2020 年度 卒業論文

麻雀における他家の手牌の待ち予測

2020 年 2 月 7 日

電気情報工学科
（学生番号： 1TE16240N）

松田 真治

九州大学 工学部

概要

　本論文では機械学習を用いた麻雀における他家の手牌の待ちを予測する手法を提案する。インターネット上で公開されている牌譜から他家の捨て牌、他家の副露などの情報を抜きだし、それらを教師データとして３層ニューラルネットワークを用いて学習を行う。

　このとき他家の捨て牌情報に順序関係を持たせることで、より精度の高い予測結果を得られると考え、比較実験などを行った。

目次

1. はじめに

1.1 研究背景

麻雀とは不完全情報ゲームである。将棋や囲碁のようにすべての情報がプレイヤーに開示されている完全情報ゲームと異なり、限られた情報の中から他のプレイヤーの状況を推察しつつ自分に与えられた手の価値の期待値を最大化するような選択を行い続けるゲームある。

例えば将来的には高くなりそうな手をもらった場合でも、他のプレイヤーにスピード感を感じた場合には、自分の手の価値を下げてでも速くあがることを優先したり、自分の手が安く勝負をする価値が低い場合でも、他のプレイヤーにのびのびとしたプレーをさせないように、ときにはブラフを用いてでも他のプレイヤーの足止めを行ったりもする。このように麻雀のような不完全情報ゲームでは自分の都合のみでプレイの選択を行うことは少なく、常に他のプレイヤーの状況を推察しながら臨機応変に立ち回りをすることが重要である。

周りの状況に臨機応変に対応しながら自分の意思決定を行う様は現実世界でいうところの押し引きと非常に類似している。したがって麻雀のような不完全情報ゲームに関する研究を行うことによって、現実で起こる押し引きの絡むような問題を解決する手助けとなる可能性があり、研究の意義は十分にあると言える。

1.2 研究目的

本研究の目的は牌譜を教師データとして学習を行ったモデルを用いて、押し引きの際に特に重要な要素の一つである他のプレイヤーの待ち牌の予測を行うことである。

1.3 研究手法

先行研究[1][2][3][4][5]では、他のプレイヤーの捨て牌から待ち牌の予測を行う際に用いられる情報は、色ごとの捨てられた枚数や捨てられた牌の種類の枚数、手出しの回数、捨てた中張牌の種類数などの情報である。これらの情報は牌が捨てられた順序は考慮されていない。

しかし一般的には、捨て牌から他のプレイヤーの待ち牌の予測を行う際に、相手がどの順で牌を切ったかという情報は極めて重要なものだと考えられている。そこで本研究では待ち牌予測に捨てられた牌の順序を考慮する手法を提案し、その予測性能を評価した。

1.4 簡単な麻雀のルール

麻雀のルールについて簡単に説明する。

まず麻雀とは 34 種*4 枚の合計 136 枚の牌を用いて 4 人のプレイヤーが互いの得点を奪い合うゲームである。

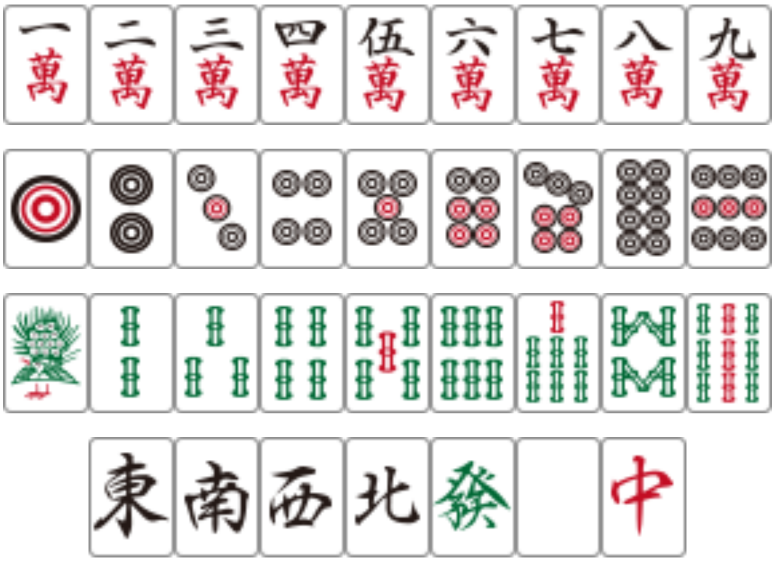

図 1-1. 麻雀牌

麻雀牌は 1~9 の牌からなる数牌と東,南,西,北,白,発,中の 7 種類からなる字牌から構成される。また数牌には萬子（マンズ）、筒子（ピンズ）、索子（ソーズ）といった種類が存在する。図 1 で言えば上の行から順番に萬子の 1~9,筒子の 1~9,索子の 1~9,字牌となっている。

図 1-2. 赤牌

また 5 萬,5 筒,5 索の各 4 枚の内 1 枚は色が赤くなっており、それぞれ赤 5 萬,赤 5 筒,赤 5 索と呼ばれ、まとめて赤牌と呼ばれる。赤牌を持った状態のまま和了することで、貰える点数が増加する。

麻雀の 1 試合は半荘という単位で呼ばれる。半荘は複数の局から成り立っており、最終局が終わった時点での得点の多寡で順位を競う。

局の開始時に各プレイヤーは 13 枚の牌を配られる（配牌）。配牌が終わると、山から 1 枚牌を引くツモという行為と手牌から牌を 1 枚捨てる打牌という行為を行う。このツモと打牌の 1 セットの動きを各プレイヤーが順番に行う。

手持ちの 13 枚にツモってきた牌 1 枚、もしくは手持ちの 13 枚に他家の捨てた牌 1 枚をあわせた 14 枚の牌が特定の形を満たし、かつ役がある場合にプレイヤーは和了（ホーラ）を宣言し、役に準ずる点数を得ることができる。
ツモってきた牌で和了することをツモと呼び、他家の捨てた牌で和了することをロンと呼ぶ。
和了を宣言するための特定の形には 3 つの種類があり、それぞれ一般形、七対子形、国士無双形と呼ばれる。
以下では最も基本的な一般形についてのみ説明を行う。
一般形とは 1 雀頭と 4 面子から構成される形である。
雀頭とは同じ牌 2 枚によって構成される形である。
面子とは 3 枚または 4 枚の牌による特定の組み合わせで、順子、刻子、槓子の 3 種類が存在する。
順子は種類の同じ連続した牌 3 枚によって構成される形である。
刻子は同じ牌 3 枚によって構成される形である。
槓子は同じ牌 4 枚によって構成される形である。

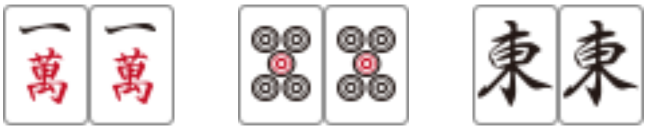

図 1-3. 雀頭の例

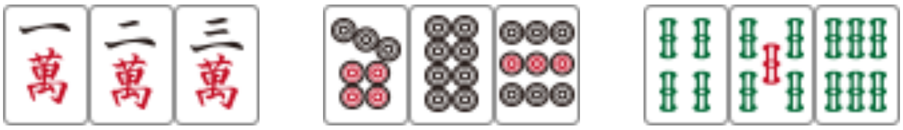

図 1-4. 順子の例

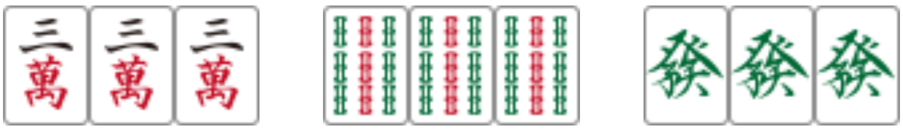

図 1-5. 刻子の例


図 1-6. 槓子の例

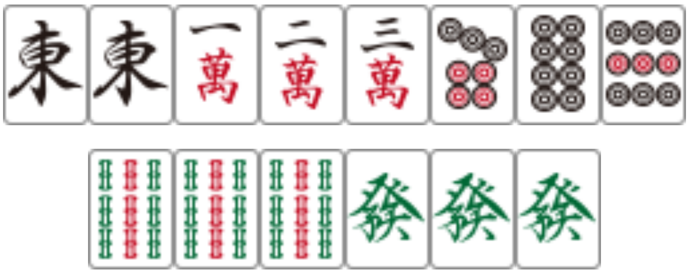

図 1-7. 和了一般形の例

また麻雀には面子を完成させる方法として副露と呼ばれるものがある。副露とは自分の手牌にあと 1 つ牌が揃えば面子が完成するという状況で、かつ他家がその足りない 1 つの牌を捨てた場合に、出来上がった面子を他家に晒すことと引き換えに、その牌を河から拾って面子を完成させることである。

