# Selekcja atrybutów

## wraz z ekstrakcją cech i elementami przetwarzania wstępnego

Dariusz Brzeziński

# Plan zajęć

- Przetwarzanie wstępne
- Czyszczenie danych
- Transformacje danych
- Selekcja cech
  - Metody typu filter
  - Metody typu wrapper
  - Metody typu embedded
- Redukcja wymiarów

# Przetwarzanie wstępne

- Czyszczenie danych
  - Wykrywanie błędów
  - Nieznane wartości atrybutów
  - Identyfikacja obserwacji odstających
- Transformacje atrybutów
- Redukcja rozmiarów danych
  - Selekcja atrybutów
  - Ekstrakcja cech
  - Wybór obiektów

# Czyszczenie danych

- Wykrywanie błędów
  - Usuwanie duplikatów
  - Idealnie skorelowane kolumny
  - Dane tekstowe zamiast liczbowych
  - Dane spoza zakresu
- Wartości odstające (outliers)
  - Błąd lub rzadka obserwacja
  - Sposoby wykrywania:
    - Statystyczne (np. reguła 3 σ)
    - Grupowanie danych (+ silhouette coefficient)
    - W oparciu o wyniki klasyfikacji

# Czyszczenie danych

- Wartości puste
  - W scikit nie można ich zostawić…
  - Opcja #1: usunąć niepełne przykłady
  - Opcja #2: uzupełnić wartości
    - Użycie globalnej stałej wartości
    - Zastąpienie średnią (w. liczbowe)
    - Zastąpienie najczęściej występującą wartością (w. nominalne)
    - Średnia lub moda dla danej klasy decyzyjnej
    - Użycie wszystkich możliwych wartości atrybutu
    - Użycie wielu wartości wraz z informacją o ich prawdopodobieństwie
    - Inne: regresja, średnie kroczące, filtr Kalmana

# Transformacje danych

- Wygładzanie szumu
- Agregacja (miesiąc -> rok)
- Normalizacja
  - min-max [0-1]
  - z-score [μ, σ]
  - precentyle
- Dyskretyzacja
  - zawsze warto spojrzeć na histogram
  - Wiedza dziedzinowa
- Inżynieria cech

# Redukcja danych

- Curse of dimensionality (Bellman 1961)

  *"the number of samples required per variable increases exponentially with the number of variables"*

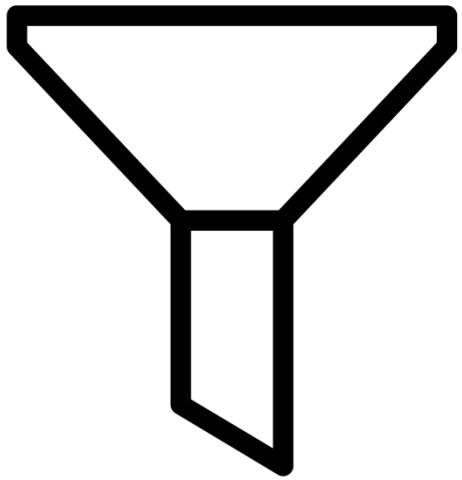
- Cele redukcji danych
  - zmniejszenie wymagań pamięciowych
  - przyspieszenie działania algorytmów
  - poprawa zdolności predykcyjnych
  - ułatwienie zbierania nowych danych
  - zwiększenie czytelności danych
  - wizualizacja danych

# Redukcja danych - podejścia

- Selekcja cech
  - wybranie podzbioru istniejących cech
  - metody: filter, wrapper, embedded
- Ekstrakcja cech
  - konstrukcja (automatyczna) nowych cech, które zastąpią istniejące
- Wykorzystanie wiedzy dziedzinowej
  - ręczne wprowadzanie nowych cech

# Selekcja cech – metody filter

- Podejścia oceniające każdy atrybut (z reguły) osobno
- Miara ocenia na ile warto zachować daną cechę
- Popularne miary:
  - zmienność wartości (usuń kolumny o małej wariancji)
  - korelacja Pearsona
  - test Chi-kwadrat
  - mutual information
  - relief-F



