

Statisztikai tanulás az idegrendszerben, 2019.

Algoritmikus következtetés valószínűségi modellekben II.

Bányai Mihály

banyai.mihaly@wigner.mta.hu

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

érzékelés

**percepció
(észlelés)**

jelenlegi
megfigyelésből
kikövetkeztetett
információ

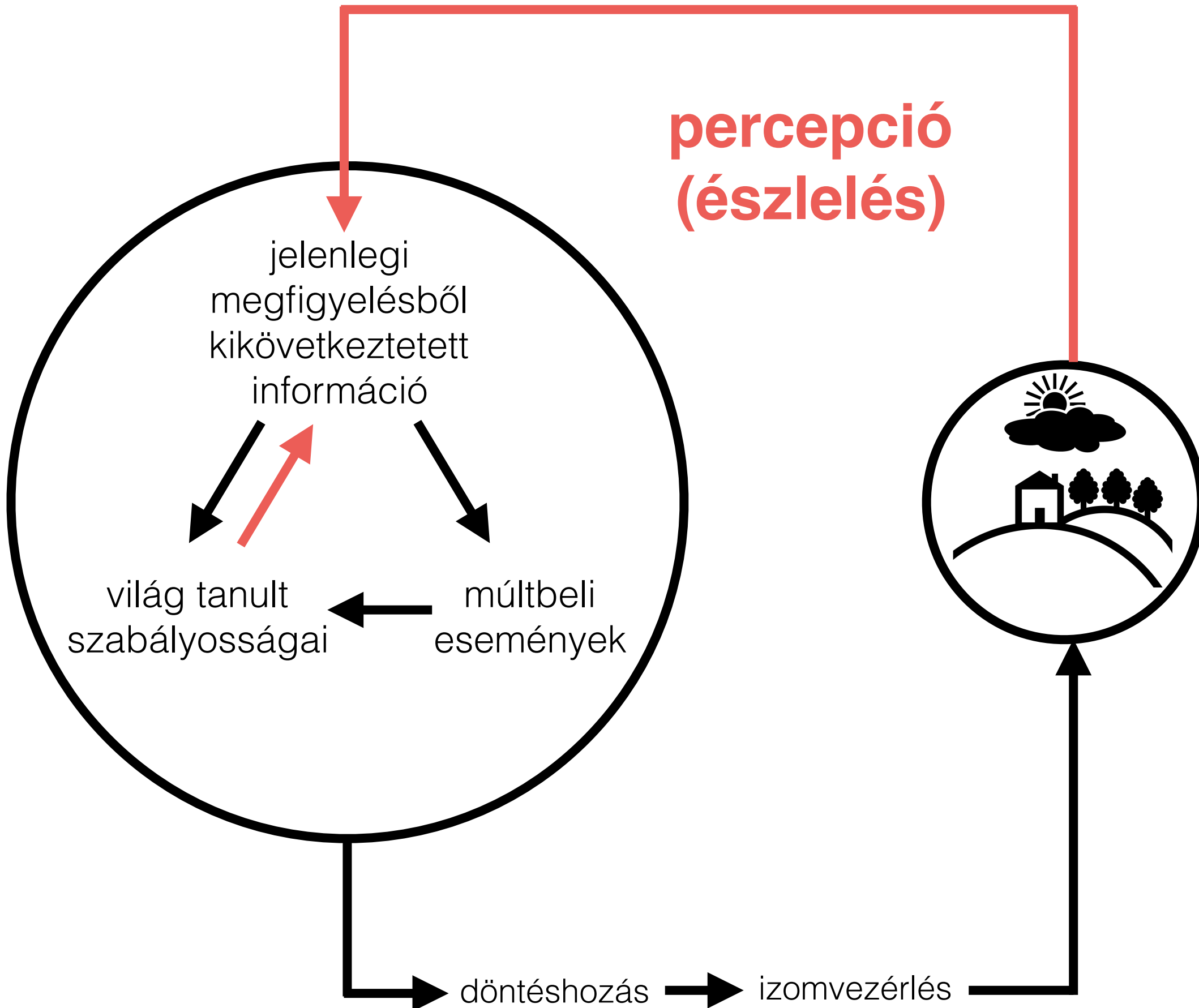
világ tanult
szabályosságai

múltbeli
események

döntéshozás → izomvezérlés

cselekvés

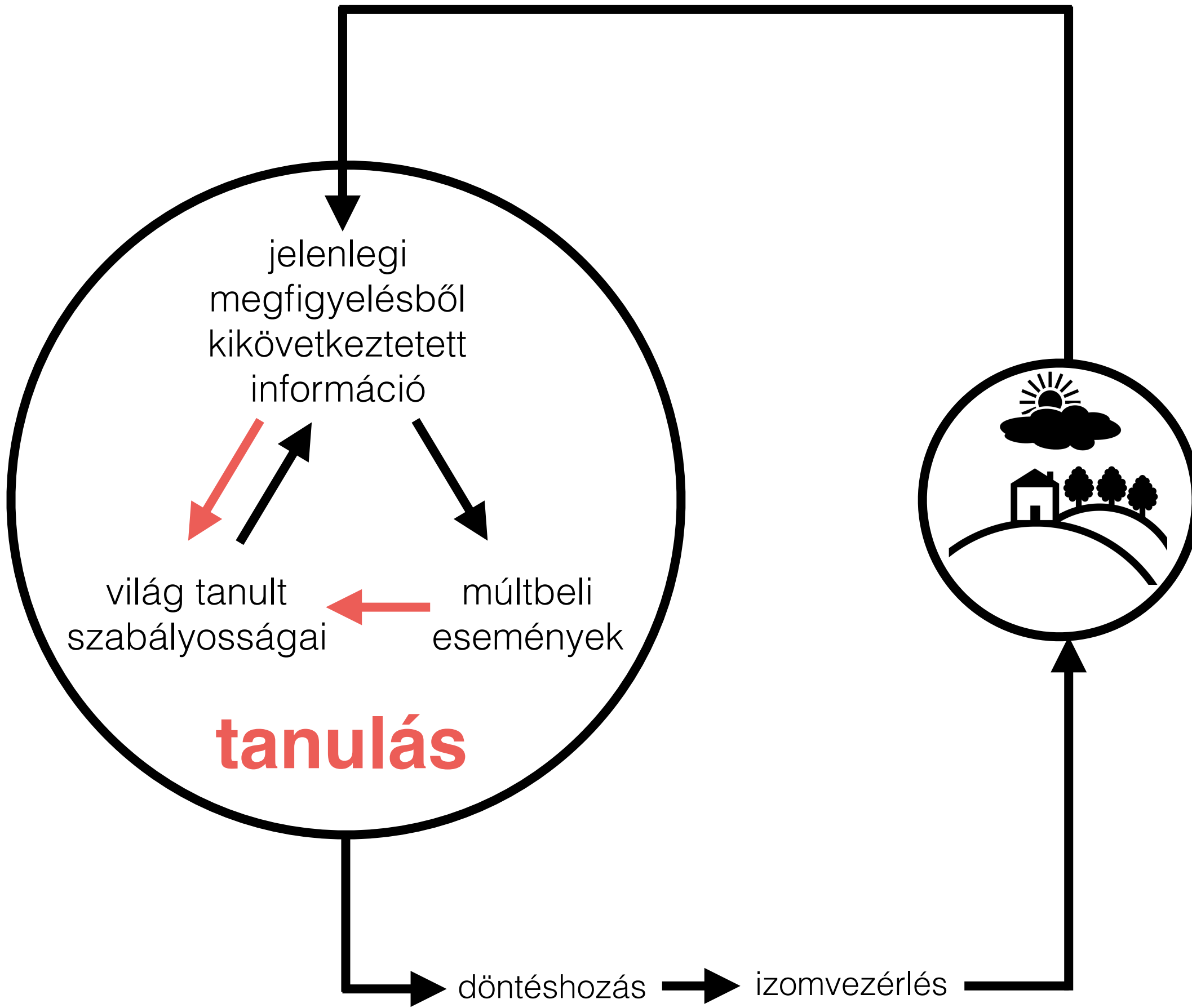
környezet



Poszterior közelítő becslése

- Sztochasztikus
 - mintákat veszünk belőle
 - véletlenszámgenerátorra van szükség
 - a minták számától függ a pontosság
- Determinisztikus
