

# 麻雀の打牌選択学習における Densely Multi Layer Perceptronの有効性と 局面から獲得可能な特徴量の有効性

藤田 竜貴<sup>1,a)</sup> 松崎 公紀<sup>2,b)</sup>

**概要:** 本稿では、一般的なニューラルネットワークモデルである Multi Layer Perceptron (MLP) と、MLP の発展系である Densely Multi Layer Perceptron (DenseMLP) を用いて、麻雀の評価関数を学習することを試みた。近年、ゲームの分野でのディープラーニングの成果はめざましい。しかし、麻雀において、MLP を用いた評価関数の学習の先行研究は存在するがテスト一致率は 43.7% と低く、成果をあげていない。一方、Convolutional Neural Network (CNN) を用いた先行研究では、68.8% と成果をあげている。そこで本稿では、先行研究で成果をあげられなかった MLP を用いて、先行研究と条件を変更し、中間層の数、各中間層のユニットの数を変化させ複数のモデルで実験を行なったところ、CNN を用いた先行研究のテスト一致率 68.8% と同程度のテスト一致率 68.6% を達成した。また、付録で示した実験ではニューラルネットワークへの入力に工夫を加えることで先行研究を大きく上回る 71.01% を記録した。また、MLP の発展系である DenseMLP についても同様の実験を行い、一致率の向上を試みたが、DenseMLP と MLP の間に大きな差は見られなかった。

## Effectivity of Densely Multi Layer Perceptron and Effectivity of Features Obtained from Positions in Machine Learning Mahjong Players

RYUKI FUJITA<sup>1,a)</sup> KIMINORI MATSUZAKI<sup>2,b)</sup>

**Abstract:** In this paper, we try to train the evaluation functions of mahjong with Multi Layer Perceptron (MLP) which is a general network model and Densely Connected Multi Layer Perceptron (DenseMLP) that is an extended model of MLP. In recent years, remarkable results of deep learning methods have been reported in the field of games. There was a study on learning of evaluation functions with MLP, but the test accuracy is as low as 43.7%. On the other hand, in a previous study with Convolutional Neural Network (CNN), the test accuracy was 68.8%. In this paper, we have conducted experiments with several models with MLP, and achieved 68.6%. In Addition, the experiment of appendix achieved 71.01%. We have conducted the same experiments with DenseMLP to improve the accuracy, but there was no significant difference between DenseMLP and MLP.

### 1. はじめに

強いコンピュータプレイヤーを実現するためには、良い手を選択するための機構が必要である。特に、手を選択するための評価関数に関する研究は広く研究されている。一般的に評価関数は、評価項目（次元数）を増やすほど、評価

<sup>1</sup> 高知工科大学大学院工学研究科基盤工学専攻情報学群  
Graduate School of Engineering, Kochi University of Technology

<sup>2</sup> 高知工科大学情報学群  
School of Information, Kochi University of Technology

a) 225127d@gs.kochi-tech.ac.jp

b) matsuzaki.kiminori@kochi-tech.ac.jp

関数の複雑さは増し、精度は高くなることが期待される。しかし、状態数の多いゲームでは、人間が手動で特徴量の抽出や重みの調整を行い、精度の高い評価関数を得ることは極めて困難である。そのため、機械学習による自動的な特徴量の抽出や重みの調整は広く行われている。特に特徴量の自動抽出と重みの自動調整で代表的なディープラーニングの研究は広く行われている、完全情報ゲームのコンピュータプレイヤーに関する研究はすでに非常に進んでおり、囲碁、将棋などにおいては、人間が到達できないレベルに達している。また、コンピュータプレイヤーが人間を打ち負かすことは困難と考えられてきた不完全情報ゲームであるポーカーにおいても、コンピュータプレイヤーが人間の4人のプロに対し勝利しており、近年のコンピュータプレイヤーの発展はめざましいものになっている [2]。しかし、不完全ゲームである麻雀において、先行研究はいくつかあるが、コンピュータプレイヤーが人間のトッププレイヤーに勝利しているとは言い難いというのが現状である。一般的なニューラルネットワークモデルである Multi Layer Perceptron (MLP) を用いた麻雀の研究として築地らの研究がある [3]。築地らの研究ではオートエンコーディングやドロップアウトなどの技術を用いて、最大 43.7% のテスト一致率を記録している。対して、Convolutional Neural Network (CNN) [4] を用いた麻雀の先行研究として Gao らの先行研究がある [5]。Gao らの研究では新しい入力セットの提案を行い、最大 68.8% のテスト一致率を記録している。CNN は主に画像判別などで成果をあげている手法で、ネットワークへの入力に対し畳み込みを行い効果的に特徴抽出を行うという特徴がある。この技術を麻雀というゲームに適用するには、麻雀という問題を一種の画像判別という問題に落とし込むために、開発者がどの情報を入力のごに配置するかなどの設計を行う必要があり、設計次第で精度が大きく変わりうると筆者は考える。