# Selekcja cech – metody wrapper

- Oceniaj klasyfikator na kolejnych podzbiorach cech
- Przestrzeń rozwiązań = $2^{\text{liczba cech}}$ (!!!)
- Typowe podejścia:
  - forward (stepwise) selection
    - subset={}; zachłannie dodawaj (lub usuwaj) po jednym atrybucie ; powtarzaj dopóki następuje poprawa
  - backward (stepwise) elimination
    - subset=Wszystkie cechy; zachłannie usuwaj (lub dodawaj) po jednym atrybucie; powtarzaj dopóki następuje poprawa
  - random mutation
    - subset=Losowe cechy; zachłannie usuwaj (lub dodawaj) po jednym atrybucie; przerwij po określonej liczbie kroków

# Selekcja cech – metody embedded

- Wykorzystanie charakterystyki algorytmów uczących do oceny atrybutów
- Klasyfikator lub regresor ocenia atrybuty jako „efekt uboczny” procesu uczenia
- Przykłady:
  - metody liniowe z regularyzacją [L1 (LASSO), L2 (Ridge regression), L1+L2 (Elastic Net)]
  - SVM i Random Forest potrafią określić ważność atrybutów
  - Recursive Feature Elimination (RFE)
  - Stability selection

# Selekcja cech – metody embedded

- Regularyzacja L1 (LASSO)
  - minimalizuje błędy bezwzględne (MAE)
  - w efekcie zeruje (usuwa) niektóre atrybuty
- Regularyzacja L2 (Ridge regression)
  - minimalizuje błędy kwadratowe (MSE)
  - w efekcie zmniejsza wagi atrybutów

# Selekcja cech – metody embedded

- Metody liniowe „gubią się”, gdy atrybuty są ze sobą mocno skorelowane

- Opcja #1: zostaw tylko jeden ze skorelowanych cech
- Opcja #2: stability selection:
  1. Wylosuj podzbiór cech
  2. Oceń cechy za pomocą metody liniowej (np. LASSO)
  3. Powtarzaj przez określoną liczbę iteracji
  4. Uśrednij ważność atrybutów

# Selekcja cech – metody embedded

- Recursive Feature Elimination:
  1. Naucz klasyfikator
  2. Wylicz ranking cech
  3. Usuń najniżej ocenioną(e) cechę(y)
  4. Powtarzaj aż otrzymasz żądaną liczbę cech

- Algorytm zachłanny
- Mieszanka metod filter/embedded i wrapper
- Kompromis między kosztem a efektem
- W połączeniu z CV automatycznie określa liczbę cech

# Selekcja cech

- **Metody filter są z reguły szybkie** i dobrze wybierają istotne atrybuty, ale nie optymalizują zdolności predykcyjnych
- **Metody wrapper poprawiają predykcje** ale są bardzo wolne i trzeba je odpalać dla każdego algorytmu osobno
- **Metody embedded można stosować tylko do wybranych algorytmów** i oferują pewien kompromis

# Selekcja cech

- Podejście pragmatyczne:
  1. zacznij od metod typu filter
  2. zostaw większy zbiór atrybutów
  3. odpal metodę typu wrapper

