynikający z klucza, tak samo jak w przypadku INNER JOIN) oraz wszystkie obserwacje znajdujące się w zborze lewym. Ponieważ wiersze dopełniające muszą być również opisane, przez wszystkie kolumny łączonych tabel, wartości dla obserwacji, które nie miały pary będą nullami.
- **FULL OUTER JOIN (FULL JOIN)** - wynikiem są wszystkie obserwacje posiadające parę dla INNER JOIN oraz wszystkie inne obserwacje, które znajdują się w obu zbiorach. Dla dopełnienia informacji dla obserwacji nie posiadających pary są null
- **FUNKCJA UNION / UNION ALL** - Jest to połączenie ze sobą dwóch zbiorów (jeden po drugim). Żeby dana funkcja była dobrze zastosowana, musi spełniać trzy kryteria:
  - Każda klawiszula SELECT musi posiadać taką samą liczbę kolumn
  - Każda z kolumn musi mieć ten sam typ np. pierwsza kolumna zbioru A i pierwsza kolumna zbioru B, ponieważ po połączeniu są jedną kolumną
  - Każda kolumna w obu zbiorach w klawiszule SELECT musi być tej samej kolejności (tylko wtedy połączenie będzie prawidłowe)
- **Różnica**
  - UNION – po połączeniu zbiorów rezultatem jest tabela z unikatowymi wierszami, tzn. że dodatkowo działa DISTINCT. Oznacza to, że jeśli w zborze wyjściowym powinny pojawić się co najmniej dwa wiersze z takimi samymi wartościami, to zostanie tylko jeden z tych wierszy.
  - UNION ALL - Łączy tabele tak samo jak UNION ale nie uruchamia się DISTINCT czyli wynikiem jest dokładnie taka sama liczbą kolumn jaką jest w obu zbiorach wejściowych

## SAS

Do łączenia zbiorów w SAS można wykorzystać funkcję PROC SQL aby łączyć zbiory polecaniami w SQL tzn. JOIN'ami

Struktura zapytani sql w PROC SQL

- Proc sql;
- Select \* from a;
- Quit;

### Używanie 4GL do łączenia zbiorów

- Łączenie pionowe (odpowiednik UNION ALL)
- Outer Join / LEFT JOIN / INNER JOIN

Efektem jest outer join, gdzie kluczem jest id. Rename dla kolumna, spowodował że w tabeli wynikowej mamy kolumny a1 i a2. Wyświetlanie właściwości in dla każdego z e zbiorów pozwala zobaczyć z którego zbioru pochodzą dane w danym wierszu. 1 dla z1 i z2\*/

Concatenate – set data1 data2 w datastep – odpowiednik union all. Możliwe też zastosowanie proc append.

Merge – join w sql

[https://documentation.sas.com/doc/en/pgmsascdc/9.4\\_3.5/lepg/p1pa3hnpchkgf7n1eten\\_sx665vr.htm](https://documentation.sas.com/doc/en/pgmsascdc/9.4_3.5/lepg/p1pa3hnpchkgf7n1eten_sx665vr.htm)

### 93. Przedstaw plusy i minusy przetwarzania danych w SAS i SQL.

Co to jest System SAS?

Struktura Systemu SAS to moduły, które służą do szeroko pojętego procesu przetwarzania danych. Moduły mogą pracować osobno lub w różnych zestawach. Wszystkie one mają ułatwić realizację nadzawanego celu, mianowicie z danych źródłowych stworzenie „informacji” przyjaznej użytkownikowi.