そこで本稿では、一般的なニューラルネットワークモデルである Multi Layer Perceptron (MLP) を用いて、中間層の数、各中間層のユニットの数を変化させ複数のモデルで実験を行なったところ、Gao らの先行研究のテスト一致率 68.8% と同程度の 68.6% を記録した。また、ニューラルネットワークの技術として、各層のニューロンの出力をそれ以降の全ての層への入力として伝達させるニューラルネットワークである Densely Network がある [7]。Densely Network には、消失勾配の問題を緩和するという利点がある。この技術を MLP に適用した Densely Multi Layer Perceptron (DenseMLP) についても同様の実験を行い一致率の向上を試みたが、MLP と DenseMLP の間で大きな差は見られなかった。

本稿は以下のように構成されている。2 章では関連する研究を紹介し、3 章では日本の麻雀のルールと用語に関する基本的な説明を行い、4 章では本書で行なった実験の設

定と用いたニューラルネットワークモデルを示し、5 章では行なった実験結果をまとめ、6 章では結論、考察と今後の展望を述べる。

## 2. 関連研究

麻雀におけるコンピュータプレイヤーの初期の研究として代表される水上らの研究 [6] では、コンピュータプレイヤーは攻撃に関する判断をする機構と防御に関する機構を持っており、攻撃に関する判断をする機構には人間が設定した特徴量を用いている。この手法はネット麻雀天鳳鳳凰卓の牌譜との打牌一致率 62.1% を記録している。対して、人工的な特徴量の抽出を必要としない自動特徴抽出能力で代表的なディープラーニングを用いた麻雀 AI の研究として、MLP、オートエンコーディング、ドロップアウトを用いた築地らの研究 (打牌一致率 44.1%) [3] や、自分の手牌以外の局面上の利用可能な情報を表現するためのデータセットを提案し、CNN を用いた Gao らの研究 (打牌一致率 68.8%) [5] などがある。本稿で用いる手法である DenseMLP を用いた研究は、筆者が知る限りではなされていない。

## 3. 麻雀の基本ルールと用語

麻雀は牌と呼ばれるタイルを用いて行う不完全情報ゲームで、一般的に 4 人でプレイされる。本書で扱うリーチ麻雀としても知られる日本の麻雀は、34 種類 4 枚ずつの計 136 枚の牌で行い。牌の種類には数牌と呼ばれる、1 から 9 の萬子 (マンズ)、1 から 9 の筒子 (ピンズ)、1 から 9 の索子 (ソウズ) の 27 種類と字牌と呼ばれる東、南、西、北、白、發、中の 7 種類の牌がある。プレイヤーにはそれぞれ 13 枚の牌が自分の手として配られ、各プレイヤーは自分の番に 1 枚引いて 1 枚捨てるということを繰り返しながらアガリとなる手牌構成を目指すゲームである。ゲーム開始時に各プレイヤーには 25000 点が牌譜され、ゲーム終了時に子の持ち点が多い順に順位が決まる。以下に重要な用語と説明を示す。

### 3.1 半荘、場、局

半荘は 1 ゲーム全体を指す。一般的に 1 ゲームのことを 1 半荘と表現する。局は半荘中に行われるサブゲームを指す。半荘には東場と南場がという状態が存在し、場は最低 4 回の局ごとに切り替わる。半荘開始時には東場という状態で始まり、東場が終わると南場という状態に推移する。南場が終わった状態でどのプレイヤーも 30000 点未満だった場合、西場という状態に切り替わり、西場が終わると強制的に半荘は終了する。

### 3.2 リーチ

リーチはアガリ状態の一手手前である時に宣言できるので、一種の役でもある。リーチを宣言すると、以降自分

の番で引いてきた牌と自分の手牌を入れ替えることができなくなるが、リーチを宣言した状態でアガリを宣言することができる。アガリ時の得点が高くなる。リーチを宣言すると自分の持ち点から 1000 点を供託として場に提供する。自分の得点が 1000 点未満の場合リーチを宣言することはできない。