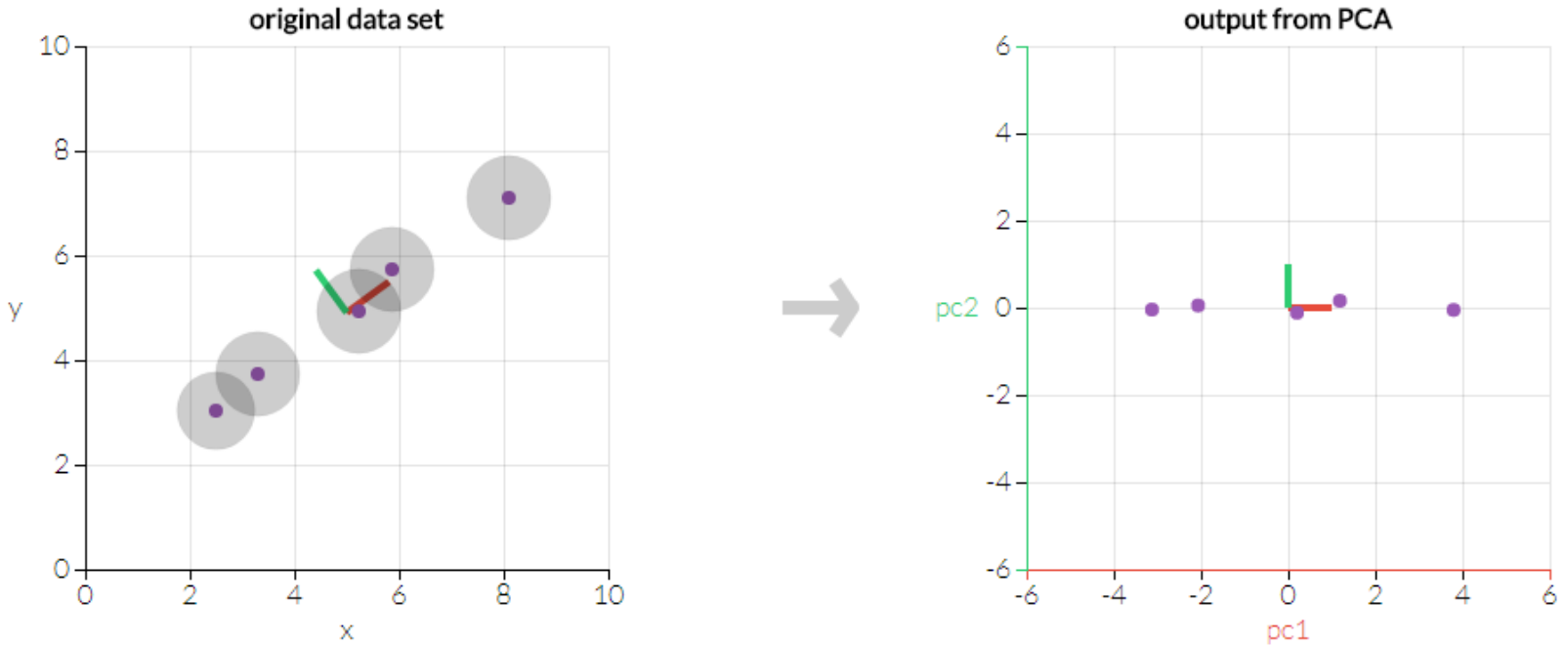
- Zawsze mniej na uwadze cel redukcji danych
  - Chcesz przedstawić, które są istotne -> filter
  - Chcesz poprawić predykcję -> wrapper

# Ekstrakcja cech

- Najpopularniejsze podejście: PCA
- Analizę PCA można wykonać na parę sposobów, najczęściej stosowany to rozkład SVD macierzy

