

Statisztikai tanulás az idegrendszerben, 2015.

Paraméterbecslés

Bányai Mihály

banyai.mihaly@wigner.mta.hu

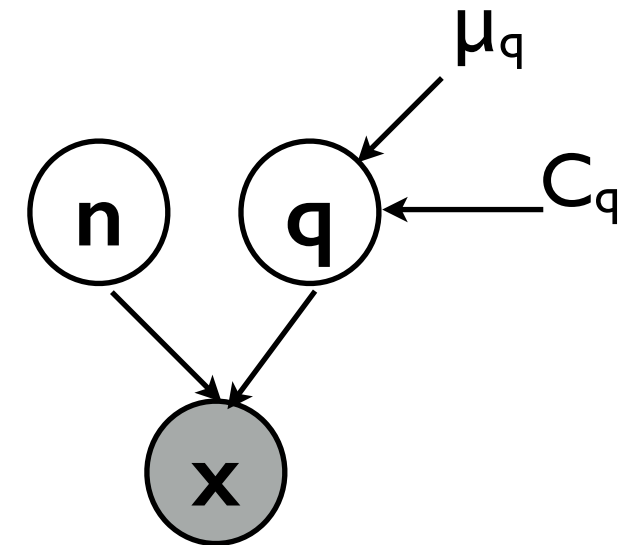
<http://golab.wigner.mta.hu/people/mihaly-banyai/>

- Paraméterek valószínűségi modellekben
- Analitikusan megoldható optimalizáció
- Iteratív közelítések paraméterbecsléshez
- Expectation Maximization algoritmus

- **Paraméterek valószínűségi modellekben**
- Analitikusan megoldható optimalizáció
- Iteratív közelítések paraméterbecsléshez
- Expectation Maximization algoritmus

Megfigyelt és rejtett változók

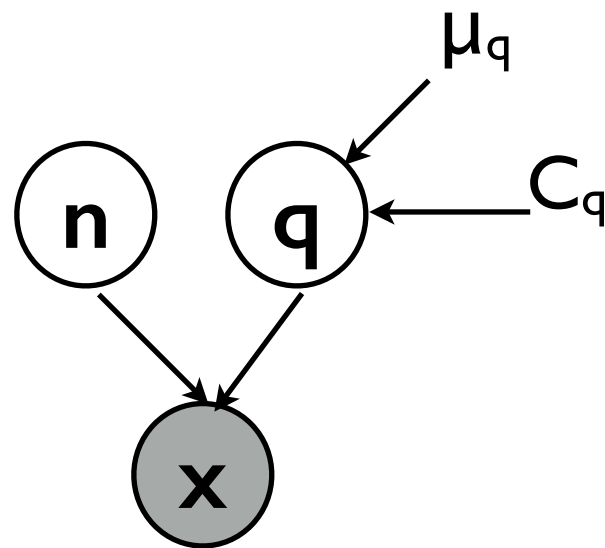
- bizonyos változókról rendelkezünk adatokkal
 - jelölés: sötét kör
- a többi csak a jelenség struktúrájáról alkotott feltételezéseinket reprezentálja
 - látens, rejtett változók: üres kör
- a prediktív eloszlás: a megfigyelt változók marginális eloszlása
- eloszláscsaládok determinisztikus paraméterei
 - ugyanolyanok, mint a látens változók, csak nincs eloszlásuk, hanem egyetlen értékük
 - tipikusan úgy használjuk őket, hogy lassabban változzanak, mint a látens változók
 - eloszlások feltételében jelenhetnek csak meg
 - kör nélküli betű



$$p(q) = \mathcal{N}(q; \mu_q, C_q)$$

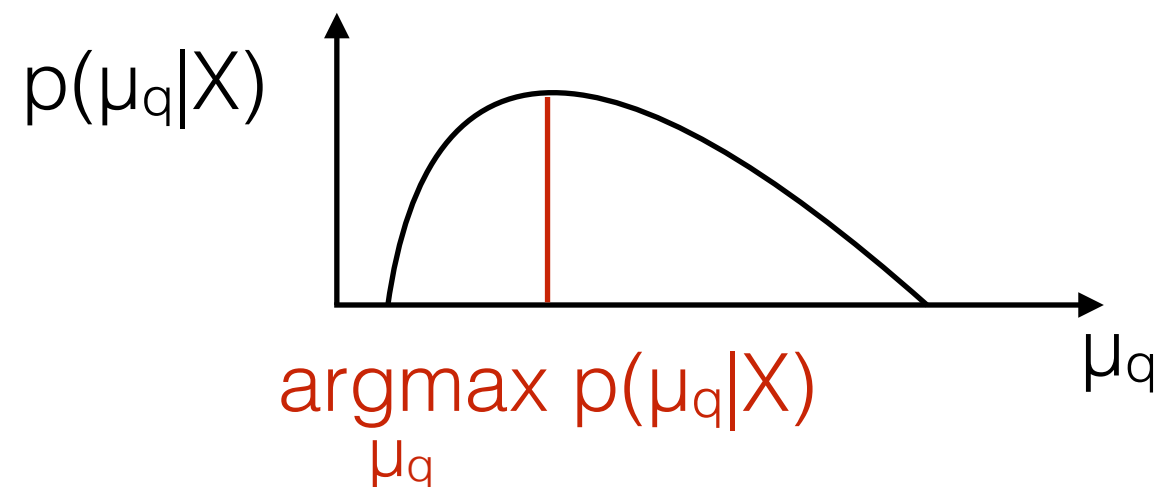
Tanulás mint optimalizáció

- Az inferencia általában a poszterior keresése
 - ha nem kell a teljes poszterior, csak egy pontbecslés, akkor szélsőértékkeresséssé redukálhatom
 - ha nincs prior: maximum likelihood
- Hogyan változik pontbecslésnél a prediktív eloszlás bizonytalansága?



$$p(\mu_q, C_q \mid X) \sim p(X \mid \mu_q, C_q) \cancel{p(\mu_q, C_q)}$$

konstans



Maximum likelihood tanulás

- Általában több mérésünk van a megfigyelt változókról, amiket függetlennek tekintünk

