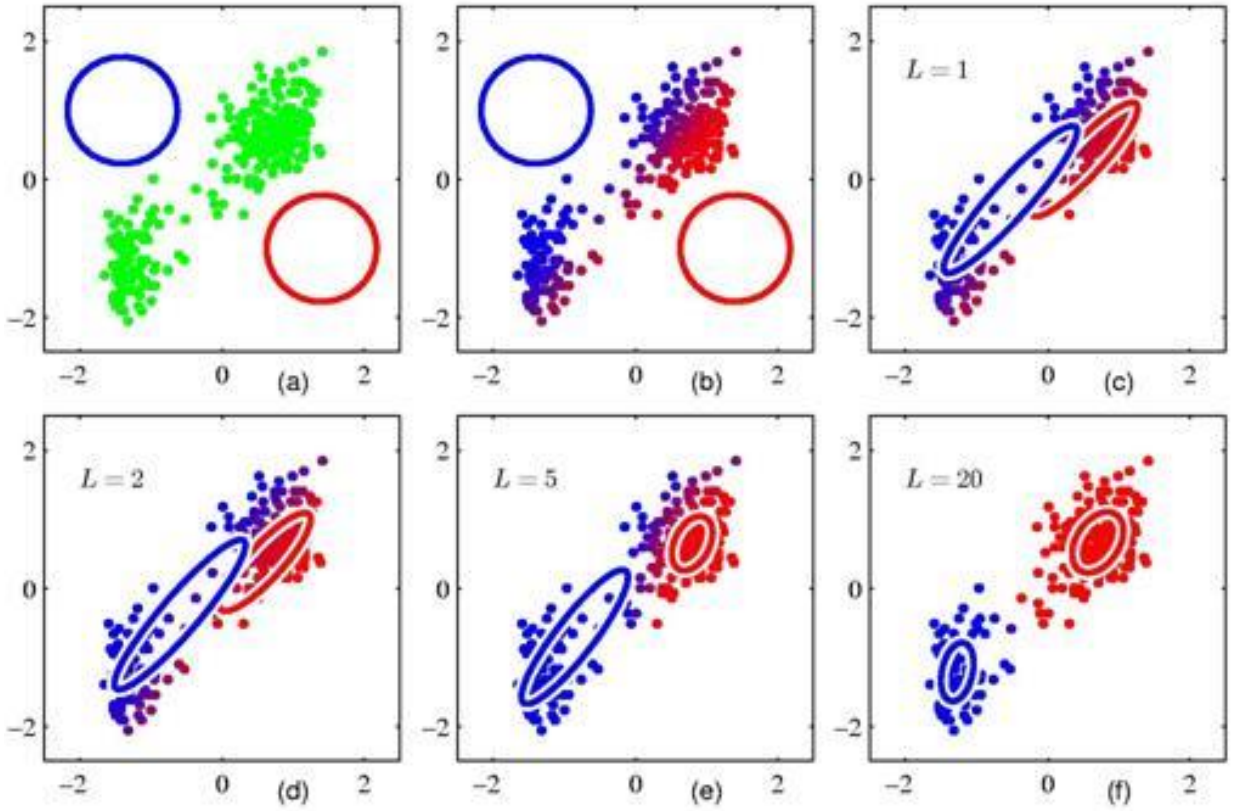


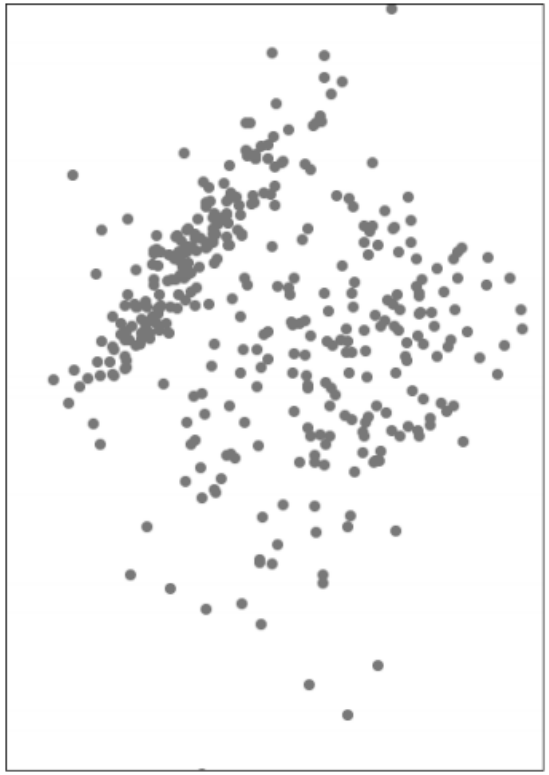
GMM: obučavanje modela

Expectation - Maximization

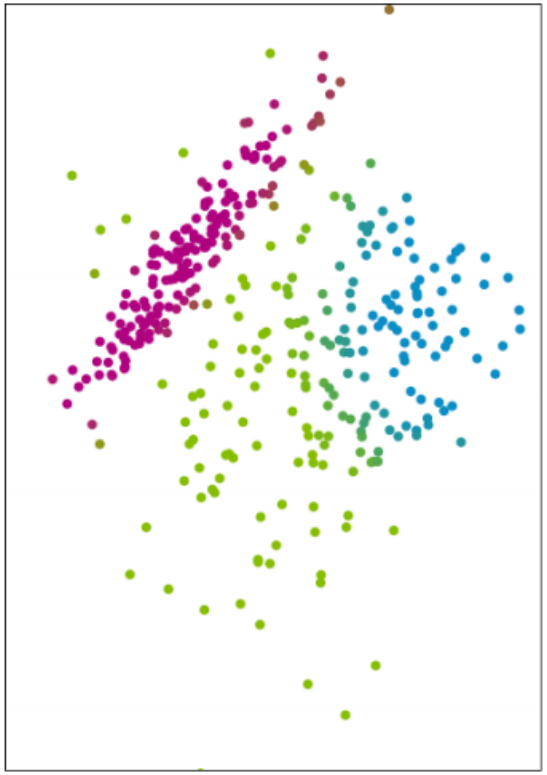
Kako da obučimo model?