副露にはチーとポンとカンが有る。

チーとは副露して順子の形を作ることであり、これは自分の上家（向かって左側のプレイヤー）の河からしか牌を拾うことができない。

ポンとは副露して刻子の形を作ることであり、全ての他家の河から牌を拾うことができる。

カンとは副露して槓子の形を作ることであり、ポンと同様に全ての他家の河から牌を拾うことができる。またカンは他家の牌を拾わずとも、自分で四枚同じ牌を集めた時点で他家に晒すことができる(暗槓)。厳密に言えば暗槓は副露ではない。

次に麻雀の役について説明する。役とは和了の形が特定の条件を満たした場合に付加されるもので、基本的に貰える点数はその役の数によって決定される。一般的なルールでは約 48 の役が存在する。

最も基本的な役は立直（リーチ）である。立直とは 1000 点支払い、自分が聴牌していることを他家に宣言することで付加される役である。

他にも 1 種類の数牌のみによって構成される清一色（チンイーソー）や么九牌を手牌に含まない断么九（タンヤオチュー）など様々な役が存在する。

例えば図 1-7 は、雀頭と全ての面子が必ず么九牌を含んでいるので混全帯么九（ホンチャンタイヤオチュー）という役が付加される。

1.5 麻雀の簡単な用語

聴牌（テンパイ） 和了に必要な牌の枚数が 1 枚の状態。
不聴（ノーテン） 聴牌していない状態。
向聴数（シャンテンスウ） 聴牌に必要な牌の枚数。
n 向聴（n シャンテン） 聴牌に必要な牌の枚数が n 枚の状態。

山 ツモで取ってくる牌が並べられている場所のこと。
河 打牌で捨てる牌を並べる場所のこと。
半荘（ハンチャン） 1ゲームの単位。
局 ゲームの最小単位のこと。複数の局によって半荘が構成される。
流局 全てのプレイヤーが和了できずに終局した状態のこと。
配牌 局開始時に全プレイヤーに配られる13枚の牌のこと。
手牌 自分の手となる牌のこと。
他家（ターチャ） 自分以外のプレイヤーのこと。
上家（カミチャ） 左側のプレイヤーのこと。
対面（トイメン） 正面のプレイヤーのこと。
下家（シモチャ） 右側のプレイヤーのこと。
数牌 1~9の牌。萬子、筒子、索子の3種類が存在する。
字牌 数牌でない牌。東南西北白発中の7種類が存在する。
色 萬子、筒子、索子の種類のこと。一色手とは混一色や清一色などの限られた種類の牌だけで構成した和了に付加される役のことを指す。
么九牌（ヤオチュウパイ） 数牌の1・9と字牌のこと。
中張牌（チュウチャンパイ） 数牌の2~8のこと。
和了（ホーラ） 手牌を一定の形に揃えて公開すること。他のゲームなどにおける「あがり」に相当する。
ツモ ツモした牌で和了すること。
ロン 他家の捨てた牌で和了すること。
副露 他家の打牌を取得することで面子を完成させること。鳴きと呼ばれることもある。
待ち牌 切ると他家にロンされる牌のこと。
打牌 牌を河に捨てること。またその牌のこと。捨てる、切る、打つなどとも言う。
降り 自分の和了は諦めて、他家にロンされないように安全そうな牌を切ること。
ダブロン 切った牌が同時に2人のプレイヤーにロンされること。
トリプルロン 切った牌が同時に3人のプレイヤーにロンされること。

2.捨て牌の打牌順序と待ち牌の予測

2.1 捨て牌から他家の待ち牌を予測する際の打牌順序の重要性

捨て牌から他家の待ち牌を読む場合に、打牌順序の情報が重要になる理由について例を用いて簡潔に説明する。

麻雀において数牌は端に近づけば近づくほど面子を作りにくくなる。例えば 1 を用いて構成できる面子は 123,111 の 2 パターンしか無いが、5 を用いて構成できる面子は 345,456,567,555 の 4 パターンである。つまり 1 を用いて構成できる面子の数は 5 を用いて構成できる面子の数の半分しかない。

したがって基本的に聴牌のしやすさ、和了のしやすさの観点から考えると、数牌の利用価値は 1,9<2,8<3-7 である。したがって、仮に自分の手牌に 1 と 5 の両方があり、かつそれが共に自分の手に不要な牌でいずれ 2 枚とも捨てる予定だとした場合、ある程度麻雀を勉強している人であれば、打牌の順序は 1,5 の順になるのが普通である。しかし実際に麻雀を遊んでみると往々にして 5,1 の順で打牌することがある。この順番に打牌する理由は様々である。例を挙げると、雀頭と全ての面子が么九牌を含んだ時に付加される役である混全帯么九などを狙っていることなどが考えられる。したがって 1,5 の順で打牌されている場合に比べると 5,1 の順で打牌されている場合の方がわずかに端に寄っている数牌や字牌の危険度が高く、中に寄っている数牌の危険度は低くなると考えことができる。

このように捨て牌の情報に打牌順序の情報をもたせると推測できる要素が増える。ここで本研究では、捨て牌のデータを、単なる捨て牌の集合で表すのではなく、打牌された順序を織り込む手法を考えた。

2.2 手出しとツモ切り

麻雀では牌の切り方に 2 つのパターンがある。手出しとツモ切りである。手出しとは牌を山からツモした後、もともと手牌にあった牌を切ることである。それに対してツモ切りとは牌を山からツモした後、そのツモした牌をそのまま切ることである。

この 2 つは似ているようで大きく意味が異なる。

手出しされた牌とは、何らかの理由があって手牌に残していたが、ツモってきた牌の価値がより高かったために切られてしまった牌である。言い換えれば捨て牌にある牌よりは価値が高く、手牌にある牌よりは価値が低い牌である。

一方ツモ切りされた牌とは手牌にある牌よりは価値が低いが、捨て牌にある牌より価値が高いとは必ずしも言えない牌である。

上の例では 5,1 の順で牌を切った場合、端に寄っている数牌や字牌の危険度が増し、中に寄っている牌の危険度は低くなると述べたが、1 がツモ切りだった場合、この推測は成立しない。あくまで、この推測は 1 と 5 が直接比較された結果、1 のほうが手牌に残す価値が高いと考えられて、5 が捨てられたことが前提となっているためである。例えば 5 を引いて 5 を捨て、1 を引いて 1 を捨てた場合などは当然当てはまらない。

したがって捨て牌の情報に打牌順序の情報を織り込む際には、その切られた牌が手出しであるかツモ切りであるかという情報も合わせて織り込むことによって、より待ち牌予測の性能が向上する可能性がある。

3.麻雀サイト天鳳の牌譜データ

待ち牌推測の提案手法の性能評価のために、本研究では、オンライン麻雀サイト天鳳の牌譜データを利用した。天鳳とは最王手のオンライン麻雀サイトであり、本来はオンラインで麻雀対戦をするためのサイトである。ゲームが行われる一方で、その対戦データは牌譜として記録され、天鳳の開発者によって無償で公開[6]されている。

本研究では天鳳五段以上の実力者のみがプレイできる鳳凰卓の牌譜を性能評価のために利用した。期間は 2017 年度のもので、ルールは最も一般的なルールである四人打ち、東南戦、赤有り、喰断么九有りのものに限定した。

天鳳の牌譜は XML ファイルによって特殊な形式で表現されているため、機械学習で用いるためには解析をして前処理を行う必要がある。解析については小林聡氏のブログ[7][8][9][10]を参考にした。