### 3.3 役

役は、アガリ時に手牌構成等が特定の条件を満たしている場合認められ、それぞれの役に対応する翻（ハン）という値を得ることができる。役は 45 つ存在し、それぞれに対応する条件がある。アガリ時の得点は概ね翻の数で決まり、翻が大きければ大きいほど得点も高くなる。日本の麻雀では最低でも 1 つの役が存在しないとアガることができない。

### 3.4 ドラ

ドラは、アガリの際に持っている 1 翻加算される重要な牌のことである。新しい局が開始される時にある 1 種類の牌がドラとして決定され、各プレイヤーに知らされる。ドラは特定の行為により局の最中に新たにある種類の牌がドラとして決定されることがある。また赤ドラと呼ばれるドラがあり、萬子、筒子、索子の 5 の牌に 1 枚ずつ赤く彫刻された牌がある。その牌は常時ドラとして認められる。

## 4. 実験設定

本稿では、ニューラルネットワークのモデルが DenseMLP の場合と MLP の場合の 2 通り、中間層 3 層、4 層、5 層の 3 通り、各層のユニットの数は 256, 512 の 2 通り、ニューラルネットワークへの入力を 2 通りで計 24 通りの実験を行なった。この実験デザインに至った学術的な理由は特になく、時間が許す限り色々なモデルで実験を行うこと考えた結果である。学習は、ニューラルネットワークがある局面において訓練データと同じ打牌を行うネットワークを獲得する方針で学習を行う。学習条件は以下のとおりである。訓練データ、テストデータにはネット麻雀天鳳 [9] の 2015 年鳳凰卓の牌譜を用いた。テストデータの生成は学習に用いていない牌譜から 1 半荘のうちの自身がリーチしていない局面からランダムで 1 局面抽出するという方法を 3 万半荘について行い、3 万局面を生成した。学習局面数について、築地らの実験では  $5.0 \times 10^5$ 、Gao らの実験では  $6.0 \times 10^6$  の局面数が使用されている。今回の実験ではそれらよりも大きい  $4.5 \times 10^7$  の局面数を使用しており、先行研究に比べ訓練データのバリエーションに富んでいる。このことについて結論で触れるので留意されたい。

表 1 入力 (1) に用いた特徴量

特徴量	次元数
自分の手牌	136
計	136

表 2 入力 (2) に用いた特徴量

特徴量	次元数
自分の手牌	136
各家が捨てた牌	680
各家がリーチしているか否か	3
各家の安全牌	102
他家スタートか	4
現在他家か	4
現在何場か	3
現在何局目か	4
供託の数	10
積み棒の数	10
ドラの牌が何か	34
各色赤牌を持っているか	3
計	993

#### 学習条件

- 最適化アルゴリズムとして Adam を使用し、バッチサイズ 5000 とする
- 学習局面数は  $4.5 \times 10^7$  局面とする
- テストデータの局面数は  $3.0 \times 10^4$  局面とする
- 活性化関数は、Relu を用いる
- 学習回数は、20 回とする

### 4.1 入力に用いた特徴量

ニューラルネットワークへの入力 (1) に用いた特徴量を表 1 に、入力 (2) に用いた特徴量を表 2 に示す。入力 (1) では、自分の手牌の情報のみを入力とし、入力 (2) では自分の手牌の情報に加え、他家の副露状態、点数、各プレイヤーの河の切り順を除く、局面から取得できる情報を用いた。以下に自分の手牌についてのエンコード方法を示す。

### 4.2 自分の手牌

自分の手牌は、1 つの種類の牌について 4 次元で表現したものを 34 種類分全て結合させることで 136 次元で表現する。自分の手牌にある牌が 1 枚もない場合は  $[0,0,0,0]$ 、1 枚だけある場合は  $[1,0,0,0]$ 、2 枚だけある場合は  $[1,1,0,0]$ 、3 枚だけある場合は  $[1,1,1,0]$ 、4 枚ある場合は  $[1,1,1,1]$  と表現する。これを萬子の 1 から 9、索子の 1 から 9、筒子の 1 から 9、東、南、西、北、白、發、中の順に結合して表現する。先行研究では、1 種類の牌に対して 5 次元、one-hot で表現しており、先行研究と異なる表現を用いている。