 - egyszerűbben kezelhető eloszlásokkal közelítjük
 - nem használunk véletlenszámokat
 - a közelítő eloszlás formájától függ a pontosság

érzékelés

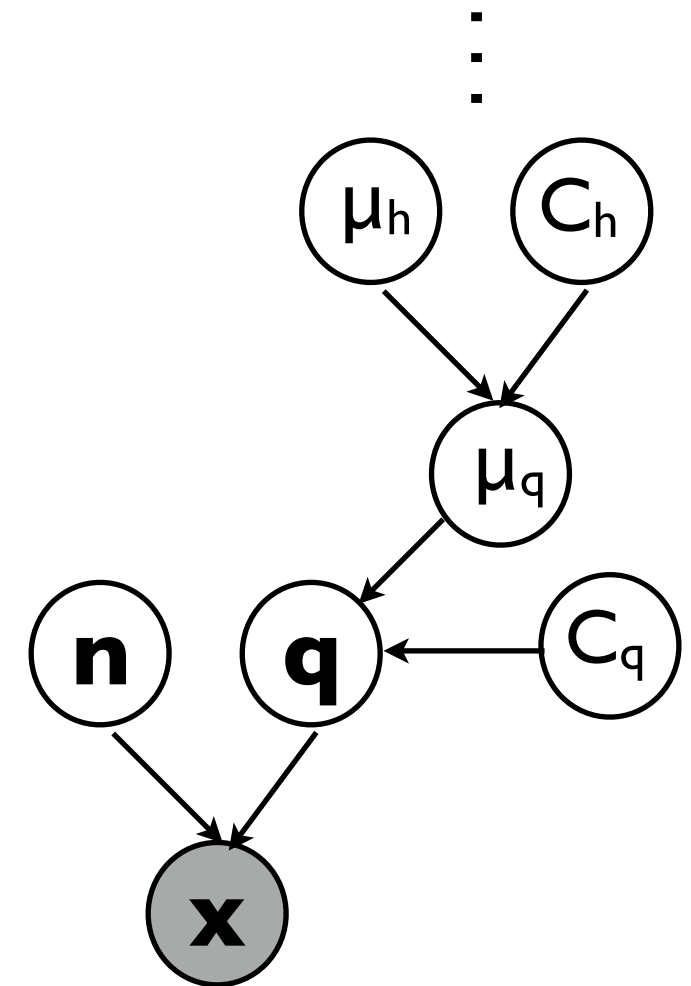


környezet

cselekvés

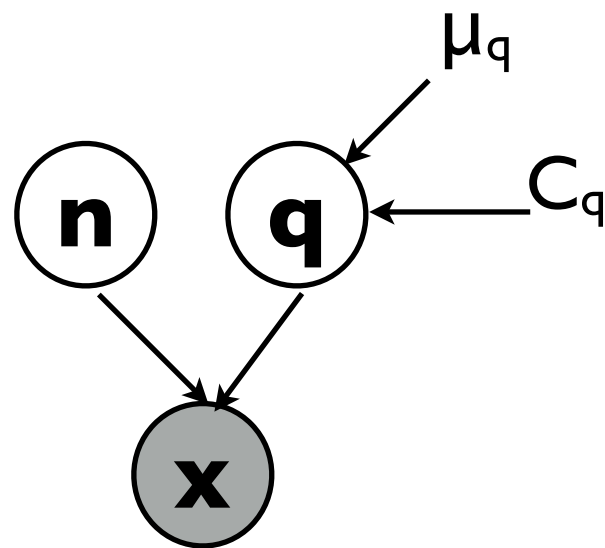
Paraméterek mint valószínűségi változók

- a valószínűségi modelben specifikálnom kell a változók prior és kondicionális eloszlásait
 - ezek gyakran parametrikus eloszláscsaládok példányai lesznek, pl. Gauss, Poisson, stb
 - ezek paramétereit szintén látens változók, amennyiben nem fixálom az értéküket teljesen
- a változó prior eloszlásának ismét lesznek paramétereit -> 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