天鳳の牌譜は拡張子こそ XML であるが、その内容は XML の仕様に準拠しておらず、実質天鳳の独自構文であると考えて良いものである。

ゲーム中に起こった事象はタグとその属性に表現よって表現され、事象が起きた順にそのまま並べられている。閉じタグなども存在していない。

天鳳の牌譜は 1 ファイルが 1 半荘を表す。

<INIT>タグは局の開始を表現し、<AGARI>タグ、<RYUKYOKU>タグはそれぞれ、和了があった場合の情報、和了がなかった場合の情報が表現されている。<AGARI>タグも<RYUKYOKU>タグも共に局の終了を表現するタグである。すなわち<INIT>タグから<AGARI>タグまたは<RYUKYOKU>タグまでで一局を表現している。1 つの XML ファイルにこの<INIT>タグから<AGARI>タグまたは<RYUKYOKU>タグまでの集合が局数分書いてある。ただしダブロンが生じた場合は<INIT>(途中の情報)<AGARI><AGARI>などの形で表現される。

開局から終局までのツモ情報や打牌情報、副露情報、立直情報などは<INIT>タグ〜<AGARI>タグまたは<RYUKYOKU>タグの間に羅列されている。本研究では和了の情報のみを利用するため、<INIT>タグ〜<AGARI>タグまでの情報のみを利用した。

本研究で利用したタグについて簡潔に説明する。

まず前提として、天鳳の牌譜では 136 枚の牌に 0~135 までの牌番号が振られている。

ツモ情報は<T(牌番号)>, <U(牌番号)>, <V(牌番号)>, <W(牌番号)>の形で表記されている。T,U,V,W は誰がツモしたかを表現している。

打牌情報は<D(牌番号)>, <E(牌番号)>, <F(牌番号)>, <G(牌番号)>の形で表記されている。同様に D,E,F,G も誰が打牌したのかを表現している。T,U,V,W の表すプレイヤーはそれぞれ DEFG の表すプレイヤーと対応している。

例えば A さんが 0 の牌をツモって 0 の牌を打牌し、その次に B さんが 1 の牌をツモして 2 の牌を打牌した場合は<T0><D0><U1><E2>のように表現される。打牌が手出しだったかツモ切りかなどの情報は牌譜には含まれていないので適宜ツモした牌の牌番号と捨てた牌の牌番号が同値かどうかを比較して、自分で判定を行う必要がある。

和了情報は<AGARI>タグで表現されている。AGARI タグは複数の属性を持つが、本研究で利用したのは和了したプレーヤーが誰かを表す who 属性、和了したときの和了したプレイヤーの手牌を表す tehai 属性、どの牌で和了したかを表す machi 属性、和了したプレーヤーの副露状況を表す m 属性である。

machi 属性はどの牌で和了したかを表現しており、厳密には待ち牌を表現しているわけではない。例えば下図の和了例を見てみる。

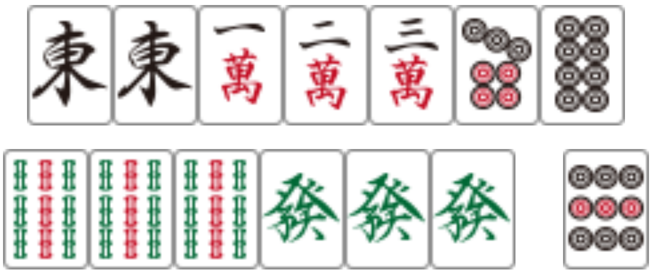

図 3-1. 和了の例

上の例について 9 筒で和了したとする。すると牌譜上の AGARI タグの machi 属性には 9 筒に対応する牌番号が記録される。しかし、実際は 6 筒も待ち牌である。

したがって厳密な待ち牌は牌譜に記録されていないため、自分で検出しなければならない。

まず tehai 属性から手牌情報を抜き出す。次に machi 属性からどの牌で和了したかを抜き出し、その牌を手牌から消し、和了する一歩手前の状態を再現する。次に 34 枚の牌全てについて、1 萬 ＋ 手牌, 2 萬 ＋ 手牌,・・・, 西 ＋ 手牌, 北 ＋ 手牌 が和了しているかどうかを判定しする。和了状態ならば手牌に足したその牌は待ち牌である。

和了は深さ優先探索などを使えば判定できる。

m 属性は副露の種類を 2 進法を用いて独特な形で表現しているため、詳細は上記の小林氏のブログを参考にしていただきたい。

4.牌譜のデータ表現

4.1 打牌にツモ切り・手出しの情報を持たせる表現

ツモ切りした牌と手出しした牌を違う牌だとみなすと、牌の種類は合計 74 種類と考えることができる。
したがって以下のような対応をさせることで 0~73 の番号でツモ切り・手出しの情報を持った牌の表現ができる。

表 4-1. 牌と番号の対応付け

牌	番号
ツモ切り 1 萬～9 萬	0~8
ツモ切り 1 筒～9 筒	9~17
ツモ切り 1 索～9 索	18~26
ツモ切り東、南、西、北	27~30
ツモ切り白、発、中	31~33
ツモ切り赤 5 萬、赤 5 筒、赤 5 索	34~36
手出し 1 萬～9 萬	37~45
手出し 1 筒～9 筒	46~54
手出し 1 索～9 索	55~63
手出し東、南、西、北	64~67
手出し白、発、中	68~70
手出し赤 5 萬、赤 5 筒、赤 5 索	71~73

4.2 打牌順序を考慮しない捨て牌の表現

打牌順序を考慮しない捨て牌とは、要するに何を切ったかという情報である。図 4-2 の捨て牌からデータを抽出する事を考える。

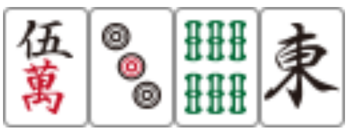

図 4-2. 捨て牌例（すべてツモ切りとする）

入力ベクトルとして列数 74、初期値 0 の表を考える。次に捨て牌のそれぞれの牌番号について対応する列番号の存在フラグを 1 にする。今回の場合は列番号 4,11,23,27 の列の存在フラグを 1 にすれば良い。

列番号	0	1	・・・	4	・・・	11	・・・	74
存在 フラグ	0	0	・・・	1	・・・	1	・・・	0

図 4-3. 打牌順序を考慮しない捨て牌を表すベクトルのイメージ

4.3 打牌順序を考慮する捨て牌の表現

順序関係の情報を持った捨て牌の作り方について説明する。

簡潔に言えば任意の 2 枚の牌を捨てる順序対について一意に番号を割り当てて表現すれば良い。2 枚の牌の組み合わせは高々 74 * 74 = 5476 であるから、予め 2 枚の牌の打牌順序を 0~5475 の番号に対応する方法を決めておけば、任意の打牌の順序関係を表現することができる。

本研究では(先に捨てた牌の牌番号) * 74 + (後に捨てた牌の牌番号)という式を用いて、打牌された順序を保った 2 枚の牌の対を番号で表現した。

例として図 4-2 の捨て牌から打牌順序を考慮する捨て牌のベクトル化を示す。

はじめに捨て牌からすべての打牌順序を抜き出す。図 4-2 の場合は（ツモ切り 5 萬，ツモ切り 3 筒），（ツモ切り 5 萬，ツモ切り 6 索），（ツモ切り 5 萬，ツモ切り東）,(ツモ切り 3 筒，ツモ切り 6 索), (ツモ切り 3 筒，ツモ切り東),(ツモ切り 6 索，ツモ切り東)である。これを牌番号に置き換える。置き換えたものが(4,11),(4,23),(4,27),(11,23),(11,27),(23,27)である。これらの番号の組み合わせを上記の(先に捨てた牌の牌番号) * 74 + (後に捨てた牌の牌番号)という式に通すとそれぞれ 307,319,323,837,841,1729 となる。

よって入力ベクトルとして図 4-3 のような 5476 列の表を用意し、列番号 307,319,323,837,841,1729 の列の存在フラグの値を 1 にする。

列番号	0	・・・	307	・・・	1729	・・・	5475
存在 フラグ	0	・・・	1	・・・	1729	・・・	0

図 4-4. 打牌順序を考慮する捨て牌を表すベクトルのイメージ

このようにすることで、任意の 2 つの牌を切った順序を表現することができ、入力に打牌順序の情報を与えることができる。

4.4 副露（チー）の表現

赤牌を含むかどうかで 2 パターン存在する。

どこの牌を鳴いたかで 3 パターン存在する。

副露した牌の中で最小の牌番号をもつ牌の種類で 21 パターン存在する。具体的に説明する。123 チーの場合 1,234 チーの場合 2,…,789 チーの場合 7 のようにチーの場合、最小の牌の番号は 1~7 の 7 種類で、それが萬子、筒子、索子の 3 種類ずつあるので合計 21 種類である。

