Metody Inżynierii Wiedzy

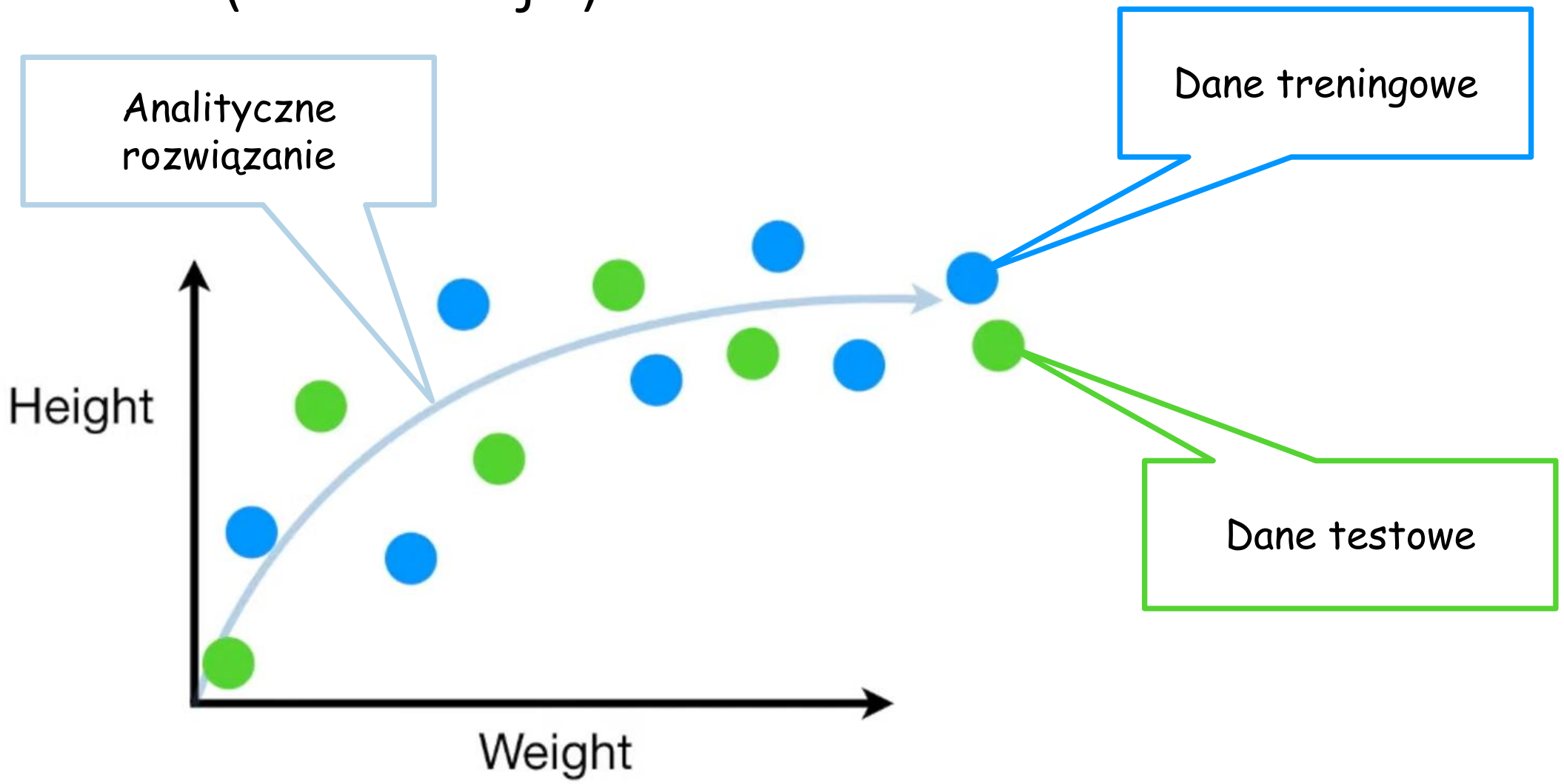
Łaczenie róznych modeli w celu działania zespołowego

Dr inż. Michał Majewski

mmajew@pjwstk.edu.pl

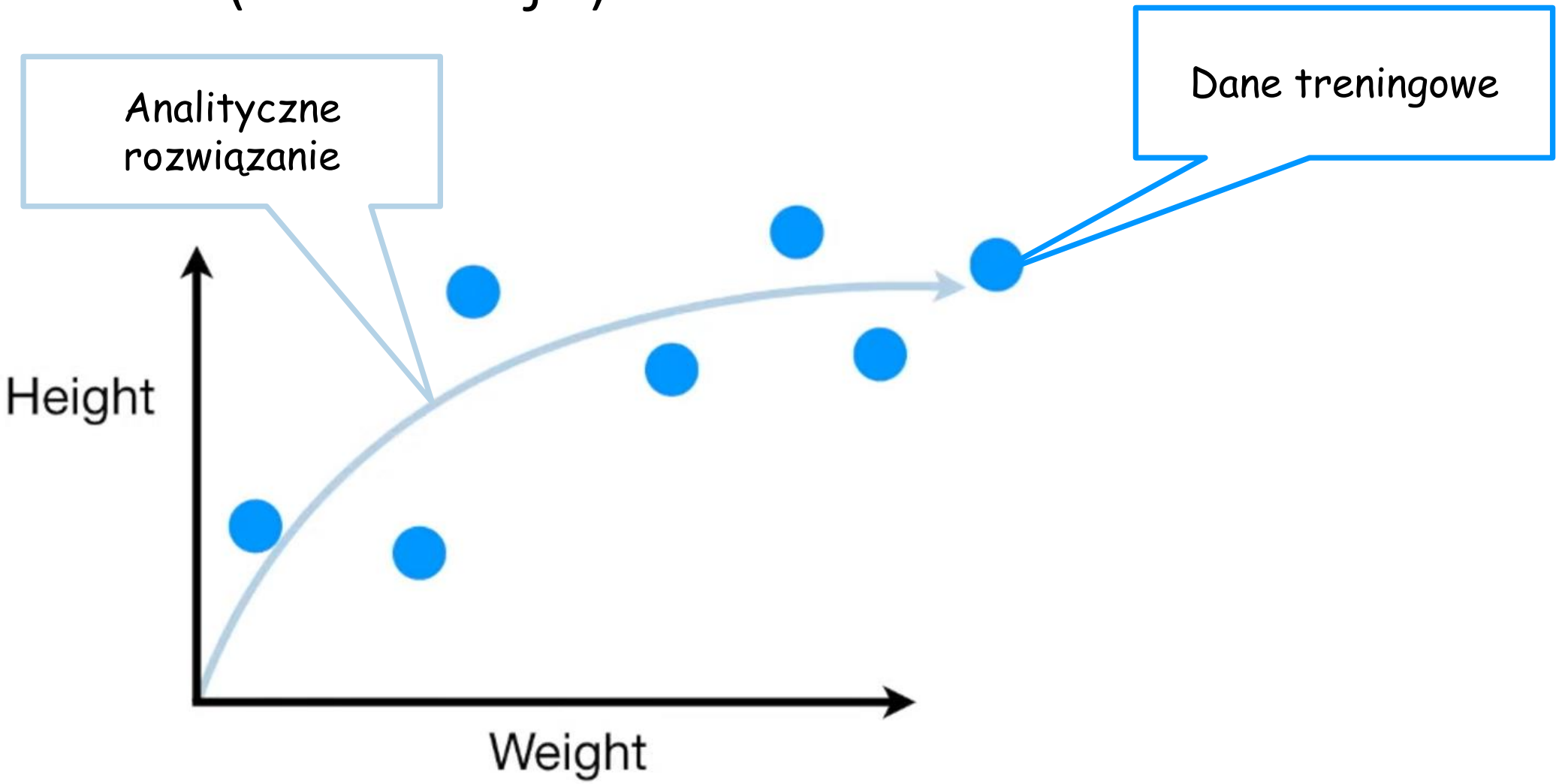
materiały: *ftp*(*public*) : *//mmajew/MIW*

Bias (błąd, stronniczość)
Variance (wariancja)



https://www.youtube.com/watch?v=EuBBz3bI-aA Machine Learning Fundamentals: Bias and Variance, StatQuest with Josh Starmer

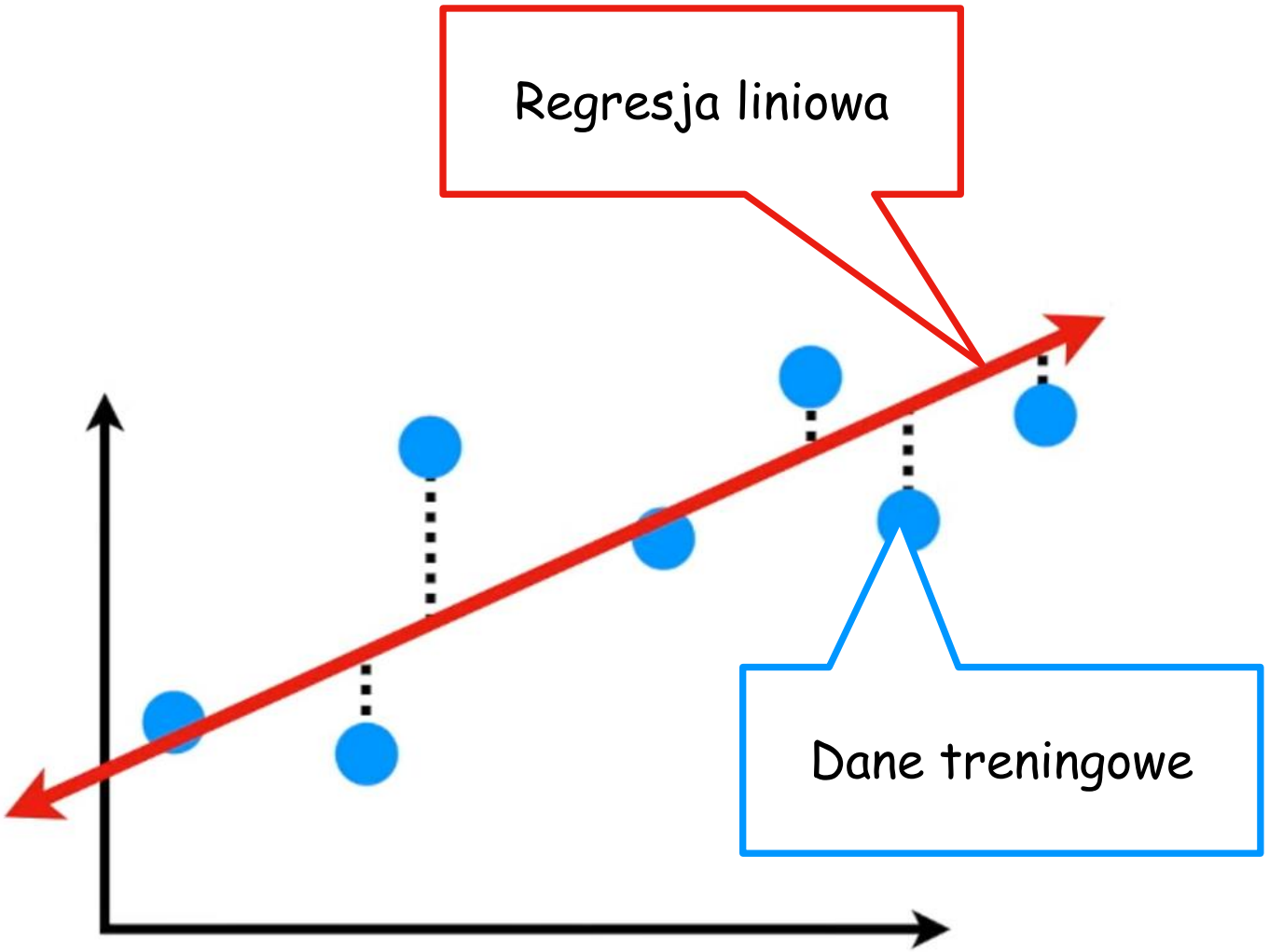
Bias (błąd, stronniczość)
Variance (wariancja)

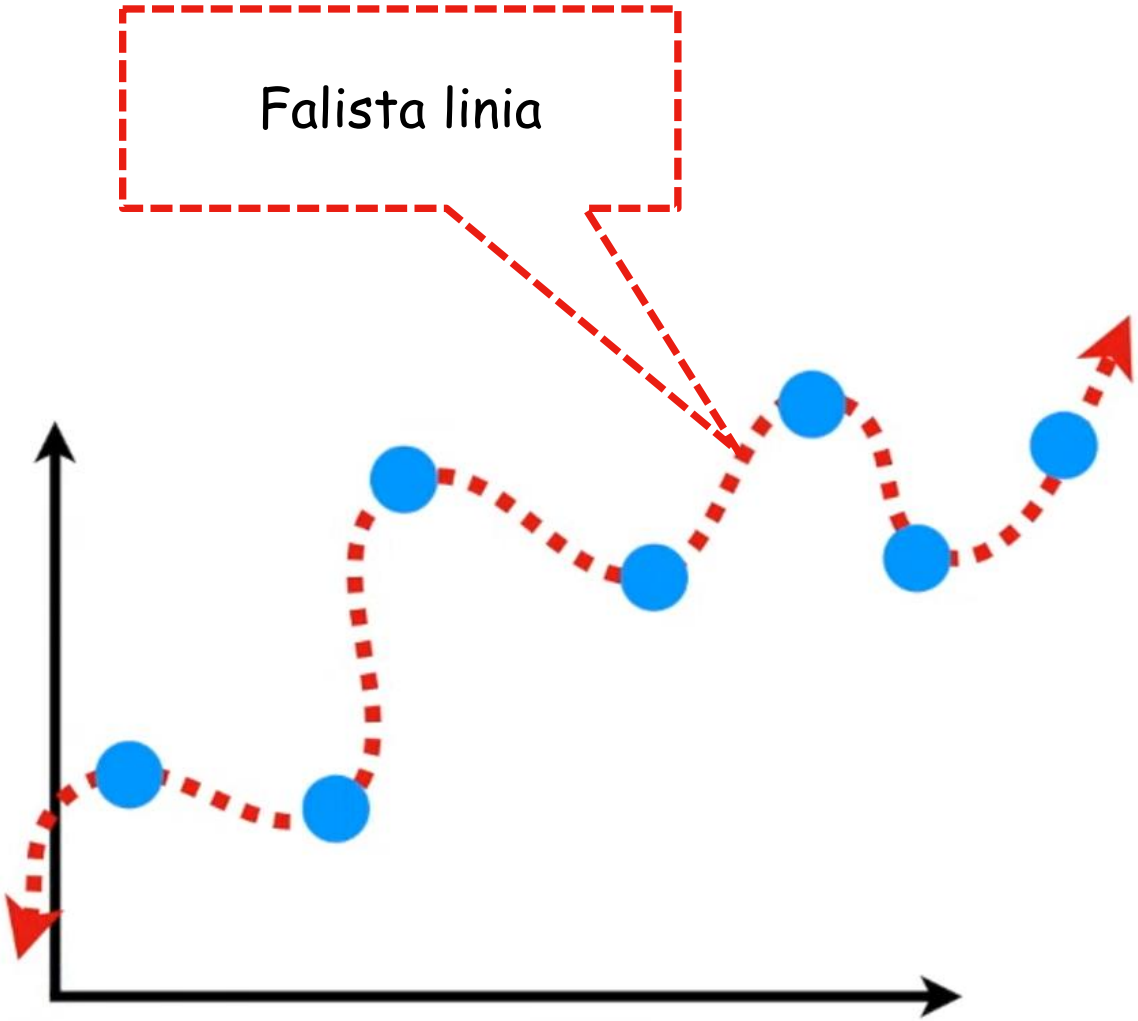


https://www.youtube.com/watch?v=EuBBz3bI-aA Machine Learning Fundamentals: Bias and Variance, StatQuest with Josh Starmer

Bias (błąd, stronniczość)
Variance (wariancja)

