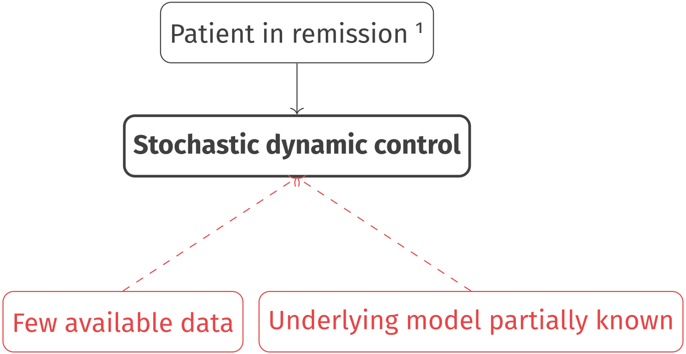
An example of medical treatment optimization under model uncertainty

Orlane Le Quellennec [1], Alice Cleynen [1,2], Benoîte de Saporta [1] and Régis Sabbadin [3]

[1]IMAG, Univ Montpellier, CNRS, Montpellier, France
[2]John Curtin School of Medical Research, The Australian National University, Canberra, ACT, Australia
[3]Univ Toulouse, INRAE-MIAT, Toulouse, France

June 30, 2023

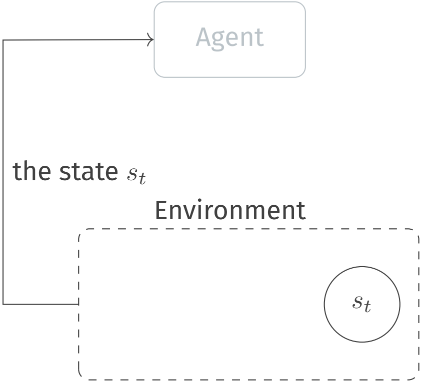
A medical context



How can these issues be addressed in a simplified problem ?

[1]Data from IUC Oncopole, Toulouse, and CRCT, Toulouse, France

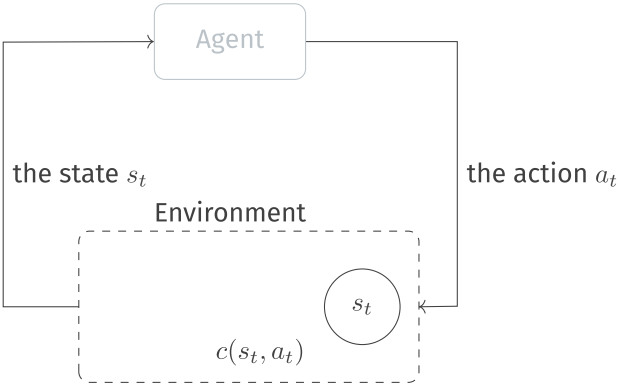
Markov Decision Process (MDP[2])



- $s \in \mathcal{S}$ the state space
- $a \in \mathcal{A}$ the action space
- $\mathcal{P}$ the transition matrix
- $c(s_t, a_t)$ the cost function

[2] ML Puterman (1994). "Finite-horizon Markov decision processes". In: Markov Decision Processes: Discrete Stochastic Dynamic Programming. New York: Wiley-Interscience, pp. 78–9.

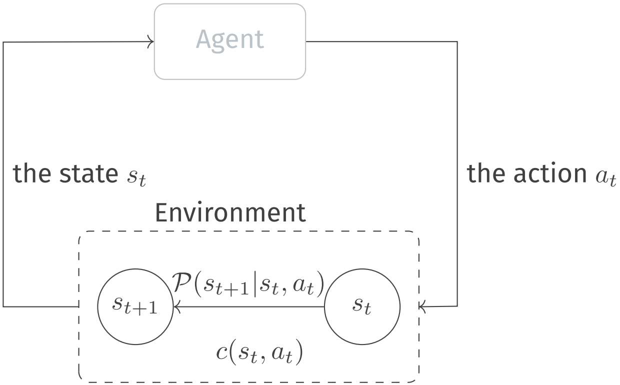
Markov Decision Process (MDP[2])



- $s \in \mathcal{S}$ the state space
- $a \in \mathcal{A}$ the action space
- $\mathcal{P}$ the transition matrix
- $c(s_t, a_t)$ the cost function

[2] ML Puterman (1994). "Finite-horizon Markov decision processes". In: Markov Decision Processes: Discrete Stochastic Dynamic Programming. New York: Wiley-Interscience, pp. 78–9.