Cechy Systemu SAS

- Uniwersalny dostęp do wszystkich danych – SAS potrafi czytać dane z różnorodnych zbiorów i katalogów niezależnie od ich źródła i formatu, a do popularnych baz danych oferuje interfejsy, które współpracują z bazą bez duplikowania informacji w formacie SAS.
- Łączność – współpracuje z istniejącymi protokołami sieciowymi zapewniając łączność i optymalne wykorzystanie zasobów; pozwala na dzielenie danych i aplikacji w prawdziwie rozproszonym środowisku dając użytkownikowi wybór, gdzie chce przechowywać dane i na jakim komputerze wykonywać programy
- Niezależność sprzętowa – pracuje w identyczny (z punktu widzenia użytkownika) sposób na komputerach dużych, mini i personalnych, a dane i aplikacje są bezpośrednio przenoszone pomiędzy różnymi systemami operacyjnymi.
- Budowa modułowa – Umożliwia instalację i wykorzystanie tylko tych elementów systemu, które są potrzebne w wybranych przez użytkownika zastosowaniach (dostęp, przetwarzanie, analiza, prezentacja).
- Produkt komercyjny - wspierany przez firmę SAS Institute (Statistical Analysis System), co gwarantuje wsparcie programowe i ciągły rozwój systemu. Idea SAS Institute jest „przekształcanie danych w informacje”. Inaczej mówiąc dostarczanie takich rozwiązań, które pozwolą na wydobycie wiedzy z ‘chaosu danych’ w odpowiednim momencie. SAS Institute jest jednym z największych producentów oprogramowania na świecie, przoduje wśród rozwiązań Customer Data Mining, Multi-Channel Campaign Management i Basel II oraz w zakresie nowatorskich platform systemowych takich jak Data Integration Tools, Corporate Performance Management oraz Marketing Resource Management. Firma ma swoje placówki na całym świecie, również w Polsce. SAS zatrudnia wysoko wykwalifikowanych pracowników, głównie programistów i konsultantów, których zadaniem jest służyć swoją wiedzą i doświadczeniem klientom. Profesjonalizm działania firmy został doceniony licznymi nagrodami a klienci, dzięki wdrożeniu SAS-a, są często wyróżniani jako najlepiej zinformatyzowane przedsiębiorstwa.
- Łatwość użytkowania - Użytkownikom Systemu SAS może być każdy. Jest to system bardzo rozbudowany, posiada rozwiązania i interfejsy dla różnej klasy użytkowników. Z SAS-a mogą korzystać wyspecjalizowani programiści, analitycy i projektanci, ale również osoby, które nie czują się na siле programować a jedynie wykorzystać możliwości jakie niesie interfejs graficzny tego systemu.

SQL

SQL (Structured Query Language) – strukturalny język zapytań jest standardowym językiem zaprojektowanym do pracy z relacyjnymi bazami danych. Jest zbiorem komend używanych do wprowadzania, modyfikowania i przeglądania zawartości relacyjnych baz danych. Język SQL pełni trzy główne funkcje: tworzenie bazy i definiowanie jej struktur, wykonywanie zapytań na bazie w celu uzyskania danych niezbędnych do wygenerowania odpowiedzi oraz kontrolowanie bezpieczeństwa bazy danych. Jest wiele sposobów uruchamiania instrukcji SQL i pobierania wyników z bazy danych. SQL jest ukierunkowany, ale i ograniczony do zastosowań w relacyjnych bazach danych. Pozwala w praktyce realizować wszystkie operacje algebry relacji, operacje grupowania danych, obliczenia, etc.

**Zalety:**

- Język wysokiego poziomu - jego składnia i słowa kluczowe mają maksymalnie ułatwić rozumienie kodu programu przez człowieka, tym samym zwiększąc poziom abstrakcji i dystansując się od sprzętowych niuansów.
- Deklaratywny charakter – użytkownik opisuje warunki, jakie musi spełniać końcowe rozwiązanie (co chcemy osiągnąć) a nie szczegółową sekwencję kroków, które do niego prowadzą (jak to zrobić).
- Standaryzacja - W 1986 roku Amerykański Narodowy Instytut Normalizacji (ANSI), a w 1987 Międzynarodowa Organizacja Normalizacyjna (ISO) oficjalnie przyjęły standard języka SQL.
- Przejrzystość, czytelność – SQL jest językiem zbliżonym konstrukcją do naturalnego języka angielskiego; wykorzystywana jest prosta składnia, której z łatwością można się nauczyć
- Solidne podstawy matematyczne - jest oparty na algebraze relacji, tzn. modelu operowania danymi w bazie danych opartym na logicznym rachunku predykatów, zawierającym logikę trójwartościową; bardzo dojrzałe, dobrze poznana i przetestowana metodologia, liczne implementacje, stabilność na rynku
- Optymalizacja zapytań - efektywne mechanizmy realizacji zapytań

