AI risinājumi uzņēmējdarbības produktivitātes celšanai:

3. Klasterizācija un Lēmumu koki

MI risinājumu izstrādātāja Aiga Andrijanova
aiga.andrijanova@gmail.com

Nodibinājums "Ventspils Augsto tehnoloģiju parks"
AI risinājumi uzņēmējdarbības produktivitātes celšanai
Projekta numurs: 2.3.1.2.i.0/2/23/A/CFLA/002

Finansē
Eiropas Savienība
NextGenerationEU

2027
Nacionālais
attīstības plāns

Kas es esmu?

Darba pieredze:

5 gadu pieredze **biznesa procesu automatizācijā** izmantojot datu zinātni

Pašlaik - MI izstrādātāja uzņēmumā **APPLY**

Izglītība:

Maģistra grāds datu zinātnē un mašīnmācībā no Imperial College London

Kad mēs tiekamies?

3. lekcija (18.12.)

Klasterizācija un lēmumu koki

6. lekcija (22.01.)

Video apstrādes modeļi (rediģēšana, GenAI)

8. lekcija (05.02.)

Mākslīgā intelekta risinājumu dzīves cikls, juridiskais regulējums

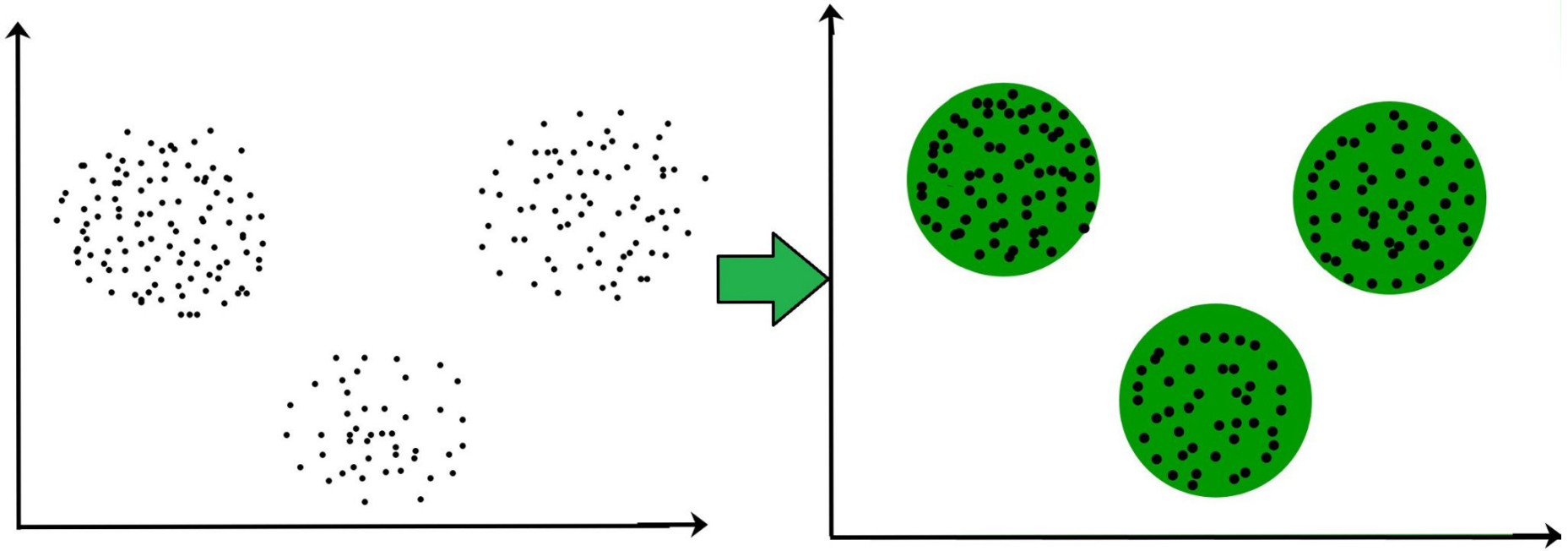
Biežākās kļūdas un labākie risinājumi mājasdarbā

Klasterizācija

Clustering

Kas ir klasterizācija?

Līdzīgo lietu grupēšana



Klasterizācijas pielietojumi

"Miljonu vērtie" klasterizācijas risinājumi:

- 🤳 Apple FaceID
- 🔐 Banku drošības sistēmas
- 🎵 Spotify mūzikas atlase

Reāli risinājumi Jūsu biznesam:

- 👥 Klientu segmentācija
- 📊 Pārdošanas datu analīze
- 🎯 Personalizēts mārketings

💡 Galvenā doma: Tie paši principi, cita mēroga risinājumi

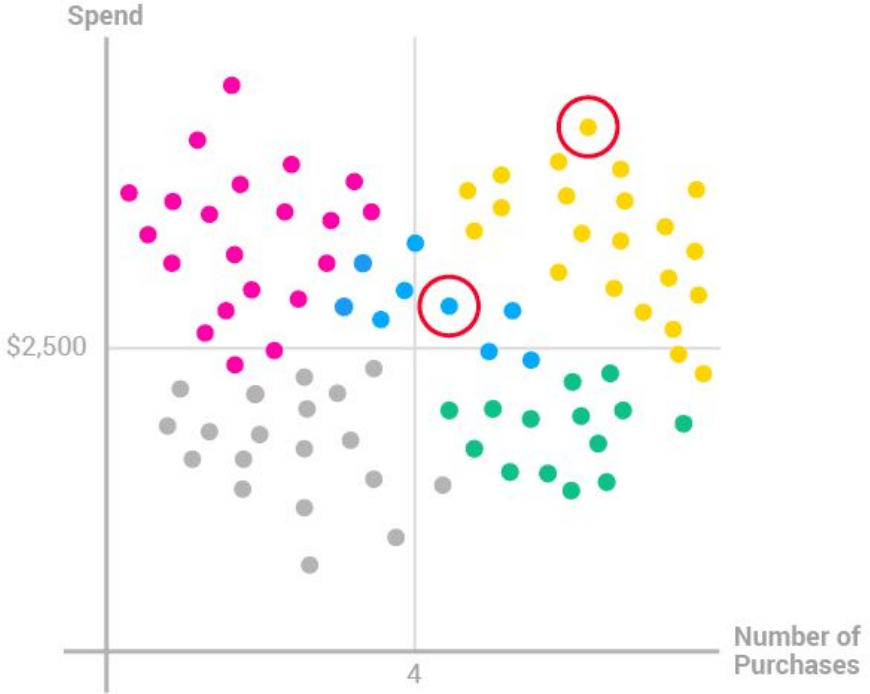
Veikala klientu segmentācija

🌟 Premium klienti (daudz tērē/bieži)

💫 Lojālie ikdienas (maz tērē/bieži)

⭐ Reti, bet dārgi (daudz tērē/reti)

⚪ Gadījuma pircēji (maz tērē/reti)



Ko mēs varam darīt ar šiem datiem tālāk?

⭐ Reti, bet dārgi

- Izveidot atgādinājumu sistēmu
- Piedāvāt ekskluzīvus produktus
- Veidot īpašus sezonālos piedāvājumus
- Koncentrēties uz e-pasta mārketingu
- Analizēt, kas kavē biežākus apmeklējumus

🌟 Premium klienti

- Izveidot īpašu VIP programmu
- Piedāvāt priekšrocības (agrāka piekļuve jaunumiem, īpaši pasākumi)
- Izmantot kā "references klientus" līdzīgu klientu piesaistei
- Analizēt viņu uzvedību, lai saprastu kas padara viņus lojālus
- Aktīvi aizsargāt no konkurentiem

⚪ Gadījuma pircēji

- Identificēt problēmas
- Piedāvāt "pirmā pirkuma" atlaides
- Veidot popularizācijas kampaņas
- Testēt dažādus komunikācijas kanālus
- Aprēķināt klientu piesaistes izmaksas pret iegūto peļņu

💫 Lojālie ikdienas klienti

- Izstrādāt "upselling" stratēģijas
- Piedāvāt atlaides
- Veidot personalizētus ieteikumus
- Ieviest lojalitātes programmu ar punktu krāšanu
- Izmantot viņus jaunu produktu testēšanā

Nekustamā īpašuma segmentācija (2019. g. dati)

Ja izveidotu karti, kur:

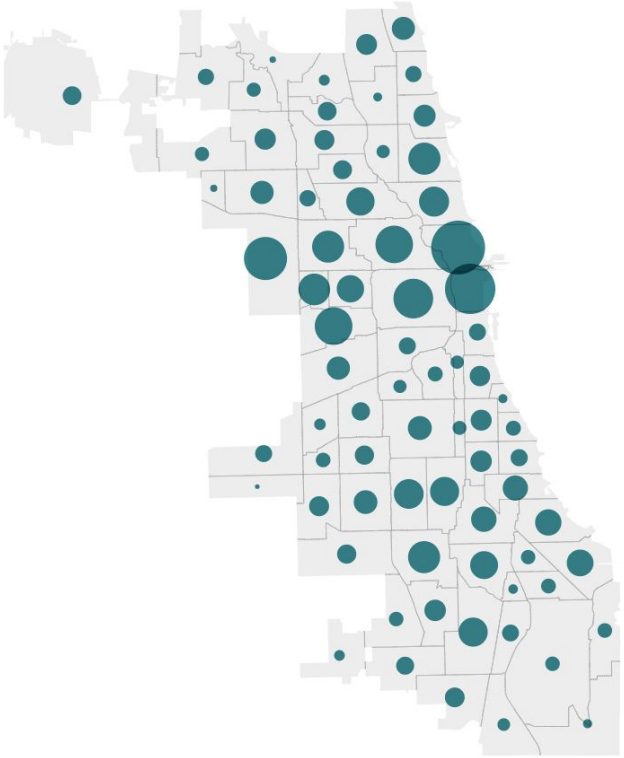
- Indikatora izmērs - platība m²,
- Indikatora krāsa - cena €/m²,

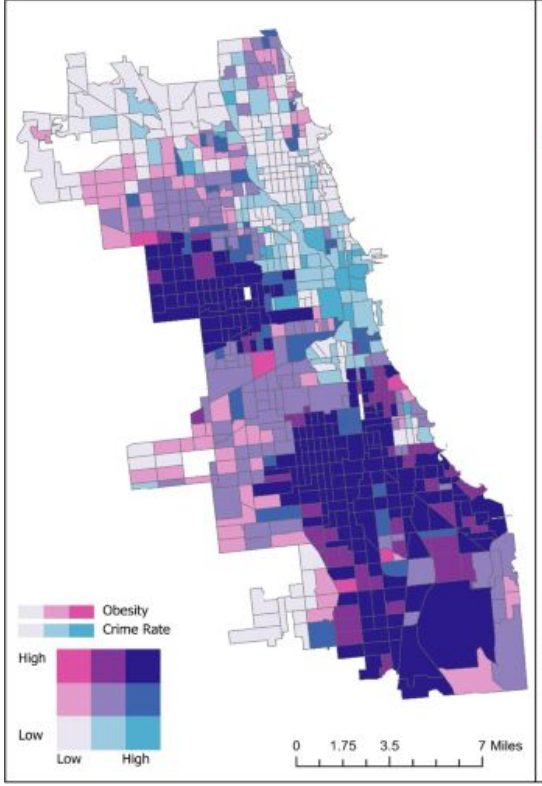
Tad iegūtu klasterus:

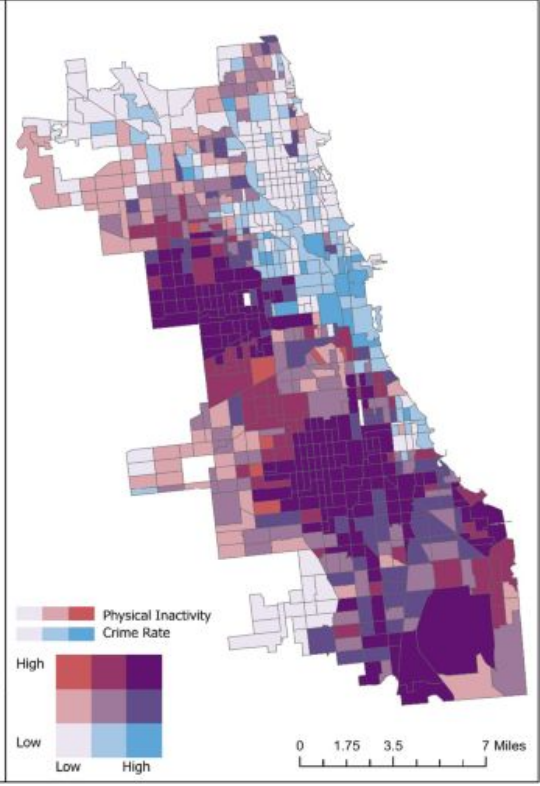
- Rīgas centrs
- Jaunie projekti
- Padomju laika
- Privātmāju rajoni

Chicago Crime by Neighborhood

During the 8 weeks ending between March 22nd & May 10th overall reported crime decreased across the City of Chicago relative to a 3 year average of the same time period.







Updated: 5-18-2020. CPD incident data is not real time. It is available 7 days after an incident occurs. *Neighborhoods with an average week crime of 30 reported incidents or less were excluded.*

Map: Loyola University Chicago, Center for Criminal Justice Research, Policy, and Practice • Source: Chicago Police Department • Get the data • Created with Datawrapper

Ko mēs varam darīt ar šiem datiem tālāk?

Rīgas centrs

- Identificēt potenciālās investīciju iespējas
- Veidot premium mārketinga materiālus
- Analizēt cenu elastību
- Sekot līdzi renovācijas projektiem apkārtnē
- Prognozēt vērtības izmaiņas

Padomju laika ēkas

- Identificēt renovācijas potenciālu
- Sekot līdzi siltināšanas projektiem
- Analizēt komunālo maksājumu izmaiņas
- Veidot tipveida risinājumus
- Prognozēt dzīvokļu tirgus izmaiņas

Jaunie projekti

- Sekot līdzi būvniecības tempiem
- Analizēt pieprasījuma-piedāvājuma attiecību
- Identificēt potenciālās problēmu zonas
- Veidot sadarbību ar attīstītājiem
- Prognozēt infrastruktūras vajadzības

Privātmāju rajoni

- Analizēt zemes vērtības izmaiņas
- Sekot infrastruktūras attīstībai
- Identificēt apbūves potenciālu
- Veidot ilgtermiņa attīstības prognozes
- Analizēt sezonalitātes ietekmi

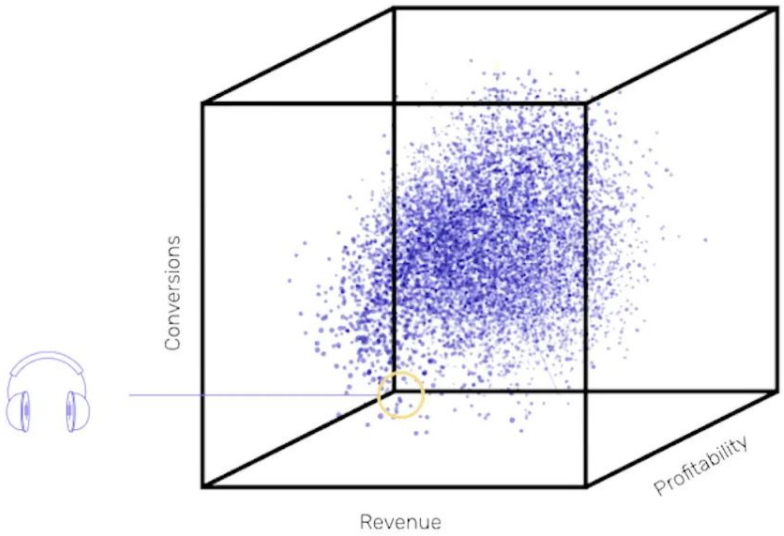
Produktu pozicionēšana tirgū

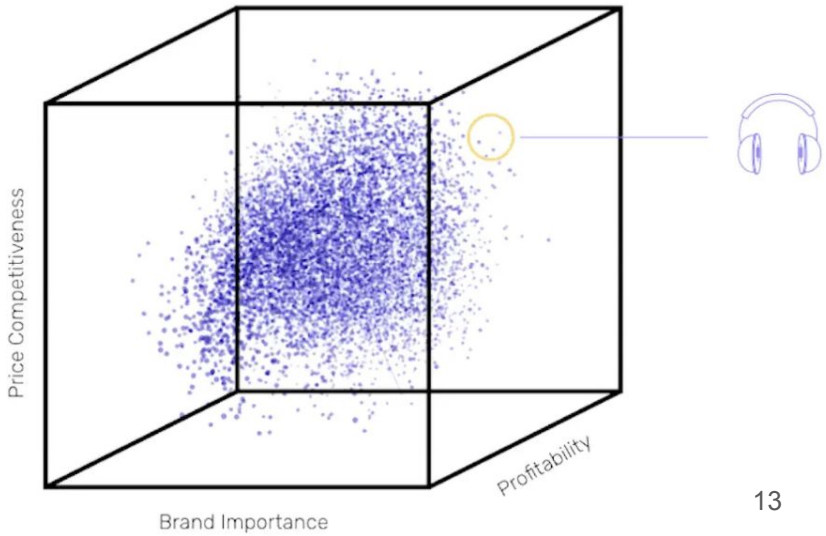
Veikala produktu portfolio analīze:

- Kā mūsu produkti konkurē?
- Kur ir "tukšās" vietas tirgū?