- A likelihood faktorizálódik
$$p(X | \theta) = \prod_{n=1}^N p(x_n | \theta)$$

- A log-likelihood maximumát keressük

- numerikusan stabilabb
$$\ln p(X | \theta) = \sum_{n=1}^N \ln p(x_n | \theta)$$

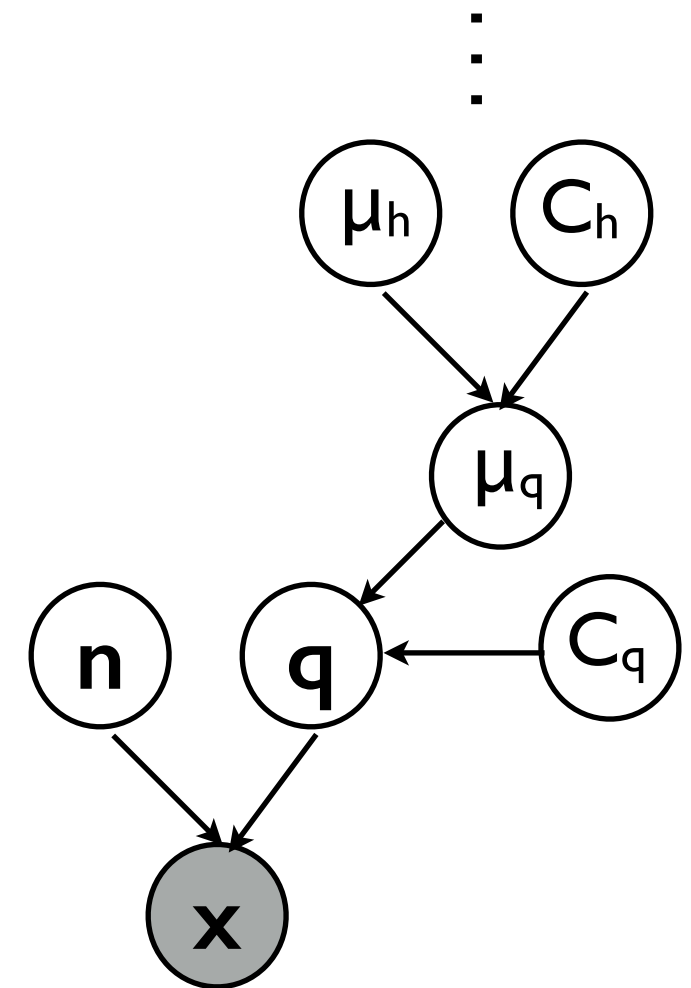
- exponenciális formájú eloszlásoknál egyszerűbb

- Ki kell integrálni a látens változókat (várható érték)

$$p(X | \theta) = \int_{-\infty}^{\infty} p(X | Z, \theta) p(Z | \theta) dZ$$

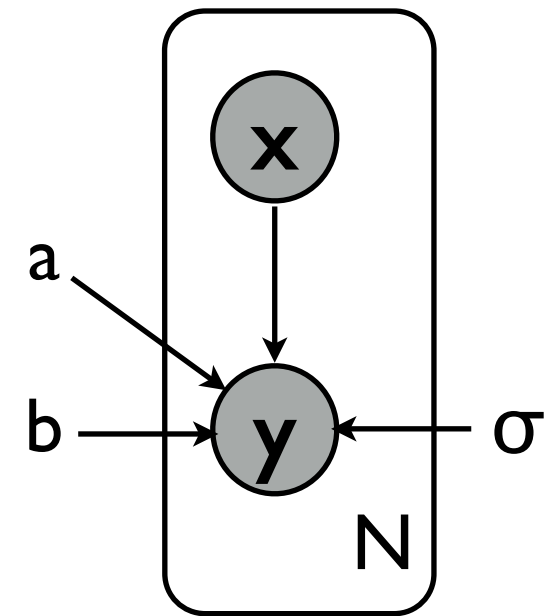
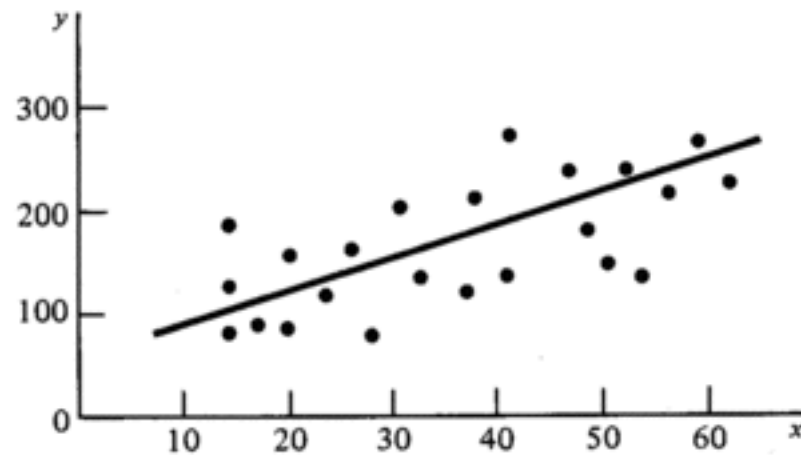
Paraméterek mint valószínűségi változók

- teljes poszterort becsülhetek, vagy adhatok ez esetben is pontbecslést
- a változó prior eloszlásának ismét lesznek paraméterei -> hierarchikus modell
 - valahol meg kell állnom
- vagy olyan priort választok, amiben nincs paraméter, illetve ami mégis, azt nem illesztem az adatra, hanem a világ konstans tulajdonságának tételezem fel az általam választott értékét

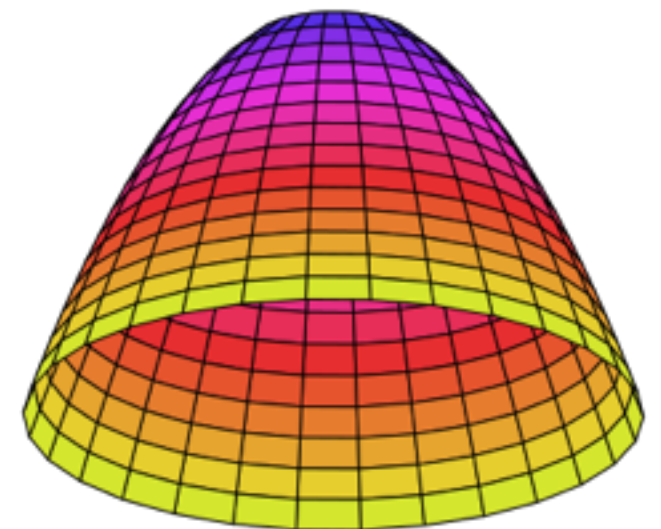


- Paraméterek valószínűségi modellekben
- **Analitikusan megoldható optimalizáció**
- Iteratív közelítések paraméterbecsléshez
- Expectation Maximization algoritmus