**Wady:**

- Nie przestrzeganie standaru przez dostawców - W 1986 roku Amerykański Narodowy Instytut Normalizacji (ANSI), a w 1987 Międzynarodowa Organizacja Normalizacyjna (ISO) oficjalnie przyjęły standard języka SQL. Nadal jednak istnieją różnice w wersjach języka SQL biorąc pod uwagę różnych producentów.
- Operowanie jedynie na strukturach tablicowych - brak możliwości reprezentacji i przetwarzania innych struktur i bardziej złożonych obiektów; Ograniczenie do danych atomowych (niepodzielnych)
- Brak rekurencji i iteracji
- Ograniczone możliwości sterowania przetwarzaniem danych
- Niezgodność modelu pojęciowego z modelem implementacyjnym

**94. Na czym polega makroprogramowanie w SAS?**

## DEFINICJA MAKR

Język makr to język, który rozszerza możliwości standardowego 4GL. Dzięki umiejętności pisania makr, użytkownik może zautomatyzować wiele procesów, uruchamiając programy warunkowo (np. kod generujący raport tygodniowy co piątek, a raport miesięczny w ostatni dzień miesiąca), czy też dynamicznie tworząc kod SAS-owy.

## CELE UŻYTKOWANIA

- Automatyzacja procesu pisania kodu SAS-owego
- Dynamiczne generowanie kodu
- Warunkowe uruchamianie kodu
- Parametryzacja kodu

## CHARAKTERYSTYKA MAKROPROGRAMÓW

- Makra rozpoczynają się znakiem procenta (%), po którym występuje nazwa wyrażenia
- Kończą się średnikiem (;)
- Wykonywane są przez procesor makr, a nie zwykły procesor 4GL
- Makrozmienne rozpoczynają się znakiem ampersand (&), po którym występuje ich nazwa

### Proces komplikacji

Zanim skaner słów przekaże kod do procesora 4GL sprawdzi, czy nie zawiera on makrowyrażenia. Jeżeli takie napotka, to przekaże je do procesora makr, który je rozwinie i zwróci z powrotem do skanera. Dopiero wtedy uruchomiony zostanie procesor 4GL. W praktyce oznacza to, że makra wykonują się na samym początku procesu komplikacji.

### Makrozmienne

Przed utworzeniem makrozmiennej należy zaznaczyć, że kompilator traktuje wszystkie wartości jako tekst i nie można wymusić, aby było inaczej. Jednak nie oznacza to, że na makrozmiennych nie można wykonywać operacji arytmetycznych - odpowiada za to funkcja eval opisana w rozdziale Funkcje makro.

## DEFINOWANIE

Makrozmienne można definiować na kilka różnych sposobów. Jednym z nich jest instrukcja %LET. Można z niej korzystać w dowolnym miejscu programu.

Znaki cudzysłowu nie są potrzebne wszystkie znaki (za wyjątkiem początkowych i kończących spacji) występujące do średnika są traktowane jako wartość zmiennej. Utworzona w taki sposób zmienna w dowolnym miejscu poza ciałem makra jest globalna, a tym samym dostępna w dowolnym miejscu programu. Aby utworzyć makrozmienną w DATA STEP należy skorzystać z funkcji symput lub symputx

## CHARAKTERYSTYKA

- Dzielą się na globalne lub lokalne
- Makrozmienne lokalne to takie, które są dostępne wyłącznie w makrze, w którym zostały one zdefiniowane
- Makrozmienne globalne są dostępne w dowolnym miejscu programu, definiowane są poza ciałem makra, w ciele makra za pomocą wyrażenia %GLOBAL nazwa\_zmiennej lub za pomocą funkcji symputx z opcją G.
- Mają minimalną długość 0 znaków (wtedy przyjmują wartość null)
- Mają maksymalną długość 65 534 znaków (64K)
- Przechowują wartości numeryczne jako tekst

**95. Przedstaw plusy, minusy sekwencyjnego przetwarzania danych oraz jego inne alternatywy.**

[https://pl.wikipedia.org/wiki/Obliczenia\\_r%C3%B3wnoleg%C5%82e](https://pl.wikipedia.org/wiki/Obliczenia_r%C3%B3wnoleg%C5%82e)