### 4.3 実験に用いたモデル

実験に用いた MLP の構造を図 1 に、DenseMLP の構

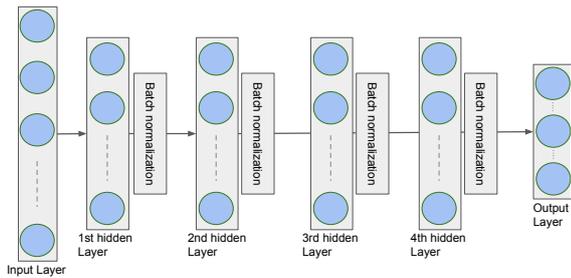


図 1 MLP の構造

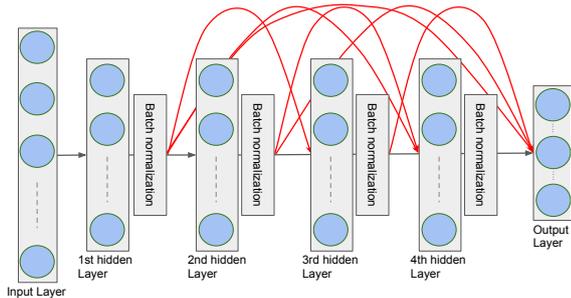


図 2 DenseMLP の構造

造を図 2 に示す。図は中間層が 4 層のものである。入力 (1) の場合の各モデルのパラメータ数を表 3 に、入力 (2) の場合の各モデルのパラメータ数を表 4 に示す。築地らの先行研究では、オートエンコーディング、ドロップアウト等の技術が使われていたが、それらは使用せず Batch Normalization[8] を用いて各層の出力を正規化していることに留意されたい。

## 5. 実験

本章では、行なった実験の結果を示す。入力 (1) の場合の各モデルの最高テスト一致率を表 5、入力 (2) における各モデルの最高テスト一致率を表 6 に示す。入力 (1)、中間層 3 層、各層のユニット数が 256 の時のテスト一致率の推移を図 3、512 の時のテスト一致率の推移を図 4 に示す。入力 (1)、中間層 4 層、各層のユニット数が 256 の時のテスト一致率の推移を図 5、512 の時のテスト一致率の推移を図 6 に示す。入力 (1)、中間層 5 層、各層のユニット数が 256 の時のテスト一致率の推移を図 7、512 の時のテスト一致率の推移を図 8 に示す。入力 (2)、中間層 3 層、各層のユニット数が 256 の時のテスト一致率の推移を図 3、512 の時のテスト一致率の推移を図 4 に示す。入力 (2)、中間層 4 層、各層のユニット数が 256 の時のテスト一致率の推移を図 5、512 の時のテスト一致率の推移を図 6 に示す。入力 (2)、中間層 5 層、各層のユニット数が 256 の時のテスト一致率の推移を図 7、512 の時のテスト一致率の推移を図 8 に示す。

表 3 入力 (1) の場合の各モデルのパラメータ数

中間層	ユニット数	モデル	パラメータ数
3	256	DenseMLP	257,536
	256	MLP	174,592
	512	DenseMLP	908,288
	512	MLP	611,328
4	256	DenseMLP	462,848
	256	MLP	240,128
	512	DenseMLP	1,712,128
5	512	MLP	873,472
	256	DenseMLP	733,696
	256	MLP	305,664
	512	DenseMLP	2,778,112
	512	MLP	1,135,616

表 4 入力 (2) の場合の各モデルのパラメータ数

中間層	ユニット数	モデル	パラメータ数
3	256	DenseMLP	476,928
	256	MLP	393,984
	512	DenseMLP	1,347,072
	512	MLP	1,050,112
4	256	DenseMLP	682,240
	256	MLP	459,520
	512	DenseMLP	2,150,912
5	512	MLP	1,312,256
	256	DenseMLP	953,088
	256	MLP	525,056
	512	DenseMLP	3,216,896
	512	MLP	1,574,400

### 5.1 まとめ

各モデルの最高一致率は入力 (2)、3 層、ユニット数 256 の場合と、入力 (2)、5 層、ユニット数 512 の場合の DenseMLP と MLP での差が見られたが、それ以外のモデルではほとんど差は見られなかった。図 6 と図 8 を見ると DenseMLP の方が過学習を起こしているように見受けられる。入力 (1) の場合は自分の手牌についてしか評価しないため、入力 (2) と比べごく簡単な問題になっている。ユニット数が 512 でかつ Densely なネットワークにすると、問題の規模に対しネットワークのパラメータ数が多すぎたことにより過学習が起こっているのだと考えられる。表 3 より、中間層 4 層、ユニット数 512 で DenseMLP の場合、パラメータ数が約 170 万で過学習の傾向が見られた。次に大きなパラメータ数のモデルは、中間層 5 層、ユニット数 512 の MLP モデルで、パラメータ数が約 110 万である。このモデルは過学習の傾向は見られなかった。また、図 12 と図 14 を見ると途中までは DenseMLP の一致率が上回っていて、MLP よりも速く学習が進んでいるように見えるが、終盤になると同様に過学習を起こしているように見受けられる。