Bias – systematyczny błąd np. gdy model jest zbyt uproszczony

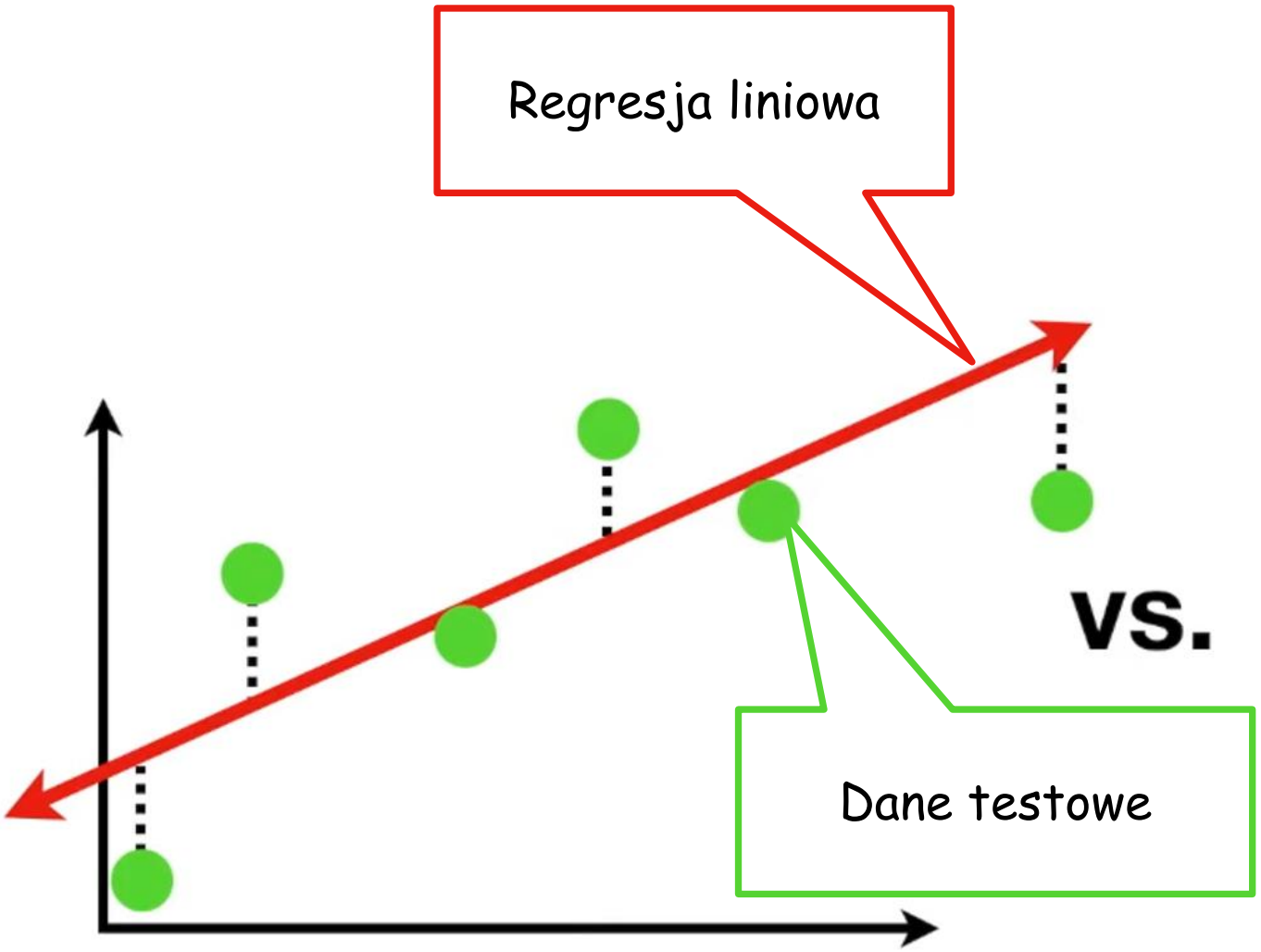


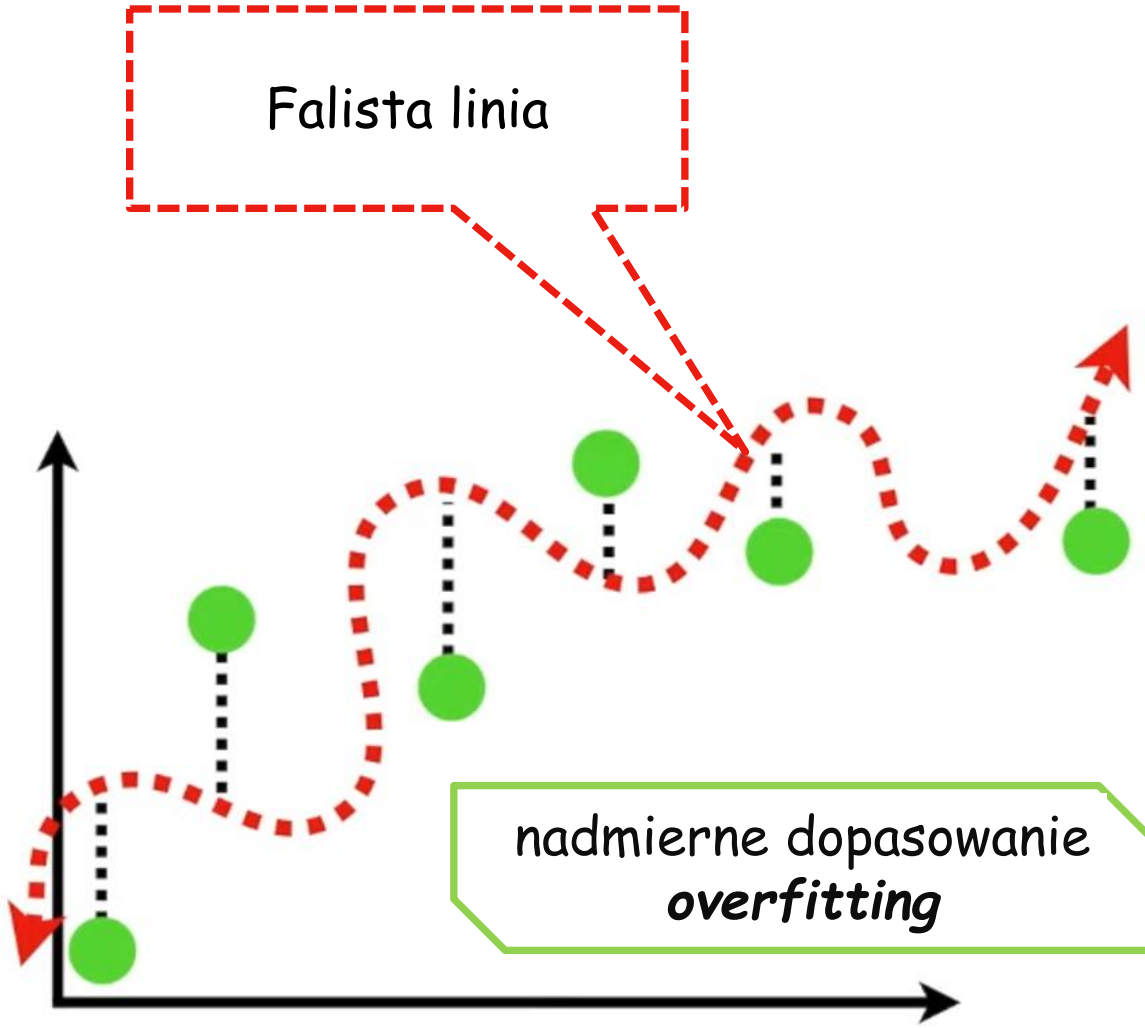


https://www.youtube.com/watch?v=EuBBz3bI-aA Machine Learning Fundamentals: Bias and Variance, StatQuest with Josh Starmer

Bias (błąd, stronniczość)
Variance (wariancja)

wariancja jak bardzo różnią się prognozy modelu, gdy jest uczony na różnych podzbiorach danych treningowych

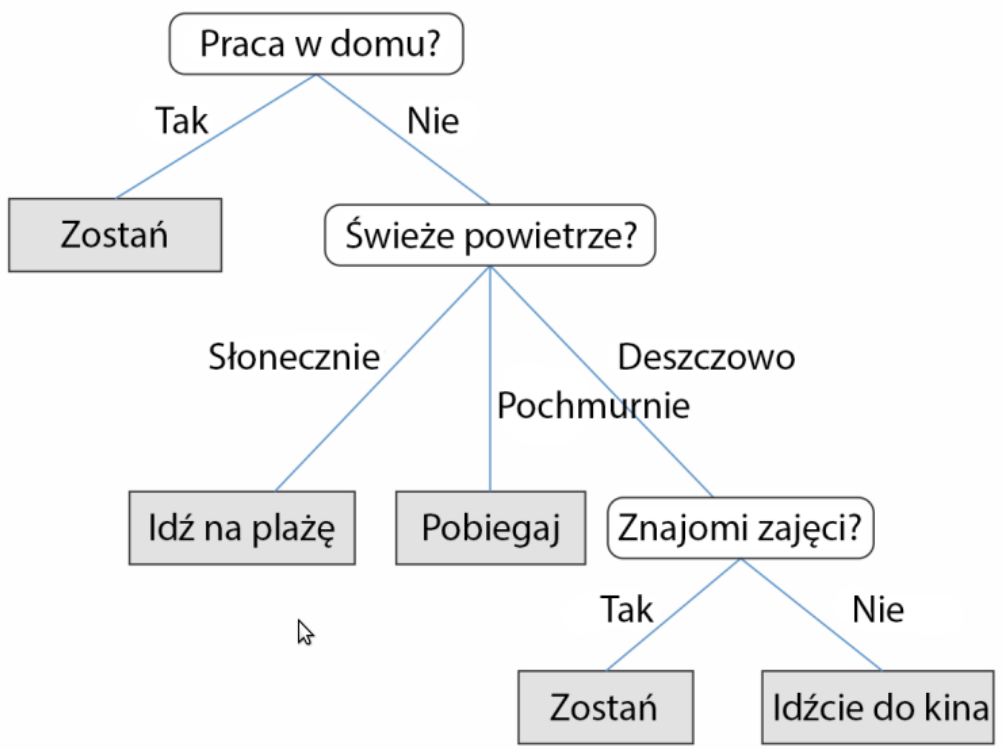




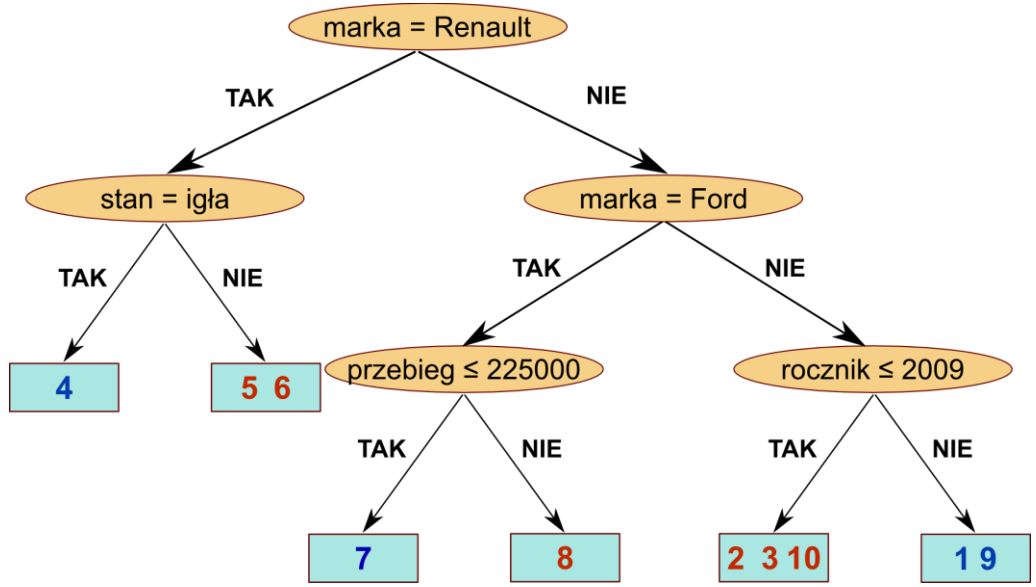
https://www.youtube.com/watch?v=EuBBz3bI-aA Machine Learning Fundamentals: Bias and Variance, StatQuest with Josh Starmer

Drzewo decyzyjne

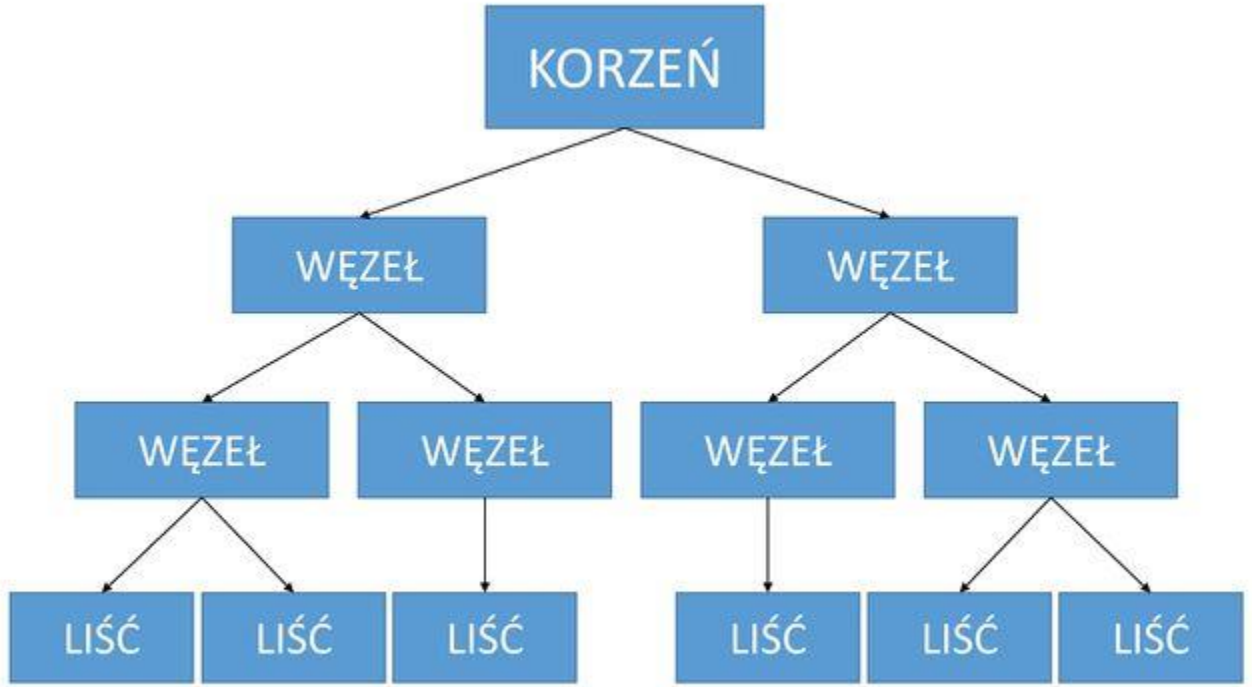
Drzewo klasyfikacji



Drzewo regresyjne



Drzewo decyzyjne

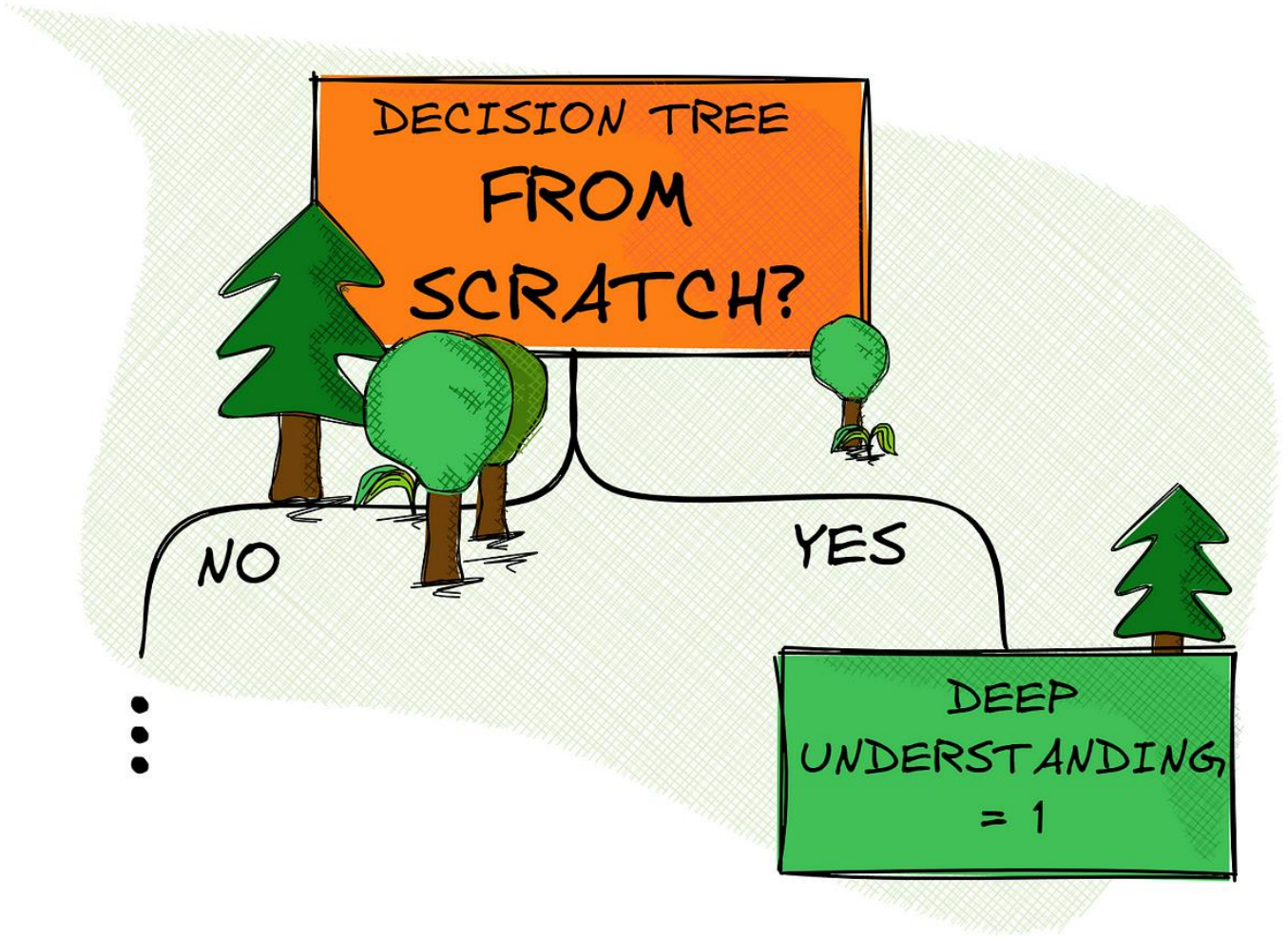


https://mfiles.pl/pl/index.php/Plik:Drzewo_decyzyjne_schemat.jpg

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



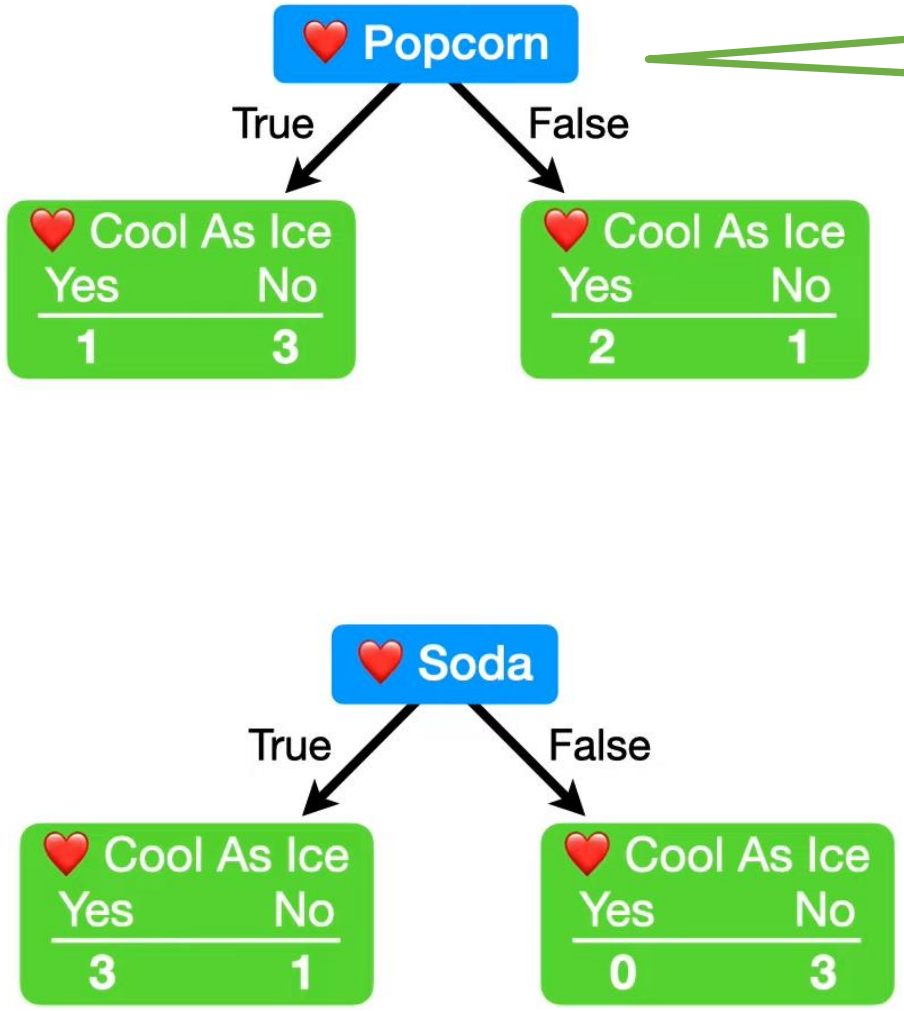
https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No

