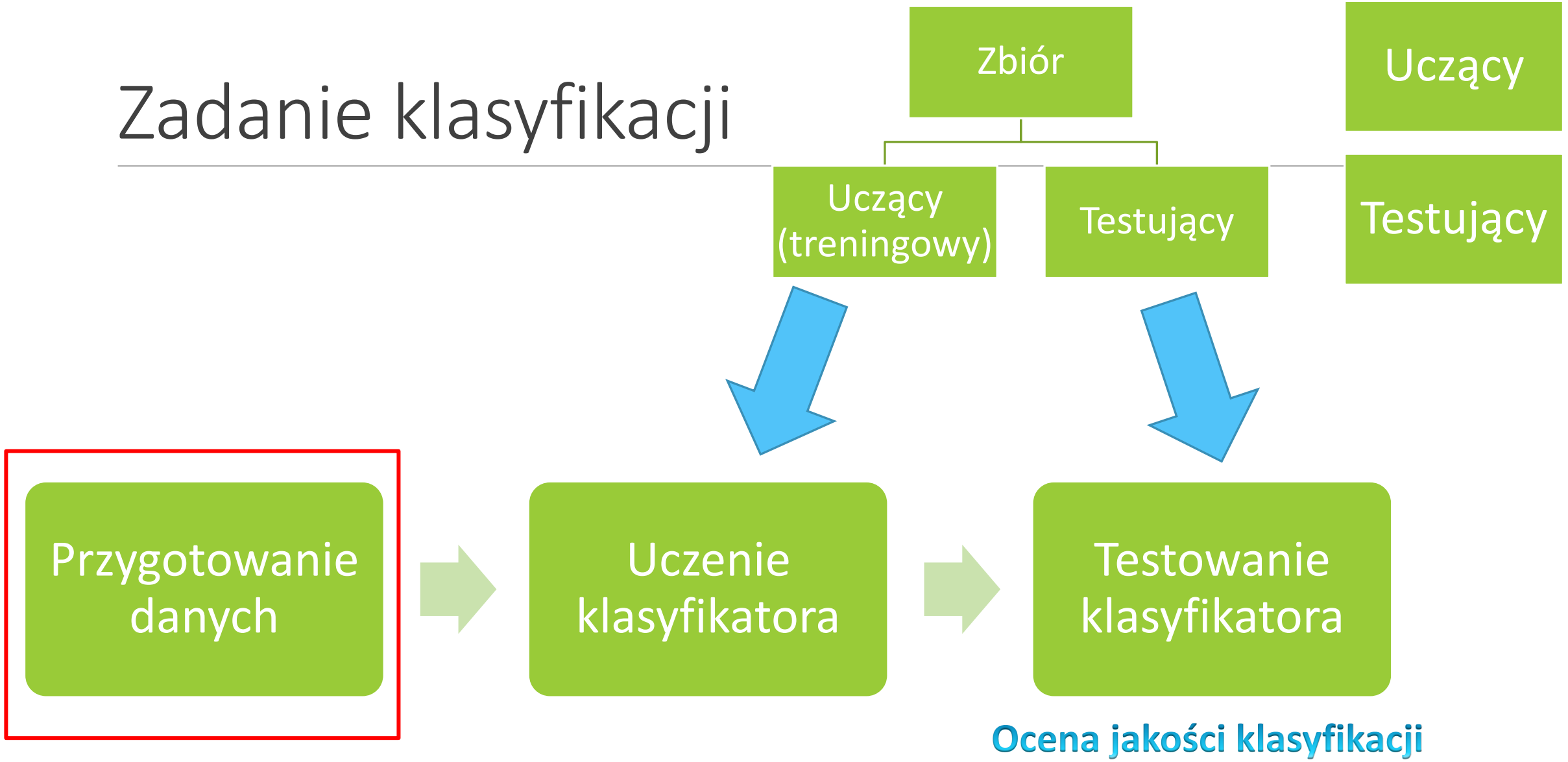
Uczenie maszynowe

PREPROCESSING

Zadanie klasyfikacji



Przygotowanie danych

- Brakujące dane
- Przekształcanie danych - standaryzacja i normalizacja
- Dyskretyzacja danych
- Redukcja wymiarów – miary oparte na entropii
- Redukcja wymiarów – PCA

Larose, Daniel T., and Anna Wilbik. Odkrywanie wiedzy z danych: wprowadzenie do eksploracji danych. Wydawnictwo Naukowe PWN, 2013.

Brakujące dane

1. Zastąpienie brakującej wartości pewną stałą, zadaną przez analityka.
2. Zastąpienie brakującej wartości średnią wartością (dla zmiennych liczbowych) lub modalną (dla zmiennych jakościowych).
3. Zastąpienie brakującej wartości losowo wygenerowaną wartością z obserwowanego rozkładu atrybutu.