- Obučavanje ćemo naučiti pomoću metode očekivanje-maksimizacija (*Expectation maximization, EM*)

Podaci

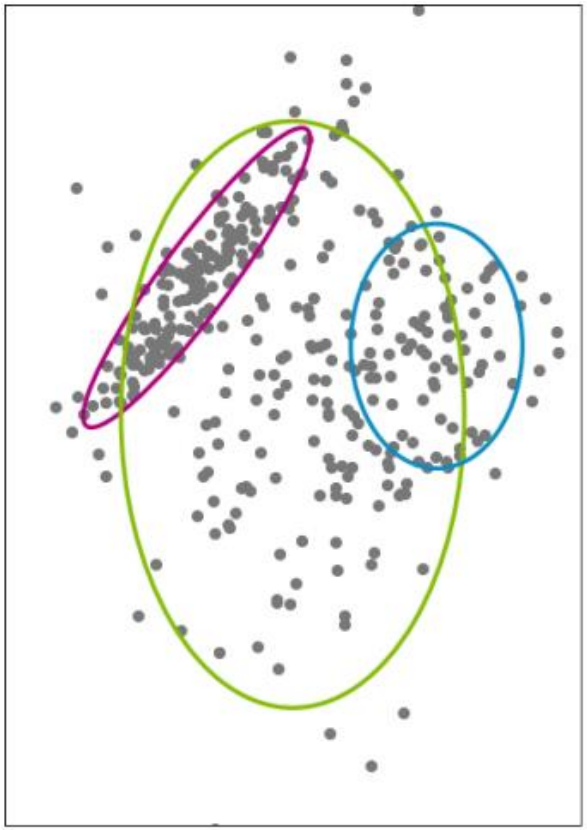


Željene „meke“ dodele



Šta ako znamo parametre klastera?

