Index

Page numbers in *italics* are recommended to be consulted first. Page numbers in **bold** contain boxed algorithms.

- k-armed bandits, 25–45

- absorbing state, 57
- access-control queuing example, 256
- action preferences, *322*, 329, 336, 455
  - in bandit problems, *37*, 42
- action-value function, *see* value function, action
- action-value methods, 321
  - for bandit problems, 27
- actor–critic, 21, 239, 321, *331–332*, 338, 406
  - one-step (episodic), **332**
  - with eligibility traces (episodic), **332**
  - with eligibility traces (continuing), **333**
  - neural, 395–415
- addiction, 409–410
- advantage actor–critic methods, 338
- afterstates, *137*, 140, 181, 182, 191, 424, 430
- agent–environment interface, 47–57, 467
- all-actions algorithm, 326
- AlphaGo, AlphaGo Zero, AlphaZero, 441–450
- Andreae, John, *17*, 21, 69, 89
- ANN, *see* artificial neural networks
- applications and case studies, 421–457
- approximate dynamic programming, 15
- artificial intelligence, xvii, 1, 472, *475–478*
- artificial neural networks, *223–228*, 238–239, 395–398, 423, 430, 436–450, 472
- associative reinforcement learning, *45*, 418
- associative search, 41
- asynchronous dynamic programming, *85*, 88
- Atari video game play, 436–441
- auxiliary tasks, *460–461*, 468, 474
- average reward setting, *249–255*, 258, 464
- averagers, 264

- backgammon, 11, 21, 182, 184, *421–426*
- backpropagation, 21, *225–227*, 239, 407, 424, 436, 439
- backup diagram, *60*, 139
  - for dynamic programming, 59, 61, 64, *172*
  - for Monte Carlo methods, 94
  - for Q-learning, 134
  - for TD(0), 121
  - for Sarsa, 129
  - for Expected Sarsa, 134
  - for Sarsa(λ), 304
  - for TD(λ), 289
  - for Q(λ), 313
  - for Tree Backup(λ), 314
  - for truncated TD(λ), 296
  - for n-step $Q(\sigma)$, 155
  - for n-step Expected Sarsa, 146
  - for n-step Sarsa, 146
  - for n-step TD, 142
  - for n-step Tree Backup, 152
  - for Samuel's Checker Player, 428
  - compound, 288
  - half backups, 62
- backward view of eligibility traces, 288, *293*
- Baird's counterexample, *261–264*, 280, 283, 285
- bandit algorithm, simple, **32**
- bandit problems, 25–45
- basal ganglia, 386
- baseline, 37–40, *329*, 330, 331, 338
- behavior policy, *103*, 110, *see* off-policy learning
- Bellman equation, 14
  - for v_π, 59
  - for q_π, 78
  - for optimal value functions: v_* and q_*, 63
  - differential, 250
  - for options, 463
- Bellman error, *268*, 270, 272, 273
  - learnability of, 274–277
  - vector, 267–269
- Bellman operator, 267–269, 286
- Bellman residual, 286, *see* Bellman error
- Bellman, Richard, *14*, 71, 89, 241
- binary features, *215*, 222, 245, 304, 305
- bioreactor example, 51
- blackjack example, *93–94*, *99*, 106
- blocking maze example, 166
- bootstrapping, *89*, 189, 308, 331
  - n-step, *141–158*, 255
  - and dynamic programming, 89
  - and function approximation, 208, *264–274*

  - and Monte Carlo methods, 95
  - and stability, 263–265
  - and TD learning, 120
  - assessment of, 124–128, 248, *264*, 291, 318
  - in psychology, 345, 349, *354*, 355
  - parameter (λ or n), *291*, 307, 399
- BOXES, *18*, 71, 237
- branching factor, *173–177*, 422
- breakfast example, *5*, 22
- bucket-brigade algorithm, 19, 21, *139*