Atrybut1	Atrybut2	Atrybut3	Klasa
1.2	200.4	?	„1”

Przekształcanie danych

- Normalizacja min-max dla X:

$$X^* = \frac{X - \min(X)}{zakres\ X} = \frac{X - \min(X)}{\max(X) - \min(X)}$$

- Standaryzacja dla X:

$$X^* = \frac{X - średnia(X)}{\sigma(X)}$$

Problem: czułość na strukturę danych

Rozwiązanie - standaryzacja atrybutów (zmienna uzyskuje średnią wartość oczekiwaną zero i odchylenie standardowe jeden)

Standaryzacja Z (najczęstsza):

$$z = \frac{x - \mu}{\sigma}$$

gdzie: x – zmienna niestandaryzowana, μ – średnia z populacji, σ – odchylenie standardowe populacji.

https://pl.wikipedia.org/wiki/Standaryzacja_(statystyka)

Problem zmiennych numerycznych

- Gra w golfa w zależności od pogody
- Atrybuty:

- Temperatura

- Wilgotność

- czyGramy? Play {yes, no}

Jak policzyć p-stwo wystąpienia poszczególnych wartości np. temperatury 69?

Rozwiązanie – dyskretyzacja!

Temperature	Humidity	Play
85	85	no
80	90	no
65	70	no
72	95	no
71	80	no
83	78	yes
70	96	yes
68	80	yes
64	65	yes
69	70	yes
75	80	yes
75	70	yes
72	90	yes
81	75	yes

http://www.simafore.com/blog/bid/107702/2-ways-of-using-Naive-Bayes-classification-for-numeric-attributes

Czy to dobry pomysł, aby podzielić temperaturę na wysoką/średnią/niską?

Nie, bo co to znaczy wysoka temperatura?

Dyskretyzacja danych

- Cel: zastąpienie ciągłych wartości atrybutu atrybutem o wartościach dyskretnych
- Podział: metody **prymitywne** i zaawansowane

Różnice: w prymitywnych nie uwzględnia się rozkładu wartości atrybutów z zbiorze uczącym, tylko dzieli zakres na ustaloną liczbę równych przedziałów + nie bierze się pod uwagę przynależności przypadków do danych klas

- Prymitywne metody dyskretyzacji:

- wg równej szerokości
- wg równej częstości

- FILM: https://www.youtube.com/watch?v=EH9IDjVm3I4

Dyskretyzacja wg równej szerokości

- Podział zakresu na stała liczbę *k* równych przedziałów (powstanie zbioru przedziałów)
- jeden przedział = jedna wartość dyskretna
- Szerokość przedziału:

$$W = \frac{max_A - min_A}{k}$$

- Zakresy przedziałów:

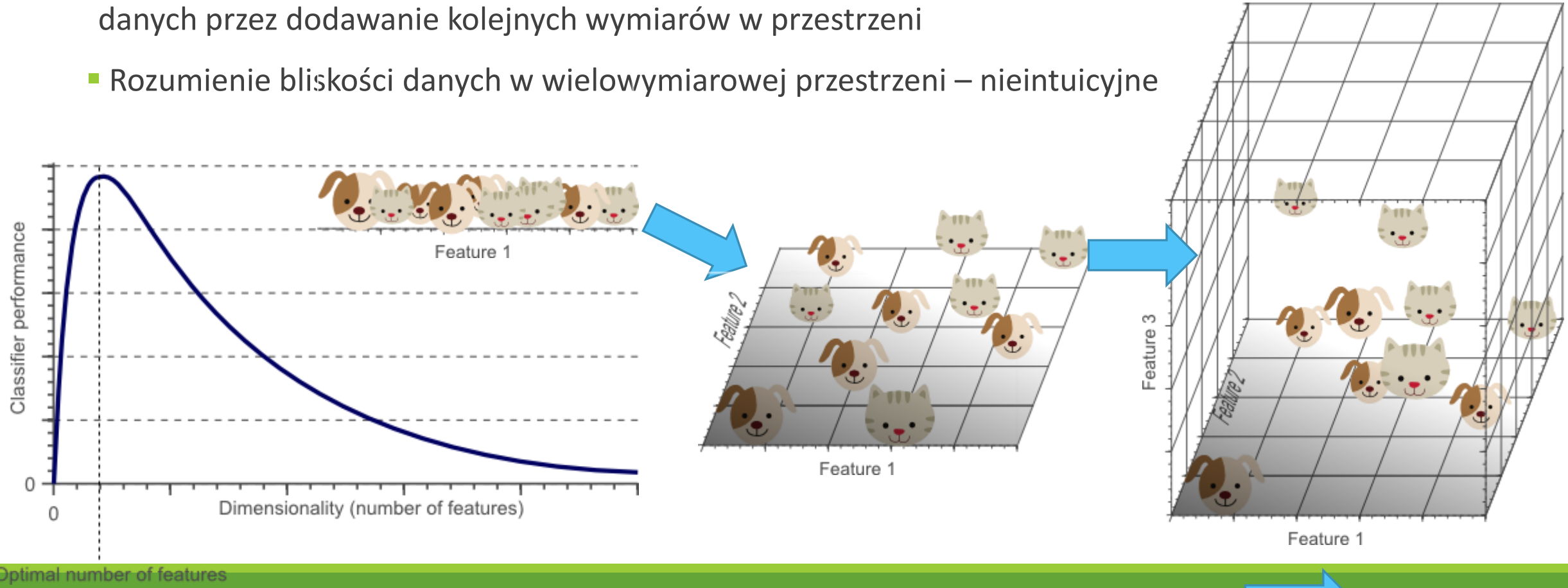
$$min_A + W; min_A + 2\mathrm{W}, \ldots, min_A + (k-1)W$$