Ko ar šo informāciju darīt?

- Optimizēt cenu stratēģijas
- Pielāgot mārketinga aktivitātes
- Plānot produktu attīstību

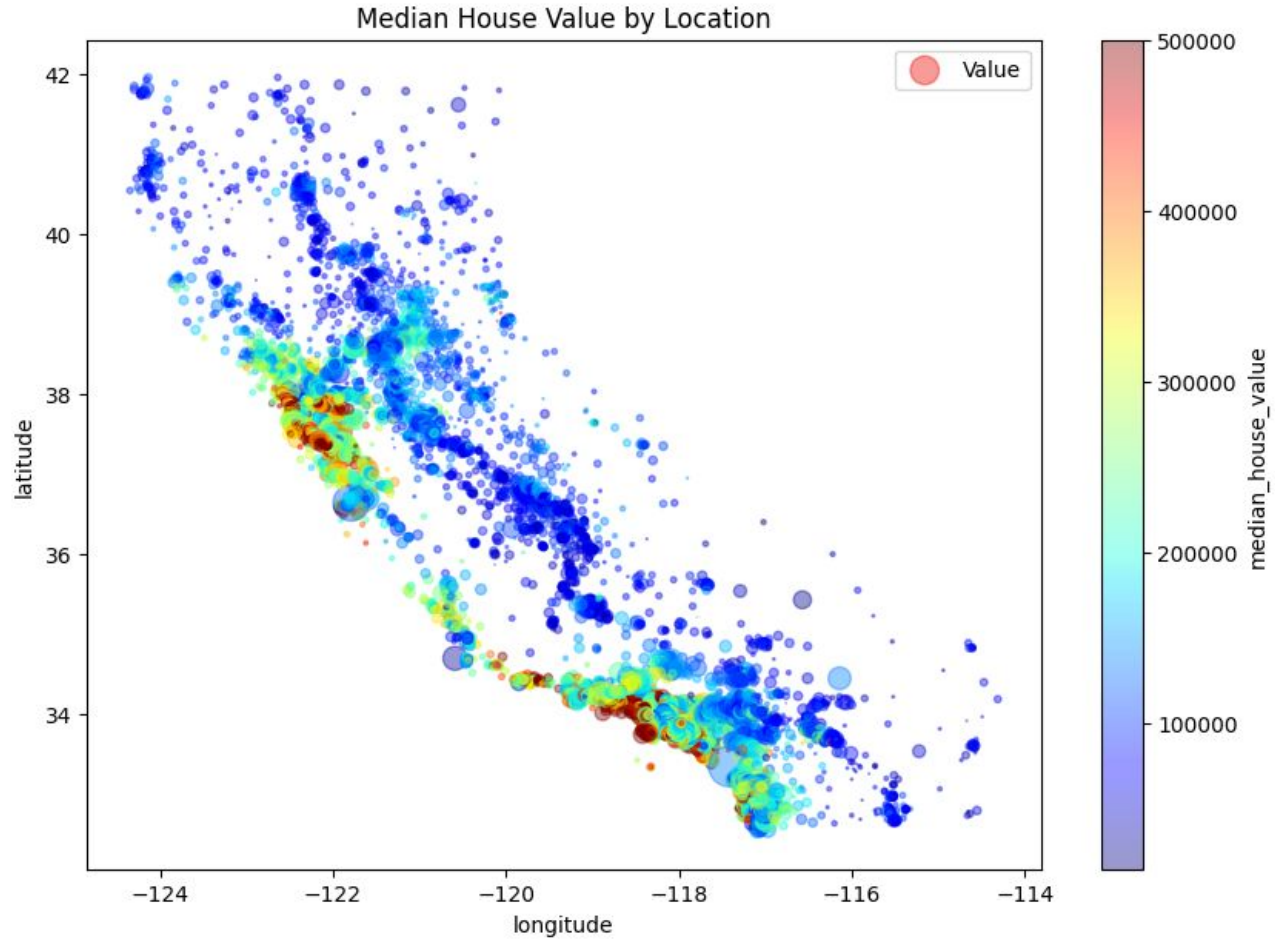




Ko ar šo informāciju nepieciešams un var darīt visos gadījumos?

1. Validēt klasterus ar biznesa ekspertiem
2. Izveidot darbības plānu katram klasterim
3. Noteikt KPI katrai darbībai
4. Veidot **automatizētus risinājumus** (e-pasti, atlaides)
5. **Regulāri pārskatīt** klasteru izmaiņas
6. Testēt dažādas pieejas katram klasterim
7. Mērīt ROI katrai aktivitātei

Vairāk nekā ⅔D gadījumi



California Housing Prices dataset:
https://www.kaggle.com/datasets/camnugent/california-housing-prices

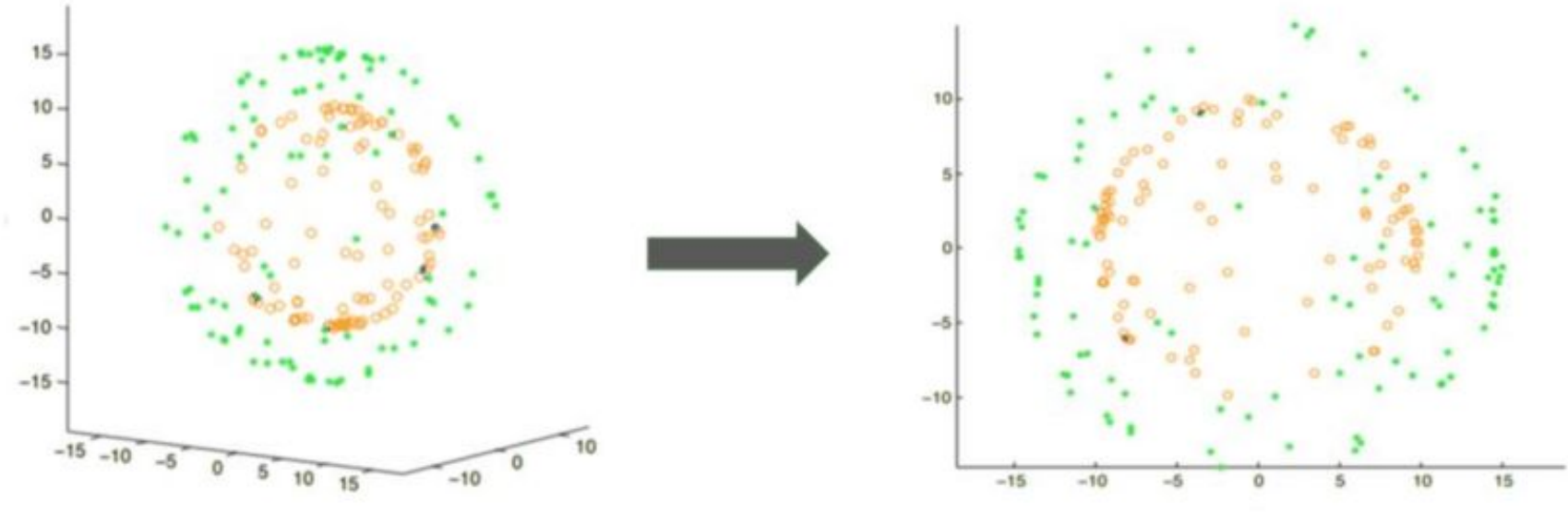
Summary Statistics

	longitude	latitude	housing_median_age	total_rooms	total_bedrooms	population	households	median_income	median_house_value
count	20640.000000	20640.000000	20640.000000	20640.000000	20433.000000	20640.000000	20640.000000	20640.000000	20640.000000
mean	-119.569704	35.631861	28.639486	2635.763081	537.870553	1425.476744	499.539680	3.870671	206855.816909
std	2.003532	2.135952	12.585558	2181.615252	421.385070	1132.462122	382.329753	1.899822	115395.615874
min	-124.350000	32.540000	1.000000	2.000000	1.000000	3.000000	1.000000	0.499900	14999.000000
25%	-121.800000	33.930000	18.000000	1447.750000	296.000000	787.000000	280.000000	2.563400	119600.000000
50%	-118.490000	34.260000	29.000000	2127.000000	435.000000	1166.000000	409.000000	3.534800	179700.000000
75%	-118.010000	37.710000	37.000000	3148.000000	647.000000	1725.000000	605.000000	4.743250	264725.000000
max	-114.310000	41.950000	52.000000	39320.000000	6445.000000	35682.000000	6082.000000	15.000100	500001.000000

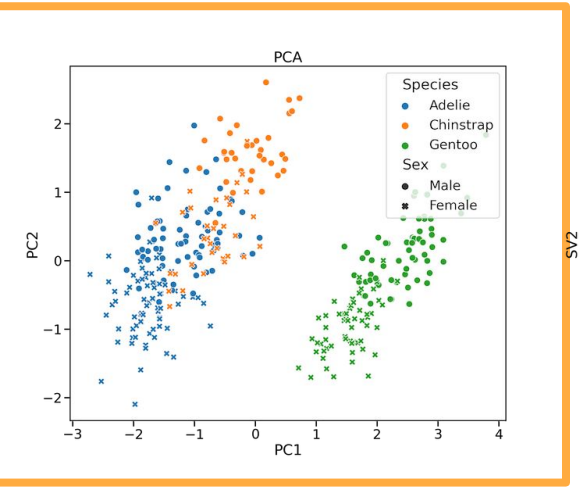
1. longitude: A measure of how far west a house is; a higher value is farther west
2. latitude: A measure of how far north a house is; a higher value is farther north
3. housingMedianAge: Median age of a house within a block; a lower number is a newer building
4. totalRooms: Total number of rooms within a block
5. totalBedrooms: Total number of bedrooms within a block
6. population: Total number of people residing within a block
7. households: Total number of households, a group of people residing within a home unit, for a block
8. medianIncome: Median income for households within a block of houses (measured in tens of thousands of US Dollars)
9. medianHouseValue: Median house value for households within a block (measured in US Dollars)

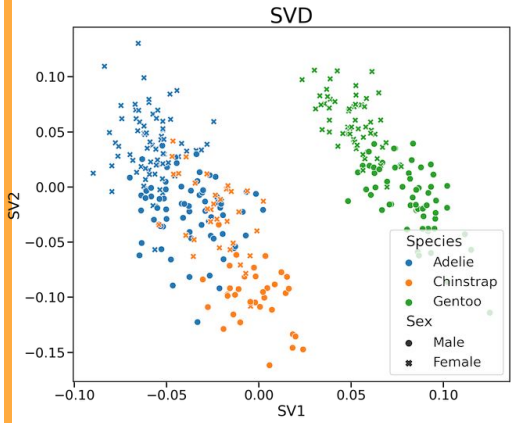
Tāpat varam lietot klasterizācijas algoritmus - tie vienkārši strādās daudzdimensiju telpā

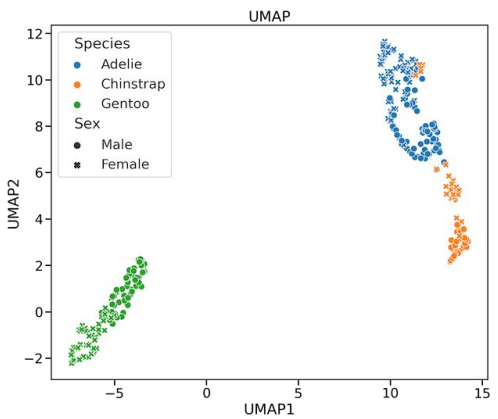
Lai vizualizētu šos klasterus mēs uz klasterizācijas rezultāta lietojam dimensiju redukcijas metodi

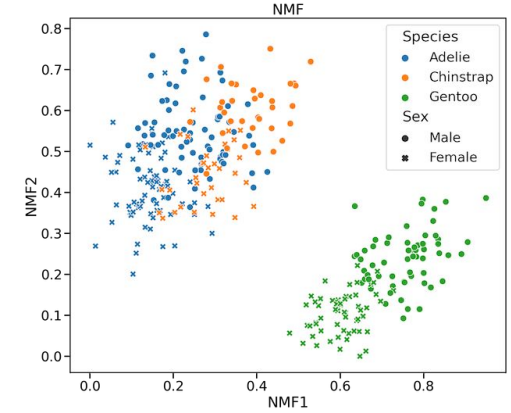


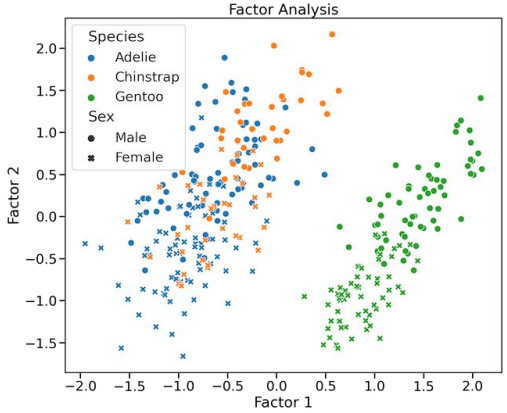
Ir daudz dažādi dimensiju redukcijas algoritmi