- catastrophic interference, 472
- certainty-equivalence estimate, 128
- chess, 4, 20, 54, 182, 450
- classical conditioning, 20, *343–357*
  - blocking, 371
    - and higher-order conditioning, 345–355
  - delay and trace conditioning, 344
  - Rescorla-Wagner model, 346–349
  - TD model, 349–357
- classifier systems, *19*, 21
- cliff walking example, *132*, 133
- CMAC, *see* tile coding
- coarse coding, *215–220*, 237
- cognitive maps, 363–364
- collective reinforcement learning, 404–407
- complex backups, *see* compound update
- compound stimulus, 345, *346–356*, 371, 382
- compound update/backup, 288, 319
- conditioned/unconditioned stimulus, conditioned response (CS/US, CR), 343
- constant-α MC, 120
- contextual bandits, 41
- continuing tasks, *54*, 57, 70, 124, 249, 294
- continuous action, 73, 244, *335–336*
- continuous state, *73*, 223, 237
- continuous time, *11*, 71
- control and prediction, 342
- control theory, *4*, 70
- control variates, *150–152*, 155, 281
  - and eligibility traces, 309–312
- credit assignment, 11, *17*, 19, 47, 294, 401
  - in psychology, 346, *361*
  - structural, *385*, 405, 407
- critic, *18*, 238, 346, 417, *see* actor–critic
- cumulant, 459
- curiosity, 474
- curse of dimensionality, 4, *14*, 221, 231
- cybernetics, xvii, 477

- deadly triad, 264
- deep learning, 11, *223*, 441, 472–474, 480
- deep reinforcement learning, 236
- deep residual learning, 226
- delayed reinforcement, 361–363
- delayed reward, *2*, 47, 249
- dimensions of reinforcement learning methods, 189–191
- direct and indirect RL, *162*, 164, 192
- discounting, *55*, 199, 243, 249, 282, 324, 328, 427, 459
  - in pole balancing, 56
  - state dependent, 307
  - deprecated, *253*, 256
- distribution models, *159*, 185
- dopamine, 377, *381–387*, 413–419
  - and addiction, 409–410
- double learning, *134–136*, 140
- DP, *see* dynamic programming
- driving-home example, 122–123
- Dyna architecture, **164**, 161–170
- dynamic programming, 13–15, *73–90*, 174, 262
  - and artificial intelligence, 89
  - and function approximation, 241
  - and options, 463
  - and the deadly triad, 264
  - computational efficiency of, 87

- eligibility traces, *287–320*, 350, 362, 398–403
  - accumulating, *301*, 306, 310
  - replacing, *301*, 306
  - dutch, 300–303
  - contingent/non-contingent, 399–403, *411*
  - off-policy, 309–316
  - with state-dependent λ and γ, 309–316
- Emphatic-TD methods, 234–235, *315*
  - off-policy, 281–282
- environment, 47–57
- episodes, episodic tasks, 11, *54–57*, 91
- error reduction property, *144*, 288
- evaluative feedback, 17, *25*, 47
- evolution, 7, 359, 374, 471
- evolutionary methods, *7*, 8–10, 19
- expected approximate value, *148*, 155
- Expected Sarsa, 133, *see also* Sarsa, Expected
- expected update, *75*, 172–181, 189
- experience replay, 440–441
- explore/exploit dilemma, *3*, 103, 472
- exploring starts, *96*, 98–100, 178

feature construction, 210–223
final time step (T), 54
Fourier basis, 211–214
function approximation, 195–200

gambler's example, 84
game theory, 19
gazelle calf example, 5
general value functions (GVFs), *459–463*, 474
generalized policy iteration (GPI), *86–87*, 92, 97, 138, 189
genetic algorithms, 19
Gittins index, 43
gliding/soaring case study, 453–457
goal, *see* reward signal
golf example, *61*, 63, 66
gradient, 201
gradient descent, *see* stochastic gradient descent
Gradient-TD methods, *278–281*, 314–315
greedy or ε-greedy
  as exploiting, 26–28
  as shortsighted, 64
  ε-greedy policies, 100
gridworld examples, 60, 65, 76, 147
  cliff walking, 132
  Dyna blocking maze, 166
  Dyna maze, 164
  Dyna shortcut maze, 167
  windy, 130, 131

habitual and goal-directed control, 364–368
hedonistic neurons, 402–404
heuristic search, *181–183*, 190
  as sequences of backups, 183
  in Samuel's checkers player, 426
  in TD-Gammon, 425
history of reinforcement learning, 13–21
Holland, John, *19*, 21, 44, 139, 241
Hull, Clark, 16, 359, 360, *362–363*

importance sampling, *103–117*, 151, 257
  ratio, *104*, 148, 258
  weighted and ordinary, *105*, 106
  and eligibility traces, 309–312
  and infinite variance, 106
  discounting aware, 112–113
  incremental implementation, 109
  per-decision, 114–115
  n-step, 148–156
incremental implementation
  of averages, 30–33
  of weighted averages, 109
instrumental conditioning, 357–361, *see also* Law of Effect
  and motivation, 360–361
  Thorndike's puzzle boxes, 358
interest and emphasis, *234–235*, 282, 316
inverse reinforcement learning, 470