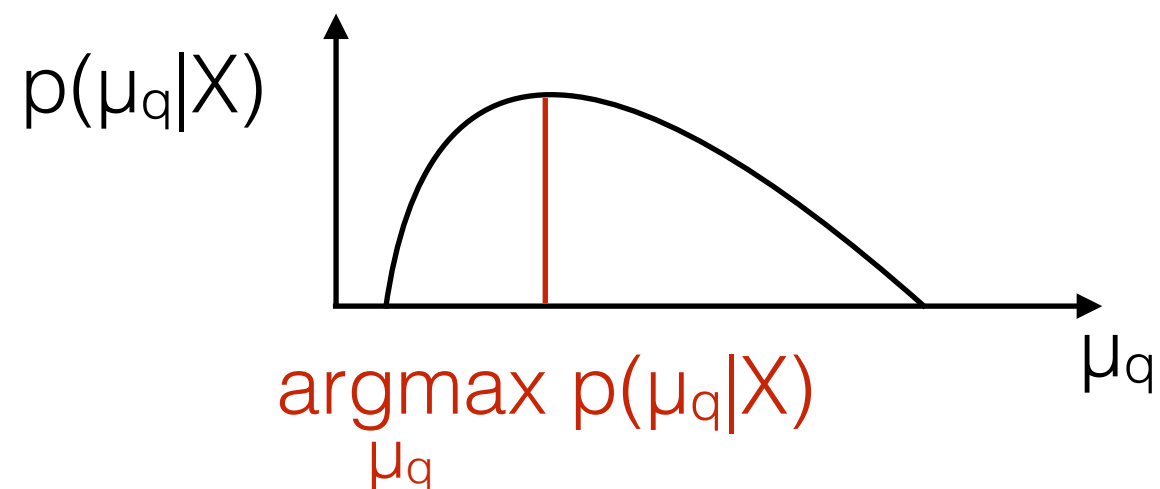
Pontbecslések

- 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

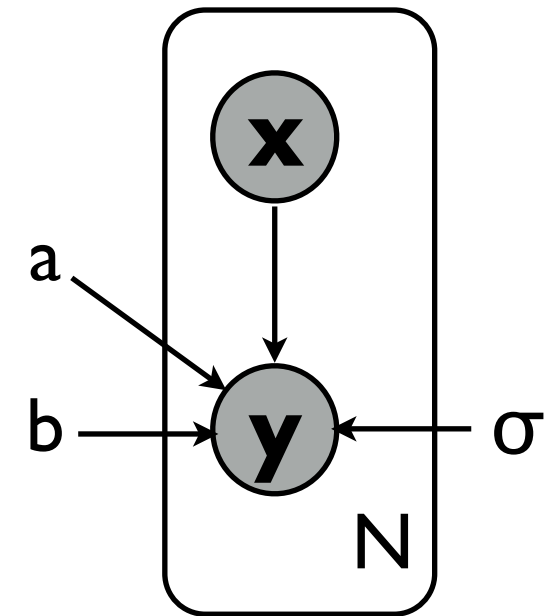
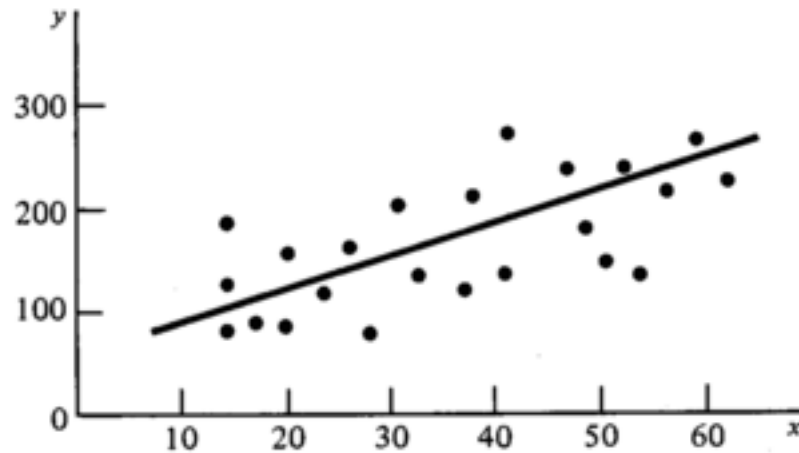
- Ki kell integrálni a látens változókat (várható érték)

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

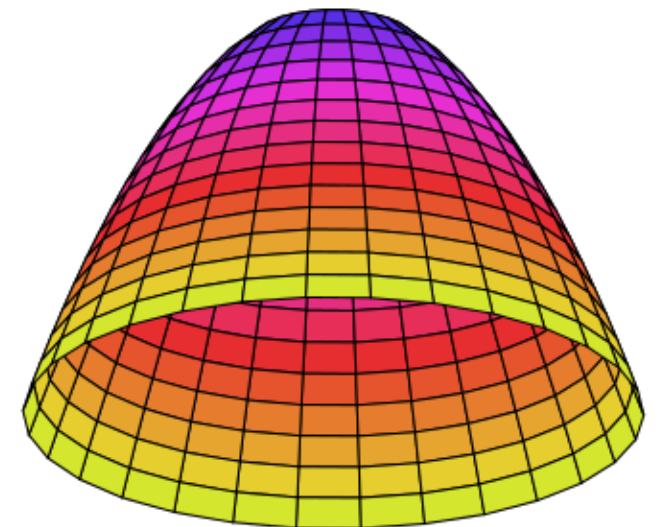
Mikor hasznosak a pontbecslések?

- Ha jellemző a becsült pont értéke a poszterior alakjára
 - unimodális
 - szimmetriája a pontbecslés természetével egyező

Példa - lineáris regresszió



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



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

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

Amikor a pontbecslés is nehéz

- Általában itt is igaz, hogy vannak olyan láttens változók amelyek felett marginalizálni akarunk, aminek lehet, hogy nincs zárt alakja
- vagy az *argmax* nem fejezhető ki zárt alakban

Algoritmikus becslés

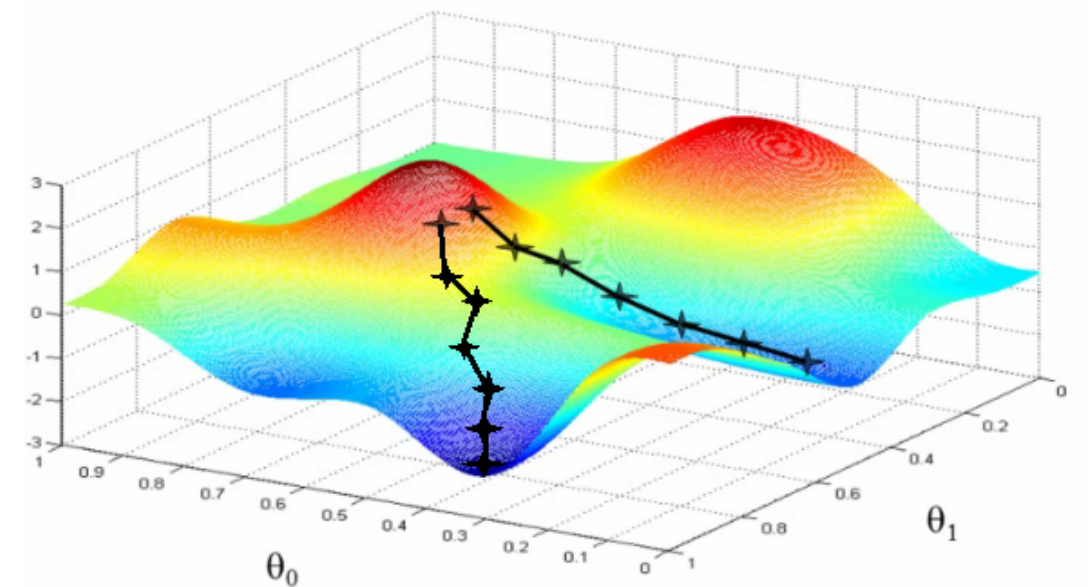
- ha a pontbecslés nem adható meg egzaktul, megfogalmazhatjuk optimalizációs problémaként, amelyben a hibafüggvény pl. a negatív log-likelihood
- a becslőalgoritmusok nem (csak) a valószínűségszámítás szabályait használják
- sokszor nagyon hatékonyak az iteratív algoritmusok, akár zárt formájú megoldásoknál is gyorsabbak lehetnek nagy dimenziójú modellekre
- viszont sokszor nem tudjuk levezetni, hogy mennyire pontosan találják meg a globális maximumot

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 paraméterterbeli mozgáshoz



notebook 3

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?

