2024/07/20
理論談話会 #16

Human-level control through deep reinforcement learning

Volodymyr M, Koray K, David S, Andrei A. R, Joel V, Marc G. B, Alex G, Martin R, Andreas K. F, Georg O, Stig P, Charles B, Amir S, Ioannis A, Helen K, Dharshan K, Daan W, Shane L & Demis H

Nature 518,
p529-533 , 25 February 2015

M1 手代木祐可子

Abstract

強化学習を現実の複雑さに近い状況で成功させるのは難しい

先行研究 特徴を手動で作成できる範囲内 or 低次元の状態空間での領域に限定されていた

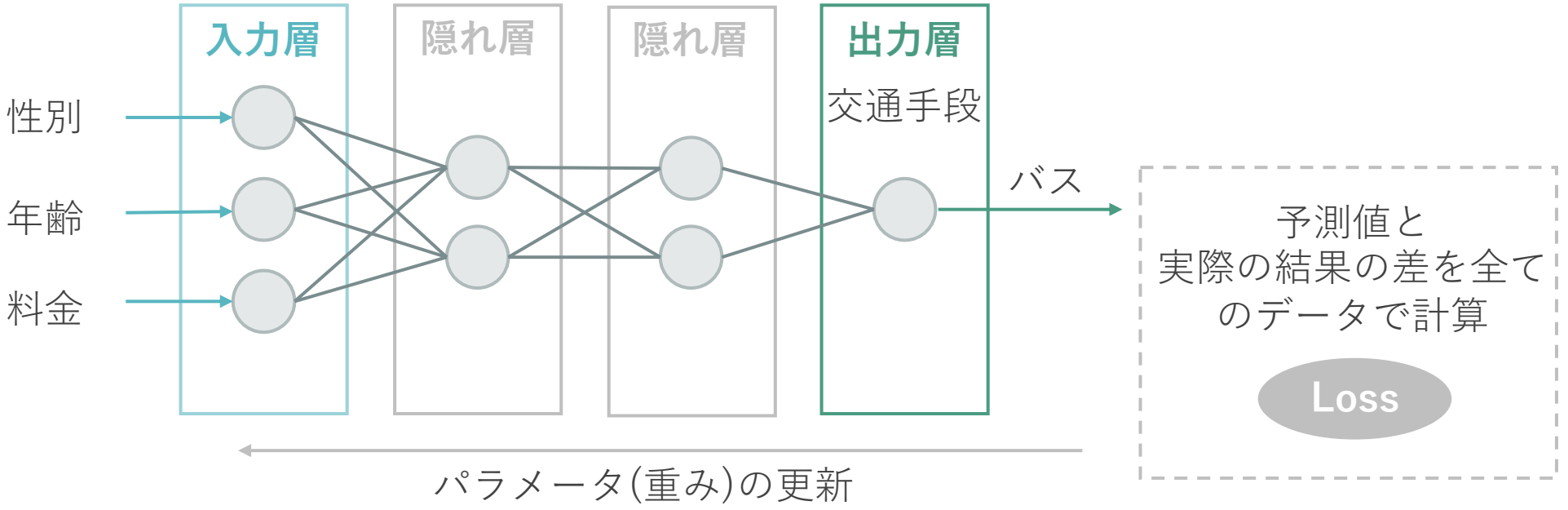
本研究 **DQN (deep Q-network)**を開発

- 強化学習の手法の1つであるQ学習とディープニューラルネットワーク(DNN)を組み合わせたもの
- 高次元の入力から直接学習を行える
- Atari 2600というゲームにおいて、ゲーム内の画像を入力として与えたところ、約半分のゲームでプロの人間に匹敵する性能を示した

▶ **広範なタスクに対して効果的である**ことを示した

Deep Neural Network (DNN)