したがって 2 * 3 * 21 = 126 より全てのチーは適宜 0~125 の番号に対応させることで表現できる。

4.5 副露（ポン）の表現

赤牌を含むかどうかで 2 パターン存在する。

どの牌を鳴いたかで 34 パターン存在する。

したがって 2 * 34 = 68 より全てのポンは適宜 0~67 の番号に対応させることで表現できる。

4.6 副露（カン）の表現

カンの種類（暗カン・明カン・加カン）で 3 パターン存在する。

どの牌を鳴いたかで 34 パターン存在する。

したがって 3 * 34 = 102 より全てのカンは適宜 0~101 の番号に対応させることで表現できる。

5.実験手法

5.1 条件

和了に至るまでの捨て牌の情報をどのような形式のデータで表現するかについて 2 通りの表現を考えた。

・打牌順序を考慮しない捨て牌 ＋ 副露
・打牌順序を考慮する捨て牌 ＋ 副露

また、N 回打牌を行った捨て牌について、打牌順序を考慮しない捨て牌は高々 N 個の牌の集合であるのに対し、打牌順序を考慮する捨て牌は高々$\binom{n}{2}$個の打牌の順序対の集合である。したがって打牌回数が増えれば増えるほど、打牌順序を考慮する場合の捨て牌の情報量が、打牌順序を考慮しない捨て牌の情報よりも大きくなると推測し、以下の 2 通りの表現で実験をした。

・打牌順序を考慮しない捨て牌（打牌 10 回以上） ＋ 副露
・打牌順序を考慮する捨て牌（打牌 10 回以上） ＋ 副露

5.2 アンダーサンプリング

本研究では、ある牌が待ち牌となるサンプルを正例、待ち牌でないサンプルを負例と定義する。

ある牌に着目したとき、全サンプルのうちの正例となるサンプルは 1%~10%程度である。そのまま機械学習を行うと、全ての場合において待ち牌ではないと予測すると accuracy が 90%以上を達成することができ、横着な学習が進んでしまう。

そこで学習を行う前に、データの負例の数を正例の数に合わせる事によって、効率の良い学習を進めることができる。この作業をアンダーサンプリングと呼ぶ。

本研究では数牌については正例 20000 件、負例 20000 件になるようにサンプリングを行った。字牌については正例が 20000 件も存在しなかったため、正例 10000 件、負例 10000 件になるようにサンプリングを行った。

また性能評価で用いるサンプルはアンダーサンプリングを行っていないものである。

5.3 実装

Chainer で 3 層ニューラルネットを構築して学習を行った。第 1 層の入力数は入力次元数、出力数は 100。第 2 層の入力数は 100、出力数は 100。第 3 層の入力数は 100、出力数は 2。最適化関数は Adam を利用した。

エポック数 50、バッチサイズ 16 で学習を行う。

データは70%を訓練データセットとして、30%を検証データセットとして利用する。

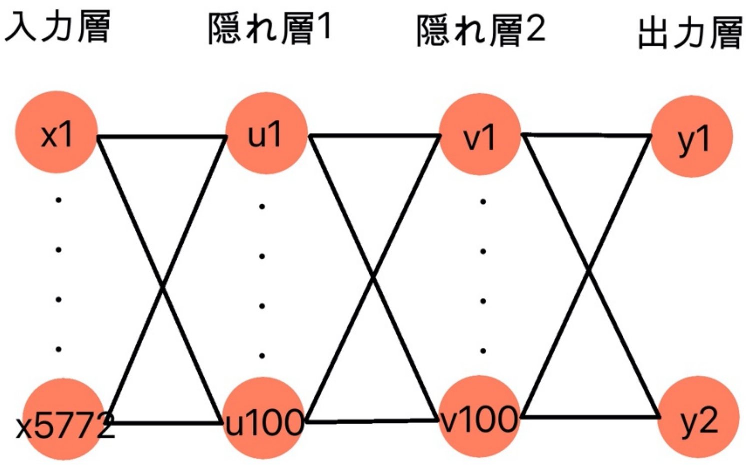

図5-1. NNイメージ図

6. 実験結果と考察

6.1 ランダムに予測した場合

比較のためにランダムに待ち牌かどうかを判定した場合の結果を示す。

正例だと判定する割合を a,サンプルのうちの正例の割合を b とすると、tp, fp, fn, tn は以下のようになる。

$$tp = ab$$
$$fp = a(1-b)$$
$$fn = (1-a)b$$
$$tn = (1-a)(1-b)$$

よって評価指標 precision, recall, f1score は以下のようになる。

$$precision = \frac{tp}{tp+fp} = b$$
$$recall = \frac{tp}{tp+fn} = a$$
$$f1score = \frac{2*precision*recall}{precision+recall} = \frac{2ab}{a+b}$$

ここで f1score を最大にするような a を考える。上の式は

$$\frac{2ab}{a+b} = \frac{2b(a + b)}{a + b} - \frac{2b^2}{a+b} = 2b - \frac{2b^2}{a+b}$$

と変形できるので a = 1、すなわち全てを正例だと判定した時に、最大の f1score を出すことができる。

各牌について全てを正例だと判定した場合の結果が表 6-1 である。

表 6-1. 全て正例だと判定したときの評価

牌	正例の割合	precision	recall	f1score
1 萬	0.0446	0.0446	1.0000	0.0854
2 萬	0.0638	0.0638	1.0000	0.1200
3 萬	0.0749	0.0749	1.0000	0.1393
4 萬	0.0923	0.0923	1.0000	0.1690

5 萬	0.0977	0.0977	1.0000	0.1780
6 萬	0.0914	0.0914	1.0000	0.1675
7 萬	0.0746	0.0746	1.0000	0.1389
8 萬	0.0646	0.0646	1.0000	0.1213
9 萬	0.0450	0.0450	1.0000	0.0861
1 筒	0.0448	0.0448	1.0000	0.0857
2 筒	0.0641	0.0641	1.0000	0.1204
3 筒	0.0749	0.0749	1.0000	0.1393
4 筒	0.0918	0.0918	1.0000	0.1682
5 筒	0.0963	0.0963	1.0000	0.1757
6 筒	0.0917	0.0917	1.0000	0.1681
7 筒	0.0745	0.0745	1.0000	0.1387
8 筒	0.0634	0.0634	1.0000	0.1192
9 筒	0.0437	0.0437	1.0000	0.0838
1 索	0.0439	0.0439	1.0000	0.0841
2 索	0.0638	0.0638	1.0000	0.1199
3 索	0.0732	0.0732	1.0000	0.1365
4 索	0.0904	0.0904	1.0000	0.1657
5 索	0.0962	0.0962	1.0000	0.1756
6 索	0.0910	0.0910	1.0000	0.1668
7 索	0.0735	0.0735	1.0000	0.1369
8 索	0.0636	0.0636	1.0000	0.1195
9 索	0.0440	0.0440	1.0000	0.0843
東	0.0147	0.0147	1.0000	0.0290
南	0.0147	0.0147	1.0000	0.0290
西	0.0122	0.0122	1.0000	0.0242
北	0.0130	0.0130	1.0000	0.0257
白	0.0166	0.0166	1.0000	0.0327
発	0.0165	0.0165	1.0000	0.0325
中	0.0166	0.0166	1.0000	0.0326

6.2 捨て牌の打牌順序を考慮しない場合と考慮する場合の比較

・打牌順序を考慮しない捨て牌 ＋ 副露
・打牌順序を考慮する捨て牌 ＋ 副露
を入力として、それぞれ学習を行った。結果は以下のようになった。

表 6-2. 打牌順序を考慮しない場合の評価

牌	precision	recall	f1score
1 萬	0.106	0.829	0.189
2 萬	0.143	0.800	0.243
3 萬	0.151	0.792	0.254
4 萬	0.177	0.790	0.289
5 萬	0.191	0.779	0.307
6 萬	0.176	0.802	0.289
7 萬	0.141	0.815	0.240
8 萬	0.131	0.812	0.226
9 萬	0.107	0.821	0.189
1 筒	0.108	0.811	0.190
2 筒	0.131	0.815	0.226
3 筒	0.155	0.765	0.257
4 筒	0.176	0.796	0.288
5 筒	0.187	0.785	0.302
6 筒	0.182	0.772	0.295
7 筒	0.145	0.776	0.245
8 筒	0.127	0.842	0.220
9 筒	0.104	0.829	0.185
1 索	0.109	0.817	0.193
2 索	0.137	0.796	0.233
3 索	0.145	0.798	0.245
4 索	0.175	0.773	0.285
5 索	0.184	0.782	0.298
6 索	0.178	0.787	0.290
7 索	0.147	0.809	0.250