## 6. 結論

本稿では、MLP と DenseMLP で複数の実験を行い、中

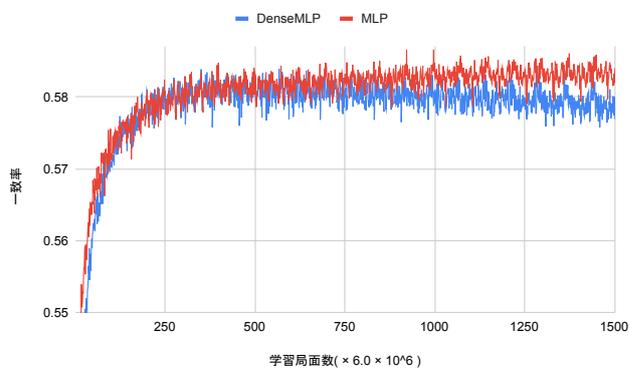
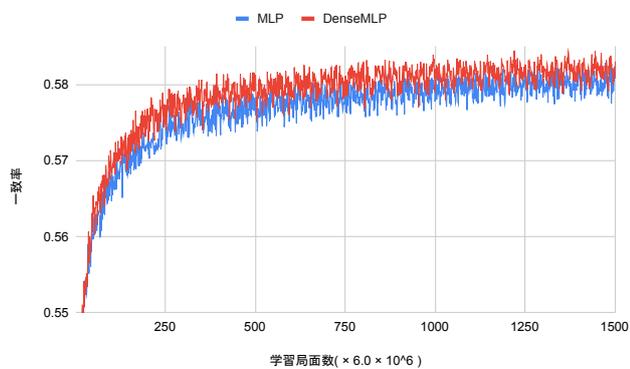


図 3 入力 (1), 中間層 3 層, 各層のユニット数 256 の時テスト一致率 図 6 入力 (1), 中間層 4 層, 各層のユニット数 512 の時テスト一致率

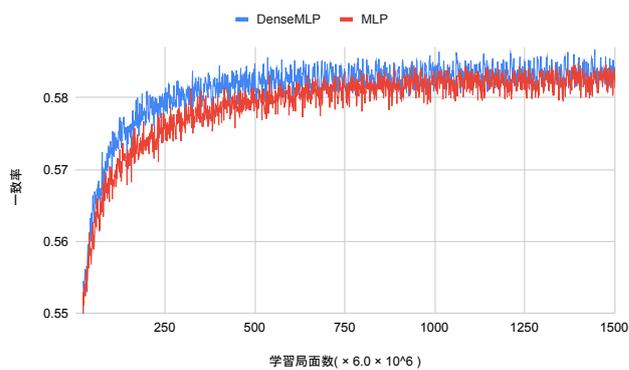
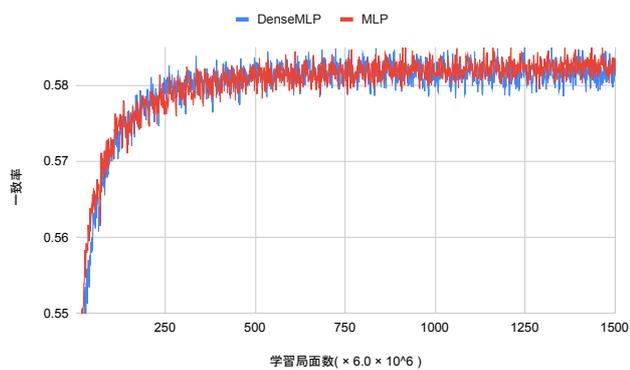


図 4 入力 (1), 中間層 3 層, 各層のユニット数 512 の時テスト一致率 図 7 入力 (1), 中間層 5 層, 各層のユニット数 256 の時テスト一致率

