Summary of part I: prediction and RL

Prediction is important for action selection

- **The problem:** prediction of future reward
- **The algorithm:** temporal difference learning
- **Neural implementation:** dopamine dependent learning in BG

⇒ A precise computational model of learning allows one to look in the brain for “hidden variables” postulated by the model
⇒ Precise (normative!) theory for generation of dopamine firing patterns
⇒ Explains anticipatory dopaminergic responding, second order conditioning
⇒ Compelling account for the role of dopamine in classical conditioning: prediction error acts as signal driving learning in prediction areas
prediction error hypothesis of dopamine

at end of trial: $\delta_t = r_t - V_t$ (just like R-W)

$$V_t = \eta \sum_{i=1}^{t} (1 - \eta)^{t-i} r_i$$

Bayer & Glimcher (2005)
Global plan

• Reinforcement learning I:
  – prediction
  – classical conditioning
  – dopamine

• Reinforcement learning II:
  – dynamic programming; action selection
  – Pavlovian misbehaviour
  – vigor

• Chapter 9 of Theoretical Neuroscience
Action Selection

- Evolutionary specification
- Immediate reinforcement:
  - leg flexion
  - Thorndike puzzle box
  - pigeon; rat; human matching
- Delayed reinforcement:
  - these tasks
  - mazes
  - chess

active coping strategies evoked from the lPAG and the dIPAG

threat / confrontational defense
hypertension and tachycardia
extracranial vasodilation
hindlimb & renal vasoconstriction
non-opioid mediated analgesia

flight
hypertension and tachycardia
hindlimb vasodilation
extracranial & renal vasoconstriction
non-opioid mediated analgesia

passive coping strategies evoked from the sIPAG

quiescence
hyporeactivity
hypotension
bradycardia
opioid mediated analgesia

Bandler; Blanchard
Immediate Reinforcement

\[ p^L[r^L] \quad p^R[r^R] \]

- stochastic policy:

\[
P[L] = \frac{\exp(\beta m^L)}{\exp(\beta m^L) + \exp(\beta m^R)} = \sigma(\beta(m^L - m^R))
\]

- based on action values: \( m^L; m^R \)
Indirect Actor

use RW rule:

\[ m_L^L \rightarrow m_L^L + \epsilon \delta \quad \text{with} \quad \delta = r_L^L - m_L^L \]

\[ \langle r_L^L \rangle_{p_L} = 0.05; \langle r_R^R \rangle_{p_R} = 0.25 \quad \text{switch every 100 trials} \]
Direct Actor

\[ E(m) = P[L] \langle r^L \rangle + P[R] \langle r^R \rangle \]

\[ \frac{\partial P[L]}{\partial m^L} = \beta P[L] P[R] \quad \frac{\partial P[R]}{\partial m^R} = -\beta P[L] P[R] \]

\[ \frac{\partial E(m)}{\partial m^L} = \beta P[L] \left( \langle r^L \rangle - \left( P[L] \langle r^L \rangle + P[R] \langle r^R \rangle \right) \right) \]

\[ \frac{\partial E(m)}{\partial m^L} = \beta P[L] \left( \langle r^L \rangle - E(m) \right) \]

\[ \frac{\partial E(m)}{\partial m^L} \approx \beta \left( r^L - E(m) \right) \quad \text{if L is chosen} \]

\[ m^L - m^R \rightarrow (m^L - m^R) + \varepsilon (r^a - E(m))(L - R) \]
Direct Actor
Could we Tell?