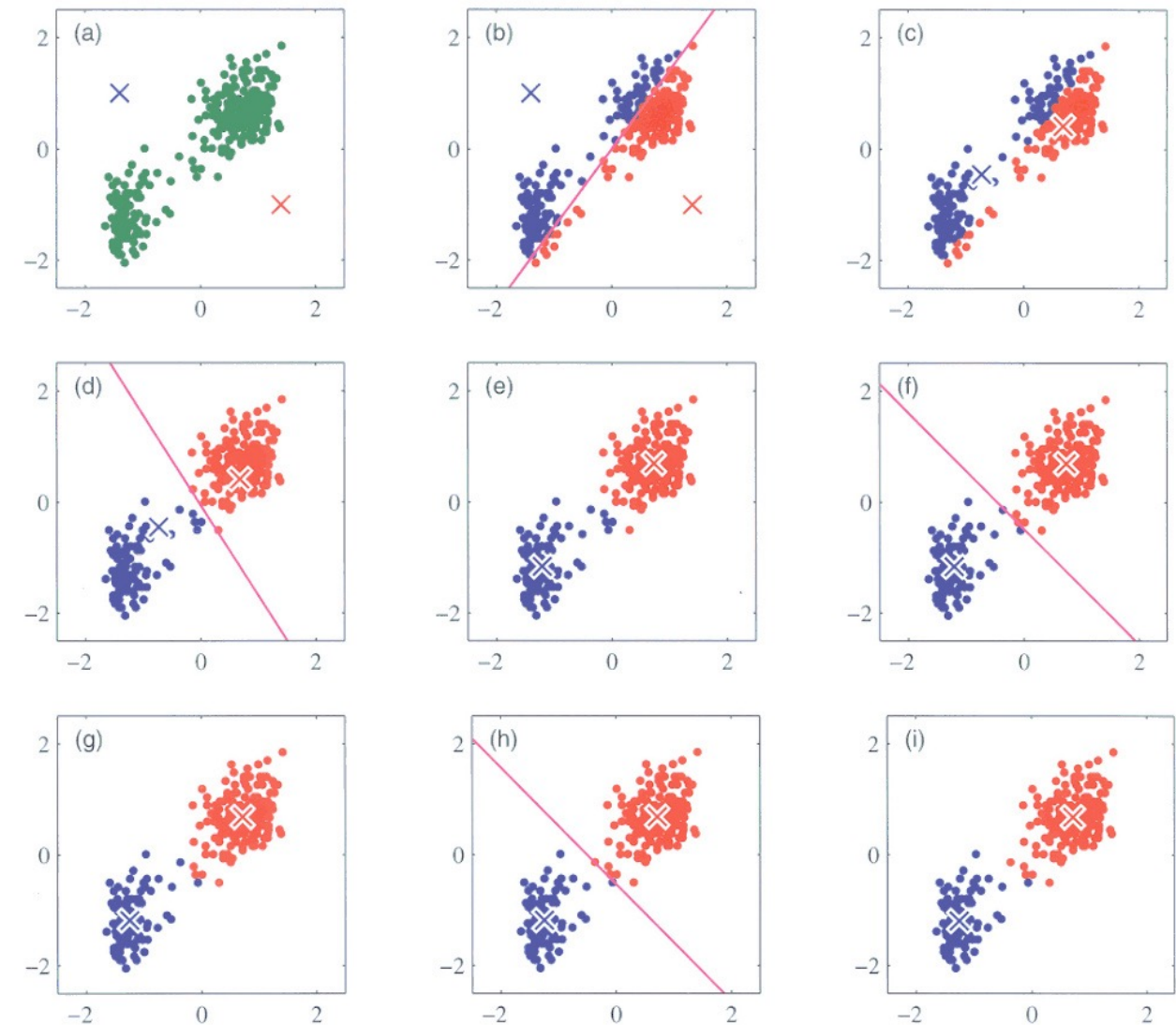
Hogyan tehetünk predikciókat az illesztett modellel?

- 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

Algoritmikus paraméterbecslés a gradiens-módszeren túl

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 hozzárendelt pontoktól
- Addig ismételjük, amíg átsorolás történik
- Mi az ekvivalens valószínűségi modell?