Dyskretyzacja o równej częstości

- W lepszym stopniu uwzględnia charakter danych
- Ustalona z góry liczba przedziałów
- Dobór końców przedziałów: aby w każdym z nich znajdowała się mniej więcej taka sama liczba przypadków
- Metoda: podział zbioru na k możliwie równolicznych podzbiorów
- Dla obu metod prymitywnych najlepiej wykonać histogram przed podziałem atrybutu

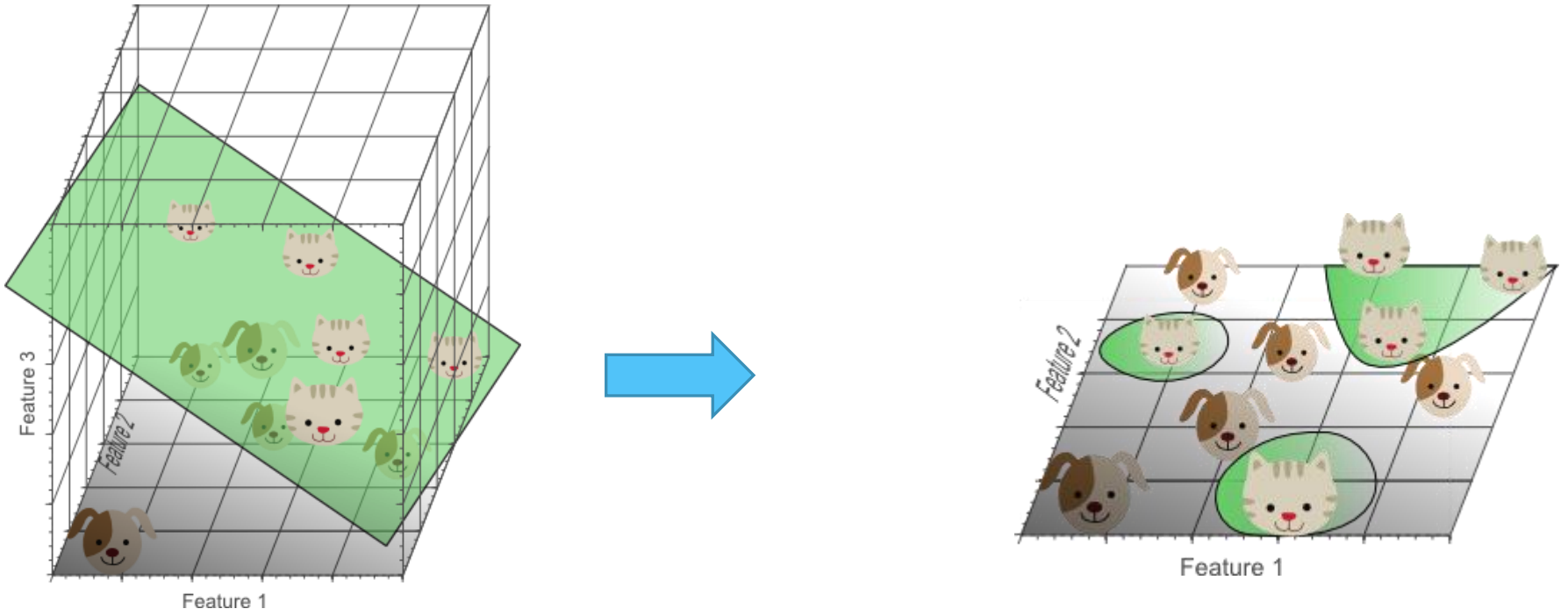
Redukcja wymiaru danych – selekcja cech

- Problem klątwy wymiarowości (ang. *curse of dimensionality*) – wykładniczy wzrost objętości danych przez dodawanie kolejnych wymiarów w przestrzeni
- Rozumienie bliskości danych w wielowymiarowej przestrzeni – nieintuicyjne



