first pattern : $\xi_i^1 = \mathrm{sgn}\left(\frac{1}{N} \sum_j \sum_{\mu=1}^{p} \xi_i^\mu \xi_j^\mu \xi_j^1 \right)$

$$= \mathrm{sgn}\left(\underbrace{\frac{1}{N} \sum_j \xi_i^1}_{\xi_i^1} + \underbrace{\frac{1}{N} \sum_{\mu=2}^{p} \sum \xi_i^\mu \xi_j^\mu \xi_j^1}_{} \right)$$

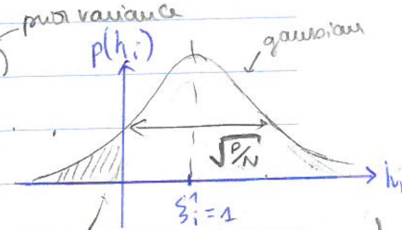
interference of competing memories.

↳ CLT $\sim \mathcal{N}(0, \alpha)$ ← prior variance

$\frac{1}{N^2} \times P \times N \times O(1)$

prefactor   nb terms

$\Rightarrow \xi_i^1 = \mathrm{sgn}(\underbrace{\xi_i^1 + \eta_i^1}_{h_i^1})$ with $\eta_i \sim \mathcal{N}(0, \frac{P}{N})$



could be a problem if P/N too big !

6/07/2017

ABOUT LEARNING : (UNSUPERVISED / SUPERVISED)

eg : STATISTICAL STRUCTURE OF NATURAL IMAGES → NETWORK CONNECTIVITY → NETWORK ACTIVITY } what are the rules of these feedback.

↳ q°: fixed points ?

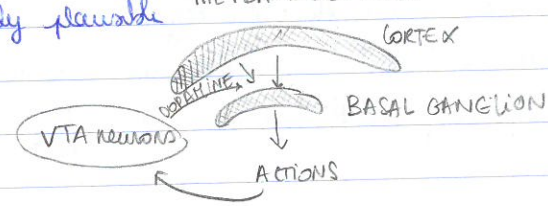
Different learning tasks : → unsupervised learning

supervised learning $\vec{x} \to \vec{y} \stackrel{?}{=} \vec{y}^*$

reinforcement learning not told $\vec{y}^*$, but tell you whether good / bad direction

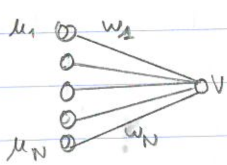
↳ more biologically plausible

THE MEANING OF LIFE :



if unexpected reward

UNSUPERVISE LEARNING : What is Hebbian learning doing ?



a single neuron $v = \vec{w} \cdot \vec{u}$ linear

Imagine streaming input pattern $\vec{u}^\alpha$ $\alpha = 1..n$

And having weights changing according to hebb rule : $\Delta \vec{w} = \lambda \vec{u}^\alpha v^\alpha$

biological assumption to the behavior of a neuron .

small lambda (slow plasticity)

↳ after n inputs : $\vec{w}(n) = \vec{w}(0) + \sum_{\alpha=1}^{n} \lambda \vec{u}^\alpha v^\alpha$ $\longrightarrow$ $\boxed{\tau \frac{d\vec{w}}{dt} = \langle \vec{u} v \rangle_{P(\vec{u})}}$

↳ plug-in $v = \vec{w}^T u$ $\longrightarrow$ $\tau \frac{dw_i}{dt} = \langle \sum_k w_k u_k u_i \rangle = \sum_k w_k \underbrace{\langle u_k u_i \rangle}_{Q_{ki}} \Rightarrow \boxed{\tau \frac{d\vec{w}}{dt} = Q\vec{w}}$

learning driven by second order correlation for linear neurons !

↳ of $P(u)$

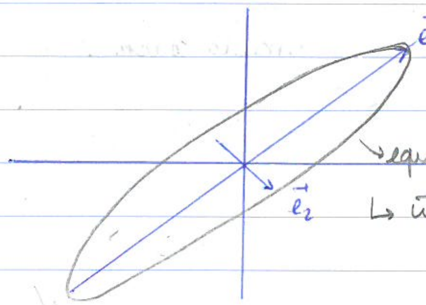
→ meaning that synapses become coupled as the inputs u are coupled.