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

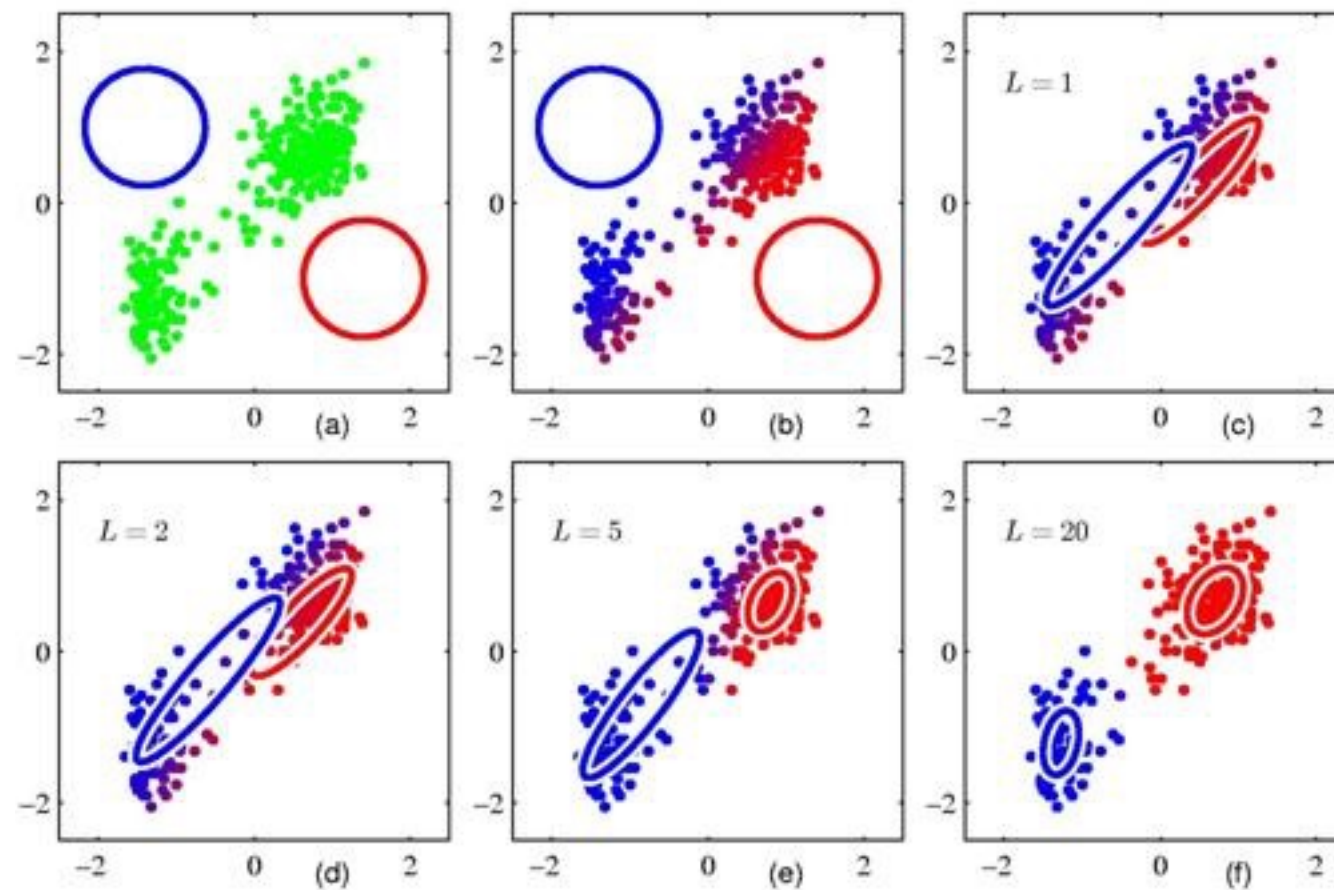
$$p(Z \mid X, \theta_t)$$

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

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



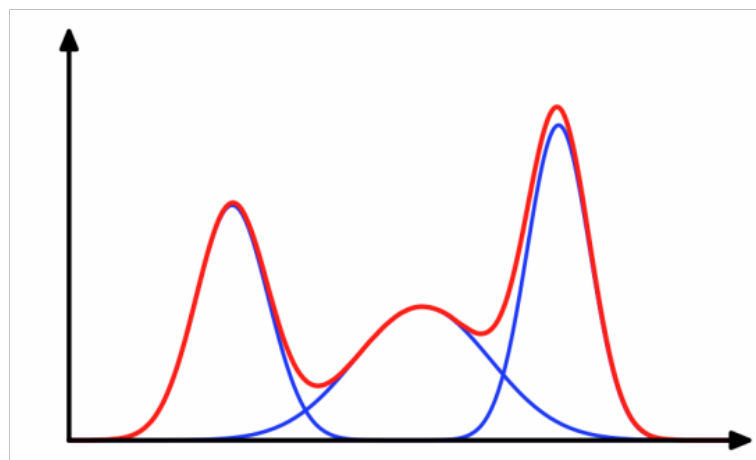
EM keverékmmodellre



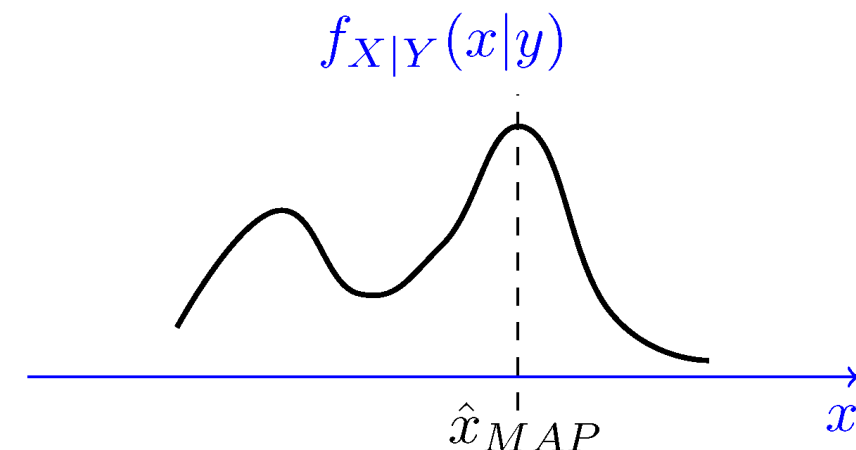
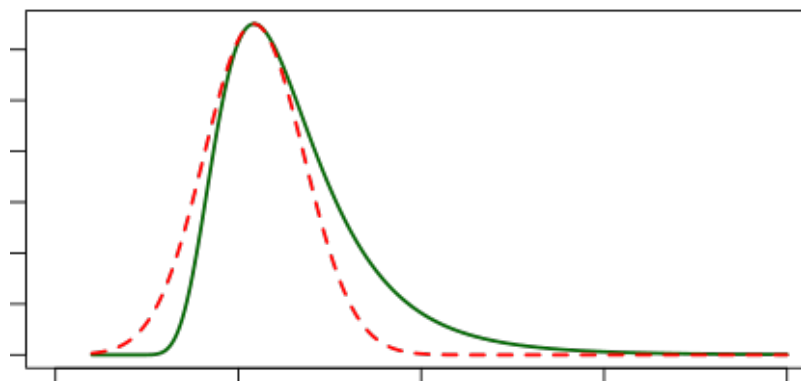
- E: mi a poszterior valószínűsége, hogy egy adott pontot egy adott komponens generált a jelenleg becsült paraméterekkel
 - a komponensek pontokért viselt “felelőssége”
- M: a felelősségekkel súlyozott pontokra mi lenne a legjobb mean és kovariancia
 - keverési együtthatók: a felelősségek összegének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

Determinisztikus közelítés - variációs módszerek

- A kérdés, hogy mennyi információt dobunk el a poszteriorból?
 - Gyakran a függőségek teszik nehezzé az inferenciát -> faktorizált közelítés:
 $p(h_1, h_2 | o) \approx p(h_1 | o) p(h_2 | o)$
 - Jól kezelhető parametrikus eloszlásokkal is közelíthetünk
 - bonyolultabbakkal, pl többkomponensű keverékeloszlások
 - egyszerűbbekkel, pl Gauss (Laplace-közelítés)
- Egyetlen jellemző értéken kívül mindent eldobunk -> delta eloszlással közelítünk -> pontbecslés