Szukamy węzła startowego (korzenia)

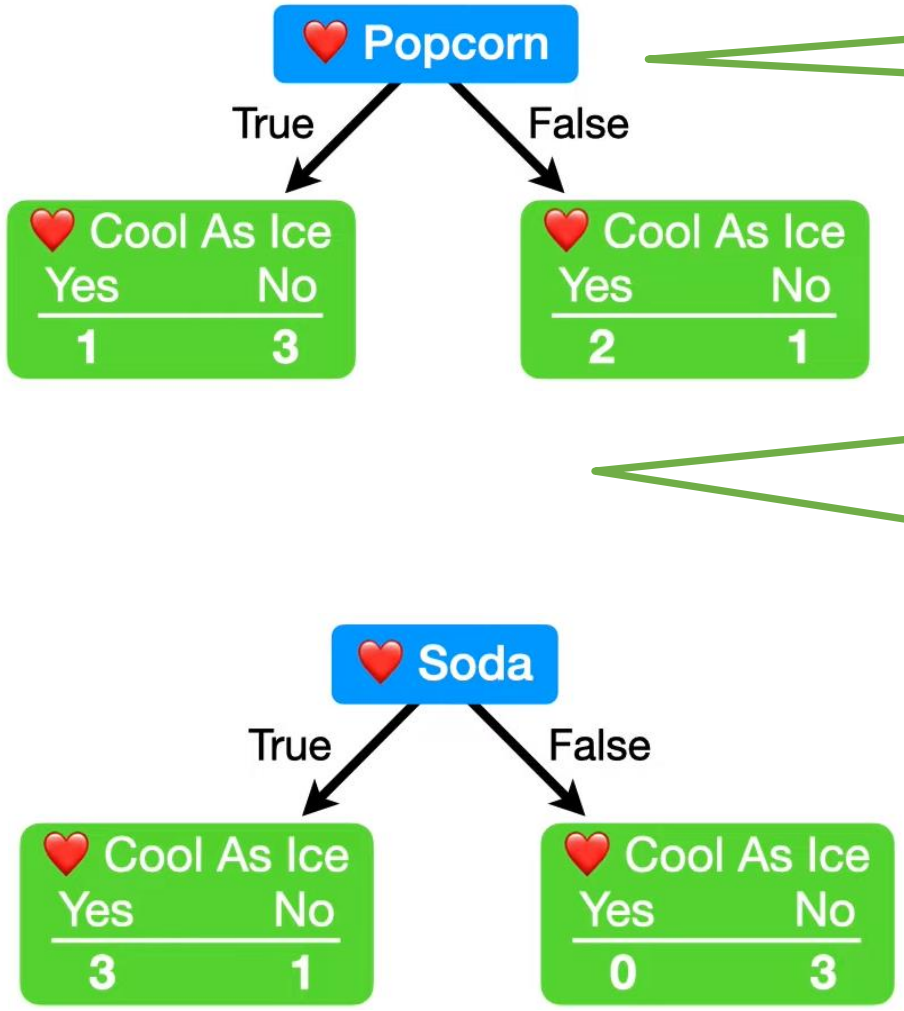


https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



Szukamy węzła startowego (korzenia)

Kryteria rozgałęzienia (miara zanieczyszczeń):
> Entropia
> **Wskaźnik Giniego**
> Błąd klasyfikacji

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Porównanie wskaźników zanieczyszczeń – wykład 5

Entropia

Dla wszystkich niepustych klas $p(i|t) \neq 0$:

$$I_H(t) = -\sum_{i=1}^{c} p(i|t) \log_2 p(i|t)$$

- Wyrażenie $p(i|t)$ oznacza proporcję pomiędzy próbkami należącymi do klasy i w danym węźle t,
- Entropia będzie wynosiła 0, jeśli wszystkie próbki w węźle będą należały do tej samej klasy,
- Maksymalną wartość osiągnie wtedy, gdy będziemy mieli do czynienia z jednorodnym rozkładem klas,
- Poprzez kryterium entropii próbujemy zmaksymalizować wzajemne informacje w drzewie.

Błąd klasyfikacji

Błąd klasyfikacji możemy określić jako:

$$I_E(t) = 1 - \max\{p(i|t)\}$$

- Jest to kryterium przydatne do przycinania,
- Nie jest zalecane do rozwijania drzewa, ponieważ wykazuje mniejszą czułość na zmiany w rozkładzie prawdopodobieństwa klas wewnątrz węzła.

Wskaźnik Giniego

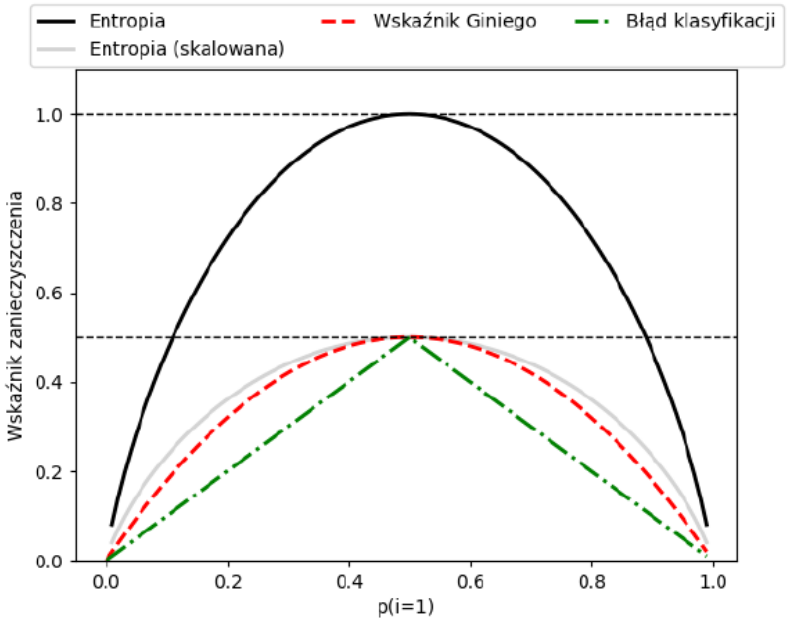
Wskaźnik Giniego możemy interpretować jako kryterium służące do minimalizowania prawdopodobieństwa nieprawidłowej klasyfikacji:

$$I_G(t) = \sum_{i=1}^{c} p(i|t)(1 - p(i|t)) = 1 - \sum_{i=1}^{c} p(i|t)^2$$

- Podobnie jak w przypadku entropii, wskaźnik Giniego uzyskuje największą wartość, gdy klasy są między sobą idealnie wymieszane; np. dla binarnej konfiguracji klas (c = 2):

$$I_G(t) = 1 - \sum_{i=1}^{c} p(i|t)^2 = 0,5$$

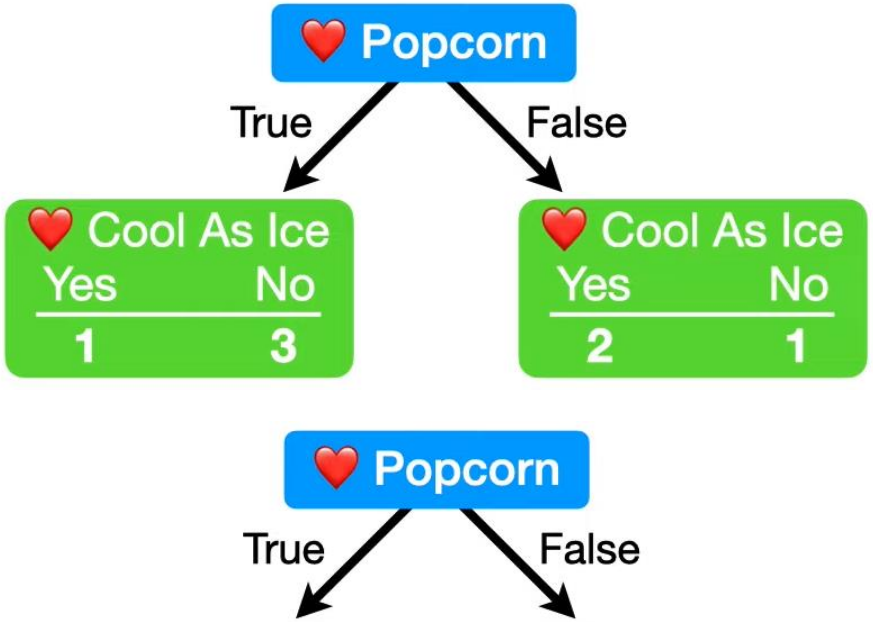
- Wskaźnik Giniego i entropia generują zazwyczaj podobne wyniki,
- Zamiast różnych kryteriów zanieczyszczeń, lepiej jest eksperymentować z różnymi wartościami granicy przycinania.



Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



Kryteria rozgałęzienia:
Wskaźnik Giniego *IG*
Gini impurity

$$I_{G,pop,false} = 1 - (\frac{2}{2+1})^2 - \left(\frac{1}{2+1}\right)^2 = 0.444$$

$$I_{G,pop,true} = 1 - (\frac{1}{1+3})^2 - \left(\frac{3}{1+3}\right)^2 = 0.375$$

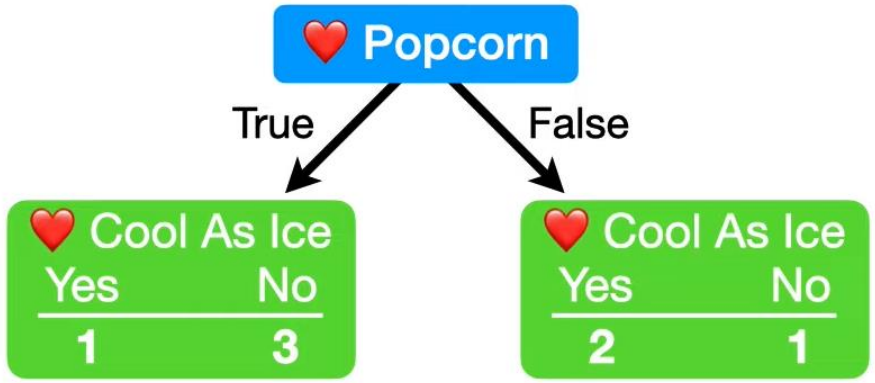
$$\boldsymbol{I_{G,popcorn}} = \left(\frac{4}{4+3}\right) 0.375 + \left(\frac{3}{4+3}\right) 0.444 = \mathbf{0.405}$$

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

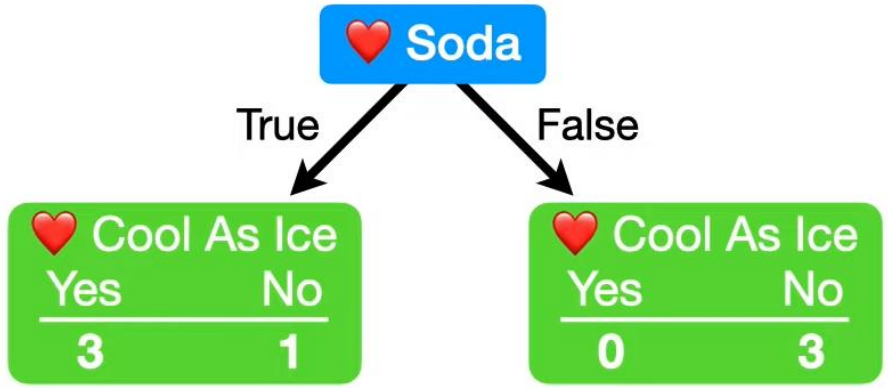
Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



$I_{G,popcorn}$=0.405



$I_{G,soda}$=0.214

Kryteria rozgałęzienia:
Wskaźnik Giniego *IG*
Gini impurity

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Sortujemy

Kryteria rozgałęzienia:
Wskaźnik Giniego *IG*
Gini impurity

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Y	7	No
Yes	N	12	No
No	Yes	18	Yes
No	Y	35	Yes
Yes	Y	38	Yes
Yes	N	50	No
No	N	83	No

9.5
15
26.5
36.5
44
66.5

Określamy granice w połowie

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

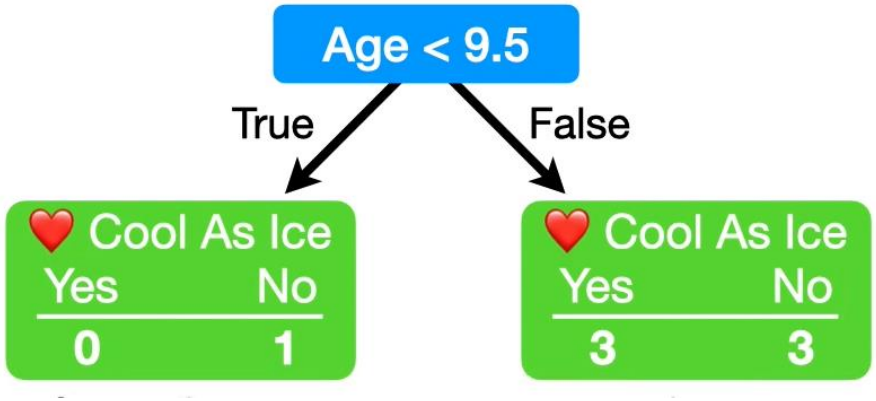
Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Y	7	No
Yes	N	12	No
No	Yes	18	Yes
No	Y	35	Yes
Yes	Y	38	Yes
Yes	N	50	No
No	N	83	No

9.5
15
26.5
36.5
44
66.5

Kryteria rozgałęzienia:
Wskaźnik Giniego *IG*
Gini impurity



$$I_{G,9.5,true} = 1 - (\frac{0}{1})^2 - \left(\frac{1}{1}\right)^2 = 0$$

$$I_{G,9.5,false} = 1 - (\frac{3}{3+3})^2 - \left(\frac{3}{3+3}\right)^2 = 0.5$$

$$\boldsymbol{I_{G,9.5}} = \left(\frac{1}{1+6}\right) 0 + \left(\frac{6}{1+6}\right) 0.5 = \mathbf{0.429}$$

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

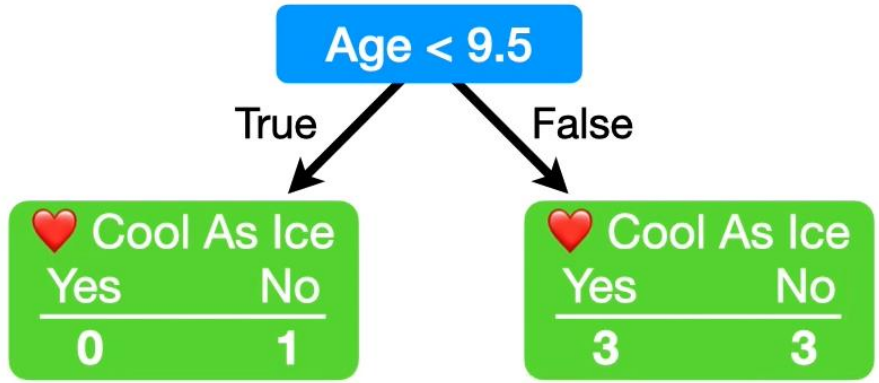
Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Y	7	No
Yes	N	12	No
No	Yes	18	Yes
No	Y	35	Yes
Yes	Y	38	Yes
Yes	N	50	No
No	N	83	No

9.5
15
26.5
36.5
44
66.5

Obliczamy IG dla każdej granicy

0.429
0.343
0.476
0.476
0.343
0.429

Kryteria rozgałęzienia:
Wskaźnik Giniego *IG*
Gini impurity



$$I_{G,9.5,true} = 1 - \left(\frac{0}{1}\right)^2 - \left(\frac{1}{1}\right)^2 = 0$$

$$I_{G,9.5,false} = 1 - \left(\frac{3}{3+3}\right)^2 - \left(\frac{3}{3+3}\right)^2 = 0.5$$

$$\boldsymbol{I_{G,9.5}} = \left(\frac{1}{1+6}\right) 0 + \left(\frac{6}{1+6}\right) 0.5 = \mathbf{0.429}$$

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Y	7	No
Yes	N	12	No
No	Yes	18	Yes
No	Y	35	Yes
Yes	Y	38	Yes
Yes	N	50	No
No	N	83	No

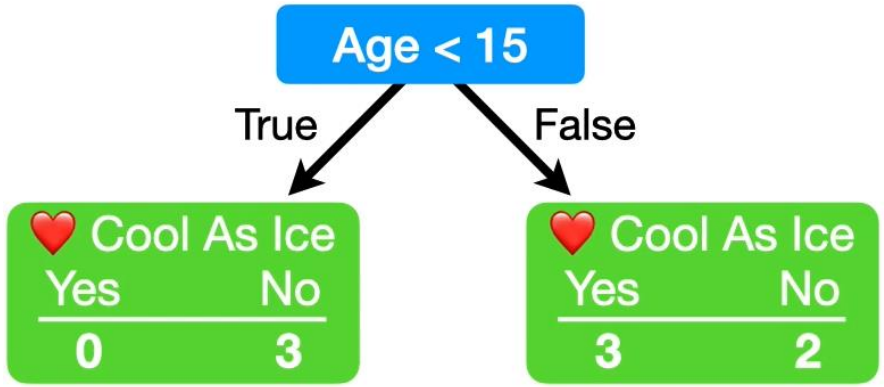
9.5 | 15 | 26.5 | 36.5 | 44 | 66.5

Szukamy minimum

0.429 | 0.343 | 0.476 | 0.476 | 0.343 | 0.429

Kryteria rozgałęzienia:
Wskaźnik Giniego *IG*
Gini impurity

W tym przykładzie wybieramy 15 lub 44



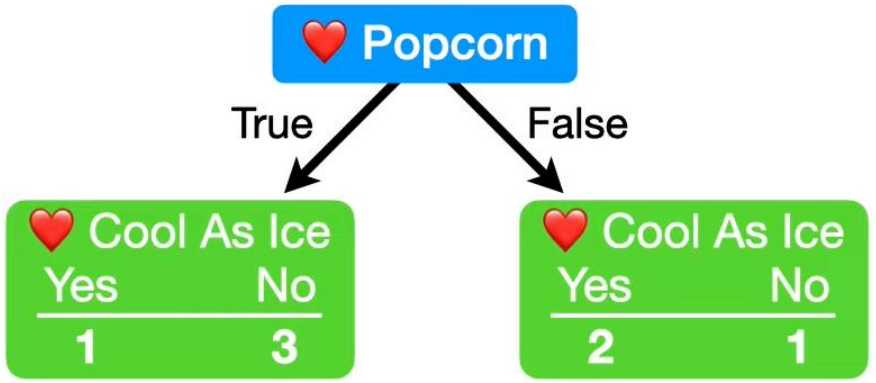
https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