- Znamo $\{\pi_k, \mu_k, \Sigma_k\}$
- Možemo da odredimo „meke“ dodele koristeći $\{\pi_k, \mu_k, \Sigma_k\}$:



Odgovornost klastera k za opservaciju i

Za date parametre modela

$$r_k^{(i)} = p\left(z^{(i)} = k \middle| \{\pi_j, \mu_j, \Sigma_j\}_{j=1}^{K}, x^{(i)}\right)$$

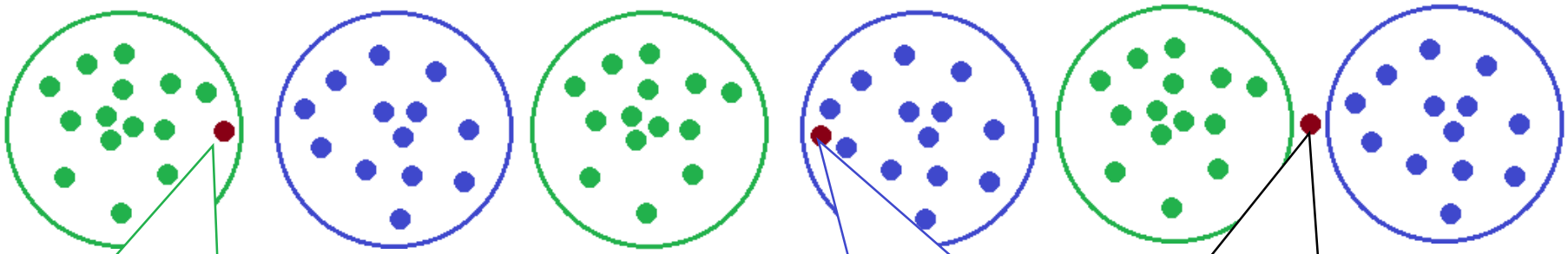
Verovatnoća dodele klasteru k

i uočenu vrednost

Za svako $x^{(i)}$ ćemo formirati vektor odgovornosti:

$$r^{(i)} = \left[r_1^{(i)}, r_2^{(i)}, \ldots, r_K^{(i)}\right]$$

Odgovornost

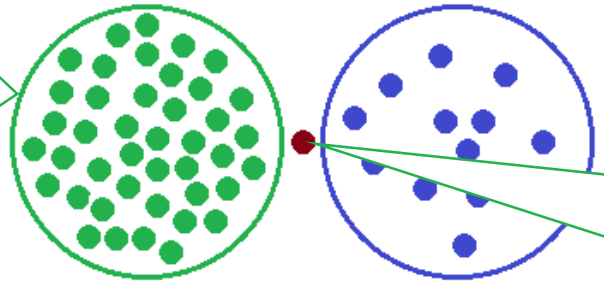


Zeleni klaster će imati veću odgovornost

Plavi klaster će imati veću odgovornost

Dodela je nesigurna – odgovornost je jednako raspodeljena

Zeleni klaster ima veću težinu (jer ima više elemenata)



Dodela je i dalje nesigurna, ali će zeleni klaster uzeti veću odgovornost

Odgovornost

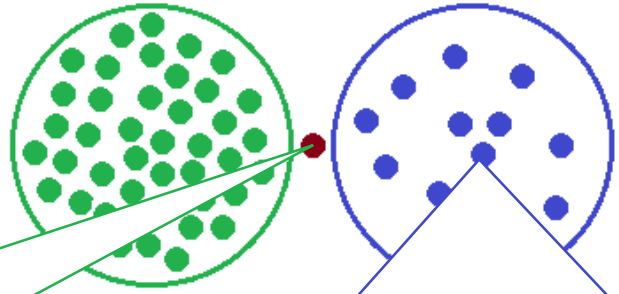
$$r_k^{(i)} = p\left(z^{(i)} = k \middle| \{\pi_j, \mu_j, \Sigma_j\}_{j=1}^{K}, x^{(i)}\right)$$

Apriorna verovatnoća pripadanja klasteru *k*

Koliko je verovatno da uočimo opsevaciju $x^{(i)}$ u ovom klasteru

$$r_k^{(i)} = \frac{\pi_k \mathcal{N}\left(x^{(i)} | \mu_k, \Sigma_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(x^{(i)} | \mu_k, \Sigma_k\right)}$$

Normalizacija preko svih mogućih klastera (da bismo imali validnu gustinu verovatnoće)



Apriorna verovatnoća zelenog klastera je veća pa ova tačka verovatnije pripada zelenom klasteru

Tačka koja se ovde nalazi bi imala veću verovatnoću pripadnosti plavom klasteru (iako je apriorna verovatnoća zelenog klastera veća)

Odgovornost

- Primena Bajesovog pravila

$P(A|\mathrm{B}, \mathrm{params})$

Događaj A — Parametri — Događaj B

$$r_k^{(i)} = p\left(z^{(i)} = k \,|\, \{\pi_j, \mu_j, \Sigma_j\}_{j=1}^{K}, x^{(i)}\right)$$