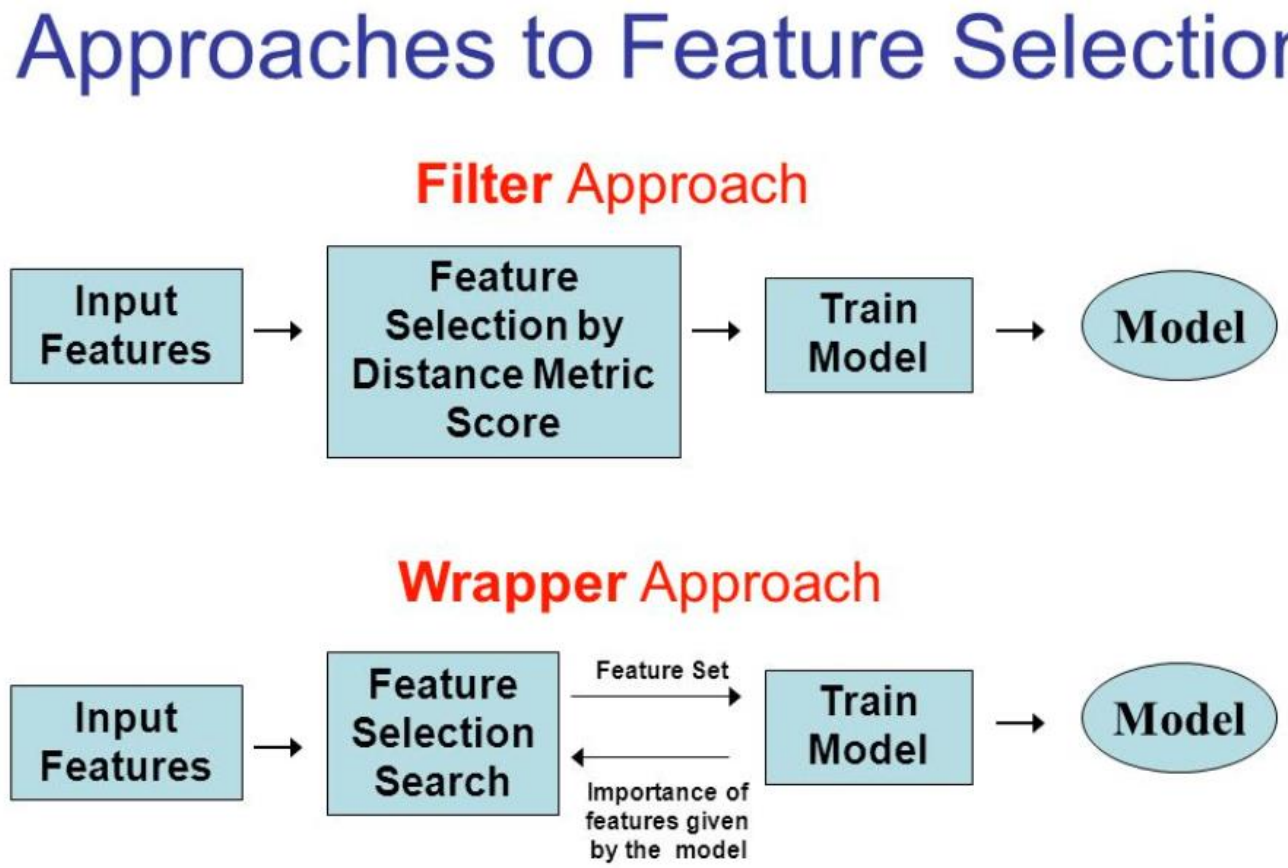
Rysunek: http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/

Redukcja wymiaru danych – selekcja cech



Rysunek: http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/

Podejścia do selekcjo cech



Rysunek: http://slideplayer.com/slide/5202605/

Redukcja danych – miary oparte o teorię informacji

- W selekcji cech stosuje się pojęcie entropii dla prawdopodobieństwa *p* zdarzenia *x*:

$$H(X) = -\sum_{x \in \mathbb{X}} p(x) \log p(x)$$

- oraz entropii warunkowej dla zdarzeń *X* i *Y* :

$$H(X|Y) = \sum_{y \in \mathbb{Y}} p(y) H(X|y)$$

- Względny zysk informacyjny (*gain ratio*) atrybutu:

$$GainRatio(Klasa, Atrybut) = \frac{H(Klasa) - H(Klasa|Atrybut)}{H(Atrybut)}$$

- Zysk informacyjny (*info gain*) atrybutu:
- Ocena osobno każdego z atrybutów

$$InfoGain(Klasa, Atrybut) = H(Klasa) - H(Klasa|Atrybut)$$

- Zalety: dobra generalizacja, minusy: duża złożoność obliczeniowa

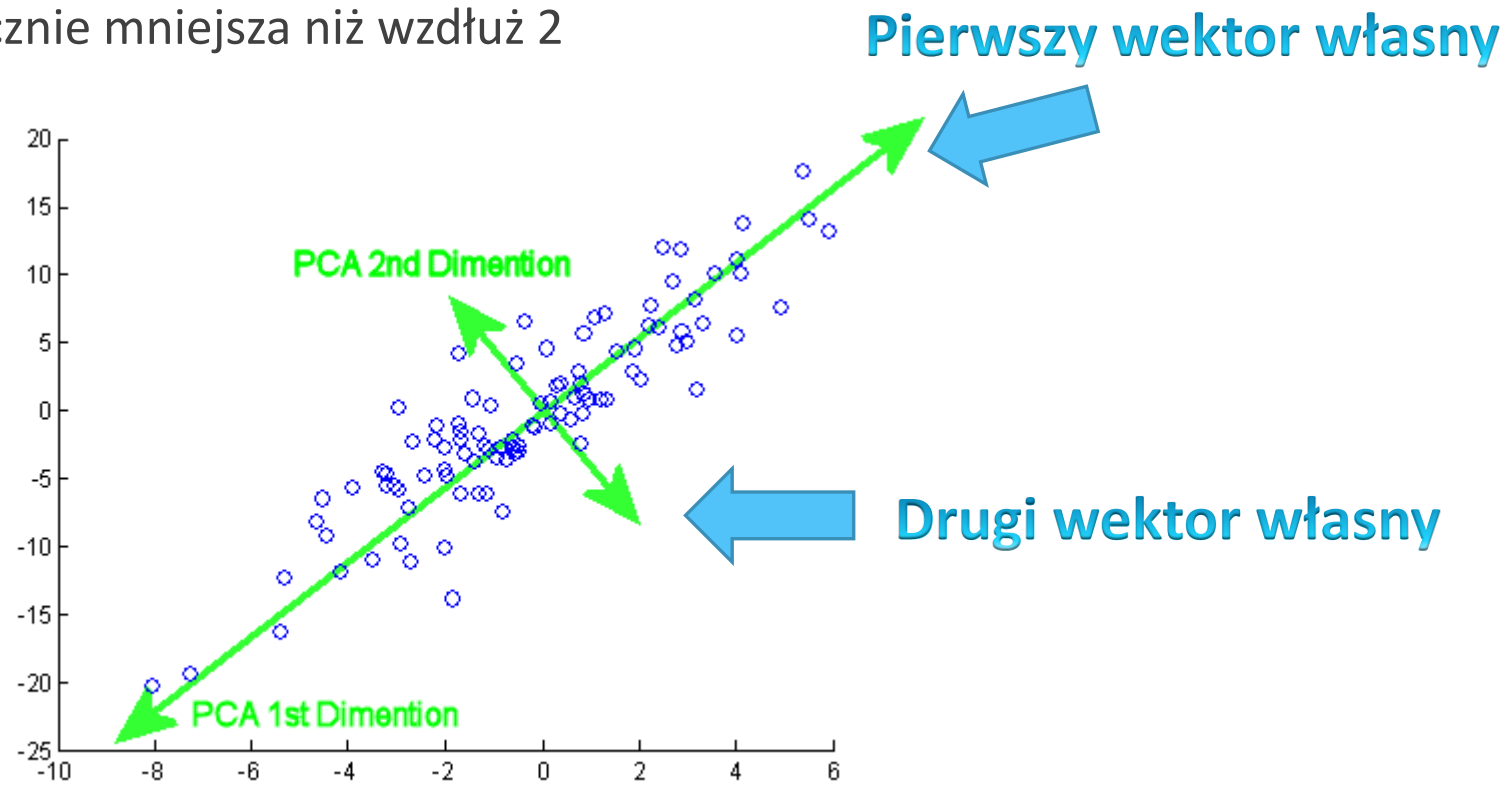
Rysunek: http://www.visiondummy.com/2014/04/curse-dimensionality-affect-classification/

Redukcja wymiaru danych - PCA

- PCA (*Principal Component Analysis)* - analiza głównych składowych
- Statystyczna metoda analizy czynnikowej:
  zbiór danych składających się z N obserwacji, a każda obejmuje K zmiennych można zinterpretować jako chmurę N punktów w przestrzeni K-wymiarowej
- Cel PCA: taki obrót układu współrzędnych, aby zmaksymalizować w pierwszej kolejności wariancję pierwszej współrzędnej, następnie drugiej itd.....
- Przekształcone wartości współrzędnych to składowe główne (początkowe wyjaśniają najwięcej zmienności)
- W uczeniu maszynowym służy do zmniejszania wymiaru danych

PCA - interpretacja

Zmienność wzdłuż 1 znacznie mniejsza niż wzdłuż 2



Rysunek: https://www.quora.com/How-do-I-interpret-the-results-of-a-PCA-analysis

PCA

1. Standaryzacja danych.
2. Obliczenie wektorów własnych (*eigenvectors*) z macierzy kowariancji/korelacji.
3. Sortowanie wartości własnych (*eigenvalues*) malejąco i wybranie *k* najwyższych wartości (*k* stanie się wymiarem nowej przestrzeni cech).
4. Konstrukcja macierzy projekcji *W* z wybranych *k* wektorów własnych.
5. Transformacja oryginalnego zbioru danych *X* przez *W* do *k*-wymiarowej podprzestrzeni *Y*.