Markov Decision Process (MDP[2])



- $s \in \mathcal{S}$ the state space
- $a \in \mathcal{A}$ the action space
- $\mathcal{P}$ the transition matrix
- $c(s_t, a_t)$ the cost function

[2] ML Puterman (1994). "Finite-horizon Markov decision processes". In: Markov Decision Processes: Discrete Stochastic Dynamic Programming. New York: Wiley-Interscience, pp. 78–9.

State transition

The transition matrix is partially known

Table: **Transition matrix when patient has no treatment ($a = \emptyset$).**

$s_t \backslash s_{t+1}$	$(0,0,0)$	$(1,0,1)$	$(1,0,2)$	$(1,1,1)$	$(1,1,2)$	$(1,2,1)$	$(1,2,2)$	$(1,3,1)$	$(1,3,2)$	$(2,4,0)$
$(0,0,0)$	$p^{\emptyset}_{(0,0,0)}$	$p^{\emptyset}_{(1,0,1)}$	$p^{\emptyset}_{(1,0,2)}$	0	0	0	0	0	0	0
$(1,0,1)$	0	0	0	**1**	0	0	0	0	0	0
$(1,0,2)$	0	0	0	0	0	0	**1**	0	0	0
$(1,1,1)$	0	0	0	0	0	**1**	0	0	0	0
$(1,1,2)$	0	0	0	0	0	0	0	0	**1**	0
$(1,2,1)$	0	0	0	0	0	0	0	**1**	0	0
$(1,2,2)$	0	0	0	0	0	0	0	0	0	**1**
$(1,3,1)$	0	0	0	0	0	0	0	0	0	**1**
$(1,3,2)$	0	0	0	0	0	0	0	0	0	**1**
$(2,4,0)$	0	0	0	0	0	0	0	0	0	**1**

State transition

The transition matrix is partially known

Table: **Transition matrix when patient has treatment ($a = \rho$).**

$s_t \backslash s_{t+1}$	$(0,0,0)$	$(1,0,1)$	$(1,0,2)$	$(1,1,1)$	$(1,1,2)$	$(1,2,1)$	$(1,2,2)$	$(1,3,1)$	$(1,3,2)$	$(2,4,0)$
$(0,0,0)$	$p^{\rho}_{(0,0,0)}$	$p^{\rho}_{(1,0,1)}$	$p^{\rho}_{(1,0,2)}$	0	0	0	0	0	0	0
$(1,0,1)$	**1**	0	0	0	0	0	0	0	0	0
$(1,0,2)$	**1**	0	0	0	0	0	0	0	0	0
$(1,1,1)$	**1**	0	0	0	0	0	0	0	0	0
$(1,1,2)$	**1**	0	0	0	0	0	0	0	0	0
$(1,2,1)$	0	0	0	**1**	0	0	0	0	0	0
$(1,2,2)$	0	0	0	0	**1**	0	0	0	0	0
$(1,3,1)$	0	0	0	0	0	0	0	**1**	0	0
$(1,3,2)$	0	0	0	0	0	0	0	0	**1**	0
$(2,4,0)$	0	0	0	0	0	0	0	0	0	**1**

Solving a MDP

Minimizing a cost

The list of costs:

- Treatment: 300
- Disease 1: 200
- Disease 2: 300
- Death: 1000

Policy π

Let $f : \mathcal{S} \to \mathcal{A}$ for all $s \in \mathcal{S}$ is a decision rule.
A sequence of decision rules $\pi = (f_0, f_1, \dots, f_{H-1})$ is a policy.

Policy cost and value function

$$J_H(\pi, s) = \mathbb{E}[\sum_{t=0}^{H-1} c(s_t, a_t) | \pi, s]$$
$$V_H(s) = \inf_{\pi \in \Pi} J_H(\pi, s)$$

Solving a MDP

Minimizing a cost

The list of costs:

- Treatment: 300
- Disease 1: 200
- Disease 2: 300
- Death: 1000

Policy π

Let $f : \mathcal{S} \to \mathcal{A}$ for all $s \in \mathcal{S}$ is a decision rule.
A sequence of decision rules $\pi = (f_0, f_1, \dots, f_{H-1})$ is a policy.