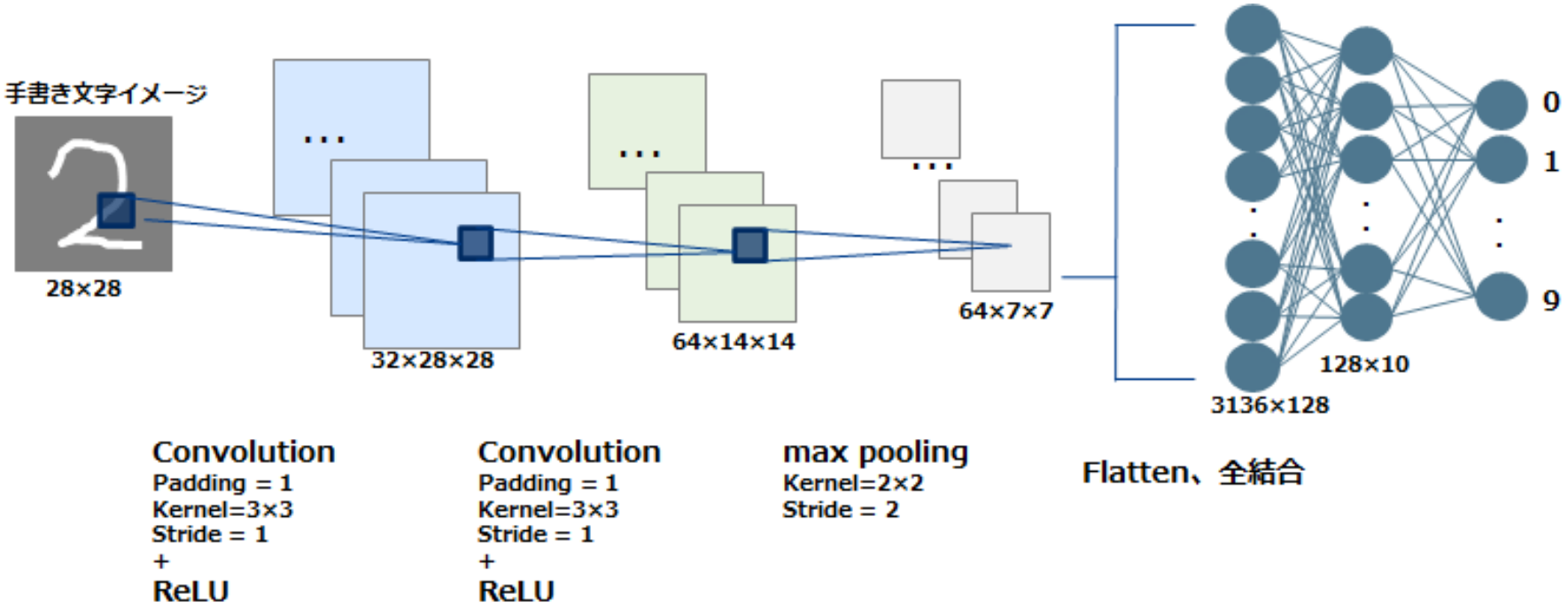
複雑なデータ(画像、音声など)からパターンを学習し、予測や分類を行うもの



CNN 参考HP

- DNNの1つで、画像認識に特化しているもの
- **畳み込み層**、**プーリング層**、**全結合層**という独自の構造を持つのが特徴

  特徴を抽出　データの次元削減　特徴量に基づいた分類



reinforcement learning

エージェントが(将来にわたって)最大の報酬を得られる行動を学習するもの

状態s　現在の環境の状況(ゲーム画面の状態)

行動a　エージェントが取れる選択(ジャンプ・移動)

報酬r　エージェントの行動に対する即時的価値(ポイント獲得)

方策π　戦略(状況sにおいてどういう行動aを取るべきか、を返す関数)

↑このモデル化をMarkov Decision Process(MDP, マルコフ決定過程)という

Q値/状態行動価値 $Q(s,a)$

ある状態sにおいてある行動aをとったときの(長期的な意味での)価値

報酬の割引総和を最大化する方策を見つけたい！

$$\mathbb{E}\left[r_t+\gamma r_{t+1}+\gamma^2 r_{t+2}+\ldots|s_t=s,\ a_t=a,\ \pi\right]$$

強化学習の学習

ある状態sである行動aを取った時、状態s'に遷移する確率がわかっている場合(モデルベース)

→Value IterationやPolicy Iteration = 繰り返し計算で状態価値関数を求めるもの

わからない場合(モデルフリー)

Q-learning

試行を繰り返すことで状態行動価値$Q(s,a)$を学習するもの

$Q(s,a) \approx r(s,a) + \gamma \max_{a'} E[Q(s',a')]$　　(γ:割引率)

≈が=になると学習完了ということなので、学習過程は次の式となる

$Q^{(i)}(s,a) \leftarrow Q^{(i-1)}(s,a) + \alpha(r^{(i)}(s,a) + \gamma \max_{a'} E\left[Q^{(i-1)}(s',a')\right] - Q^{(i-1)}(s,a))$　参考HP