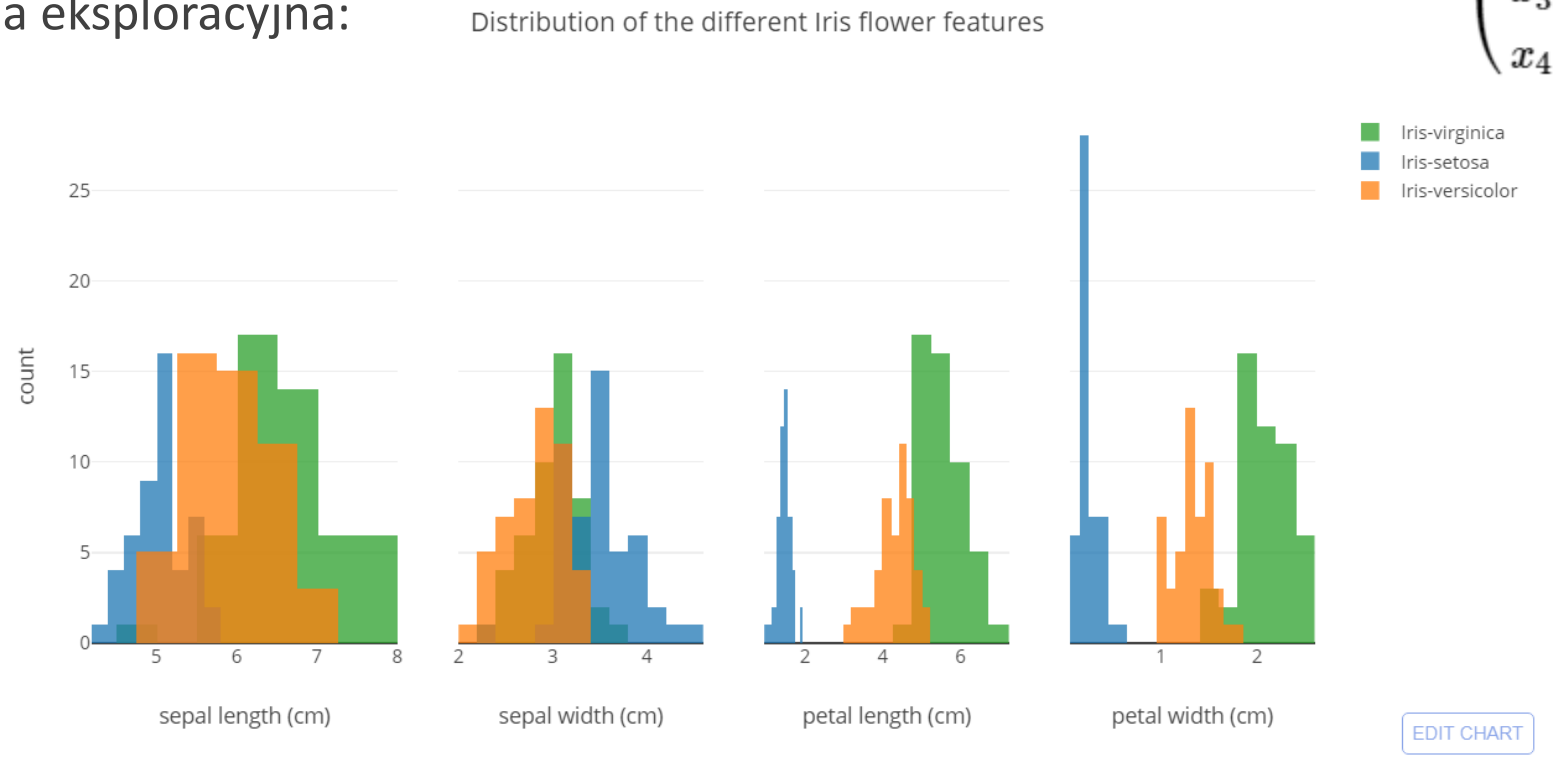
https://plot.ly/ipython-notebooks/principal-component-analysis/#pca-and-dimensionality-reduction

PCA – przykład dla zbioru Iris

- Zbiór o wymiarze 150x4 – kolumny cechy, a rzędy przykłady kwiatów. Transpozycja macierzy zawierającej zbiór danych:

$$\mathbf{x}^{\mathbf{T}} = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} = \begin{pmatrix} \text{sepal length} \\ \text{sepal width} \\ \text{petal length} \\ \text{petal width} \end{pmatrix}$$

- Wizualizacja eksploracyjna:



https://plot.ly/ipython-notebooks/principal-component-analysis/#pca-and-dimensionality-reduction

PCA – przykład dla zbioru Iris

- Standaryzacja danych ze względu na maksymalizowanie wariancji wzdłuż osi
  W przypadku Iris wszystkie miary są w cm, więc transformuje się je do postaci średnia zero i wariancja jeden.

