Strumenti di analisi mediante le tecniche di simulazione con il “gemello digitale”

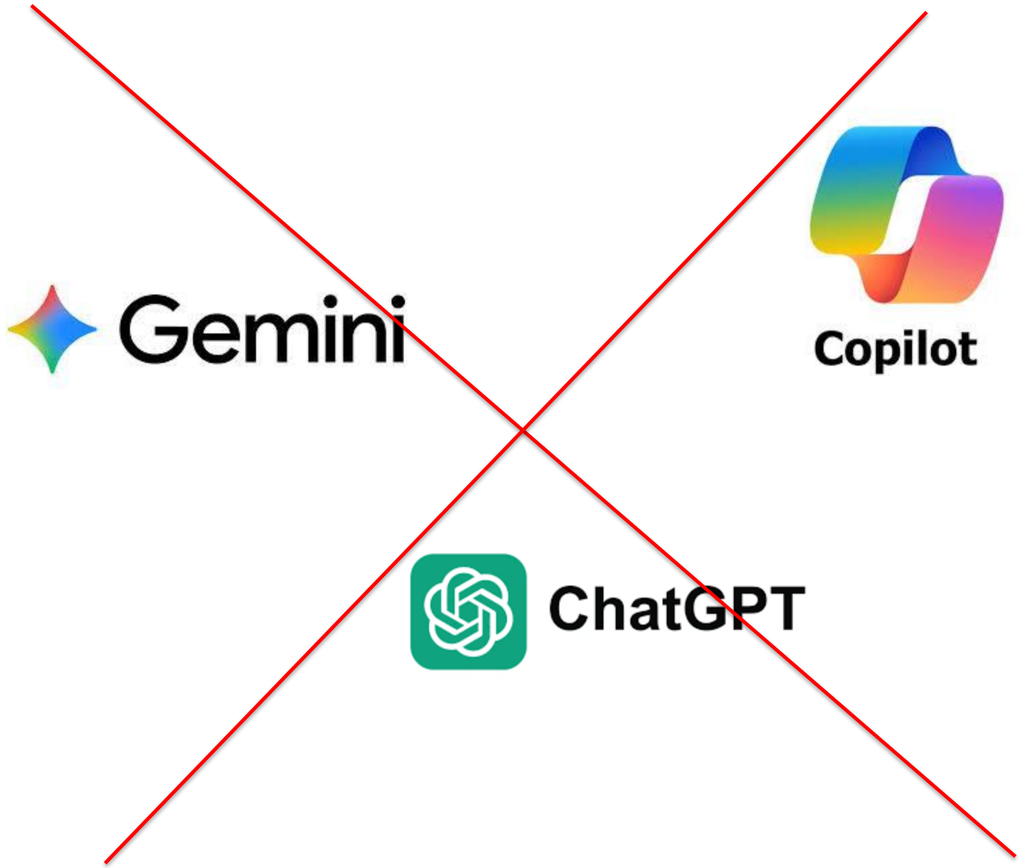
18 aprile 2026

Daniele Caligiore (CNR)

Consiglio Nazionale delle Ricerche
ISTC
CNR - Istituto di Scienze e Tecnologie della Cognizione

Copilot

ChatGPT


Gemini
Copilot
ChatGPT

Di cosa parleremo

- IA **theory-driven** e **data-driven**
- Approccio **personalizzato** e di **sistema**
- **Esempi** di modelli theory-driven

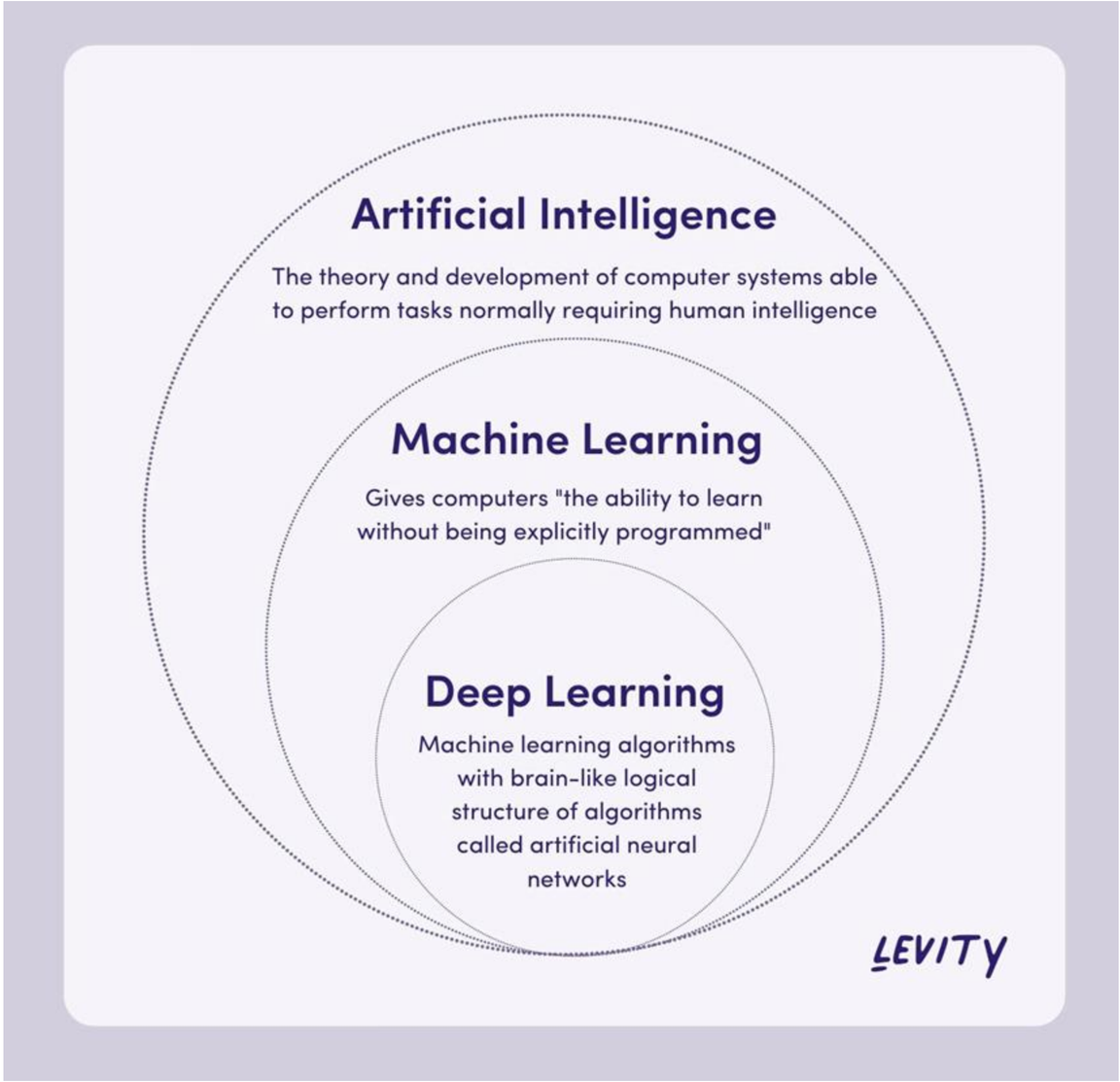
Applicazioni “storiche” IA in medicina

Come l'IA che apprende rivoluzionerà diagnosi, trattamento e prevenzione?

IA, machine learning, deep learning

IA che sta cambiando tutto: analizza le situazioni e impara dall'esperienza come risolvere autonomamente i problemi.

L'**apprendimento** non è una prerogativa di tutte le forme di IA, ci sono infatti alcuni sistemi che sono programmati dall'uomo per svolgere un determinato compito (ripetitivo).



From Wikipedia, the free encyclopedia

Frank Rosenblatt (July 11, 1928 – July 11, 1971) was an American psychologist notable in the field of artificial intelligence. He is sometimes called the father of deep learning[1] for his pioneering work on artificial neural networks.

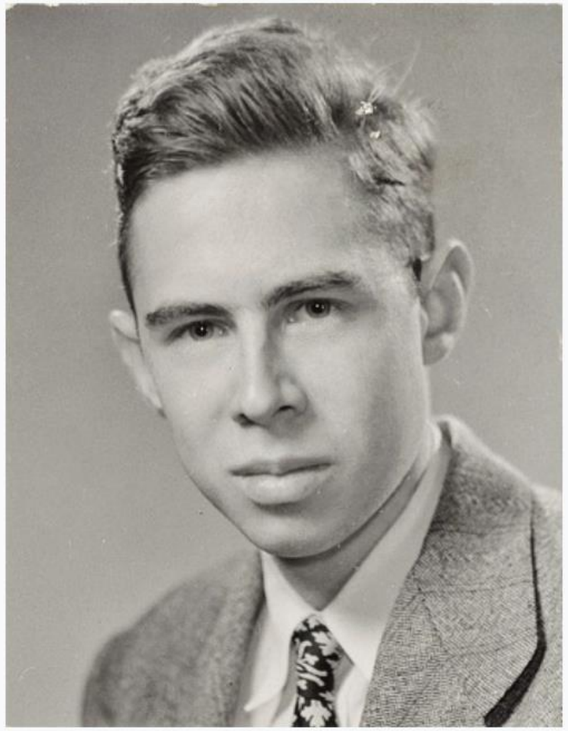
Life and career [edit]

Rosenblatt was born into a Jewish family in New Rochelle, New York as the son of Dr. Frank and Katherine Rosenblatt.[2]

After graduating from The Bronx High School of Science in 1946, he attended Cornell University, where he obtained his A.B. in 1950 and his Ph.D. in 1956.[3]

For his PhD thesis he built a custom-made computer, the Electronic Profile Analyzing Computer (EPAC), to perform multidimensional analysis for psychometrics. He used it between 1951 and 1953 to analyze psychometric data collected for his PhD thesis. The data were collected from a paid, 600 item survey of more than 200 Cornell undergraduates. The total computational cost was 2.5 million arithmetic operations, necessitating the use of an IBM CPC as well.[4] It was

Frank Rosenblatt



Rosenblatt in 1950

Born	Frank Rosenblatt July 11, 1928 New Rochelle, New York, U.S.

Psychological Review
Vol. 65, No. 6, 1958

THE PERCEPTRON: A PROBABILISTIC MODEL FOR INFORMATION STORAGE AND ORGANIZATION IN THE BRAIN[1]

F. ROSENBLATT

Cornell Aeronautical Laboratory

If we are eventually to understand the capability of higher organisms for perceptual recognition, generalization, recall, and thinking, we must first have answers to three fundamental questions:

1. How is information about the physical world sensed, or detected, by the biological system?
2. In what form is information stored, or remembered?
3. How does information contained in storage, or in memory, influence recognition and behavior?

and the stored pattern. According to this hypothesis, if one understood the code or "wiring diagram" of the nervous system, one should, in principle, be able to discover exactly what an organism remembers by reconstructing the original sensory patterns from the "memory traces" which they have left, much as we might develop a photographic negative, or translate the pattern of electrical charges in the "memory" of a digital computer. This hypothesis is appealing in its simplicity and ready intelligibility, and a large family of theoretical brain

nature

Explore content | About the journal | Publish with us | Subscribe

nature > letters > article

Letter | Published: 09 October 1986

Learning representations by back-propagating errors

David E. Rumelhart, Geoffrey E. Hinton & Ronald J. Williams

Voce Discussione Leggi Modifica Modifica wikitesto Cronologia Strumenti

David Rumelhart (Wessington Springs, 12 giugno 1942 – Chelsea, 13 marzo 2011[1]) è stato uno psicologo statunitense.

Durante la sua carriera ha dato un importante contributo all'analisi della cognizione umana[2], lavorando principalmente nell'ambito della psicologia matematica, dell'intelligenza artificiale simbolica e del connessionismo.

Psicologia matematica (la pagina non esiste)

Geoffrey Hinton

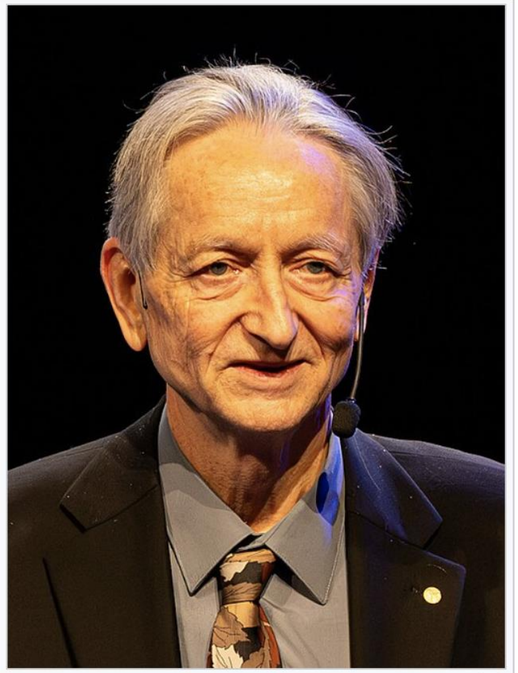
58 lingue

Voce | Discussione | Leggi | Modifica | Modifica wikitesto | Cronologia | Strumenti

Geoffrey Everest Hinton (Wimbledon, 6 dicembre 1947) è uno psicologo e informatico britannico naturalizzato canadese.

Noto per i suoi contributi allo sviluppo dell'apprendimento automatico, è considerato uno tra i ricercatori più influenti nello sviluppo dell'apprendimento profondo.[1][2][3][4][5] Professore all'Università di Toronto e ricercatore presso Google Brain, nel 2017 fondò il Vector Institute for Artificial Intelligence a Toronto, di cui è consulente scientifico principale *pro bono*.[6][7]

Nel 1986 ha pubblicato con David Rumelhart e Ronald J. Williams un articolo di particolare influenza, che popolarizzò il già noto[8] uso dell'algoritmo di retropropagazione dell'errore



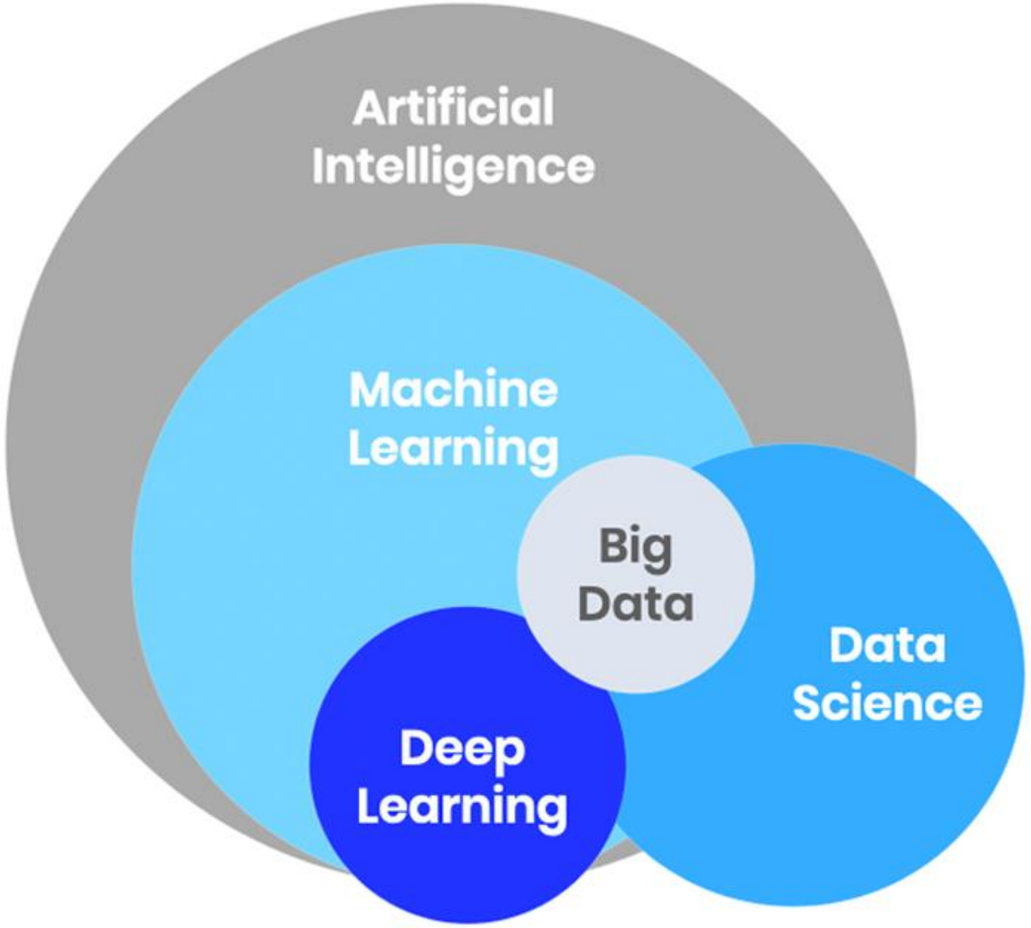
Geoffrey Hinton nel 2024

Premio Nobel per la fisica 2024

Due approcci

Data driven

L'importanza dei dati



Cosa può fare un modello data-driven?

Classificazione

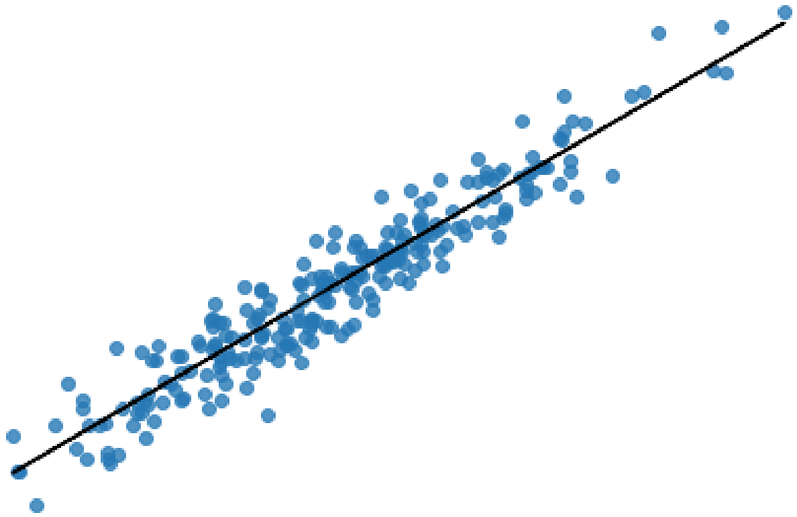
- Suddivide oggetti in categorie basandosi su caratteristiche specifiche e etichette assegnate.
- **Psicologia Clinica.** Classificare pazienti con "depressione" o "ansia" basandosi su sintomi e test psicologici.

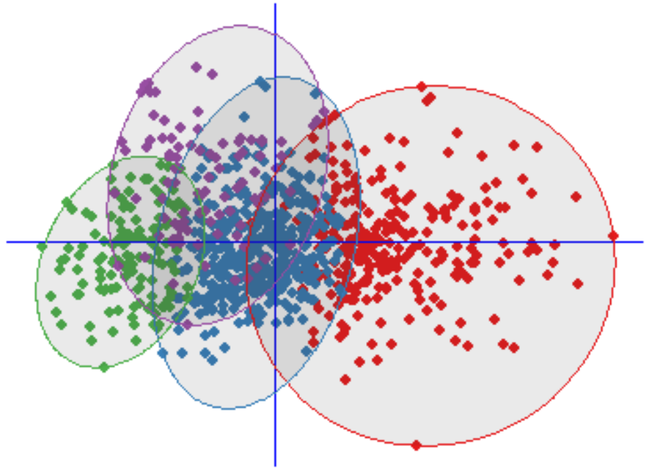
Predizione (Regressione)

- Prevede un valore numerico basandosi su dati passati
- **Psicologia Clinica.** Prevedere il tempo di recupero da un trauma psicologico basandosi su dati storici di altri pazienti.