$P(A|\mathrm{params})$ $\quad$ $P(B|A, \mathrm{params})$

$$r_k^{(i)} = \frac{\pi_k \mathcal{N}\left(x^{(i)}|\mu_k, \Sigma_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(x^{(i)}|\mu_k, \Sigma_k\right)}$$

Sumiramo preko svih događaja C $\quad$ $P(C|\mathrm{params})$ $\quad$ $P(B|C, \mathrm{params})$

$P(B|\mathrm{params})$

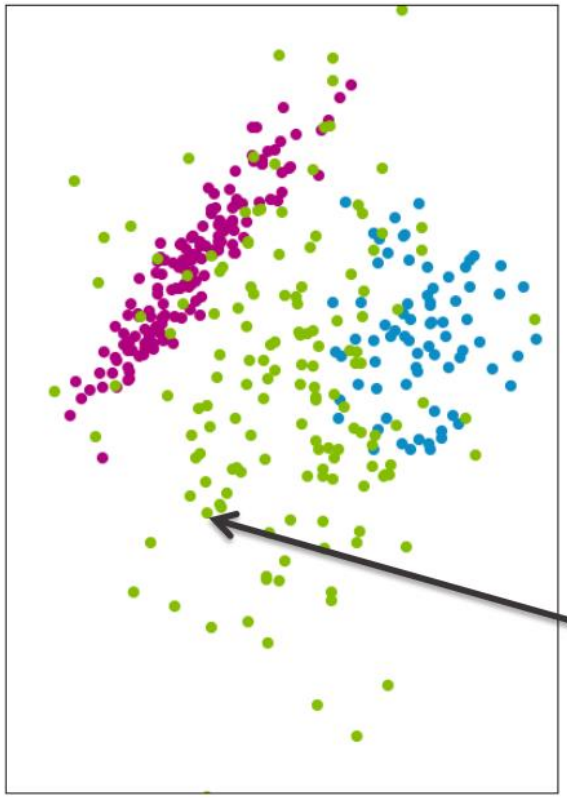
Zaključak

- Ako znamo parametre klastera – lako možemo odrediti „meke“ dodele opservacija klasterima

- Ali, mi ne znamo parametre klastera...

Šta ako znamo (tvrde) pripadnosti?

- Ako bismo znali (tvrde) pripadnosti klasteru $z^{(i)}$ mogli bismo da odredimo parametre klastera



- Estimiraćemo $\{\pi_k, \mu_k, \Sigma_k\}$ isključivo na osnovu podataka iz klastera *k*
- Pomoću MLE metode (*Maximum Likelihood Estimation*): pronaći parametre koji maksimizuju verodostojnost podataka

Ova zelena tačka ne utiče na parametre plavog i ljubičastog klastera

MLE za određivanje parametara klastera

- MLE: Klaster *k* je specificiran Gausovom distribucijom koja ima dva parametra:

$\hat{\mu}_k = \frac{1}{N_k} \sum_{x^{(i)} \in k} x^{(i)}$ (srednja vrednost uzorka)

$\hat{\Sigma}_k = \frac{1}{N_k} \sum_{x^{(i)} \in k} (x_i - \hat{\mu}_k)(x_i - \hat{\mu}_k)^T$ (kovarijansa uzorka)

I težinom dodeljenom klasteru:

$\hat{\pi}_k = \frac{N_k}{N}$

N_k – broj opservacija u klasteru *k*, *N* – ukupan broj opservacija

Zaključak

- Ako znamo „tvrde“ pripadnosti klasterima – lako možemo odrediti parametre klastera

- Da li bismo mogli proceniti parametre klastera iz „mekih“ pripadnosti $r_k^{(i)}$?

Šta ako znamo „meke“ pripadnosti?