• correlate past rewards, actions with present choice

• indirect actor (separate clocks):

\[
\log \frac{P_S[L]}{P_S[R]} = \beta (m_S^L - m_S^R) = \beta \epsilon \left( \sum_i (1-\epsilon)^i r_i^L - \sum_i (1-\epsilon)^i r_i^R \right)
\]

• direct actor (single clock):

\[
\begin{align*}
\log \frac{P_{K+1}[L]}{P_{K+1}[R]} &= \beta (m_{K+1}^L - m_{K+1}^R) \\
&= \beta \epsilon \sum_{k=0} (1-\epsilon)^k r_{K-k}^{a} (L_{K-k} - R_{K-k}) - \beta \epsilon \sum_{k=0} (1-\epsilon)^k v_{K-k} (L_{K-k} - R_{K-k})
\end{align*}
\]
Matching: Concurrent VI-VI

<table>
<thead>
<tr>
<th>Arming probabilities</th>
<th>Magnitudes</th>
<th>Number of blocks</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.24/0.04</td>
<td>0.35/0.35</td>
<td>16</td>
</tr>
<tr>
<td>0.21/0.07</td>
<td>0.35/0.35</td>
<td>12</td>
</tr>
<tr>
<td>0.07/0.21</td>
<td>0.35/0.35</td>
<td>11</td>
</tr>
<tr>
<td>0.04/0.24</td>
<td>0.35/0.35</td>
<td>19</td>
</tr>
<tr>
<td>0.24/0.04</td>
<td>0.4/0.4</td>
<td>5</td>
</tr>
<tr>
<td>0.21/0.07</td>
<td>0.4/0.4</td>
<td>8</td>
</tr>
<tr>
<td>0.07/0.21</td>
<td>0.4/0.4</td>
<td>8</td>
</tr>
<tr>
<td>0.04/0.24</td>
<td>0.4/0.4</td>
<td>4</td>
</tr>
<tr>
<td>0.24/0.04</td>
<td>0.45/0.45</td>
<td>7</td>
</tr>
<tr>
<td>0.21/0.07</td>
<td>0.45/0.45</td>
<td>10</td>
</tr>
<tr>
<td>0.07/0.21</td>
<td>0.45/0.45</td>
<td>5</td>
</tr>
<tr>
<td>0.04/0.24</td>
<td>0.45/0.45</td>
<td>10</td>
</tr>
</tbody>
</table>

Lau, Glimcher, Corrado, Sugrue, Newsome
Matching

\[
\log\left(\frac{p_{R,i}}{p_{L,i}}\right) = \sum_{j=1}^{\infty} \alpha_j (r_{R,i-j} - r_{L,i-j})
\]

- income not return
- approximately exponential in \( r \)
- alternation choice kernel
Action at a (Temporal) Distance

- learning an appropriate action at $x=1$:
  - depends on the actions at $x=2$ and $x=3$
  - gains no immediate feedback
- idea: use prediction as surrogate feedback
Action Selection

start with policy: \( P[L; x] = \sigma(m^L(x) - m^R(x)) \)

evaluate it: \( V(1), V(2), V(3) \)

improve it:
\[
\delta(t) = r_t + \ V(x_{t+1}) - V(x_t)
\]

thus choose R more frequently than L;C

\( \Delta m \propto \alpha \delta \)
Policy

\[ \delta > 0 \text{ if} \]
- value is too pessimistic \( \Rightarrow \Delta v \)
- action is better than average \( \Rightarrow \Delta P \)
actor/critic

dopamine signals to both motivational & motor striatum appear, surprisingly the same

suggestion: training both values & policies
Formally: Dynamic Programming

- \( V^*(x_t) = \max_u \{E[r(x_t, u) + V^*(x_{t+1})]\} \)
- \( = \max_u \{E[r(x_t, u) + \sum_y P(y|x_t, u) V^*(y)]\} \)
- \( Q^*(x, u) = E[r(x, u) + \sum_y P(y|x, u) V^*(y)] \)
- \( V^*(y) = \max_{u'} \{Q^*(y, u')\} \)

- policy iteration:
  - \( V^\pi(x) = \sum_u \pi(u|x) \{E[r(x_t, u) + \sum_y P(y|x_t, u) V^\pi(y)]\} \)
  - \( \pi'(x) = \arg\max_u \{Q^\pi(x, u)\} \)

- value iteration
  - \( V^{n+1}(x) = \max_u \{E[r(x_t, u) + \sum_y P(y|x_t, u) V^{n+1}(y)]\} \)
Variants: SARSA

\[ Q^* (1, C) = E[r_t + V^*(x_{t+1}) \mid x_t = 1, u_t = C] \]

\[ Q(1, C) \rightarrow Q(1, C) + \varepsilon (r_t + Q(2, u_{\text{actual}}) - Q(1, C)) \]