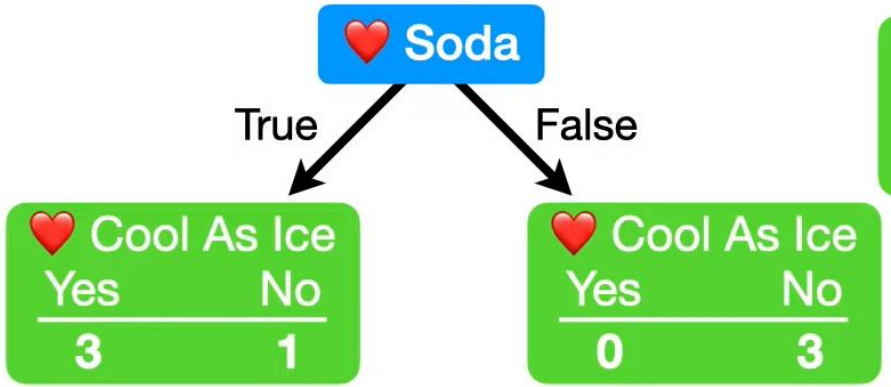
Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No

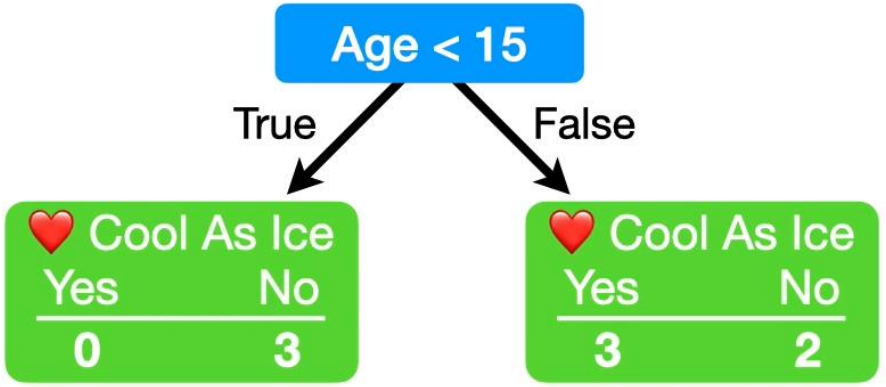
Kryteria rozgałęzienia:
Wskaźnik Giniego *IG*
Gini impurity



$I_{G,popcorn}$=0.405



$I_{G,soda}$=0.214



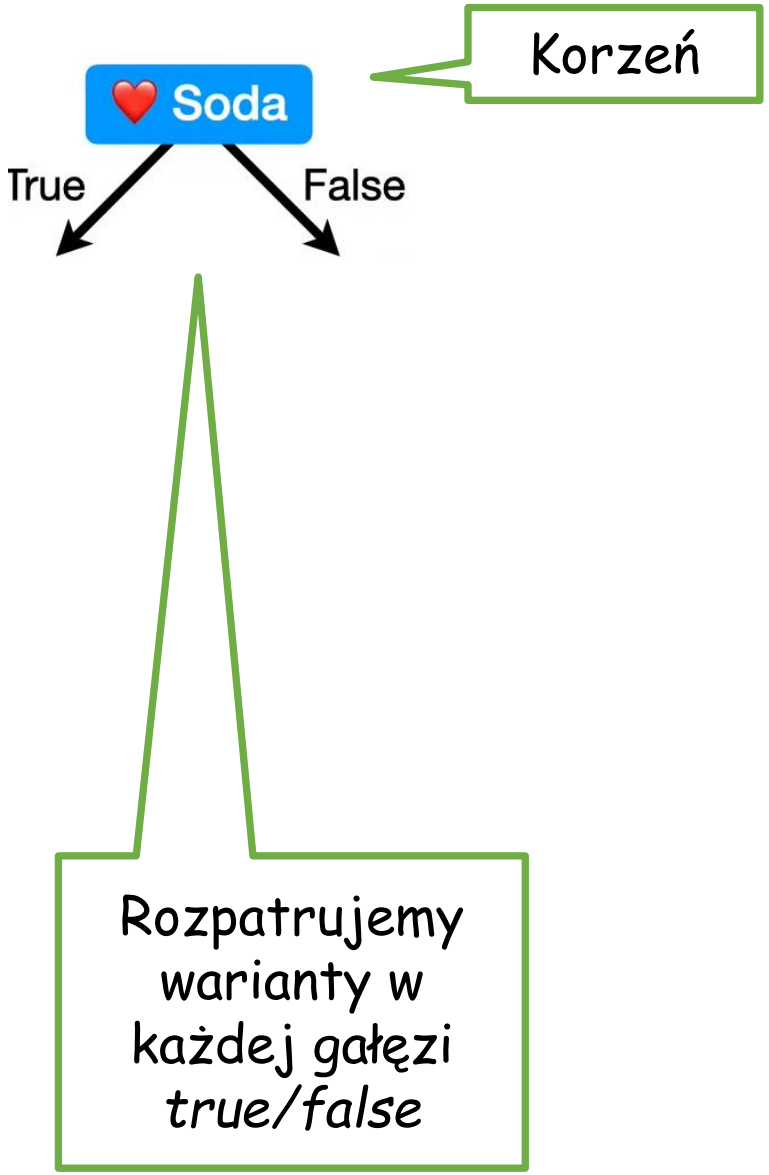
$I_{G,15}$=0.343

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No

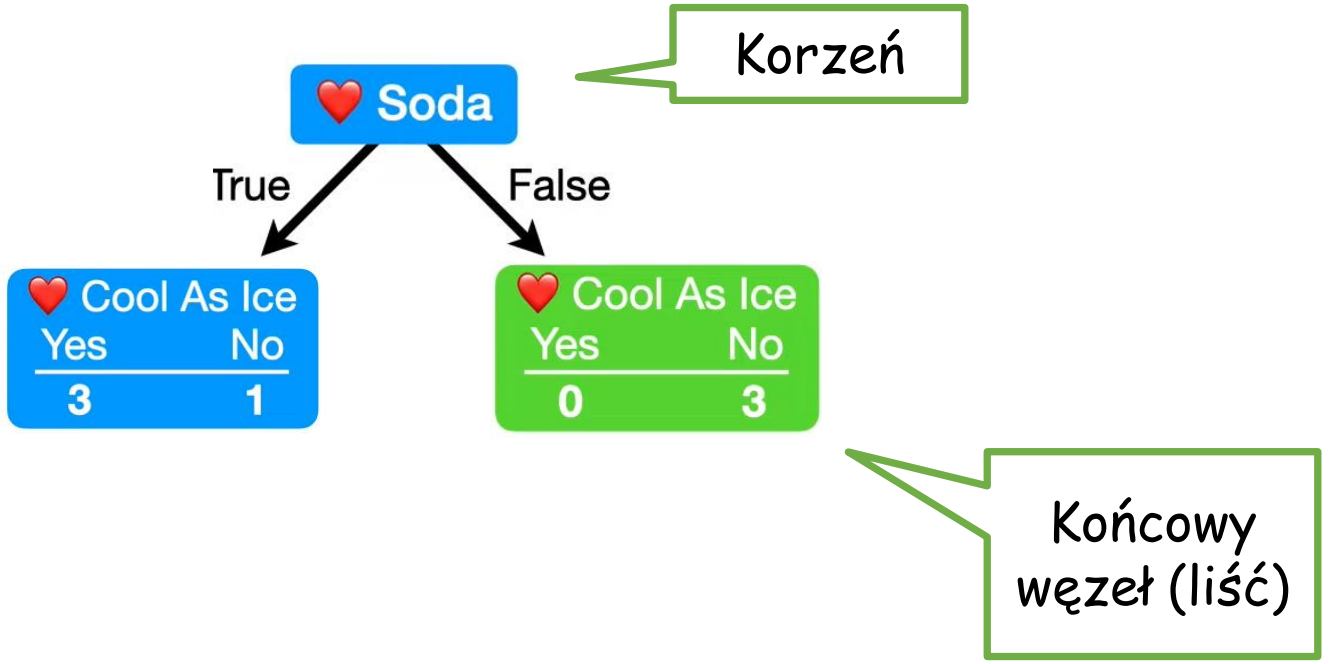


https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No

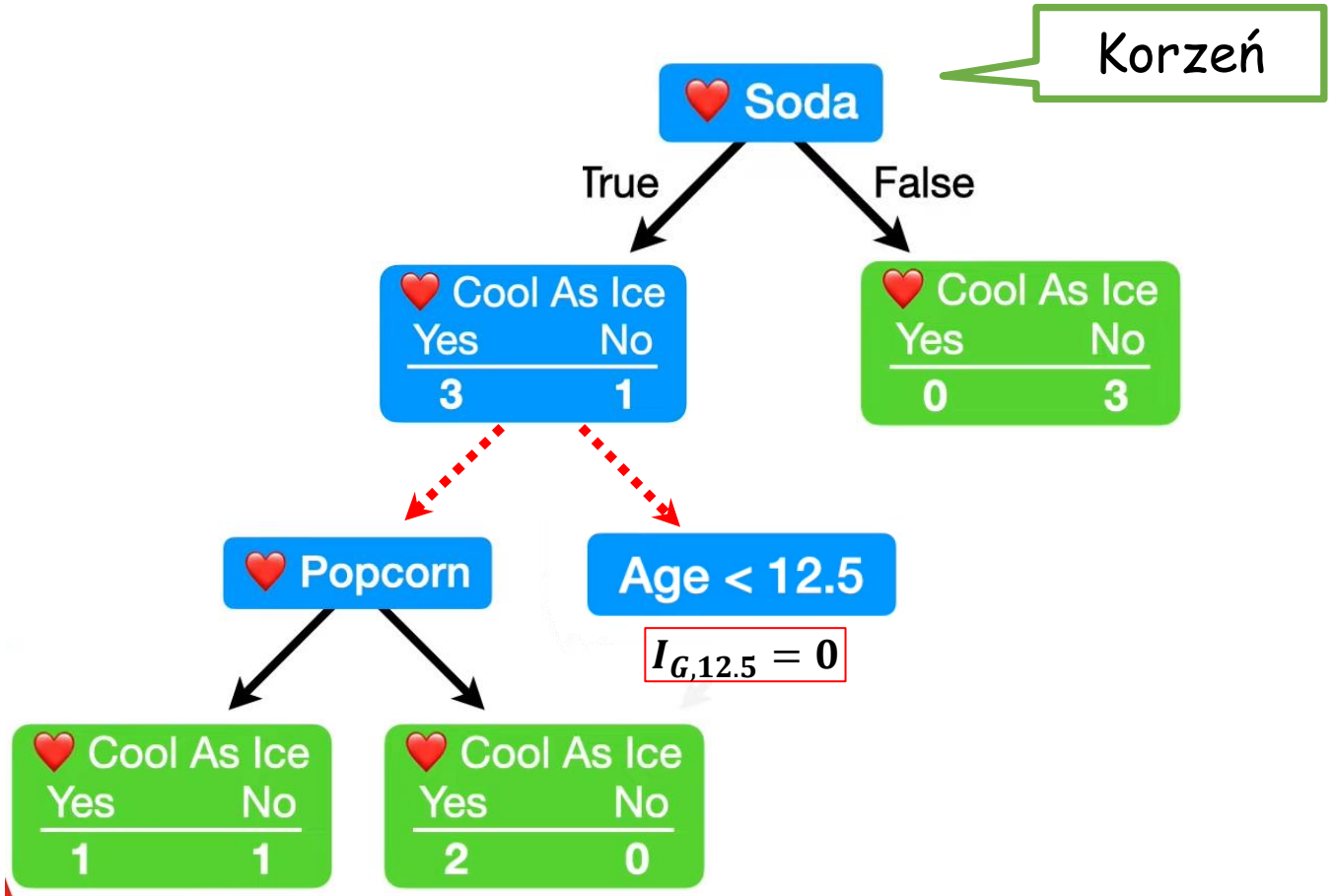


https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



$$I_{G,pop,true} = 1 - (\frac{1}{1+1})^2 - \left(\frac{1}{1+1}\right)^2 = 0.5$$

$$I_{G,pop,false} = 1 - (\frac{2}{0+2})^2 - \left(\frac{0}{0+2}\right)^2 = 0$$

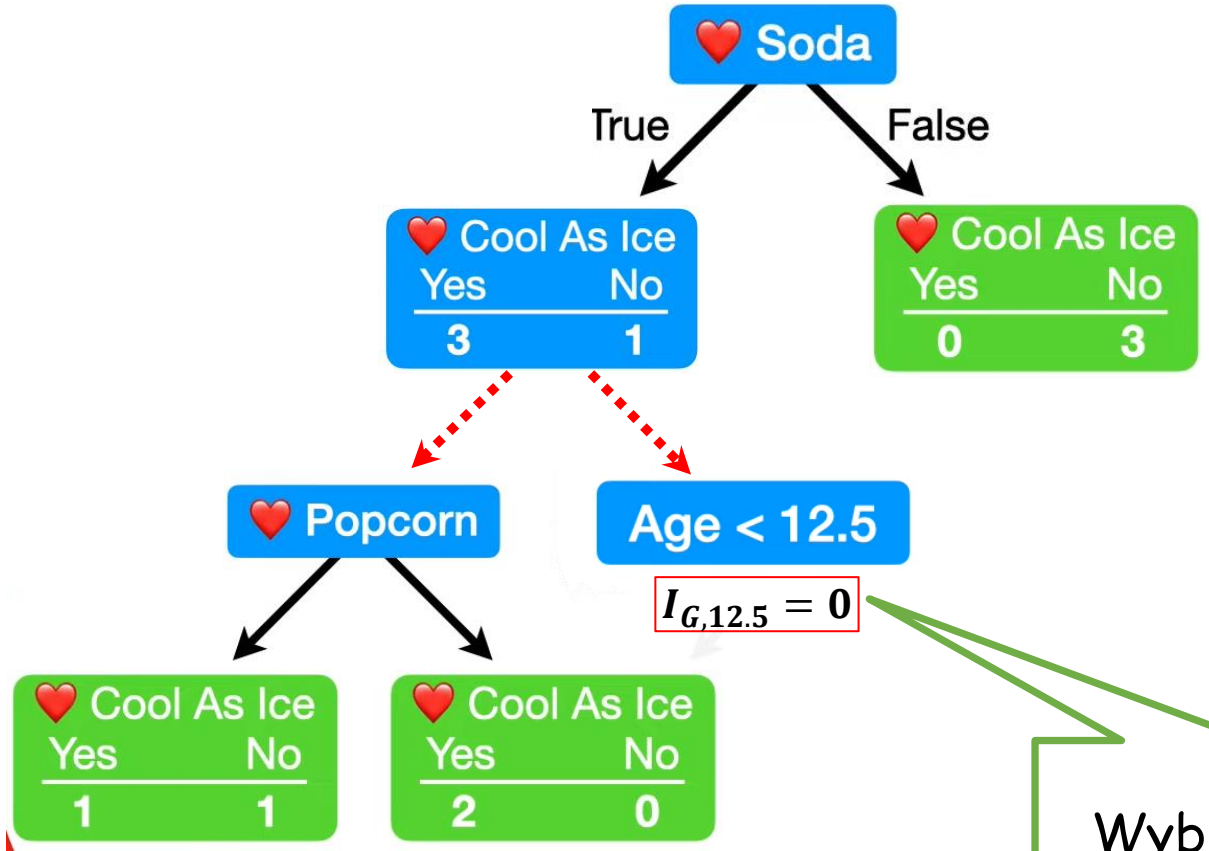
$$\boldsymbol{I_{G,popcorn}} = \left(\frac{2}{2+2}\right) 0.0 + \left(\frac{2}{2+2}\right) 0.5 = \mathbf{0.25}$$

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



$$I_{G,pop,true} = 1 - \left(\frac{1}{1+1}\right)^2 - \left(\frac{1}{1+1}\right)^2 = 0.5$$

$$I_{G,pop,false} = 1 - \left(\frac{2}{0+2}\right)^2 - \left(\frac{0}{0+2}\right)^2 = 0$$

$$I_{G,popcorn} = \left(\frac{2}{2+2}\right) 0.0 + \left(\frac{2}{2+2}\right) 0.5 = \mathbf{0.25}$$

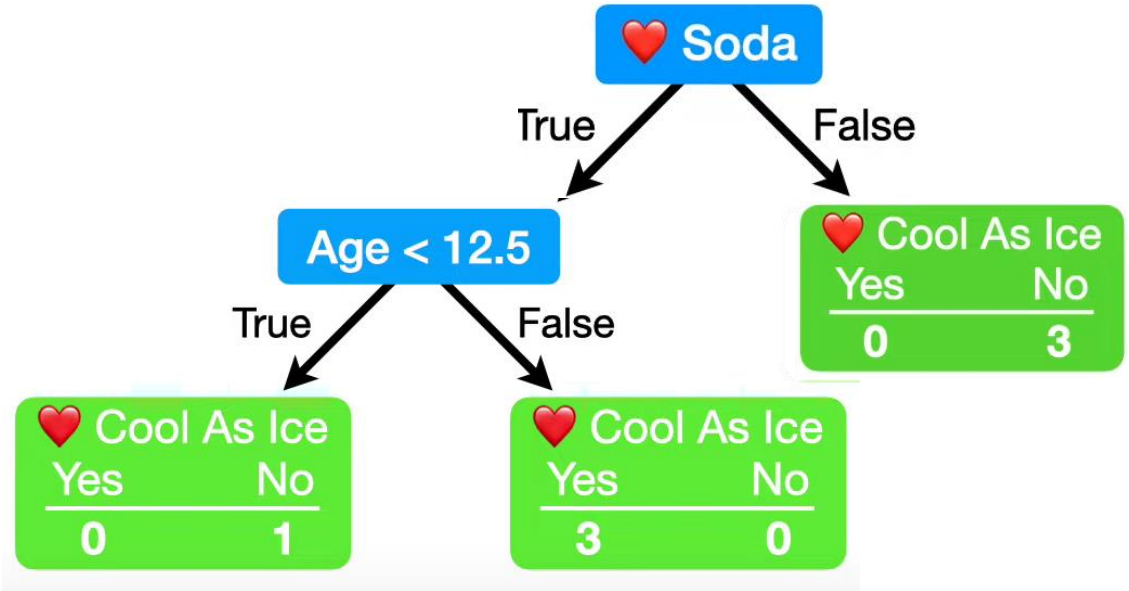
Wybieramy wiek 12.5 jako kolejne rozgałęzienie