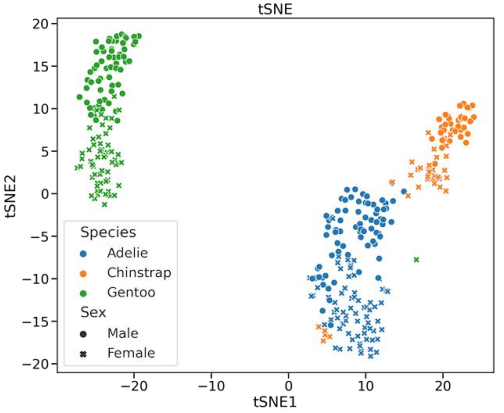




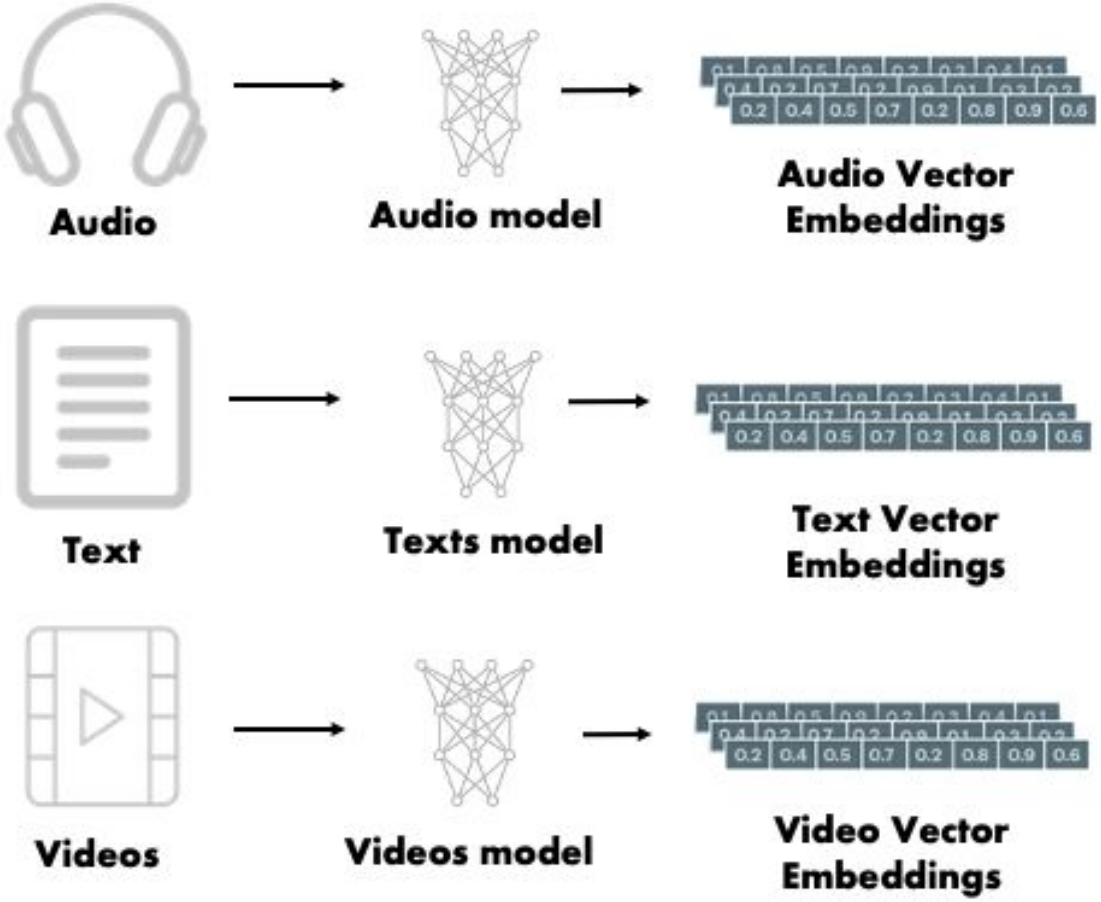




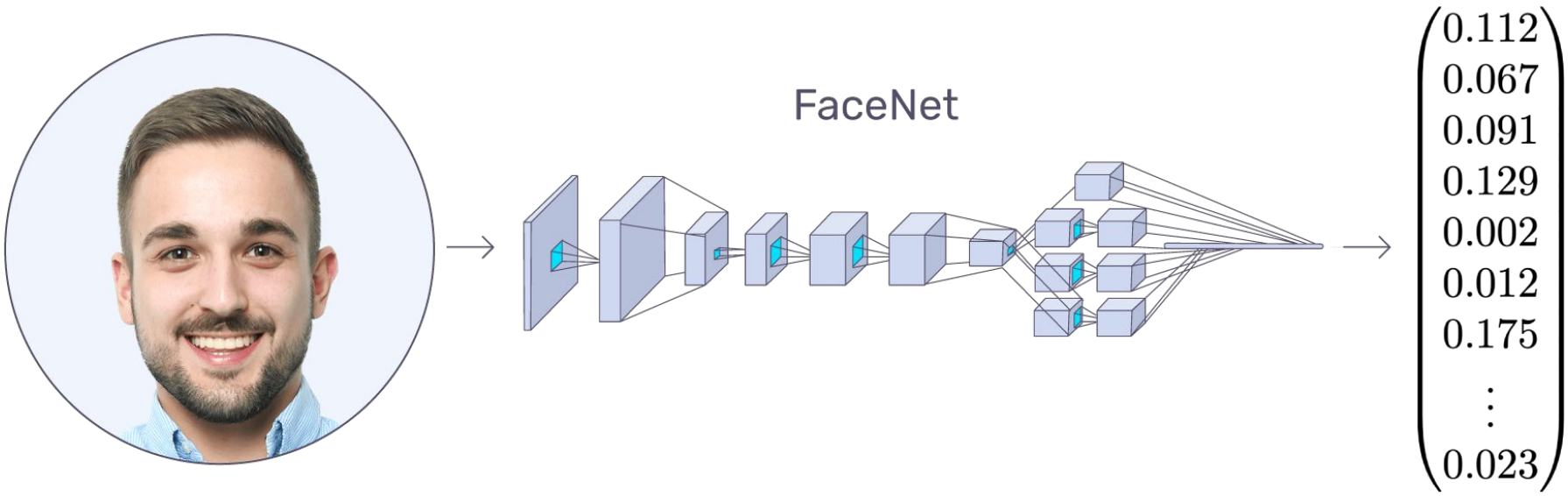


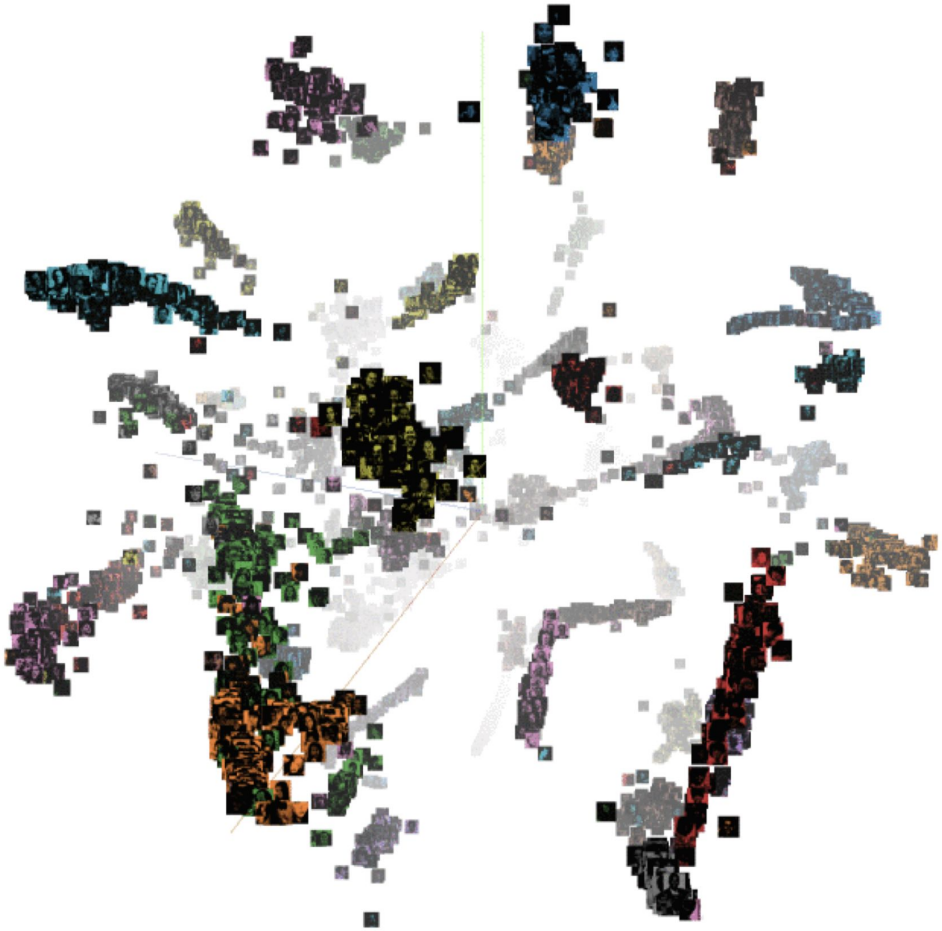


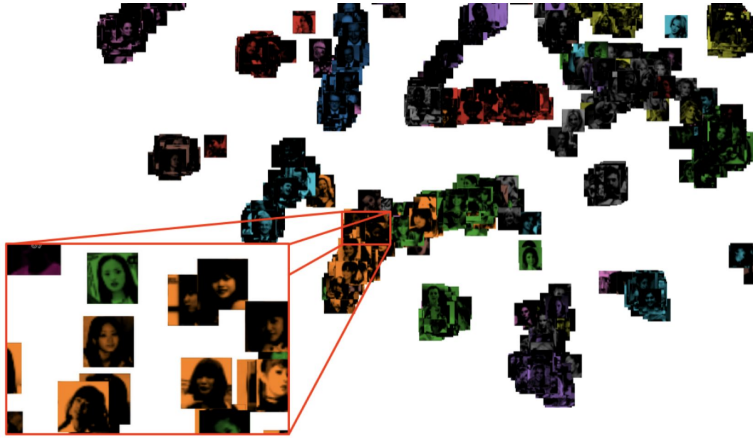
Kartēšanas (*Embedding*) izveidošana



Kā strādā FaceID?







Carl

Sideshow Bob

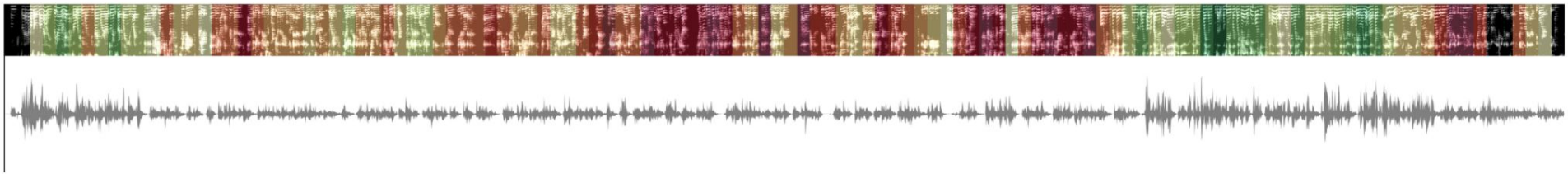
Lisa

Selma

Kids/Boys

Tieši tāpat ar audio datiem!

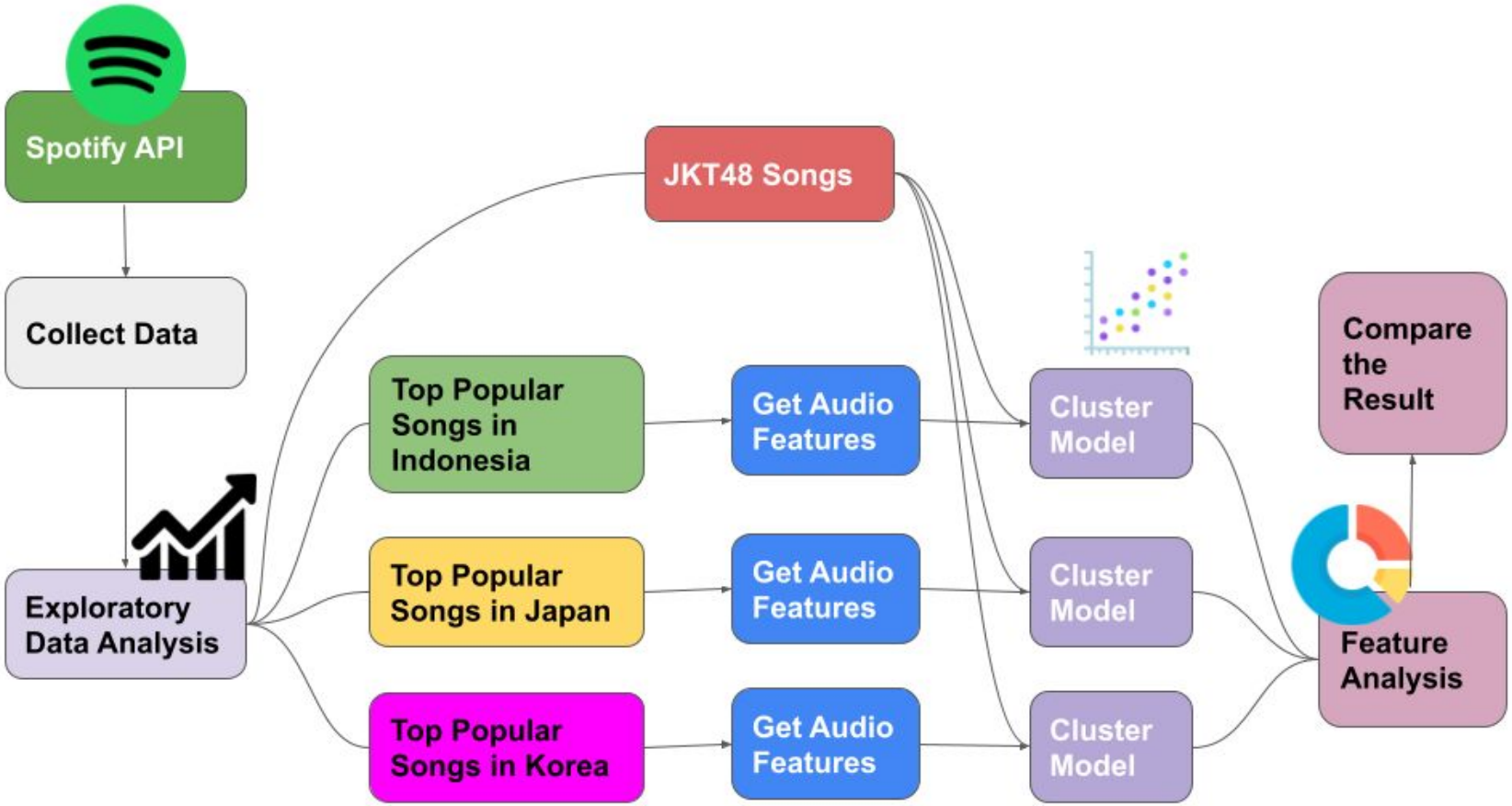
asya



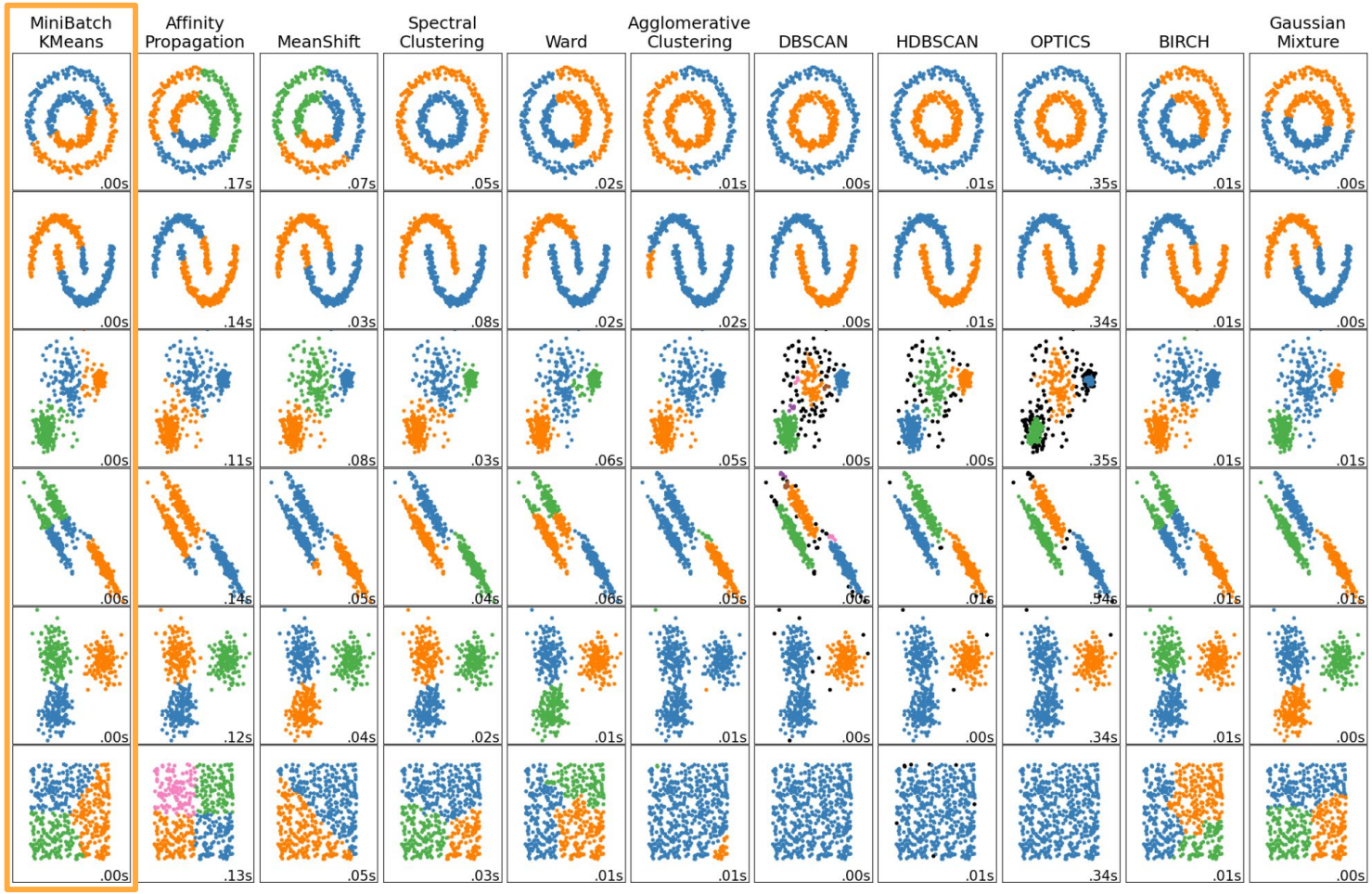
* Green frames represent target speaker. Red frames are furthest away from target speaker.

