

10. Clustering

Definiții

Algoritmul k -means

Convergența algoritmului k -means

Funcția de distorsiune

Clustering-ul este o problemă de învățare nesupervizată, în care setul de date de antrenare conține doar caracteristicile $\{x^{(1)}, x^{(2)}, \dots, x^{(m)}\}$ ($x^{(i)} \in \mathbb{R}^n$), etichetele corespunzătoare $y^{(i)}$ nefiind disponibile. În această situație, obiectivul metodei este de a grupa datele în câteva clustere.

Pașii algoritmului k -means sunt următorii:

1. inițializarea aleatoare a centrelor clusterelor $\mu_1, \mu_2, \dots, \mu_k \in \mathbb{R}^n$.

2. repetarea până la convergență: {

Pentru fiecare i se setează

$$c^{(i)} := \arg \min_j \|x^{(i)} - \mu_j\|^2 \quad (10.1)$$

Pentru fiecare j se setează

$$\mu^{(j)} := \frac{\sum_{i=1}^m 1\{c^{(i)} = j\}x^{(i)}}{\sum_{i=1}^m 1\{c^{(i)} = j\}} \quad (10.2)$$

}

În metoda prezentată, parametrul algoritmului este k . k este numărul de clustere pe care dorim să le găsim, iar centrele clusterelor μ_j reprezintă estimatele curente pentru pozițiile centrelor clusterelor. Centrele inițiale ale clusterelor (pasul 1 de mai sus) se pot seta pe valorile a k exemple de antrenare alese aleatoriu din baza de date de antrenare. Bineînțeles, sunt posibile și alte metode de inițializare.

Bucula internă a algoritmului efectuează repetitiv doi pași: (i) asignează fiecare exemplu de antrenare $x^{(i)}$ celui mai apropiat centru de cluster μ_j și (ii) deplasează fiecare centru de cluster μ_j la media punctelor asiguate lui. Două iterații ale algoritmului k -means sunt ilustrate în figura 10.1.