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No

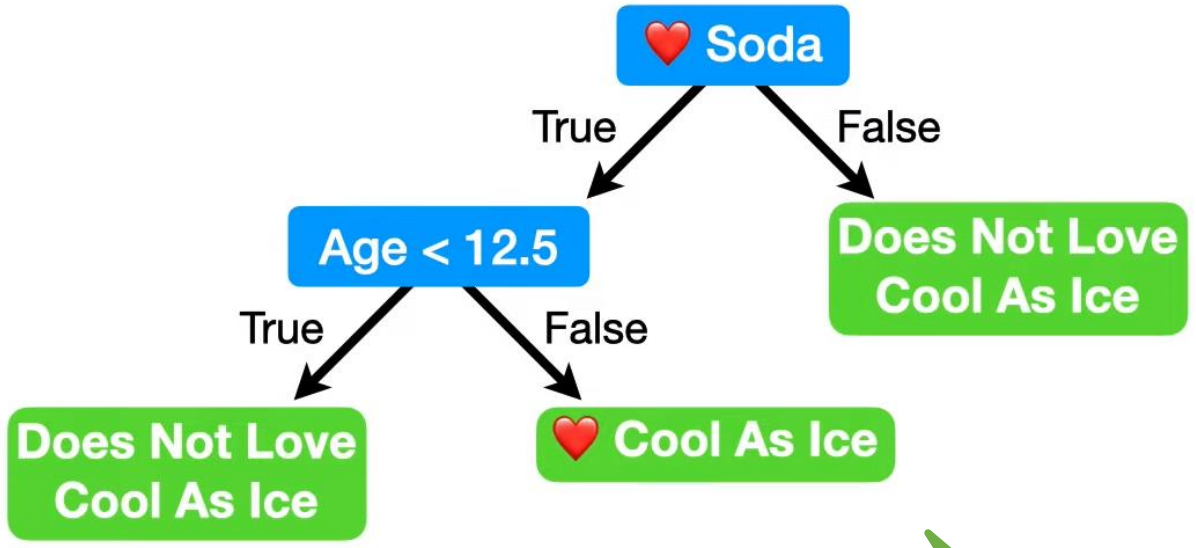


https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



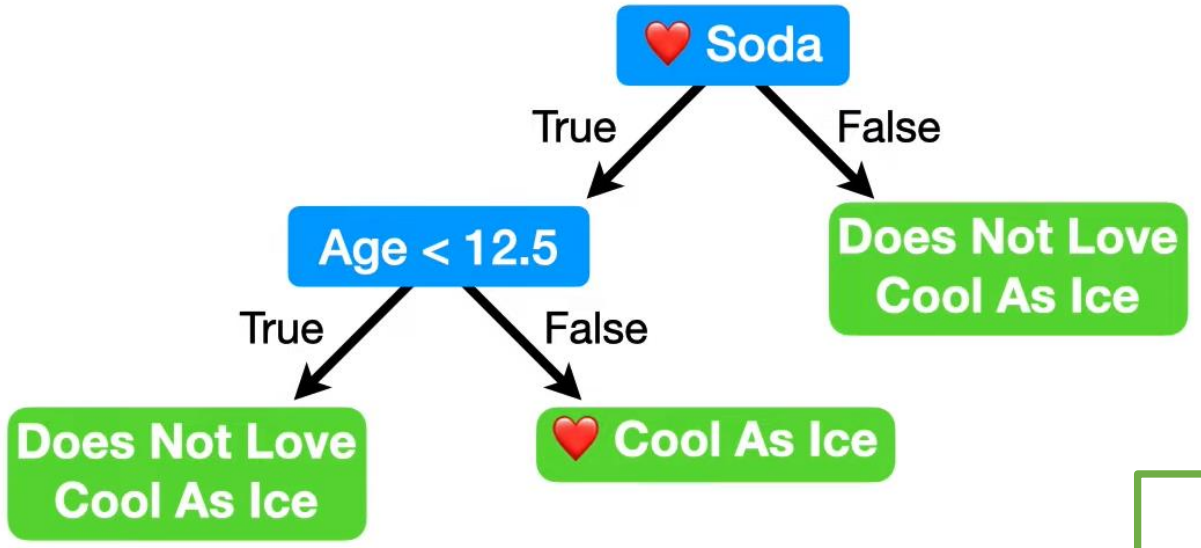
Finalne drzewo decyzyjne

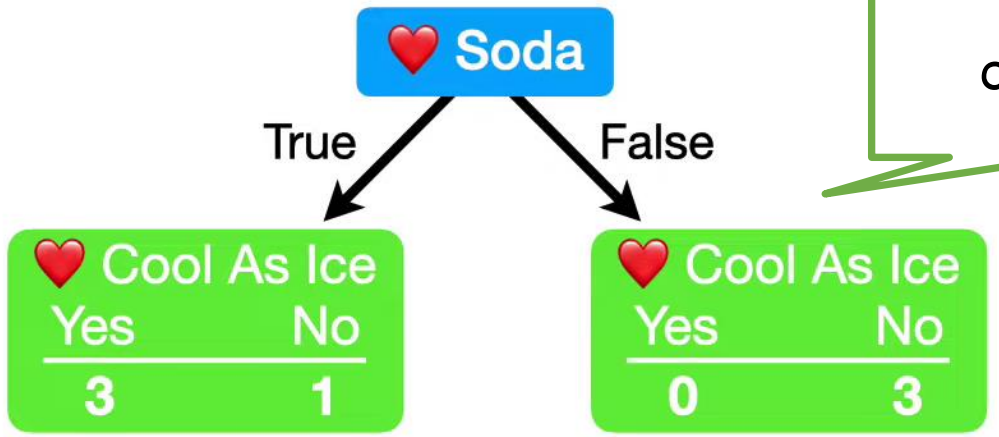
https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No





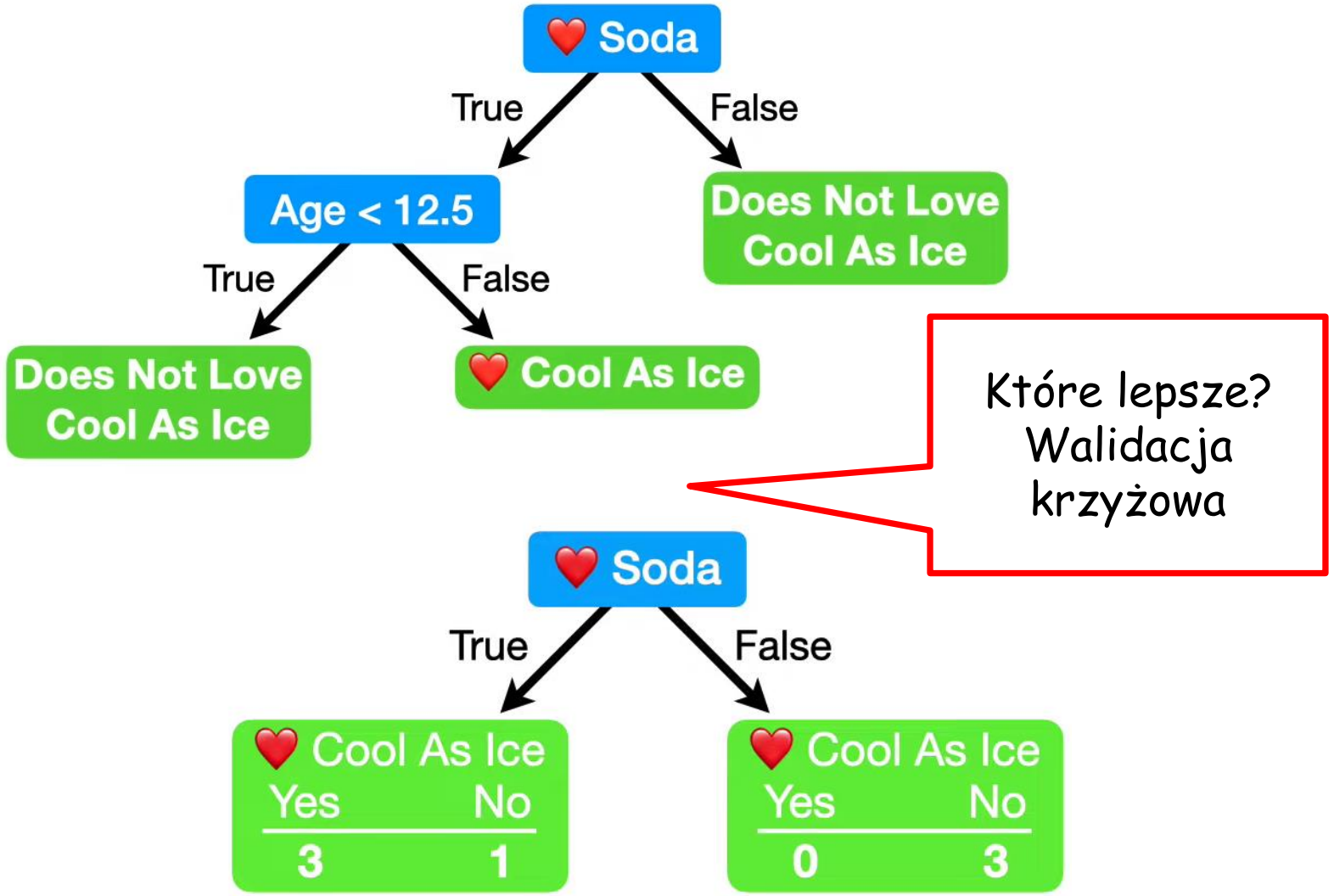
Uproszczone drzewo decyzyjne

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – jak zbudować

Dane wejściowe

Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	18	Yes
No	Yes	35	Yes
Yes	Yes	38	Yes
Yes	No	50	No
No	No	83	No



Które lepsze? Walidacja krzyżowa

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Drzewo decyzyjne – brak danych

Dane wejściowe

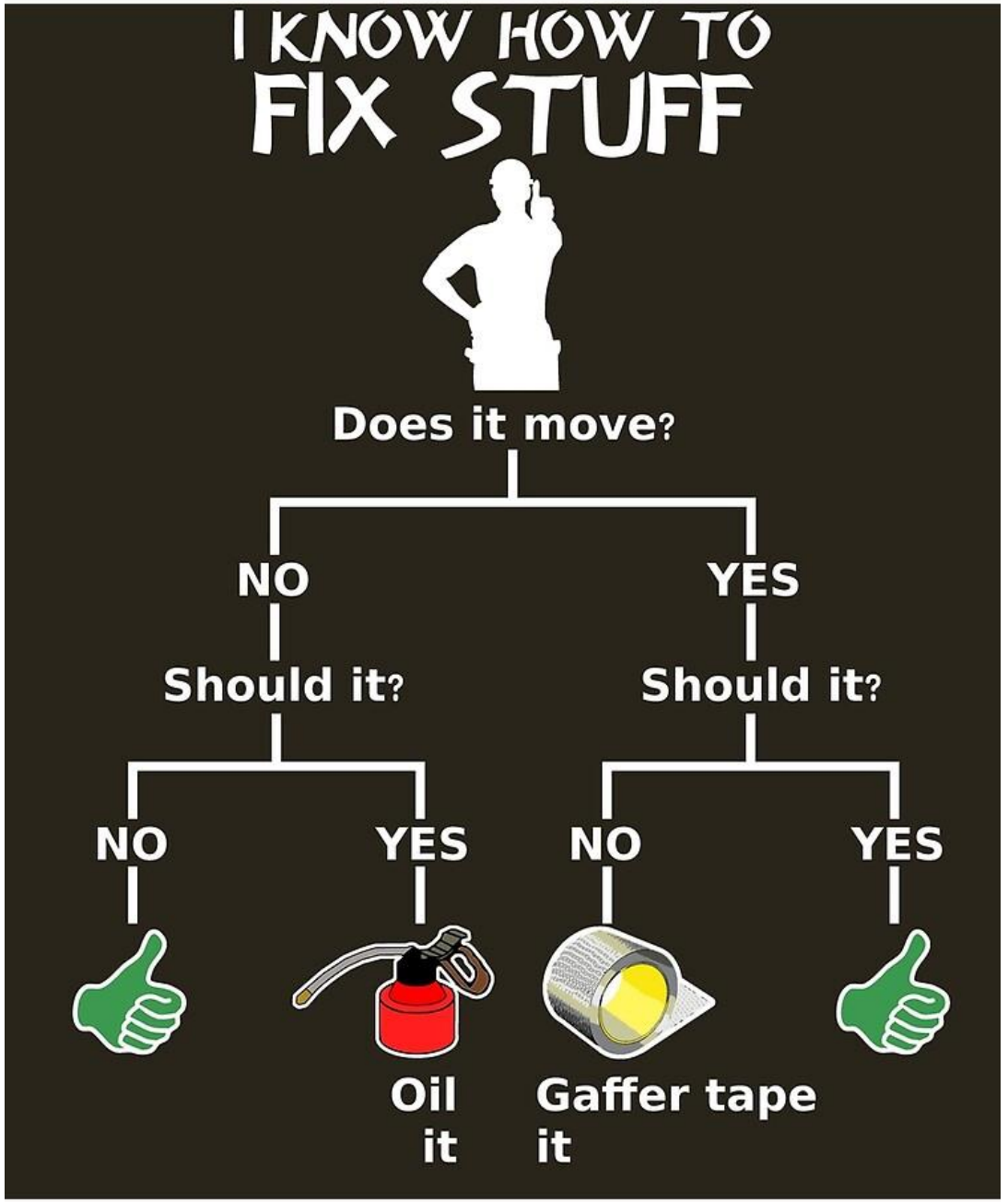
Loves Popcorn	Loves Soda	Age	Loves Cool As Ice
Yes	Yes	7	No
Yes	No	12	No
No	Yes	?	Yes
No	Yes	35	Yes
Yes	?	38	Yes
Yes	No	50	No
No	No	83	No

- Usuwanie rekordów z brakującymi danymi
- Uzupełnianie brakujących danych
- Użycie algorytmów imputacji
- Znaczniki brakujących danych
- Użycie algorytmów uczenia maszynowego do prognozowania brakujących wartości

https://www.youtube.com/watch?v=_L39rN6gz7Y Decision and Classification Trees, StatQuest with Josh Starmer

Kodujemy

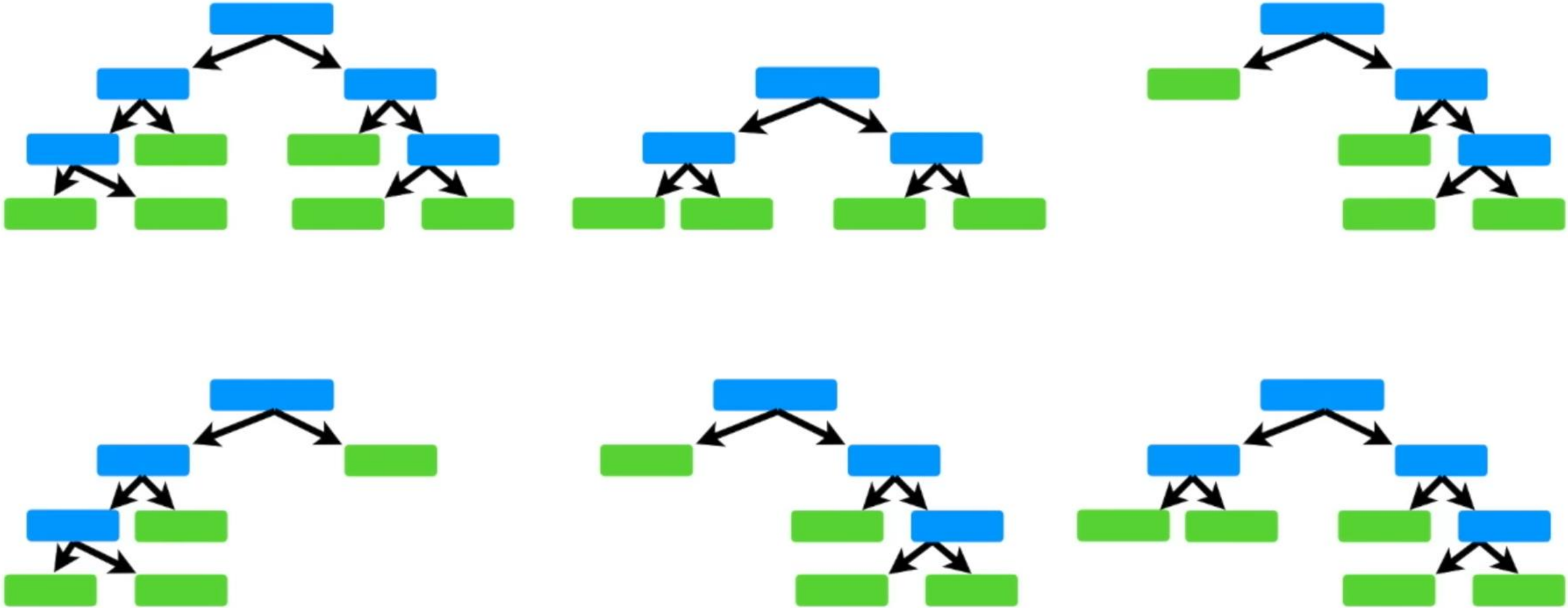
public/mmajew/MIW/05/
00_decision tree.py



Losowy las

Losowy las to **zbiór drzew decyzyjnych**, które są **szkolone niezależnie** od siebie na losowych podzbiorach danych i/lub losowych podzbiorach cech, a **następnie agregowane** w celu uzyskania stabilnego i **wydajnego modelu klasyfikacji** lub regresji.