▶ PLAY ■ STOP

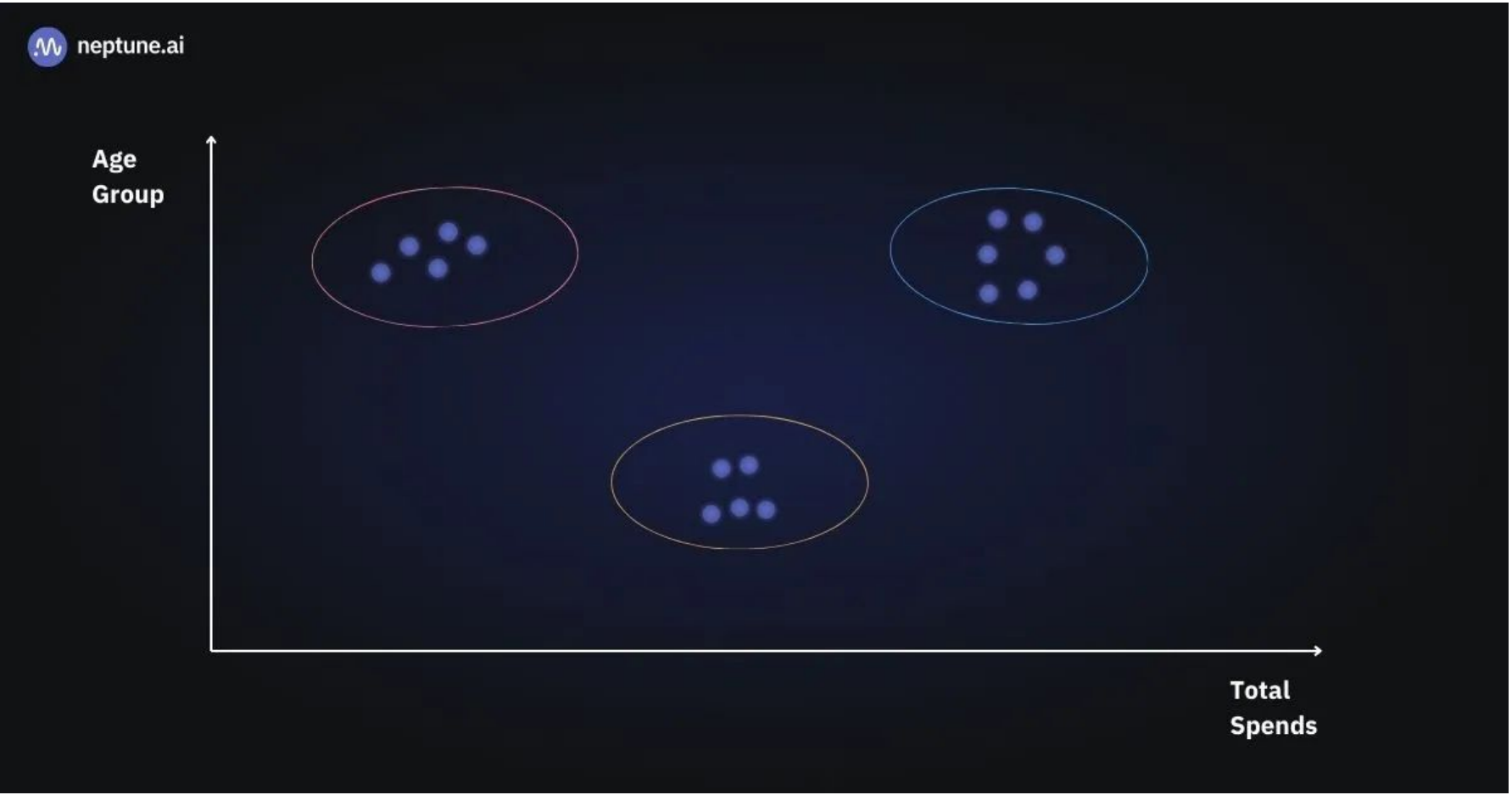
Ko dara Spotify?



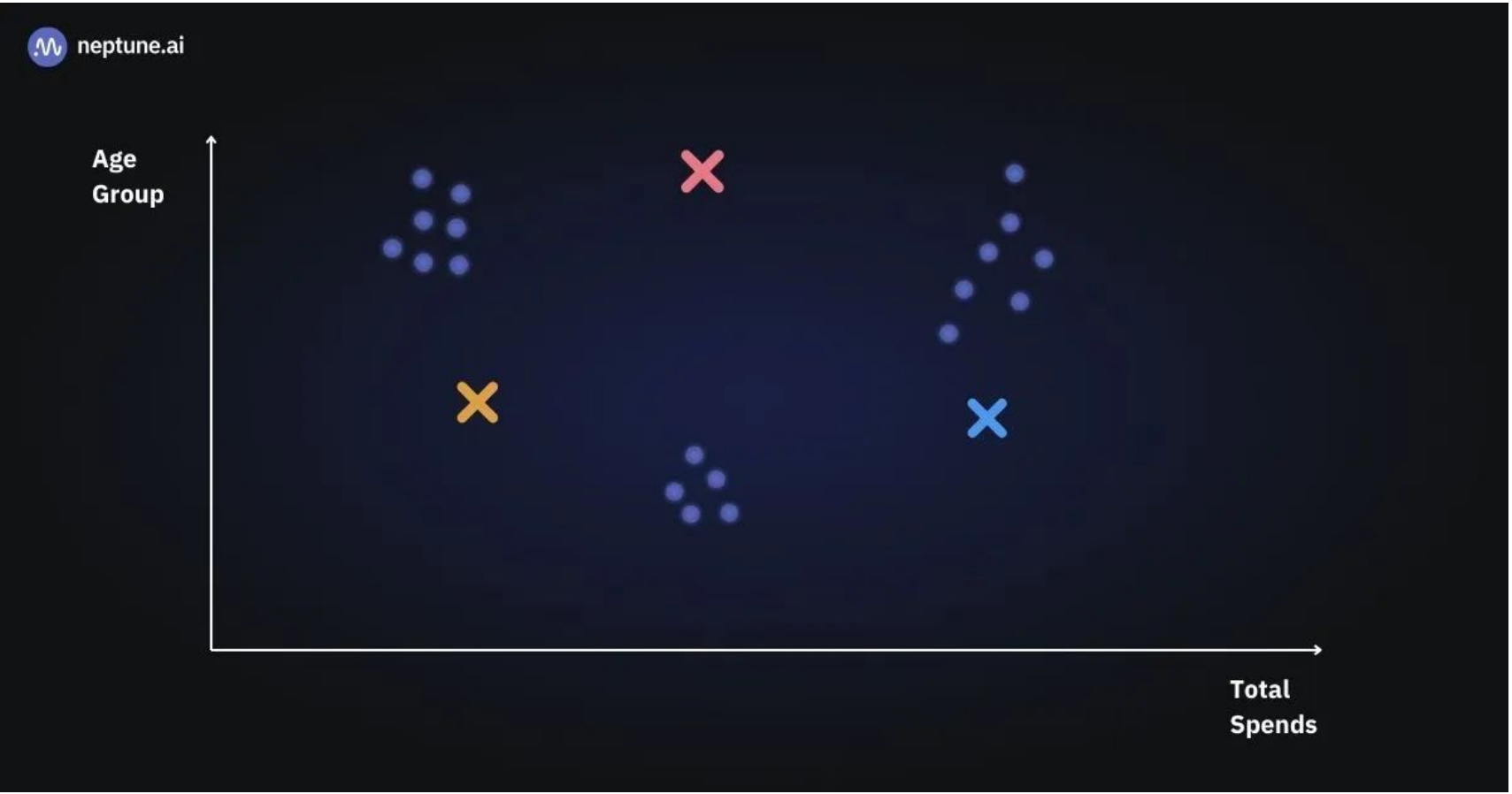
Ir daudz dažādi klasterizācijas algoritmi



K-means algoritma darbība



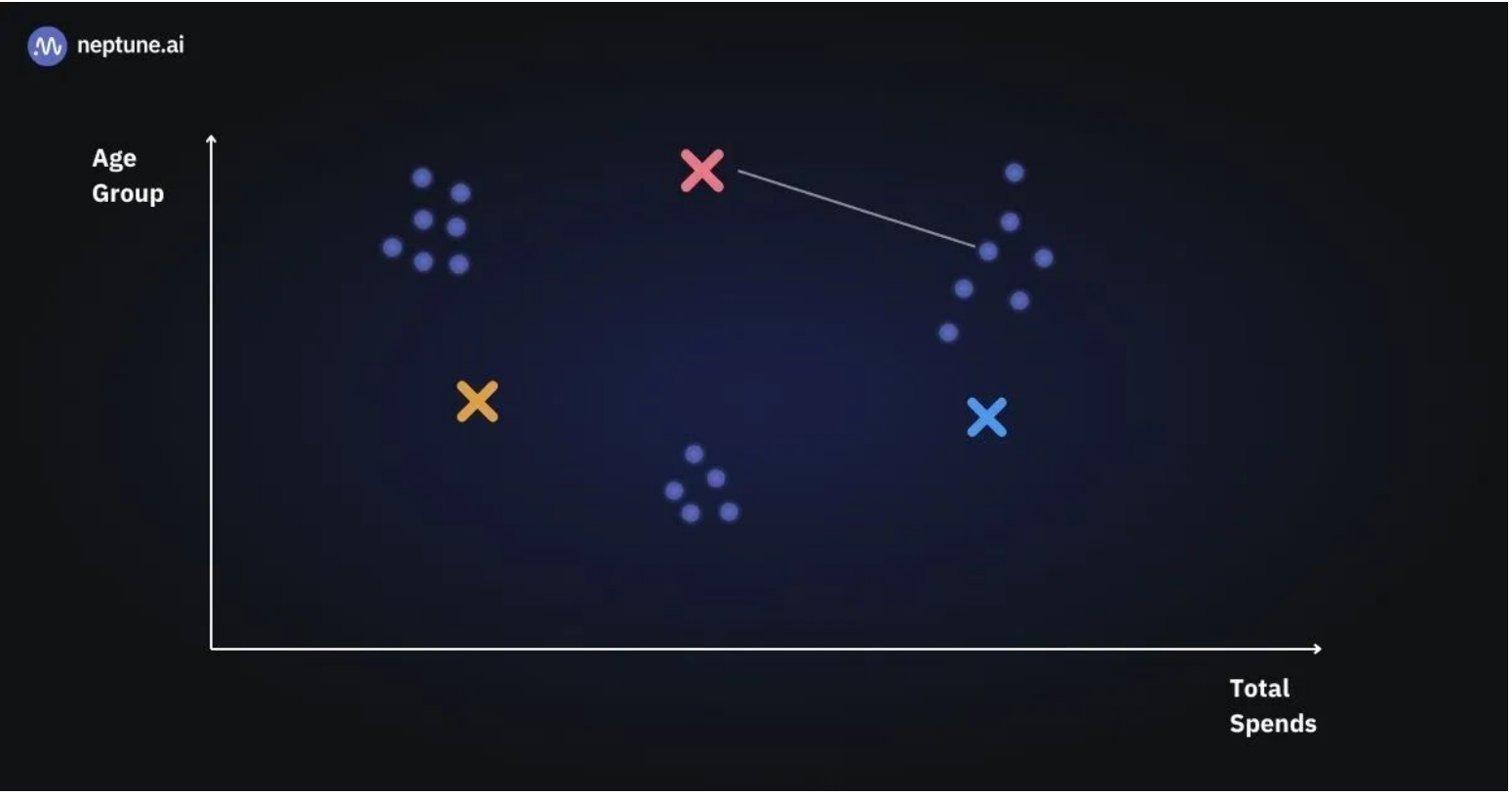
1. Izvēlamies klasteru skaitu
2. Nejauši inicializējam klasteru centrus



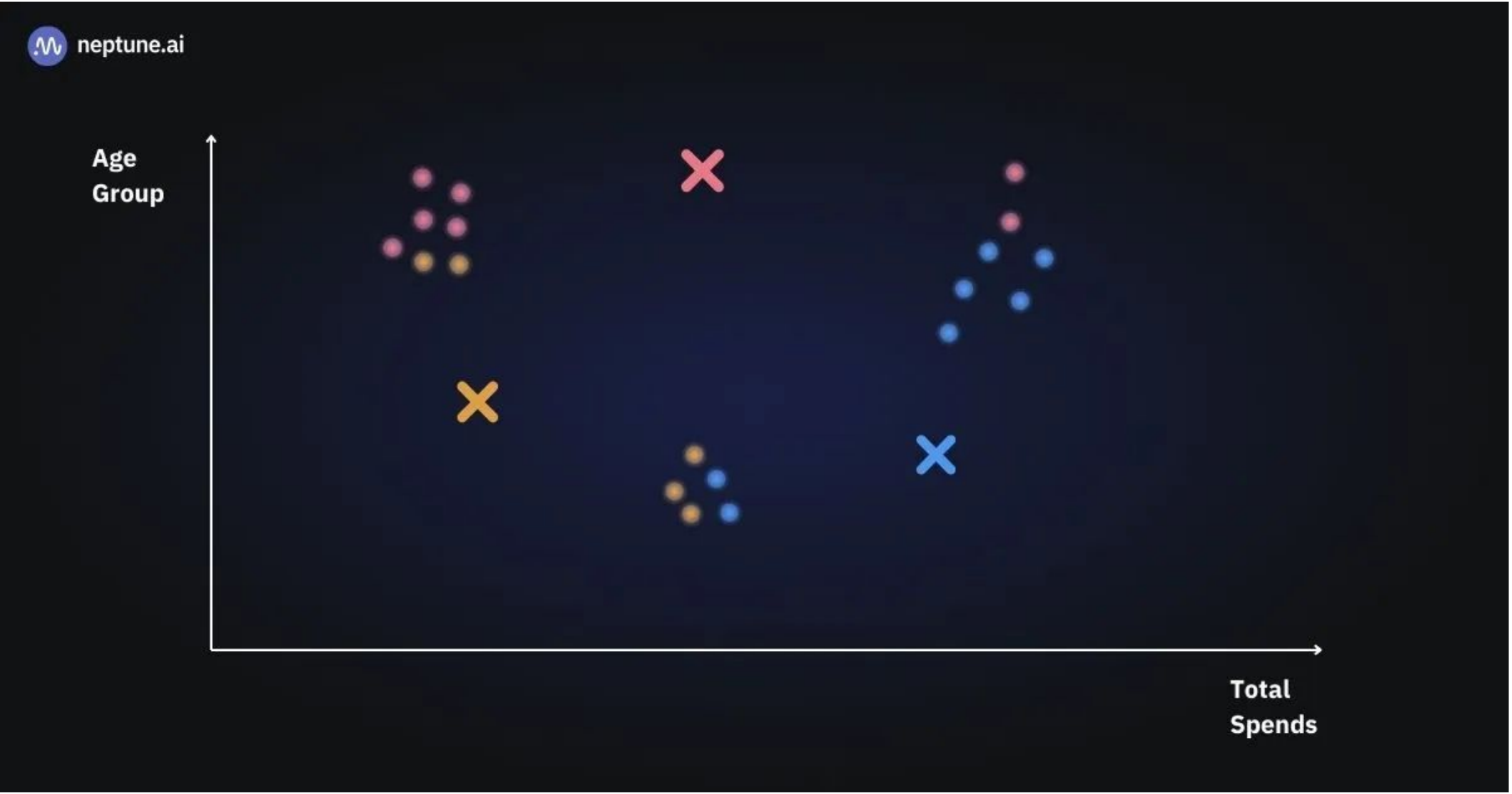
3. Katram datu punktam piešķiram vienu no centriem (tuvāko pēc kādas metrikas)

Bieži izmanto eiklīda distanci! (Ir daudz citu)

$$d(x, y) = \sqrt{\sum_{i=1}^{n} (x_i - y_i)^2}$$



Iepriekšējā soļa rezultāts:

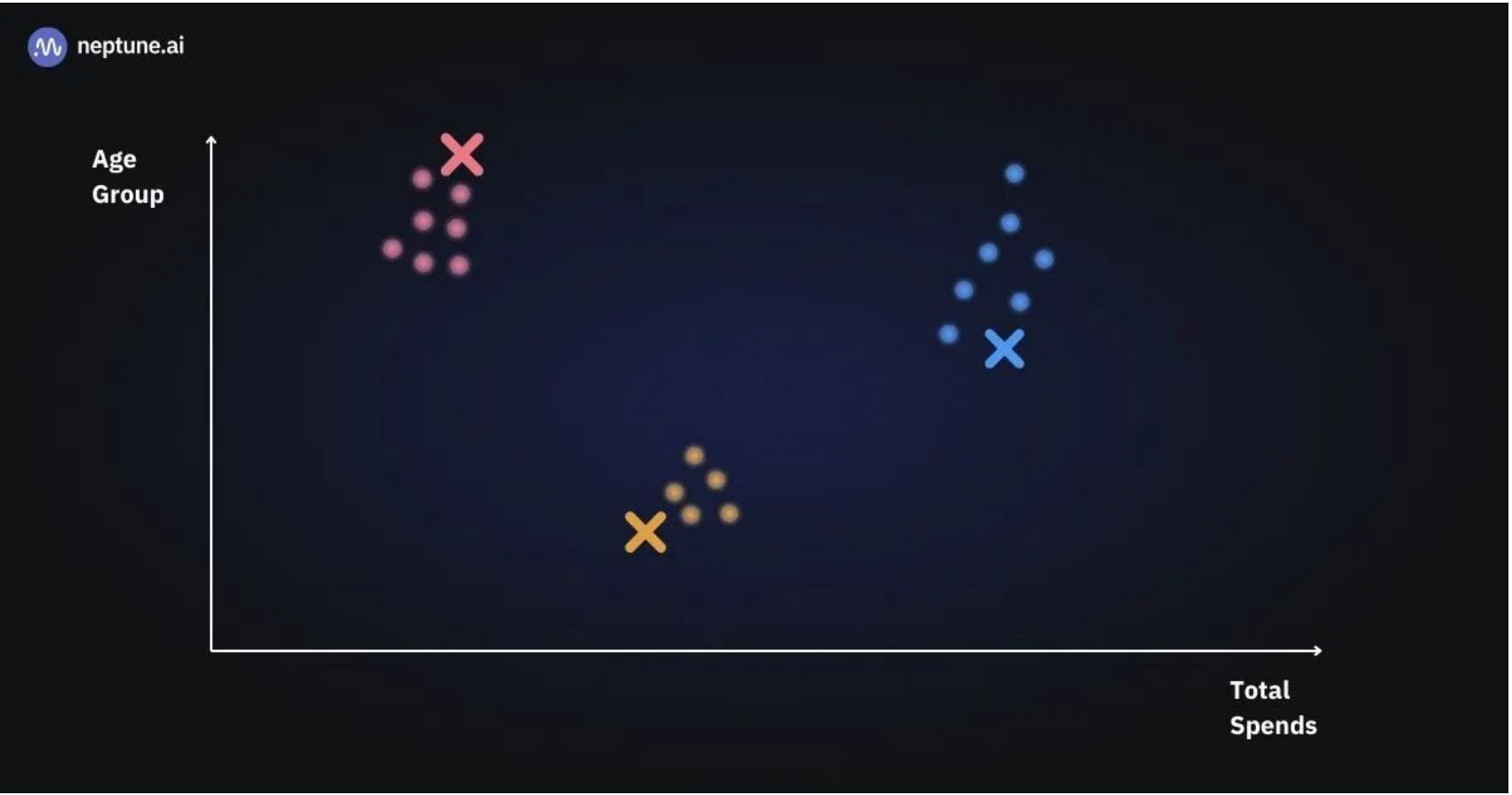


4. Aprēķinam jaunus klasteru centrus

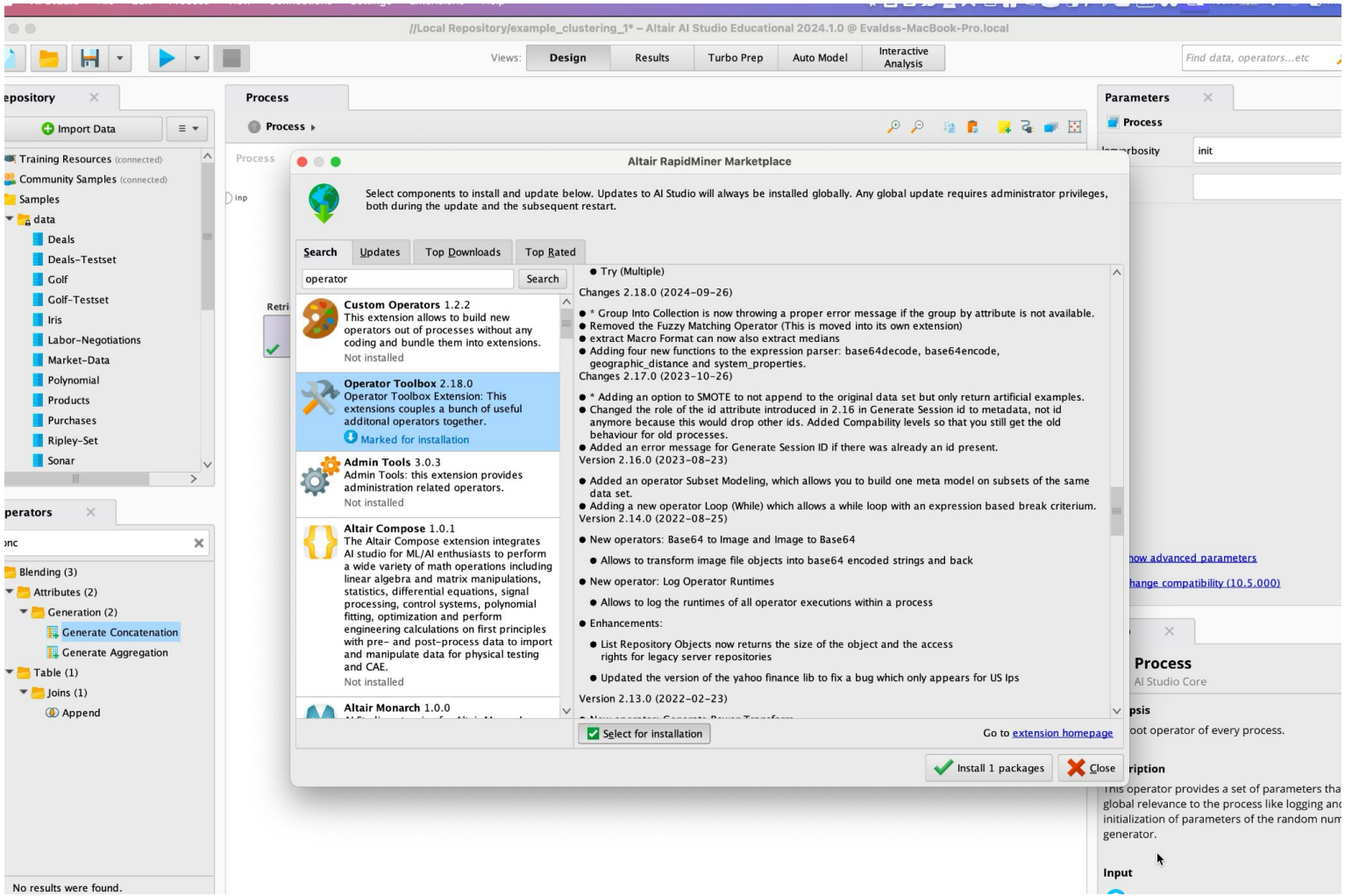
To daram izrēķinot katra izveidotā klastera viduspunktu

$$Ci = \frac{1}{|Ni|} \sum xi$$

5. Atkārtojam 3. un 4. soli vairākas reizes



Praktiskais darbs


//Local Repository/example_clustering_1* – Altair AI Studio Educational 2024.1.0 @ Evaldss-MacBook-Pro.local
Views:
Design
Results
Turbo Prep
Auto Model
Interactive Analysis
Find data, operators...etc
Repository
Import Data
Training Resources (connected)
Community Samples (connected)
Samples
data
Deals
Deals-Testset
Golf
Golf-Testset
Iris
Labor-Negotiations
Market-Data
Polynomial
Products
Purchases
Ripley-Set
Sonar
Operators
Blending (3)
Attributes (2)
Generation (2)
Generate Concatenation
Generate Aggregation
Table (1)
Joins (1)
Append
No results were found.
Process
Process
Altair RapidMiner Marketplace
Select components to install and update below. Updates to AI Studio will always be installed globally. Any global update requires administrator privileges, both during the update and the subsequent restart.
Search
Updates
Top Downloads
Top Rated
operator
Search
Custom Operators 1.2.2
This extension allows to build new operators out of processes without any coding and bundle them into extensions.
Not installed
Operator Toolbox 2.18.0
Operator Toolbox Extension: This extensions couples a bunch of useful additonal operators together.
Marked for installation
Admin Tools 3.0.3
Admin Tools: this extension provides administration related operators.
Not installed
Altair Compose 1.0.1
The Altair Compose extension integrates AI studio for ML/AI enthusiasts to perform a wide variety of math operations including linear algebra and matrix manipulations, statistics, differential equations, signal processing, control systems, polynomial fitting, optimization and perform engineering calculations on first principles with pre- and post-process data to import and manipulate data for physical testing and CAE.
Not installed
Altair Monarch 1.0.0
● Try (Multiple)
Changes 2.18.0 (2024-09-26)
● * Group Into Collection is now throwing a proper error message if the group by attribute is not available.
● Removed the Fuzzy Matching Operator (This is moved into its own extension)
● extract Macro Format can now also extract medians
● Adding four new functions to the expression parser: base64decode, base64encode, geographic_distance and system_properties.
Changes 2.17.0 (2023-10-26)
● * Adding an option to SMOTE to not append to the original data set but only return artificial examples.
● Changed the role of the id attribute introduced in 2.16 in Generate Session id to metadata, not id anymore because this would drop other ids. Added Compability levels so that you still get the old behaviour for old processes.
● Added an error message for Generate Session ID if there was already an id present.
Version 2.16.0 (2023-08-23)
● Added an operator Subset Modeling, which allows you to build one meta model on subsets of the same data set.
● Adding a new operator Loop (While) which allows a while loop with an expression based break criterium.
Version 2.14.0 (2022-08-25)
● New operators: Base64 to Image and Image to Base64
● Allows to transform image file objects into base64 encoded strings and back
● New operator: Log Operator Runtimes
● Allows to log the runtimes of all operator executions within a process
● Enhancements:
● List Repository Objects now returns the size of the object and the access rights for legacy server repositories
● Updated the version of the yahoo finance lib to fix a bug which only appears for US Ips
Version 2.13.0 (2022-02-23)
Select for installation
Go to extension homepage
Install 1 packages
Close
Parameters
Process
init
how advanced parameters
hange compatibility (10.5.000)
Process
AI Studio Core
psis
oot operator of every process.
ription
This operator provides a set of parameters tha
global relevance to the process like logging and
initialization of parameters of the random num
generator.
Input

Uzdevums

3.2. Implementēt dimensiju samazināšanas un klasterizācijas modeli

Īstenot klasterizācijas modeli dzīvnieku datu kopai, krāsot klasterus pēc "sugas". Rezultātus attēlot, izmantojot PCA, t-SNE vai UMAP 2D. Izmantot Altair RapidMiner AI Studio, Knime, Weka vai BigML (tiešsaistē). Pārliecinieties, ka funkcijas tiek iestatītas ar pareiziem datu tipiem. Iesniedziet klasteru rezultātu ekrānuzņēmumus.

Lēmumu koki

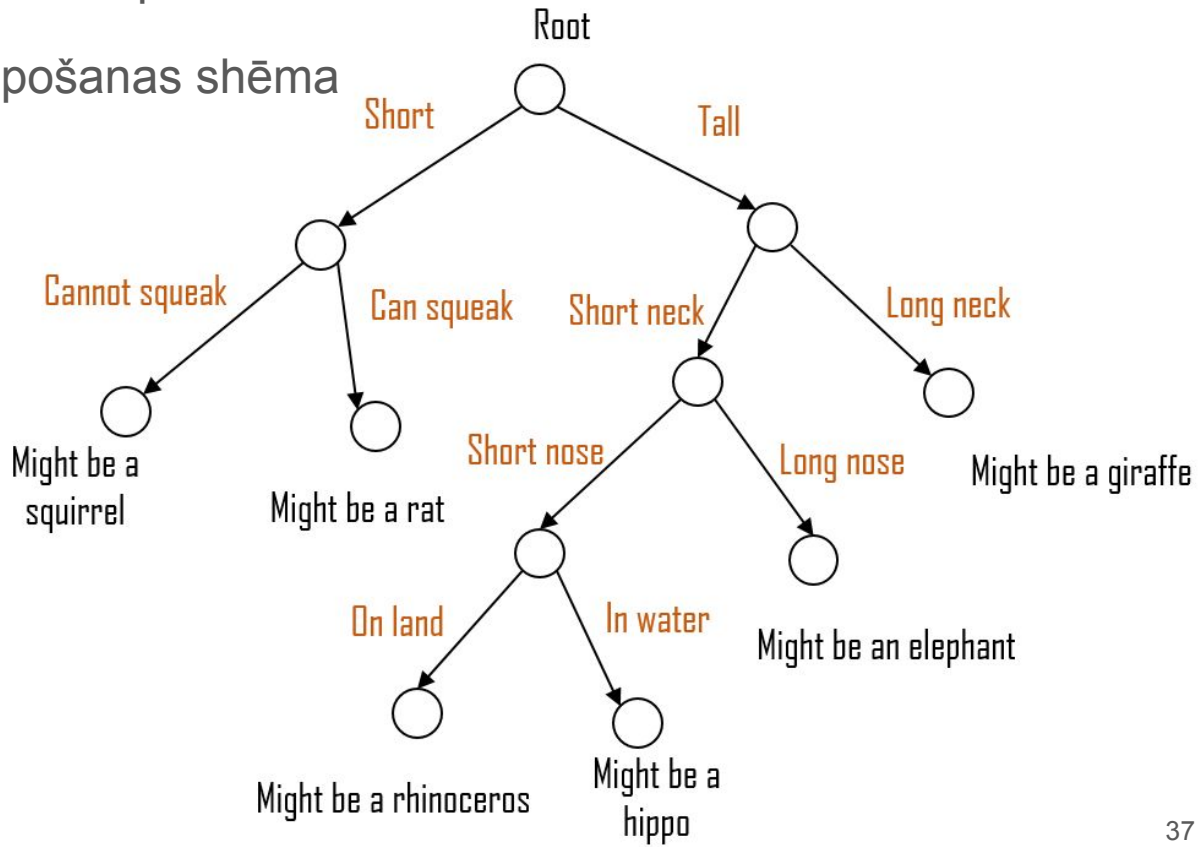
Decision Trees

Kas ir lēmumu koki?