Raggruppamento (Clustering)

- Raggruppa oggetti in base a somiglianze, senza etichette specifiche.
- **Psicologia Clinica.** Identificare gruppi di pazienti con sintomi simili per studiare approcci terapeutici comuni.





Come si realizza un modello data-driven ML

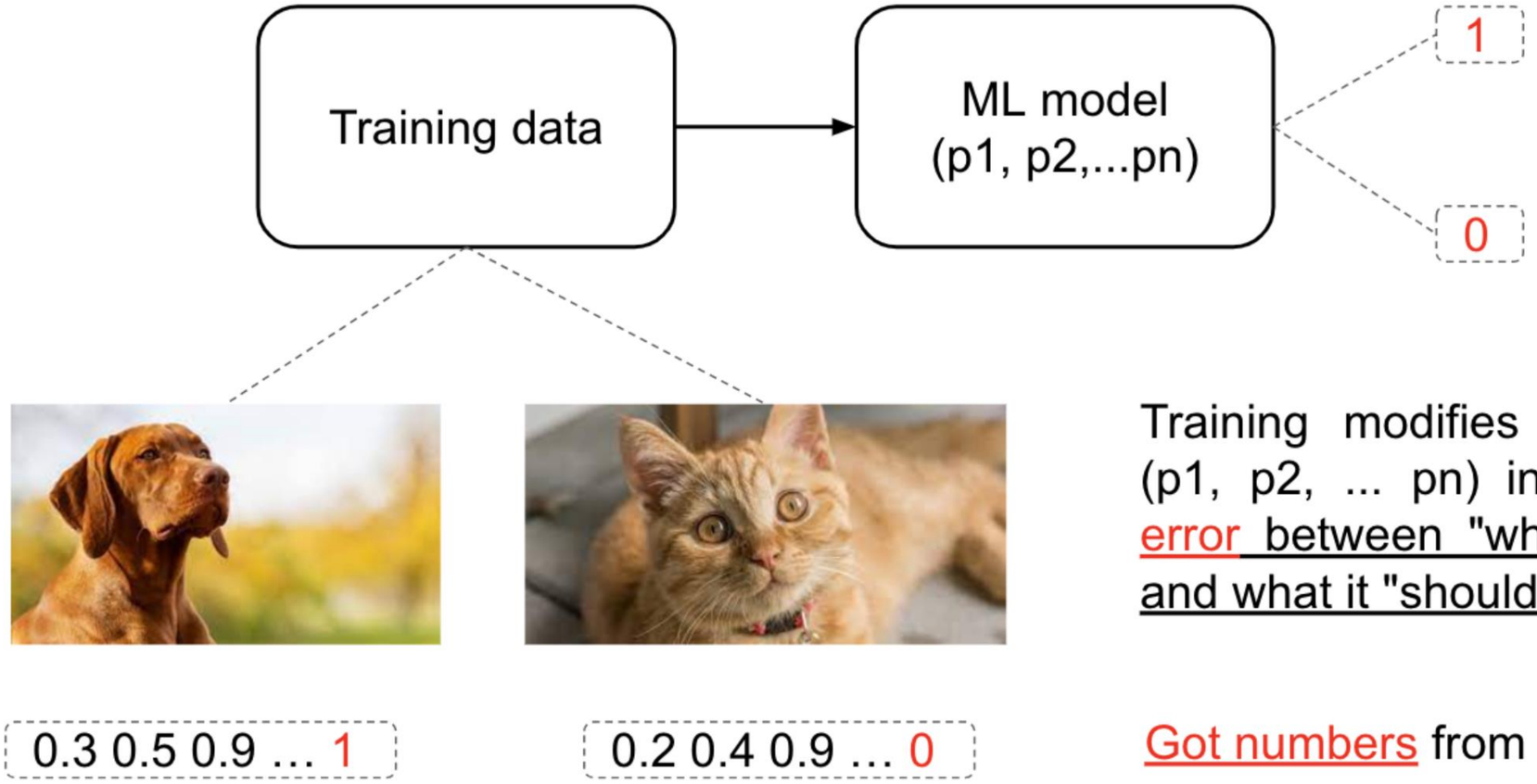
Fase 1: Raccolta e processamento dati – Selezione, pulizia e preparazione dei dati.

Fase 2: Scelta modello ML – Identificazione dell'algoritmo di machine learning più adatto.

Fase 3: Addestramento modello ML – Il modello apprende dai dati raccolti.

Fase 4: Ottimizzazione e test del modello ML – Miglioramento performance e valutazione finale.

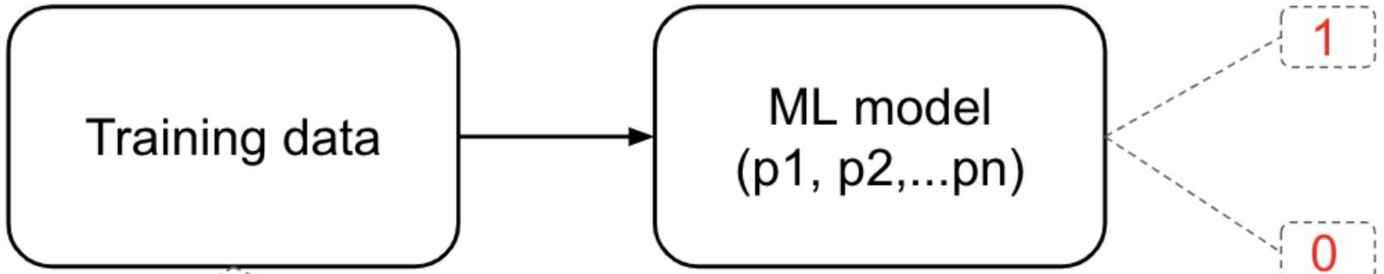
Example: Supervised Learning (Classification)



Training modifies the model parameters (p1, p2, ... pn) in order to minimize the error between "what the ML model says" and what it "should have said".

Got numbers from any type of data!

Example: Supervised Learning (Classification)



0.1 0.3 0.7 … 1

0.1 0.3 0.8 … 0

Training modifies the model parameters (p1, p2, ... pn) in order to minimize the error between "what the ML model says" and what it "should have said".

Got numbers from any type of data!

Esempio IA in psicologia clinica

ID_Paziente	Età	Genere	Ansia	Energia	Sonno	Isolamento	Storia_Depressione	Supporto	Diagnosi_Depressione
1	34	0	7	3	5	6	0	1	1
2	28	1	5	6	8	4	0	1	0
3	45	0	8	4	6	7	1	0	1
4	50	1	3	7	9	3	0	1	0
5	23	0	6	2	4	8	0	0	1
6	30	1	4	7	8	5	0	1	0
7	38	1	9	3	4	7	1	0	1

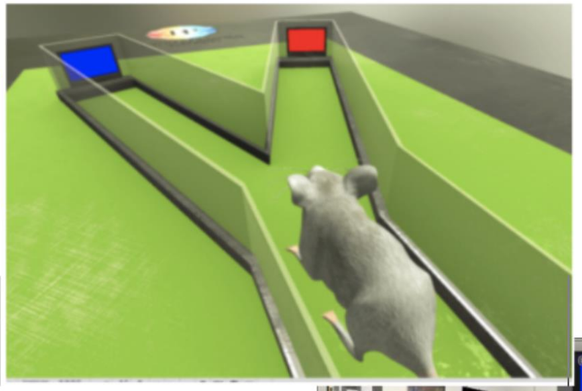
Un modello di Machine Learning che apprende in modo supervisionato a **prevedere** la **probabilità di depressione** basandosi sui dati clinici e psicologici del paziente.

Esempio IA in psicologia clinica

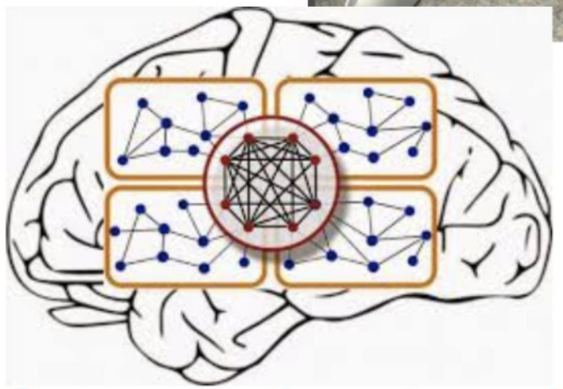
ID_Paziente	Età	Genere	Ansia	Energia	Sonno	Isolamento	Storia_Depressione	Supporto	Diagnosi_Depressione
1	34	0	7	3	5	6	0	1	1
2	28	1	5	6	8	4	0	1	0
3	45	0	8	4	6	7	1	0	1
4	50	1	3	7	9	3	0	1	0
5	23	0	6	2	4	8	0	0	1
6	30	1	4	7	8	5	0	1	0
7	38	1	9	3	4	7	1	0	1

Un modello di Machine Learning che apprende in modo supervisionato a **prevedere** la **probabilità di depressione** basandosi sui dati clinici e psicologici del paziente.

Due approcci

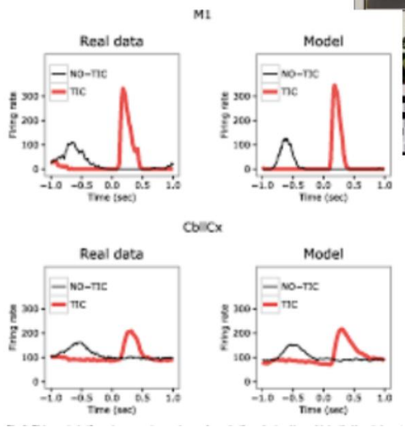


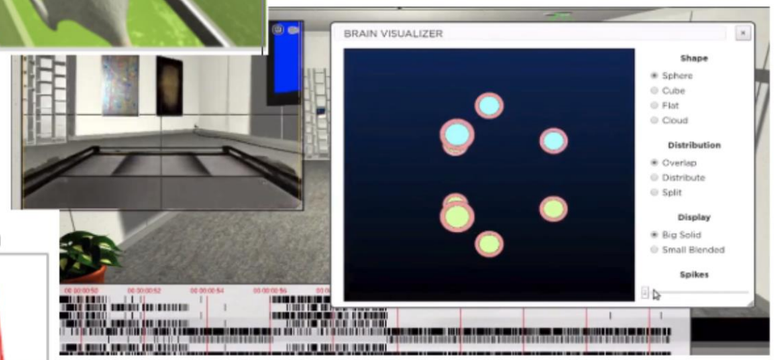
Neurorobotics



Computational model

Reproduction





Explain mechanisms

Predictions

Theory driven (digital twin)

Modelli computazionali in generale

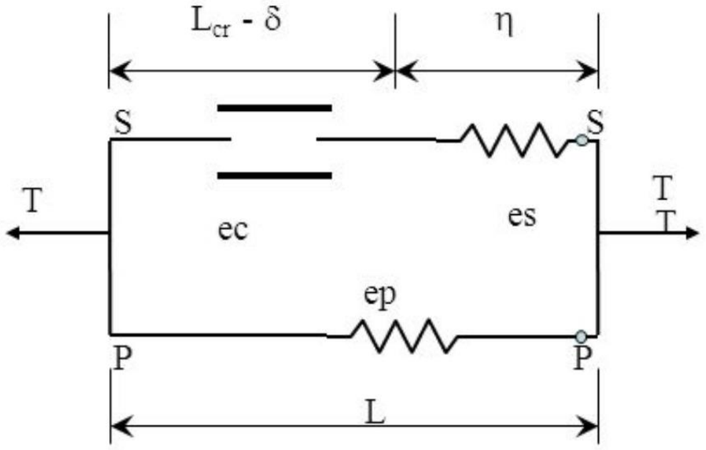
Modello classico del muscolo: equazione di Hill.

L'approccio classico è basato, come già detto su un modello funzionale "a tre elementi" . Quando il muscolo **non è stimolato**, **l'elemento contrattile si può estendere liberamente senza sviluppare forza e le caratteristiche del muscolo sono rappresentate dall'elemento elastico in parallelo; quando il muscolo è stimolato l'elemento contrattile si accorcia con una velocità che è funzione della lunghezza attuale del muscolo e del carico. L'elemento elastico serie è stato introdotto per tener conto del comportamento osservato quando intervengano brusche variazioni di carico o di lunghezza.** L'equazione costitutiva di tale modello si ottiene considerando che la lunghezza totale del muscolo L è uguale alla somma delle lunghezze dell'elemento contrattile e dell'elemento serie.

$$L = LCR - \delta + \eta$$

dove *LCR* è la lunghezza dell'elemento contrattile a riposo, δ è il suo accorciamento ed η è la lunghezza dell'elemento serie. Derivando rispetto al tempo si ottiene:

$$dL/dt = - d\delta/dt + d\eta/dt$$



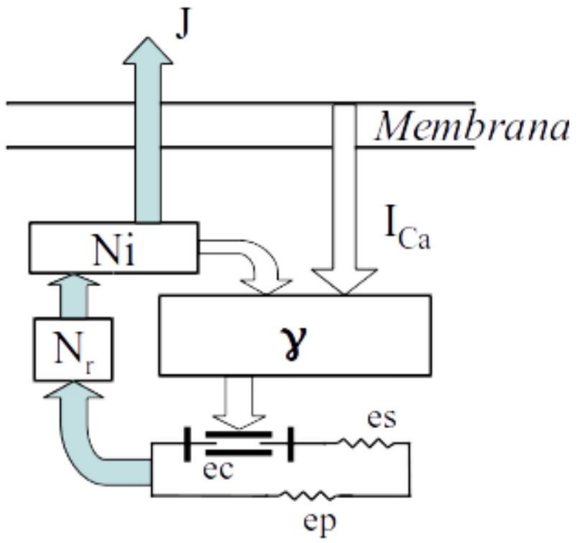
Modello strutturale della contrazione muscolare. *Modello di Huxley.*

Meccanismo di attivazione. Il modello dei meccanismi di attivazione è un modello compartimentale costituito dai tre compartimenti riprodotti in Figura dove vengono rappresentati:

Un deposito interno di ioni Ca++, a concentrazione Ni. Tale compartimento, durante la depolarizzazione rilascia ioni Ca++ in quantità dipendente dal valore del potenziale di membrana al di sopra di una soglia di polarizzazione. Durante la fase di ripolarizzazione, in cui il potenziale della membrana cellulare ritorna al valore di riposo, il compartimento assume calcio fino al livello di riposo Ni0. Superato tale livello il calcio in eccesso viene espulso attivamente dalla cellula.

Il compartimento di attivazione, indicato con la variabile **γ** che controlla la velocità di formazione dei ponti tra i filamenti di actina e miosina. Durante la depolarizzazione esso riceve ioni Ca++ dal compartimento Ni e dall'esterno della cellula attraverso un flusso ICa.

Un compartimento Nr addetto alla rimozione del calcio dalla fibra muscolare e al suo trasferimento al deposito Ni. Durante la fase di depolarizzazione tale rimozione avviene a una velocità estremamente ridotta, per cui questo ingresso al compartimento Ni può in pratica essere trascurato.



Il progresso tecnologico nel modello di Solow

L'efficienza del lavoro

Possiamo esprimere tutte le variabili per unità di lavoro effettivo:

Reddito: $y = Y/LE = F(K/LE,1) = f(k)$

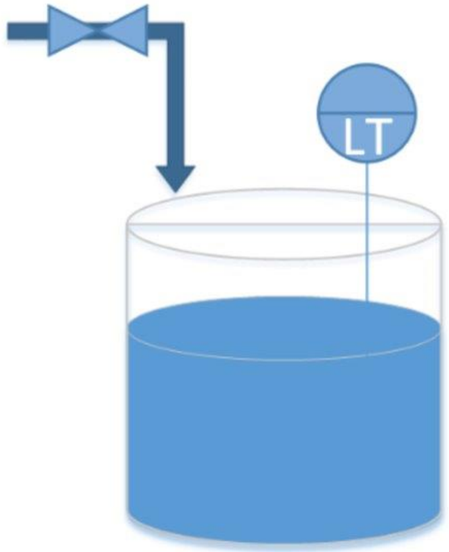
Capitale: $k = K/LE$

Risparmio, investimenti: $s\,y = s\,f(k)$

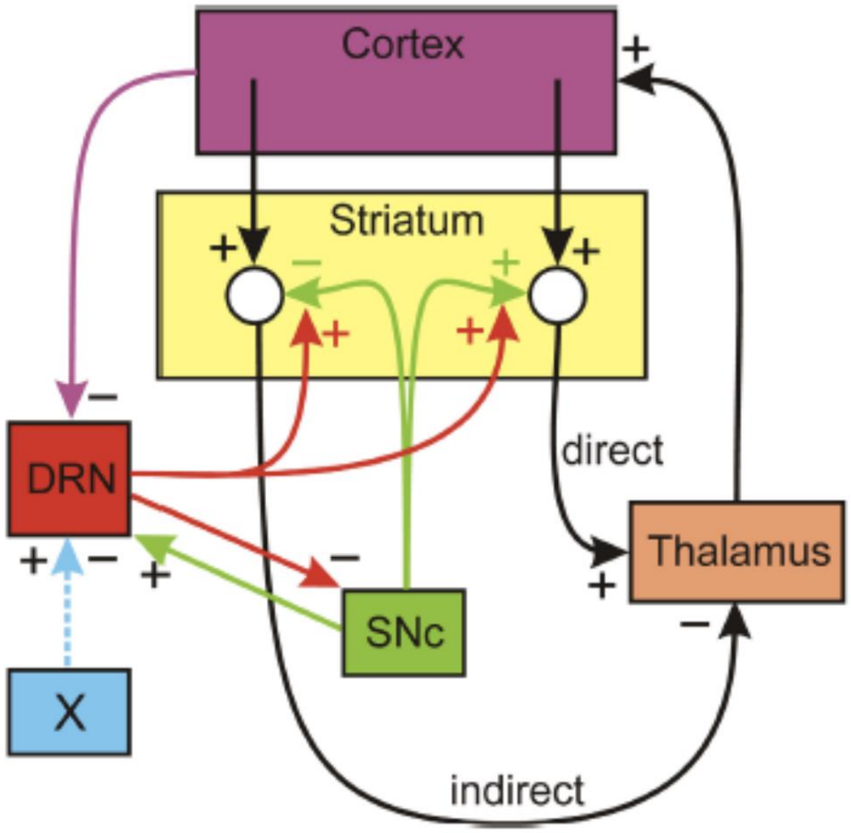
Population growth

$$\frac{dn}{dt} = born \cdot n(t) - dead \cdot n(t)$$

Another example … filling a water tank

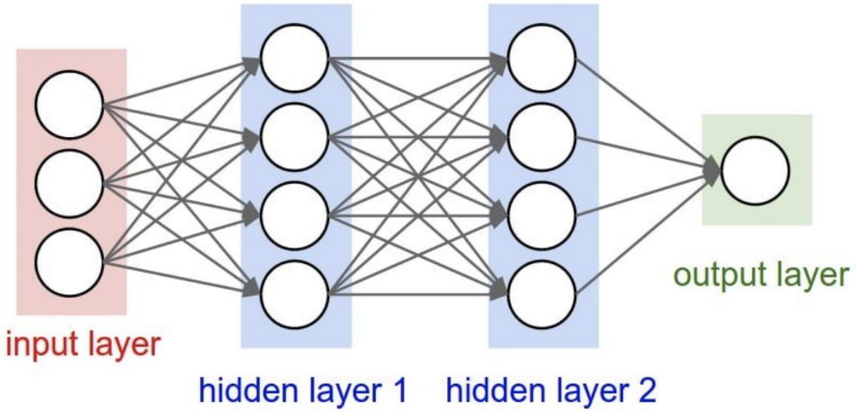


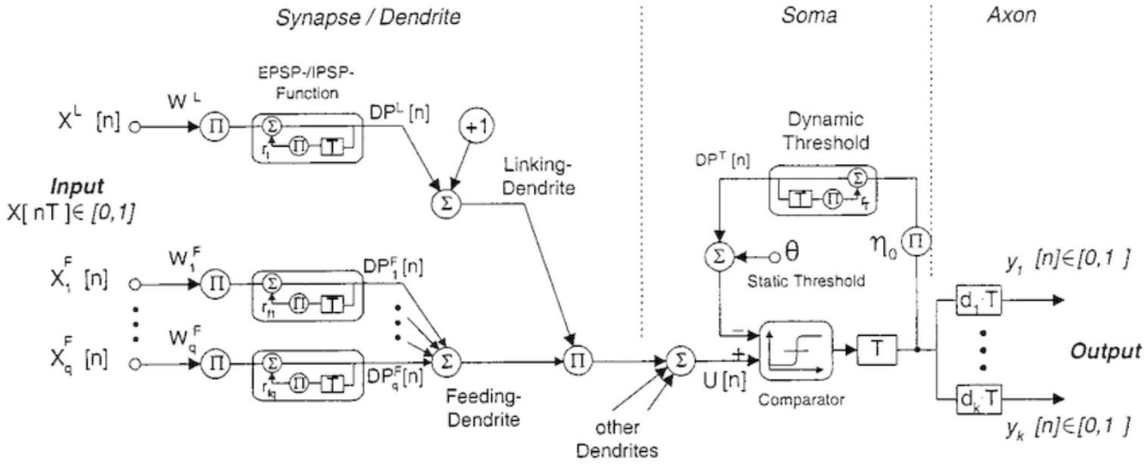
$$\frac{dm}{dt} = \dot{m}_{in} - \dot{m}_{out}$$