- Svaka opservacija $x^{(i)}$ je podeljena među svim klasterima, što je određeno vektorom odgovornosti $r^{(i)} = \left[r_1^{(i)}, r_2^{(i)}, \ldots, r_K^{(i)}\right]$
- Slično *boosting*-u, gde smo imali otežinjene opservacije:

R	G	B	$r_1^{(i)}$	$r_2^{(i)}$	$r_3^{(i)}$
$x_1^{(1)}$	$x_2^{(1)}$	$x_3^{(1)}$	0.3	0.18	0.52
$x_1^{(2)}$	$x_2^{(2)}$	$x_3^{(2)}$	0.01	0.26	0.73
$x_1^{(3)}$	$x_2^{(3)}$	$x_3^{(3)}$	0.002	0.008	0.99
$x_1^{(4)}$	$x_2^{(4)}$	$x_3^{(4)}$	0.75	0.10	0.15
$x_1^{(5)}$	$x_2^{(5)}$	$x_3^{(5)}$	0.05	1.93	0.02
$x_1^{(6)}$	$x_2^{(6)}$	$x_3^{(6)}$	0.13	0.86	0.01
Ukupna težina u klasteru:			**1.242**	**2.8**	**2.42**

52% šanse da je ova opservacija u klasteru 3

Estimacija parametara klastera

- Slično kao kod „tvrdih“ pripadnosti, samo sada sve instance pripadaju svim klasterima sa određenom težinom:

$$\hat{\mu}_k = \frac{1}{N_k^{soft}} \sum_{i=1}^{N} r_k^{(i)} x^{(i)}$$

$$\hat{\Sigma}_k = \frac{1}{N_k^{soft}} \sum_{i=1}^{N} r_k^{(i)} \left(x^{(i)} - \hat{\mu}_k\right)\left(x^{(i)} - \hat{\mu}_k\right)^T$$

Gde je N_k^{soft} ukupna težina klastera *k*:

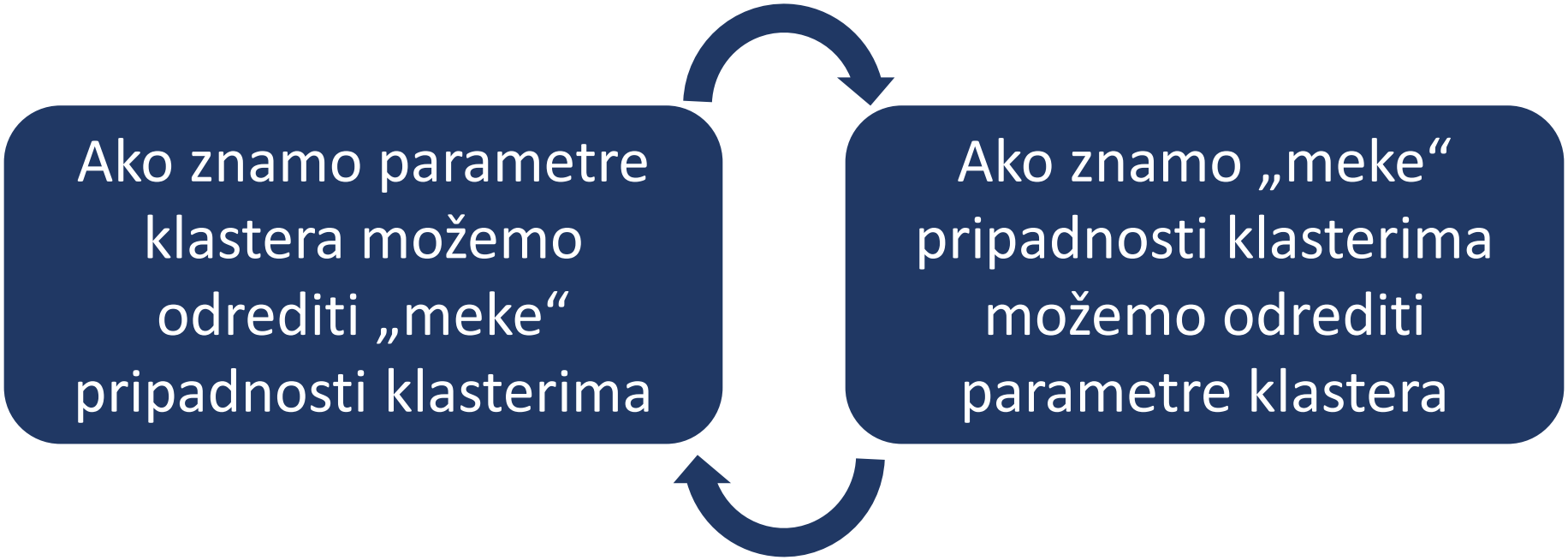
$$N_k^{soft} = \sum_{i=1}^{N} r_k^{(i)}$$

- Određivanje težina klastera: $\hat{\pi}_k = N_k^{soft} / N$

Zaključak

- Ako znamo „meke“ pripadnosti klasterima – lako možemo odrediti parametre klastera

- Ali, mi ne znamo „meke“ pripadnosti klasterima...