Policy cost and value function

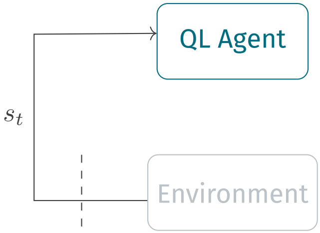
$$J_H(\pi, s) = \mathbb{E}[\sum_{t=0}^{H-1} c(s_t, a_t) | \pi, s]$$
$$V_H(s) = \inf_{\pi \in \Pi} J_H(\pi, s)$$

Optimization criterion

$$V^\star(s_t) = \min_{a \in \mathcal{A}} [c(s_t, a_t) + \sum_{s_{t+1} \in \mathcal{S}} \mathcal{P}(s_{t+1} | s_t, a_t) V^\star(s_{t+1})]$$

A model-free method

Q-learning[3,4] algorithm

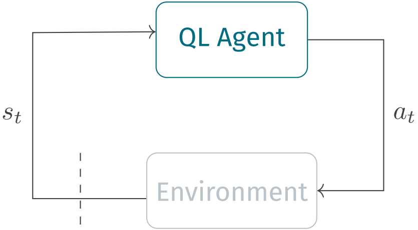


[3] Christopher J. C. H. Watkins and Peter Dayan (May 1992). "Q-learning". In: Mach. Learn. 8.3, pp. 279–292. ISSN: 1573-0565. DOI: 10.1007/BF00992698.

[4] VP Vivek and Dr. Shalabh Bhatnagar (Aug. 2022). "Finite Horizon Q-learning: Stability, Convergence, Simulations and an application on Smart Grids". In: arXiv:2110.15093v3. DOI: 10.48550/arXiv.2110.15093. eprint: 2110.15093v3.

A model-free method

Q-learning[3,4] algorithm



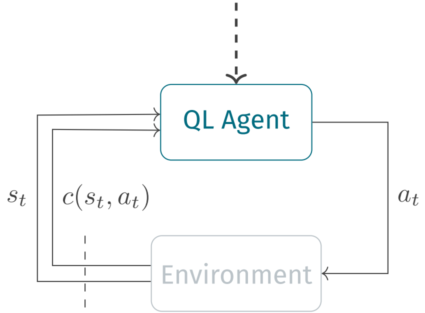
[3] Christopher J. C. H. Watkins and Peter Dayan (May 1992). "Q-learning". In: Mach. Learn. 8.3, pp. 279–292. ISSN: 1573-0565. DOI: 10.1007/BF00992698.

[4] VP Vivek and Dr. Shalabh Bhatnagar (Aug. 2022). "Finite Horizon Q-learning: Stability, Convergence, Simulations and an application on Smart Grids". In: arXiv:2110.15093v3. DOI: 10.48550/arXiv.2110.15093. eprint: 2110.15093v3.

A model-free method

Q-learning[3,4] algorithm

$$Q_n(s_t,a_t) = (1-\alpha)Q_{n-1}(s_t,a_t) + \alpha[c(s_t,a_t) + \min_{a_{t+1}\in\mathcal{A}} Q_{n-1}(s_{t+1},a_{t+1})]$$



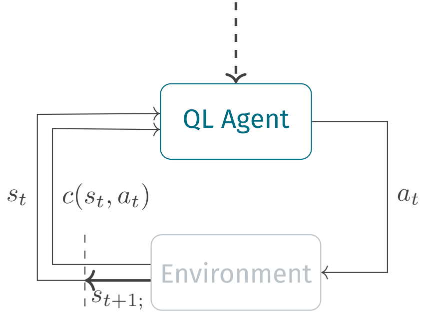
[3] Christopher J. C. H. Watkins and Peter Dayan (May 1992). “Q-learning”. In: Mach. Learn. 8.3, pp. 279–292. ISSN: 1573-0565. DOI: 10.1007/BF00992698.

[4] VP Vivek and Dr. Shalabh Bhatnagar (Aug. 2022). “Finite Horizon Q-learning: Stability, Convergence, Simulations and an application on Smart Grids”. In: arXiv:2110.15093v3. DOI: 10.48550/arXiv.2110.15093. eprint: 2110.15093v3.

