15.04.2017

PROJEKT MBI - dokumentacja wstępna

Analiza składowych głównych (PCA)

Zespół:
Magdalena Dziarczykowska
Karolina Walędzik

Zadanie

Celem projektu jest napisanie aplikacji umożliwiającej zrozumienie działania algorytmu służącego do analizy składowych głównych.

Algorytm PCA

W projekcie przedstawione będą kolejne kroki algorytmu PCA (ang. *Principal Component Analysis*). Uproszczona analiza składowych głównych dla zadanych danych wejściowych składa się z następujących kroków:

Dane wejściowe:

	atrybut			
pomiar	1	2	...	m
1	x_{11}	x_{12}	...	x_{1m}
2	x_{21}	x_{22}	...	x_{2m}
...	...	...	...	...
n	x_{n1}	x_{n2}	...	x_{nm}

Krok 1: Obliczenie **średniej i odchylenia standardowego** (dla każdego atrybutu).
Krok 2: Normalizacja zgodnie ze wzorem. Po normalizacji wszystkie atrybuty będą miały parametry: średnia = 0 i odchylenie = 1.

$$z_{ij} = \frac{x_{ij} - \mu_j}{\sigma_j}$$

Krok 3: Znalezienie **nowego układu współrzędnych** - tylko pomocą przekształceń liniowych:

Krok 3.1. Wyznaczamy macierz kowariancji Cz będącej wyrazem liniowej zależności między kolejnymi atrybutami.

$$\mathbf{C_z} = \frac{1}{n}\mathbf{Z}\mathbf{Z}^T = \begin{bmatrix} \sigma_1^2 & \sigma_{12} & \dots & \sigma_{1m} \\ \sigma_{21} & \sigma_2^2 & \dots & \sigma_{2m} \\ \dots & \dots & \dots & \dots \\ \sigma_{n1} & \sigma_{n2} & \dots & \sigma_m^2 \end{bmatrix} = \begin{bmatrix} 1 & \sigma_{12} & \dots & \sigma_{1m} \\ \sigma_{21} & 1 & \dots & \sigma_{2m} \\ \dots & \dots & \dots & \dots \\ \sigma_{n1} & \sigma_{n2} & \dots & 1 \end{bmatrix}$$

gdzie $\sigma_{a_i a_j}$ to kowariancja między atrybutami i i j.

Krok 3.2. Poszukiwanie takiego **przekształcenia** P **macierzy kowariancji**, aby otrzymać **postać diagonalną** C_Y **.**

$$\mathbf{C_Y} = \begin{bmatrix} \lambda_1 & 0 & \dots & 0 \\ 0 & \lambda_2 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ 0 & 0 & \dots & \lambda_m \end{bmatrix}$$

$$\mathbf{C_Y} = \frac{1}{n}\mathbf{Y}\mathbf{Y}^T = \frac{1}{n}(\mathbf{PZ})(\mathbf{PZ})^T = \frac{1}{n}\mathbf{P}\mathbf{Z}\mathbf{Z}^T\mathbf{P}^T = \mathbf{P}\mathbf{C_Z}\mathbf{P}^T$$

Krok 4. Wyznaczenie z macierzy kowariancji wartości własnych i wektorów własnych.

Rezultat: Otrzymane **wektory własne** będą naszymi **składowymi głównymi**, a **wartości własne** będą mówić **jak dużo wariancji przynależy do tego wektora własnego**.

Do wyboru nowego układu współrzędnych w algorytmie PCA stosuje się często k**ryterium *Kaisera-Guttmana,*** które mówi, iż należy zachować składowe, dla których wartości własne są większe od 1 (czyli wkład składowej większy, niż wkład pojedynczej zmiennej).

Podstawowe założenie aplikacji

Aplikacja będzie miała charakter webowy, a wszelkie obliczenia będą wykonywać się po stronie klienta. Na stronie będą pola umożliwiające wprowadzanie własnych danych orazi śledzenie kolejnych kroków algorytmu PCA.

Wykaz funkcjonalności

Podstawowe funkcjonalności z punktu widzenia użytkownika aplikacji:

1. **Możliwość prześledzenia najważniejszych kroków algorytmu PCA**:
   - ➢ normalizacja danych
   - ➢ obliczenie macierzy kowariancji
   - ➢ wyznaczenie macierzy diagonalnej
   - ➢ wyznaczenie wektorów własnych i wartości własnych
2. **Wprowadzanie własnych danych lub użycie przykładowych**
3. **Przedstawienie działania algorytmu dla 2 wymiarów**
4. **Propozycja dobrania ilości składowych głównych zgodnie z kryterium *Kaisera-Guttmana***

Technologie i narzędzia

Planowane do wykorzystania narzędzia:

- ➢ HTML5 + CSS3
- ➢ język JavaScript + biblioteka z zaimplementowanym algorytmem PCA (prawdopodobnie [2])

Źródła

[1] R. Nowak, materiały wykładowe do przedmiotu *Metody bioinformatyki.*
[2] 12.04.2017, https://mljs.github.io/pca/