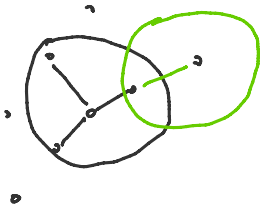
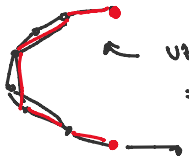
RDD - 10
27 April 2021 11:38

Minule: Isomap
- zachytenie štruktúry

1.) Graf susednosti



2.) 

3.) štúdium na vzdialenostiach grafových

RKo: Swiss roll data
: MNIST

---

t-SNE

t-distributed stochastic neighbor embedding

- $x_1, \dots, x_n \in \mathbb{R}^q$ - chceme aproximovať pomocou $y_1, \dots, y_n \in \mathbb{R}^p$
  → zachovať vzájomné polohy, štruktúru dát

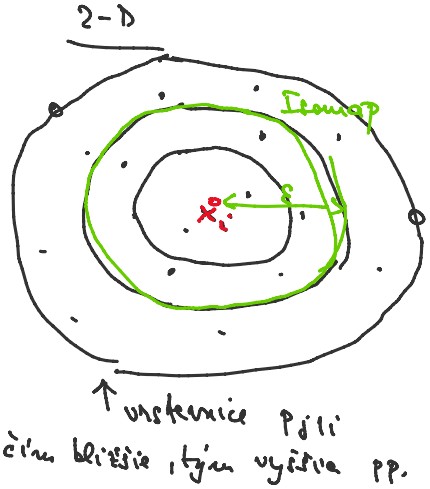
História: najprv - SNE (Hinton & Roweis, 2002)
potom - t-SNE (van der Maaten & Hinton, 2008)

- vzdialenosti medzi $x_1, \dots, x_n$ vyjadríme pomocou pp.:
  $p_{j|i}$ = pp., že x_i si vyberie bod x_j ako svojho suseda (v grafe)
  → pravdepodobnostný prístup k Isomap

$$p_{j|i} = \frac{e^{-\frac{\|x_i - x_j\|^2}{2\sigma_i^2}}}{\sum_{k \neq i} e^{-\frac{\|x_i - x_k\|^2}{2\sigma_i^2}}}$$

$e^{-\frac{\|x_i - x_j\|^2}{2\sigma_i^2}}$ } ~ proporcionálne hustote $N(x_i, \sigma_i^2 I)$

$\sum_{k \neq i} e^{-\frac{\|x_i - x_k\|^2}{2\sigma_i^2}}$ } ~ normalizácia



↑ vrstevnice $p_{j|i}$
čím bližšie, tým vyššia pp.

↪ Isomap: susedia: $\forall$ body bližšie ako ε

- blízkosť x_i a x_j: $p_{ij} = \frac{p_{j|i} + p_{i|j}}{2n}$ $\forall j \neq i$
  ↑ normalizácia

- vzdialenosť y_i a y_j: pomocou viacrozmerného t-rozdelenia (= "t-" SNE)
  ↳ $q_{ij} = \frac{(1 + \|y_i - y_j\|^2)^{-1}}{\sum_{r \neq s} (1 + \|y_r - y_s\|^2)^{-1}}$
  $(1 + \|y_i - y_j\|^2)^{-1}$ } ~ hustota t-rozdelenia
  $\sum_{r \neq s} (1 + \|y_r - y_s\|^2)^{-1}$ } normalizácia

$$\sum_{r \neq s} \sum (1 + \|y_r - y_s\|^2)^{-1} \quad \} \text{ normalizácia}$$

Odbočka: $U \sim N(0_q, \Sigma)$, $V \sim \chi^2_m$, nez. $\Rightarrow Y = \frac{U}{\sqrt{\frac{V}{m}}} + \mu \sim t_m(\mu, \Sigma)$

pre Σ-reg., $q = 2$, $m = 1$, má Y hustotu: ↑ q-rozmerné

$$f(y) = \frac{1}{\pi\left(1 + \|y - \mu\|^2\right)^{-3/2}} \quad \rightsquigarrow \text{pre } \mu = y_i \Rightarrow f(y) = \frac{1}{\left(\pi \|y - y_i\|^2\right)^{-3/2}}$$