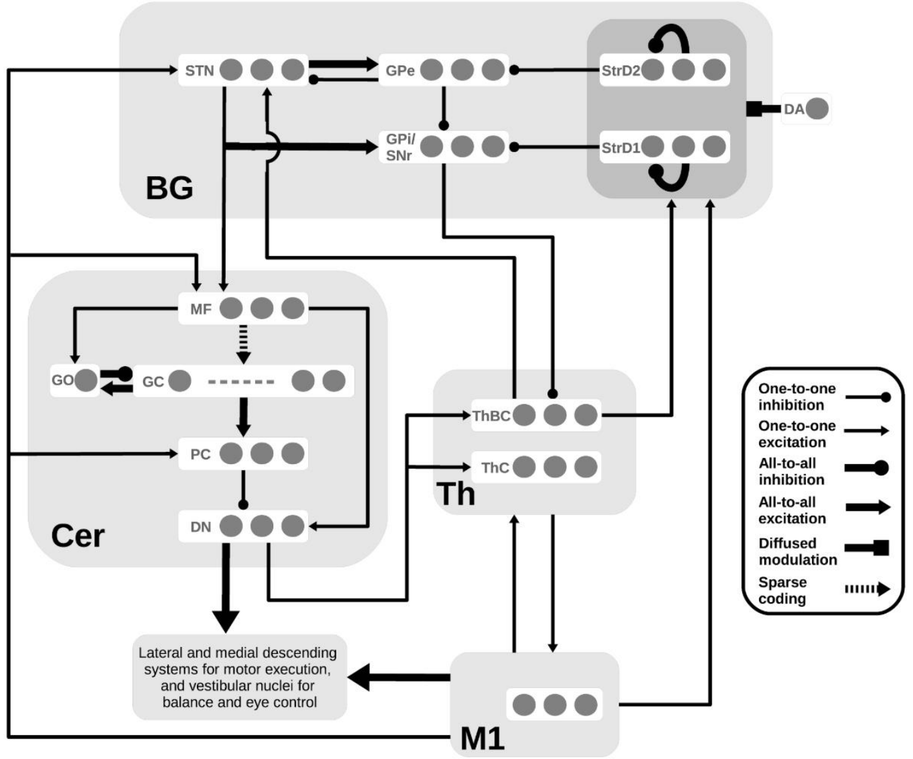


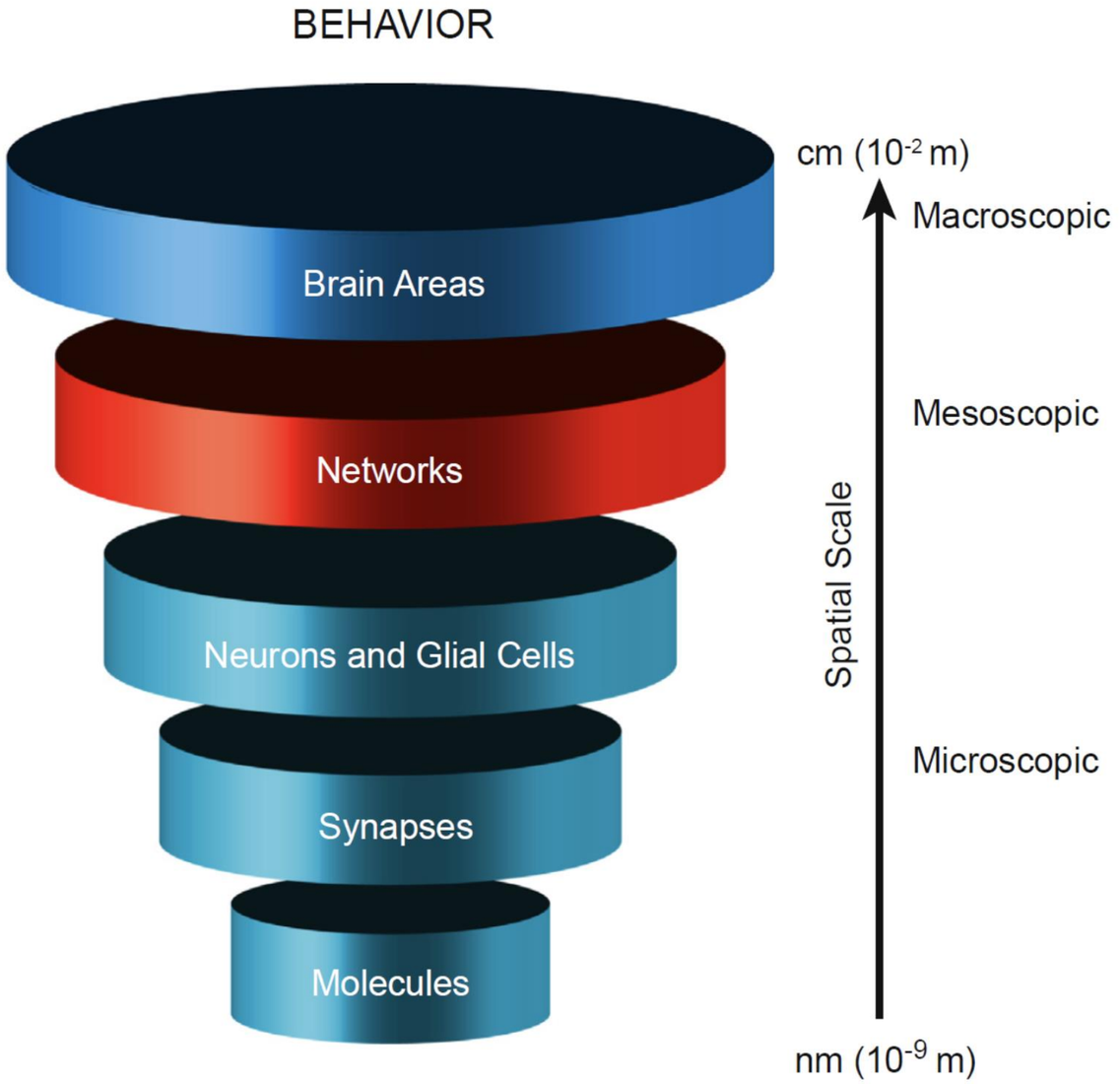
$$\frac{dMI}{dt} = a_{1c} - a_{1da} \cdot DA - d_1 \cdot MI$$
$$\frac{dMD}{dt} = a_{2c} + a_{2da} \cdot DA - d_2 \cdot MD$$
$$\frac{dTH}{dt} = a_3 + a_{3md} \cdot MD - a_{3mi} \cdot MI - d_3 \cdot TH$$
$$\frac{dCX}{dt} = a_{4th} \cdot TH - d_4 \cdot CX$$
$$\frac{dDRN}{dt} = a_5 - a_{5cx} \cdot CX + a_{5sn} \cdot SN - d_5 \cdot DRN$$
$$\frac{dDA}{dt} = G \cdot 5HT \cdot SN - d_6 \cdot DA$$
$$\frac{d5HT}{dt} = a_7 \cdot DRN - d_7 \cdot 5HT$$
$$\frac{dSN}{dt} = a_8 - a_{8drn} \cdot DRN - d_8 \cdot SN$$

Modelli a diversi livelli di astrazione





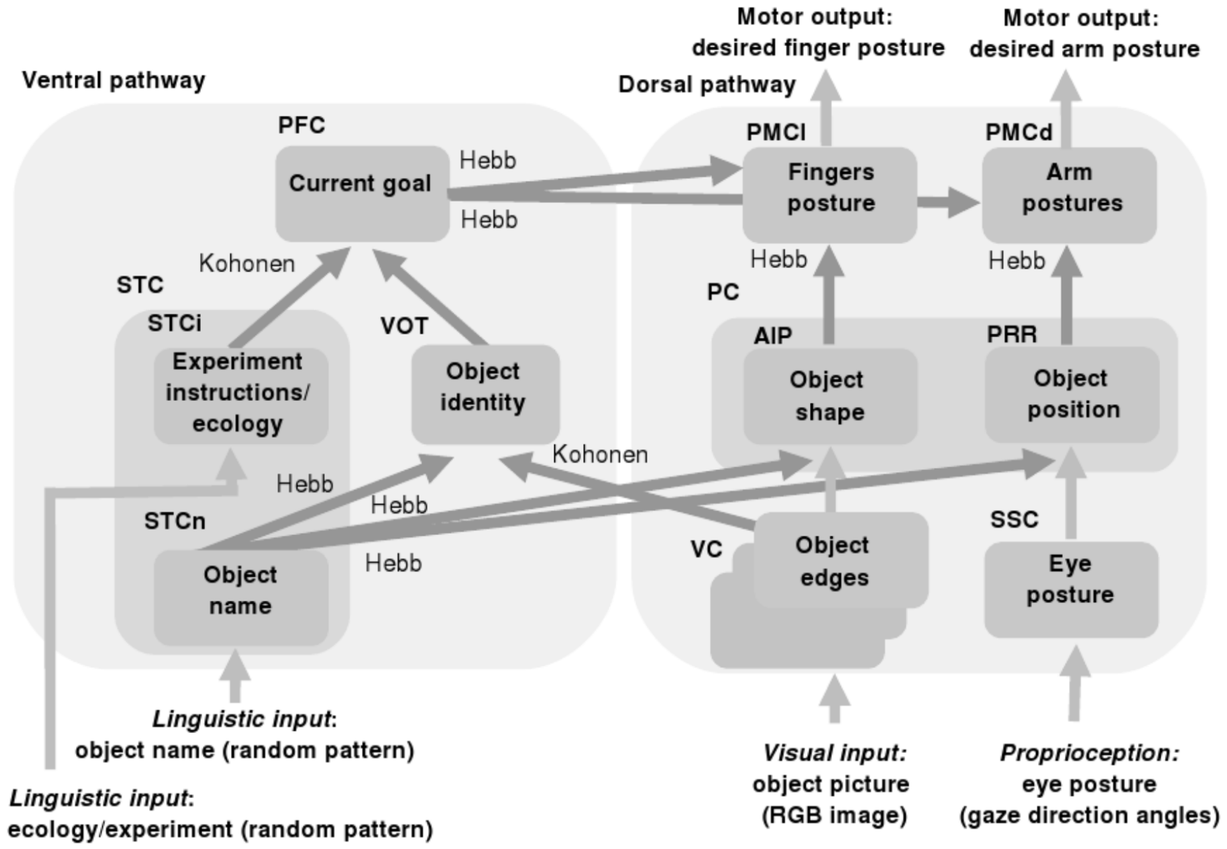




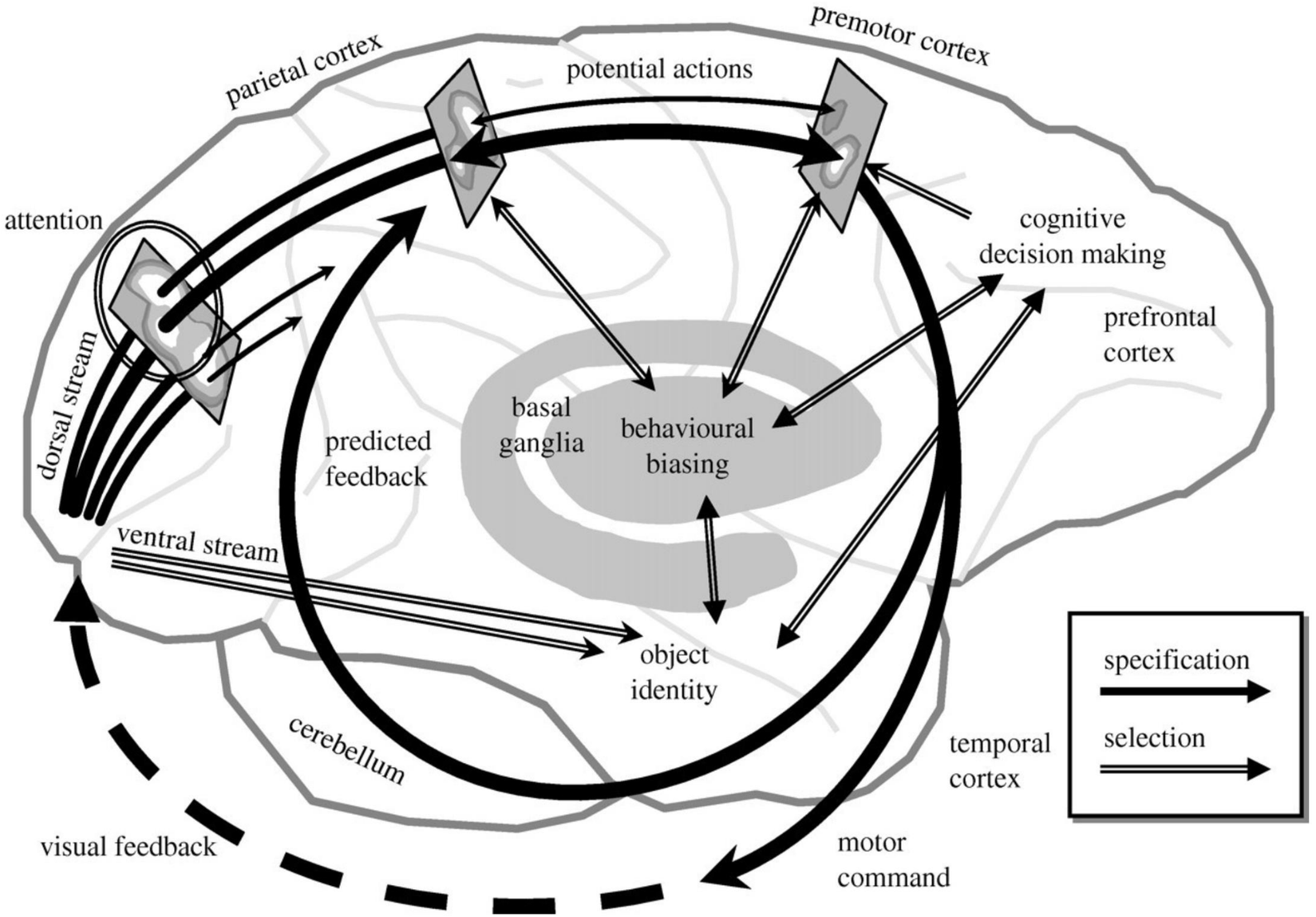
from Fröhlich F (2016). *Network Neuroscience*

	Brain	Body	Behaviour	Learning	Time
Probabilistic	**	*	*****	**	**
Connectionist	**	***	*****	*****	*
Leaky integrator	***	**	**	***	****
Mean field	***	*	**	**	**
Spiking neurons	****	*	*	**	*****
Single neuron	***	-	-	-	***

Architettura del modello TRoPICALS



Caligiore et al. (2010). *Psychological Review*


parietal cortex
premotor cortex
potential actions
attention
cognitive
decision making
prefrontal
cortex
dorsal stream
basal
ganglia
behavioural
biasing
predicted
feedback
ventral stream
object
identity
cerebellum
temporal
cortex
specification
selection
visual feedback
motor
command

Equazioni dinamica di popolazione

$$s_j[t] = \overbrace{\sum_{i=1}^{N} w_{(PMC\rightarrow PMC)ji} a_{PMCi}[t]}^{Inner} + \overbrace{\sum_{i=1}^{N} w_{(PC\rightarrow PMC)ji} a_{PCi}[t]}^{Dorsal\ stream} + \overbrace{\sum_{i=1}^{N} w_{(PFC\rightarrow PMC)ji} a_{PFCi}[t]}^{Ventral\ stream}$$

$$u_j[t + \Delta t] = \left(1 - \frac{\Delta t}{\tau}\right) u_j[t] + \frac{\Delta t}{\tau} s_j[t] \qquad a_j[t + \Delta t] = [\tanh[u_j[t + \Delta t]]]^+$$

$$\Delta w_{ji} = \mu \cdot \exp\left[-\frac{\|v_{j^*} - v_j\|^2}{2\sigma^2}\right] \cdot (a_i - w_{ji})$$