Zalety	Wady
<ul style="list-style-type: none"><li>Nie występuje problem zakleszczenia</li><li>Spójność danych</li><li>Łatwo określić na którym etapie występuje problem z przetwarzaniem i go rozwiązać</li></ul>	<ul style="list-style-type: none"><li>Działanie kolejnych bloków jest uzależnione od wyników poprzednich bloków – niepowodzenie jednego bloku powoduje niepowodzenie procesu</li><li>Wolne przetwarzanie całego zbioru danych</li></ul>

## 96. Przedstaw przykłady procedur Base SAS i SAS/STAT.

**Proc append** - Procedura APPEND dodaje obserwacje z jednego zestawu danych SAS na końcu innego zestawu danych SAS.

Możliwe opcje (oczywiście nie wszystkie):

- 1) base - zbiór do którego insertowane są wartości z innego zbioru
- 2) data - zbiór który jest insertowany do głównego zbioru
- 3) force - zmusza procedurę append do konkatenacji zestawów danych.

**Proc contents** – Procedura pozwalającą zobaczyć zawartość zestawu danych i ukazuje katalog biblioteki SAS. Generuje raport, w którym można zobaczyć metadane dotyczące tabeli i metadane dotyczące zmiennych.

**Proc copy** – Procedura ta przenosi jeden lub więcej zbiorów SAS z jednej biblioteki do drugiej. Z biblioteki źródłowej zbiorystają skasowane.

**Proc delete** - określa jeden lub więcej plików SAS, które mają zostać usunięte z danej

**Proc import** – Za pomocą tej procedury można zaimportować zewnętrzne pliki do zestawu danych SAS. Ta procedura ma mnóstwo opcji m.in.:

**Proc means** – Zapewnia narzędzia do podsumowywania danych do obliczania statystyki opisowej dla zmiennych we wszystkich obserwacjach oraz grupach obserwacji

**Proc export** – Odczytuje dane z podanej biblioteki SAS i zapisuje je w zewnętrznymźródle.

**Proc format** – Procedura umożliwia zdefiniowanie własnych formatów zmiennych.

**Proc tabulate** – Oblicza podobne statystyki opisowe ja inne procedury statystyczne. M.in. means, freq. Zapewnia proste i skuteczne tworzenie raportów tabelarycznych, elastyczność w klasyfikowaniu wartości zmiennych i ustalaniu hierarchicznych relacji między zmiennymi .

**Proc score** – procedura z wielokrotnią wartością z dwóch zestawów danych SAS, z których jeden zawiera współczynniki (np.. Oceny czynników, czy regresji) a drugi zawiera surowe dane, które mają zostać ocenione przy użyciu współczynnika z pierwszego zestawu danych. Wynikiem tego mnożenia jest zestaw danych SAS zawierający (ocena wyników jako) liniowe kombinacje współczynników

**Proc freq** – Generuje tabele częstotliwości i nieprzewidzianych wartości w jedną stronę. W przypadku tabel dwuwymiarowych procedura oblicza testy i miary powiązania. W przypadku tabel n-way, procedura zapewnia analizę wartościową poprzez obliczanie statystyk w warstwach.

Modele nieparametryczne – **lifetest**

Modele parametryczne – **lifereg**

Modele semiparametryczne – **phreg**

**Proc surveyselect** – procedura zapewnia różnorodne metody wyboru losowych próbek opartych na prawdopodobieństwie. Procedura może wybrać prostą próbkę losową lub próbkę zgodnie ze złożonym projektem wieloetapowym który obejmuje stratyfikację, grupowanie i nierówne prawdopodobieństwa selekcji. Procedura pozwala uniknąć stronnictwa selekcji i umożliwia wykorzystanie teorii statystycznej do dokonowania prawidłowych wniosków z próby populacji.

+ Proc Reg, Proc Logistic, Proc MI

+ BASE – Proc Corr, Freq, sql, print

## 97. Jakie statystyki opisowe są odporne na wartości nietypowe?

Często zmienne odstające biorą się z błędów pomiarowych lub pomyłek przy wprowadzaniu informacji do systemu/baz danych. Obserwacje nietypowe utrudniają, a czasami wręcz uniemożliwiają przeprowadzenie prawdziwej analizy.

Miary w statystyce opisowej można podzielić na klasyczne i pozycyjne.