Loģisko jautājumu secība, kas noved pie lēmuma

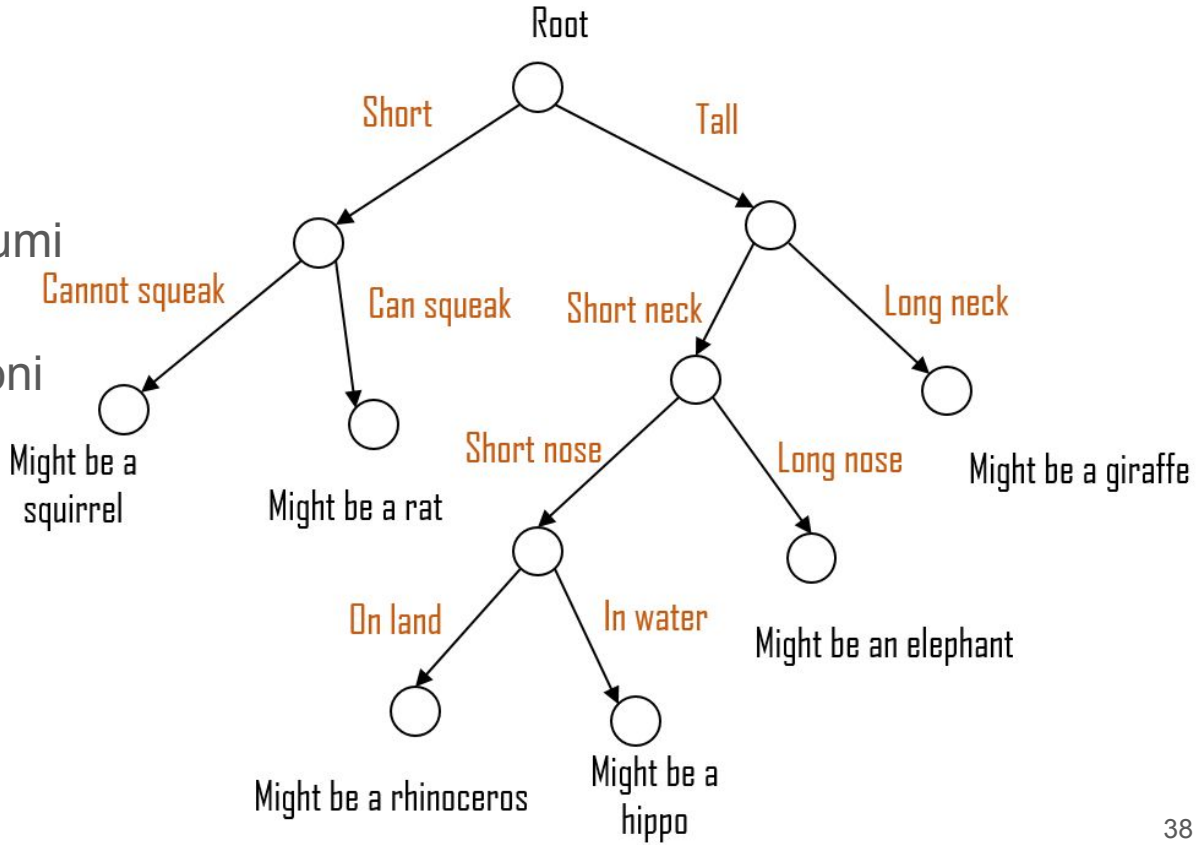
Ikdienas piemērs: Klienta apkalpošanas shēma

- "Vai klients ir reģistrēts?"
- "Vai pirkums > 100€?"
- "Vai ir bijušas sūdzības?"



Lēmumu koka struktūra

- Sakne (Root) - sākuma jautājums
- Mezgli (Nodes) - papildus jautājumi
- Lapas (Leaves) - gala lēmumi
- Zari (Branches) - atbildes "Jā/Nē" vai vērtību diapazoni



Klasterizācija vs Kategorizācija

Klasterizācija:

- Grupē līdzīgos objektus
- Nezināmas grupas
- Balstās uz attālumiem
- "Atrod struktūru datos"

Kategorizācija:

- Piešķir kategoriju
- Zināmas grupas
- Balstās uz pazīmēm
- "Mācās no piemēriem"

Klientu segmentēšana pēc uzvedības

Klienta riska līmeņa noteikšana

Kur izmanto lēmumu kokus?

Klientu analīze:

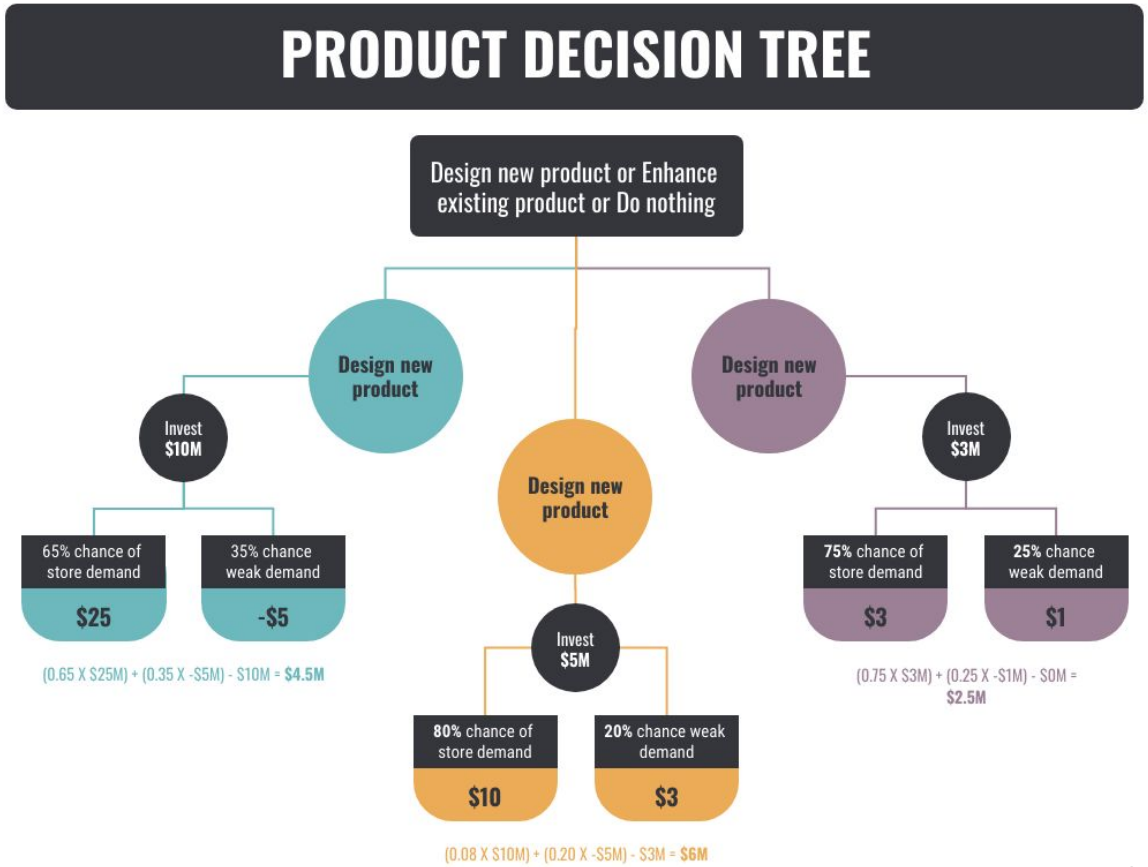
- Aizejošo klientu prognoze
- Kredītriska novērtējums
- Klientu segmentācija

Operāciju optimizācija:

- Kvalitātes kontrole
- Procesu automatizācija
- Resursu plānošana

Mārketings:

- Kampaņu efektivitāte
- Mērķauditorijas atlase
- Kanālu optimizācija



Priekšrocības un trūkumi

✅ Priekšrocības:

- Viegli saprotams rezultāts
- Labi vizualizējami
- Skaidri lēmumu kritēriji
- Var kombinēt dažādu tipu datus

❌ Trūkumi:

- Var pārmācīties (overfit)
- Nestabili pret izmaiņām
- Grūti modelēt sarežģītas attiecības datos
- Neprecīzi ar ļoti lielām datu kopām

Populārākie algoritmi, lai veidotu lēmumu kokus

ID3:

- Pamata algoritms
- Strādā ar datiem kategorijās
- Izmanto entropiju

C4.5:

- ID3 uzlabojums
- Strādā arī ar skaitliskiem datiem
- Labāka trūkstošo vērtību apstrāde

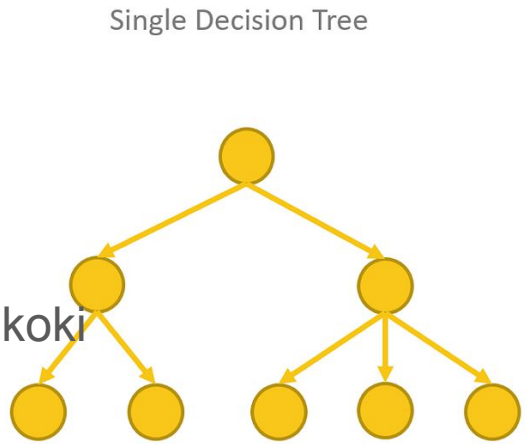
CART:

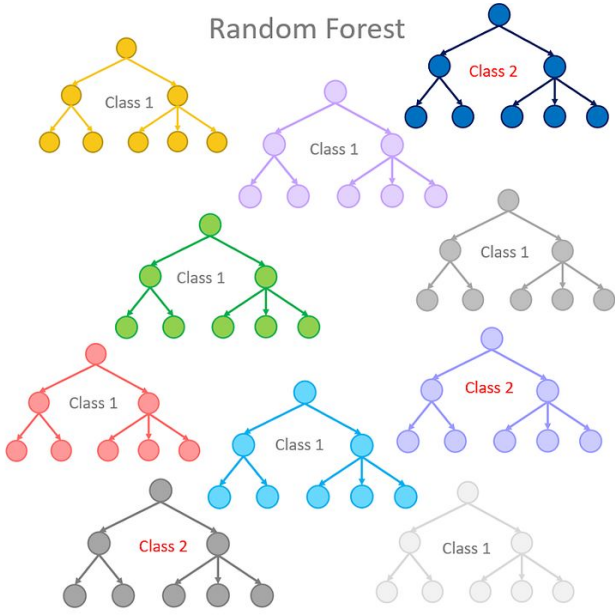
- Binārā dalīšana
- Labi strādā ar skaitliskiem datiem
- Populārs modernajās bibliotēkās

Features	*ID3*	*C4.5*	*CART*
Type of data	Categorical	Continuous and Categorical	continuous and nominal attributes data
Speed	Low	Faster than ID3	Average
Boosting	Not supported	Not supported	Supported
Pruning	No	Pre-pruning	Post pruning
Missing Values	Can't deal with	Can't deal with	Can deal with
Formula	Use information entropy and information Gain	Use split info and gain ratio	Use Gini diversity index

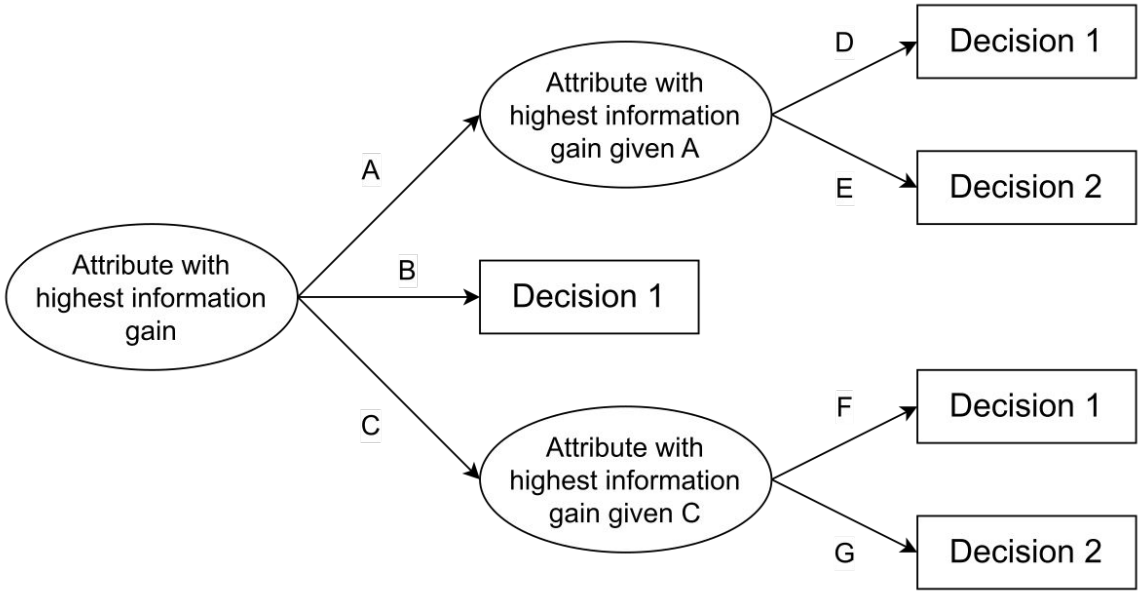
Citas kategorizācijas metodes

1. **Random Forest**:
   - Daudzu lēmumu koku "mežs" (apvienojums)
   - Labāka precizitāte
   - Stabilāki rezultāti
2. **XGBoost**:
   - Gradientu pastiprinošie koki
   - Ātrāks un precīzāks
   - Industrijas standarts (State-of-the-Art)





ID3 algoritma darbība



- Mērķis: Izveidot visvienkāršāko koku
- Izmanto entropiju informācijas mērīšanai
- Izvēlas labāko dalīšanas pazīmi katrā solī

Entropija un informācijas pieaugums

- Entropija = nesakārtotības mērs
- Information Gain = entropijas samazinājums
- Mērķis: maksimizēt Information Gain

$$Entropy = \sum_{i=1}^{C} -p_i * \log_2(p_i)$$

$$Gain(T, X) = Entropy(T) - Entropy(T, X)$$

G(PlayGolf, Outlook) = **E**(PlayGolf) – **E**(PlayGolf, Outlook)

= 0.940 – 0.693 = 0.247

Entropija un informācijas pieaugums aprēķina piemērs

Laikapstākļi	Spēlēt tenisu?
Saulains	Jā
Saulains	Nē
Saulains	Nē
Mākoņains	Jā
Mākoņains	Jā
Lietus	Nē
Lietus	Nē
Lietus	Nē

Sākotnējā entropija (visa datu kopa)

- Kopā: 8 gadījumi
- "Jā": 3 gadījumi
- "Nē": 5 gadījumi

Entropy = $-P(Jā)\times\log_2(P(Jā)) - P(Nē)\times\log_2(P(Nē))$

- P(Jā) = 3/8 = 0.375
- P(Nē) = 5/8 = 0.625

$E = -(0.375 \times \log_2(0.375)) - (0.625 \times \log_2(0.625))$
$E = -(0.375 \times (-1.415)) - (0.625 \times (-0.678))$
$E = 0.530 + 0.424$
$E = 0.954$ biti

Entropija un informācijas pieaugums aprēķina piemērs

Laikapstākļi	Spēlēt tenisu?
Saulains	Jā
Saulains	Nē
Saulains	Nē
Mākoņains	Jā
Mākoņains	Jā
Lietus	Nē
Lietus	Nē
Lietus	Nē

Entropija katrai laika kategorijai

Saulains (3 gadījumi)

- "Jā": 1
- "Nē": 2
- $E(\text{Saulains}) = -(1/3 \times \log_2(1/3)) - (2/3 \times \log_2(2/3))$
- $E(\text{Saulains}) = 0.918$ biti

Mākoņains (2 gadījumi)

- "Jā": 2
- "Nē": 0
- $E(\text{Mākoņains}) = -(1 \times \log_2(1)) - (0 \times \log_2(0))$
- $E(\text{Mākoņains}) = 0$ biti

Lietus (3 gadījumi)

- "Jā": 0
- "Nē": 3
- $E(\text{Lietus}) = -(0 \times \log_2(0)) - (1 \times \log_2(1))$
- $E(\text{Lietus}) = 0$ biti

Entropija un informācijas pieaugums aprēķina piemērs

Laikapstākļi	Spēlēt tenisu?
Saulains	Jā
Saulains	Nē
Saulains	Nē
Mākoņains	Jā
Mākoņains	Jā
Lietus	Nē
Lietus	Nē
Lietus	Nē

Information Gain aprēķins

IG = Sākotnējā_Entropija - Σ(P(kategorija) × E(kategorija))

- P(Saulains) = 3/8
- P(Mākoņains) = 2/8
- P(Lietus) = 3/8

IG = 0.954 - (3/8 × 0.918 + 2/8 × 0 + 3/8 × 0)
IG = 0.954 - 0.344
IG = 0.610 biti

Secinājums

- Information Gain = 0.610 biti
- Šis nozīmē, ka, zinot laika apstākļus, mēs samazinām nenoteiktību par 0.610 bitiem
- Jo lielāks Information Gain, jo labāk pazīme sadala datus