Jack's car rental example, *81–82*, 137, 210

kernel-based function approximation, 232–233
Klopf, A. Harry, *xv*, xvii, 19–21, 402–404, 411

latent learning, 192, *363*, 366
Law of Effect, *15–16*, 45, 342, 358–361, 417
learning automata, 18
Least Mean Square (LMS) algorithm, 279, *301*
Least-Squares TD (LSTD), 228–229
linear function approx., *204–209*, 266–269
linear programming, 87, 90
local and global optima, 200

Markov decision process (MDP), 2, 14, *47–71*
Markov property, *49*, 115, 465–468
Markov reward process (MRP), 125
maximization bias, 134–136
maximum-likelihood estimate, 128
MC, *see* Monte Carlo methods
Mean Squared
  Bellman Error, $\overline{\text{BE}}$, 268
  Projected Bellman Error, $\overline{\text{PBE}}$, 269
  Return Error, $\overline{\text{RE}}$, 275
  TD Error, $\overline{\text{TDE}}$, 270
  Value Error, $\overline{\text{VE}}$, 199–200
memory-based function approx., 230–232
Michie, Donald, *17*, 71, 116
Minsky, Marvin, *16*, 17, 20, 89
model of the environment, 7, *159*
model-based and model-free methods, 7, *159*
  in animal learning, 363–368
model-based reinforcement learning, 159–193
  in neuroscience, 407–409
Monte Carlo methods, 91–117
  first- and every-visit MC, 92
  first-visit MC control, **101**
  first-visit MC prediction, **92**

    gradient method for v_π, **202**
    Monte Carlo ES (Exploring Starts), **99**
    off-policy control, **111**, 110–112
    off-policy prediction, 103–109, **110**
Monte Carlo Tree Search (MCTS), 185–188
motivation, 360–361
mountain car example, *244–248*, 305, 306
multi-armed bandits, 25–45

n-step methods, *141–158*
    $Q(\sigma)$, **156**
    Sarsa, **147**, **247**
        differential, **255**
        off-policy, **149**
    TD, **144**
    Tree Backup, **154**
    truncated λ-return, 295
naughts and crosses, *see* tic-tac-toe
neural networks, *see* artificial neural networks
neurodynamic programming, 15
neuroeconomics, *413*, 419
neuroscience, 4, 21, *377–419*
nonstationarity, 30, *32–36*, 41, 44, 255
    inherent, 91, *198*
notation, xiii, *xix*

observations, 464
off-policy methods, 257–286
    vs on-policy methods, 100, 103
    Monte Carlo, 103–115
    Q-learning, **131**
    Expected Sarsa, 133–134
    n-step, 148–156
    n-step $Q(\sigma)$, **156**
    n-step Sarsa, **149**
    n-step Tree Backup, **154**
    and eligibility traces, 309–316
    Emphatic-TD(λ), 315
    GQ(λ), 315
    GTD(λ), 314
    HTD(λ), 315
    Q(λ), 312–314
    Tree Backup(λ), 312–314
    reducing variance, 283–284
on-policy distribution, *175*, 199, 208, 258, 262, 281, 282
    vs uniform distribution, 176
on-policy methods, 100
    actor–critic, **332, 333**
    approximate
        control, **244**, **247**, **251**, **255**
        prediction, **202, 203, 209**
    Monte Carlo, **101**, 100–103, **328**, **330**
    n-step, **144**, **147**
    Sarsa, **130**, 129–131
    TD(0), **120**, 119–128
    with eligibility traces, **293**, **300**, **305**, **307**
operant conditioning, *see* instrumental learning
optimal control, 2, *14–15*, 21
optimistic initial values, *34–35*, 192
optimizing memory control, 432–436
options, 461–464
    models of, 462

pain and pleasure, *6*, 16, 413
Partially Observable MDPs (POMDPs), *466*
Pavlov, Ivan, 16, *343–345*, 362
Pavlovian
    conditioning, *see* classical conditioning
    control, 343, *371*, 373, 479
personalizing web services, 450–453
planning, 3, 5, 7, 11, 138, *159–193*
    in psychology, 363, 364, 366
    with learned models, *161–168*, 473
    with options, 461, *463*
policy, 6, 41, *58*
    hierarchical, 462
    soft and ε-soft, 100–103, 110
policy approximation, 321–324
policy evaluation, 74–76, *see also* prediction
    iterative, **75**
policy gradient methods, 321–338
    REINFORCE, **328**, **330**
    actor–critic, **332, 333**
policy gradient theorem, 324–326
    proof, episodic case, 325
    proof, continuing case, 334
policy improvement, 76–80
    theorem, *78*, 101
policy iteration, 14, **80**, *80–82*
polynomial basis, 210–211
prediction, *74–76*, *see also* policy evaluation
    and control, 342
    Monte Carlo, 92–97
    off-policy, 103–108
    TD, 119–126
    with approximation, 197–241
prior knowledge, 11, 34, 54, 137, 236, 324, 470