**Miary klasyczne** to miary wynikowe, które są obliczane na podstawie wszystkich zaobserwowanych wartości cechy. Możemy powiedzieć, że są wynikową wszystkich wartości cechy. Miary te nazywamy miarami opartymi o momenty, gdyż są one momentami bądź są one skonstruowane w oparciu o momenty zwykłe lub centralne.

**Miary pozycyjne** nie są miarami wynikowymi, obliczamy je na podstawie tylko niektórych wartości cechy, które wyróżniają się swoją pozycją w rozkładzie (np. mediana, czyli wartość środkowa).

Miary pozycyjne nie są wrażliwe na wartości odstające (ekstremalne).

Wśród miar średnich, miarami pozycyjnymi są:

- Mediana (drugi kwartyl);
- wartość środkowa (moda)

**Dominanta:**

- w szeregu punktowym jest to ten wariant cechy, któremu odpowiada największa liczebność
- w szeregu przedziałowym - należy do przedziału, któremu odpowiada największa liczebność

**Kwantyle:** dzielą badaną zbiórowość na określone części pod względem liczebności np. kwartale, decyle, centyle. Kwartyli jest trzy: Q1, Q2, Q3 i dzielą zbiórowość na 4 części. Kwartył drugi to mediana. Decyl jest 9 i dzieli zbiórowość na 10 części, decyl piąty to mediana itd.

Wśród miar rozproszenia, miarami pozycyjnymi są:

- Rozstęp - empiryczny obszar zmienności:  $R = \text{xmax} - \text{xmin}$
- Odchylenie ćwiartkowe - mierzy poziom zróżnicowania w połowie obszaru zmienności

#### **Miary klasyczne cd**

**średnia arytmetyczna** – Jest ilorazem sumy wartości zmiennej i liczby badanej zbiorowości.

**Średnia harmoniczna** – Jest odwrotnością średniej arytmetycznej z odwrotnością wartości zmiennych.

**Średnia geometryczna** – Jest pierwiastkiem k-tego stopnia z iloczynu zmiennych

**wariancja (drugi moment centralny)** – klasyczna miara zmienności. Intuicyjnie utożsamiana ze zróżnicowaniem zbiorowości; jest średnią arytmetyczną kwadratów odchyleń (różnic) poszczególnych wartości cechy od wartości oczekiwanej.

### **98. Jakie statystyki opisowe należy stosować w przypadku prób pobranych z populacji o rozkładzie innym niż rozkład normalny?**

Dla zmiennych ilościowych możemy mówić o takich statystykach opisowych jak:

- Mediana zamiast średniej arytmetycznej
- Rozstęp kwartylowy zamiast odchylenia standardowego
- Korelacja Spearmana zamiast Pearsona

Można powiedzieć o dominancie, jednak ona jest niezależna od rozkładu – dla obu rozkładów jest dobra, nie ważne czy rozkład jest normalny czy nie. Dominanta odnosi się zarówno do zmiennej jakościowej jak i ilościowej.

Dla zmiennych jakościowych nie można mówić o rozkładzie normalnym.

W sytuacji gdy dane nie podlegają rozkładowi normalnemu, a pomiary są w najlepszym wypadku wyrażone na skali porządkowej, wówczas obliczanie standardowych statystyk opisowych (np. średniej, odchylenia standardowego) nie jest najlepszym sposobem zbiorczego przedstawienia danych.

Statystyki nieparametryczne i rozkłady pozwalają na wyliczanie szerokiego zakresu różnych miar położenia (średnia, mediana, moda itd.) i dyspersji (wariancja, odchylenie przeciętne, rozstęp kwartylowy itd.), dając w ten sposób pełny obraz danych.

Miary współzależności:

**R Spearmana.** Przy obliczaniu R Spearmana (por. Siegel i Castellan, 1988) zakłada się, że rozważane zmienne zostały zmierzone co najmniej na skali porządkowej (rangowej), tzn. że indywidualne obserwacje mogą być zestawione w dwóch uporządkowanych szeregach. Współczynnik R Spearmana można traktować podobnie jak współczynnik korelacji liniowej Pearsona, tj. w kategoriach procentu wyjaśnianej zmienności, tyle że R Spearmana jest wyliczany w oparciu o rangi.

**Tau Kendalla.** Przy stosowaniu tego współczynnika powinny być spełnione te same podstawowe założenia jak w przypadku R Spearmana. Podobna jest też ich moc statystyczna. Jednakże wielkości obu współczynników zwykle nie pokrywają się, gdyż ich podstawy logiczne oraz formuły obliczeniowe bardzo się różnią.