EM metoda

- Inicijalizovati parametre klastera $\{\pi_k, \mu_k, \Sigma_k\}^{(0)}$

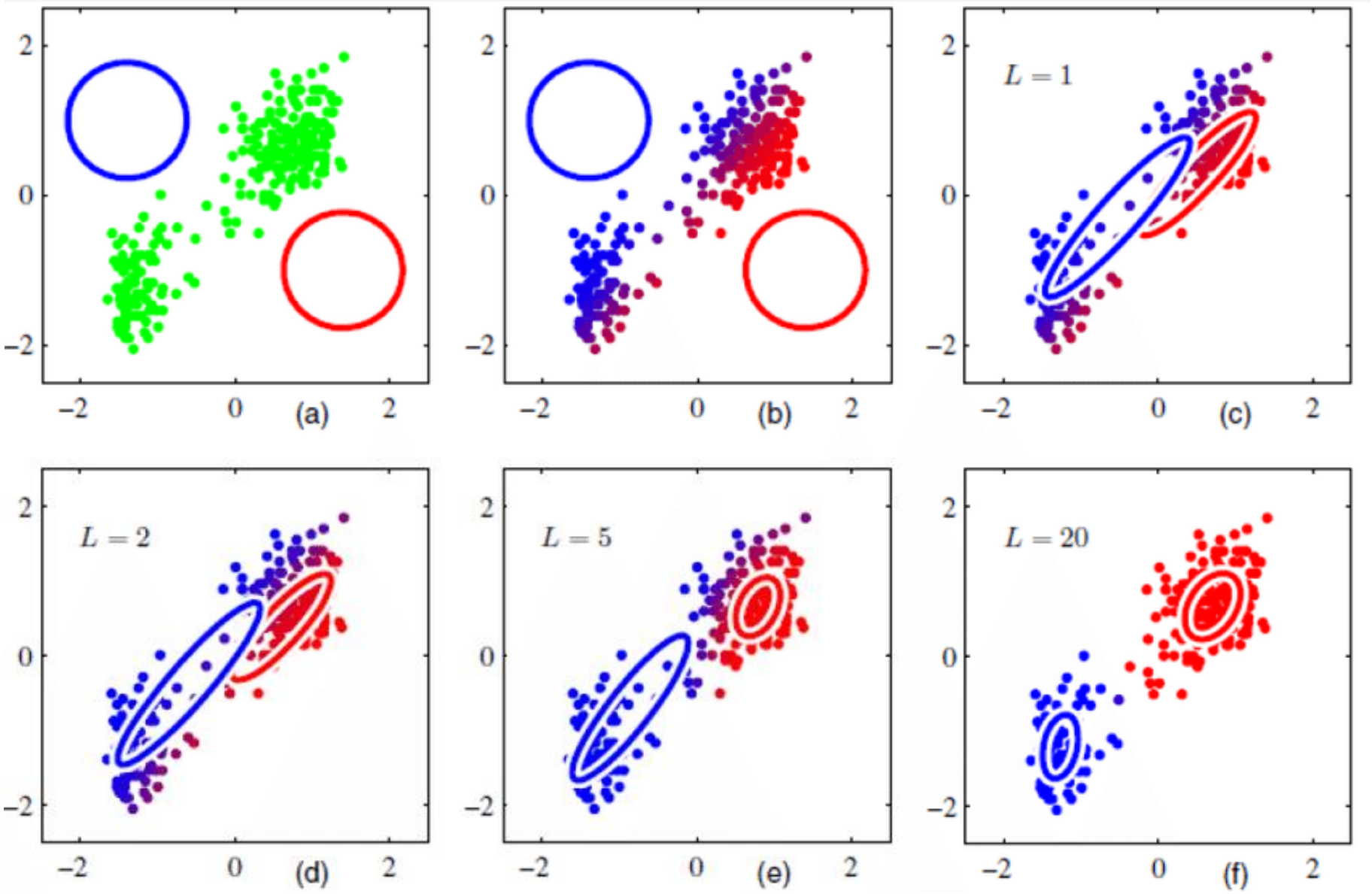
- Iterativno do konvergencije:
  1. E-korak: proceniti odgovornosti klastera na osnovu estimiranih parametrara klastera:

$$r_k^{(i)} = \frac{\pi_k \mathcal{N}\left(x^{(i)} | \mu_k, \Sigma_k\right)}{\sum_{j=1}^{K} \pi_j \mathcal{N}\left(x^{(i)} | \mu_k, \Sigma_k\right)}$$