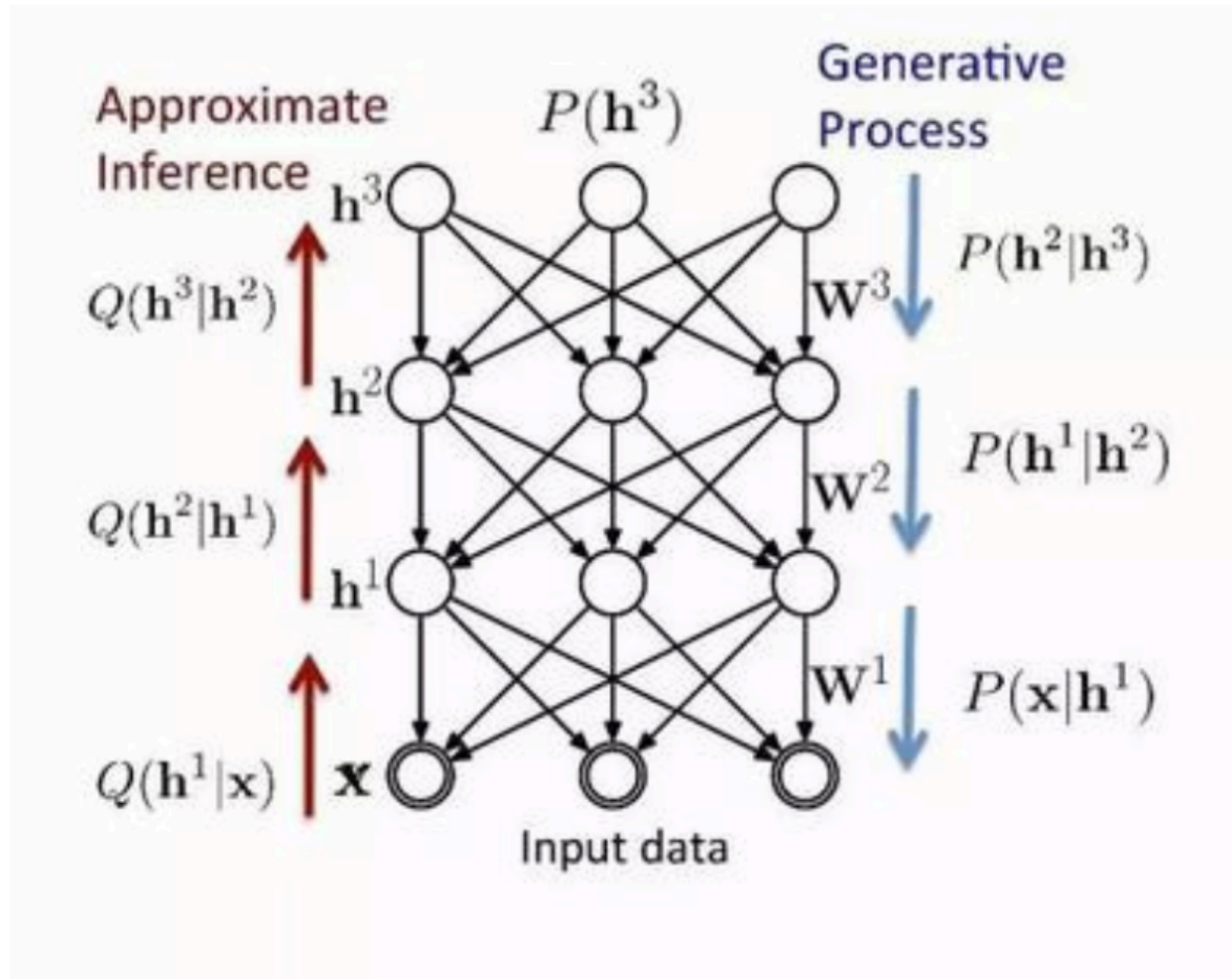


Laplace approximation of posterior of Normal(10,σ)

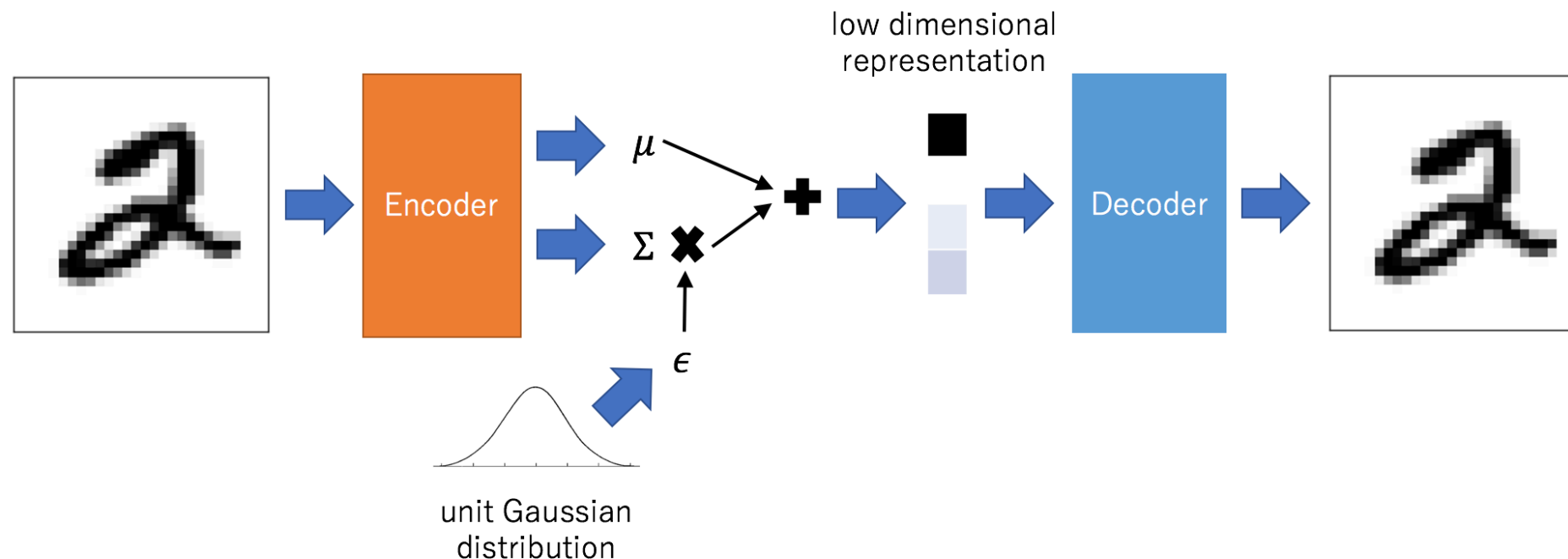


Helmholtz-gép

- Hermann von Helmholtz, 1890: az érzékelés következtetés
- Horace Barlow, 1989: az érzékelés valószínűségi következtetés
- Geoff Hinton, 1995: a generatív modellben a látens változók inferenciáját külön rekognitív modell végzi
- GPU, ImageNet, autograd: inkább várjatok meg minket is