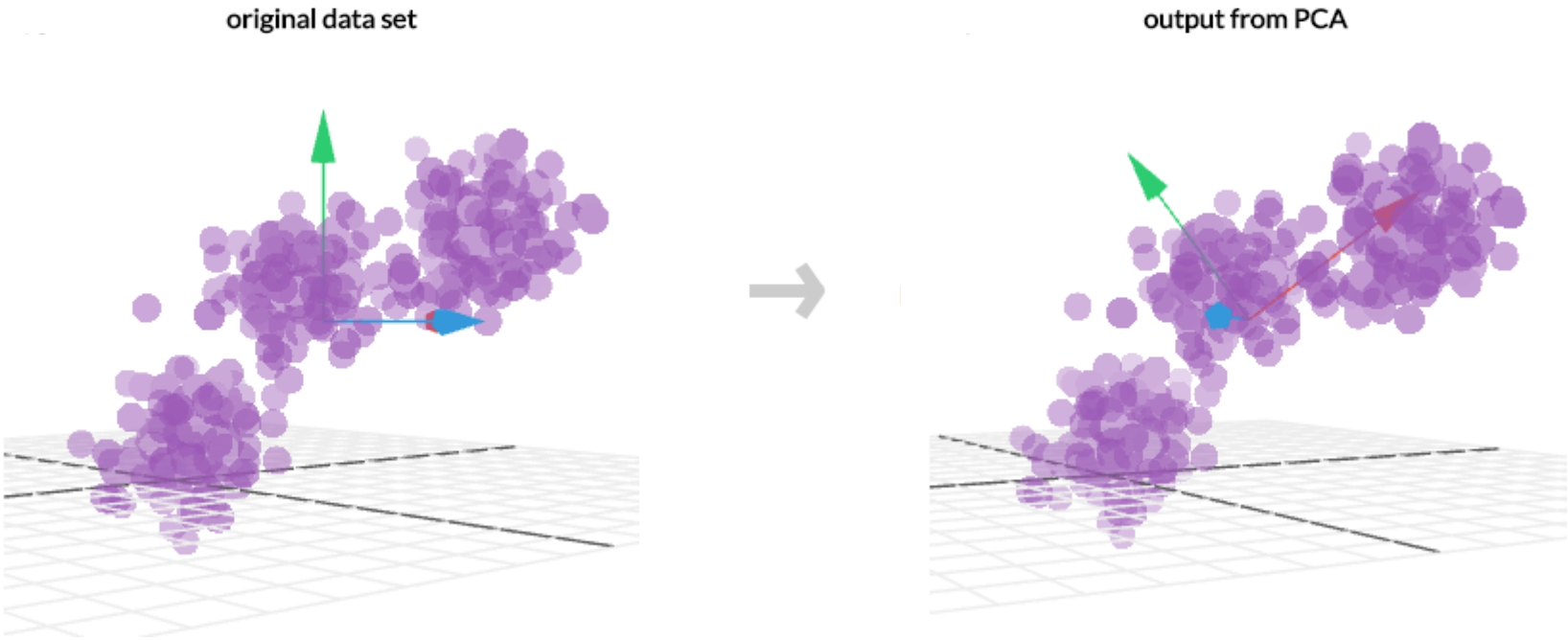
- PCA = Principal Component Analysis
  - dla macierzy kowariancji (zależności między atrybutami) oblicz **wartości własne** i **wektory własne**
  - wektory własne to transformacje do nowych atrybutów
  - wartości własne to istotności tych nowych atrybutów
  - pozostaw tylko najważniejsze z nowych atrybutów

# Ekstrakcja cech




http://setosa.io/ev/principal-component-analysis/

# Ekstrakcja cech

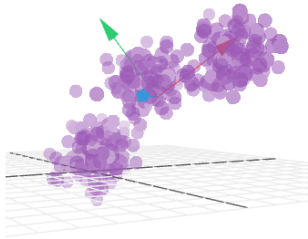



http://setosa.io/ev/principal-component-analysis/

# Podsumowanie

- Przetwarzanie wstępne
- Czyszczenie danych
- Transformacje danych
- Selekcja cech
  - Metody typu filter
  - Metody typu wrapper
  - Metody typu embedded
- Redukcja wymiarów





# Zadanie

1. Pobierz dane i notatnik z repozytorium
2. Rozpakuj folder z danymi i notatnikiem
3. Uruchom notatnik i wykonuj po kolei zadania

# Przydatne linki

- http://scikit-learn.org/stable/modules/feature_selection.html
- https://blog.datadive.net/selecting-good-features-part-i-univariate-selection/
- http://scikit-learn.org/stable/modules/generated/sklearn.decomposition.PCA.html
- http://setosa.io/ev/principal-component-analysis/