特に複雑なデータの場合、$Q(s,a)$の収束は大変！→ **$Q(s,a)$をNNを用いて近似しよう！**

Deep Q-learning

$$Q^{(i)}(s,a) \leftarrow Q^{(i-1)}(s,a) + \alpha\left(r^{(i)}(s,a) + \gamma \max_{a'} E\left[Q^{(i-1)}(s',a')\right] - Q^{(i-1)}(s,a)\right)$$

NNの重みをθとし、 $y = r + \gamma \max_{a'} Q(s',a';\theta_i^-)$ とすると

θ_i^-：i回目より前の更新でのθ

誤差$L_i(\theta_i)$：

$$L_i(\theta_i) = \mathbb{E}_{s,a,r}\left[(\mathbb{E}_{s'}[y|s,a] - Q(s,a;\theta_i))^2\right]$$

$$= \mathbb{E}_{s,a,r,s'}\left[(y - Q(s,a;\theta_i))^2\right] + \mathbb{E}_{s,a,r}[\mathbb{V}_{s'}[y]].$$

微分すると、誤差伝播する際の勾配は次のようになる

$$\nabla_{\theta_i} L(\theta_i) = \mathbb{E}_{s,a,r,s'}\left[\left(r + \gamma \max_{a'} Q(s',a';\theta_i^-) - Q(s,a;\theta_i)\right)\nabla_{\theta_i} Q(s,a;\theta_i)\right].$$

学習のコツ

✓ **Experience Replay**

強化学習で与えられるデータは時系列的に連続していてデータ間に相関が出てしまう

⇒バラバラにするために、一旦全ての状態s/行動a/報酬$r(s,a)$/遷移先s'をメモリーに蓄積→学習の際にはランダムにサンプリングする

✓ **Fixed Target Q-Network**

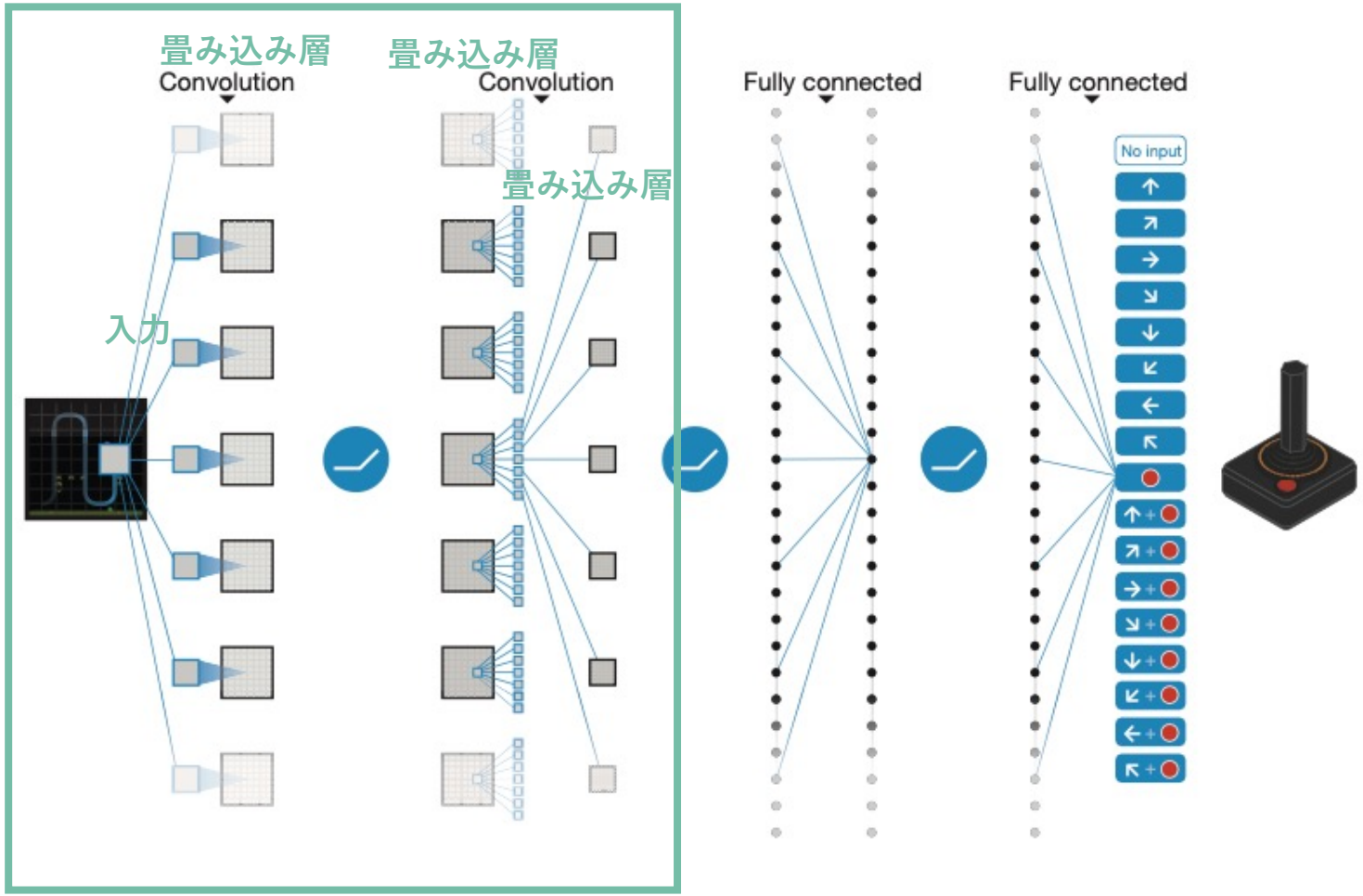
$Q_{\theta_{i-1}}(s',a')$：ラベルがパラメータの更新のたびに変わってしまうので学習が不安定になる可能性があり⇒データからいくつかのサンプルを抽出しミニバッチを作成しその学習中は期待値計算のためのθを固定する