Losowy las



Dobór liczby drzew w losowym lesie może być ustalony **empirycznie przy użyciu walidacji krzyżowej** lub za pomocą **metod automatycznego strojenia hiperparametrów**, takich jak RandomizedSearchCV lub GridSearchCV, w celu znalezienia optymalnej liczby drzew, która zapewnia najlepszą wydajność na zbiorze testowym. **Typowe wartości liczby drzew** znajdują się **w zakresie od kilkudziesięciu do kilkuset**, ale ostateczny wybór zależy od charakterystyki danych i złożoności problemu klasyfikacji lub regresji.

Drzewo decyzyjne vs. Losowy las

Drzewo decyzyjne (Decision Tree Classifier)

- ✓ **Prostota interpretacji:** Drzewa decyzyjne są łatwe do interpretacji i wizualizacji
- ✓ **Obsługa** zarówno **danych kategorycznych**, jak i **numerycznych**
- ✓ **Skuteczność w danych o niskiej do średniej złożoności**: W przypadku danych o prostszej strukturze, drzewa decyzyjne mogą osiągać dobre wyniki bez konieczności zbyt skomplikowanych procedur.

- **Podatność na przeuczenie:** Drzewa decyzyjne mają tendencję do tworzenia zbyt skomplikowanych modeli, które dobrze dopasowują się do danych treningowych, ale słabo generalizują na nowe dane.
- **Niestabilność:** Małe zmiany w danych treningowych mogą prowadzić do znaczących zmian w strukturze drzewa decyzyjnego.

Losowy las (Random Forest Classifier)

- ✓ **Odporność na przeuczenie:** Losowe lasy agregują wiele drzew, co pomaga w średnich wynikach i zmniejsza wariancję modelu.
- ✓ **Skuteczność na dużych zbiorach danych:** Losowe lasy dobrze radzą sobie z dużymi zbiorami danych i wieloma cechami
- ✓ **Brak potrzeby ręcznego strojenia parametrów**: W przeciwieństwie do drzew decyzyjnych, losowe lasy rzadko wymagają ręcznego dostrojenia parametrów

- **Mniej interpretowalne**: Losowe lasy są mniej interpretowalne niż pojedyncze drzewa decyzyjne
- **Zwiększone zużycie zasobów**: W porównaniu z pojedynczym drzewem decyzyjnym, losowe lasy mogą wymagać więcej zasobów obliczeniowych ze względu na potrzebę szkolenia i predykcji wielu drzew.

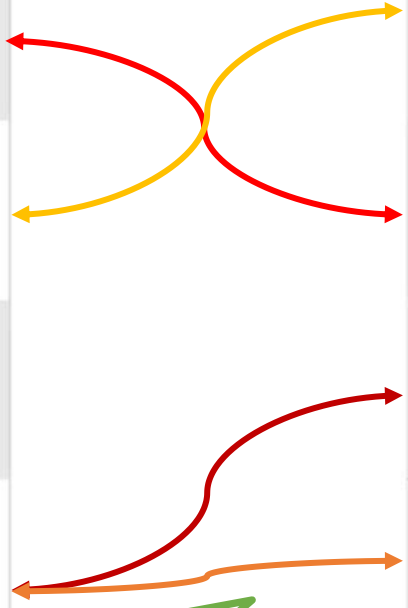
Losowy las – próbki początkowe

Original Dataset

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
No	No	No	125	No
Yes	Yes	Yes	180	Yes
Yes	Yes	No	210	No
Yes	No	Yes	167	Yes

Bootstrapped Dataset

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes



bootstrap losowa próba z powtórzeniami

https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

Losowy las – próbki początkowe

"feature bagging" lub *"feature subsampling„*
losujemy losowe podzbiory cech
np. kolumny GBC i BA

Bootstrapped Dataset

Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

Losowy las – próbki początkowe

Np. po obliczeniu IG jako korzeń mamy GC…

Bootstrapped Dataset

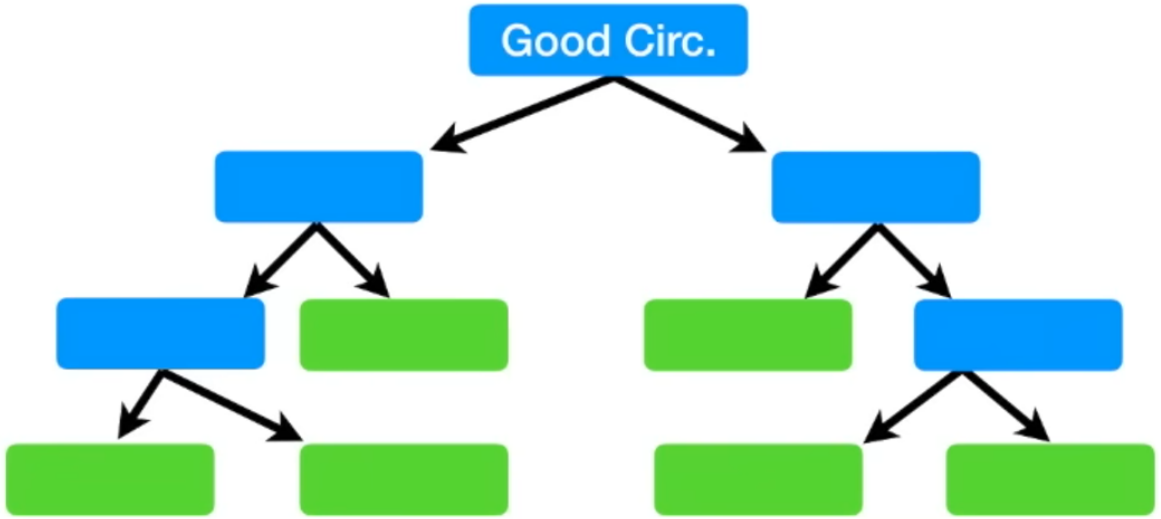
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

…i losowo wybieramy kolejne kolumny do analizy

https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

Losowy las – próbki początkowe

I dla danego bootstrapped data mamy drzewko

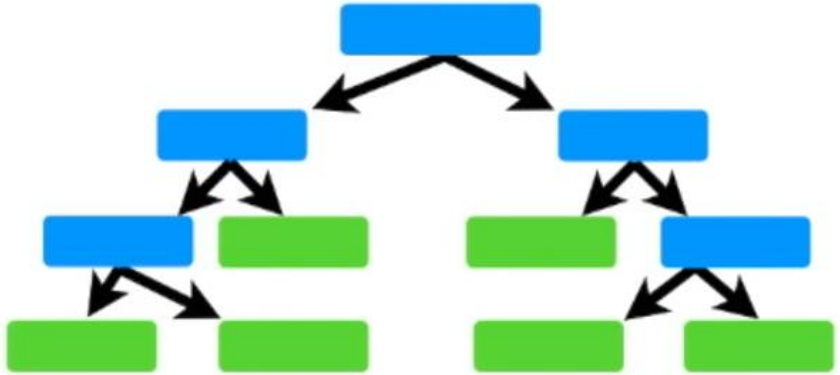


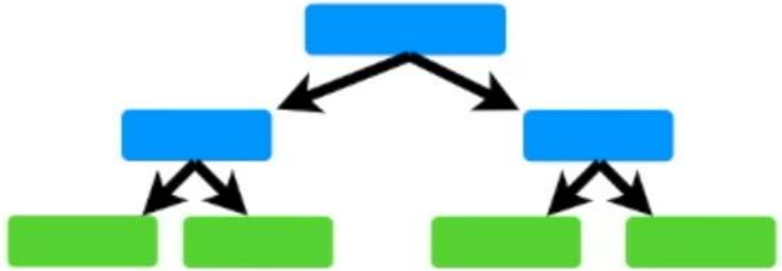
Bootstrapped Dataset

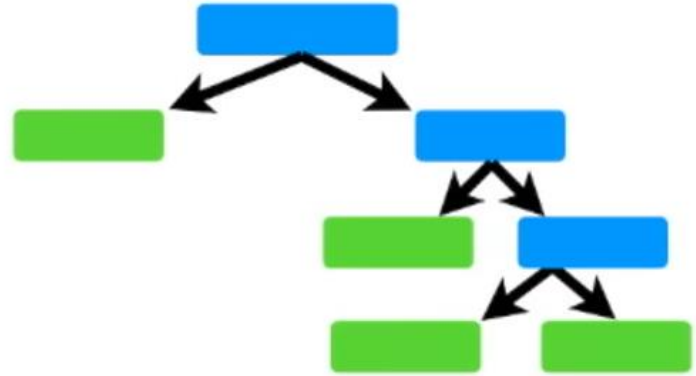
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	Yes	Yes	180	Yes
No	No	No	125	No
Yes	No	Yes	167	Yes
Yes	No	Yes	167	Yes

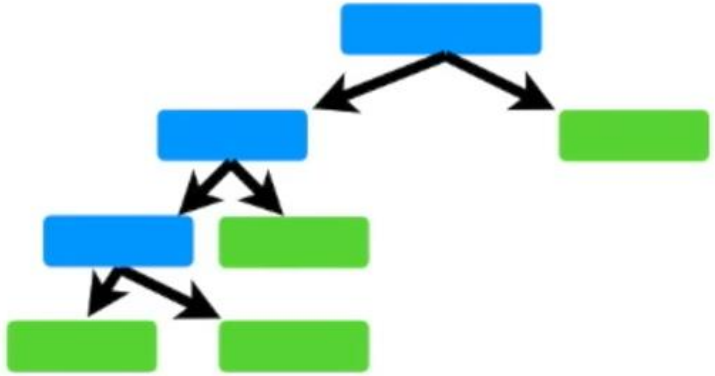
https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

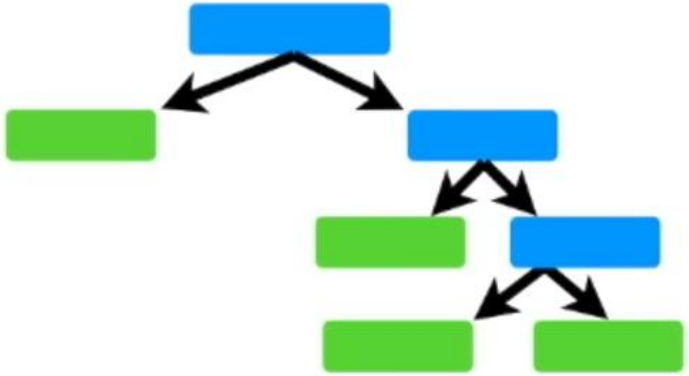
Losowy las

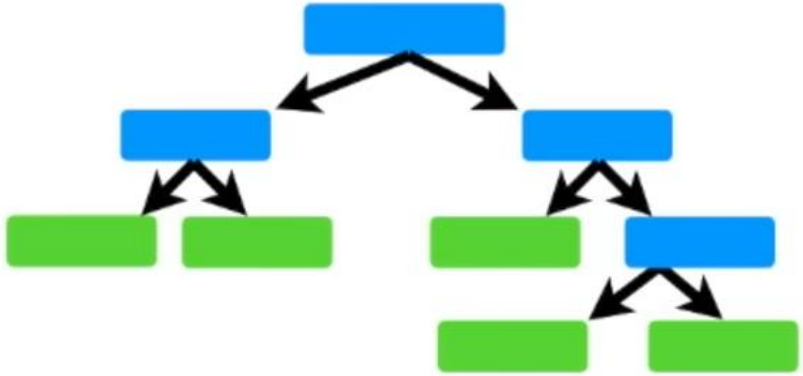








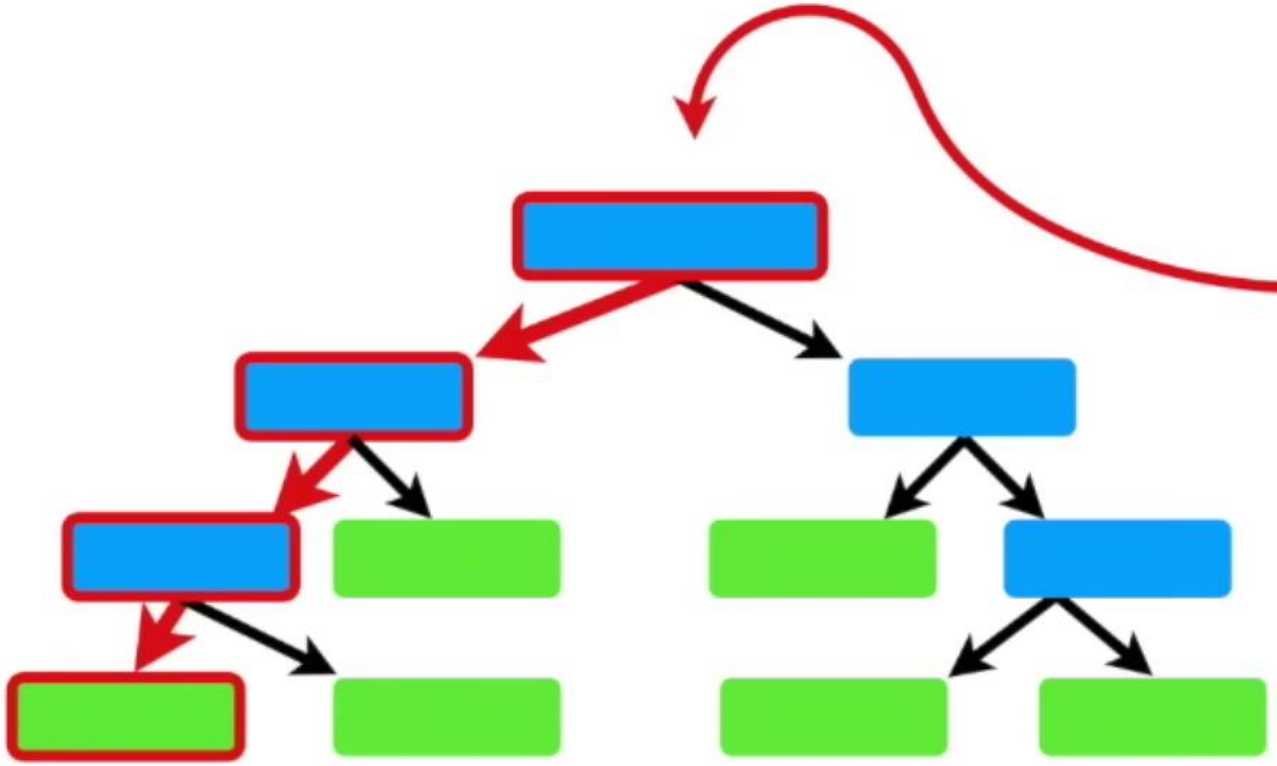




I tak mamy losowy las, jak go użyć?

https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

Losowy las – nowy pacjent



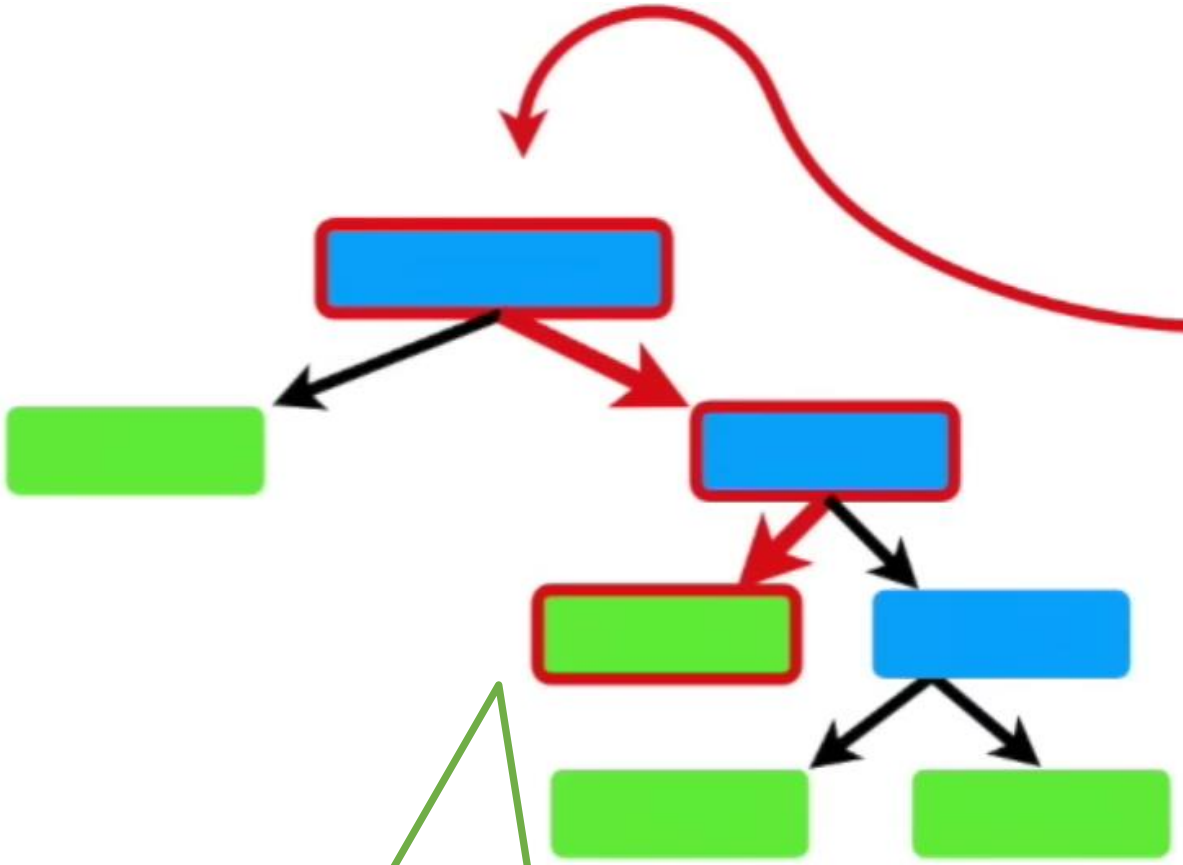
Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	

Heart Disease	
Yes	No
1	0

Pierwsze drzewo -> Heart disease = YES

https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

Losowy las – nowy pacjent



Chest Pain	Good Blood Circ.	Blocked Arteries	Weight	Heart Disease
Yes	No	No	168	

Ostatnie (szóste) drzewo -> Heart disease = YES

Heart Disease	
Yes	No
5	1

Końcowy rezultat z lasu

https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

Losowy las – testowanie danych

https://www.youtube.com/watch?v=J4Wdy0Wc_xQ StatQuest: Random Forests Part 1 - Building, Using and Evaluating

Losowy las – na co zwrócić uwagę?