  2. M-korak: odrediti parametre klastera maksimizacijom verodostojnosti podataka:

$$\hat{\pi}_k, \hat{\mu}_k, \hat{\Sigma}_k | \left\{r_k^{(i)}, x^{(i)}\right\}$$

EM ilustracija



Konvergencija EM metode

- EM je *coordinate-ascent* algoritam
  - E- i M- koraci se mogu povezati sa alternirajućim maksimizacijama ciljne funkcije
- Konvergira u lokalni optimum
- Konvergenciju možemo proveriti iscrtavanjem vrednosti logaritma verodostojnosti podataka u svakoj iteraciji
  - nakon određenog vremena ova vrednost bi trebala da prestane da se povećava

Inicijalizacija EM metode

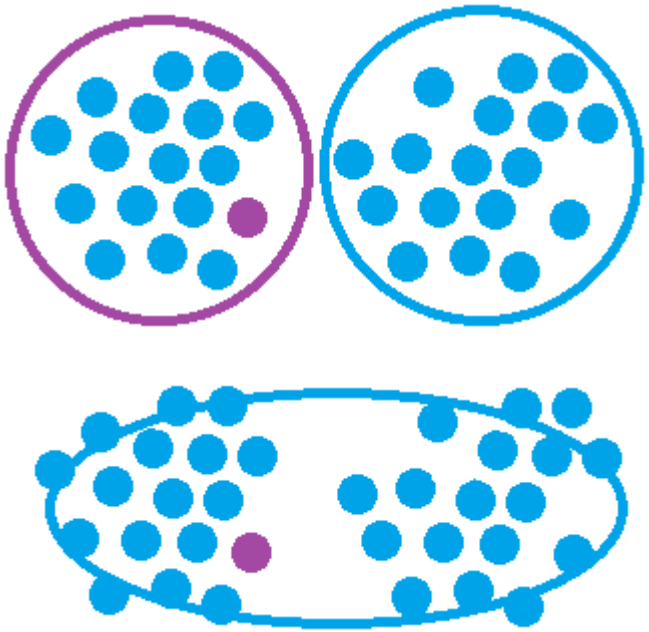
- EM konvergira u lokalni optimum → inicijalizacija algoritma je od velikog značaja za kvalitet dobijenog rešenja, kao i za brzinu konvergencije
- Sa lošim inicijalnim centroidima, EM može da rezultuje veoma rasutim klasterima koji se prilično preklapaju

- Postoji više načina inicijalizacije:
  - Na slučajan način odabrati *K* opservacija koje će predstavljati „centroide“. Dodeliti preostale opservacije njima najbližem centroidu. Formirati inicijalnu estimaciju parametara klastera
  - Odabrati centre sekvencijalno kako bi se dobila dobra pokrivenost podataka kao u *k*-means++
  - Iskoristiti rešenje dobijeno putem *k*-means algoritma za inicijalizaciju
  - Formirati model mešavina podelom (i ponekad uklanjanjem) klastera sve dok *K* klastera nije formirano

EM overfitting

- Makismizacija verodostojnosti je podložna overfittingu

- Npr. $K = 2$ i imamo samo jedan primer dodeljen klasteru 1, dok su svi ostali primeri dodeljeni klasteru 2



Parametri koji maksimizuju verodostojnost:

- Klaster 1 (ljubičasti): centar će da bude jednak datoj opservaciji, a varijansa će da bude 0
- Klaster 2 (plavi): varijansa će se uvećati da obuhvati sve dodeljene tačke
- Verodostojnost u ovom slučaju raste na ∞
- Ipak, u praksi se ovo gotovo nikada ne dešava