✓ **報酬のclipping**

学習を進みやすくするため、報酬は正なら1、負なら-1と決めてしまう

Model Architecture

前処理　Atari 2600：210×160ピクセルの画像、128色のパレット→84×84ピクセルに縮小



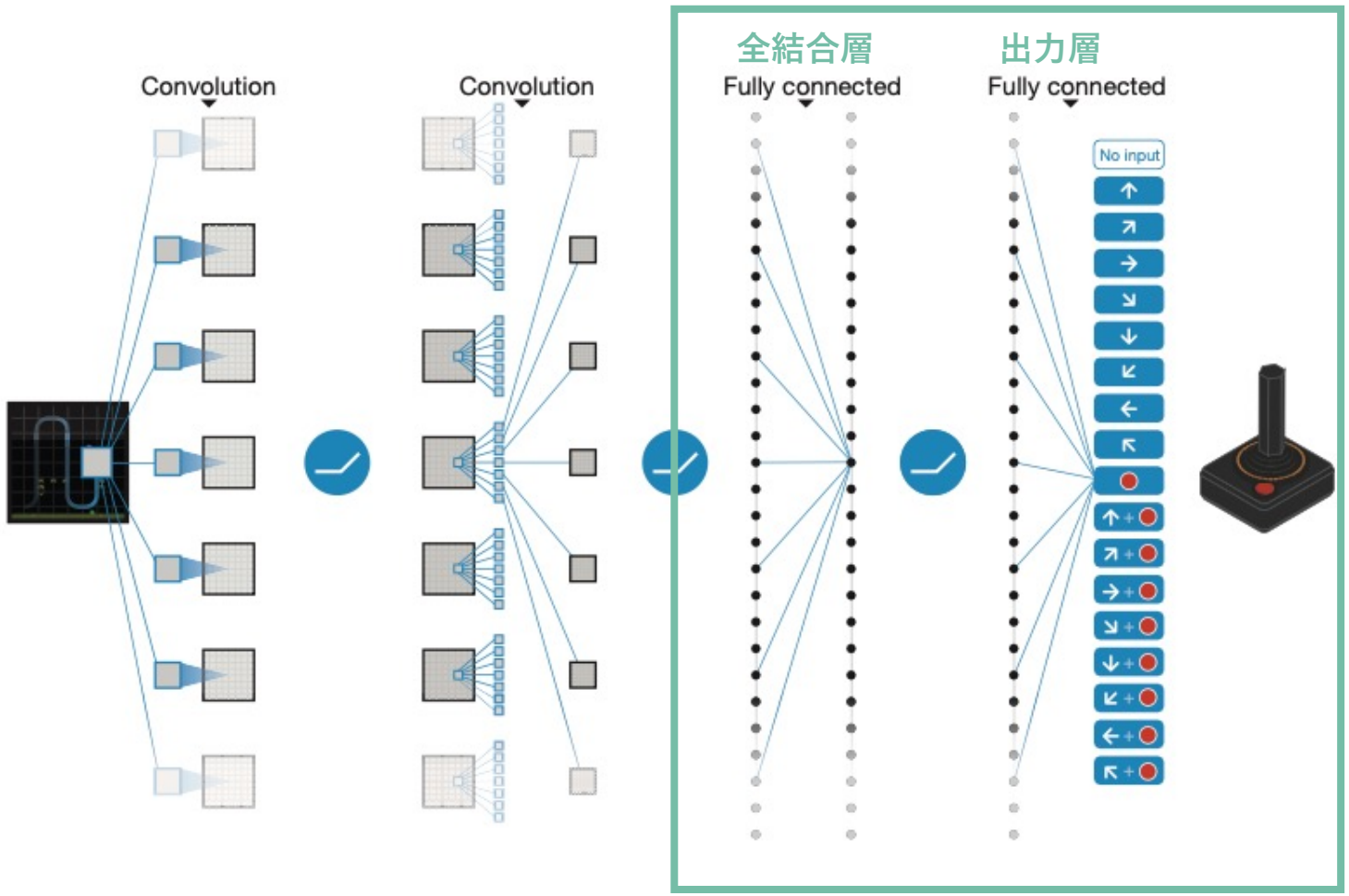
入力

- 前処理で作成された84×84×4枚の画像(直近4枚の画像を重ねたものを単一の入力とする)