- Dekompozycja macierzy – macierz kowariancji o wymiarach *d* x *d* (każdy element odpowiada kowariancji między dwiema cechami)
  Kowariancja między dwiema cechami:

$$\sigma_{jk} = \frac{1}{n-1} \sum_{i=1}^{N} \left(x_{ij} - \bar{x}_j\right)\left(x_{ik} - \bar{x}_k\right)$$

- Sumowanie obliczeń macierzy kowariancji na podstawie równania:

$$\Sigma = \frac{1}{n-1}\left((\mathbf{X} - \bar{\mathbf{x}})^T (\mathbf{X} - \bar{\mathbf{x}})\right) \qquad \bar{\mathbf{x}} = \sum_{k=1}^{n} x_i$$

Średni wektor ma wymiar *d* x *d* a każda wartość wektora odpowiada średniej próbki w kolumnie cech.

https://plot.ly/ipython-notebooks/principal-component-analysis/#pca-and-dimensionality-reduction

PCA – przykład dla zbioru Iris

- Wektory i wartości własne (macierz korelacji)
- Sortowanie wartości własnych malejąco

```
Eigenvalues in descending order:
2.91081808375
0.921220930707
0.147353278305
0.0206077072356
```

```
Eigenvectors
[[ 0.52237162 -0.37231836 -0.72101681  0.26199559]
 [-0.26335492 -0.92555649  0.24203288 -0.12413481]
 [ 0.58125401 -0.02109478  0.14089226 -0.80115427]
 [ 0.56561105 -0.06541577  0.6338014   0.52354627]]

Eigenvalues
[ 2.91081808  0.92122093  0.14735328  0.02060771]
```

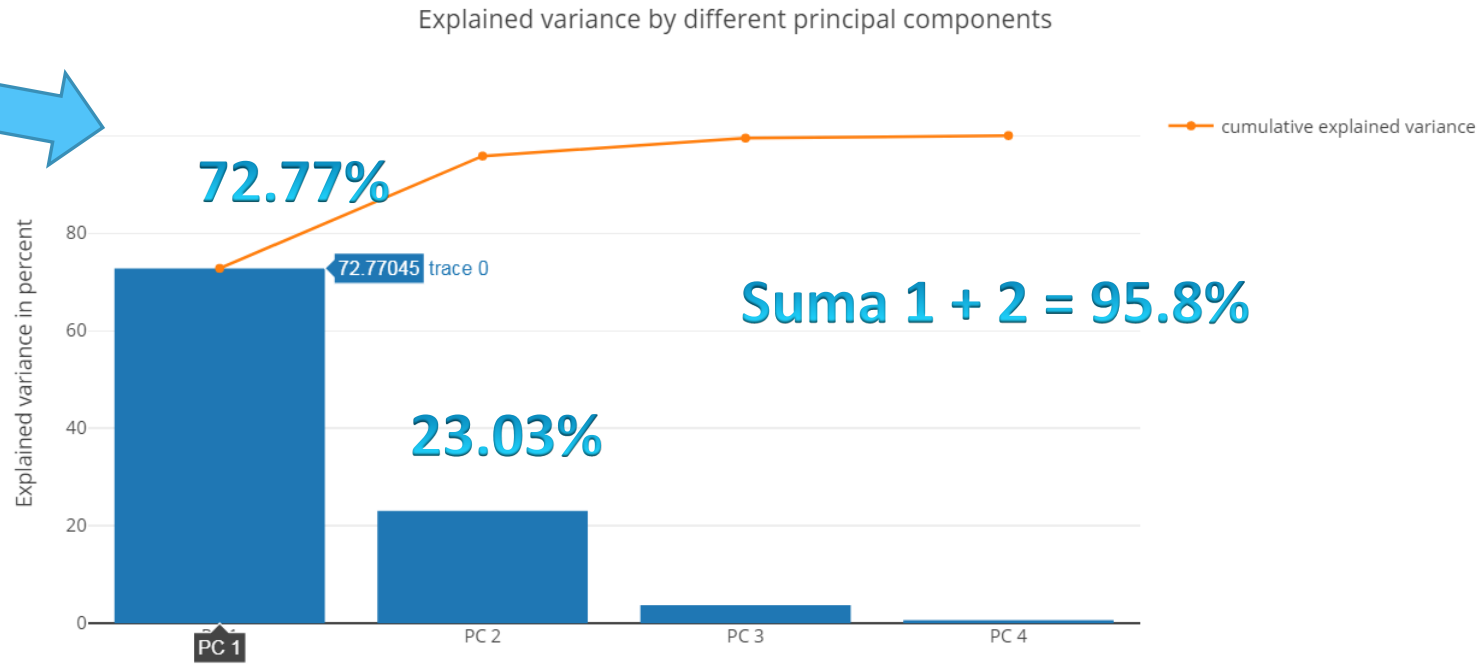
- Wybór składowych głównych – ile? → miara „explained variance” liczona na podstawie wartości własnych (ile wariancji przypisuje się składowej głównej)

https://plot.ly/ipython-notebooks/principal-component-analysis/#pca-and-dimensionality-reduction

PCA – przykład dla zbioru Iris

- „Variance explained”

95.8% informacji zawierają dwie główne składowe



- Konstrukcja macierzy projekcji do transformacji zbioru do nowej podprzestrzeni cech
  Ta macierz jest konkatenacją *k* wektorów własnych

https://plot.ly/ipython-notebooks/principal-component-analysis/#pca-and-dimensionality-reduction