Equazioni apprendimento

$$\Delta w_{ji} = \eta(a_j - \bar{a}_j)(a_i - \bar{a}_i)(w_{max} - |w_{ji}|)$$

$$\Delta w_{ji} = \mu \cdot \exp\left[-\frac{\|v_{j^*} - v_j\|^2}{2\sigma^2}\right] \cdot (a_i - w_{ji})$$

frontiers in Neuroscience

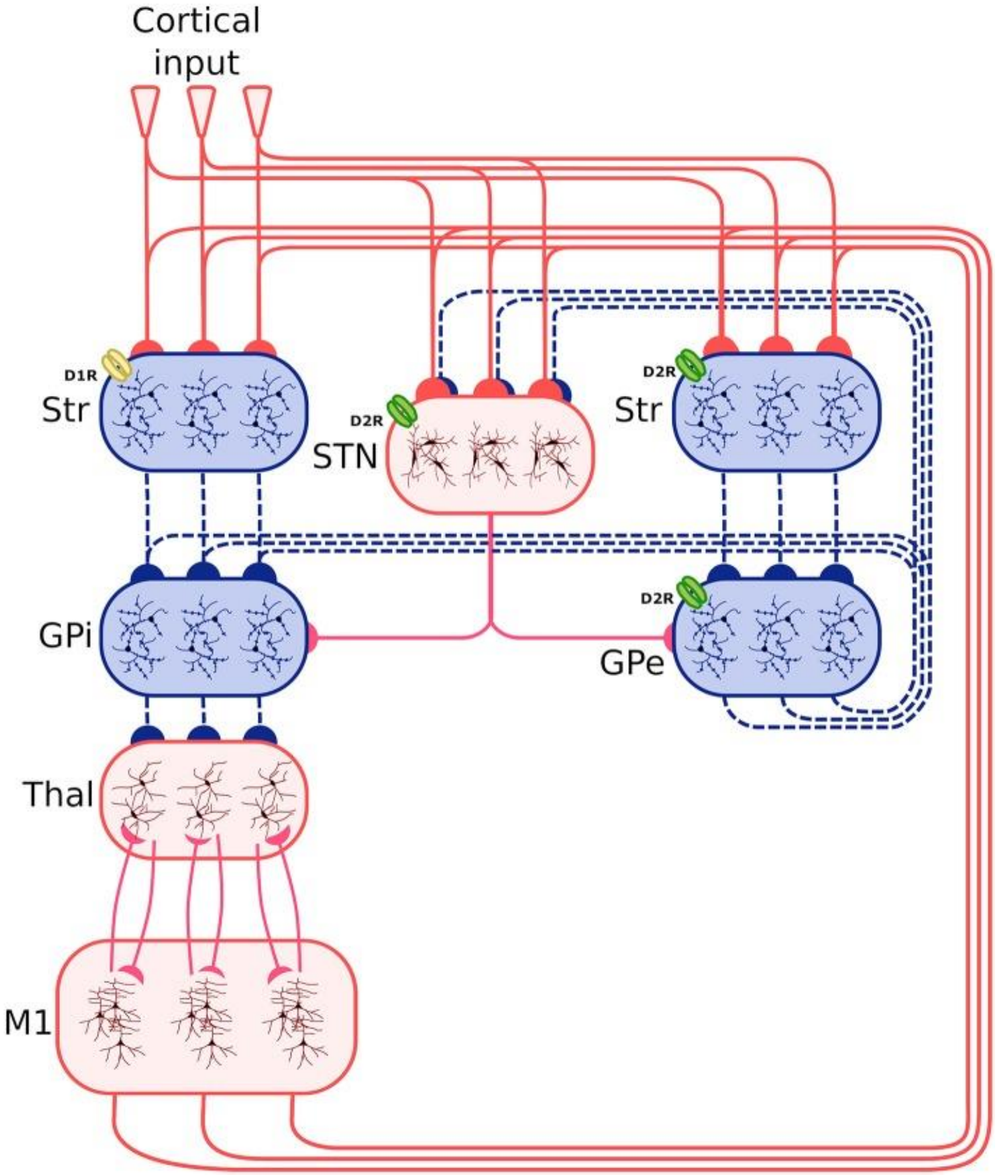
ORIGINAL RESEARCH
published: 29 May 2019
doi: 10.3389/fnins.2019.00550

Check for updates

Different Dopaminergic Dysfunctions Underlying Parkinsonian Akinesia and Tremor

Daniele Caligiore, Francesco Mannella and Gianluca Baldassarre*

National Research Council, Institute of Cognitive Sciences and Technologies, Rome, Italy


Cortical
input
D1R
Str
D2R
STN
D2R
Str
GPi
D2R
GPe
Thal
M1

Spiking neurons (dinamica di un neurone)

Equazioni in genere più complicate che richiedono tempi di calcolo maggiori

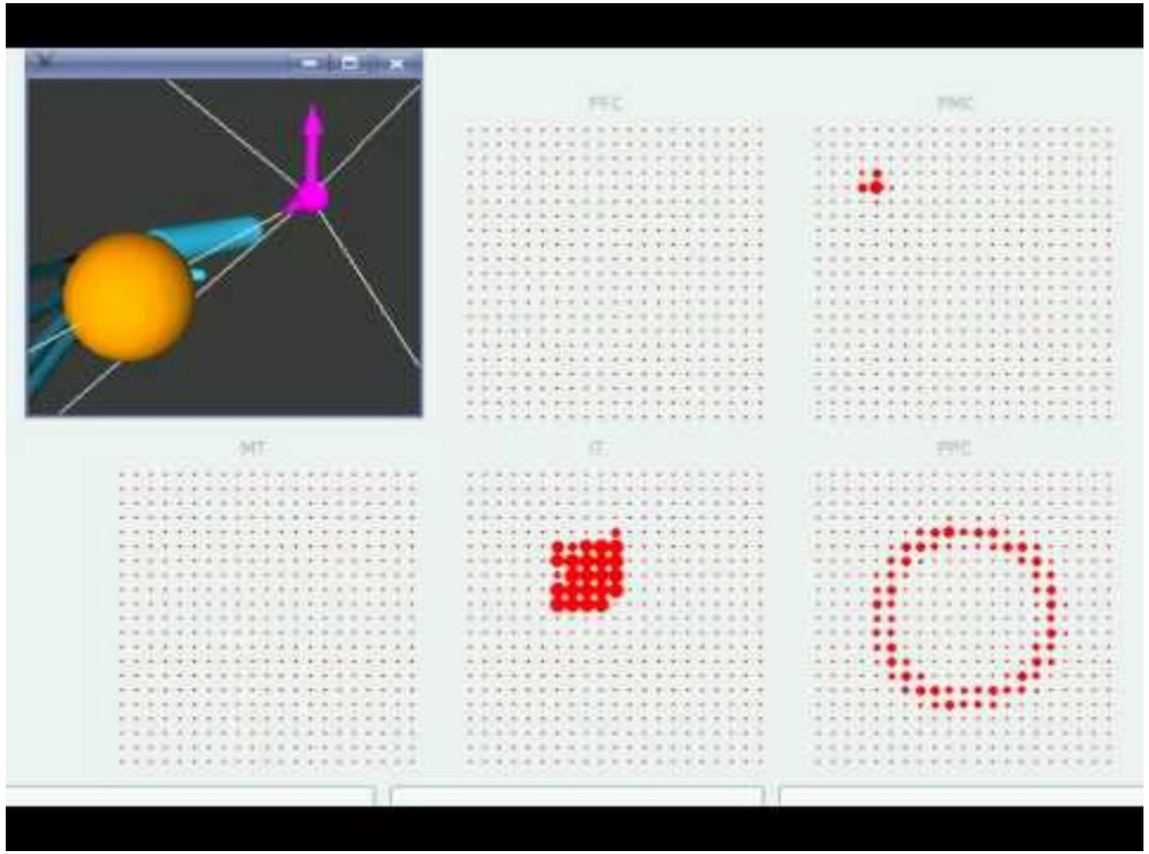
$$\tau^{\mathrm{m}} \frac{dV^{\mathrm{m}}}{dt} = -V^{\mathrm{m}}(t) + RI(t)$$

$$I^{\mathrm{s}}(t) = \sum_{j \in \Lambda} w_j \hat{I}_j^{\mathrm{s}} \sum_{t^{\mathrm{f}} \in \Upsilon_j} \exp[-(t - (t^{\mathrm{f}} + t_j^{\mathrm{d}}))/\tau^{\mathrm{s}}] H(t - (t^{\mathrm{f}} + t_j^{\mathrm{d}}))$$

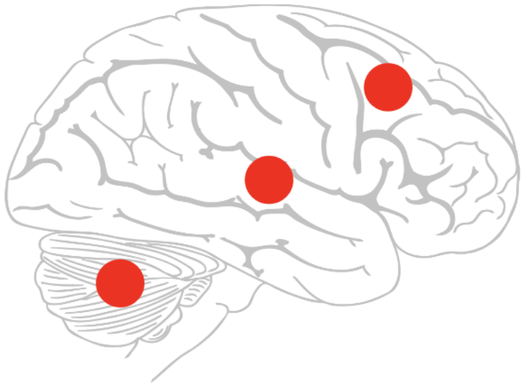
Spiking neurons

Parameter	Source
$\langle R \rangle = 42\ M\Omega$	Jiang and North (1991); Flores-Hernandez et al. (1997)
$\langle \tau^m \rangle = 25$ ms	Bennott and Wilson (2000)
$\theta = 30$ mV	
$I^{down} = -0.25$ nA	
$\langle R \rangle = 18\ M\Omega$	Kita et al. (1983b)
$\langle \tau^m \rangle = 6$ ms	Kita et al. (1983b); Paz et al. (2005)
$\theta = 20$ mV	Beurrier et al. (1999, 2000)
$\{\alpha_1, \alpha_2\} = \{0.5, 0.25\}$	Shen and Johnson (2000); Cragg et al. (2004)
$\langle \theta^{Ca} \rangle = -10$ mV	Beurrier et al. (1999); Song et al. (2000)
$\langle J^{Ca} \rangle = 90$ nA	Fitted to data from Beurrier et al. (1999); Bevan and Wilson (1999)
$\langle t_1 \rangle = 200$ ms	Beurrier et al. (1999); Bevan and Wilson (1999)
$\langle t_2 \rangle = 1000$ ms	Beurrier et al. (1999); Bevan and Wilson (1999)
$\langle R \rangle = 88\ M\Omega$	Kita and Kitai (1991)
$\langle \tau^m \rangle = 14$ ms	Kita and Kitai (1991)
$\theta = 30$ mV	Cooper and Stanford (2000)
$\{\beta_1, \beta_2\} = \{0.5, 0.5\}$	No data, but dopamine concentration in GP assumed to be less than striatum, hence $\beta_i < 1, i = 1, 2$
$\langle R \rangle = 112\ M\Omega$	Nakanishi et al. (1987, 1997)
$\langle \tau^m \rangle = 8$ ms	Nakanishi et al. (1987, 1997)
$\theta = 30$ mV	Atherton and Bevan (2005)

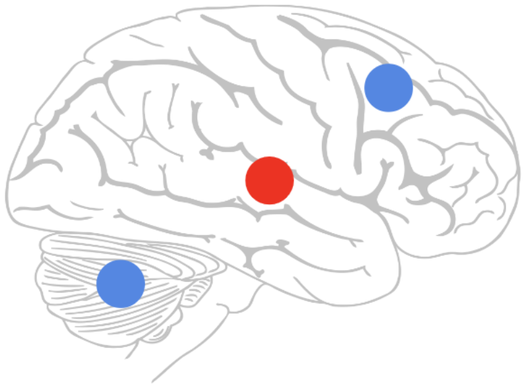
Digital twin



Caligiore et al. (2010). Psychological Review



System-level approach



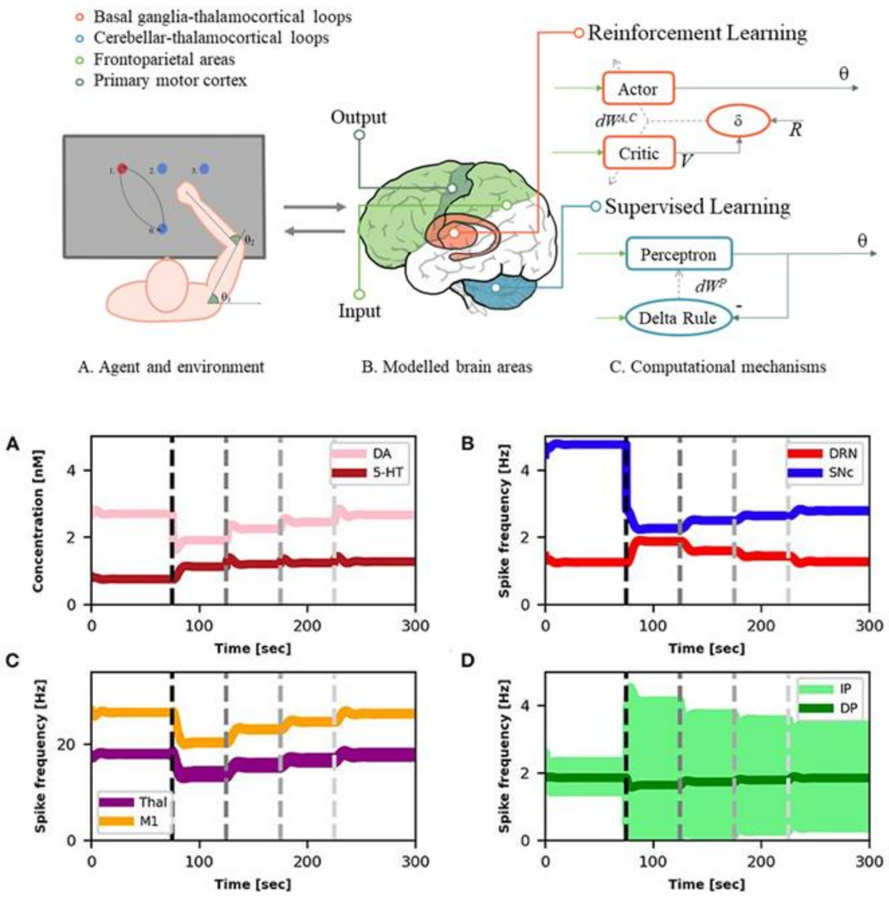
Single area approach

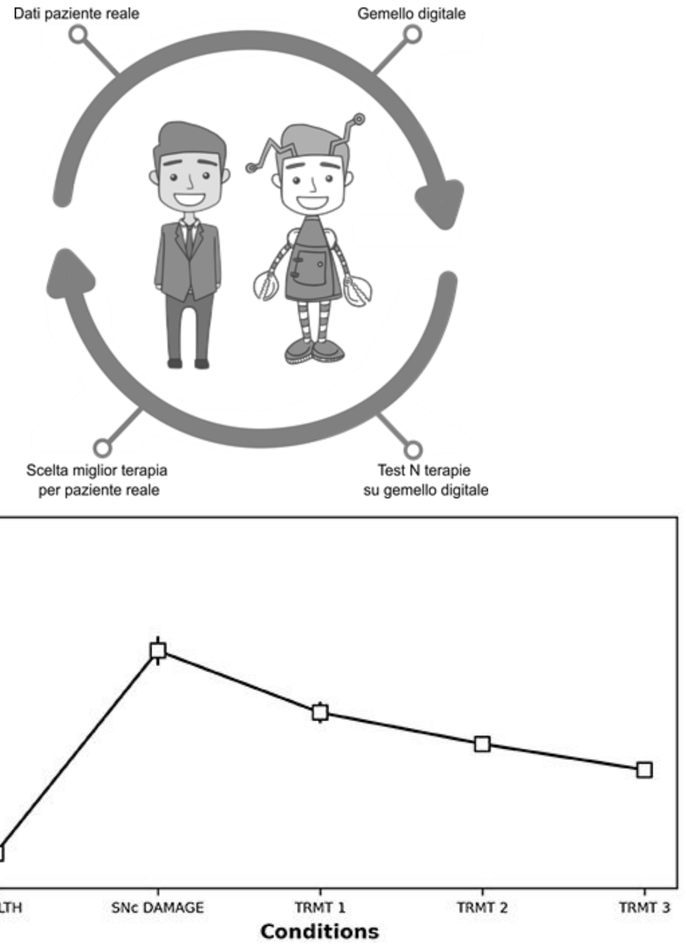
L'importanza dell'approccio integrato: Teoria, Modelli, Esperimenti

- Comprendere il cervello e il comportamento richiede un **approccio sistemico** che consideri le interazioni tra **cervello, corpo e ambiente**.
- Nessuna singola metodologia è sufficiente per comprendere il **sistema (complesso)** cervello-corpo-ambiente.
- **Teoria, modelli (IA e computazionali) e esperimenti** sono approcci complementari, ognuno con i suoi vantaggi e limitazioni.

Esempi di modelli theory-driven

Serotonina e Parkinson



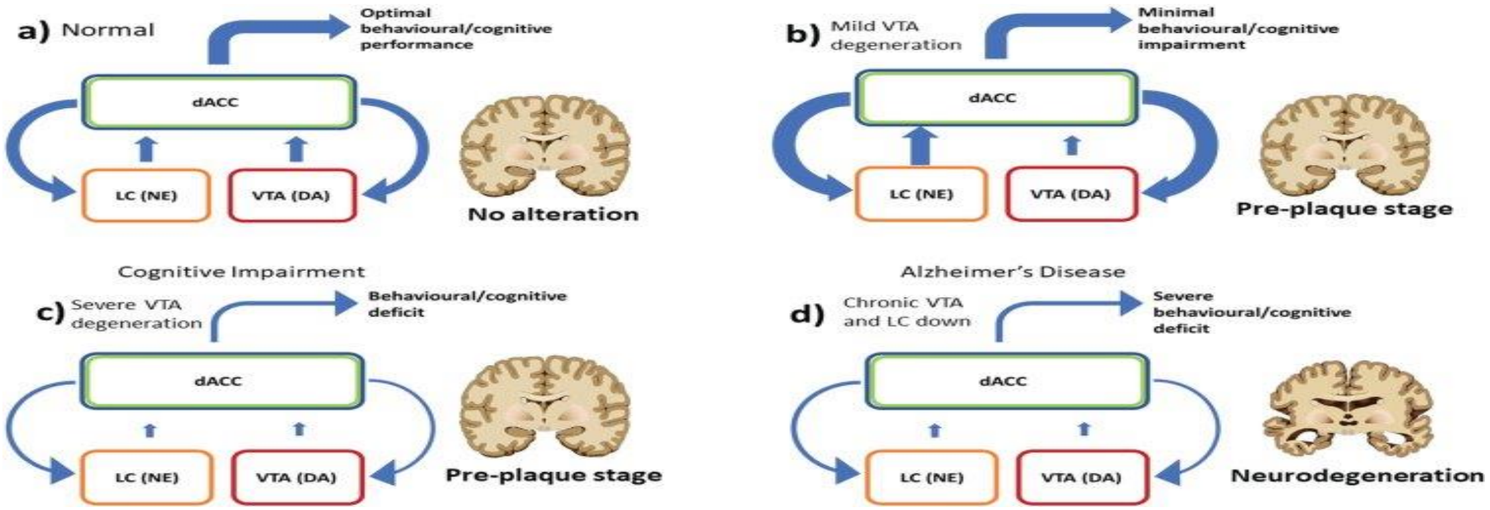


Caligiore et al. (2021). *Frontiers in Systems Neuroscience*

J Alzheimers Dis. 2020 Sep 1;77(1):275–290. doi: 10.3233/JAD-200276

Computational Modeling of Catecholamines Dysfunction in Alzheimer's Disease at Pre-Plaque Stage

Daniele Caligiore [a], Massimo Silvetti [a,*], Marcello D'Amelio [c,d], Stefano Puglisi-Allegra [e], Gianluca Baldassarre [b]



PLOS COMPUTATIONAL BIOLOGY

RESEARCH ARTICLE

Dysfunctions of the basal ganglia-cerebellar-thalamo-cortical system produce motor tics in Tourette syndrome

Daniele Caligiore[1]*, Francesco Mannella[1], Michael A. Arbib[2], Gianluca Baldassarre[1]

1 Laboratory of Computational Embodied Neuroscience, Institute of Cognitive Sciences and Technologies, National Research Council (CNR-ISTC-LOCEN), Roma, Italy, **2** Neuroscience Program, USC Brain Project, Computer Science Department, University of Southern California, Los Angeles, California, United States of America

* daniele.caligiore@istc.cnr.it

Check for updates

Abstract

Motor tics in Tourette's Syndrome

Traditional perspective: dopamine dysfunctions in the **basal ganglia**

New data suggest an involvement of **other areas** aside from the basal ganglia …

… towards a **system-level** perspective

Neurobiology of Disease

Global Dysrhythmia of Cerebro-Basal Ganglia–Cerebellar Networks Underlies Motor Tics following Striatal Disinhibition

Kevin W. McCairn,[1,2] Atsushi Iriki,[2] and Masaki Isoda[1,2]

[1]Unit on Neural Systems and Behavior, Okinawa Institute of Science and Technology, Onna-son, Okinawa 904-0495, Japan and [2]Laboratory for Symbolic Cognitive Development, RIKEN Brain Science Institute, Wako, Saitama 351-0198, Japan