Lineáris regresszió



- Nincs rejtett változó a paramétereken kívül
- Gauss maximum likelihood = négyzetes hiba
- polinomokra változtatás nélkül kiterjeszthető



$$p(y \mid x, a, b, \sigma) = \mathcal{N}(y; ax + b, \sigma)$$

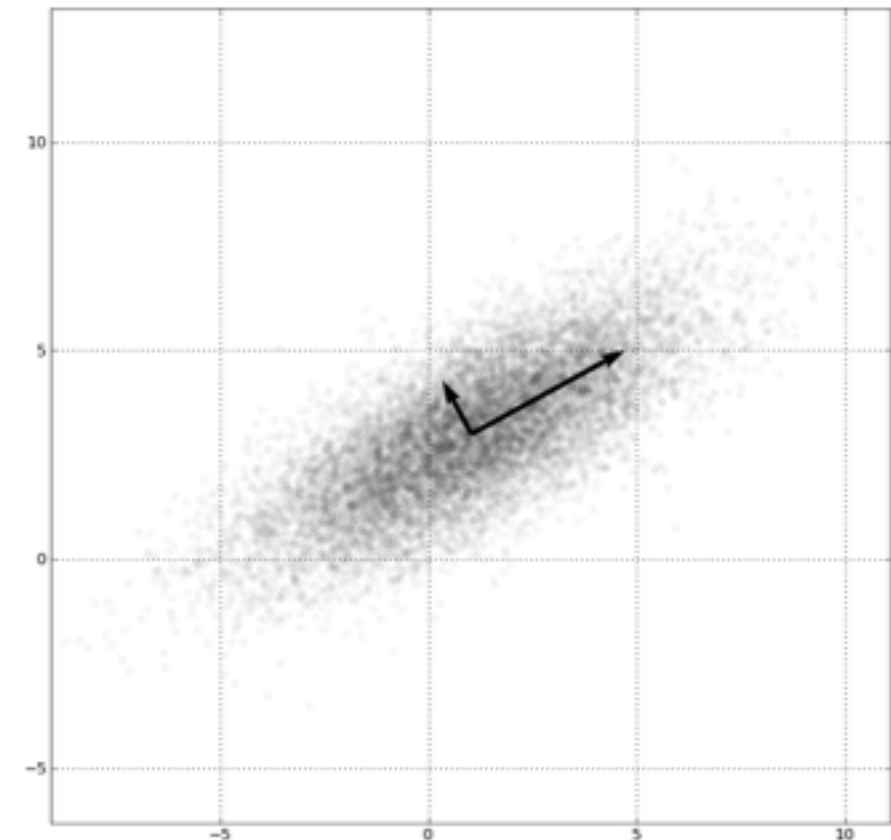
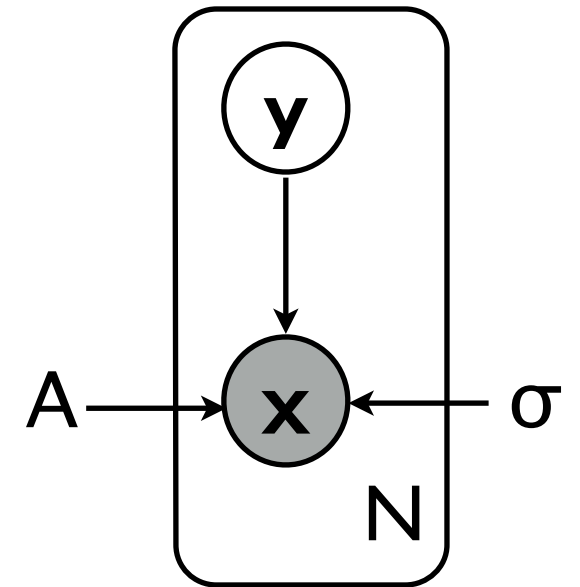
$$\ln p(Y \mid X, a, b, \sigma) \sim - \sum_{n=1}^N [(ax_n + b) - y_n]^2$$

PCA

$$p(x \mid y) = \mathcal{N}(x; Ay, \sigma I)$$

$$p(y) = \mathcal{N}(y; 0, I)$$

- a PCA egy bázistranszformáció, ahol a legnagyobb szórás irányába szeretnénk beforgatni a tengelyeket
- maximum likelihood megoldás az A keverőmátrix elemeire fix szórás mellett
- ekvivalens a megfigyelések kovarianciamátrixának sajátvektor-keresési problémájával



- Paraméterek valószínűségi modellekben
- Analitikusan megoldható optimalizáció
- **Iteratív közelítések paraméterbecsléshez**
- Expectation Maximization algoritmus

Algoritmikus becslés

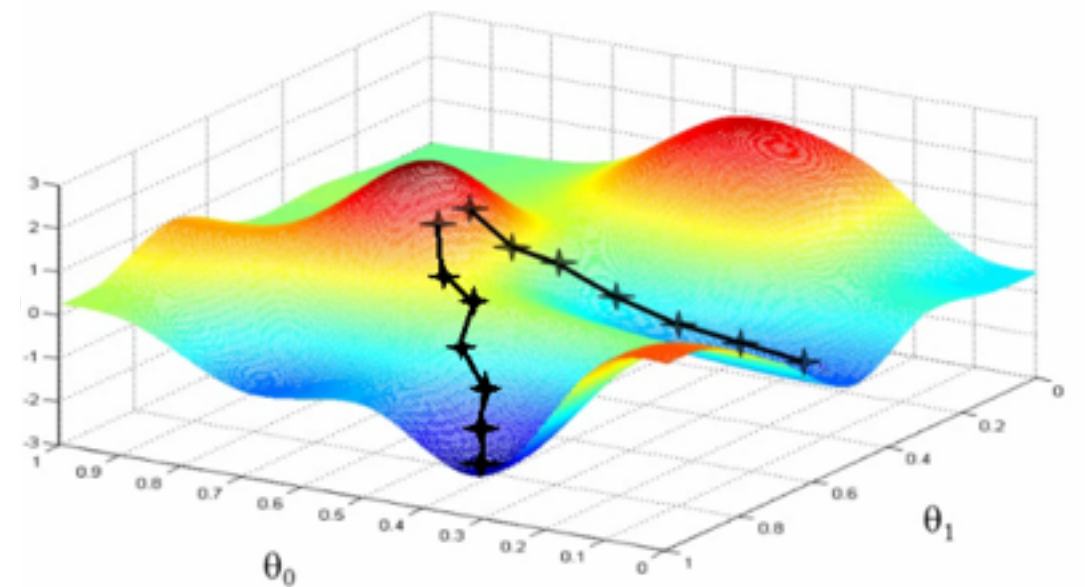
- a becselőalgoritmusok nem a valószínűségszámítás elemei
- be kell bizonyítani a konvergenciát a hibafüggvény szélsőértékéhez
- Ez sokszor csak lokális szélsőérték, a probléma természete adja meg, hogy ez mennyire hasznos
- sokszor nagyon hatékonyak az iteratív algoritmusok, akár zárt formájú megoldásoknál is gyorsabbak lehetnek nagy dimenziójú modellekre