**Gamma.** Statystyka gamma (Siegel i Castellan, 1988) jest zalecana w przypadkach, gdy dane zawierają wiele powiązanych obserwacji (tzn. obserwacji o takich samych wartościach). W kategoriach podstawowych założzeń jest ona odpowiednikiem R Spearmana lub tau Kendalla, natomiast pod względem interpretacji i obliczania jest bardziej podobna do współczynnika tau Kendalla. Krótko mówiąc, współczynnik gamma opiera się również na prawdopodobieństwie; liczy się go jako różnicę między prawdopodobieństwem, że uporządkowanie dwóch zmiennych jest zgodne a prawdopodobieństwem, że jest niezgodne, podzieloną przez 1 minus prawdopodobieństwo występowania obserwacji powiązanych. W tym sensie jest bardziej odpowiednikiem tau Kendalla, prócz tego, że powiązania są wprost uwzględniane w obliczeniach.

### **99. Przedstaw plusy i minusy struktur danych: analitycznej i transakcyjnej.**

#### OLAP vs OLTP

Zadania OLAP można scharakteryzować porównując je z drugim popularnym rodzajem przetwarzania danych w systemach bazodanowych: OLTP (On-Line Transactional Processing, przetwarzanie transakcyjne). Transakcyjne przetwarzanie danych to operacje dokonywane w bieżących (produkcyjnych) bazach danych przedsiębiorstwa, wykorzystywanych do codziennej pracy. Są to systemy optymalizowane pod kątem maksymalnej wydajności transakcyjnej, wysokiej równoległości i dostępności. Przykłady: system bankowy obsługujący odczytywanie i modyfikację salda rachunków klientów; system finansowo-księgowy obsługujący supermarket i połączony z kasami fiskalnymi; baza danych obsługująca aktywną zawartość portalu internetowego, system billingowy sieci komórkowej itp.

#### Podstawowe cechy systemów OLTP to:

- wykonywanie dużej liczby prostych zapytań pochodzących od wielu użytkowników (nierazko są to setki zapytań na sekundę)
- system bazodanowy powinien być zoptymalizowany pod kątem szybkiego wyszukiwania danych
- częste operacje dodawania, usuwania i modyfikacji pojedynczych rekordów
- wymagany natychmiastowy dostęp do aktualnych informacji

Przetwarzanie typu OLAP to przede wszystkim tworzenie raportów (zwyczajnie predefiniowanych) obejmujących zestawienia tabelaryczne i wykresy. Ten rodzaj przetwarzania przeznaczony jest zwykle dla innego rodzaju użytkowników: kierownictwa, analityków, administratorów. Przykłady: raport dynamiki sprzedaży produktów w różnych krajach, dla którego źródłem są pojedyncze zapisy wszystkich transakcji przy kasach 100 supermarketów danej sieci z ostatnich trzech lat; raporty podsumowujące obroty i prowizje klientów banku w rozbiocie na miesiące, rodzaje opłat i grupy klientów; typowe statystyki miesięczne ruchu internetowego na serwerach WWW.

#### Podstawowe cechy systemów OLAP to:

- niewielka liczba zapytań, lecz dotyczących wielkich ilości danych (podsumowania itp., mogą to być zapytania zadawane raz na kilka minut przez kilku-kilkuset użytkowników)
- systemy te zasadniczo tylko odczytują informację z bazy; jeśli system OLAP jest logicznie oddzielony od baz transakcyjnych, to informacje są cyklicznie uzupełniane (dodawanie dużych grup nowych rekordów)
- nie zakładamy pełnej aktualności informacji: dane mogą być dostępne z opóźnieniem (najlepiej znany z góry, np. jednodniowym), a same obliczenia mogą trwać od sekund do wielu godzin.