8 索	0.135	0.823	0.232
9 索	0.107	0.825	0.189
東	0.037	0.835	0.071
南	0.035	0.832	0.068
西	0.030	0.856	0.059
北	0.030	0.853	0.057
白	0.041	0.848	0.078
発	0.042	0.843	0.081
中	0.042	0.837	0.080

表 6-3. 打牌順序を考慮する場合の評価

牌	precision	recall	f1score
1 萬	0.112	0.835	0.198
2 萬	0.142	0.810	0.242
3 萬	0.149	0.805	0.252
4 萬	0.175	0.809	0.288
5 萬	0.183	0.820	0.300
6 萬	0.178	0.807	0.292
7 萬	0.140	0.829	0.240
8 萬	0.137	0.828	0.235
9 萬	0.106	0.866	0.189
1 筒	0.109	0.857	0.193
2 筒	0.144	0.803	0.244
3 筒	0.144	0.818	0.245
4 筒	0.179	0.805	0.293
5 筒	0.188	0.798	0.305
6 筒	0.173	0.815	0.285
7 筒	0.146	0.796	0.247
8 筒	0.141	0.820	0.240
9 筒	0.112	0.816	0.197
1 索	0.112	0.838	0.198

2 索	0.143	0.819	0.243
3 索	0.152	0.794	0.255
4 索	0.188	0.777	0.303
5 索	0.187	0.797	0.303
6 索	0.179	0.805	0.293
7 索	0.150	0.796	0.253
8 索	0.142	0.820	0.242
9 索	0.106	0.864	0.189
東	0.036	0.854	0.070
南	0.039	0.803	0.075
西	0.030	0.877	0.058
北	0.032	0.851	0.062
白	0.038	0.892	0.073
発	0.039	0.875	0.074
中	0.040	0.882	0.077

以上のデータから
・ランダムに判定した場合と打牌順序を考慮しない場合の f1score の比較
・ランダムに判定した場合と打牌順序を考慮する場合の f1score の比較
・打牌順序を考慮しない場合と打牌順序を考慮しない場合の f1score の比較
をまとめたものが以下の図である。