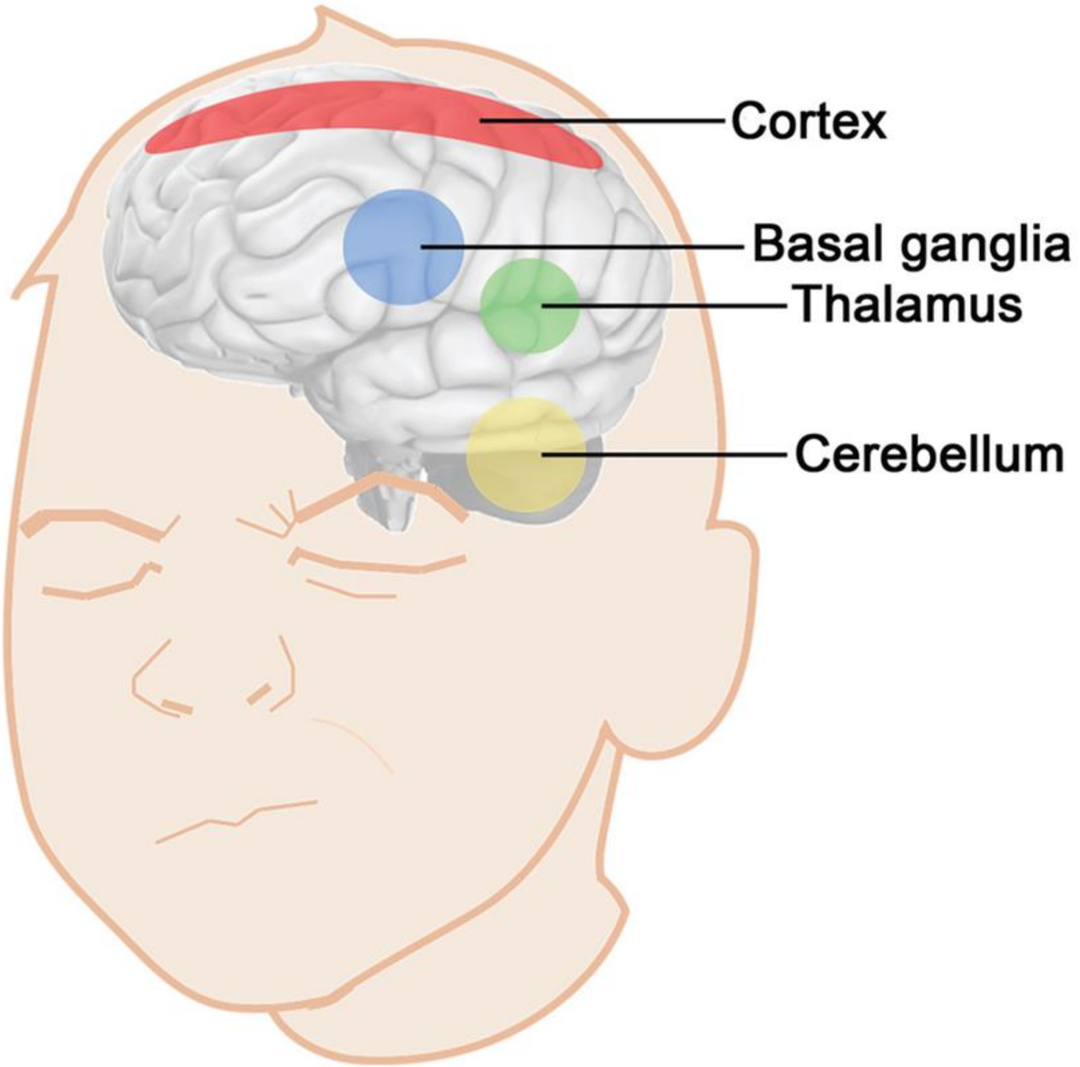
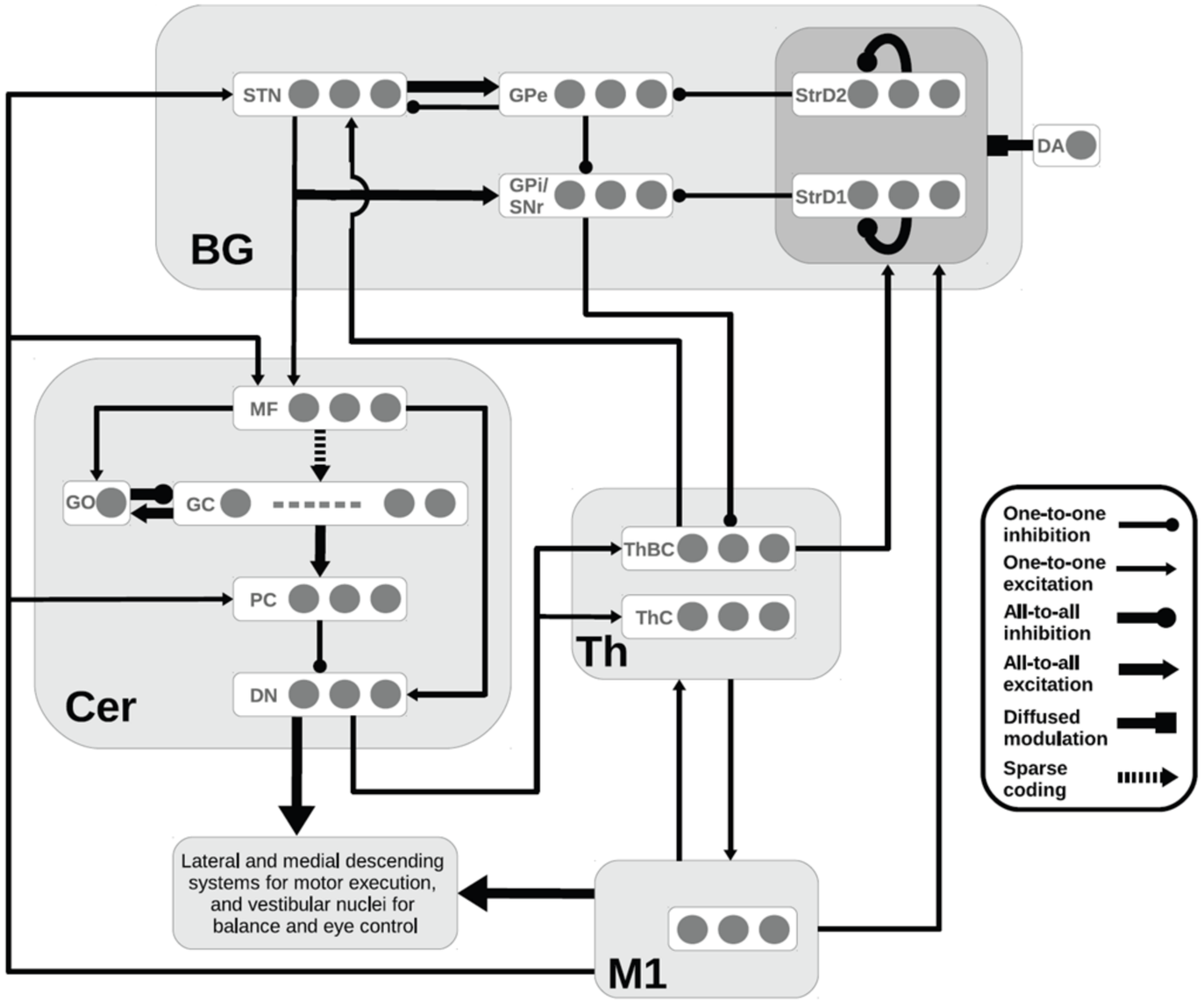


Fig 1. Key brain areas forming the "system" underlying tic production.


STN
GPe
StrD2
DA
GPi/
SNr
StrD1
BG
MF
GO
GC
PC
DN
Cer
ThBC
ThC
Th
Lateral and medial descending systems for motor execution, and vestibular nuclei for balance and eye control
M1
One-to-one inhibition
One-to-one excitation
All-to-all inhibition
All-to-all excitation
Diffused modulation
Sparse coding

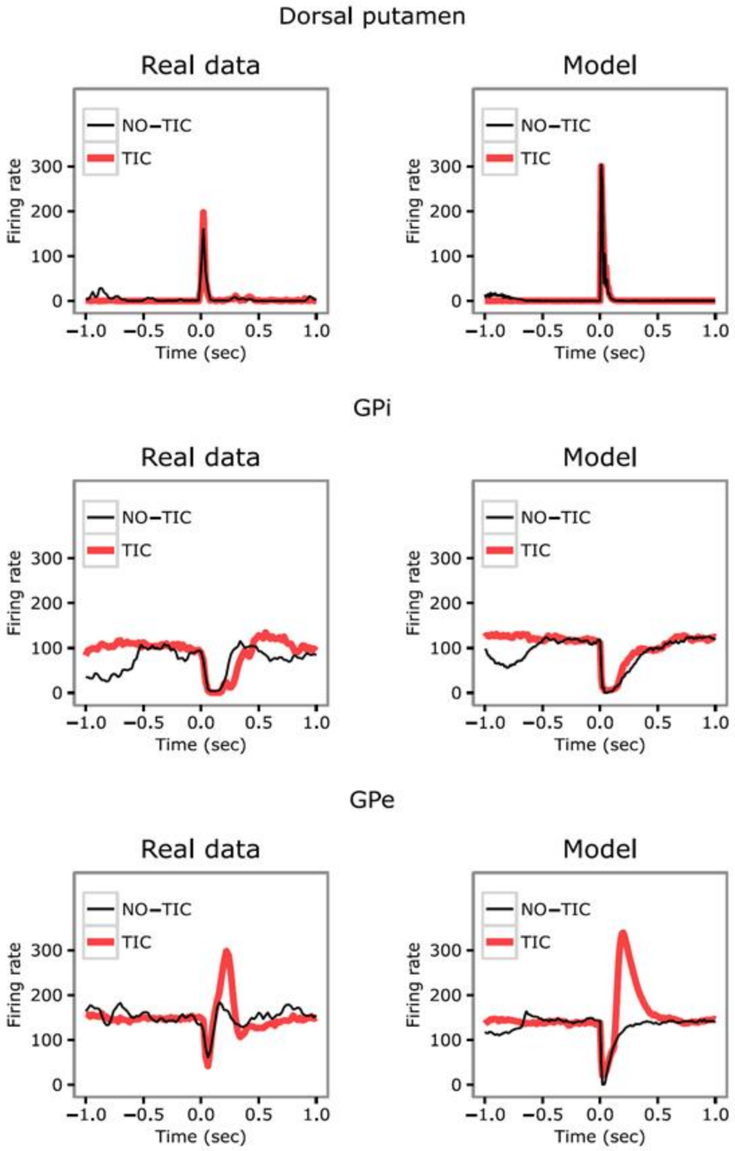
Model validation
(data reproduction)

Neurobiology of Disease

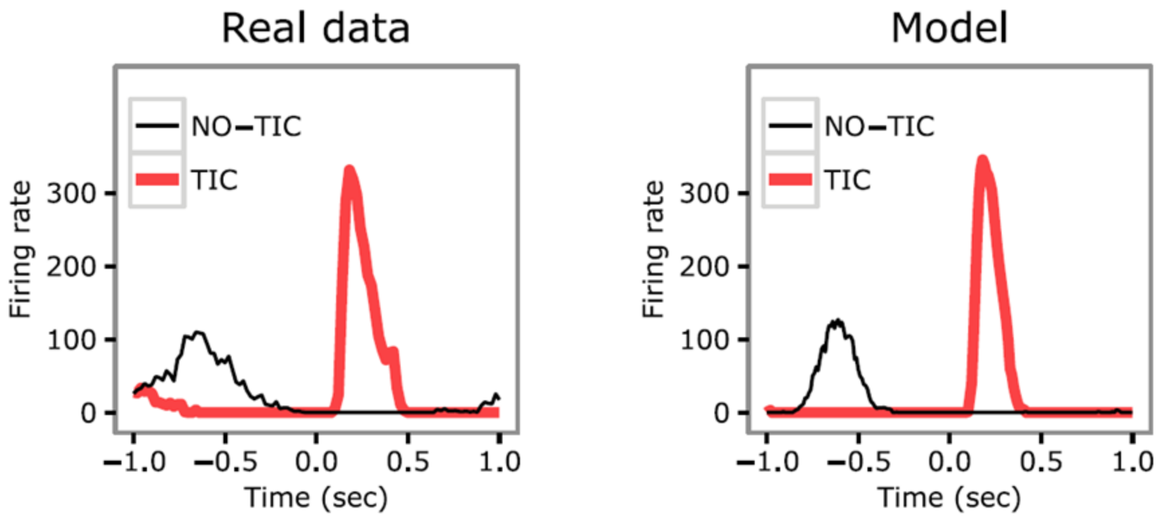
Global Dysrhythmia of Cerebro-Basal Ganglia–Cerebellar Networks Underlies Motor Tics following Striatal Disinhibition

Kevin W. McCairn,[1,2] Atsushi Iriki,[2] and Masaki Isoda[1,2]

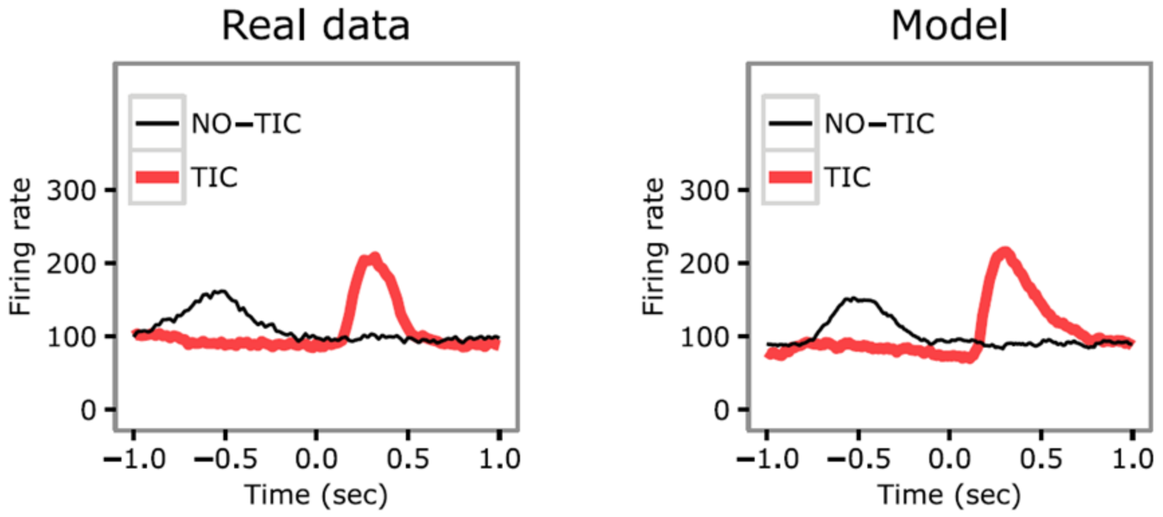
[1]Unit on Neural Systems and Behavior, Okinawa Institute of Science and Technology, Onna-son, Okinawa 904-0495, Japan and [2]Laboratory for Symbolic Cognitive Development, RIKEN Brain Science Institute, Wako, Saitama 351-0198, Japan


Dorsal putamen
Real data
Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)
GPi
Real data
Model
NO−TIC
TIC
Firing rate
Time (sec)
GPe
Real data
Model
NO−TIC
TIC
Firing rate
Time (sec)

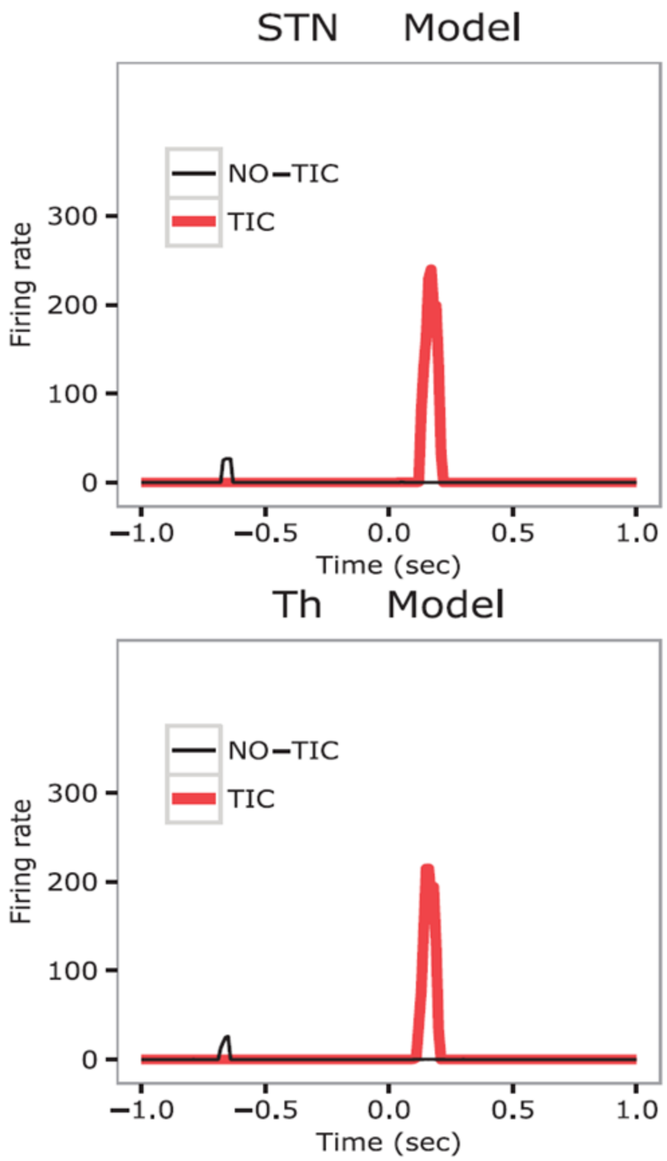
M1

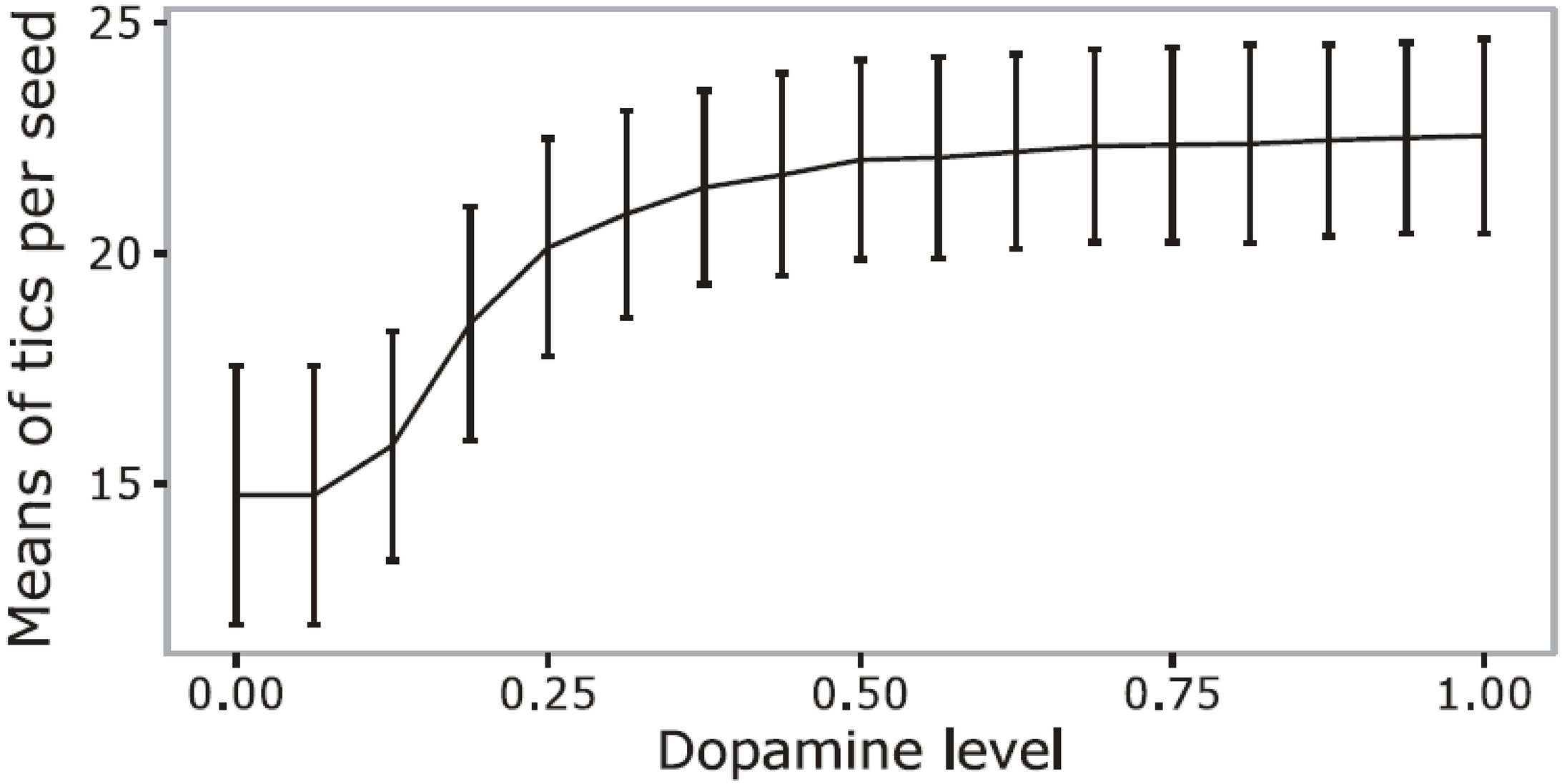

Real data
Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)