A model-free method

Q-learning[3,4] algorithm

$$Q_n(s_t, a_t) = (1-\alpha)Q_{n-1}(s_t, a_t) + \alpha[c(s_t, a_t) + \min_{a_{t+1}\in\mathcal{A}} Q_{n-1}(s_{t+1}, a_{t+1})]$$



[3] Christopher J. C. H. Watkins and Peter Dayan (May 1992). "Q-learning". In: Mach. Learn. 8.3, pp. 279–292. ISSN: 1573-0565. DOI: 10.1007/BF00992698.

[4] VP Vivek and Dr. Shalabh Bhatnagar (Aug. 2022). "Finite Horizon Q-learning: Stability, Convergence, Simulations and an application on Smart Grids". In: arXiv:2110.15093v3. DOI: 10.48550/arXiv.2110.15093. eprint: 2110.15093v3.

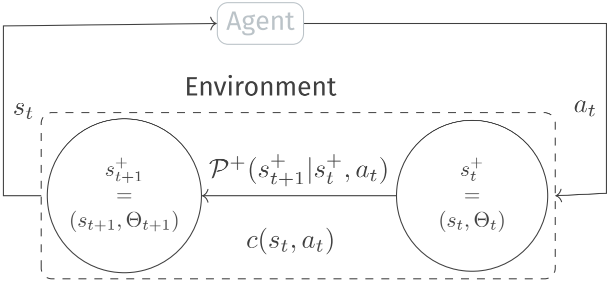
A bayesian approach

$s_t \backslash s_{t+1}$	$(0,0,0)$	$(1,0,1)$	$(1,0,2)$	$(1,1,1)$	$(1,1,2)$	$(1,2,1)$	$(1,2,2)$	$(1,3,1)$	$(1,3,2)$	$(2,4,0)$
$(0,0,0)$	$p^{\emptyset}_{(0,0,0)}$	$p^{\emptyset}_{(1,0,1)}$	$p^{\emptyset}_{(1,0,2)}$	0	0	0	0	0	0	0

Remark:

- $P(.|s=(0,0,0), a=\emptyset) \sim \mathcal{M}(p^{\emptyset}_{(0,0,0)}, p^{\emptyset}_{(1,0,1)}, p^{\emptyset}_{(1,0,2)})$
- Conjugate distribution : $f(p^{\emptyset}|\Theta^{\emptyset}) \sim \mathcal{D}(\theta^{\emptyset}_{(0,0,0)}, \theta^{\emptyset}_{(1,0,1)}, \theta^{\emptyset}_{(1,0,2)})$

Bayes-Adaptive Markov Decision Process (BAMDP[5])

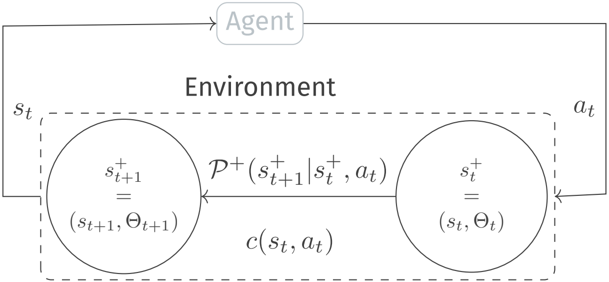


- $s^+ \in \mathcal{S}^+$ the hyper-state space
- $\mathcal{P}^+$ the transition matrix
- $\Theta_{t+1} = \Theta_t + \Delta^{a_t}_{s_{t+1}}$, with

$$\Delta^{a_t}_{s_{t+1}} = \begin{cases} 1 & \text{if } (s = (0, 0, 0), a_t, s_{t+1}), \\ 0 & \text{else.} \end{cases}$$

[5] Michael O'Gordon Duff (2002). "Optimal learning: Computational procedures for Bayes -adaptive Markov decision processes". PhD thesis. University of Massachusetts Amherst.

