3. domača naloga: razvrščanje v skupine

24. april 2018

1 Uvod

Cilj naloge je pozitivne zvezne podatke, 61637 primerov opisanih z 25133 atributi, čim boljše razvrstiti v skupine. Problem je, ker ne vemo števila skupin in pa tudi zaradi ogromnega števila podatkov, je potrebno biti pozoren na čas izvajanja.

2 Metoda

Podatke smo najprej binarizirali in izbrali 2000 stolpcev, ki so imeli največ ne ničelnih vrednosti (enako tudi pri konsenznem razvrščanju). Nad temi podatki smo potem uporabili "Non-negative matrix factorization"(NMF) za dodatno zmanjšanje dimenzij in nad pridobljeno matriko uporabili "Mini batch Kmeansža razrstitev v skupine.

3 Konsenzno razvrščanje

Konsenzno razvrščanje poskuša določiti najboljše število skupin za podane podatke. Opisno algoritem deluje tako, da iteriramo skozi različno število potencialnih skupin. Pri vsakem številu nekajkrat vzamemo naključen poljuben delež podatkov (naključni izbrani), jih razvrstimo v prej določeno število skupin s poljubnim algoritmom. Ko za določeno število skupin, končamo postopek opisan v prejšnji povedi, izračunamo konsenzno matriko. Konsenzna matrika je simetrična matrika kjer (i,j) element pomeni, kolikokrat sta bila i-ti in j-ti vrstici razvrščeni v isti skupini, deljeno s kolikokrat sta bila sploh izbrana pri naključnem izbiranju. Torej so vrednosti konsenzne matrike med 0 in 1. Izberemo tisto število skupin, za katere ima konsenzna matrika vrednosti najbolj razvrščene k 1 ali 0, in čim manj vmesnih vrednosti. Matriko razdalj lahko potem dobimo tako, da matriko samih 1 odštejemo najboljšo konsenzno matriko. Izgradimo model s poljubnim algoritmom in dobimo razvrstitve.
Problem je, v številu primerov v podatkih. Ker imamo 61637 primerov, pomeni, da bo konsenzna matrika velika $3.8 * 10^8$, kar je ogromno preveliko za shranjevanje v RAM pri osebnem računalniku. Zaradi tega smo vzeli le 10% naključnih primerov in nad njimi izvedli konsenzno razvrščanje. Potem pa te primere in njihove razvrstitve v skupine uporabili za izgraditev naključnih gozdov. Te gozdove pa smo uporabili za razvrstitev preostalih primerov v skupine.
Za razvvrščanje v skupine znotraj konsenznega razvrščanja smo uproabili "mini batch Kmeans",

za izgraditev modela glede na dobljeno matriko razdalja pa "Agglomerative clustering". Izvajanje algoritma je trajalo 3 ure. Vmesni rezultat na strežniku pa je bil: 0.31292.