CbllCx


Real data
Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)

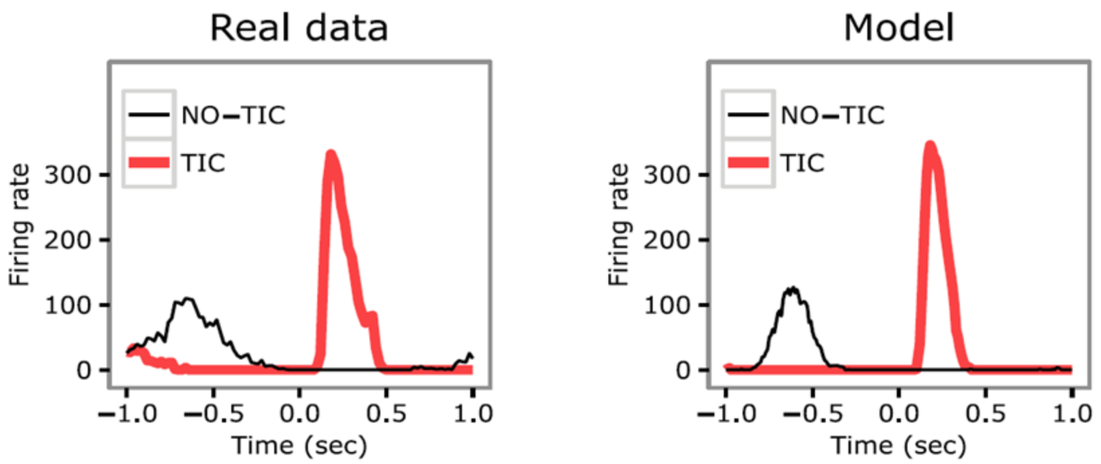
Model to get predictions


STN Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)
Th Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)

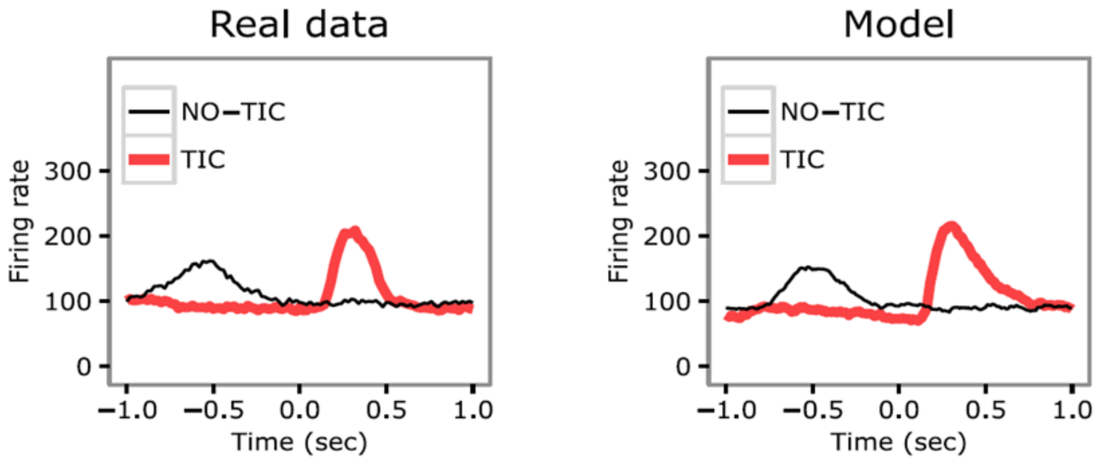

Means of tics per seed
25
20
15
0.00
0.25
0.50
0.75
1.00
Dopamine level

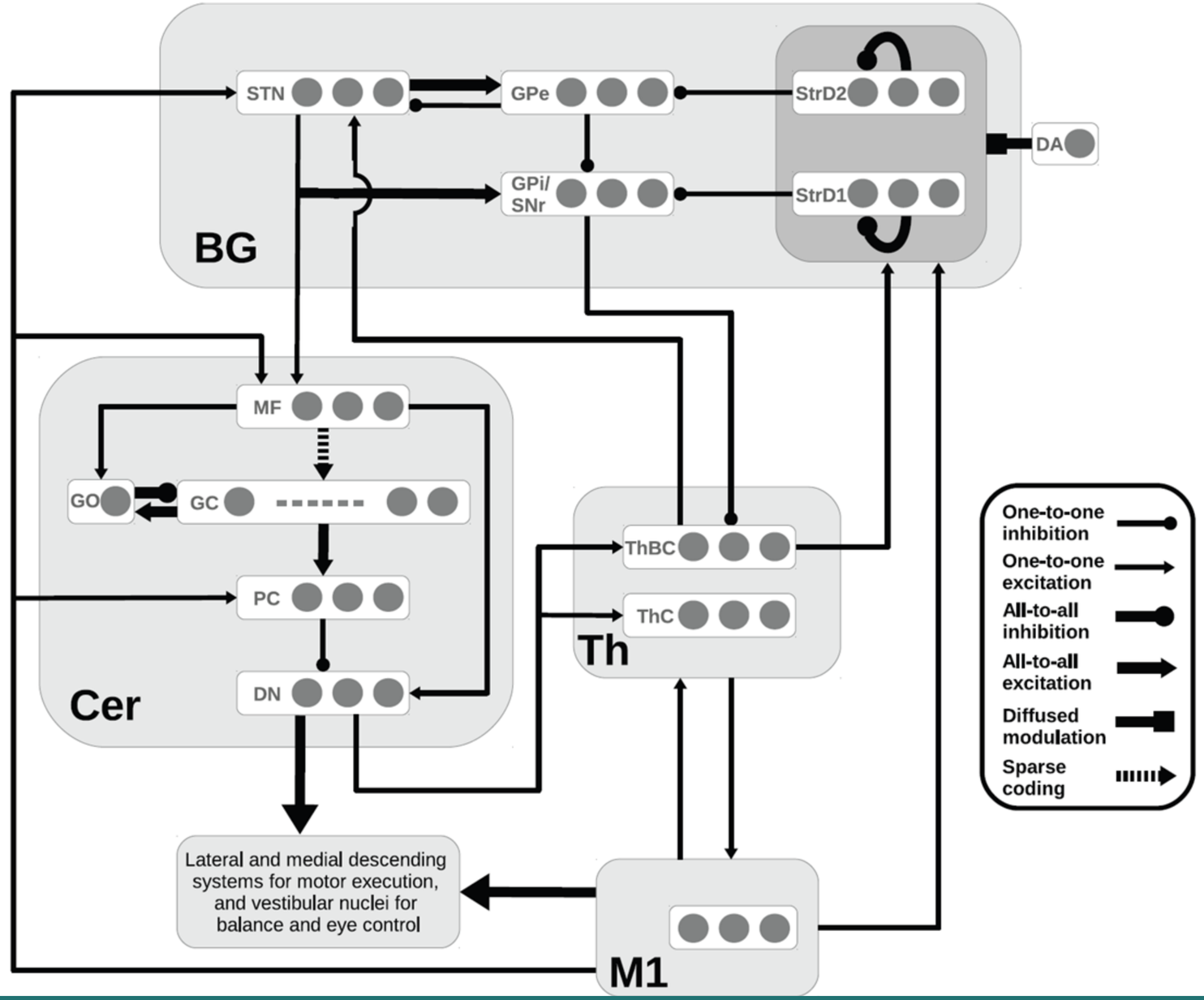
Model to suggest a mechanism underlying tic production

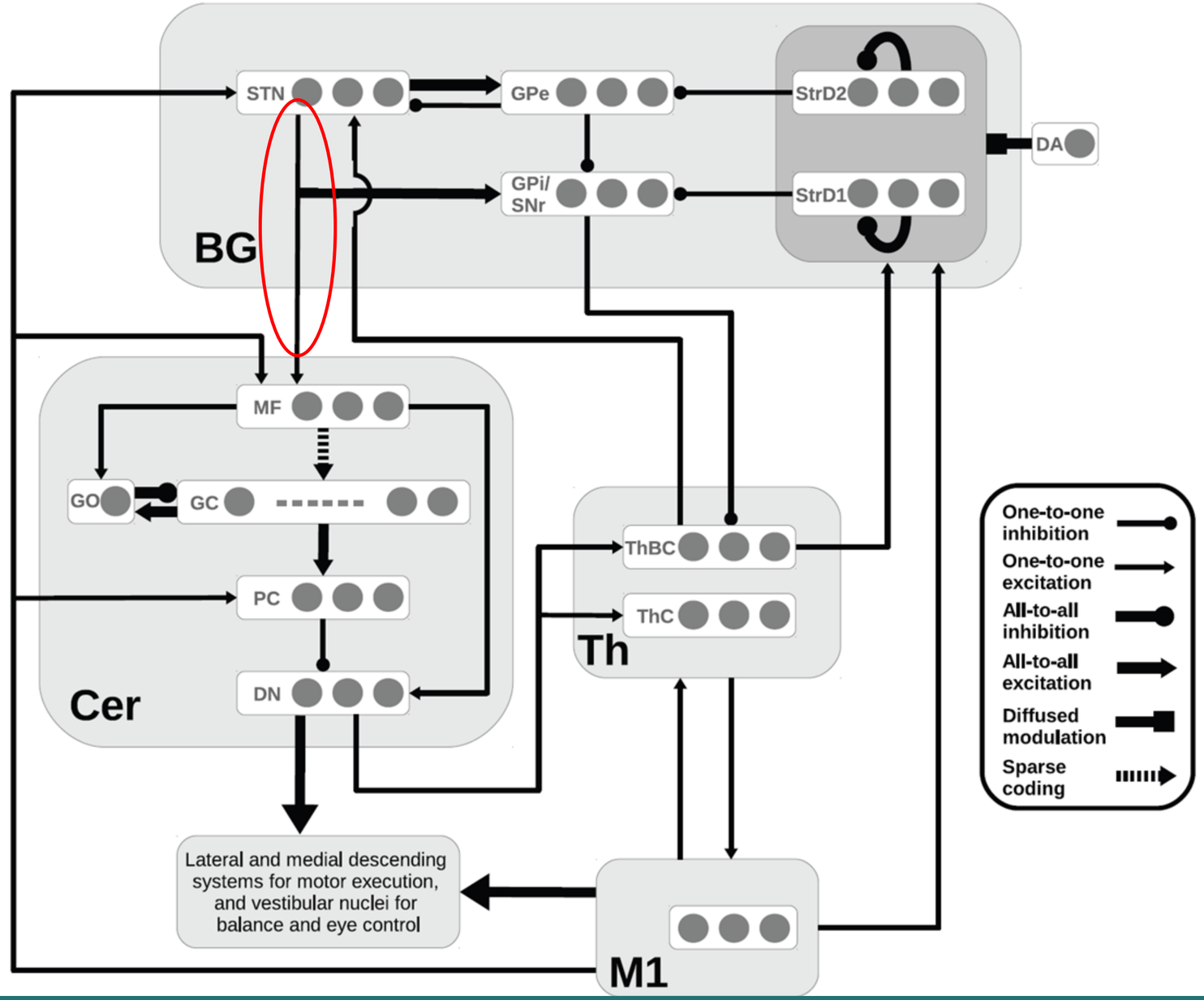
M1


Real data
Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)

CbIICx


Real data
Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)


STN
GPe
StrD2
DA
GPi/
SNr
StrD1
BG
MF
GO
GC
PC
DN
Cer
ThBC
ThC
Th
Lateral and medial descending systems for motor execution, and vestibular nuclei for balance and eye control
M1
One-to-one inhibition
One-to-one excitation
All-to-all inhibition
All-to-all excitation
Diffused modulation
Sparse coding


STN
GPe
StrD2
DA
GPi/
SNr
StrD1
BG
MF
GO
GC
PC
DN
Cer
ThBC
ThC
Th
Lateral and medial descending systems for motor execution, and vestibular nuclei for balance and eye control
M1
One-to-one inhibition
One-to-one excitation
All-to-all inhibition
All-to-all excitation
Diffused modulation
Sparse coding

The basal ganglia communicate with the cerebellum

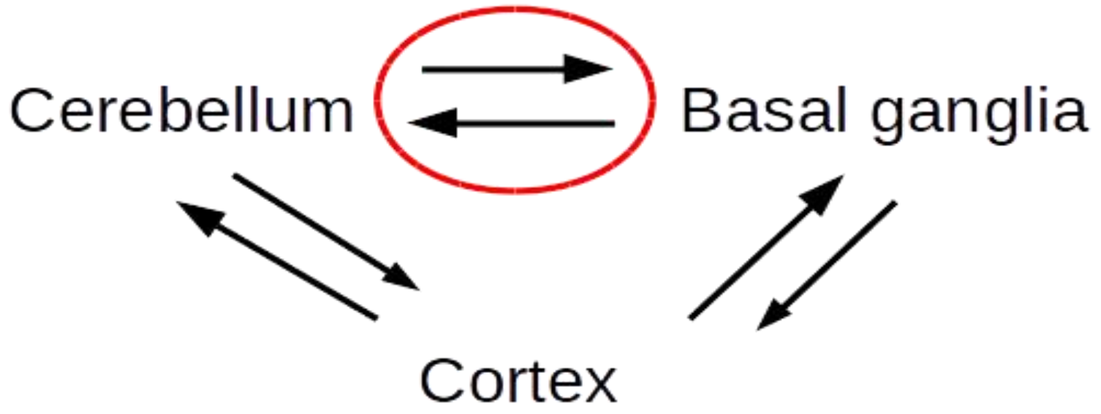
Andreea C. Bostan[a], Richard P. Dum[a], and Peter L. Strick[a,b,1]

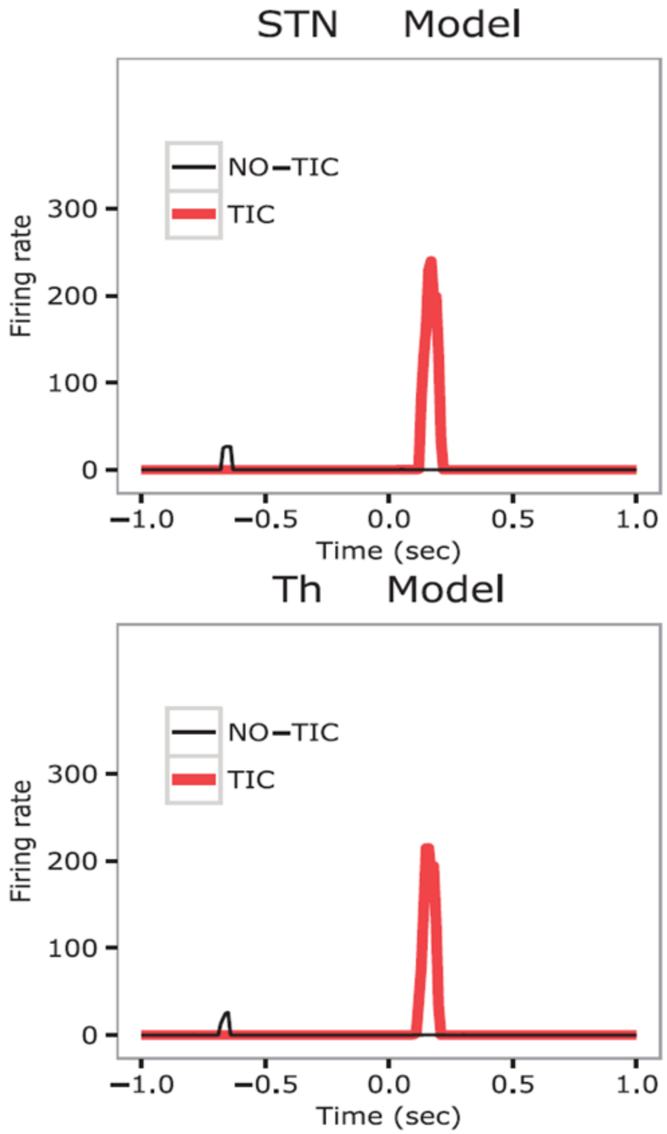
[a]Center for the Neural Basis of Cognition, Systems Neuroscience Institute and Department of Neurobiology, University of Pittsburgh School of Medicine, Pittsburgh, PA 15261; and [b]Pittsburgh Veterans Affairs Medical Center, Pittsburgh, PA 15240

Edited* by Ann M. Graybiel, Massachusetts Institute of Technology, Cambridge, MA, and approved March 29, 2010 (received for review January 13, 2010)

The basal ganglia and cerebellum are major subcortical structures that influence not only movement, but putatively also cognition and affect. Both structures receive input from and send output to the cerebral cortex. Thus, the basal ganglia and cerebellum form

Our injections targeted Crus IIp (n = 2) and the hemispheric expansion of lobule VIIB (HVIIB) (n = 1) (Fig. 2*A*). In all cases we mixed RV with a conventional tracer, the β subunit of cholera toxin (CTb, 0.02%). We used this mixture to facilitate identifi-




STN Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)
Th Model
NO−TIC
TIC
Firing rate
300
200
100
0
−1.0
−0.5
0.0
0.5
1.0
Time (sec)

Come si realizza un modello theory-driven

Fase 1: Costruire schema a blocchi sulla base della teoria

Fase 2: Scrivere le equazioni del modello

Fase 3: Tradurre le equazioni in Python

Fase 4: Vincolare le equazioni per riprodurre i dati

Caligiore, D., & Carli, S. (2025). Simulating the Brain. Springer Nature

Esempio

Costruire un modello theory-driven (ad equazioni differenziali) per studiare il ruolo del cervelletto nella malattia di Alzheimer

Step 1. Letteratura per costruire il modello (circuito e attività)

Review > Brain. 2018 Jan 1;141(1):37-47. doi: 10.1093/brain/awx194.