Bayes-Adaptive Markov Decision Process (BAMDP[5])



- $s^+ \in \mathcal{S}^+$ the hyper-state space
- $\mathcal{P}^+$ the transition matrix
- $\Theta_{t+1} = \Theta_t + \Delta^{a_t}_{s_{t+1}}$, with

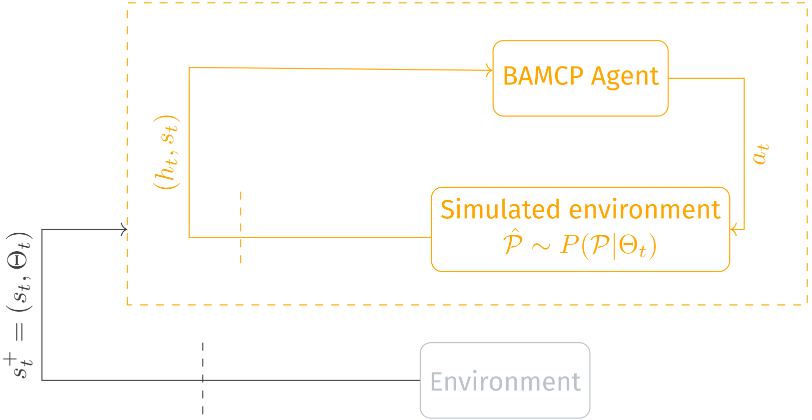
$$\Delta^{a_t}_{s_{t+1}} = \begin{cases} 1 & \text{if } (s = (0, 0, 0), a_t, s_{t+1}), \\ 0 & \text{else.} \end{cases}$$

Optimization criterion

$$V^\star(s_t, \Theta_t) = \min_{a \in \mathcal{A}} [c(s_t, a_t) + \sum_{s^+_{t+1} \in \mathcal{S}^+} \mathcal{P}^+(s^+_{t+1} | s^+_t, a_t) V^\star(s_{t+1}, \Theta_{t+1})]$$

[5] Michael O'Gordon Duff (2002). "Optimal learning: Computational procedures for Bayes -adaptive Markov decision processes". PhD thesis. University of Massachusetts Amherst.

with $h_t = (s_0, a_0, s_1, \dots, s_{t-1}, a_{t-1})$,

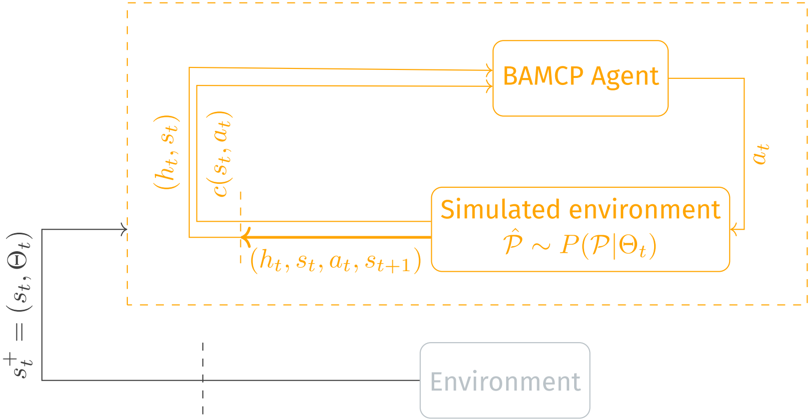


[6] Arthur Guez, David Silver, and Peter Dayan (2012). "Efficient Bayes-Adaptive Reinforcement Learning using Sample-Based Search". In: Advances in Neural Information Processing Systems. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc.

A model-based method

Bayes-Adaptive Monte-Carlo Planning (BAMCP[6])