Rozbieżność wymagań pomiędzy przetwarzaniem typu OLTP i OLAP uzasadnia rozdzielenie tych zadań. Jest to jeden z powodów, dla których tworzy się hurtownie danych - oddzielne (logicznie i fizycznie) systemy informatyczne, wykorzystujące inne rodzaje silników bazodanowych, mające inaczej skonstruowaną zawartość, niż systemy produkcyjne (transakcyjne) przedsiębiorstwa. Z drugiej strony, wymagania użytkowników hurtowni danych powodują wprowadzanie coraz większej liczby elementów OLTP do funkcjonalności hurtowni danych, co może być związane, np., z potrzebą generowania raportów w czasie rzeczywistym podczas ładowania nowych danych, tudzież z koniecznością wykonywania raportów operacyjnych przez wielu użytkowników jednocześnie, co wiąże się z coraz bardziej ostatnio popularnym pojęciem Operational BI.

### 100. Co to jest PDV i sekwencyjne przetwarzanie danych w systemie SAS?

**PDV** jest to struktura stworzona w trakcie komplikacji, zawierająca zmienne ze wszystkich zdefiniowanych zbiorów wejściowych oraz wszystkie zmienne zadeklarowane w kodzie.

Kolejność zmiennych w wektorze PDV to kolejność ich występowania w trakcie odczytywania. Przed fazą wykonania wartości wszystkich zmiennych są inicjowane na braki, poza przypadkiem podania wartości początkowych. Podczas fazy wykonania wartości zmiennych są nadpisywane wartościami otrzymanymi w każdym obrocie pętli głównej.

Wektor PDV zawiera dwie automatycznie tworzone zmienne, które mogą być wykorzystywane podczas przetwarzania danych:

- `_N_` – numer bieżącej iteracji pętli głównej;
- `_ERROR_` – zmienna informująca o wystąpieniu błędu, jej wartość domyślna to 0 (brak błędu), wartość 1 oznacza wystąpienie błędu podczas fazy uruchomienia.

## SEKWENCYJNE PRZETWARZANIE DANYCH

### Definicja

Najprościej mówiąc, przetwarzanie sekwencyjne to sposób przetwarzania danych w zadanej kolejności, najczęściej takiej, w jakiej są one przechowywane w tabeli.

### Przetwarzanie sekwencyjne w SAS

#### Teoria

Każdy program w języku SAS 4GL składa się z bloków (ang. Step), które wykonywane są sekwencyjnie. Wpierw wszystkie instrukcje są kompliowane, a następnie wykonywane.

Kompilacja i wykonywanie programu jest również wykonywane sekwencyjnie. Komunikacja między blokami może odbywać się z pomocą makrozmiennych, makroprogramów lub z wykorzystaniem pośrednich zbiorów.

Sekwencyjne przetwarzanie danych jest defaultową opcją dla instrukcji pisanych w 4GL.

Jak to działa w praktyce?

Omówmy to na przykładzie podstawowej struktury w 4GL, czyli data stepie.

Instrukcja SET czyta wszystkie obserwacje ze zbioru począwszy od pierwszej (w przypadku gdy chcemy iterować po pliku, a nie po tabeli SET trzeba zamienić na INFILE).

W ramach DATA STEP-u tworzona jest automatycznie pętla główna (ang. implicit loop), w obrębie której: czytana jest kolejna obserwacja z wejściowego zbioru danych (lub wiersz z wejściowego pliku płaskiego), wykonywane są instrukcje będące treścią danego kroku, finalna postać obserwacji zapisywana jest do zbioru wynikowego. Domyślnie pętla główna wykonywana jest dla każdej obserwacji w zbiorze wejściowym. W przypadku, gdy w DATA STEPIE nie ma żadnej instrukcji czytającej ze zbiorów, pętla główna wykonuje się tylko jeden raz.

**PDV (Program Data Vector)** – struktura tworzona w trakcie komplikacji DATA – step do wektora, który zawiera zmienne ze zbiorów wejściowych, zmienne zdeklarowane i zmienne automatyczne.

Do zmiennych automatycznych należą:

- `_N_` - zawiera numer bieżcej iteracji
- `_ERROR_` - sygnalizuje pojawienie się błędu podczas przetwarzania

#### Data step – zasada działania

zb_in	
ZMIENNA_A	ZMIENNA_B
1	A
2	B
3	C
4	D

DATA zb\_out;  
SET zb\_in;  
*instrukcja1;*  
*instrukcja2;*  
  
*output\*;*  
*return\*;*  
RUN;



**Sekwencyjne przetwarzanie danych** – sposób przetwarzania danych w kolejności ich przechowywania w tabeli.