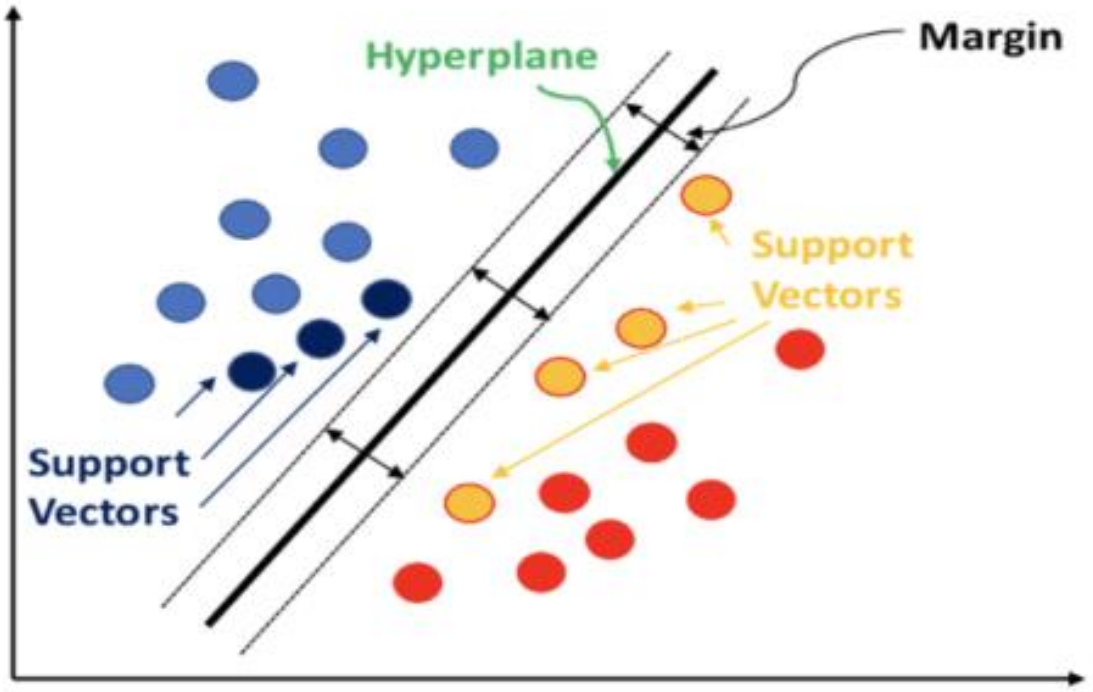
- Liczba drzew **OVER 9000!!!**
- Parametry drzewa, np.: maksymalna głębokość, minimalna liczba próbek wymaganych do podziału węzła i minimalna liczba próbek wymaganych w liściu
- Liczba cech
- Kryterium podziału
- Różnorodność danych (zwiększenie różnorodności między drzewami)
- Ocena modelu (OOB, walidacja krzyżowa)
- Interpretowalność modelu
- Skalowanie
- Monitorowanie wydajności

Kodujemy

public/mmajew/MIW/05/01_random forest.py

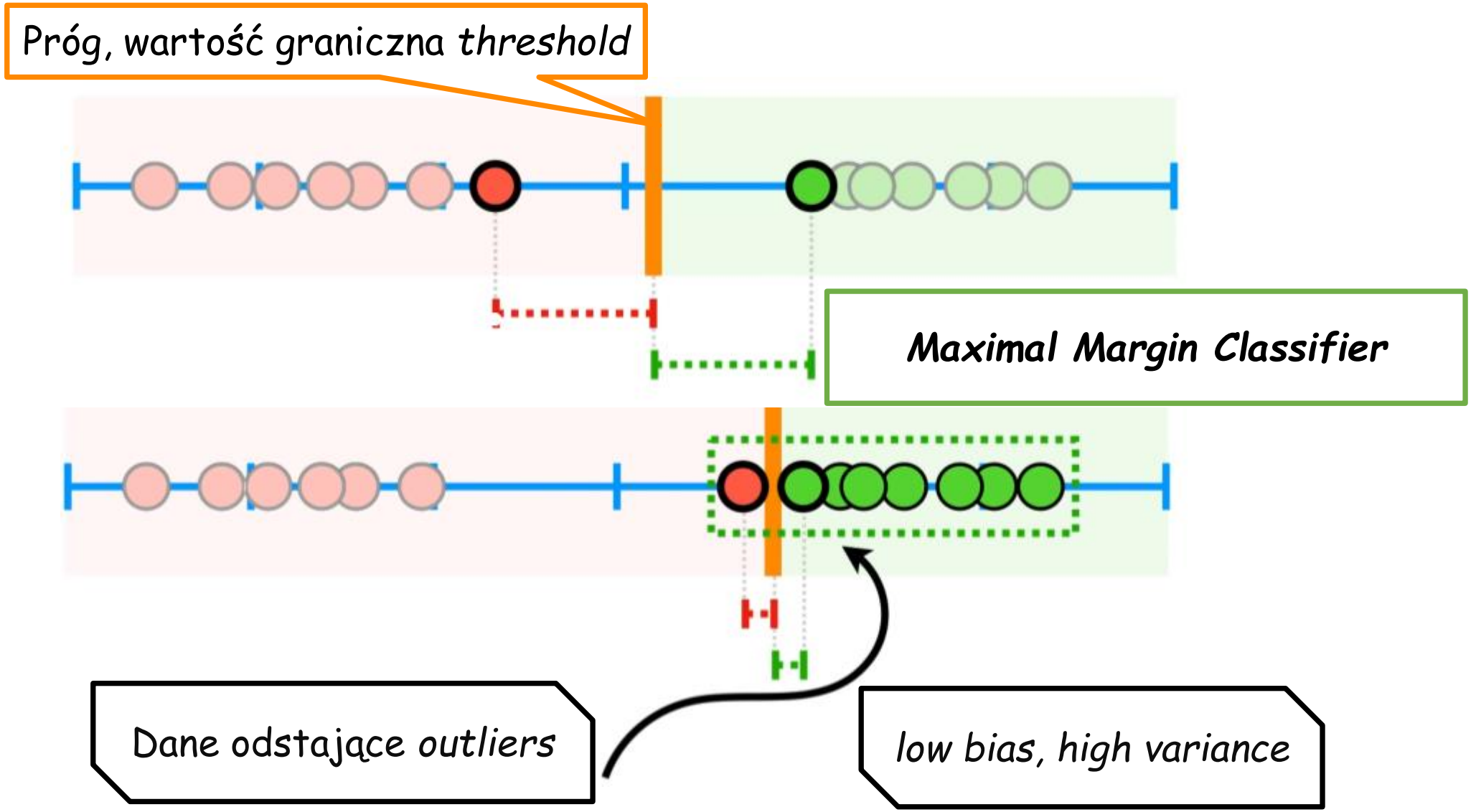
Maszyny wektorów nośnych (Support Vector Machines, SVM)

- znalezienie **optymalnego zestawu hiperpłaszczyzn**, które mogą skutecznie rozdzielić dane należące do różnych klas lub przewidywać wartości w przypadku regresji.
- **maksymalizacja marginesu między hiperpłaszczyzną** a najbliższymi punktami danych różnych klas, zwanych **wektorami nośnymi**.
- w przypadku **nieliniowych zbiorów danych**, SVM może stosować tzw. **funkcje jądra**, które pozwalają na wyznaczenie bardziej skomplikowanych granic decyzyjnych.



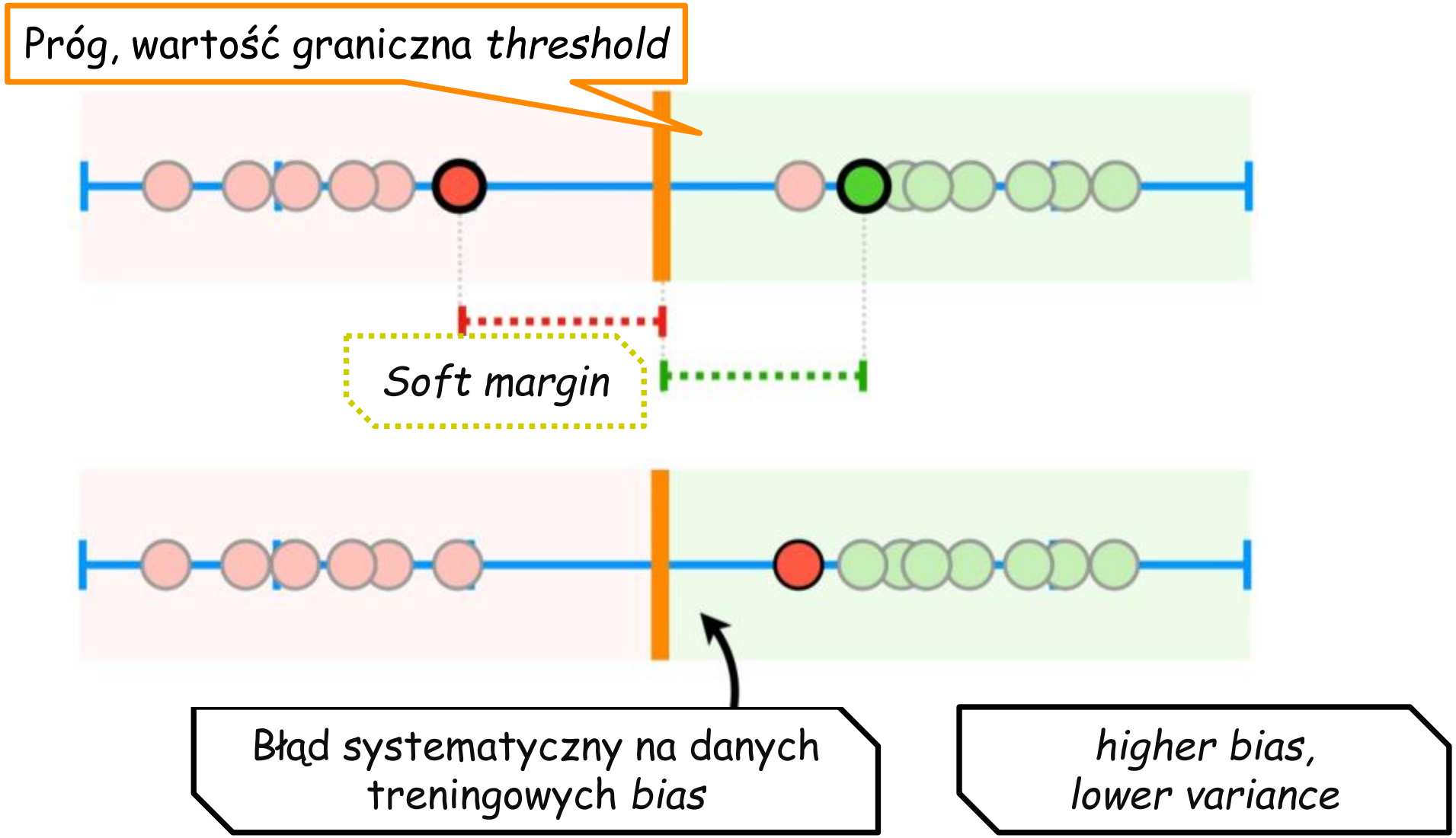
https://datatron.com/wp-content/uploads/2021/05/Support-Vector-Machine.png

Maszyny wektorów nośnych (Support Vector Machines, SVM)



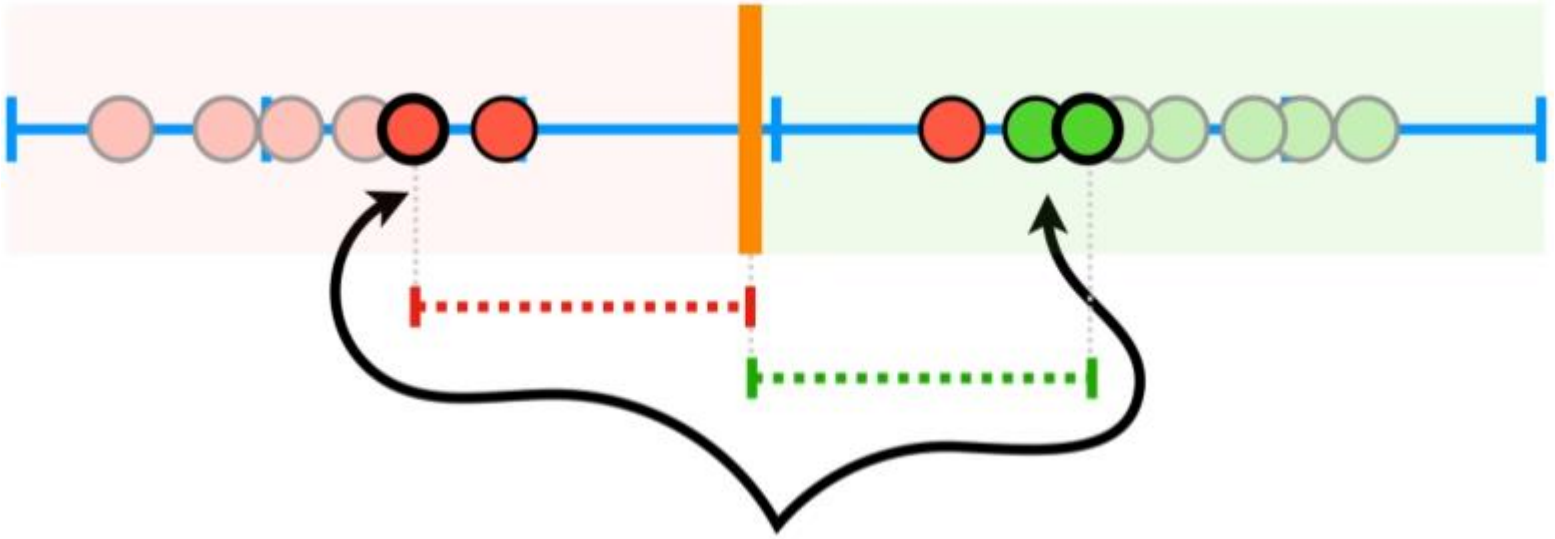
https://www.youtube.com/watch?v=efR1C6CvhmE Support Vector Machines, StatQuest with Josh Starmer

Maszyny wektorów nośnych (Support Vector Machines, SVM)



https://www.youtube.com/watch?v=efR1C6CvhmE Support Vector Machines, StatQuest with Josh Starmer

Maszyny wektorów nośnych (Support Vector Machines, SVM)



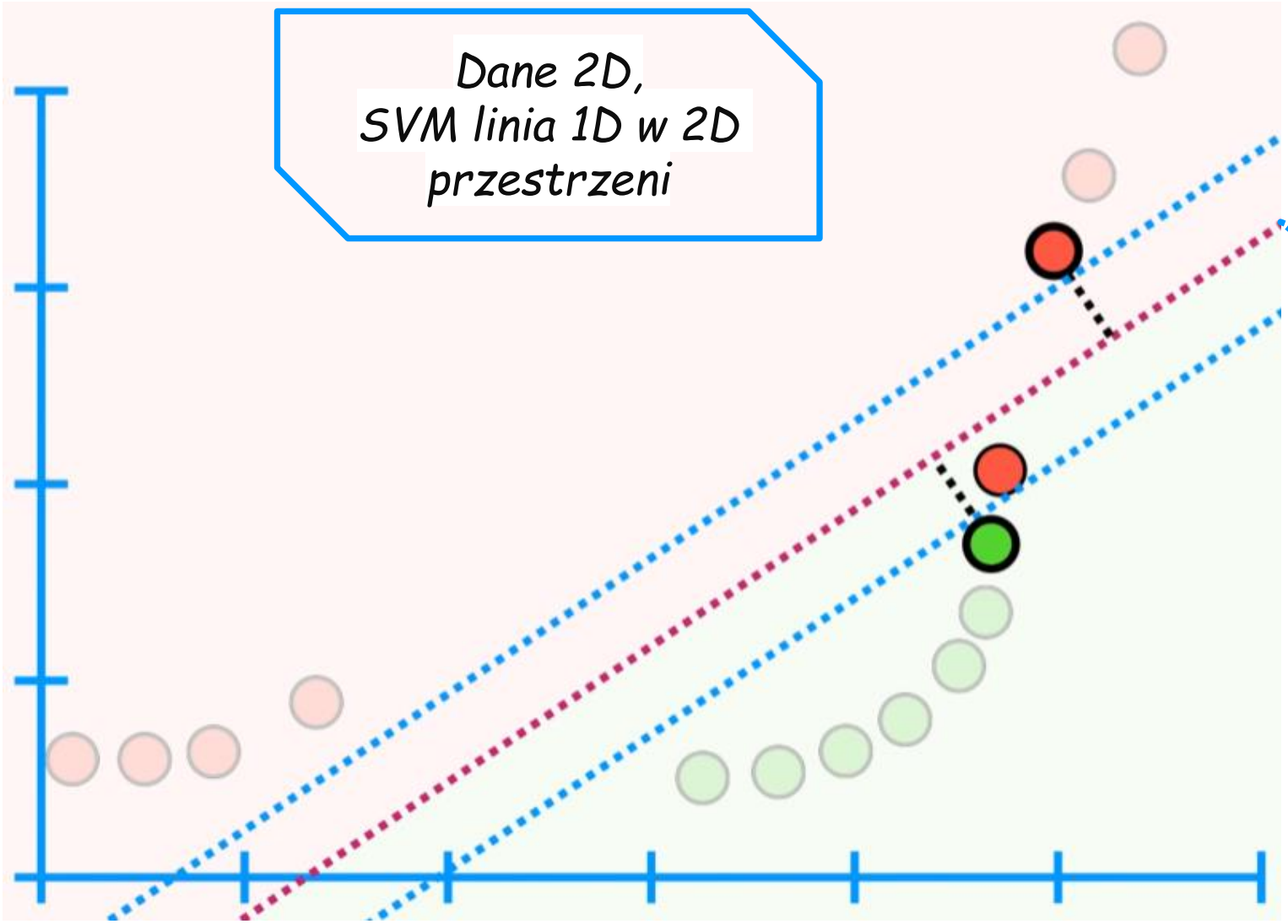
Dane na krawędzi i wewnątrz *soft margin* nazywamy wektorami nośnymi/wspierającymi *support vectors*

Walidacja krzyżowa w celu ustalenia *soft margin*

Soft Margin Classifier
aka
Support Vector Classifier

https://www.youtube.com/watch?v=efR1C6CvhmE Support Vector Machines, StatQuest with Josh Starmer

Maszyny wektorów nośnych (Support Vector Machines, SVM)



Granica decyzji,
soft margin

"**Overlapping**" klasy w danych treningowych nakładają się. Nie ma jednoznacznej granicy decyzyjnej (hiperpłaszczyzny)

https://www.youtube.com/watch?v=efR1C6CvhmE Support Vector Machines, StatQuest with Josh Starmer