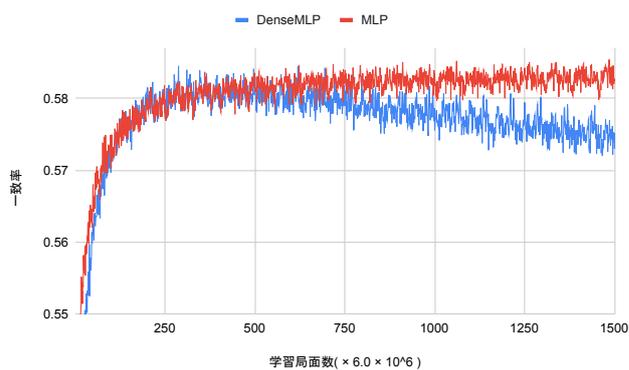
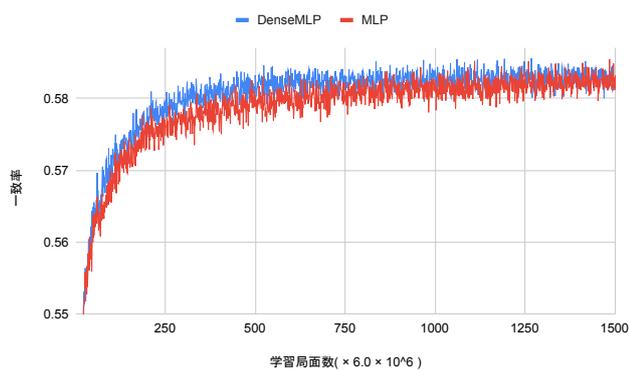


図 5 入力 (1), 中間層 4 層, 各層のユニット数 256 の時テスト一致率 図 8 入力 (1), 中間層 5 層, 各層のユニット数 512 の時テスト一致率

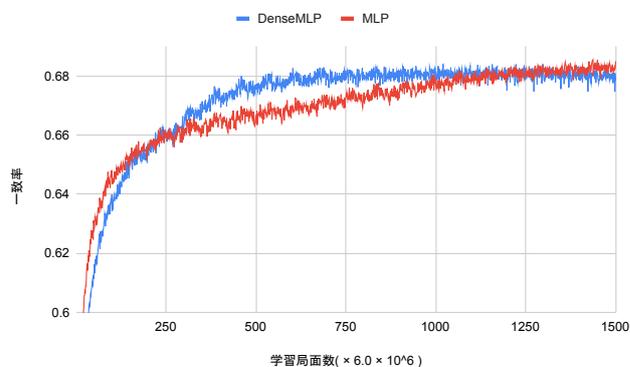
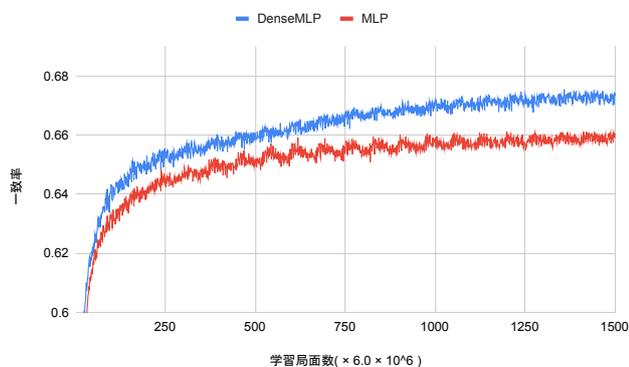


図 9 入力 (2), 中間層 3 層, 各層のユニット数 256 の時テスト一致率 図 12 入力 (2), 中間層 4 層, 各層のユニット数 512 の時テスト一致率

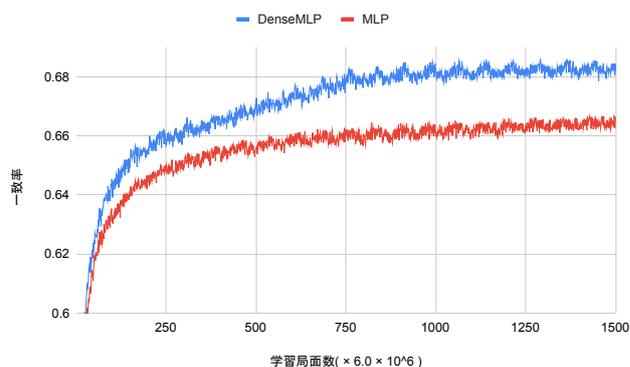
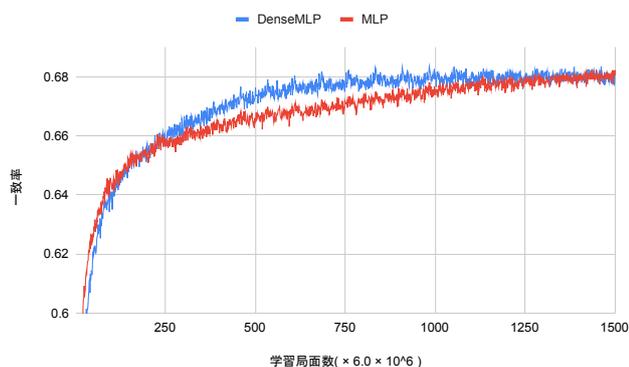