ID3 soli pa solim

Solis: Sākotnējā entropija

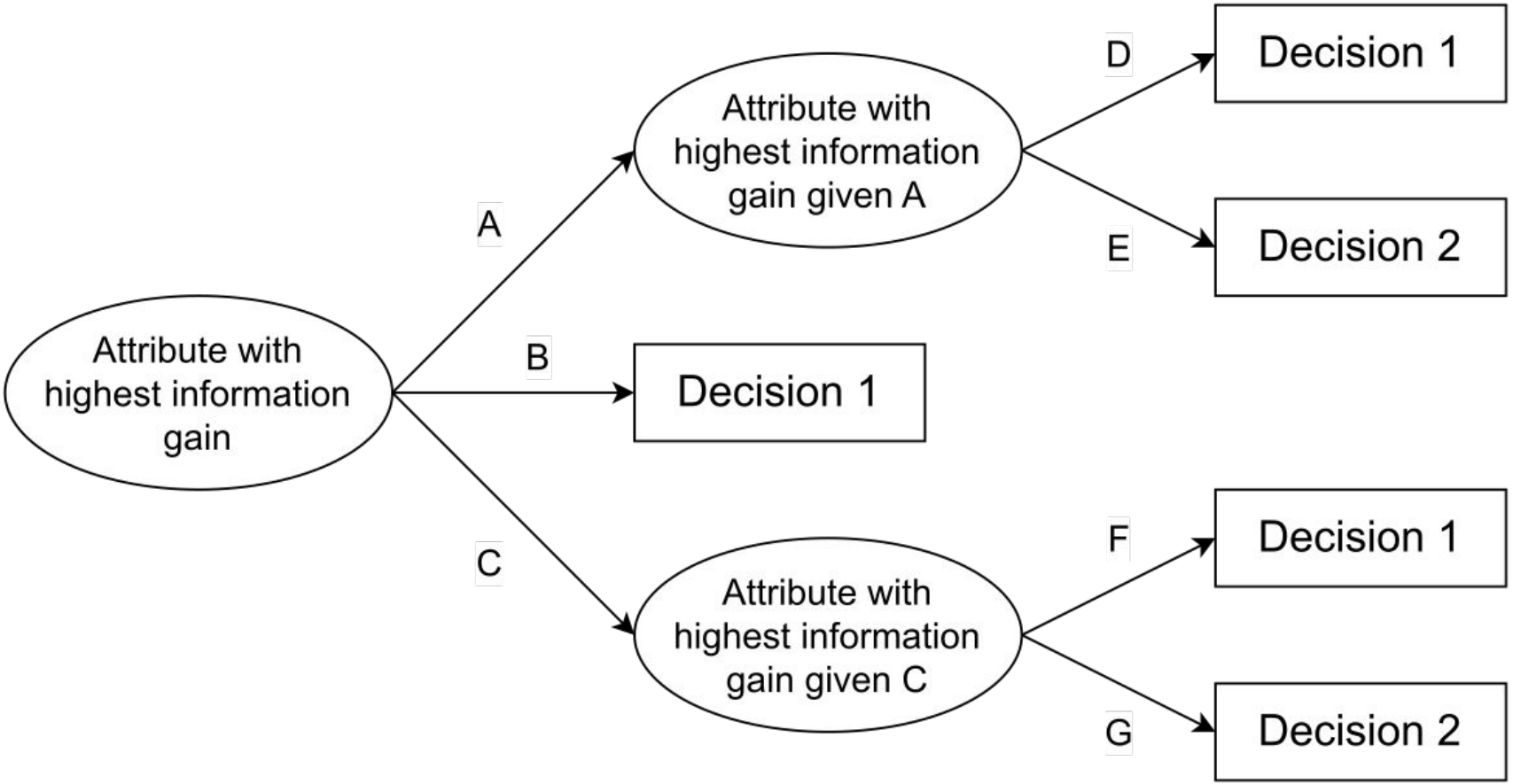
- Aprēķināt klašu proporcijas
- Aprēķināt kopējo entropiju

Solis: Pazīmju novērtēšana

- Aprēķināt Information Gain katrai pazīmei
- Izvēlēties labāko dalīšanas pazīmi

Solis: Koka būvēšana

- Sadalīt datus pēc izvēlētās pazīmes
- Atkārtot procesu katram zaram


Attribute with highest information gain
A
B
C
Attribute with highest information gain given A
Decision 1
Attribute with highest information gain given C
D
E
F
G
Decision 1
Decision 2
Decision 1
Decision 2

Praktiskais darbs

Uzdevums

3.3. Implementēt lēmumu pieņemšanas koka modeli

Īstenot lēmumu pieņemšanas koka modeli pārdošanas datiem, lai prognozētu "Customer Type". Izmantot Altair RapidMiner AI Studio, Knime, Weka vai BigML (tiešsaistē). Pārliecinieties, ka funkcijas tiek iestatītas ar pareiziem datu tipiem. Iesniedziet lēmumu koka rezultātu ekrānuzņēmumus.

Labā prakse: Kad izmantot lēmumu kokus?

Ideāli piemēroti kad:

- Nepieciešami skaidri, interpretējami rezultāti
  - Kredīta piešķiršana (jāvar izskaidrot klientam)
  - Medicīniskā diagnostika (jāvar pamatot ārstam)
  - Riska novērtējums (jāvar izskaidrot vadībai)
- Dati ir jaukta tipa
  - Gan skaitliski (vecums, summa)
  - Gan kategoriāli (dzimums, pilsēta)
  - Gan bināri (jā/nē)
- Vajadzīgi ātri lēmumi reālajā laikā
  - Klientu apkalpošanas sistēmas
  - Automatizētas atbildes
  - Reāllaika cenu noteikšana

Nav ieteicami kad:

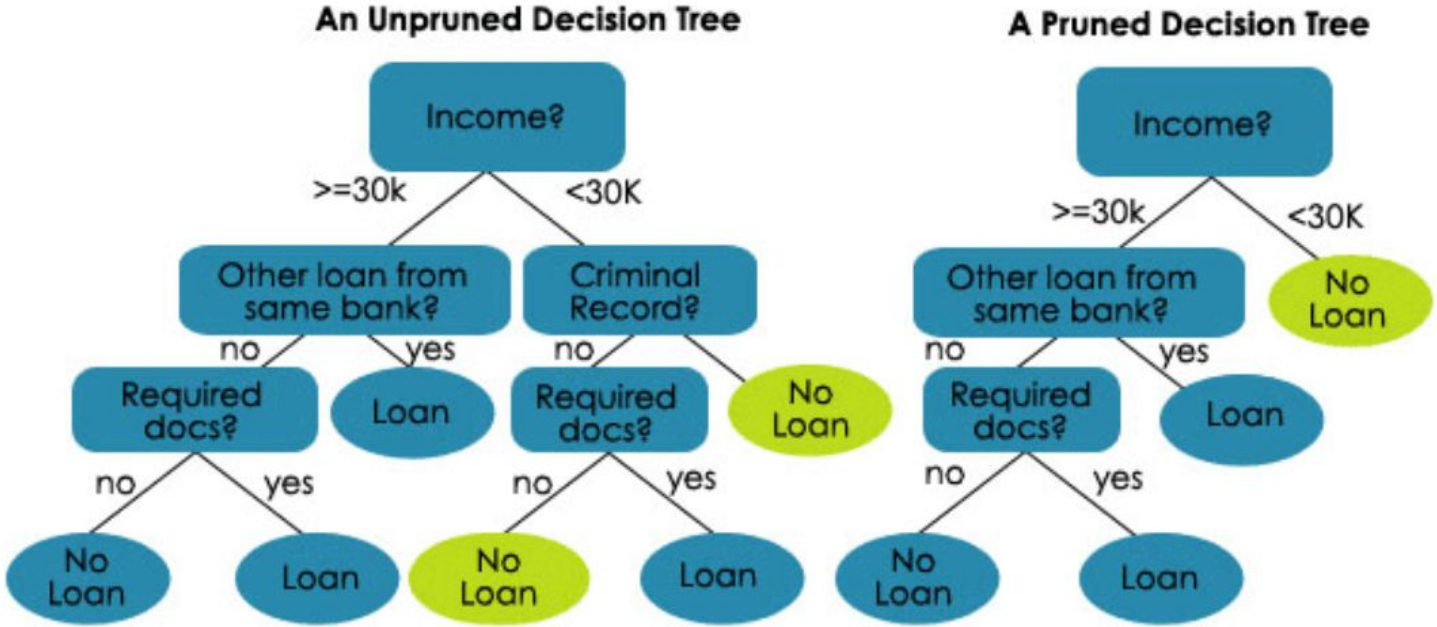
- Nepieciešama ļoti augsta precizitāte
- Dati ir ļoti trokšņaini
- Attiecības datos ir ļoti kompleksas
- Datu kopa ir ļoti maza

Labā prakse: Kā izvairīties no pārmācīšanās (Overfitting)?

1. Koka dziļuma ierobežošana
   - Noteikt maksimālo dziļumu
   - Ierobežot minimālo instanču skaitu lapā
   - Ierobežot minimālo instanču skaitu dalīšanai
2. Apgriešana (Pruning)
   - Pre-pruning: apturēt augšanu pēc kritērijiem
   - Post-pruning: apgriezt pēc apmācības
3. Ansambļu metodes
   - **Random Forest**
   - **Bagging**
   - **Boosting**

Lēmuma koka apgriešana (*Pruning*)

- 🎯 Novērš pārmācīšanos
- 🚀 Paātrina lēmumu pieņemšanu
- 📊 Uzlabo vispārināšanas spējas
- 🧮 Samazina kļūdas uz jauniem datiem



Labā prakse: Kā validēt rezultātus?

Kvantitatīvā validācija:

1. Precizitātes mērījumi
   - Accuracy (kopējā precizitāte)
   - Precision (cik precīzi pozitīvie)
   - Recall (cik daudz pozitīvo atrasts)
   - F1 score (precision un recall balance)
2. Kļūdu analīze
   - Confusion matrix
   - ROC curve
   - Precision-recall curve
3. Stabilitātes testi
   - K-fold cross validation
   - Train-test splits
   - Time-based validation

Kvalitatīvā validācija:

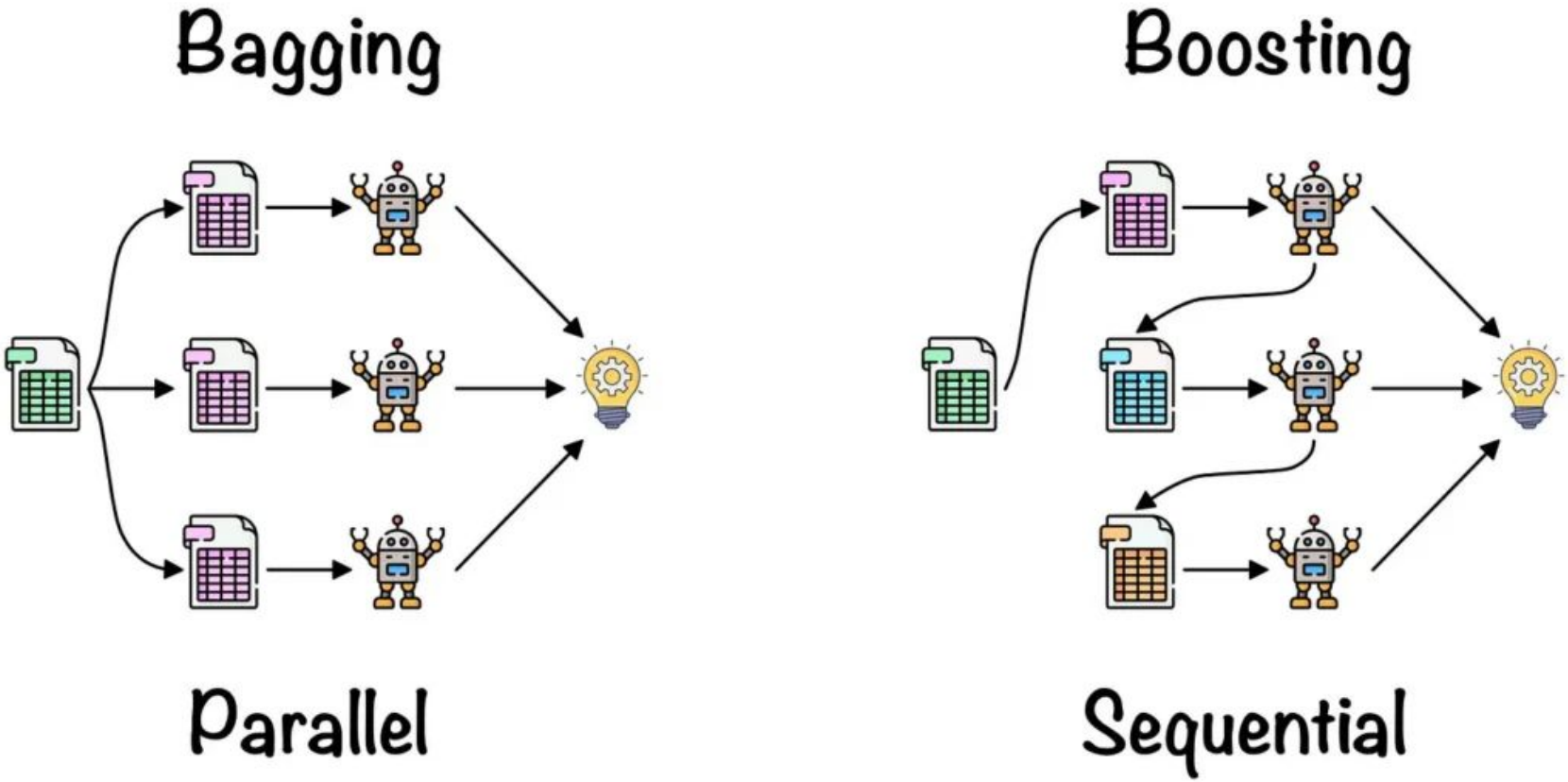
1. Biznesa loģikas pārbaude
   - Vai lēmumi ir loģiski?
   - Vai rezultāti atbilst eksperta zināšanām?
   - Vai ir neparedzētas blakusparādības?
2. Feature importance analīze
   - Kuri faktori ir vissvarīgākie?
   - Vai tas atbilst biznesa izpratnei?
3. Edge case analīze
   - Kā modelis uzvedas ekstrēmos gadījumos?
   - Vai ir neparedzētas uzvedības?

Praktiskās pārbaudes:

1. A/B testēšana
   - Mazā mērogā
   - Kontrolētā vidē
   - Ar skaidriem mērījumiem
2. Monitorings produkcijā
   - Regulāra rezultātu pārbaude
   - Alertu sistēma anomālijām
   - Periodiska pārapmācība

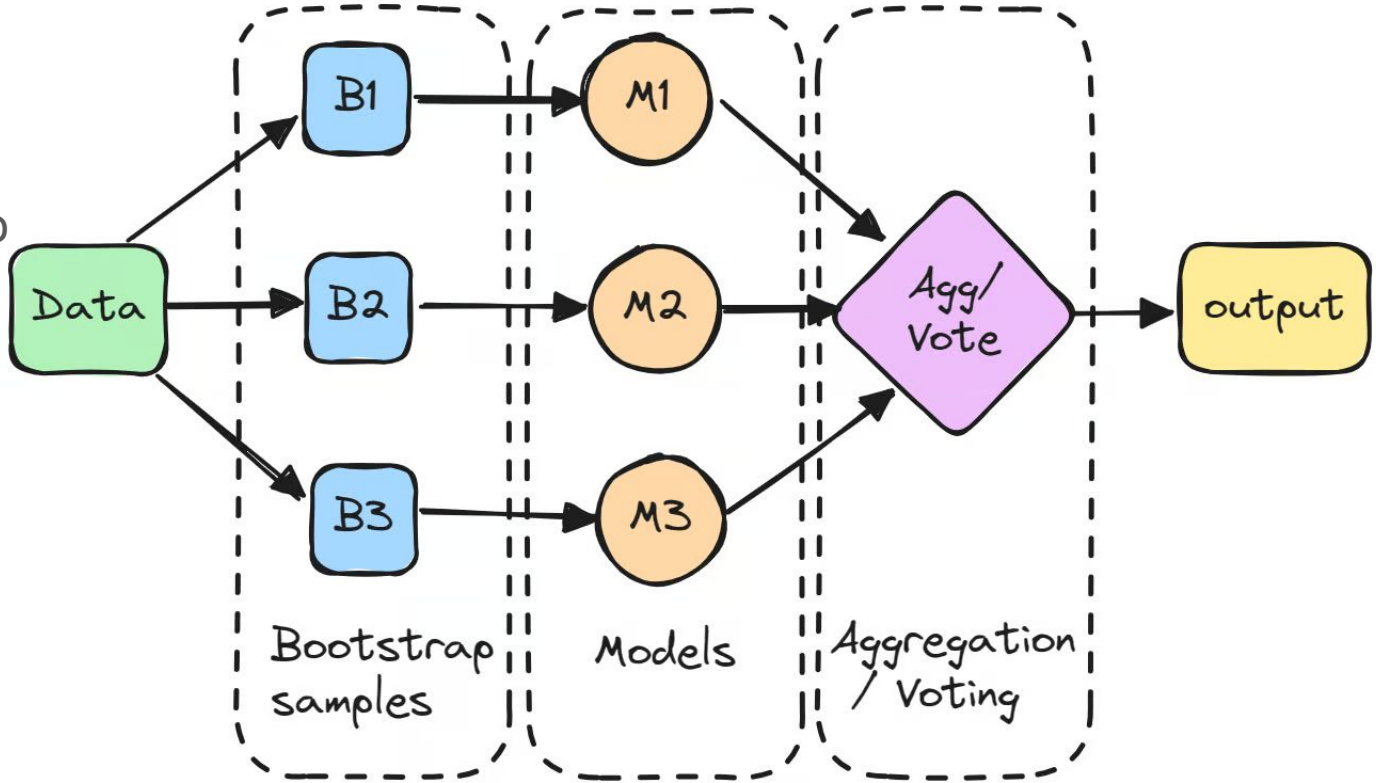
Boosting and Bagging
Random Forests and XGBoost