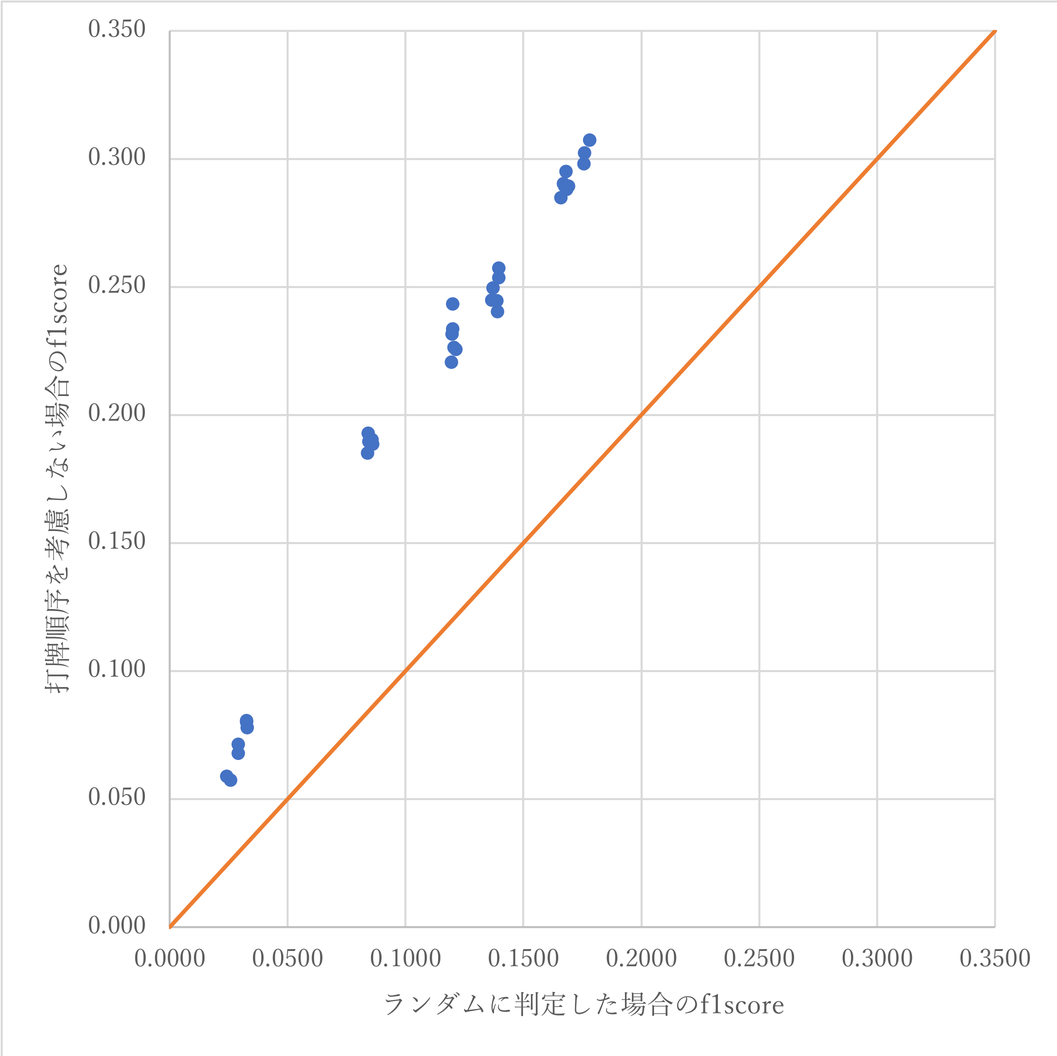

図 6-4. ランダムに判定した場合と打牌順序を考慮しない場合の f1score の比較

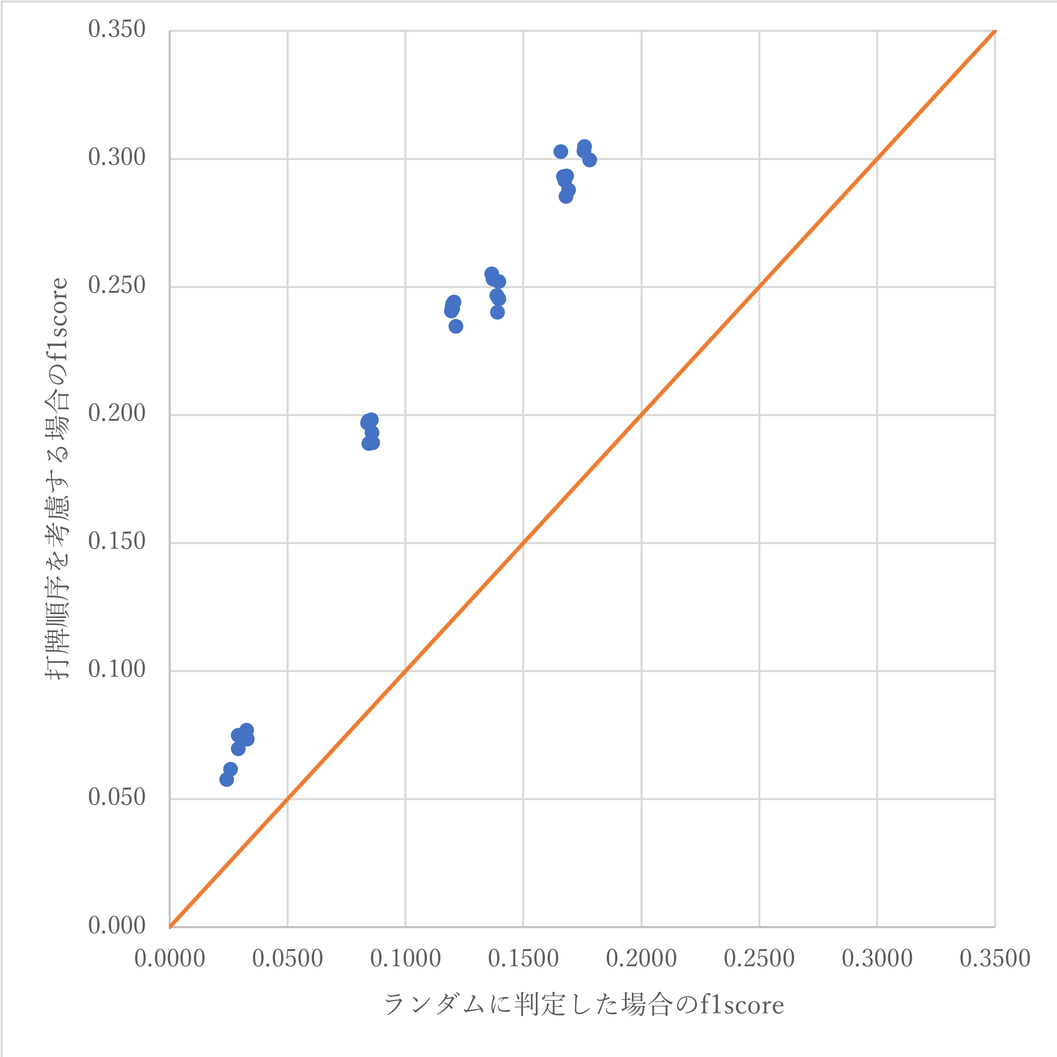

図 6-5. ランダムに判定した場合と打牌順序を考慮する場合の f1score の比較

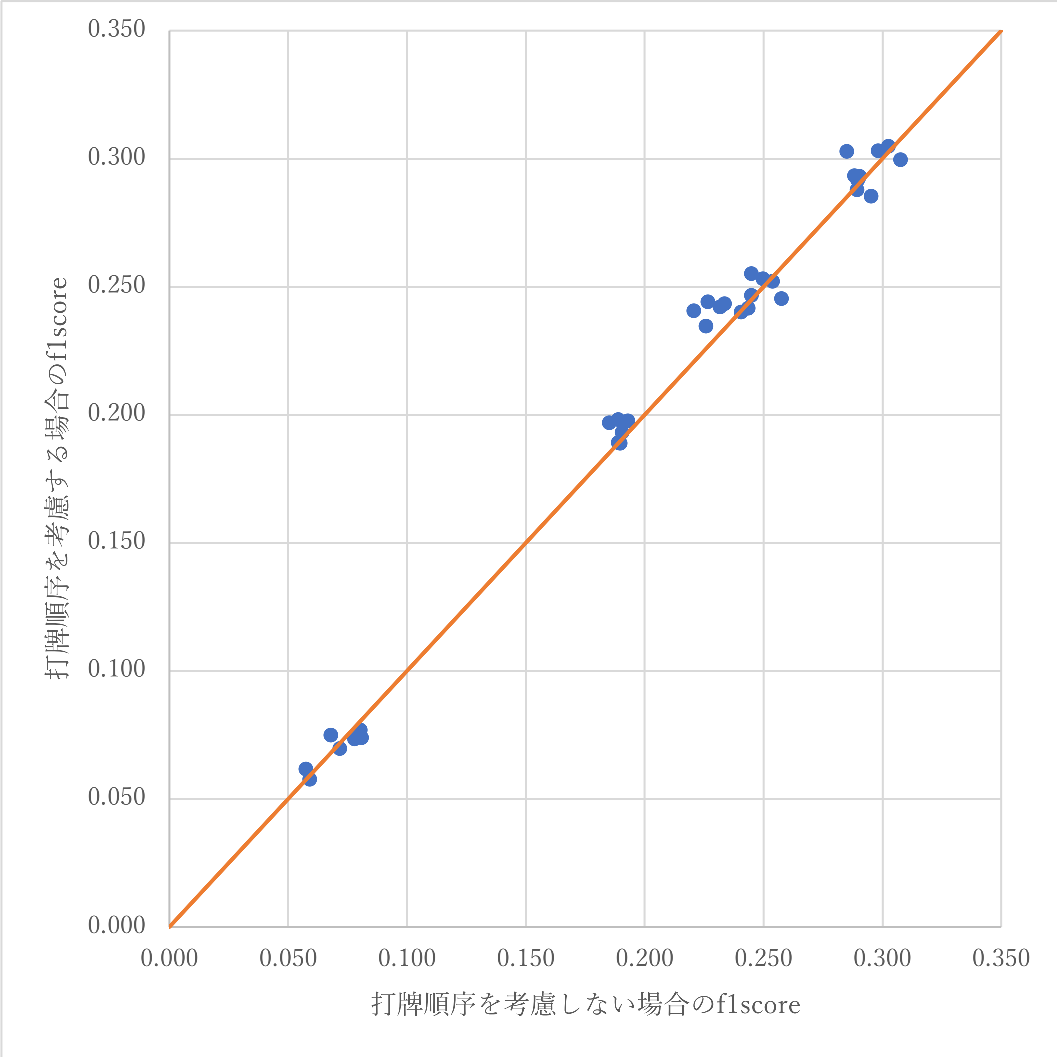

図 6-6. 打牌順序を考慮しない場合と打牌順序を考慮する場合の f1score の比較

以上の図から、打牌順序を考慮しない場合と打牌順序を考慮する場合はランダムに判定した場合より全ての牌において待ち牌予測の精度が高くなることが分かった。

また打牌順序を考慮する場合と打牌順序を考慮しない場合とを比較すると、いくつかの牌においては打牌順序を考慮しない場合の方が待ち牌予測の精度が高いが、全体的に見ると、微差、打牌順序を考慮する場合の方が待ち牌予測の精度が高くなっていることがわかる。

6.3 打牌回数を 10 回以上に制限した場合における打牌順序を考慮しない場合と考慮する場合の比較

・打牌順序を考慮しない捨て牌（打牌 10 回以上） ＋ 副露
・打牌順序を考慮する捨て牌（打牌 10 回以上） ＋ 副露
を入力として、それぞれ学習を行った。結果は以下のようになった。

表 6.7 打牌順序を考慮しない場合（打牌回数 10 回以上）の評価

牌	precision	recall	f1score
1 萬	0.108	0.840	0.191
2 萬	0.142	0.810	0.242
3 萬	0.160	0.787	0.266
4 萬	0.187	0.783	0.302
5 萬	0.200	0.772	0.318
6 萬	0.182	0.799	0.297
7 萬	0.156	0.801	0.261
8 萬	0.136	0.827	0.234
9 萬	0.108	0.831	0.191
1 筒	0.111	0.793	0.195
2 筒	0.138	0.804	0.235
3 筒	0.154	0.799	0.258
4 筒	0.195	0.750	0.310
5 筒	0.192	0.800	0.310
6 筒	0.187	0.781	0.301
7 筒	0.162	0.777	0.268
8 筒	0.146	0.803	0.247
9 筒	0.097	0.847	0.174
1 索	0.103	0.859	0.184
2 索	0.143	0.803	0.242
3 索	0.156	0.782	0.260
4 索	0.177	0.809	0.290
5 索	0.186	0.820	0.303
6 索	0.183	0.808	0.299

7 索	0.150	0.813	0.254
8 索	0.141	0.809	0.241
9 索	0.107	0.843	0.189
東	0.034	0.853	0.066
南	0.034	0.864	0.065
西	0.026	0.847	0.051
北	0.032	0.844	0.061
白	0.038	0.867	0.073
発	0.045	0.810	0.085
中	0.044	0.832	0.083

表 6.8 打牌順序を考慮する場合（打牌回数 10 回以上）の評価

牌	precision	recall	f1score
1 萬	0.115	0.842	0.202
2 萬	0.138	0.855	0.238
3 萬	0.160	0.817	0.268
4 萬	0.200	0.769	0.317
5 萬	0.197	0.820	0.318
6 萬	0.187	0.807	0.304
7 萬	0.153	0.821	0.258
8 萬	0.146	0.817	0.248
9 萬	0.115	0.844	0.203
1 筒	0.107	0.858	0.191
2 筒	0.150	0.817	0.254
3 筒	0.165	0.805	0.274
4 筒	0.178	0.824	0.293
5 筒	0.196	0.815	0.316
6 筒	0.190	0.796	0.307
7 筒	0.157	0.807	0.263
8 筒	0.149	0.821	0.252
9 筒	0.116	0.848	0.203