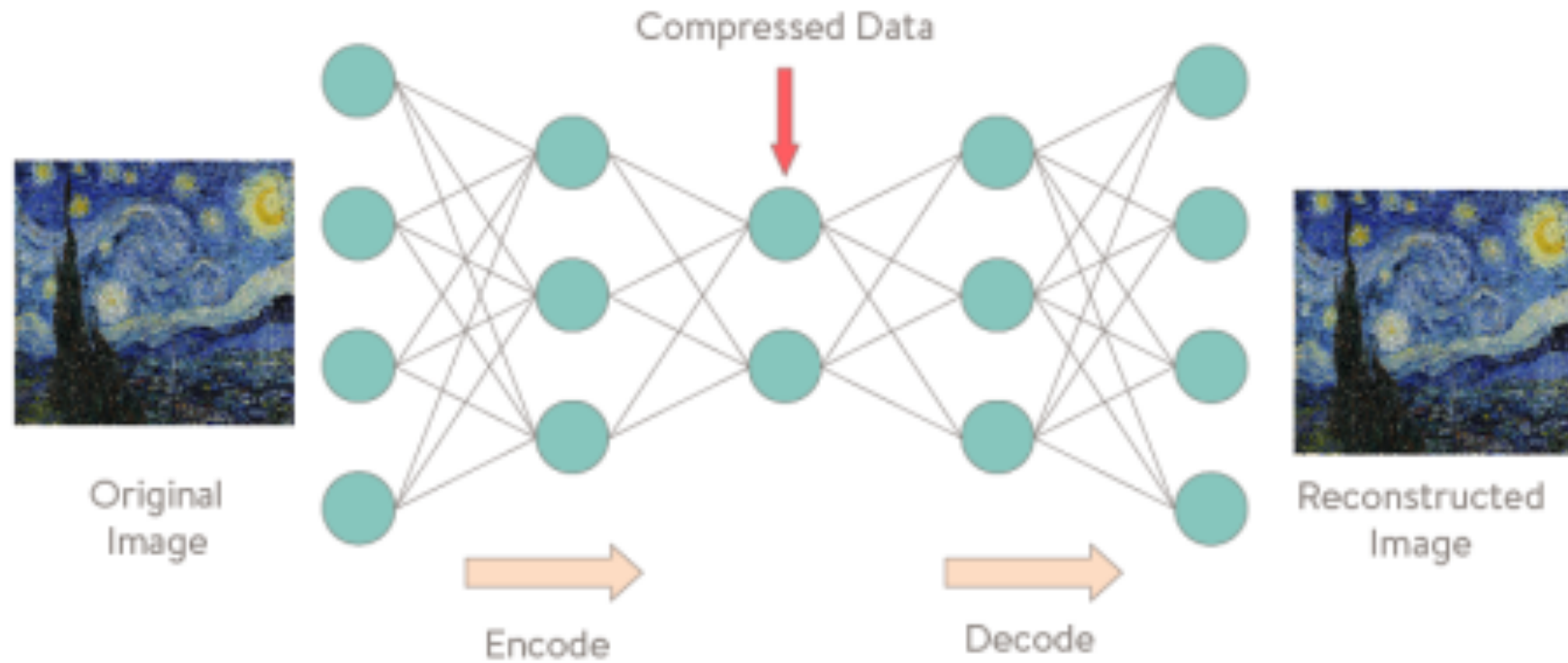


Variational autoencoder



- Olyan Helmholtz-gép, ami működik
- Generatív modell = dekóder
- Rekognitív modell = enkóder
- Tipikusan deep network mind a kettő
- ELBO közelítés a maximum likelihood helyett