Gradiens-módszer

- használjuk ki azt az információt, hogy adott pontban merre emelkedik a célfüggvény (vagy csökken a hibafüggvény), pl likelihood

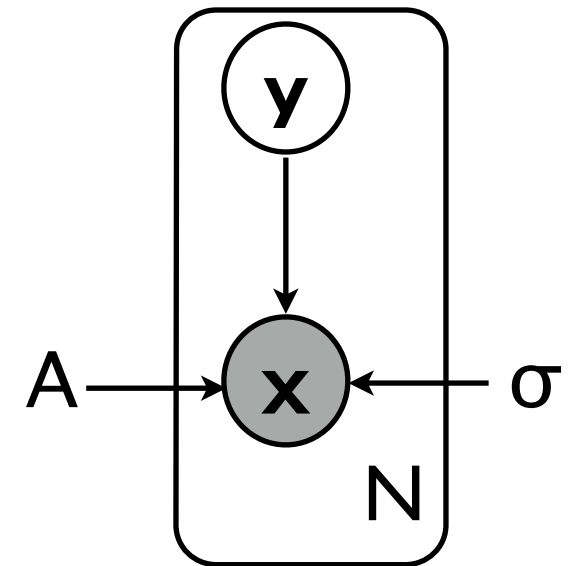
$$\theta_{t+1} = \theta_t + \varepsilon \frac{\partial}{\partial \theta} \ln p(X | \theta_t)$$

- lokális szélsőértéket talál
- a tanulási rátát megfelelően be kell állítani
- kiterjesztések
 - második derivált, Hessian-based módszerek
 - lendületet is definiálhatunk a paraméterterbéli mozgáshoz



ICA

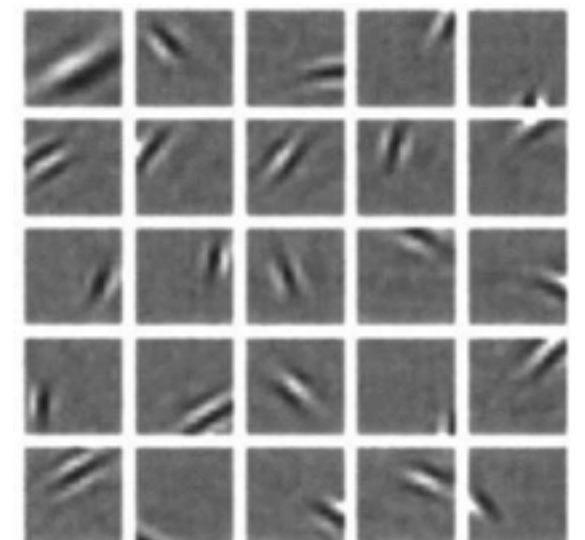
- Olshausen-Field modell: természetes képek független komponensei
- pontbecslés, kombinálva a maximum likelihood és a sparsity kritériumokat
- log-poszterior = kvadratikus log-likelihood + sparse prior
- Egyszerre optimalizáljuk az adatra illeszkedést és a látens aktivációk ritkaságát
- Grandiens-módszerrel lépkedünk az így konstruált hibafüggvényen