畳み込み層

- 3つの畳み込み層があり、それぞれが異なるサイズとストライドフィルタを使用して画像の特徴を抽出する
- それぞれ処理後はReLU関数で処理される

Model Architecture



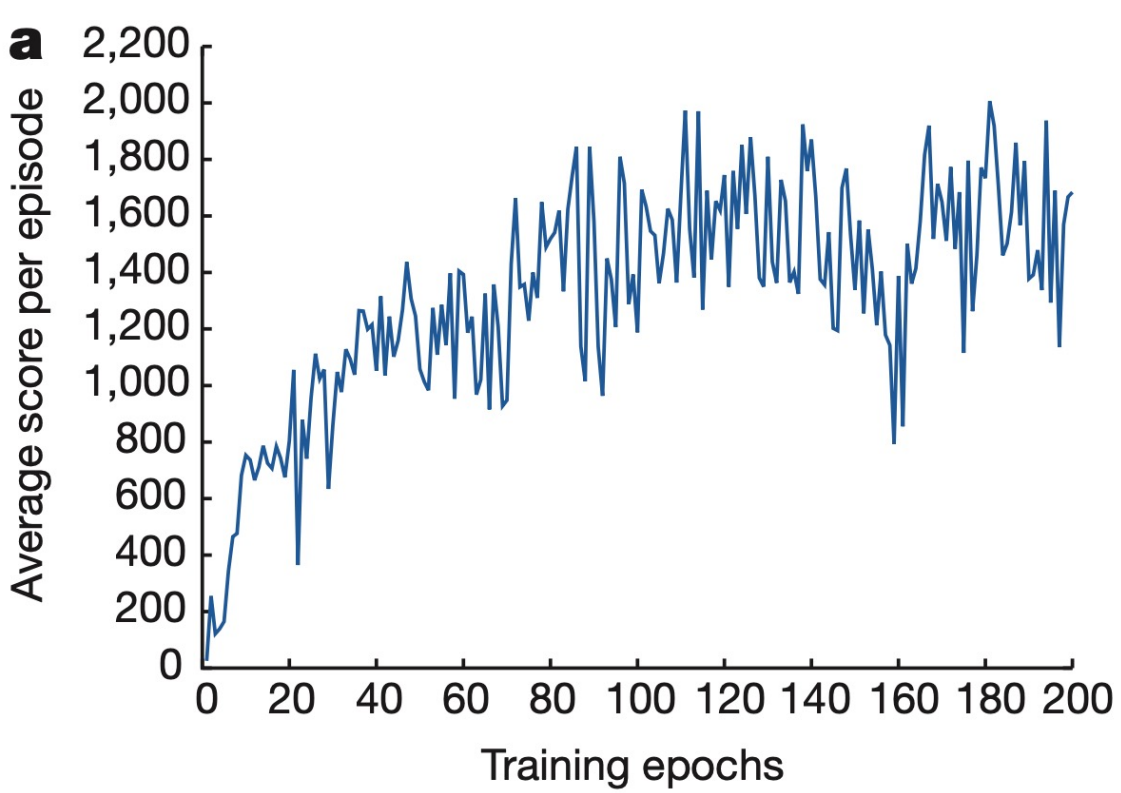
全結合層

- 畳み込み層の出力が、512個の隠れユニット全てに接続する

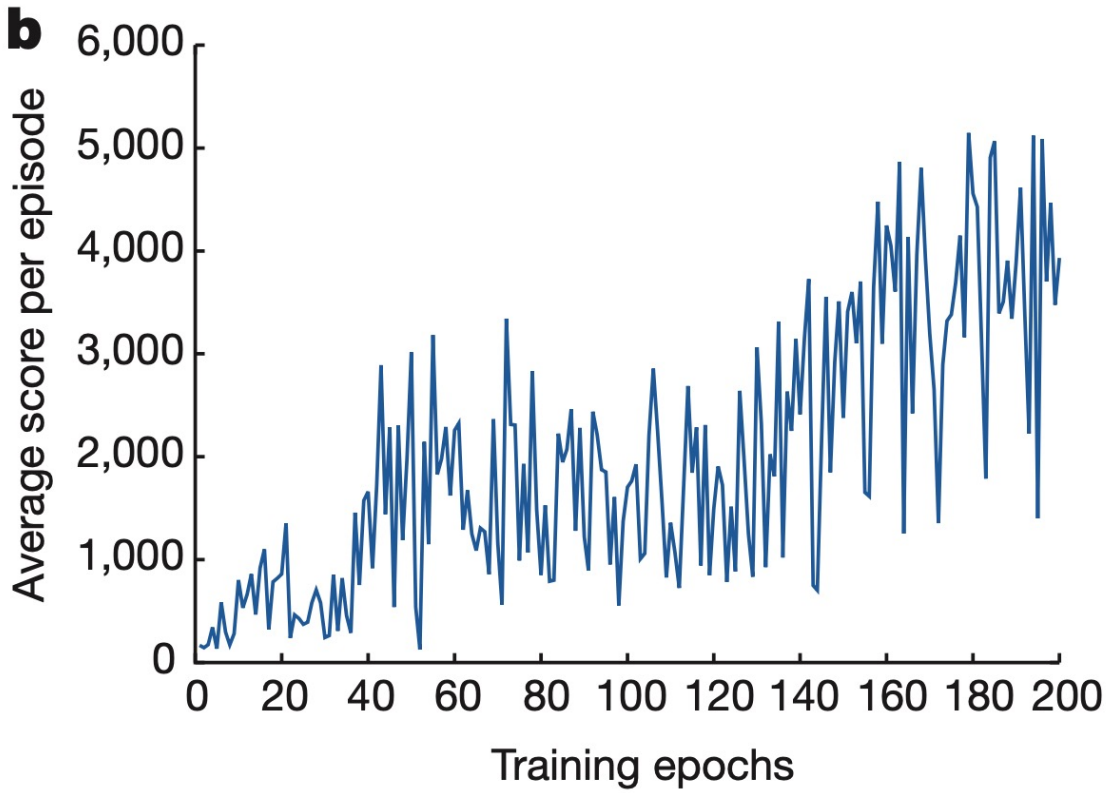
出力層