- prioritized sweeping, **170**, 168–171
- projected Bellman error, 285
  - vector, 267, *269*
- proximal TD methods, 286
- pseudo termination, 282, *308*
- psychology, 4, 13, 18, 20, *341–376*

- Q(λ), Watkins's, 312–314
- Q-function, *see* action-value function
- Q-learning, 21, **131**, *131–135*
  - double, **136**
- Q-planning, **161**
- $Q(\sigma)$, **156**, 154–156
- queuing example, 251

- R-learning, 256
- racetrack exercise, 111
- radial basis functions (RBFs), 221–222
- random walk, 95
  - 5-state, *125*, 126, 127
  - 19-state, *144*, 291
    - TD(λ) results on, *294, 295*, 299
  - 1000-state, *203–209*, 217, 218
    - Fourier and polynomial bases, 214
- real-time dynamic programming, 177–180
- recycling robot example, 52
- REINFORCE, **328**, 326–331
  - with baseline, **330**
- reinforcement learning, 1–21
- reinforcement signal, *380*
- representation learning, 473
- residual-gradient algorithm, *272–274*, 277
  - naive, 270, 271
- return, 54–57
  - n-step, 143
    - for $Q(\sigma)$, 155
    - for action values, 146
    - for Expected Sarsa, 148
    - for Tree Backup, 153
    - with control variates, 150, 151
    - with function approximation, 209
  - differential, 250, 255, 334
  - flat partial, 113
  - with state-dependent termination, 308
  - λ-return, 288–291
    - truncated, 296
- reward prediction error hypothesis, *381–383*, 387–395
- reward signal, 1, 6, *48*, *53*, 361, 380, 383, 397
  - and reinforcement, 373–375, 380–381
  - design of, *469–472*, 476
  - intrinsic, 474
  - sparse, 469–470
- rod maneuvering example, 171
- rollout algorithms, 183–185
- root mean-squared (RMS) error, 125

- safety, 434, *475–478*
- sample and expected updates, *121*, 170–174
- sample or simulation model, 115
- sample-average method, 27
- Samuel's checkers player, 20, 241, *426–429*
- Sarsa, **130**, *129–131*, **244**
  - vs Q-learning, 132
  - differential, one-step, **251**
  - Expected, *133–134*, 140
    - n-step, 148
    - n-step off-policy, 150
    - double, 136
  - n-step, **147**, *145–148*, **247**
    - differential, **255**
    - off-policy, **149**
- Sarsa(λ), **305**, 303–307
  - true online, **307**
- Schultz, Wolfram, 387–395, *410*
- search control, 163
- secondary reinforcement, *20*, 346, 354, 369
- selective bootstrap adaptation, 239
- semi-gradient methods, 202, *258–259*
- SGD, *see* stochastic gradient descent
- Shannon, Claude, 16, *20*, 70, 71, 426
- shaping, *360*, 470
- Skinner, B. F., *359–360*, 375, 470, 480
- soap bubble example, 95
- soft and ε-soft policies, 100–103, 110
- soft-max, *322–323*, 329, 336, 400, 445, 455
  - for bandits, *37*, 45
- spike-timing-dependent plasticity (STDP), 401
- state, *7*, 48, 49
  - kth-order history approach, 468
  - and observations, 464–468
    - Markov property, 465–468
  - belief, 466
  - latent, 466
  - observable operator models (OOMs), 467
  - partially observable MDPs, 14, 466
  - predictive state representations, 466
  - state-update function, 466

state aggregation, 203–204
state-update function, 466
step-size parameter, 10, *31–33*, 120, 125, 126
  automatic adaptation, 238
  in DQN, 439, 440
  in psychological models, 347, 348
  selecting manually, 222–223
  with coarse coding, 216
  with Fourier features, 213
  with tile coding, *217*, 223
stochastic approx. convergence conditions, 33
stochastic gradient descent (SGD), 200–204
  in the Bellman error, 269–277
strong and weak methods, 4
supervised learning, xvii, *2*, 16–19, 198
sweeps, *75*, 160, *see also* prioritized sweeping
synaptic plasticity, 379
  Hebbian, 400
  two-factor and three factor, 400
system identification, 364