1 索	0.116	0.836	0.203
2 索	0.149	0.819	0.252
3 索	0.158	0.798	0.263
4 索	0.183	0.810	0.298
5 索	0.188	0.805	0.305
6 索	0.189	0.805	0.306
7 索	0.173	0.777	0.284
8 索	0.148	0.815	0.250
9 索	0.114	0.862	0.202
東	0.036	0.847	0.069
南	0.035	0.865	0.068
西	0.029	0.868	0.057
北	0.032	0.888	0.062
白	0.041	0.873	0.079
発	0.039	0.896	0.075
中	0.040	0.868	0.076

以上のデータから
・打牌順序を考慮しない場合（打牌回数の制限なし）と打牌順序を考慮しない場合（打牌回数 10 回以上）の比較
・打牌順序を考慮する場合（打牌回数の制限なし）と打牌順序を考慮する場合（打牌回数 10 回以上）の比較
をまとめたものが以下の図である。

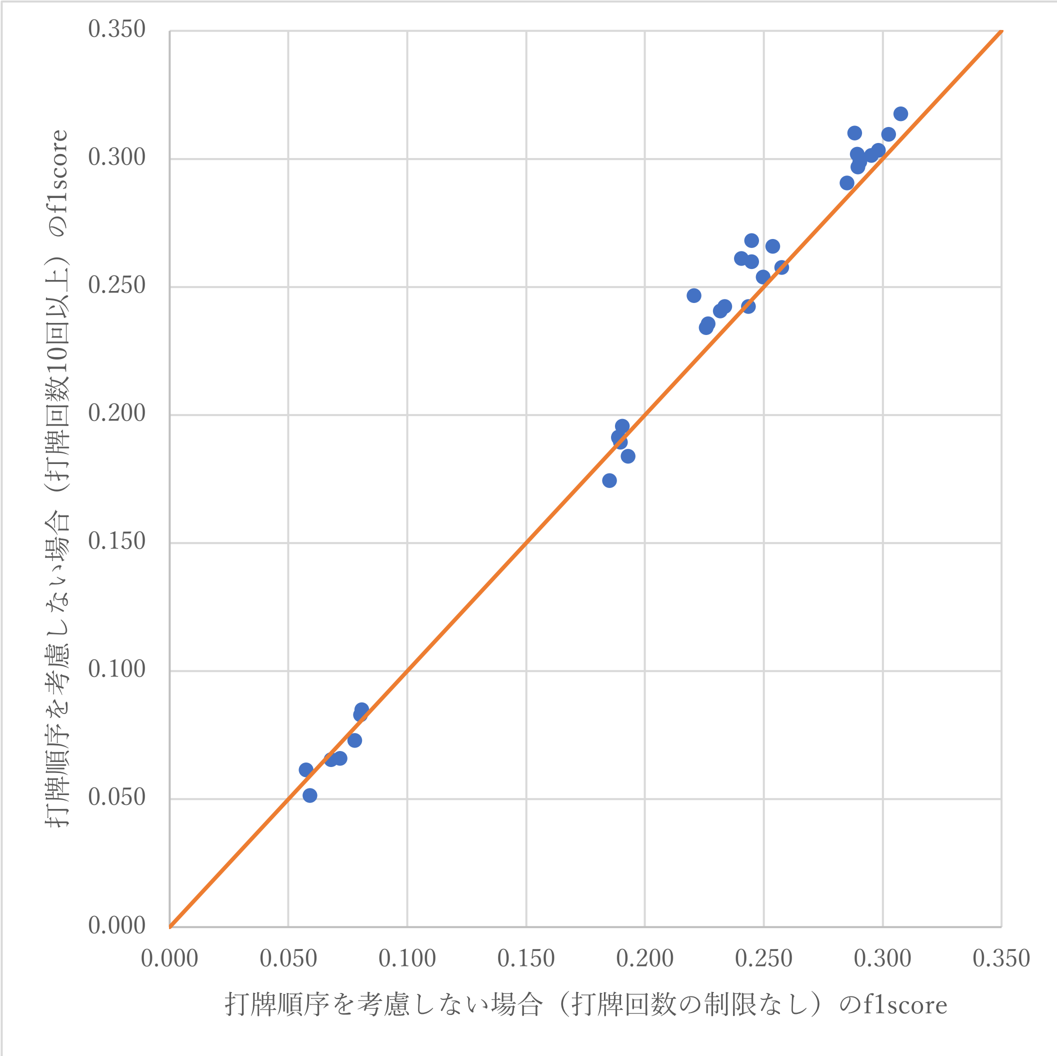

図 6.9 打牌順序を考慮しない場合（打牌回数の制限なし）と
打牌順序を考慮しない場合（打牌回数 10 回以上）の f1score の比較

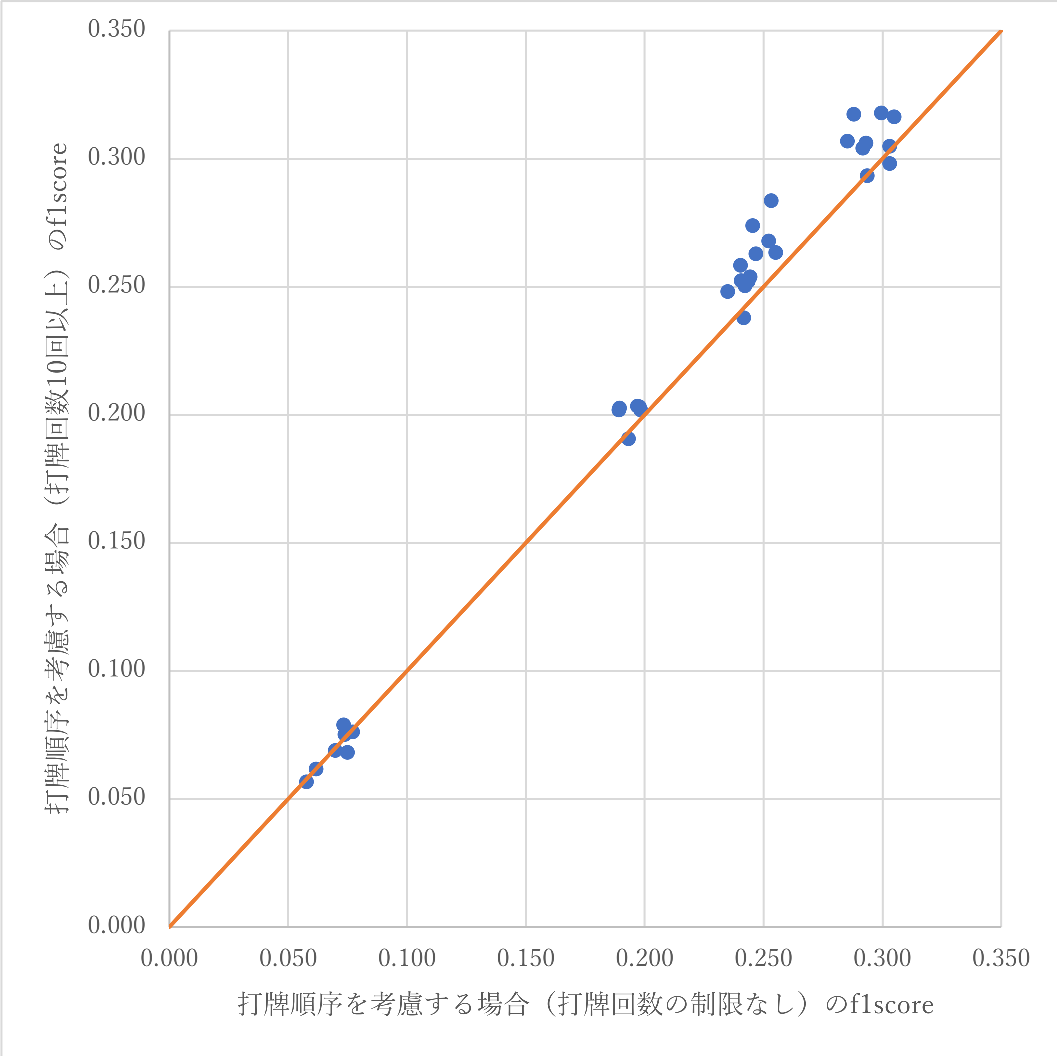

図 6.10 打牌順序を考慮する場合（打牌回数の制限なし）と
打牌順序を考慮しない場合（打牌回数 10 回以上）の f1score の比較

打牌順序を考慮する場合・しない場合に関わらず、打牌回数を 10 回以上に制限した場合、いくつかの牌において例外はあるものの、全体的にみると打牌回数の制限をしない場合より微差精度が上っている。