Morris et al, 2006
Variants: Q learning

\[
Q^*(1, C) = E\left[ r_t + V^* (x_{t+1}) \right| x_t = 1, u_t = C \]

\[
Q(1, C) \rightarrow Q(1, C) + \varepsilon \left( r_t + \max_u Q(2, u) - Q(1, C) \right)
\]

Roesch et al. 2007
Summary

• prediction learning
  – Bellman evaluation

• actor-critic
  – asynchronous policy iteration

• indirect method (Q learning)
  – asynchronous value iteration

\[
\begin{align*}
V^*(1) &= E[r_t + V^*(x_{t+1}) | x_t = 1] \\
Q^*(1, C) &= E[r_t + V^*(x_{t+1}) | x_t = 1, u_t = C]
\end{align*}
\]
Impulsivity & Hyperbolic Discounting

• humans (and animals) show impulsivity in:
  – diets
  – addiction
  – spending, …

• intertemporal conflict between short and long term choices
• often explained via hyperbolic discount functions

• alternative is Pavlovian imperative to an immediate reinforcer
• framing, trolley dilemmas, etc
Direct/Indirect Pathways

- **direct:** D1: GO; learn from DA increase
- **indirect:** D2: noGO; learn from DA decrease
- **hyperdirect (STN)** delay actions given strongly attractive choices
Frank

- DARPP-32: D1 effect
- DRD2: D2 effect
Three Decision Makers

- tree search
- position evaluation
- situation memory
Multiple Systems in RL

• model-based RL
  – build a forward model of the task, outcomes
  – search in the forward model (online DP)
    • optimal use of information
    • computationally ruinous

• cached-based RL
  – learn Q values, which summarize future worth
    • computationally trivial
    • bootstrap-based; so statistically inefficient

• learn both – select according to uncertainty
Animal Canary

- OFC; dlPFC; dorsomedial striatum; BLA?
- dosolateral striatum, amygdala
Two Systems:

a) Tree System

\[ S_0 \]
- Initial state
  - Press lever
  - Enter magazine

\[ S_1 \]
- Food delivered
  - Press lever
  - Enter magazine

\[ S_2 \]
- No reward
  - \( R = 0 \)

\[ S_3 \]
- Food obtained
  - \( R = 1 \)

b) Cache system

\[ S_0 \]
- Initial state
  - Press lever \( Q = 1 \)
  - Enter magazine \( Q = 0 \)

\[ S_1 \]
- Food delivered
  - Press lever \( Q = 0 \)
  - Enter magazine \( Q = 1 \)

\[ S_2 \]
- No reward
  - \( Q = 0 \)

\[ S_3 \]
- Food obtained
  - \( Q = 1 \)
Behavioural Effects

- Actions based on model will decline
- Actions based on model-free will persist
Effects of Learning

- distributional value iteration
  - (Bayesian Q learning)
- fixed additional uncertainty per step
One Outcome

shallow tree implies goal-directed control wins
Human Canary...

- if $a \rightarrow c$ and $c \rightarrow \£££$, then do more of $a$ or $b$?
  - MB: $b$
  - MF: $a$ (or even no effect)
Behaviour

- action values depend on both systems:
  \[ Q_{\text{tot}} (x, u) = Q_{\text{MF}} (x, u) + \beta Q_{\text{MB}} (x, u) \]
- expect that \( \beta \) will vary by subject (but be fixed)
Neural Prediction Errors (1→2)

• note that MB RL does not use this prediction error – training signal?
Neural Prediction Errors (1)

- right nucleus accumbens
Vigour

• Two components to choice:
  – what:
    • lever pressing
    • direction to run
    • meal to choose
  – when/how fast/how vigorous
    • free operant tasks

• real-valued DP
The model

Costs

Rewards

choose (action, τ) = (LP, τ₁)

choose (action, τ) = (LP, τ₂)