図 10 入力 (2), 中間層 3 層, 各層のユニット数 512 の時テスト一致率 図 13 入力 (2), 中間層 5 層, 各層のユニット数 256 の時テスト一致率

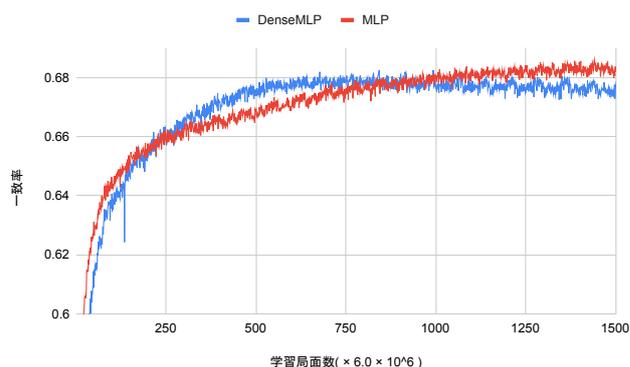
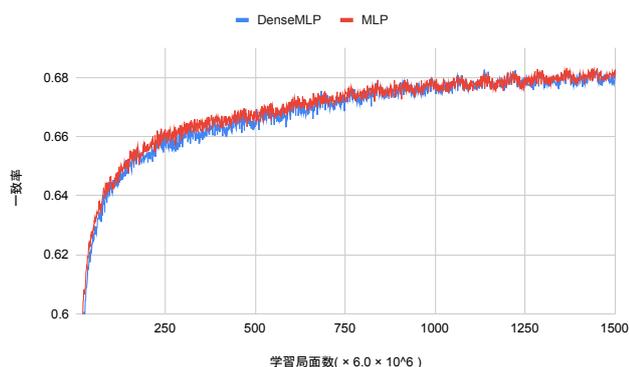


図 11 入力 (2), 中間層 4 層, 各層のユニット数 256 の時テスト一致率 図 14 入力 (2), 中間層 5 層, 各層のユニット数 512 の時テスト一致率

表 5 入力 (1) の各モデルの最高一致率

中間層	ユニット数	モデル	最高一致率
3	256	DenseMLP	58.45%
	256	MLP	58.29%
	512	DenseMLP	58.58%
4	512	MLP	58.53%
	256	DenseMLP	58.56%
	256	MLP	58.54%
5	512	DenseMLP	58.39%
	512	MLP	58.66%
	512	DenseMLP	58.67%
5	256	MLP	58.48%
	512	DenseMLP	58.45%
	512	MLP	58.54%

表 6 入力 (2) の各モデルの最高一致率

中間層	ユニット数	モデル	最高一致率
3	256	DenseMLP	67.55%
	256	MLP	66.12%
	512	DenseMLP	68.36%
4	512	MLP	68.24%
	256	DenseMLP	68.27%
	256	MLP	68.33%
5	512	DenseMLP	68.41%
	512	MLP	68.55%
	256	DenseMLP	68.58%
5	256	MLP	66.67%
	512	DenseMLP	68.26%
	512	MLP	68.60%

間層 5 層, 各層のユニット数 512, MLP のモデルで, CNN を用いた先行研究と同程度のテスト一致率 68.6% を達成した. また, 実験結果からは層を 4 層以上かつ各層のユニット数を 512 以上かつ Densely なネットワークにすると, MLP に比べ学習が速く進む傾向が見られたが, 過学習の傾向も見られた. 本稿で行なった実験の入力セットでは, 問題の複雑さに対しこれらのモデルはパラメータ数が多すぎたことにより過学習が起こったものと推察する. また, 4 で説明したとおり, 先行研究よりも学習局面のバリエーションが多く, MLP を用いた築地らの先行研究とは異なり, Batch Normalization を使用している. また, ネットワークへの入力セットも, 先行研究とは異なるものを用いている. 本稿の実験では, 先行研究との比較が適切に行える実験設定ではなかったと言える. 今後の展望として, 条件を統一した追試と検定が考えられる. また, 今回使用したネットワークへの入力, 麻雀の重要な要素である他家の副露状態, 各プレイヤーの点数, 各プレイヤーの河の切り順については一切無視されており評価されていない. 展望として, これらの情報を入力に加えた追試が考えられる.