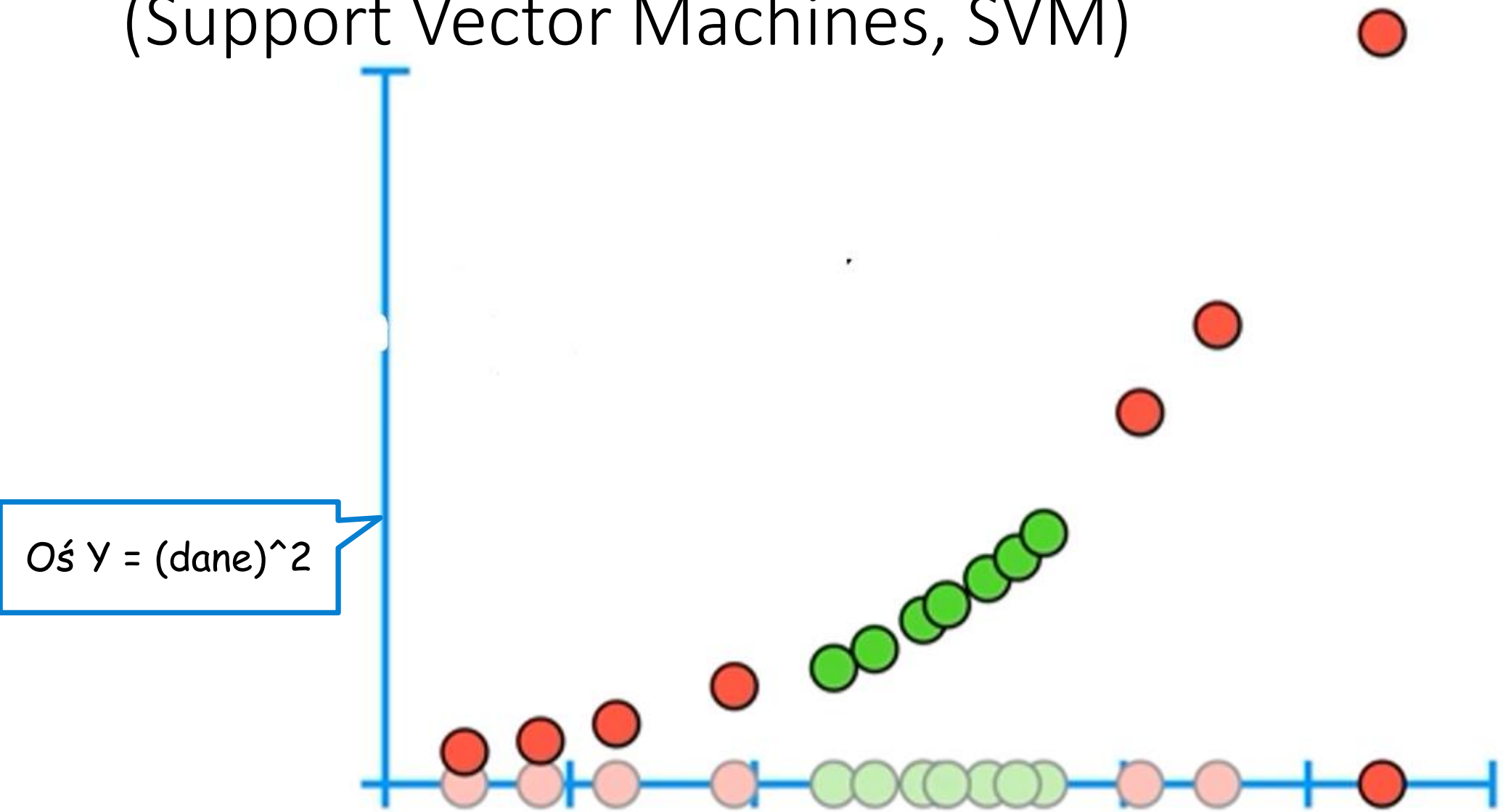
Maszyny wektorów nośnych (Support Vector Machines, SVM)

"**Overlapping**" klasy w danych treningowych nakładają się.
Nie ma jednoznacznej granicy decyzyjnej (hiperpłaszczyzny)

Support Vector Classifier → ***Support Vector Machines***

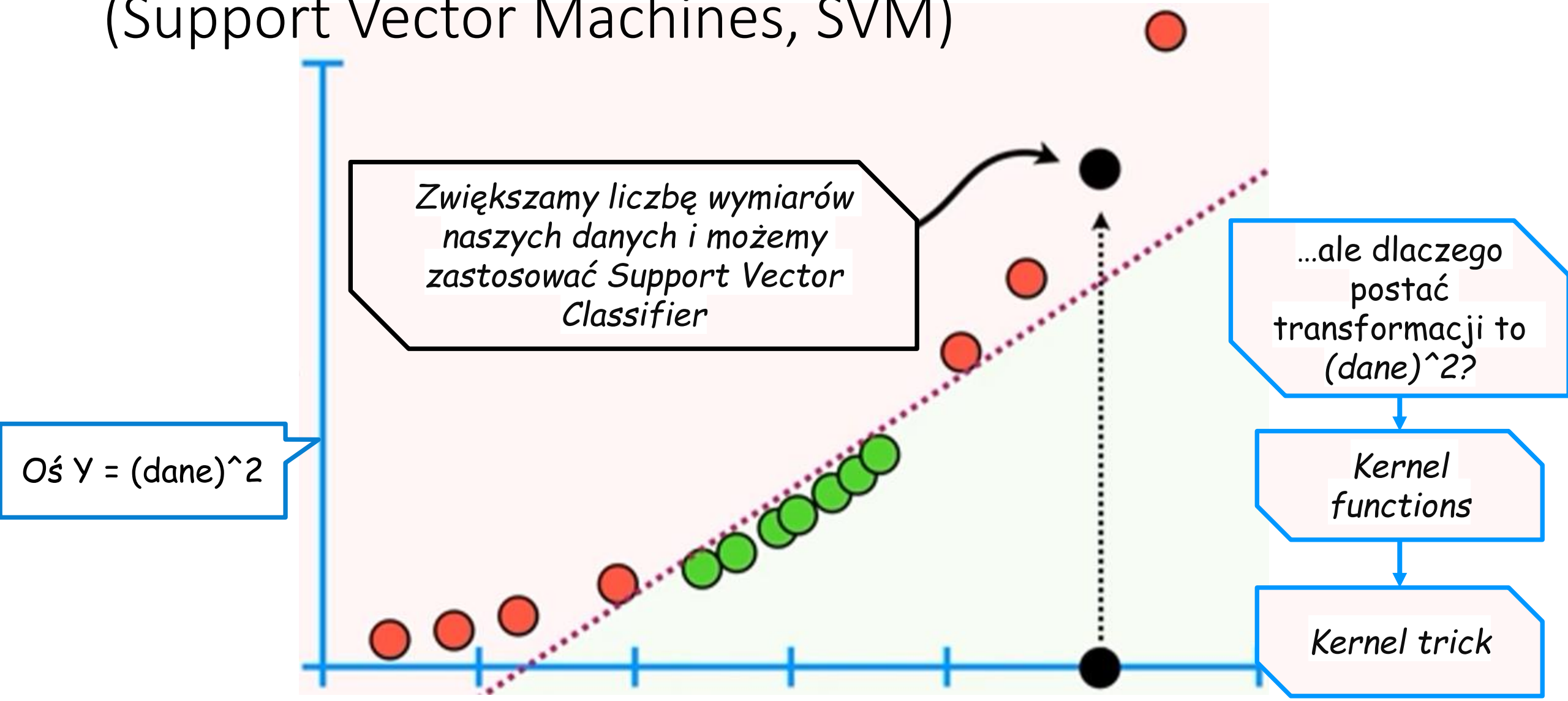
https://www.youtube.com/watch?v=efR1C6CvhmE Support Vector Machines, StatQuest with Josh Starmer

Maszyny wektorów nośnych (Support Vector Machines, SVM)



https://www.youtube.com/watch?v=efR1C6CvhmE Support Vector Machines, StatQuest with Josh Starmer

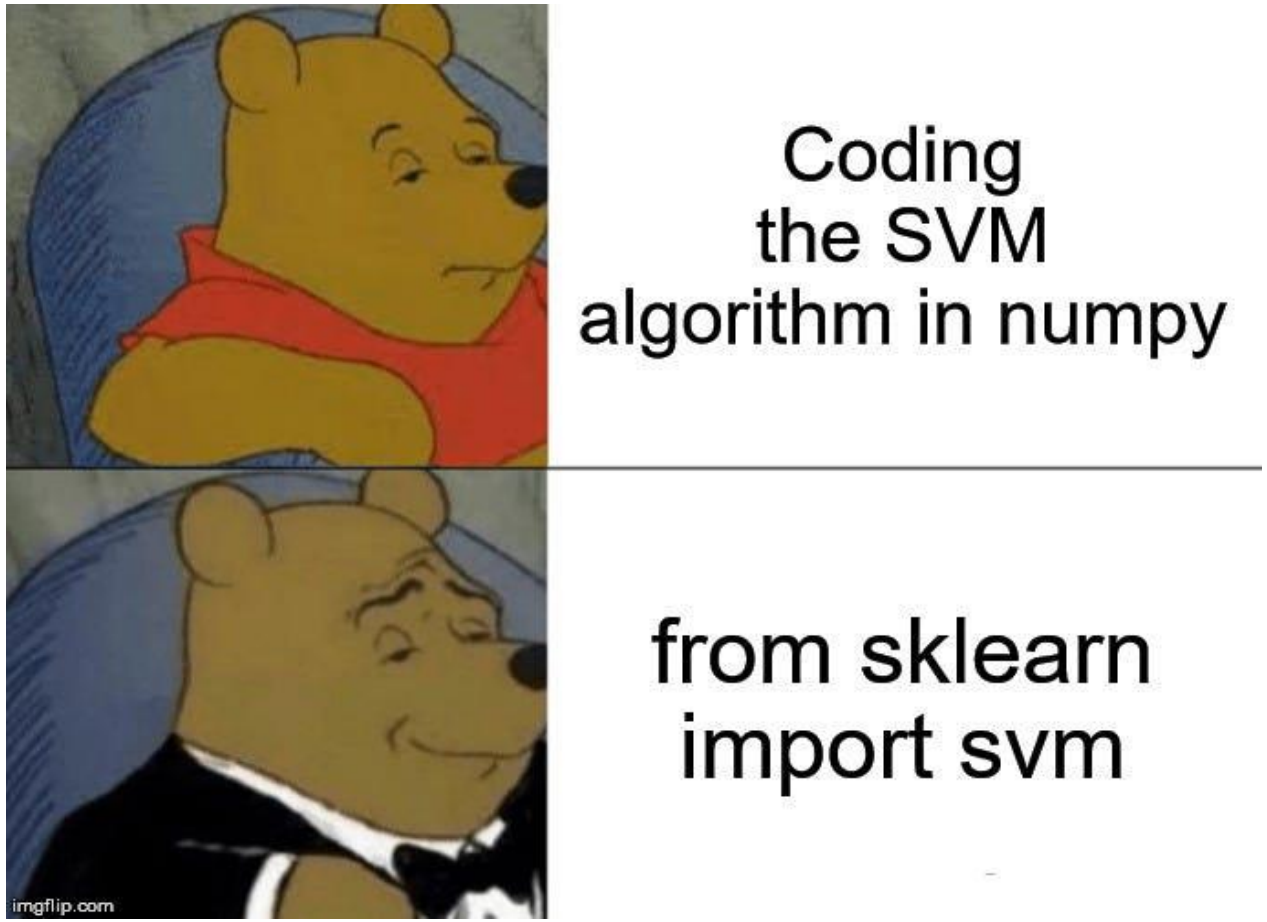
Maszyny wektorów nośnych (Support Vector Machines, SVM)



https://www.youtube.com/watch?v=efR1C6CvhmE Support Vector Machines, StatQuest with Josh Starmer

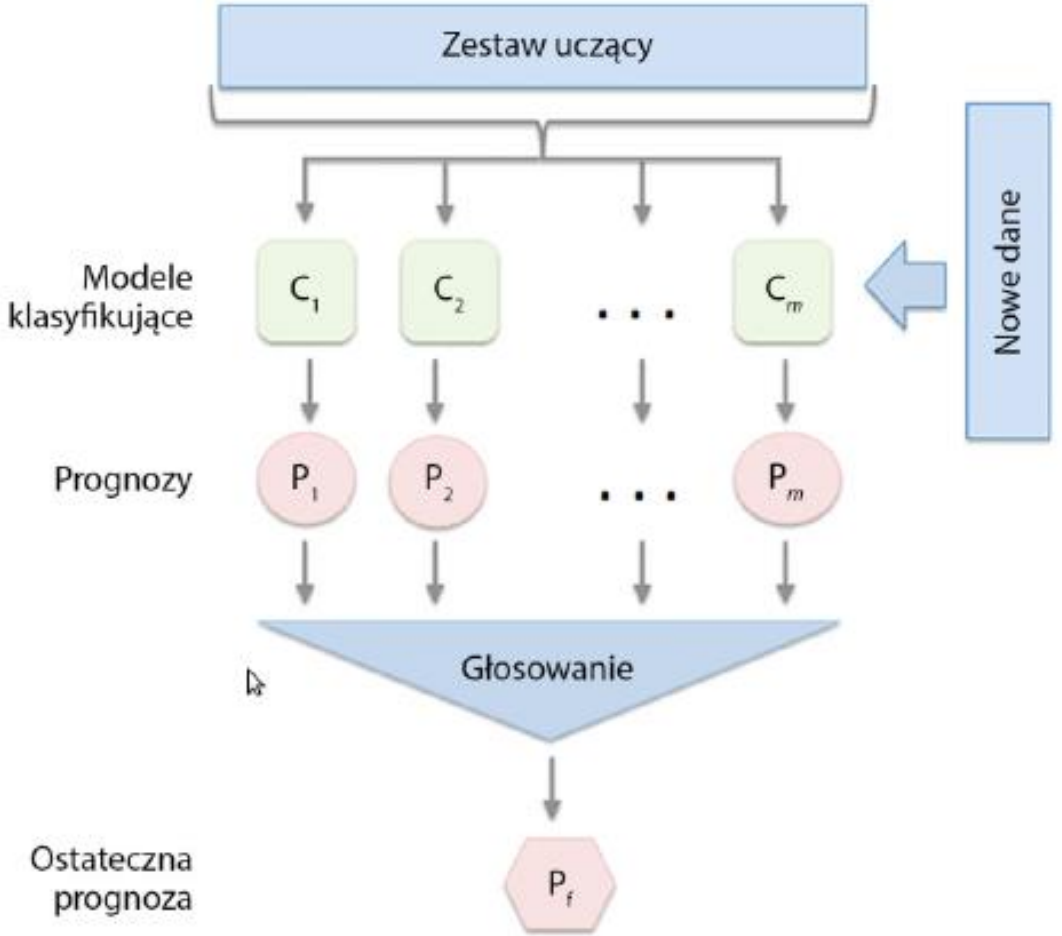
Kodujemy

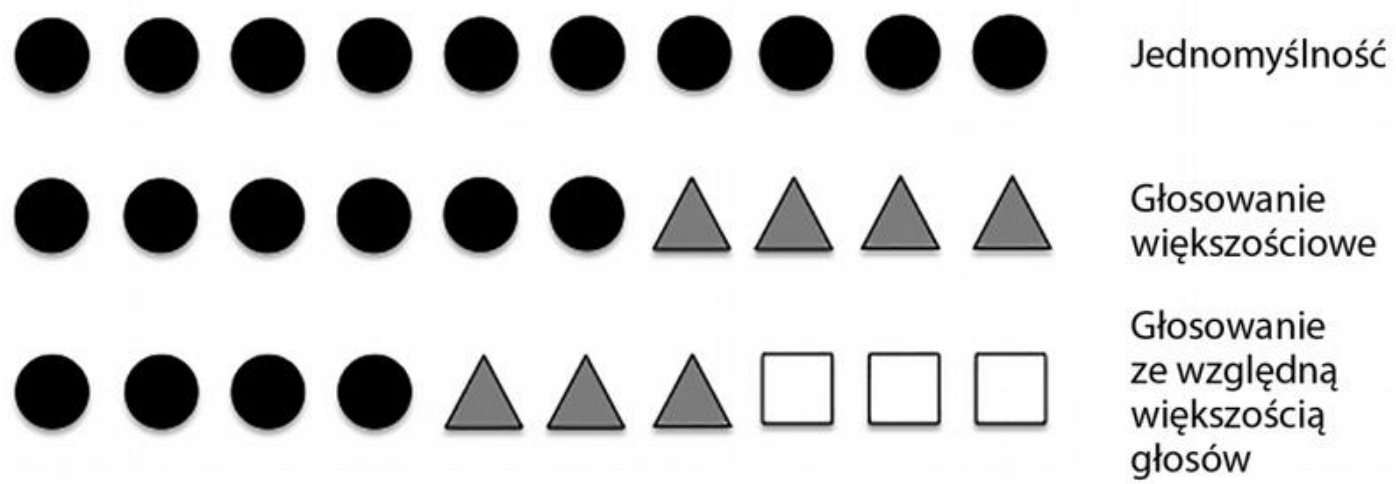
public/mmajew/MIW/05/
02_SVM_log reg.py



Uczenie zespołów

Celem metod zespołowych (*ensemble methods*) jest łączenie różnych klasyfikatorów w jeden meta klasyfikator wykazujący większa skuteczność uogólniania niż każdy ze składowych algorytmów.





Projekt 3 a. (na 5 pkt.)

Szkic: public/mmajew/MIW/05/03_VotingClassifier.py

Projekt wykonaj z wykorzystaniem pakietu scikit-learn

1. Stwórz zbiór danych za pomocą funkcji make_moons(n_samples = 10000, noise = 0.4).

2. Rozdziel uzyskany zestaw danych na podzbiory uczący i testowy przy użyciu metody train_test_split().

3.a Wytrenuj klasyfikatory LogisticRegression, SVM oraz RandomForestClassifier i połącz klasyfikatory SVM, LogisticRegression oraz RandomForestClassifier **w jeden zespół (VotingClassifier).**

4. Oceń osiągnięte rezultaty: wyświetl dokładność trenowania i testowania, narysuj kontur decyzji na podstawie predykcji VotingClassifier.

Termin 30.04

Drugą część projektu (projekt 3 b. za kolejne 5 pkt podam na kolejnych zajęciach)