Costs

Rewards

Goal
The model

**Goal:** Choose actions and latencies to maximize the *average rate of return* (rewards minus costs per time)
Average Reward RL

Compute differential values of actions

\[ Q_{L,\tau}(x) = \text{Rewards} - \text{Costs} + \frac{\text{Future Returns}}{\tau} - \tau \rho \]

- steady state behavior (not learning dynamics)

\[ \rho = \text{average rewards minus costs, per unit time} \]

(Extension of Schwartz 1993)
Average Reward Cost/benefit Tradeoffs

1. Which action to take?
   ⇒ Choose action with largest expected reward minus cost

2. How fast to perform it?
   • slow → less costly (vigour cost)
   • slow → delays (all) rewards
   • net rate of rewards = cost of delay
     (opportunity cost of time)
   ⇒ Choose rate that balances vigour and opportunity costs

explains faster (irrelevant) actions under hunger, etc

masochism
Optimal response rates

Niv, Dayan, Joel, unpublished

Experimental data

Model simulation

Optimal response rates

Nose poke
Optimal response rates

Experimental data

Model simulation

Herrnstein 1961

More:
- # responses
- interval length
- amount of reward
- ratio vs. interval
- breaking point
- temporal structure
- etc.
Effects of motivation (in the model)

**RR25**

\[
Q(x, u, \tau) = p_r R - C_u - \frac{C_v}{\tau} + V(x') - \tau \cdot \bar{R}
\]

\[
\frac{\partial Q(x, u, \tau)}{\partial \tau} = \frac{C_v}{\tau^2} - \bar{R} = 0 \implies \tau_{opt} = \sqrt{\frac{C_v}{R_{opt}}}
\]

![Graph showing mean latency with low and high utility](image)
Effects of motivation (in the model)

- RR25
- Directing effect
- Energizing effect
- $U_R \uparrow 50\%$
- Mean latency
- Low utility
- High utility
Relation to Dopamine

Phasic dopamine firing = reward prediction error

What about tonic dopamine?

less ←________________________________→ more
Tonic dopamine = Average reward rate

1. explains pharmacological manipulations
2. dopamine control of vigour through BG pathways

- eating time confound
- context/state dependence (motivation & drugs?)
- less switching=perseveration

NB. phasic signal RPE for choice/value learning
Tonic dopamine hypothesis

...also explains effects of phasic dopamine on response times

Satoh and Kimura 2003

Ljungberg, Apicella and Schultz 1992
Sensory Decisions as Optimal Stopping

- consider listening to:

- decision: choose, or sample
Optimal Stopping

- equivalent of state $u=1$ is $u_1 = n_1$

$$P[L|u_1] = \sigma \left( d' \frac{n_1 - n_{\text{ave}}}{\sigma_n} \right) \text{ for } n_{\text{ave}} = \frac{1}{2}(n^L + n^R)$$

- and states $u=2,3$ is $u_2 = \frac{1}{2}(n_1 + n_2)$

$$P[L|u_2] = \sigma \left( 2d' (u_2 - n_{\text{ave}})/\sigma_n \right)$$

$\sigma = 2.5$

$\langle r^C \rangle = -0.1$
Transition Probabilities

\[ p[u_2 | u_1; C] = P[L | u_1] p[u_2 | L, u_1] + P[R | u_1] p[u_2 | R, u_1] \]

\[ = P[L | u_1] \mathcal{N}(2u_2 - u_1; n^L, \sigma_n^2) + P[R | u_1] \mathcal{N}(2u_2 - u_1; n^R, \sigma_n^2) \]

\[ m^C(u_1) = \langle r^C(u_1) + v(u_2) \rangle_{u_1} \]

\[ = r^C(u_1) + \int_{u_2} du_2 \ p[u_2 | u_1; C] \ \text{max}\{P[L | u_2], P[R | u_2]\} \]
Computational Neuromodulation

- **dopamine**
  - phasic: prediction error for reward
  - tonic: average reward (vigour)

- **serotonin**
  - phasic: prediction error for punishment?