↳ eigen decomposition of Q: $\vec{e}^{\,\mu}, \lambda^{\mu} \longrightarrow \vec{w}(t) = C^{\mu}(t)\,\vec{e}^{\,\mu}$ $\lambda_\mu \geq 0 \rightarrow$ POSITIVE SEMI DEF

↳ $\tau_w \dfrac{dC(t)^{\mu}}{dt} = \lambda_\mu C^{\mu}(t) \Rightarrow C_\mu(t) = e^{\lambda_\mu t} C_\mu(0)$ ↳ exponentional growth

neuron that fire together, wire together –

geometrically:



assume $P(u) \propto e^{-\frac{1}{2} u^T Q^{-1} u}$

↳ $\vec{w}$ will align with $\vec{e}_1$ $(\lambda_1 > \lambda_2)$ – ≡ PCA –

PCA: which directions to keep to describe your data if we want to reduce dimensionality?

↳ reconstruction error with projection $\hat{u}(v)$: $\|\vec{u} - \hat{u}(v)\|^2$ $(\hat{u}(v) = \vec{w}.\vec{v}) \Rightarrow$ min for $\hat{u} = \vec{e}_1$

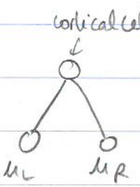
How to deal with the exponential growth? → restrict to a constrained surface → $|w| = cste$; $\|w\|^2 = cste \rightarrow$ PCA

→ projection during learning $\tau \dfrac{dw}{dt} = \underbrace{P}_{\text{projection}}(Qw)$.

modification of Hebbian rule for ℓ_1-norm: $\dfrac{dw_i}{dt} = v\left(u_i - \dfrac{1}{N}\sum_i u_i\right)$ → look at errors with the mean

ℓ_2-norm: $\dfrac{dw_i}{dt} = v u_i - \dfrac{v^2 w_i}{\sum_j w_i^2}$ homeostatic plasticity

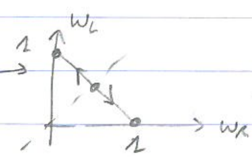
Eg: right/left eyes



$$\frac{d}{dt}\begin{bmatrix} w_L \\ w_R \end{bmatrix} = \begin{bmatrix} LL & LR \\ RL & RR \end{bmatrix}\begin{bmatrix} w_L \\ w_R \end{bmatrix} \rightarrow \vec{e}.\vec{v} \begin{bmatrix} 1 \\ 1 \end{bmatrix}\begin{bmatrix} 1 \\ -1 \end{bmatrix} \longrightarrow$$

$$\equiv \begin{bmatrix} q_0 & q_1 \\ q_1 & q_0 \end{bmatrix}$$

$q_1 < q_0$



LECTURE II HIGH DIMENSIONAL STATISTICS

what makes high dimensional statistics special? $\alpha = \dfrac{N}{P}$ → ∞ for classical data ; → $O(1)$ for modern data

take bunch of data first, ask questions later –

REGRESSION IN HIGH DIMENSIONS:

↳ planted problem of linear regression $y_\mu = x_\mu . s^0 + \varepsilon_\mu \rightarrow \hat{s}$? ← $P(\varepsilon)$, $\alpha = N/p$ measurement

↳ optimization formulation: $\hat{s} = \underset{s}{\operatorname{argmin}} \underbrace{\sum_\mu \rho(y_\mu - x_\mu . s)}_{\text{loss function}} + \underbrace{\sum_j \mathcal{C}(s_j)}_{\text{regulariser}}$ ⇒ which algorithm is the best?

cf PRX

stat phys analogy $s \equiv$ thermal dof

data $X, y \equiv$ quenched disorder

$\hat{s} \equiv$ ground state

TENSOR DECOMPOSITION : Figuring out how neural circuits learn.