The cerebellum in Alzheimer's disease: evaluating its role in cognitive decline

Heidi I L Jacobs [1 2 3], David A Hopkins [4 5], Helen C Mayrhofer [2], Emiliano Bruner [6], Fred W van Leeuwen [4], Wijnand Raaijmakers [2], Jeremy D Schmahmann [7]

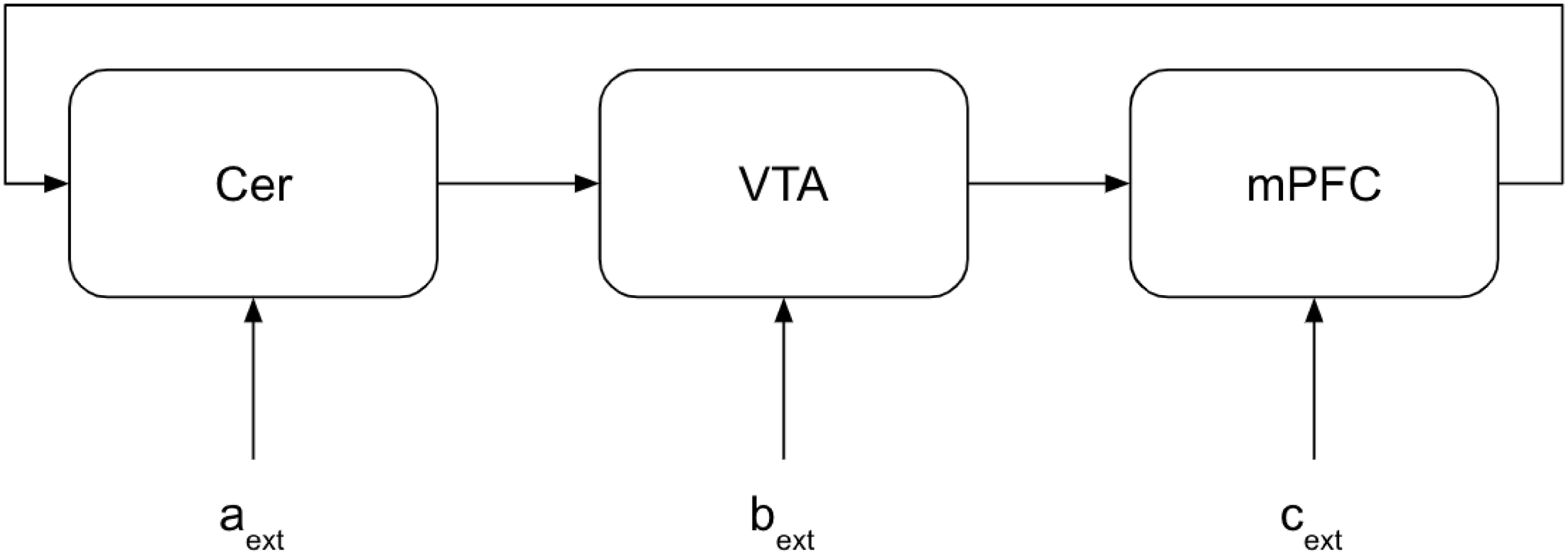
Cerebellum. Author manuscript; available in PMC 2014 Aug 1.
PMCID: PMC3854915
Published in final edited form as:
NIHMSID: NIHMS449131
Cerebellum. 2013 Aug; 12(4): 10.1007/s12311-013-0462-2.
PMID: 23436049
doi: 10.1007/s12311-013-0462-2

Reorganization of circuits underlying cerebellar modulation of prefrontal cortical dopamine in mouse models of autism spectrum disorder

Tiffany D. Rogers,[1] Price E. Dickson,[1] Eric McKimm,[1] Detlef H. Heck,[2] Dan Goldowitz,[3] Charles D. Blaha,[1,*] and Guy Mittleman[1]

Approccio di sistema anche quando studio

Leggo letteratura al di fuori del dominio di studio, ma potenzialmente collegata


Cer
VTA
mPFC
a ext
b ext
c ext

- Cervelletto: aspetti motori, anticipazione, aspetti non motori
- VTA: rilascio di dopamina
- mPFC: memoria di lavoro (working memory)

Come si realizza un modello theory-driven

Fase 1: Costruire schema a blocchi sulla base della teoria

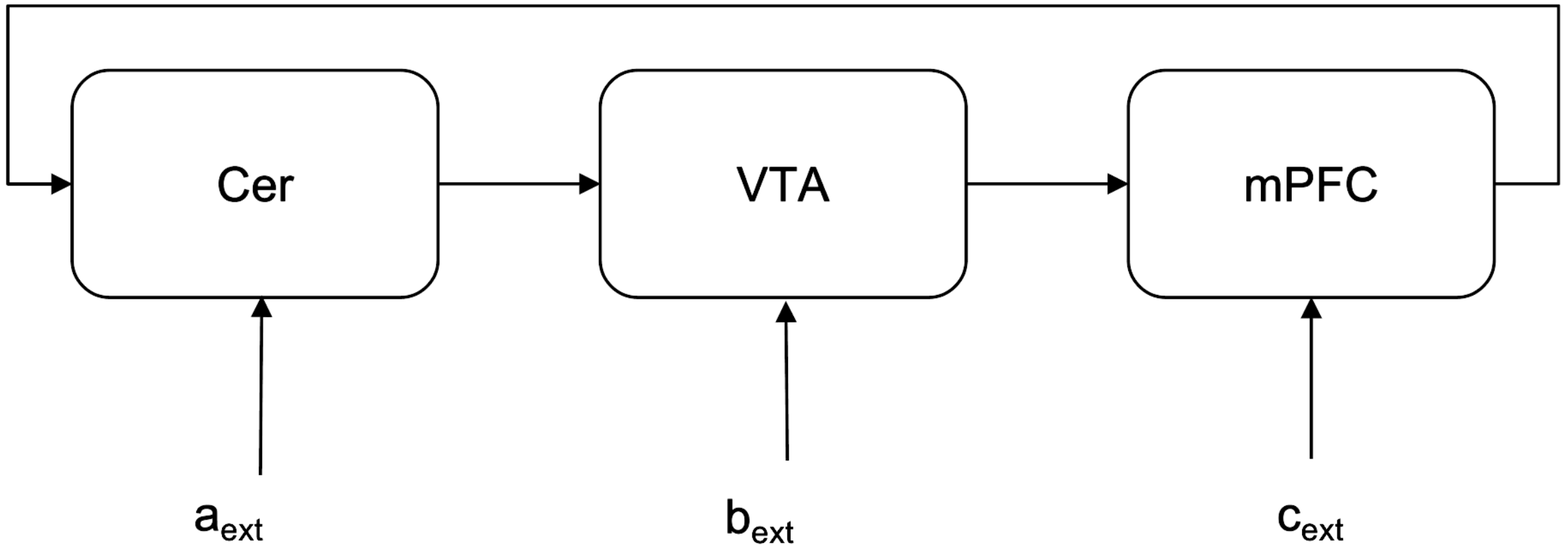
Fase 2: Scrivere le equazioni del modello

Fase 3: Tradurre le equazioni in Python

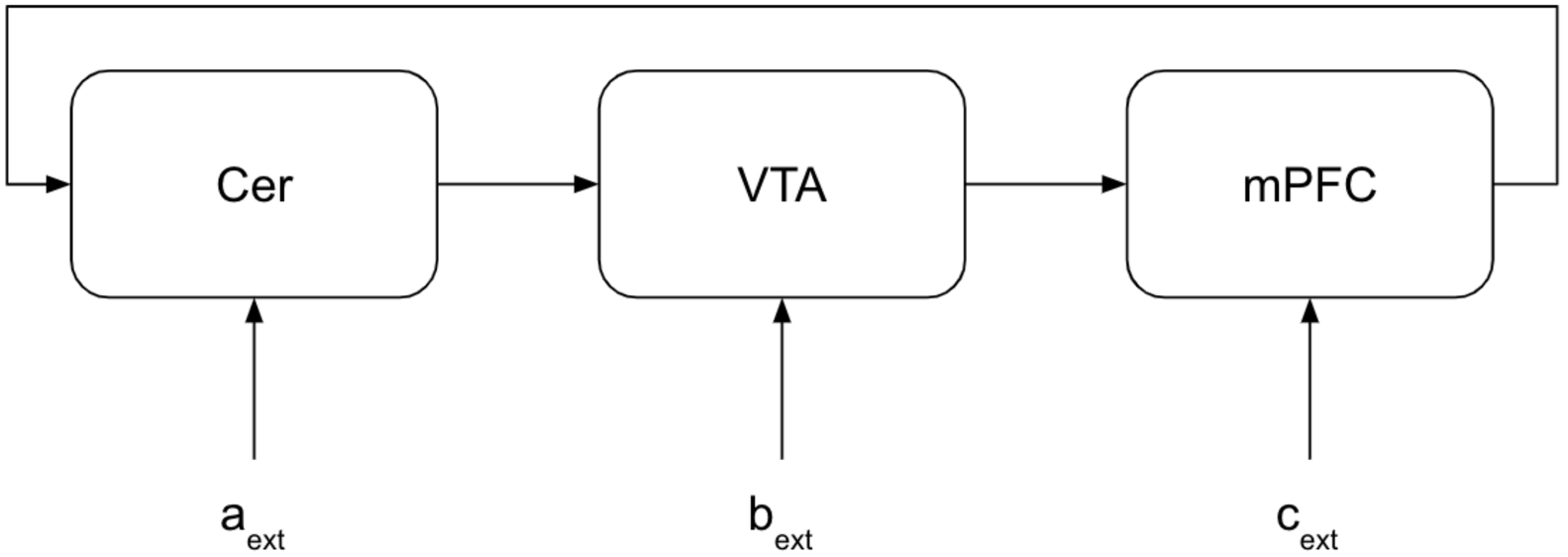
Fase 4: Vincolare le equazioni per riprodurre i dati

Caligiore, D., & Carli, S. (2025). Simulating the Brain. Springer Nature

Step 1. Letteratura per costruire il modello (circuito e attività)



Step 2. Scrivere equazioni del modello



$$\frac{dCer}{dt} = a_{ext} + a_1 \cdot mPFC - a_2 \cdot Cer$$
$$\frac{dVTA}{dt} = b_{ext} + b_1 \cdot Cer - b_2 \cdot VTA$$
$$\frac{dmPFC}{dt} = c_{ext} + c_1 \cdot VTA - c_2 \cdot mPFC$$

Step 1. Letteratura per costruire il modello (circuito e attività)

Step 2. Scrivere equazioni del modello

Step 3. Scrivere il modello in Python

```python
# Import of needed library
import numpy as np
import matplotlib.pyplot as plt
import matplotlib
from scipy.integrate import odeint
```

```python
def model (x, t, a_ext, a_1, a_2, b_ext, b_1, b_2, c_ext, c_1, c_2):
    Cer  = x[0]      # Firing rate of cerebellar neurons (Hz)
    VTA  = x[1]      # Firing rate of ventral tegmental area neurons (Hz)
    mPFC = x[2]      # Firing rate of medial prefrontal cortex neurons (Hz)

    # Simulate AD
    if (t > 100 and t < 400):
      c_1 = 2.0 # AD
      #a_1 = 0.01
      #a_1 = 1.0

    if (t>300 and t<400):
      a_ext = 20.0

    dCerdt  = a_ext + a_1*mPFC - a_2*Cer
    dVTAdt  = b_ext + b_1*Cer - b_2*VTA
    dmPFCdt = c_ext + c_1*VTA - c_2*mPFC

    return [dCerdt, dVTAdt, dmPFCdt]
```

frontiers in Aging Neuroscience

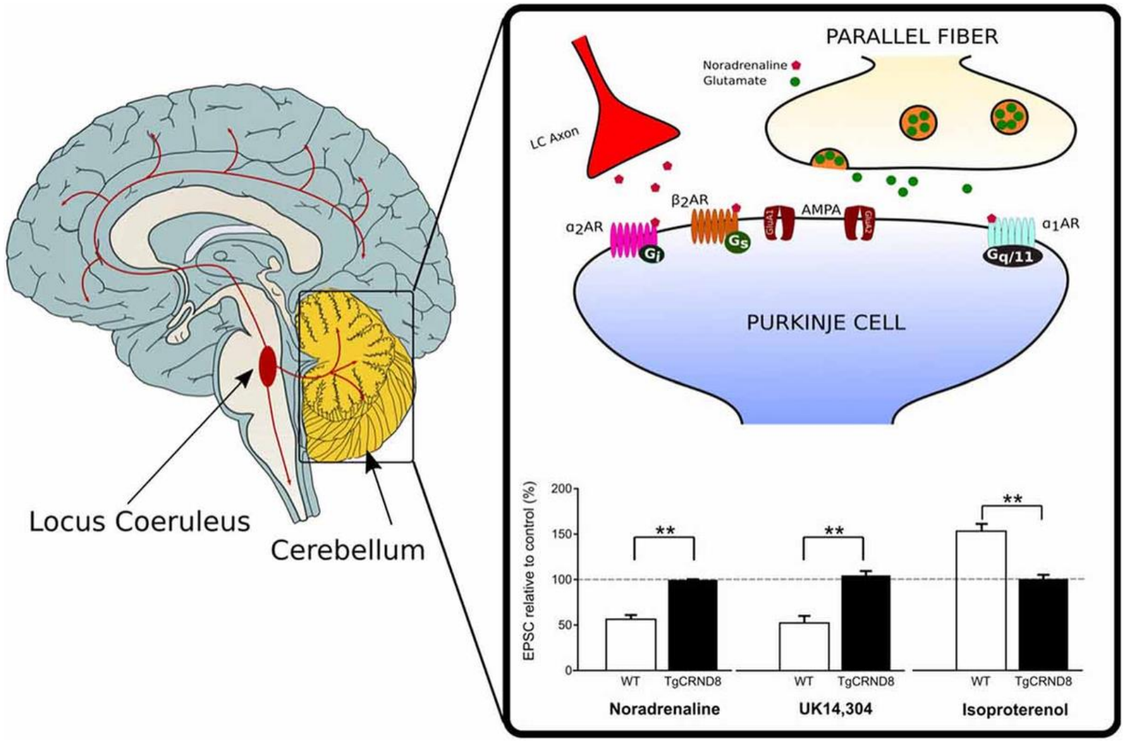
REVIEW
published: 27 November 2018
doi: 10.3389/fnagi.2018.00396

The Emerging Role of Altered Cerebellar Synaptic Processing in Alzheimer's Disease

https://www.frontiersin.org/articles/10.3389/fnagi.2018.00396/full

Eriola Hoxha[1,2], Pellegrino Lippiello[3], Fabio Zurlo[3], Ilaria Balbo[1,2], Rita Santamaria[3], Filippo Tempia[1,2,4] and Maria Concetta Miniaci[3]*

[1]Neuroscience Institute Cavalieri Ottolenghi (NICO), Turin, Italy, [2]Department of Neuroscience, University of Torino, Turin, Italy, [3]Department of Pharmacy, School of Medicine, University of Naples Federico II, Naples, Italy, [4]National Institute of Neuroscience (INN), Turin, Italy



La letteratura mi aiuta ad assegnare i valori di alcuni parametri del modello in modo da riprodurre inizialmente l'attività neurale misurata in soggetti sani (**steady-state**)

Sito dove è possibile trovare i valori di alcuni parametri https://neuroelectro.org/

Cer: 20 Hz
VTA: 5 Hz
mPFC: 110 Hz

```
a_ext = 2.2     # External drive to cerebellum (Cer)
a_1   = 0.15    # Influence of mPFC on Cer
a_2   = 1       # Decay constant of cerebellar neurons

b_ext = 1.35     # External drive to ventral tegmental area (VTA)
b_1   = 0.25     # Influence of Cer on VTA
b_2   = 1        # Decay constant of VTA neurons

c_ext = 1.6      # External drive to medial prefrontal cortex (mPFC)
c_1   = 3.5      # Influence of VTA on mPFC
c_2   = 0.2      # Decay constant of mPFC neurons
```

Soggetto sano

$$\frac{dCer}{dt} = a_{ext} + a_1 \cdot mPFC - a_2 \cdot Cer$$
$$\frac{dVTA}{dt} = b_{ext} + b_1 \cdot Cer - b_2 \cdot VTA$$
$$\frac{dmPFC}{dt} = c_{ext} + c_1 \cdot VTA - c_2 \cdot mPFC$$

Cer: 20 Hz
VTA: 5 Hz
mPFC: 110 Hz

```
a_ext = 2.2    # External drive to cerebellum (Cer)
a_1   = 0.15   # Influence of mPFC on Cer
a_2   = 1      # Decay constant of cerebellar neurons

b_ext = 1.35    # External drive to ventral tegmental area (VTA)
b_1   = 0.25    # Influence of Cer on VTA
b_2   = 1       # Decay constant of VTA neurons

c_ext = 1.6     # External drive to medial prefrontal cortex (mPFC)
c_1   = 3.5     # Influence of VTA on mPFC
c_2   = 0.2     # Decay constant of mPFC neurons
```

Step 1. Letteratura per costruire il modello (circuito e attività)

Step 2. Scrivere equazioni del modello

Step 3. Scrivere il modello in Python

Step 4. Modificare i parametri (liberi) per riprodurre i **dati target**

Soggetto sano

Cer: 20 Hz
VTA: 5 Hz
mPFC: 110 Hz

Alzheimer

Cer: 7 Hz
VTA: 3 Hz
mPFC: 40 Hz

Genetic Algorithm

Invece di trovare i parametri del modello a mano per prove ed errori posso utilizzare un algoritmo genetico

Genetic Algorithm

Invece di trovare i parametri del modello a mano per prove ed errori posso utilizzare un algoritmo genetico

Modello
(p_1, p_2, p_3, p_4, p_5)

Genetic Algorithm

Invece di trovare i parametri del modello a mano per prove ed errori posso utilizzare un algoritmo genetico

Modello
(p_1, p_2, p_3, p_4, p_5)

p_1, p_2, p_3
sono parametri liberi che non conosco e devo trovare. Inizialmente gli assegno valori casuali

p_4, p_5
sono invece parametri che conosco, ad esempio presi da articoli scientifici

Genetic Algorithm

Invece di trovare i parametri del modello a mano per prove ed errori posso utilizzare un algoritmo genetico

Modello
(a_{ext}, a_1, a_2, b_{ext}, b_1, b_2, c_{ext}, c_1, c_2)

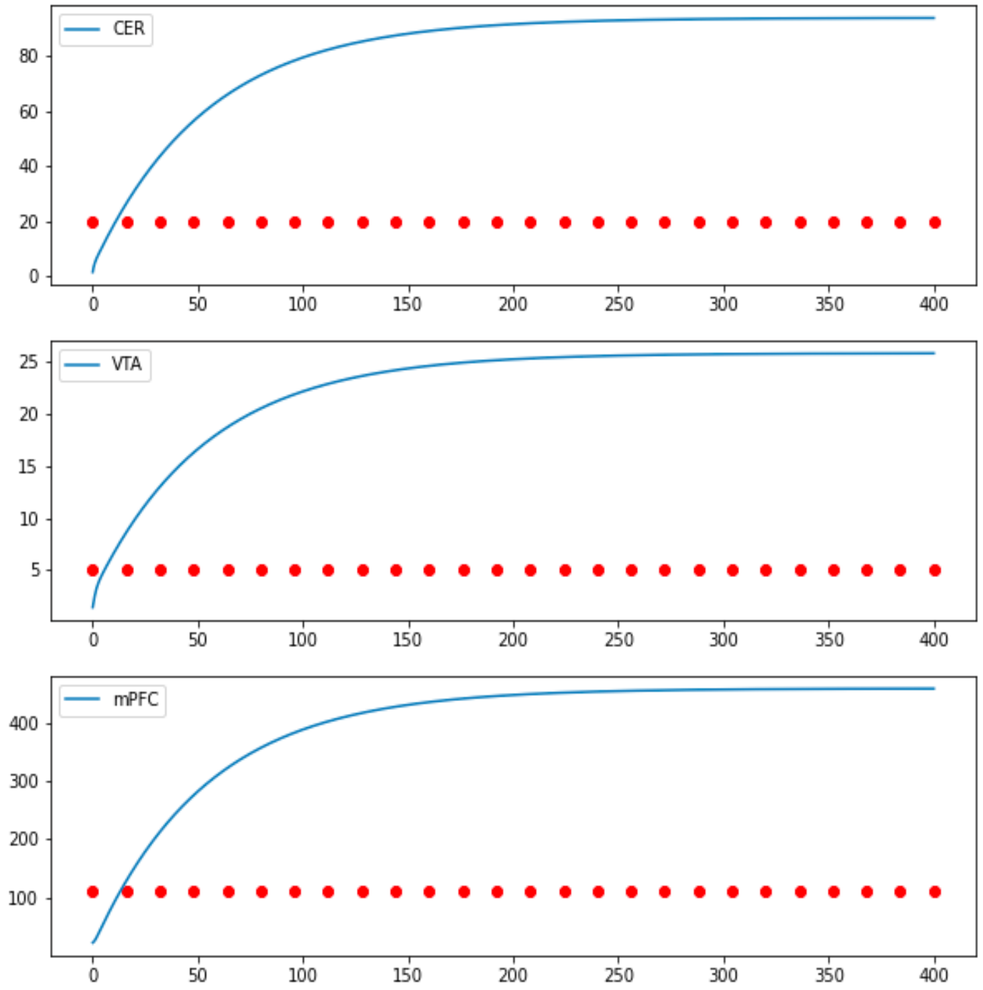
a_{ext}, a_1, a_2, b_{ext}, b_1, b_2
sono parametri liberi che non conosco e devo trovare. Inizialmente gli assegno valori casuali

c_{ext}, c_1, c_2
sono invece parametri che conosco, ad esempio presi da articoli scientifici

Performace iniziale modello

Modello
(a_{ext}, a_1, a_2, b_{ext}, b_1, b_2, c_{ext}, c_1, c_2)

i puntini rossi indicano il valore desiderato (target data)



Utilizzo GA per trovare i parametri liberi

Utilizzo GA per trovare i parametri liberi

1. Creo tante copie del modello con valori dei parametri liberi (a_{ext}, a_1, a_2, b_{ext}, b_1, b_2) leggermente diversi - prima **generazione**.