- **acetylcholine**:
  - expected uncertainty?

- **norepinephrine**
  - unexpected uncertainty; neural interrupt?
Conditioning

prediction: of important events
control: in the light of those predictions

• Ethology
  – optimality
  – appropriateness

• Psychology
  – classical/operant conditioning

• Computation
  – dynamic progr.
  – Kalman filtering

• Algorithm
  – TD/delta rules
  – simple weights

• Neurobiology
  neuromodulators; amygdala; OFC
  nucleus accumbens; dorsal striatum
Markov Decision Process

class of stylized tasks with
states, actions & rewards

- at each timestep $t$ the world takes on
  state $s_t$ and delivers reward $r_t$, and the
  agent chooses an action $a_t$
Markov Decision Process

World: You are in state 34.
Your immediate reward is 3. You have 3 actions.
Robot: I’ll take action 2.

World: You are in state 77.
Your immediate reward is -7. You have 2 actions.
Robot: I’ll take action 1.

World: You’re in state 34 (again).
Your immediate reward is 3. You have 3 actions.
Markov Decision Process

Stochastic process defined by:
- reward function:
  \[ r_t \sim P(r_t \mid s_t) \]
- transition function:
  \[ s_t \sim P(s_{t+1} \mid s_t, a_t) \]
Markov Decision Process

Stochastic process defined by:
- reward function:
  \[ r_t \sim P(r_t | s_t) \]
- transition function:
  \[ s_t \sim P(s_{t+1} | s_t, a_t) \]

Markov property
- future conditionally independent of past, given \( s_t \)
The optimal policy

Definition: a policy such that at every state, its expected value is better than (or equal to) that of all other policies.

Theorem: For every MDP there exists (at least) one deterministic optimal policy.

→ by the way, why is the optimal policy just a mapping from states to actions? couldn’t you earn more reward by choosing a different action depending on last 2 states?
Pavlovian & Instrumental Conditioning

• Pavlovian
  – learning values and predictions
  – using TD error

• Instrumental
  – learning actions:
    • by reinforcement (leg flexion)
    • by (TD) critic
  – (actually different forms: goal directed & habitual)
Pavlovian-Instrumental Interactions

- **synergistic**
  - conditioned reinforcement
  - Pavlovian-instrumental transfer
    - Pavlovian cue predicts the instrumental outcome
    - behavioural inhibition to avoid aversive outcomes

- **neutral**
  - Pavlovian-instrumental transfer
    - Pavlovian cue predicts outcome with same motivational valence

- **opponent**
  - Pavlovian-instrumental transfer
    - Pavlovian cue predicts opposite motivational valence
    - negative automaintenance
-ve Automaintenance in Autoshaping

- **simple choice task**
  - N: nogo gives reward $r=1$
  - G: go gives reward $r=0$

- **learn three quantities**
  - average value
  - $Q$ value for N
  - $Q$ value for G

- **instrumental propensity is**

$$p(a(t) = N) = \frac{e^{\mu(q_N(t) - v(t))}}{e^{\mu(q_N(t) - v(t))} + e^{\mu(q_G(t) - v(t))}}$$

$$= \sigma(\mu(q_N(t) - q_G(t)))$$
-ve Automaintenance in Autoshaping

- **Pavlovian action**
  - assert: Pavlovian impetus towards G is $v(t)$
  - weight Pavlovian and instrumental advantages by $\omega$ – competitive reliability of Pavlov

- **new propensities**

  $N: \quad q_N(t) - v(t) \Rightarrow (1 - \omega)(q_N(t) - v(t))$

  $G: \quad q_G(t) - v(t) \Rightarrow (1 - \omega)(q_G(t) - v(t)) + \omega v(t)$

- **new action choice**

  $p(a(t) = N) = \sigma (\mu ((1 - \omega)(q_N(t) - q_G(t)) - \omega v(t)))$
-ve Automaintenance in Autoshaping

- basic –ve automaintenance effect ($\mu=5$)
- lines are theoretical asymptotes
- equilibrium probabilities of action