Boosting and Bagging



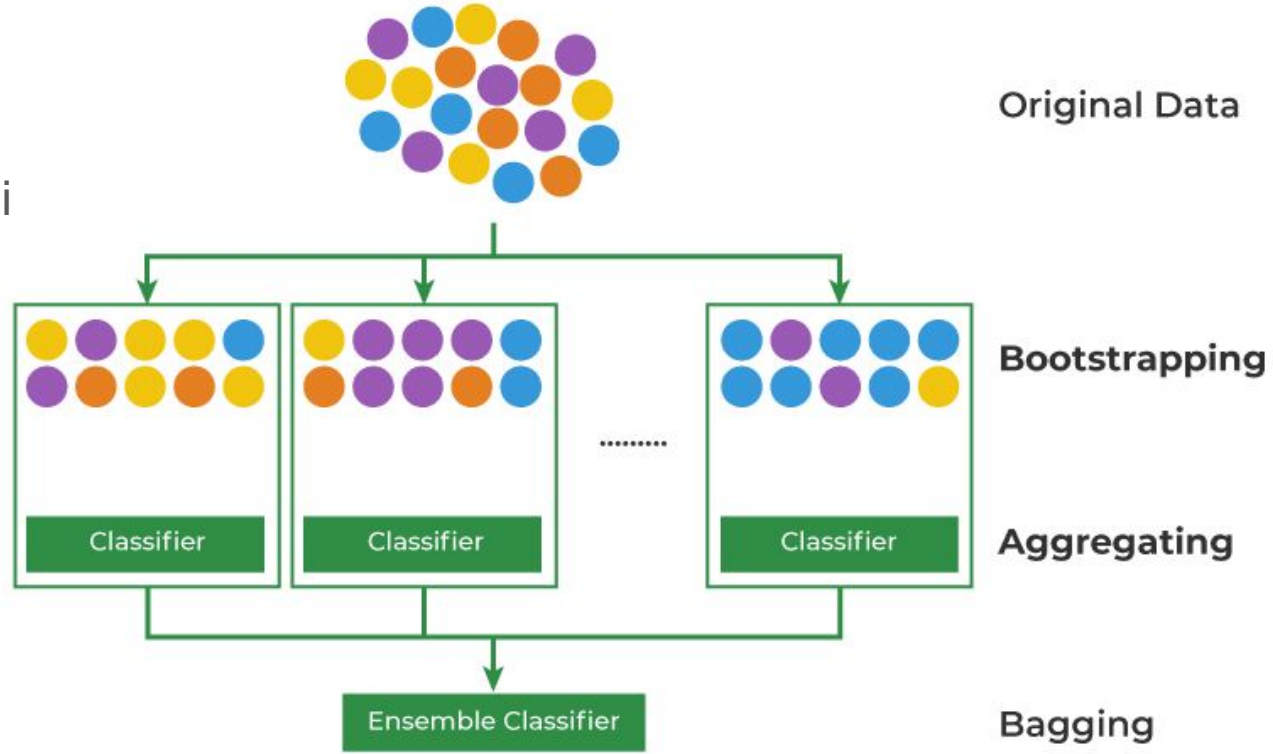
Bagging

- Bootstrap Aggregating = Bagging
- "Daudzu viedokļu apkopošana"
- Vairāki modeļi balso par gala rezultātu
- Līdzīgi kā ekspertu komisija



Bagging darbības princips

- Izveidojam vairākas datu kopijas
- Katrai kopijai apmācām savu modeli
- Apkopojam visu modeļu rezultātus
- Pieņemam gala lēmumu balstoties uz vairākumu



Bagging piemērs dzīvē

🏠 Mājas pirkšana:

- Konsultējamies ar nekustamo īpašumu aģentu
- Runājam ar banku
- Prasām būvniekam
- Aprunājamies ar draugiem

➡️ Pieņemam lēmumu balstoties uz visiem viedokļiem

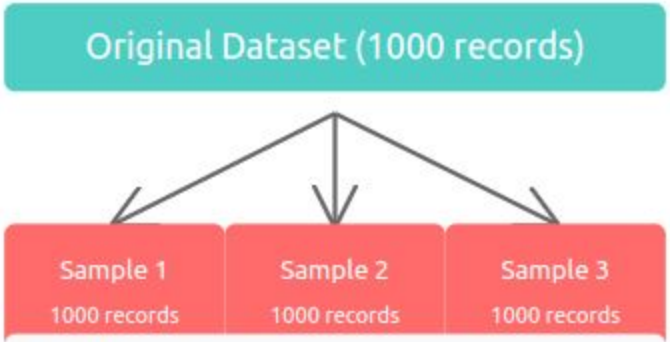
Bagging piemērs ar datiem

1. Oriģinālie dati (1000 klienti)
2. Izveidojam 3 datu kopijas ar sajaukšanu (ieraksti var atkārtoties)
   a. Kopija 1: 1000 nejauši izvēlēti klienti
   b. Kopija 2: 1000 nejauši izvēlēti klienti
   c. Kopija 3: 1000 nejauši izvēlēti klienti
3. Apmācām modeli katrai kopijai
4. Apvienojam rezultātus

Random Forest - Populārākais Bagging pielietojums

- Bagging + Lēmumu koki
- Katrs koks redz:
  - Nejaušu datu izlasi
  - Nejaušu pazīmju izlasi
- "Mežs" pieņem lēmumu kopā

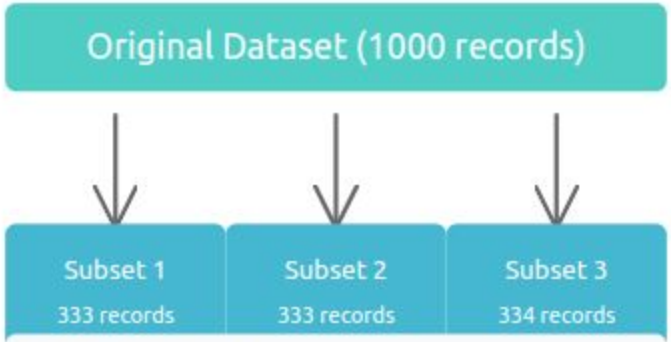
Bootstrap Sampling



Priekšrocības:

- ✓ Saglabā datu sadalījumu
- ✓ Katrs modelis redz visu datu diapazonu
- ✓ Variācija caur atkārtojumiem
- ✓ Labāka noturība pret outlier datiem
- ✓ Mazāk jutīgs pret datu zudumiem

Subsetting (Apakškopas)



Ierobežojumi:

- ⚠️ Katrs modelis redz tikai daļu datu
- ⚠️ Var zaudēt svarīgus modeļus
- ⚠️ Mazāk stabili individuālie modeļi
- ⚠️ Grūtāk apstrādāt nevienmērīgus datus
- ⚠️ Lielāka jutība pret datu kvalitāti

Bagging priekšrocības un salīdznājums ar vienu modeli

✅ Uzlabo precizitāti:

- Samazina pārmācīšanos
- Stabilāki rezultāti
- Mazāk jutīgs pret trokšņiem datos

✅ Praktiskie ieguvumi:

- Viegli parallelizējams
- Var apstrādāt lielas datu kopas
- Samazina risku pieņemt nepareizu lēmumu

Viens liels modelis:

- Ātrāka apmācība
- Vieglāk interpretēt
- Lielāks risks kļūdīties

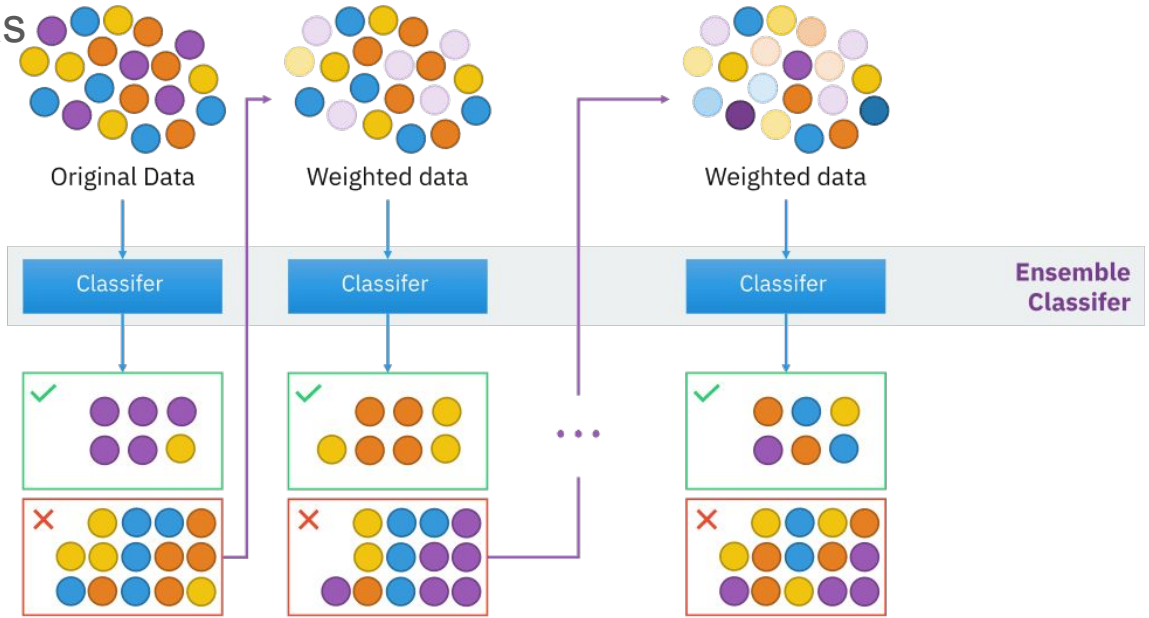
Bagging:

- Precīzāks
- Stabilāks
- Mazāk jutīgs pret kļūdām
- Prasa vairāk resursus

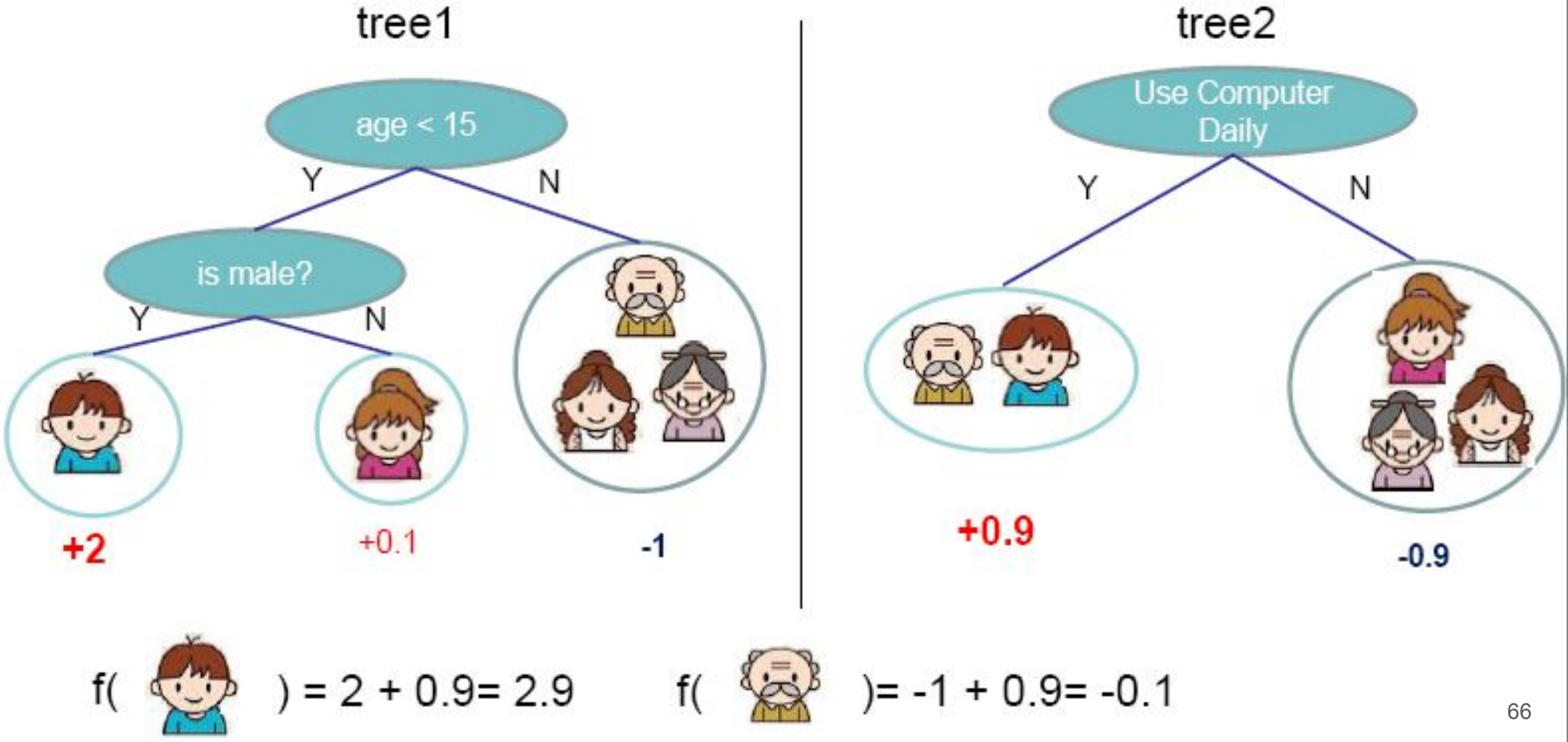
Boosting

- Pirmais koks - pamata prognozes
- Otrais koks - labo pirmā kļūdas
- Trešais koks - labo atlikušās kļūdas

...un tā tālāk



Boosting piemērs



Boosting piemērs

Koks #1:

- Prognoze: 100€
- Reālais: 150€
- Kļūda: +50€

Koks #2:

- Fokusējas uz +50€ kļūdu
- Pievieno korekciju

Koks #3:

- Fokusējas uz atlikušo kļūdu
- Vēl precīzāka prognoze

XGBoost (eXtreme Gradient Boosting)

- Industrijas standarts
- Ātrāks un precīzāks
- Automātiski novērš pārmācīšanos

🚄 Ātrums

- Paralēla apstrāde
- Optimizēti aprēķini

🎯 Precizitāte

- Uzlabots boosting
- Labāka regularizācija

🛠️ Pielāgojamība

- Daudz parametru
- Var optimizēt pēc vajadzības

XGBoost darbības princips

Sākotnējā prognoze

- Vienkāršs bāzes modelis
- Pirmā aproksimācija

Kļūdu aprēķins

- Cik tālu no mērķa?
- Kur vajag uzlabot?

Jaunu koku pievienošana

- Katrs fokusējas uz kļūdām
- Pakāpeniski uzlabo precizitāti

Regularizācija

- Novērš pārmācīšanos
- Kontrolē koku sarežģītību
-

Mājasdarbs

Mājasdarbs

3.4. Implementēt klasterizāciju ar PCA un Lēmumu koku

- Izmantojot AI Studio veikt klasterizāciju uz PCA 2D īpašībām ar Iris datu kopu, neņemt vērā “spieces”.
- Atkārtot eksperimentu bez PCA soļa ar visām 4 input vērtībām.
- Izmantojo laika apstākļu datu kopu izveidojat lēmumu pieņemšanas kokus iekš AI Studio