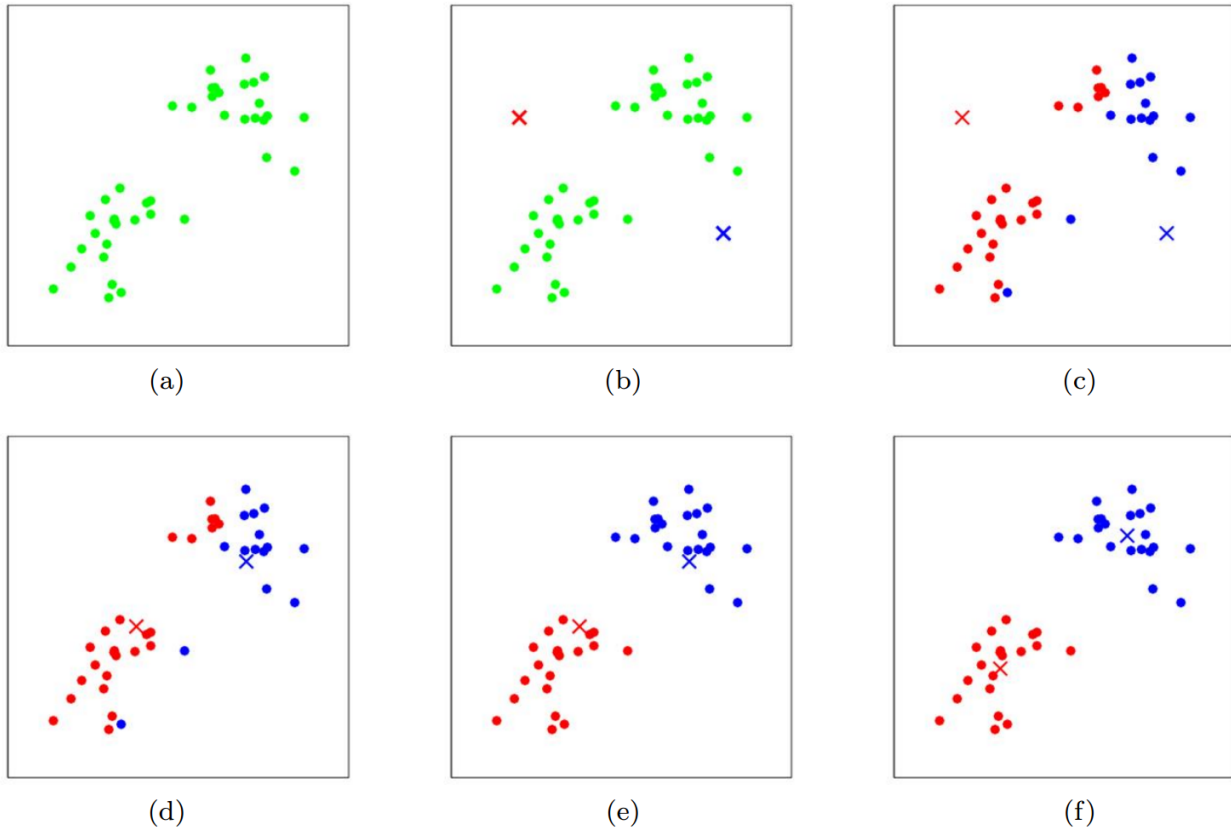


Fig. 10.1 Algoritmul k -means. Exemplele de antrenare sunt ilustrate sub formă de puncte, iar centrele clusterelor sub formă de x : (a) setul de date de antrenare original; (b) centrele clusterelor alese aleatoriu (în acest exemplu nu sunt alese ca și două valori din exemplele de antrenare); (c-f) ilustrarea rezultatului după rularea a două iterații ale algoritmului k -means. În fiecare iterație, fiecare exemplu de antrenare este asignat celui mai apropiat centru de cluster (acest lucru este vizibil prin colorarea fiecărui exemplu de antrenare în culoarea clusterului de care aparține); în pasul următor, deplasăm fiecare centru de cluster la media punctelor ce îi sunt atribuite.

Într-un anumit sens, convergența algoritmului k -means este garantată. În mod particular, se poate defini funcția de distorsiune ca:

$$J(c, \mu) := \sum_{i=1}^m \|x^{(i)} - \mu_{c(i)}\|^2. \quad (10.3)$$

J măsoară suma pătratică a diferențelor dintre fiecare exemplu de antrenare $x^{(i)}$ și centrele clusterelor $\mu_{c(i)}$ cărora exemplele i-au fost atribuite. Se poate demonstra că metoda k -means este echivalentă minimizării coordonatelor lui J . În mod particular, bucla internă a algoritmului minimizează J în mod repetat față de c , în timp ce μ este ținut constant. Apoi, minimizează J față de μ , în timp ce c este ținut constant. Astfel, J trebuie să descrească monoton, iar valoarea lui J trebuie să convergă. (De obicei, acest lucru implică de asemenea faptul că atât c , cât și μ , vor converge. Teoretic, este posibil pentru k -means să oscileze între câteva configurații de cluster (e.g. câteva valori diferite pentru c și/sau μ) ce au exact aceeași valoare a funcției J . Cu toate acestea, acest fenomen se întâmplă foarte rar în

practică.)

Funcția de distorsiune J este o funcție non-convexă. Astfel, minimizarea coordonatelor lui J nu garantează convergența către minimul global. Cu alte cuvinte, k -means este susceptibil la minime locale. Cu toate acestea, k -means va funcționa optim. În cazul în care se dorește o verificare suplimentară a convergenței, o abordare utilă este rularea algoritmului de mai multe ori, utilizându-se diferite valori aleatoare pentru centrele clusterelor μ_j . Din multitudinea de clusterse obținute, se poate alege configurația ce întoarce cea mai mică valoare a funcției de distorsiune $J(c, \mu)$.

Bibliografie

- [1] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Secaucus, NJ, USA: Springer-Verlag New York, Inc., 2006.
- [2] I. Goodfellow, Y. Bengio and A. Courville, *Deep Learning*. MIT Press, 2016, <http://www.deeplearningbook.org>.
- [3] M. Lutz, *Learning Python*, 2nd Ed. Sebastopol, CA, USA: O'Reilly & Associates, Inc., 2003.
- [4] A. Ng, "Stanford cs229 - machine learning," 2008.
- [5] S. J. Russell and P. Norvig, *Artificial Intelligence: A Modern Approach*, 2nd Ed. Pearson Education, 2003.
- [6] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. u. Kaiser and I. Polosukhin, "Attention is all you need," *Advances in Neural Information Processing Systems*, Vol. 30. Curran Associates, Inc., 2017.