$$p(x \mid y) = \mathcal{N}(x; Ay, \sigma I)$$



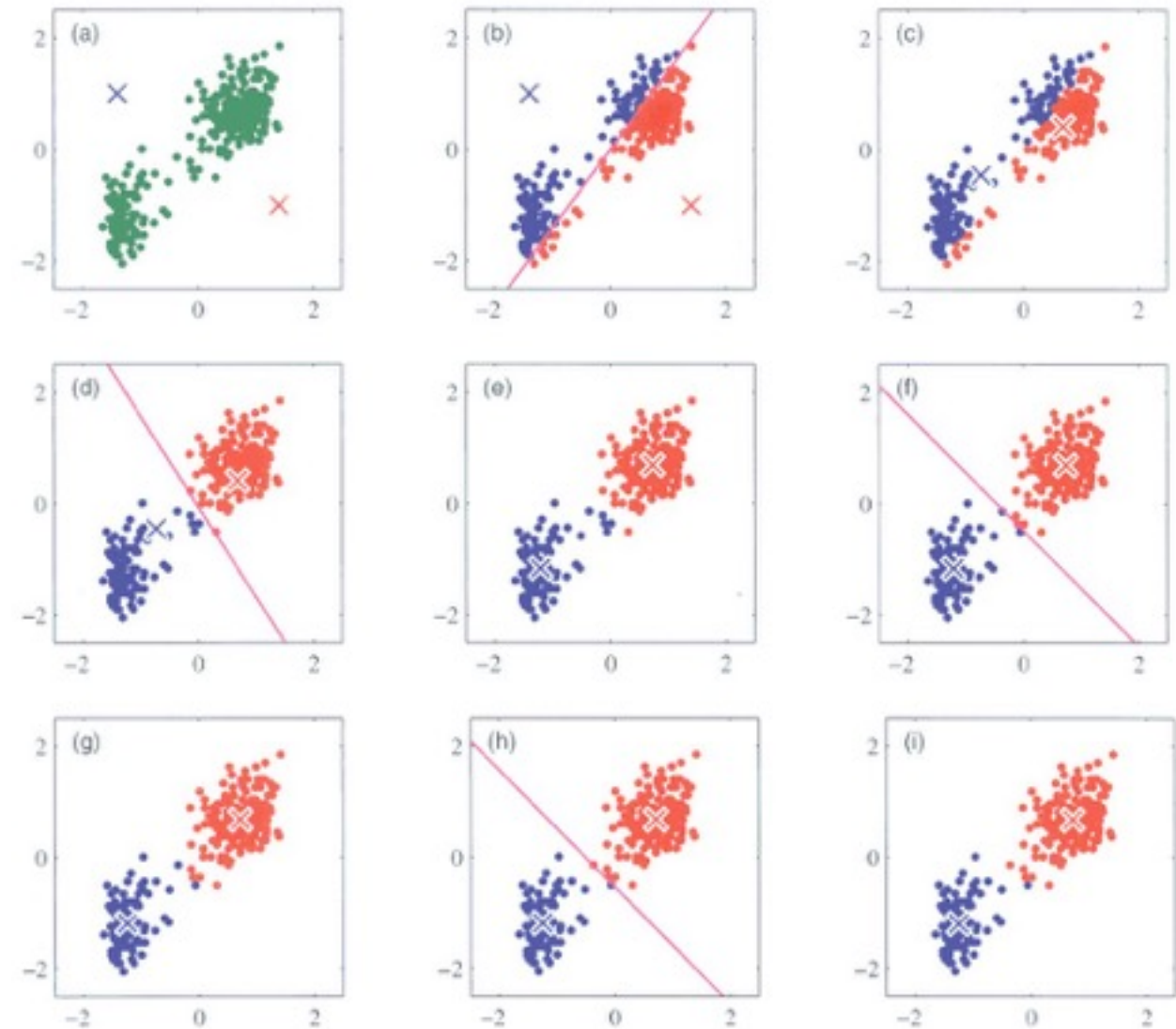
sparsity



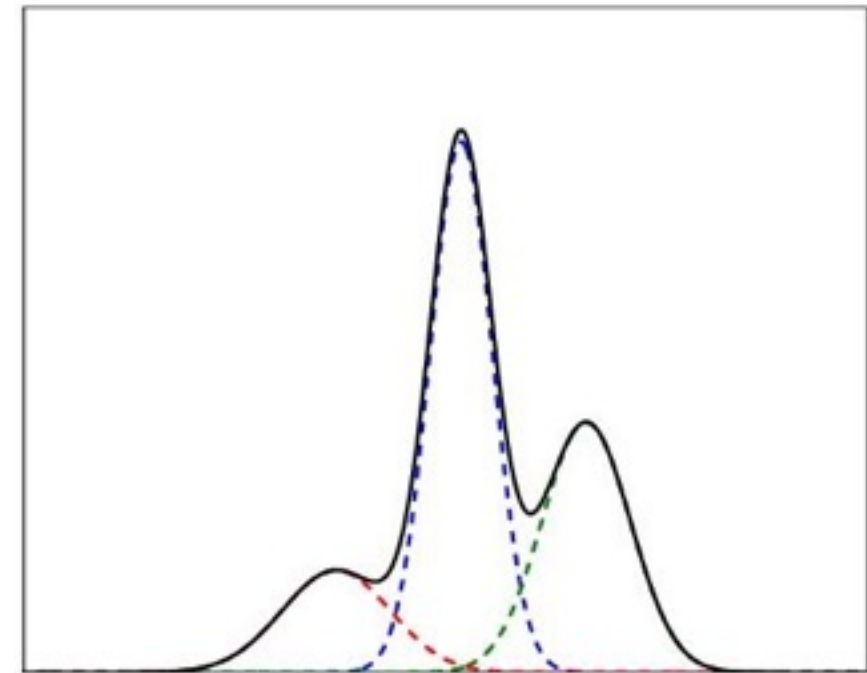
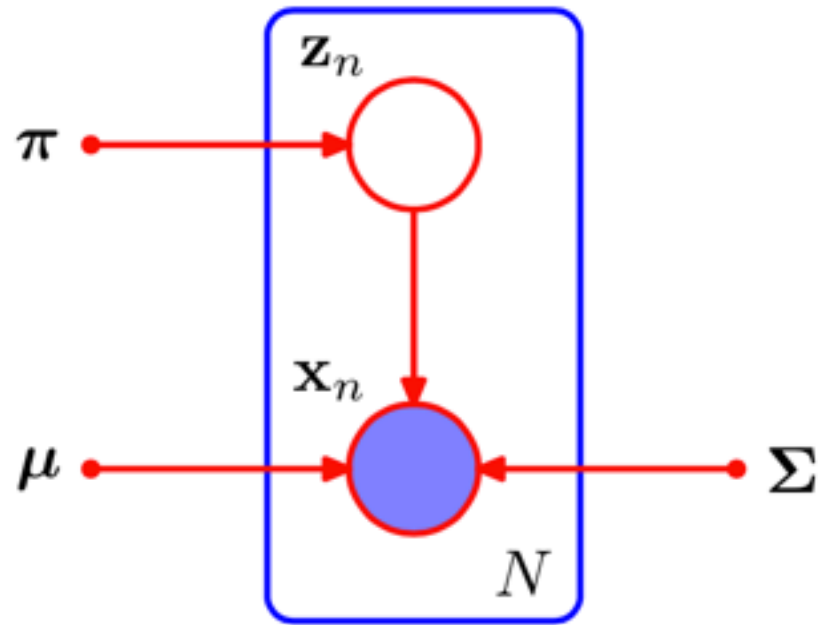
- Paraméterek valószínűségi modellekben
- Analitikusan megoldható optimalizáció
- Iteratív közelítések paraméterbecsléshez
- **Expectation Maximization algoritmus**