Tömörítés neurális hálózattal



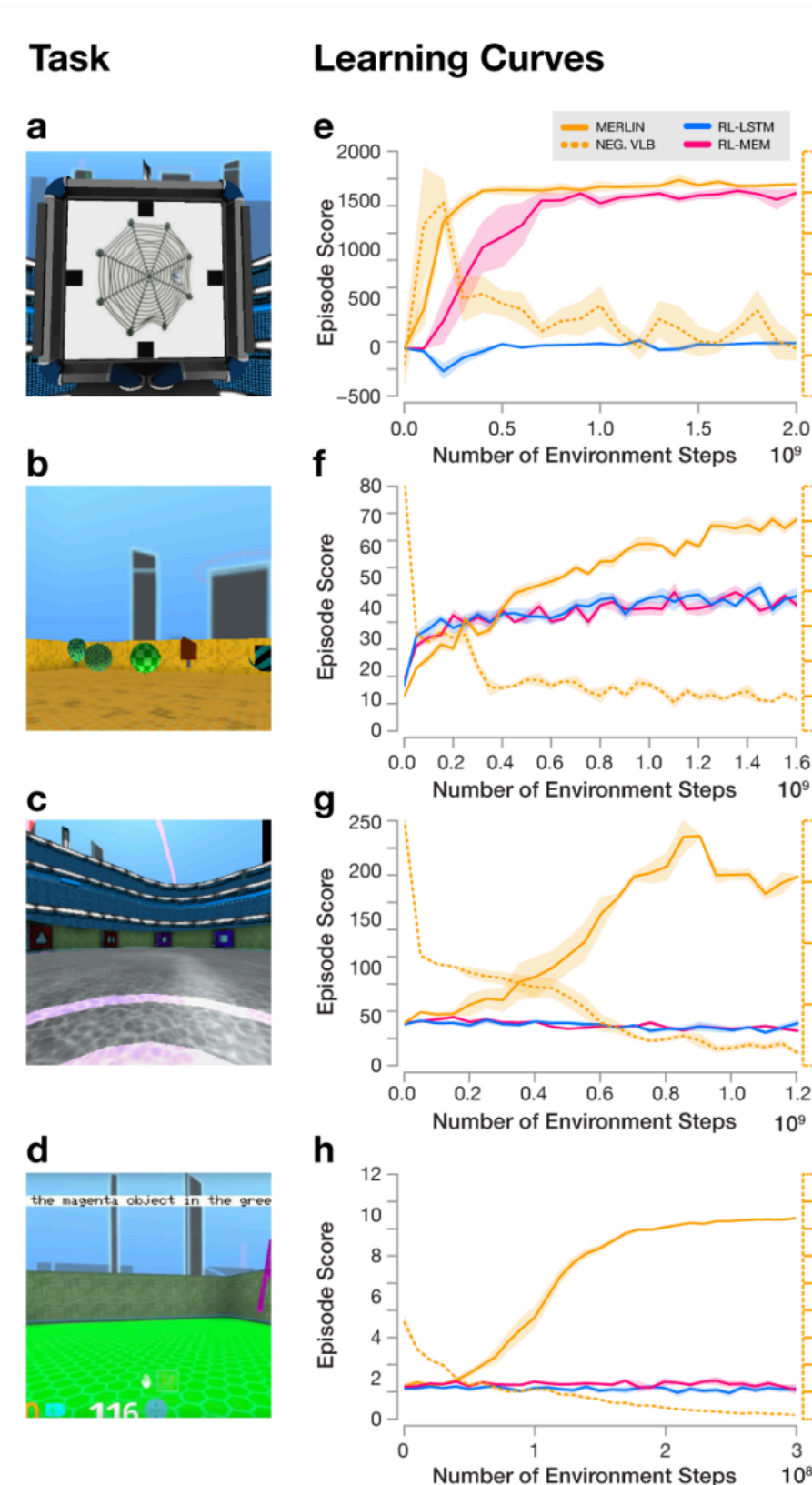
original

bicubic
(21.59dB/0.6423)

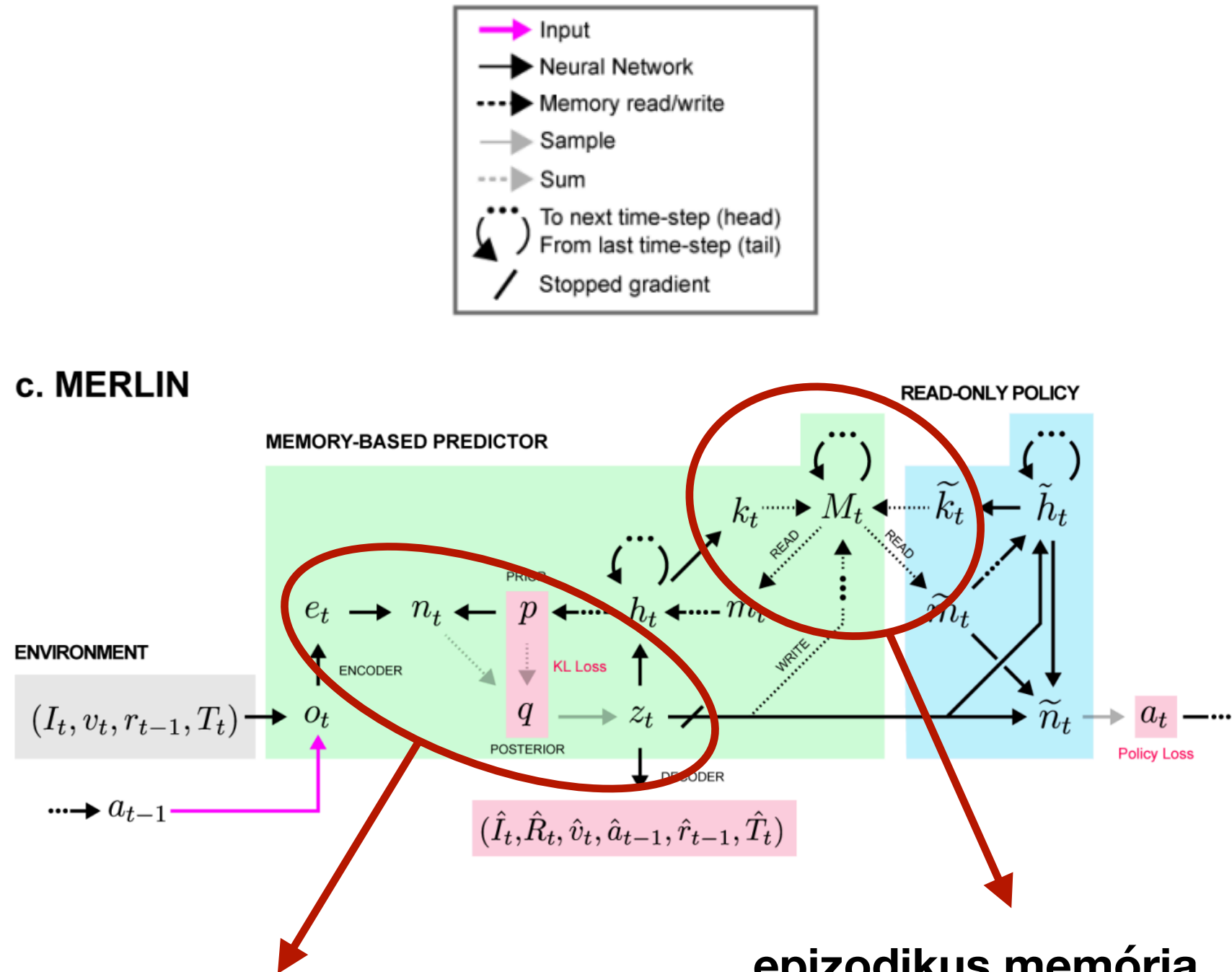
SRGAN
(20.34dB/0.6562)



Feladatok közötti általánosítás



c. MERLIN



előrecsatolt háló helyett
variational autoencoder állítja
elő a környezet állapotának
predikcióját

epizodikus memória
javítja a tárolt állapot
pontosságát

VAE az idegtudományban

- A generatív modellek sikeresek a válaszvariabilitás és kovariabilitás predikciójában
 - de a paraméterillesztésük a nagyon egyszerű eseteknél túl nehéz, a stimulus dimenzionalitás miatt többek között
- A VAE eszközt ad bonyolultabb generatív modellekkel való munkához
 - a rekognitív modell az idegrendszeri előrecsatolt alrendszer modellje lehet
- Új terület, friss eredmény pl a V1-beli receptív mezők predikciója természetes képeken tanított VAE-val (korábbi eredmények reprodukciója)

Házi feladat

- Egészítsd ki az órán használt 1-es notebookot az EM-algoritmussal történő paraméterbecsléssel
 - miután szintetikus adatokat generáltál a modellből, a paramétereit állítsd át véletlen értékekre
 - implementáld az EM-algoritmust, ahogy C. M. Bishop: Pattern Recognition and Machine Learning c. könyvének megfelelő, a tárgyhonlapról letölthető fejezetében le van írva
 - hasonlítsd össze az eredményt az adatok generálásához ténylegesen használt paraméterekkel
- Eredményedet illusztráld