- voľba $y_1, \dots, y_n$, aby $\underset{\|}{Q}$ zodpovedalo $\underset{\|}{P}$
  q_{ij} p_{ij}

Vzdialenosť pp. rozdelení: **Kullbackova - Leiblerova divergencia**

$$I(P, Q) = \sum_i \sum_{j \neq i} p_{ij} \cdot \ln\left(\frac{p_{ij}}{q_{ij}}\right) \quad \rightarrow \text{ako veľmi sú } P \text{ a } Q \text{ rozdielne}$$

$I(P, Q) \geq 0 \quad \forall P, Q$

$I(P, Q) = 0 \Leftrightarrow P = Q$ skoro všade (pre nás konečné: $\Leftrightarrow P = Q$)

$I(Q, P) \neq I(P, Q)$

- **cost function**: $C(Y) = I(P, Q) \longrightarrow \min_Y$
  ↑ dané ↑ určené vyplnením

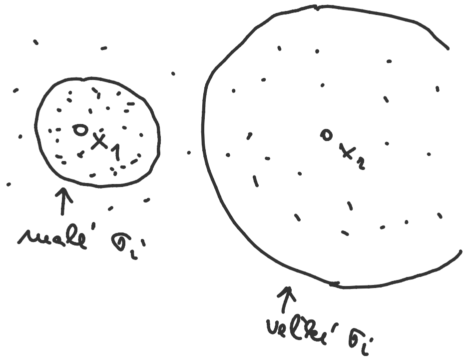
→ nekonvexná úloha ⇒ zložitá numerická optimalizácia

- upravená gradientná metóda (pomocou $\frac{\partial C}{\partial y_i}$)
- v poč. iteráciách: pridávame k y-ovom náhodný šum – aby sme nevviazli v lok. optime
- poč. riešenie: $y_1, \dots, y_n \sim N(0, \sigma I)$ pre malé $\sigma > 0$
- trik: early exaggeration – skoré zvýraznenie
  ↳ prvých niekoľko iterácií: $p_{ij} \mapsto c \cdot p_{ij}$ pre nejaké $c > 1$ (napr. $c = 12$)
  ⇒ Q nedokáže zachytiť P → optimalizácia sa pokúsi zachytiť hlavne veľké p_{ij} pomocou veľkých q_{ij} (ostatné $q_{ij} \approx 0$)

  ⇒ vytvára zhluky – husté
  – celkom dobre od seba oddelené

Voľba σ_i :

- čím väčšie σ_i, do tým väčšej diaľky sa x_i pozerá
- ak sú body blízko seba, stačí malé σ_i, ak sú riedke, tak treba veľké – aby mali porovnateľne veľa "susedov"



→ $Perp(P_i) = 2^{H(P_i)}$, $H(P_i) = -\sum_{j \neq i} p_{j|i} \cdot \log_2(p_{j|i})$: entropia pravdepodobnosti P_i $(p_{1|i}, \dots, p_{n|i})$

perplexity ; $H(P_i)$ ↘ rozdelenie $p_{j|i}$

: čím väčšie, tým menej rôznorodé je rozdelenie

⇒ $Perp(P_i)$: vyjadruje efektívny # susedov pre dané x_i

: čím väčšie σ_i, tým väčšie P_i

⇒ x_1 má $2 + \frac{1}{4}$ efektívnych susedov

→ σ_i nastavíme, aby všetky x_i mali rovnako veľa "susedov", teda aby $Perp(P_i)$ bolo rovnaké $\forall i$ (napr. $= 10$)

---

t-SNE : veľmi dobre rozlišuje zhluky

: môže nachádzať zhluky aj ak v dátach žiadne nie sú

: voľba t-rozdelenia – v mnohorozmere je "veľa priestoru": body sú vo všeob. ďaleko od seba → ako ich pomestiť do 2-rozmeru?

– crowding problem

→ v 2-D (q-D) volíme t-rozd. s ťažkými chvostami a tak sa tváríme, že body sú od seba ďalej (aby sme vedeli modelovať vzdialenejšie body)

Pr.: MNIST → tsne : veľmi pomalé

: umožňuje priebežné vykresľovanie

: nedávam scale, lebo štandardizáciu som už spravil pomocou PCA

→ R tsne : C++ kód od van der Maatena volaný cez Rko

: tiež spraví PCA vnútri (default)