with $h_t = (s_0, a_0, s_1, \ldots, s_{t-1}, a_{t-1})$,

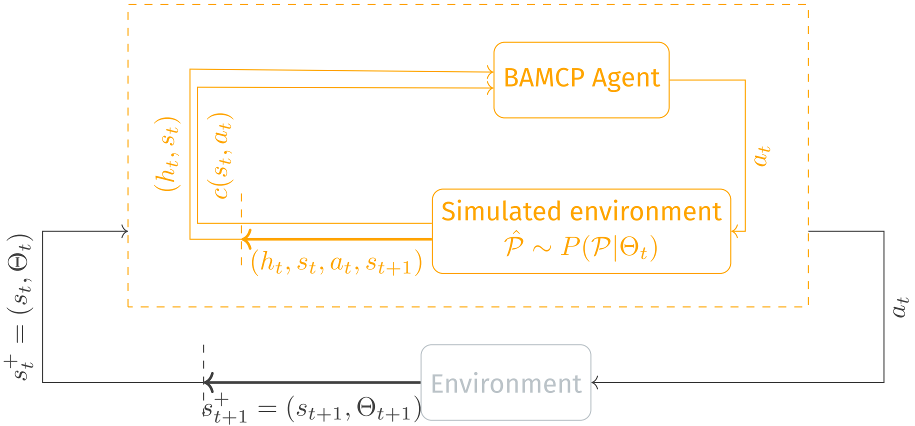


[6] Arthur Guez, David Silver, and Peter Dayan (2012). "Efficient Bayes-Adaptive Reinforcement Learning using Sample-Based Search". In: Advances in Neural Information Processing Systems. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc.

A model-based method

Bayes-Adaptive Monte-Carlo Planning (BAMCP[6])

with $h_t = (s_0, a_0, s_1, \ldots, s_{t-1}, a_{t-1})$,



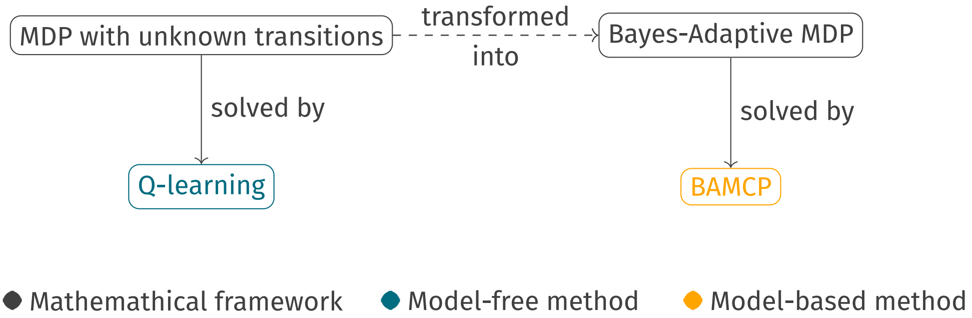
[6] Arthur Guez, David Silver, and Peter Dayan (2012). "Efficient Bayes-Adaptive Reinforcement Learning using Sample-Based Search". In: Advances in Neural Information Processing Systems. Ed. by F. Pereira et al. Vol. 25. Curran Associates, Inc.

Results

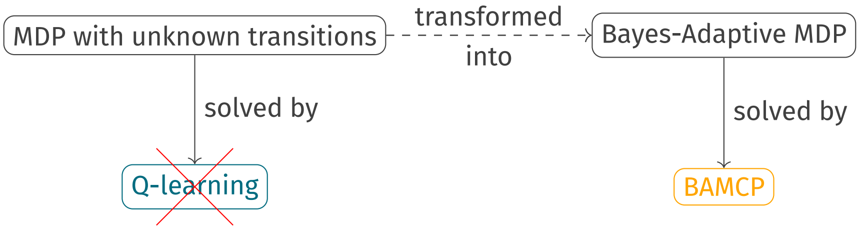
The optimal policy exact cost: 888.89

Simulated patients	Q-learning		BAMCP	
	Cost	Time	Cost	Time
10^2	1427.06 ± 1.05	0.07 sec	1302.58 ± 1.32	2.07 hours
10^3	936.96 ± 0.70	2.48 min	1297.64 ± 1.32	2.22 hours
10^4	936.93 ± 0.70	4.17 min	NC	4 days
10^6	891.6 ± 0.68	10.21 min	NC	1.5 years

Conclusion


MDP with unknown transitions
transformed
into
Bayes-Adaptive MDP
solved by
solved by
Q-learning
BAMCP
Mathemathical framework
Model-free method
Model-based method

Conclusion



- Mathemathical framework
- Model-free method
- Model-based method

Perspectives

MDP model	$\rightarrow$	PDMP[7] model
Finite state space	$\rightarrow$	Continuous state space
Markovian	$\rightarrow$	Semi-Markovian
Complete observations	$\rightarrow$	Hidden observations

Unlike model-free methods and deep reinforcement learning, **bayesian approaches** do not require as much interaction with the environment.

[7] Mark H. A. Davis (1984). "Piecewise-Deterministic Markov Processes: A General Class of Non-Diffusion Stochastic Models". In: Journal of the Royal Statistical Society Series B (Methodological) 46, pp. 353–376.