## 参考文献

- [1] J.Schmidhuber.: Deep Learning in neural networks: An overview, *Neural networks*, Vol.61, pp85-117 (2015).
- [2] N.Brawn, T.Sandholm.: Superhuman AI for heads-up no-limit poker: Libratus beats top professional, (2018).
- [3] 築地毅, 柴原一友: ディープラーニング麻雀-オートエンコードとドロップアウトの有効性, *ゲームプログラミングワークショップ 2015 論文集*, pp.136-142 (2015).
- [4] K.Alex, I.Sutskever, *et al.*: Imagenet classification with deep convolutional neural networks, *Advances in neural information processing systems* (2012).
- [5] S.Gao, F.Okuya, *et al.*: Supervised Learning of Imperfect Information Data in the Game of Mahjong via Deep Convolutional Neural Networks, *ゲームプログラミングワークショップ 2017 論文集*, pp.163-170 (2017).
- [6] 水上直紀, 鶴岡慶雅.: 期待最終順位に基づくコンピュータ麻雀プレイヤーの構築, *ゲームプログラミングワークショップ 2015 論文集*, pp.179-186 (2015).
- [7] H.Gao, L.Zhuang.: Densely Connected Convolutional Networks, *arXiv preprint arXiv: 1608.06993v5*, (2018).
- [8] I.Sergey, S.Christian.: Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift, *arXiv preprint arXiv: 1502.03167v3*, (2015)
- [9] オンラインネット麻雀天鳳. <https://tenhou.net/>

## 付 録

## A.1 追加実験

締め切りの都合により, 本文に掲載できなかった実験を示す. 中間層 4 層, 各層のユニット数 512 の MLP と DenseMLP のモデルで, 表 A-1 の入力を用いて実験を行った. 本稿で行なった実験で用いた各家が捨てた牌の入力を廃止し, 代わりに, 各種類の牌の見えている枚数を使用した.

表 A-1 入力 (3) に用いた特徴量

特徴量	次元数
自分の手牌	136
各種類の牌の見えている枚数	136
各家がリーチしているか否か	3
各家の安全牌	102
他家スタートか	4
現在他家か	4
現在何場か	3
現在何局目か	4
供託の数	10
積み棒の数	10
ドラの牌が何か	34
各色赤牌を持っているか	3
計	449

最高テスト一致率を A-2 に, 一致率の推移を図 A-1 に示す.

## A.1.1 まとめ

麻雀というゲームでは, 基本自分のアガリを目指して手

表 A.2 最高テスト一致率

中間層	ユニット数	モデル	最高一致率
4	512	DenseMLP	71.01%
	512	MLP	70.71%

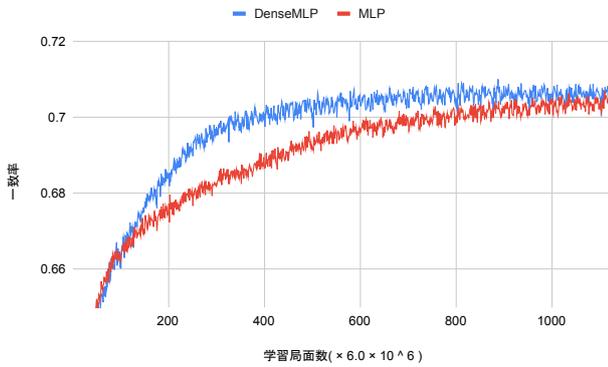


図 A.1 テスト一致率の推移

を作るゲームだ。すでに多く見えている牌はその分だけ引ける確率が低くなっていることを示す。極端な例で言うと、すでに自分の視点から4枚見えている牌を引ける確率は0である。引ける見込みの小さい牌を引くことを期待するような手牌構成にしないようにすることは麻雀では基本となる。変更前の入力(2)では誰が何を何枚切っているかという情報を使っていたが、この情報は結局何が何枚見えているかという情報に集約されると考えた。そこで、各家が捨てた牌の代わりに各種類の牌の見えている枚数を直接与えた。入力の情報量を減らすことなく次元数を約半分に減らすことで精度が上がると考えた。結果は、精度は向上しGaoらの先行研究68.8%を大きく超える71.01%を記録した。