K-means klaszterezés

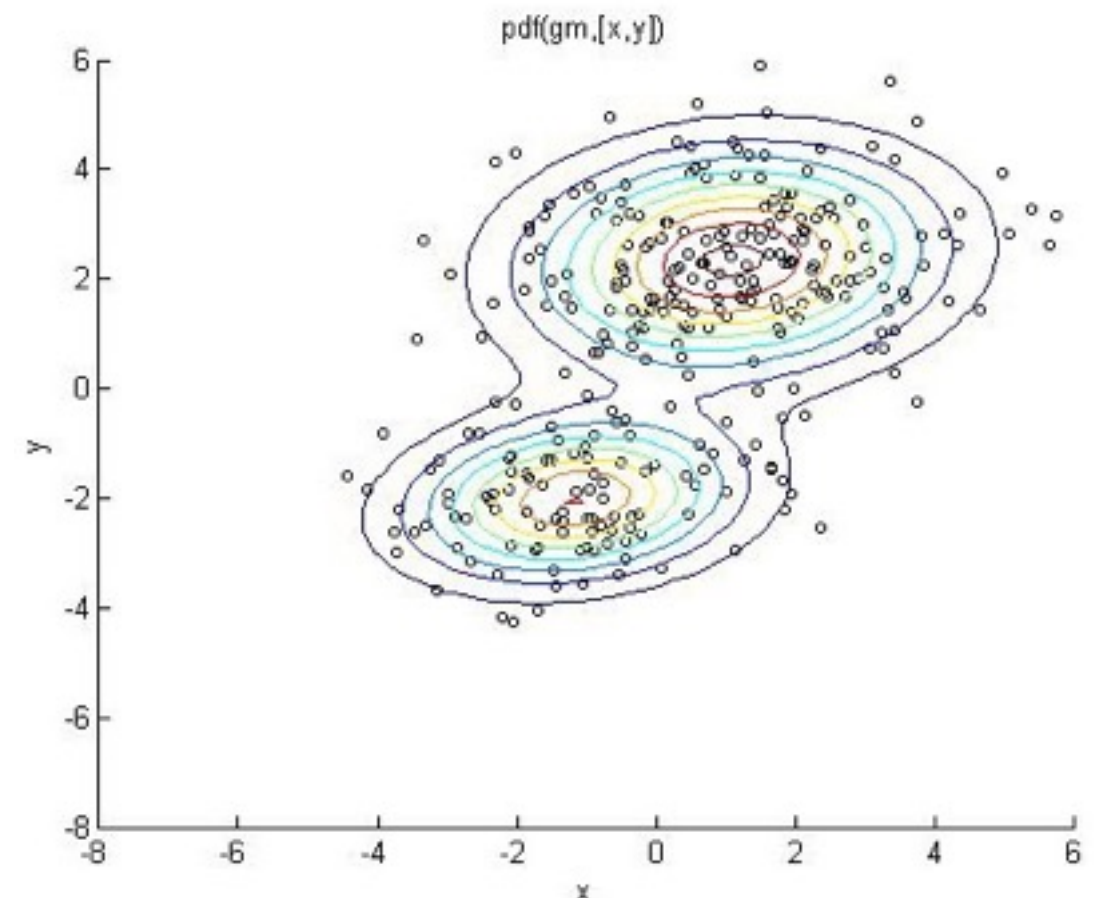
- Klaszterközéppontokat keresünk
- Kiválasztjuk a klaszterek számát
- Véletlenszerűen inicializáljuk a középpontokat
- Hozzárendeljük a megfigyeléseket a legközelebbi klaszterközépponthoz
- Elmozgatjuk a klaszterközéppontokat úgy, hogy a négyzetes távolság a legkisebb legyen a hozzárendelt pontoktól
- Addig ismételjük, amíg átsorolás történik
- Mi az ekvivalens valószínűségi modell?



Keverékmodellek



$$p(z) = Mult(z; \pi)$$
$$p(x | z) = \mathcal{N}(x; \mu_z, \Sigma_z)$$



Expectation Maximization

- Általánosítsuk a k-means ötletét
 - complete-data likelihood: mintha minden változó megfigyelt lenne

$$p(X, Z \mid \theta)$$

- Az algoritmus kétfajta lépést váltogat
 - E: megbecsüljük a látens változók poszterior eloszlását a paramétereket fixen tartva