EM overfitting

- Overfitting je mnogo veći problem ako imamo više dimenzija

- Npr. vršimo klasterizaciju tekstualnih dokumenata
  - Visoka dimenzionalnost usled velikog broja reči
  - Neka samo jedan dokument dodeljen klasteru *k* ima reč *w* (ili se reč *w* u svim dokumentima klastera *k* pojavljuje isti broj puta)
  - Parametri koji maksimizuju verodostojnost su $\mu_k[w] = x_w^{(i)}$ i $\sigma_{k,w}^2 = 0$
  - Ovo nije tako nerealan slučaj s obzirom na broj dimenzija (naročito ako imamo veliki broj klastera da podelimo podatke) – lako se može desiti da od svih dokumenata u klasteru ni jedan nema datu reč ili da samo jedan dokument ima datu reč
  - Rezultat je da je (u kasnijim iteracijama) verodostojnost da bilo koji dokument sa različitom frekevncijom reči *w* bude u klasteru 0

EM overfitting

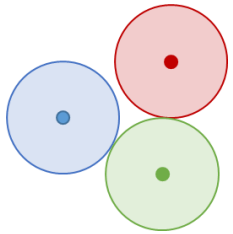
- Najčešće se opisani problem u praksi dešava zbog *underflow* problema
  - Rešenje je jednostavna ispravka koda: ne dozvoliti da varijanse padnu na 0!
  - Svaki put kada procenjujemo kovarijansu dodajemo malu vrednost na dijagonalu estimirane matrice
  - *smoothing the parameter estimate/regularizing EM updates*

- Alternativno imamo (formalniji) bajesovski pristup gde se na parametre klastera dodaje prior
  - Izglađivanje putem pseudo-opservacija koje dodajemo klasteru
  - Kao da smo svakom klasteru dodali „virtuelne“ opservacije
  - Izbegava se slučaj kada nemamo nijednu opservaciju sa nekom vrednošću atributa
  - Nasuprot prethodno opisanog pristupa (gde se „izglađuje“ samo varijansa) ovde modifikujemo sve parametre klastera

GMM vs. *k*-means

- GMM sa sferno simetričnim klasterima:

$$\Sigma = \begin{pmatrix} \sigma^2 & & \\ & \sigma^2 & \\ & & \dots \\ & & \sigma^2 \end{pmatrix}$$



- Smanjimo varijansu na 0 ($\sigma \to 0$)

- Posledice:
  - Klasteri su sferni sa jednakim varijansama, pa su relativne verodostojnosti samo funkcija rastojanja od centra klastera
  - Pošto varijanse teže 0, verodostojnosti postaju 0/1 vrednosti
  - Odgovornosti su otežinjene proporcijama klastera, ali 0/1 vrednosti verodostojnosti dominiraju
  - Opservacije će biti u potpunosti dodeljene najbližem klasteru, baš kao u *k*-means algoritmu

GMM vs. *k*-means

- EM sa beskonačno malim varijansama = *k*-means

1. E-korak (procena odgovornosti za date estimacije parametara klastera):

$$\hat{r}_k^{(i)} = \frac{\hat{\pi}_k \mathcal{N}\left(x^{(i)} | \hat{\mu}_k, \sigma^2 I\right)}{\sum_{j=1}^{K} \hat{\pi}_j \mathcal{N}\left(x^{(i)} | \hat{\mu}_j, \sigma^2 I\right)} \in \{0,1\}$$

Odluka bazirana na rastojanju od najbližeg centra

Beskonačno malo

2. M-korak (maksimizacija verodostojnosti optimizacijom parametara klastera za procenjene odgovornosti):

$$\hat{\pi}_k, \hat{\mu}_k | \left\{\hat{r}_k^{(i)}, x^{(i)}\right\}$$

Sumarizacija

- Motivisali smo probabilistički pristup klasterovanju
- Naučili smo EM algoritam
- Uporedili smo EM (probabilistički pristup) sa *k*-means (pristup baziran na modelu)
  - Uvideli smo da je *k*-means specijalni slučaj GMM
- EM je generalizacija *k*-means sa određenim prednostima:
  - Fleksibilniji (oblik i orijentacija klastera)
  - Deskriptivniji izlaz („meke“ dodele)
- Ali ima svoju cenu:
  - Mnogo više parametara koje treba da naučimo iz podataka
  - Računarski zahtevniji od *k*-means (E- i M- koraci su mnogo računarski zahtevniji od koraka u *k*-means)