PCA – przykład dla zbioru Iris

- Redukcja czterech wymiarów do dwóch – wybierając dwa wektory własne

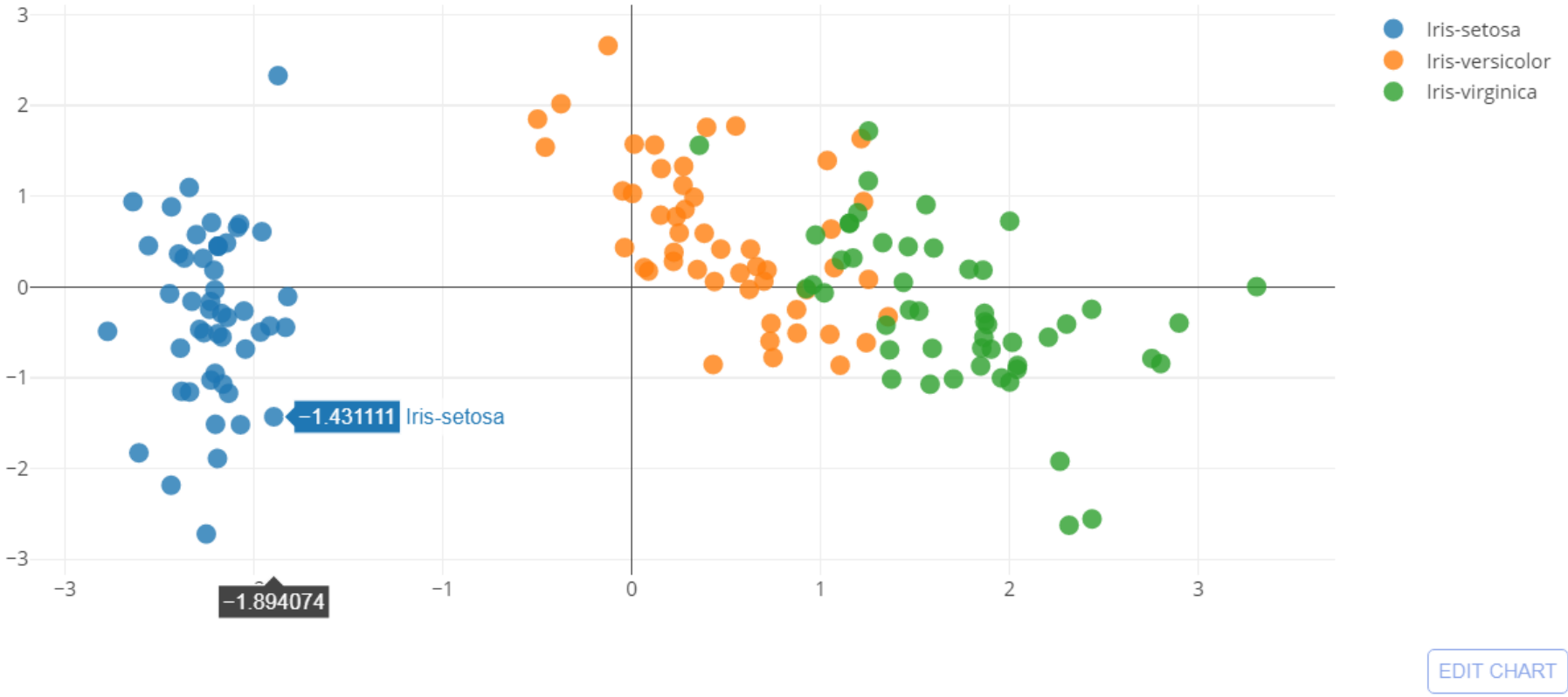
```
Eigenvectors
[[ 0.52237162 -0.37231836 -0.72101681  0.26199559]
 [-0.26335492 -0.92555649  0.24203288 -0.12413481]
 [ 0.58125401 -0.02109478  0.14089226 -0.80115427]
 [ 0.56561105 -0.06541577  0.6338014   0.52354627]]
```

```
('Matrix W:\n', array([[ 0.52237162, -0.37231836],
       [-0.26335492, -0.92555649],
       [ 0.58125401, -0.02109478],
       [ 0.56561105, -0.06541577]]))
```

- Macierz projekcji ***W*** o wymiarze *4 x 24 x 2* służy do transformacji zbioru na podstawie równania $\mathbf{Y} = \mathbf{X} \times \mathbf{W}$ gdzie ***Y*** jest macierzą o wymiarze *150 x 2* zawierającą przetransformowane przypadki.

https://plot.ly/ipython-notebooks/principal-component-analysis/#pca-and-dimensionality-reduction

PCA – przykład dla zbioru Iris



https://plot.ly/ipython-notebooks/principal-component-analysis/#pca-and-dimensionality-reduction

PCA - uwaga

- PCA oparte na macierzy kowariancji:
  - zmienne w zbiorze wejściowym o największej wariancji - największy wpływ na wynik
  - wskazane dla zmiennych o porównywalnych wartościach
- PCA oparte na macierzy korelacji:
  - wstępna normalizacja zbioru wejściowego – wszystkie zmienne mają na wejściu identyczną wariancję
  - wskazane dla zmiennych o nieporównywalnych wielkościach