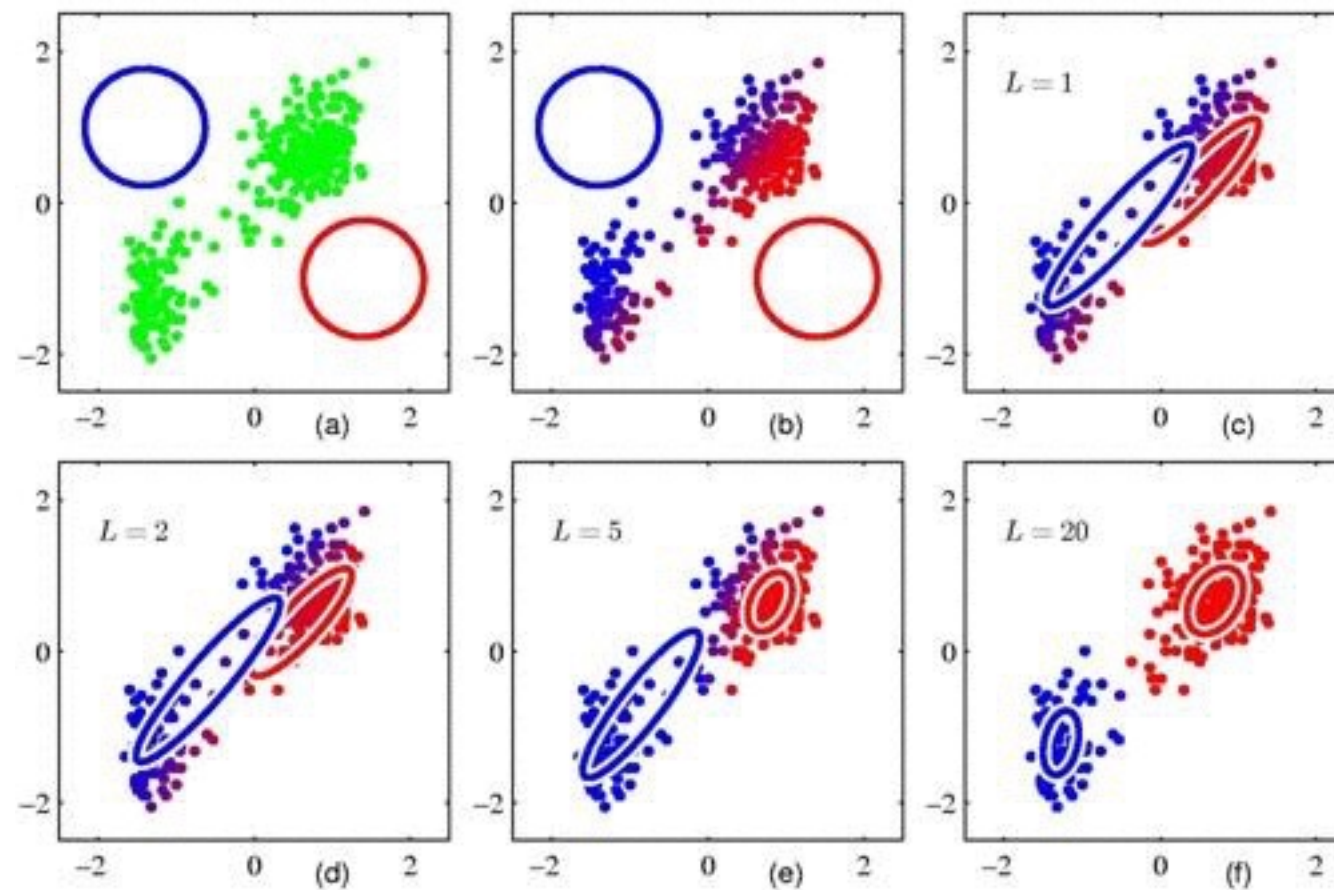
$$p(Z \mid X, \theta_t)$$

- M: megbecsüljük a paraméterek értékét az becsült poszterior alapján
 - a CDL logarimusának vesszük a poszterior feletti várható értékét, és ezt maximalizáljuk a likelihood helyett

$$\theta_{t+1} = \operatorname{argmax}_{\theta} \int_{-\infty}^{\infty} p(Z \mid X, \theta_t) \ln p(X, Z \mid \theta) dZ$$



EM keverékmmodellre



- E: mi a poszterior valószínűsége, hogy egy adott pontot egy adott komponens generált a jelenleg becsült paraméterekkel
 - a komponensek pontokért viselt “felelőssége”
- M: a felelősségekkel súlyozott pontokra mi lenne a legjobb mean és kovariancia
 - keverési együtthatók: a felelősségek összegének aránya a pontok számához

Általánosított EM

- Mi van, ha nem tudjuk zárt alakban megadni az M-lépésben keresett szélsőértéket?
 - gradients módszer
- Mi van, ha nem tudjuk zárt alakban megadni a CDLL poszterior szerinti várható értékét?
 - mintavételezés a poszteriorból, az integrál közelítése véges számú minta feletti átlaggal

Honnan tudjuk, hogy az algoritmus eredménye hasznos?

- Konvergencia
 - likelihoodban
 - paraméterek értékeiben
- Az illesztett modell predikciói beválnak
- Honnan tudjuk, hogy jól választottuk meg az olyan hiperparamétereket, mint pl. a komponensek száma?

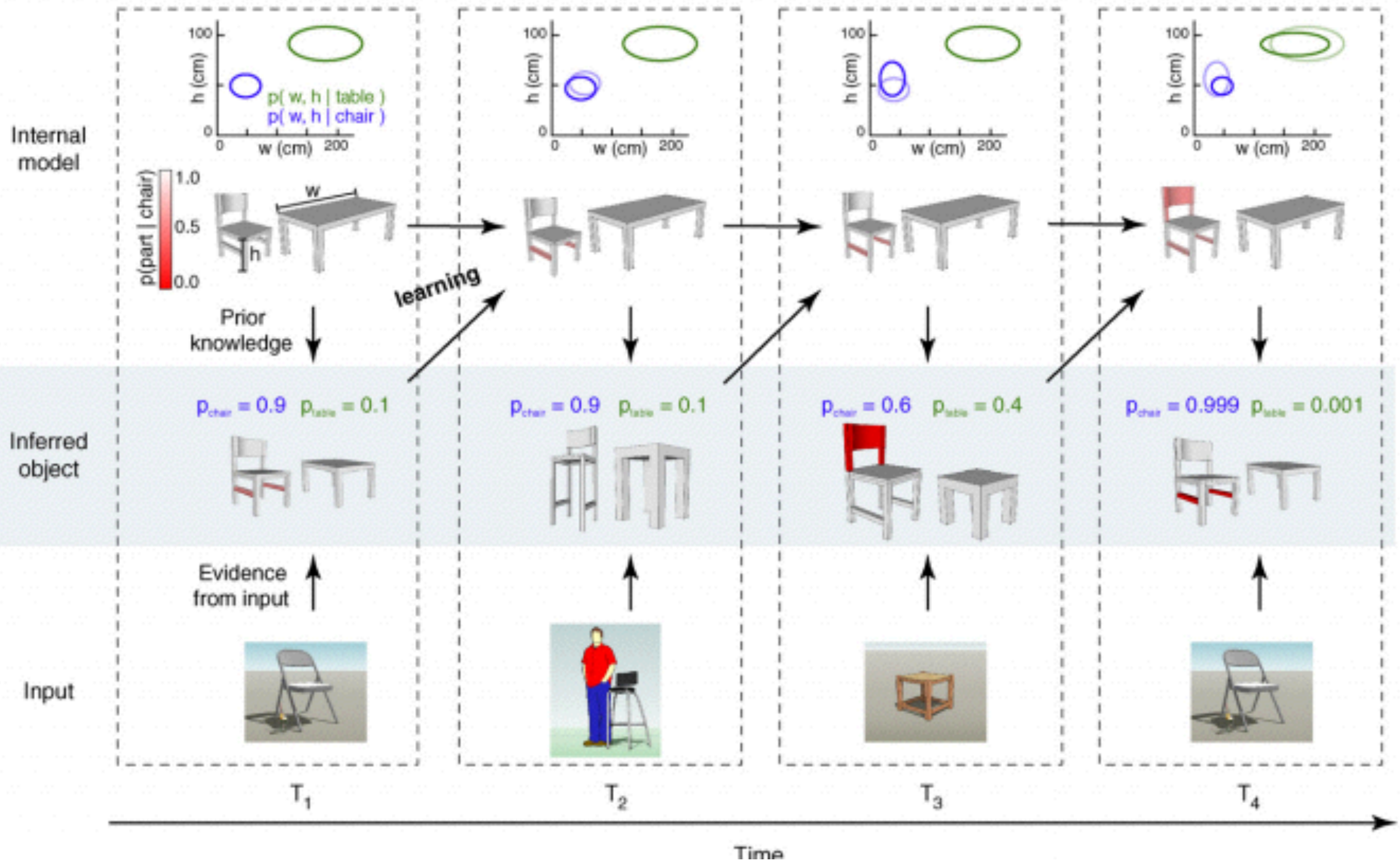
Teljes poszterior becslése

- Sztochasztikus
 - mintákat veszünk belőle (MCMC)
- Determinisztikus
 - egyszerűbben kezelhető eloszlásokkal közelítjük
 - pl. Gauss eloszlással
 - faktorizált eloszlással: variational approximation