- 各ゲームタスクの有効アクションに対して1つの出力を持つ
- 各アクションに対するQ値を出力する

実験結果



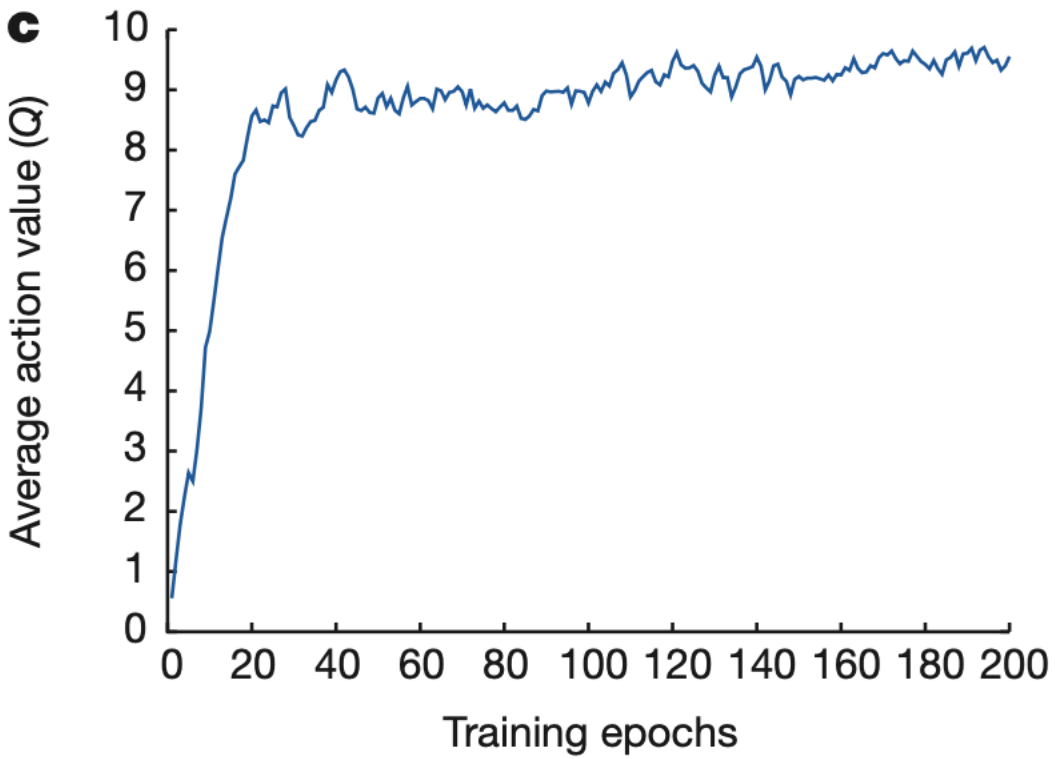
Space Invaders での平均スコア



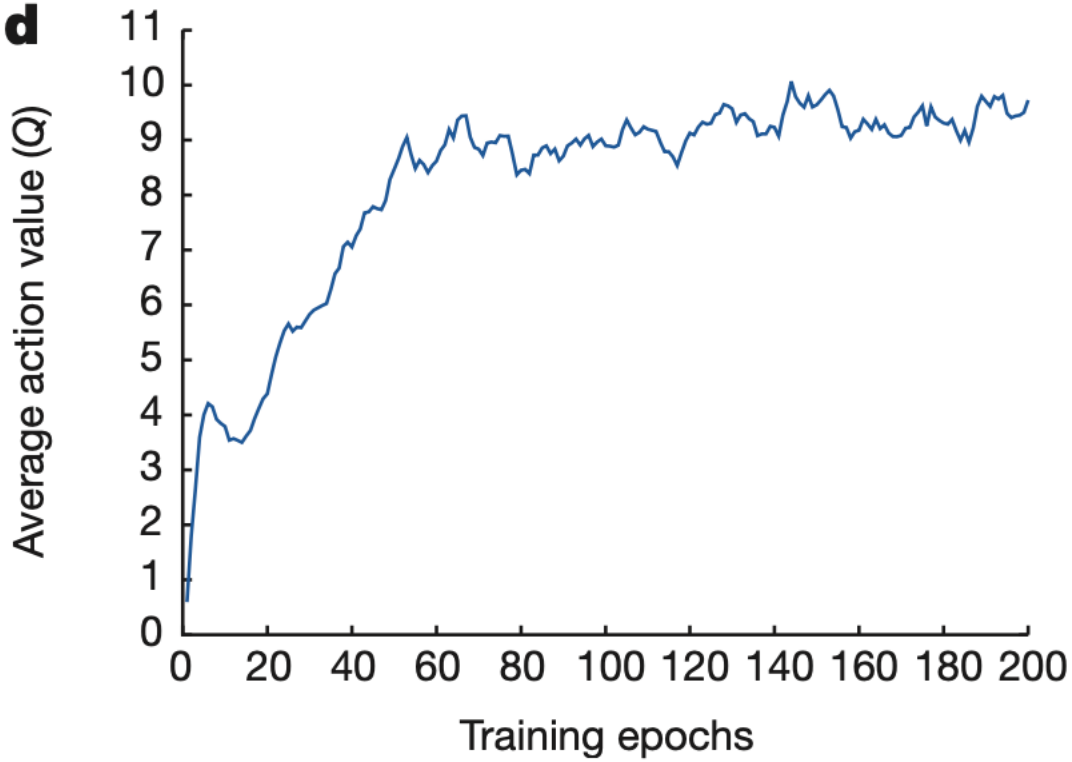
Seaquest での平均スコア

どちらもエポック数が増えるとスコアが上昇している
→DQNが異なるゲーム環境でも効果的にタスクを学習できることを示している

実験結果