tabular solution methods, 23
target
  policy, *103*, 110
  of update, 31, 143, *198*
TD, *see* temporal-difference learning
TD error, 121
  n-step, 255
  differential, 250
  with function approximation, 270
TD(λ), **293**, 292–295
  truncated, 295–297
  true online, **300**, 299–301
TD-Gammon, 21, *421–426*
temporal abstraction, 461–464
temporal-difference learning, 10, *119–140*
  history of, 20–21
  advantages of, 124–126
  optimality of, 126–128
  TD(0), **120**, **203**
  TD(1), 294
  TD(λ), **293**, 292–295
    true online, **300**, 299–301
  λ-return methods
    offline, 290
    online, 297–299
  n-step, **144**, *141–158*, **209**
termination function, 307, 459
Thompson sampling, *43*, 45
Thorndike, Edward, *see* Law of Effect
tic-tac-toe, *8–13*, 17, 137
tile coding, *217–221*, 223, 238, 246, 434, 435
Tolman, Edward, *364*, 408
trace-decay parameter (λ), 287, *289*, 290, 292
  state dependent, 307
trajectory sampling, 174–177
transition probabilities, 49
Tree Backup
  n-step, *152–153*, **154**
  Tree-Backup(λ), 312–314
trial-and-error, 2, 7, *15–21*, 403, 404, *see also* instrumental conditioning
true online TD(λ), **300**, 299–301
Tsitsiklis and Van Roy's Counterexample, 263

undiscounted continuing tasks, *see* average reward setting
unsupervised learning, *2*, 226

value, *6*, 26, 47
value function, 6, *58–67*
  for a given policy: v_π and q_π, 58
  for an optimal policy: v_* and q_*, 62
  action, *58*, 63, 64, 71, 129, 131
  approximate action values: $\hat{q}(s, a, \mathbf{w})$, 243
  approximate state values: $\hat{v}(s,\mathbf{w})$, 197
  differential, 243
  vs evolutionary methods, 10
value iteration, **83**, 82–85
value-function approximation, 198

Watkins, Chris, *14*, 21, 89, 320
Watson (*Jeopardy!* player), 429–432
Werbos, Paul, 14, 21, *69*, 89, 139, 238
Witten, Ian, *21*, 69

Adaptive Computation and Machine Learning

Francis Bach, Editor

Bioinformatics: The Machine Learning Approach, Pierre Baldi and Søren Brunak

Reinforcement Learning: An Introduction, Richard S. Sutton and Andrew G. Barto

Graphical Models for Machine Learning and Digital Communication, Brendan J. Frey

Learning in Graphical Models, Michael I. Jordan

Causation, Prediction, and Search, second edition, Peter Spirtes, Clark Glymour, and Richard Scheines

Principles of Data Mining, David Hand, Heikki Mannila, and Padhraic Smyth

Bioinformatics: The Machine Learning Approach, second edition, Pierre Baldi and Søren Brunak

Learning Kernel Classifiers: Theory and Algorithms, Ralf Herbrich

Learning with Kernels: Support Vector Machines, Regularization, Optimization, and Beyond, Bernhard Schölkopf and Alexander J. Smola

Introduction to Machine Learning, Ethem Alpaydin

Gaussian Processes for Machine Learning, Carl Edward Rasmussen and Christopher K.I. Williams

Semi-Supervised Learning, Olivier Chapelle, Bernhard Schölkopf, and Alexander Zien, Eds.

The Minimum Description Length Principle, Peter D. Grünwald

Introduction to Statistical Relational Learning, Lise Getoor and Ben Taskar, Eds.

Probabilistic Graphical Models: Principles and Techniques, Daphne Koller and Nir Friedman

Introduction to Machine Learning, second edition, Ethem Alpaydin

Machine Learning in Non-Stationary Environments: Introduction to Covariate Shift Adaptation, Masashi Sugiyama and Motoaki Kawanabe

Boosting: Foundations and Algorithms, Robert E. Schapire and Yoav Freund

Machine Learning: A Probabilistic Perspective, Kevin P. Murphy

Foundations of Machine Learning, Mehryar Mohri, Afshin Rostami, and Ameet Talwalker

Introduction to Machine Learning, third edition, Ethem Alpaydin

Deep Learning, Ian Goodfellow, Yoshua Bengio, and Aaron Courville

Elements of Causal Inference, Jonas Peters, Dominik Janzing, and Bernhard Schölkopf

Machine Learning for Data Streams, with Practical Examples in MOA, Albert Bifet, Ricard Gavaldà, Geoffrey Holmes, Bernhard Pfahringer