打牌回数を 10 回以上に制限すると、f1score は打牌順序を考慮しない場合において平均 0.00595 、打牌順序を考慮する場合において 0.00901 上昇した。

当初の予想通り、打牌順序を考慮する場合の方が f1score の伸びが良くなっている。

打牌回数を 10 回以上に制限した場合における、打牌順序を考慮しない場合と打牌順序を考慮する場合の f1score の比較が下図である。

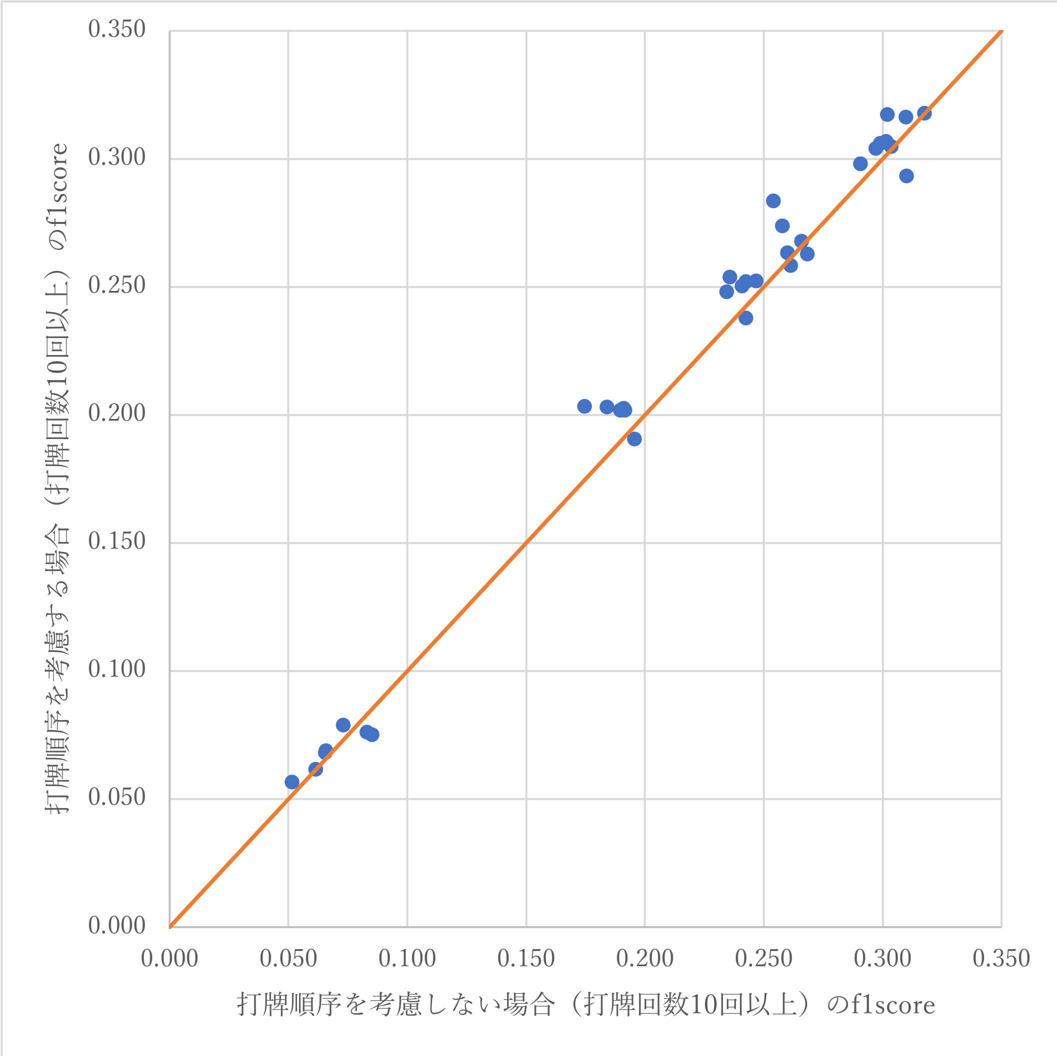

図 6.11 打牌順序を考慮しない場合（打牌回数 10 回以上）と
打牌順序を考慮する場合（打牌回数 10 回以上）の f1score の比較

6.4 考察

打牌の順序を考慮する捨て牌を入力として学習を行えば、考慮しない場合を入力として学習した場合と比較して飛躍的に良い予測の精度が出ると考え、実験を行ったが、実際は平均で見ると僅かに良くなったばかりで、牌によっては精度が下がっているものもあった。

この原因としては、捨て牌の打牌のすべての順序関係を拾ってしまうことによって、学習においてノイズとなるような情報も拾っていることなどが考えられる。例えば第 1 打と第 16 打の打牌の関係に重要な情報が含まれることは殆どない。

また打牌の回数を 10 回以上にするという制限を設けた場合、打牌の順序を考慮する場合の方が考慮しない場合よりも、予測の精度の上昇幅が大きくなると予想したが、実際は精度の上場幅の差は僅かなものだった。牌の番号によっては打牌回数に制限をかけることで予測の精度が下がっているものもあった。

この原因も上記と同じような理由が考えられる。

7. まとめと今後の課題

学習に用いる捨て牌情報について、打牌の順序を考慮することで、打牌の順序を考慮しない場合と比較して、僅かに待ち牌の予測の精度を向上させることに成功した。

また打牌の回数が 10 回以上のサンプルのみを用いて、打牌の順序関係を考慮する場合・考慮しない場合でそれぞれ学習を行ったところ、打牌の回数を制限していないサンプルで学習した場合と比較して、僅かに待ち牌の予測精度が向上した。そのときの予測精度の上昇幅は打牌順序を考慮する場合の方が考慮しない場合と比較して僅かに大きくなった。

今後の課題としては、打牌の順序を考慮した捨て牌の情報に重み付けを行う手法の開発がある。

例えば、第 1 打と第 16 打のように、打牌された時間の差が大きい場合は、情報の重みを小さくする、序盤の打牌の情報より終盤の打牌情報の重みを大きくするなどの手法が考えられる。

参考文献

[1] 我妻敦,原田将旗,森田一,古宮嘉那子,小谷善行. SVR を用いた麻雀における捨て牌危険度の推定. 報告ゲーム情報学(GI), Vol. 2014-GI-31, No. 12, pp. 1-3, 2014-03-10

[2] 栗田萌,保木邦仁. 麻雀における他家の手牌と待ちの予測に基づく放銃確率推定. 報告ゲーム情報学(GI), Vol. 2017-GI-38, No. 5, pp. 1-8, 2017-07-08

[3] 矢ノ口裕貴,篠埜功. ニューラルネットワークを用いた麻雀の捨て牌危険度推定. 第 79 回全国大会講演論文集, Vol. 2017, No. 1, pp. 469-470, 2017-03-16

[4] 水上直紀,中張遼太郎,浦晃,三輪誠,鶴岡慶雅,近山隆. 多人数性を分割した教師付き学習による 4 人麻雀プログラムの実現. 情報処理学会論文誌, Vol. 55, No. 11, pp. 2410-2420, 2014-11-15

[5] 北川竜平,三輪誠,近山隆. 麻雀の牌譜からの打ち手評価関数の学習. ゲームプログラミングワークショップ 2007 論文集, Vol. 2007, No. 12, pp. 76-83, 2007-11-09

[6] オンライン対戦麻雀 天鳳 / ログ, https://tenhou.net/sc/raw/

[7] 天鳳の牌譜を解析する(1), https://blog.kobalab.net/entry/20170225/1488036549

[8] 天鳳の牌譜を解析する(2), https://blog.kobalab.net/entry/20170228/1488294993

[9] 天鳳の牌譜を解析する(3), https://blog.kobalab.net/entry/20170312/1489315432

[10] 天鳳の牌譜を解析する(4), https://blog.kobalab.net/entry/20170720/1500479235