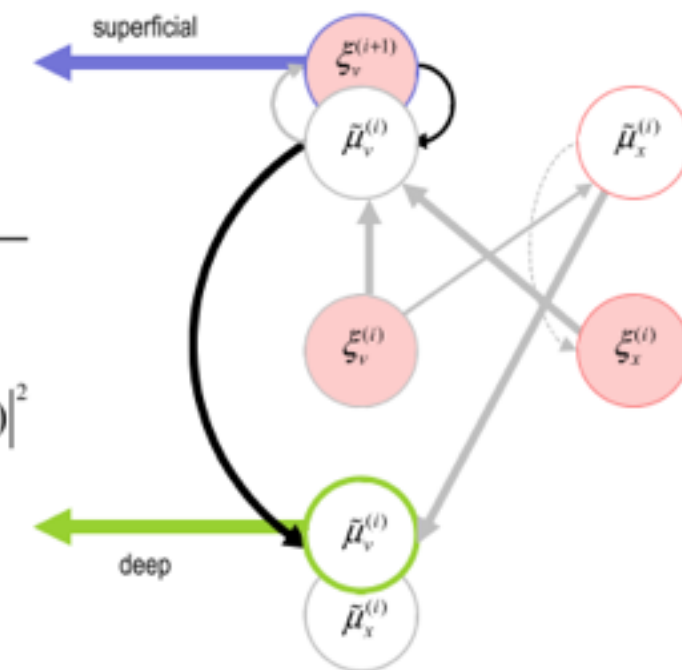
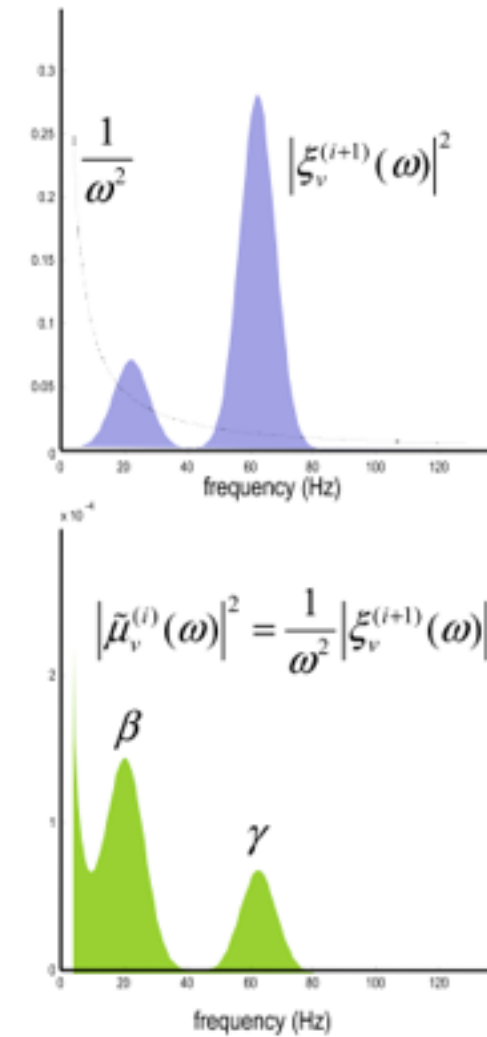
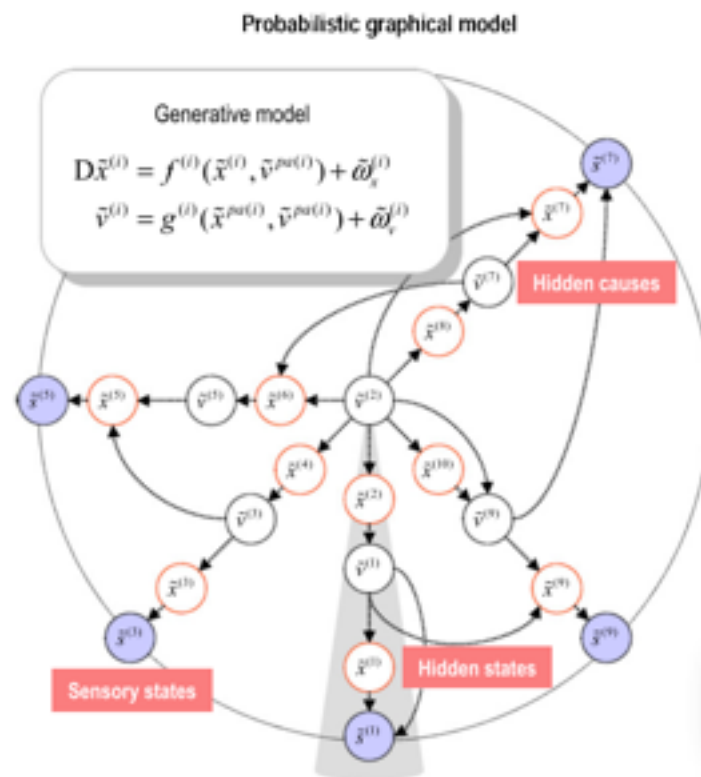
Hogyan tehetünk predikciókat az illesztett modellel?

- A jövőre a prediktív eloszlásból
- A látens változók értékeire a poszteriorból
- Ha agykérgi tanulórendszer modelljét alkotjuk, akkor el kell döntenünk, hogy
 - a paraméterillesztési algoritmust biofizikai folyamatok predikciójára akarjuk használni
 - vagy az algoritmust csak arra használjuk, hogy eljussunk egy optimális modellig, és csak azt feltételezzük, hogy az agy is megteszi ezt valahogy, de nem feltétlenül így

Iteratív becslés az agyban - viselkedési szinten



Iteratív becslés az agyban - fiziológiai szinten



Házi feladat

- Töltsd le a tárgyhonlapról az em_hf.txt file-t
- Az adatokra illesz 3-komponensű Gaussian Mixture modellt
- Eredményedet illusztráld