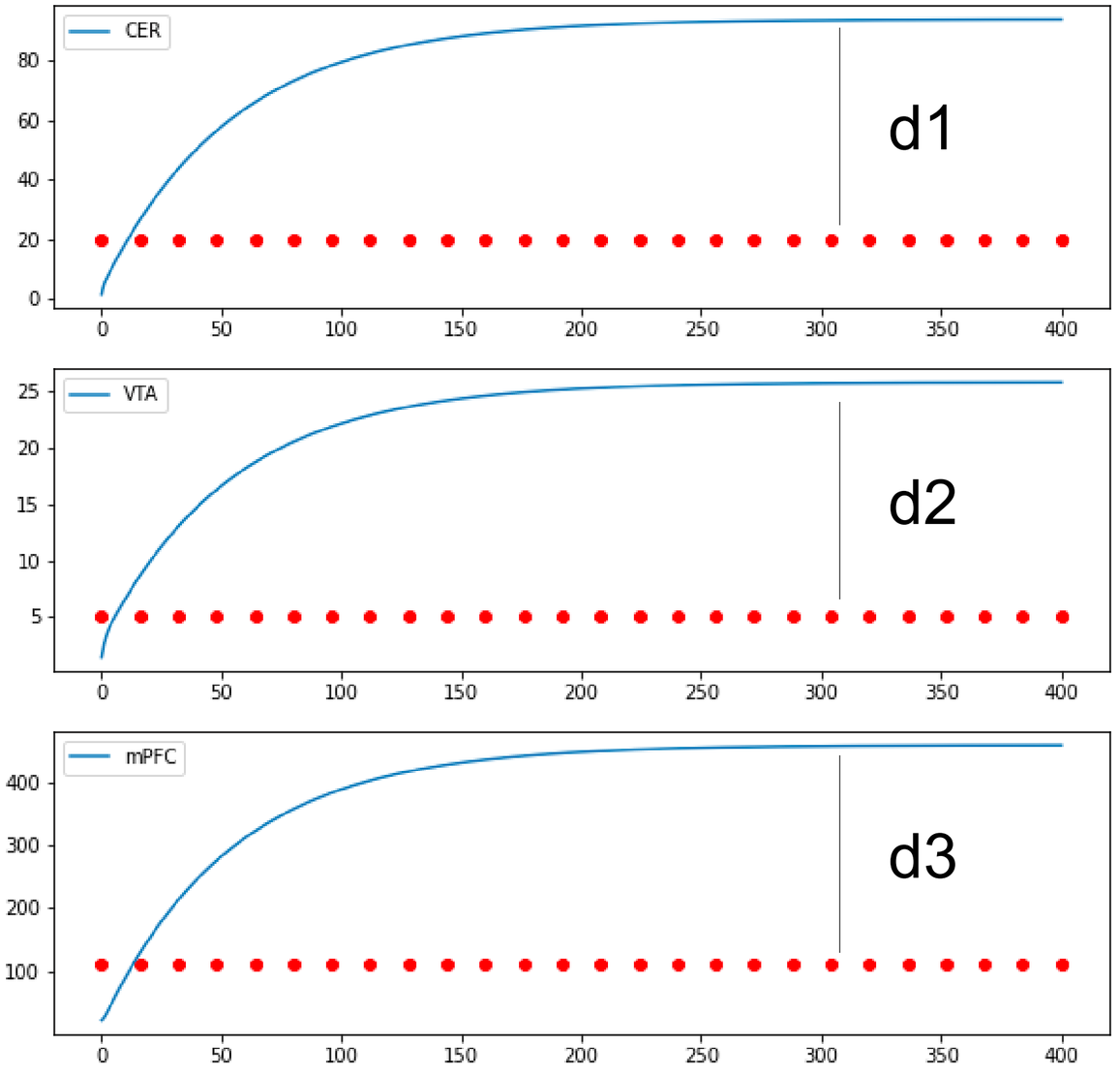
Utilizzo GA per trovare i parametri liberi

1. Creo tante copie del modello con valori dei parametri liberi (a_{ext}, a_1, a_2, b_{ext}, b_1, b_2) leggermente diversi - prima **generazione**.

1. Calcolo la performance di tutti questi modelli e la relativa **fitness** (performance effettiva - performance desiderata)

La fitness …

fitness = (d1 + d2 + d3)/3



Utilizzo GA per trovare i parametri liberi

1. Creo tante copie del modello con valori dei parametri liberi (a_{ext}, a_1, a_2, b_{ext}, b_1, b_2) leggermente diversi - prima **generazione**.

1. Calcolo la performance di tutti questi modelli e la relativa **fitness** (performance effettiva - performance desiderata)

Utilizzo GA per trovare i parametri liberi

1. Creo tante copie del modello con valori dei parametri liberi (a_{ext}, a_1, a_2, b_{ext}, b_1, b_2) leggermente diversi - prima **generazione**.

1. Calcolo la performance di tutti questi modelli e la relativa **fitness** (performance effettiva - performance desiderata)

1. Creo una seconda generazione di modelli a partire dai modelli della prima generazione con fitness più alta (**mutando** leggermente i loro parametri).

Utilizzo GA per trovare i parametri liberi

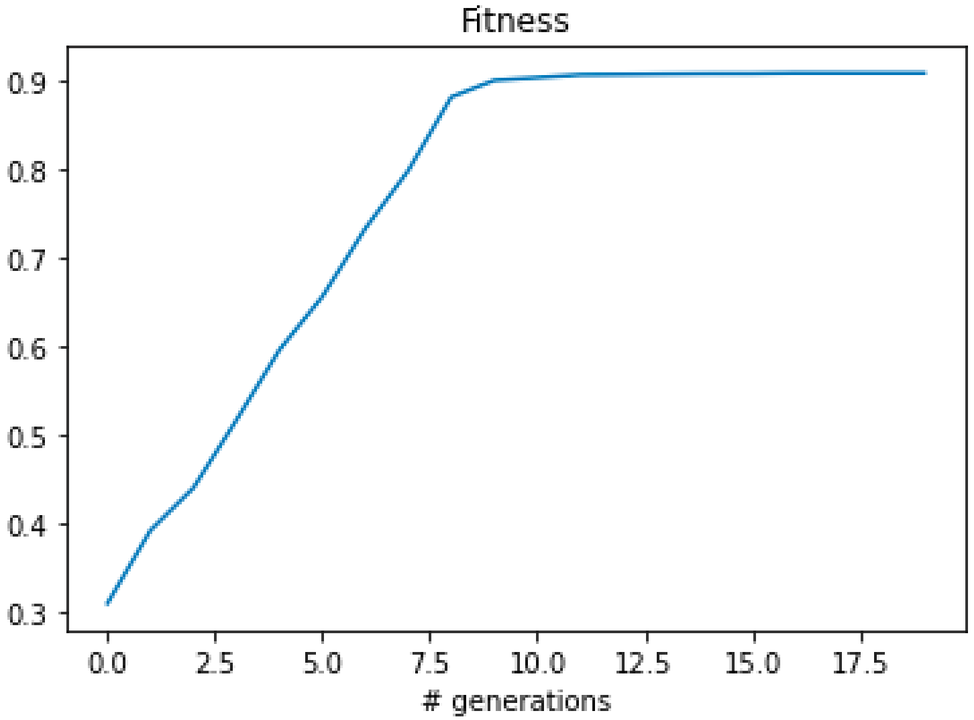
1. Creo tante copie del modello con valori dei parametri liberi (a_{ext}, a_1, a_2, b_{ext}, b_1, b_2) leggermente diversi - prima **generazione**.

1. Calcolo la performance di tutti questi modelli e la relativa **fitness** (performance effettiva - performance desiderata)

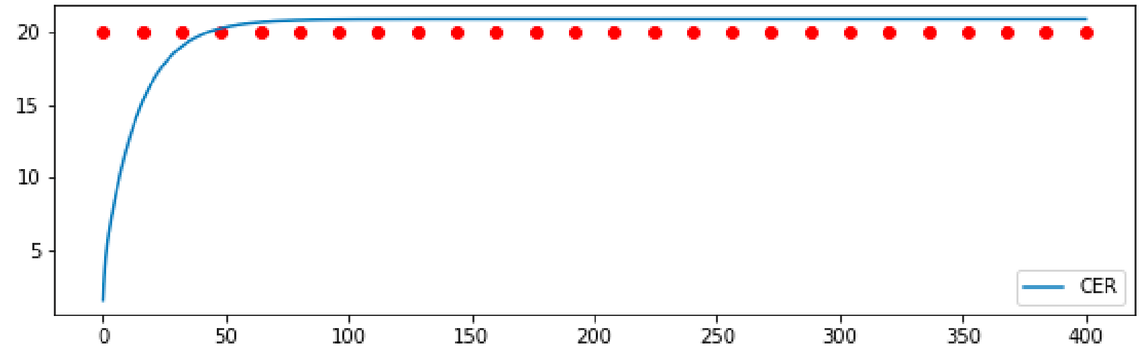
1. Creo una seconda generazione di modelli a partire dai modelli della prima generazione con fitness più alta (**mutando** leggermente i loro parametri).

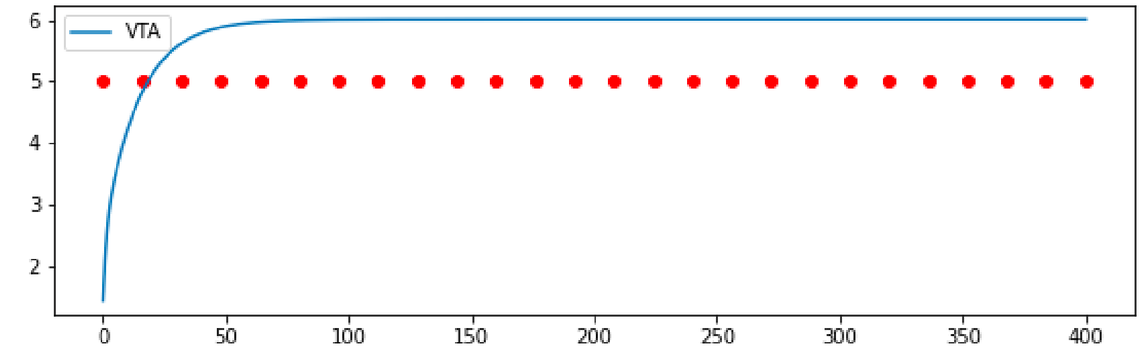
1. Ripeto la procedura 1-3 per N generazioni finché la fitness massima tende ad 1

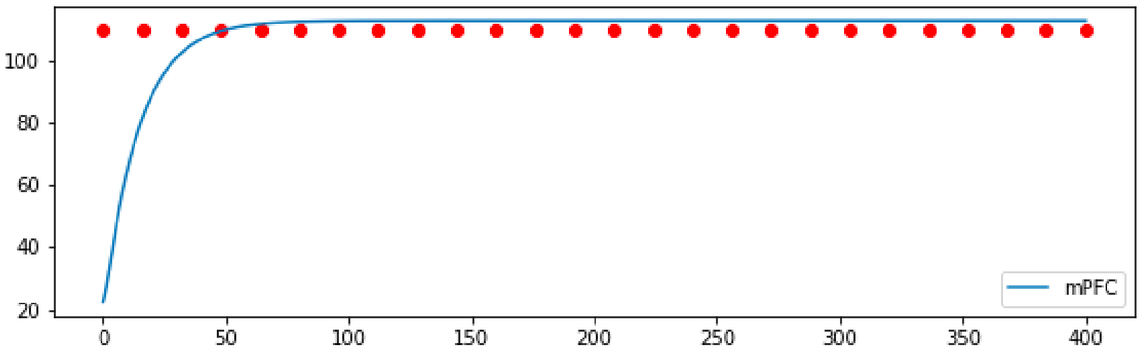
Il modello con fitness più alta dell'ultima generazione è quello con i parametri migliori


Fitness
0.9
0.8
0.7
0.6
0.5
0.4
0.3
0.0
2.5
5.0
7.5
10.0
12.5
15.0
17.5
generations

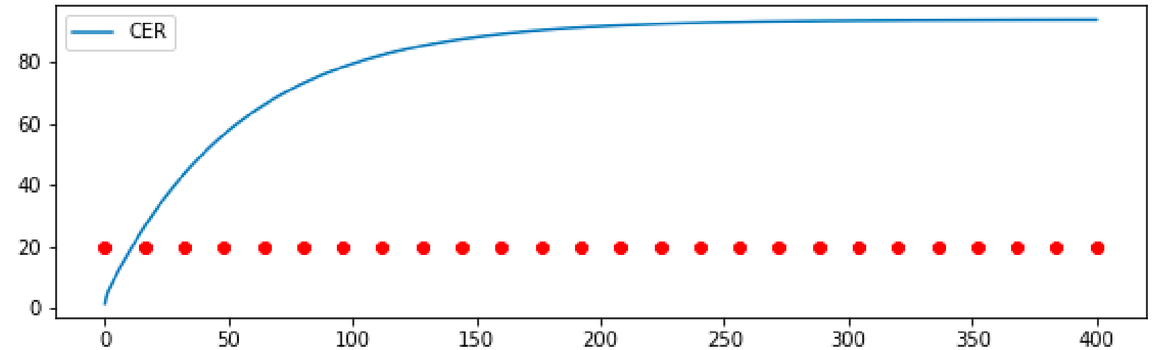
Performance finale

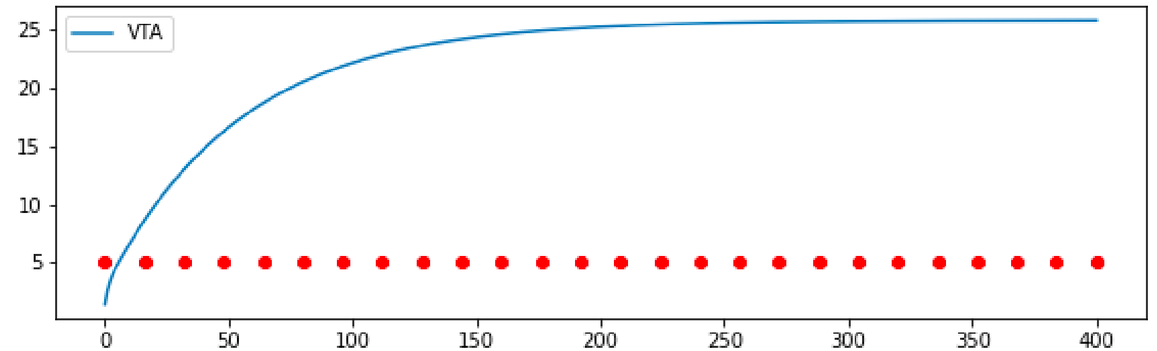


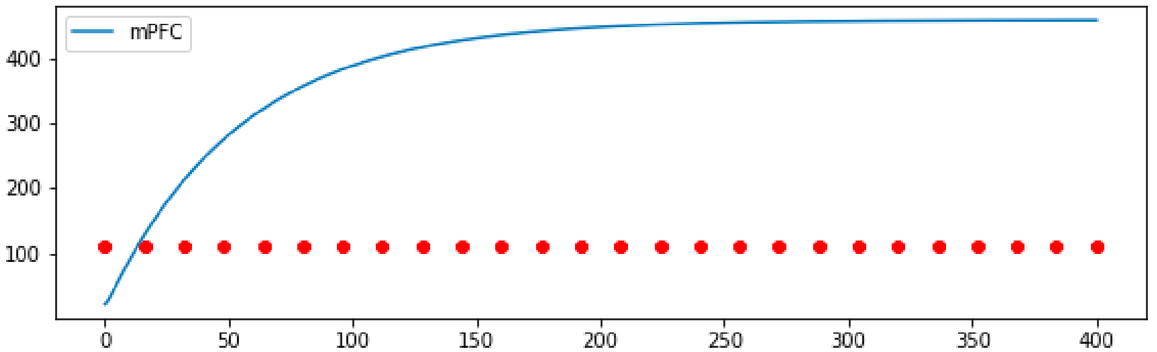




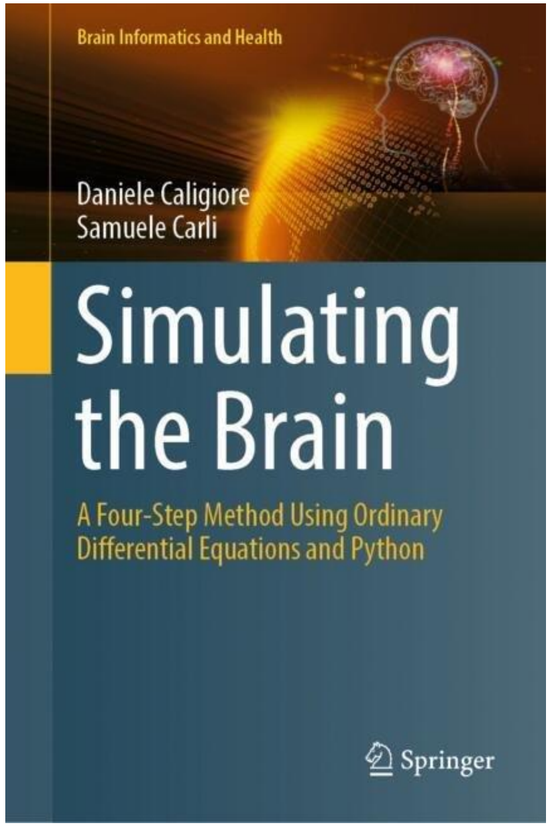
Performance iniziale







Le quattro fasi sono descritte in questo libro



http://bit.ly/4e7Gyfz

Cap 1: Python
Cap 2: Matematica Essenziale
Cap 3: Esempi di Modelli Theory-Driven in Generale
Cap 4: Esempi di Modelli del Cervello