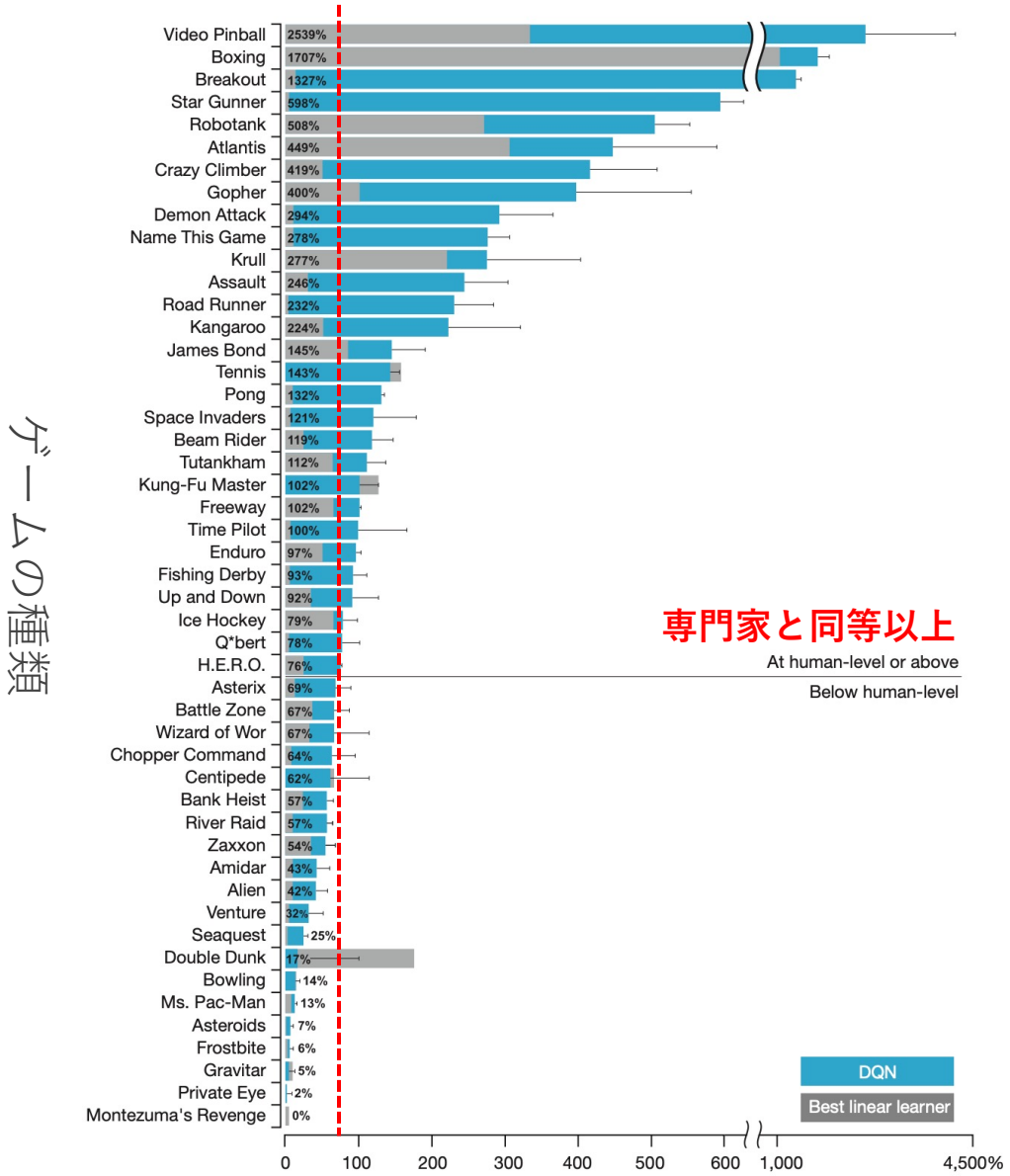
Space Invaders での平均予測行動価値(Q値)



Seaquest での平均予測行動価値(Q値)

どちらもエポック数が増えるとQ値が上昇している
→DQNが異なるゲーム環境でも効果的に行動価値を予測できることを示している

実験結果



横軸：
DQNでのパフォーマンスを、専門家によるプレー(100%)とランダムでのプレー(0%)を基準として正規化したもの

半分以上のゲームで専門家と同等以上のレベル(75%以上)で動作していることがわかる
⇔全てのゲームで高性能というわけではない

まとめ

結論

- 強化学習の手法の1つであるQ学習とCNNを組み合わせたモデル手法を提案した
  - 複雑な状況下でも強化学習の学習をすることが可能となった

- 提案手法は、全ての環境において高性能というわけではないが、様々なゲーム環境で人間と同等以上の性能を示した

所感

- 強化学習、ニューラルネットについてほとんど0から学ぶ良い機会でした。機械学習むずかしい。