

AI トレーダー コンピュータビジョン編

AI trader computer vision edition

河合 継¹ 小澤 昂^{1,2}

Kawai Kei, Ozawa Takashi^{1,2}

¹クリスタルメソッド株式会社

¹Crystal method co.ltd

²東京工業大学/数学科 4年/数理ファイナンス専攻

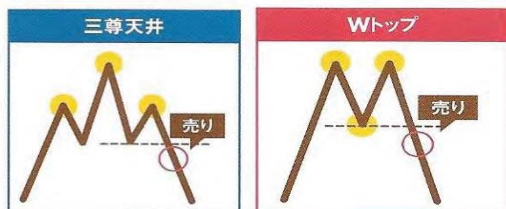
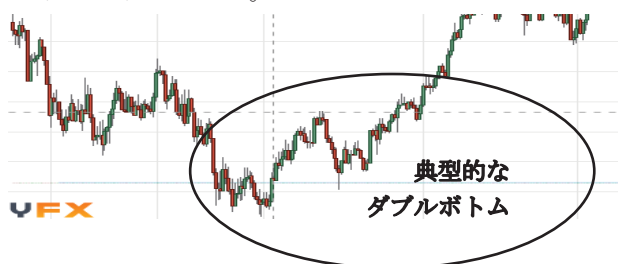
²Tokyo Institute of Technology Department of Mathematics

1. はじめに

1-1. トレーダー

2010 年末からトレーダーになる為、半年ほど FX スクールに通って取引の勉強をしていた。当時やっていたのは、全くの裁量取引で、人工知能とはかけ離れたもの。毎日 FX のチャートを 5 通貨ペアほどプリントアウトして、日足に決められた線を引き、傾向分析を行い、15 分足でその日のレンジなどを確認したり、支持線・抵抗線などを定めたりしていた。

そんな中で勝率が高いケースがいくつか存在していた。“三尊”・“ダブルトップ”・“フラッグ”などチャートのロウソク足の並びに名前を付けたものを目印にトレードしていた。



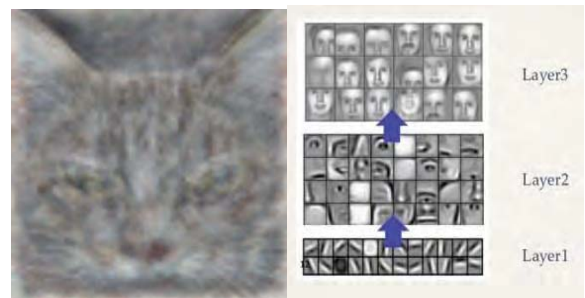
詳しいトレード手法は別の機会としたいが、ルールが守られている限りは 20 勝 1 敗というような成績を残すことが可能で、すぐに資産は 2 倍ほどに成長した。

ただし、そのトレードは非常に重労働であったし、見ているのは単純な値動きだった。私は自分がこの

世界に役に立っていない人間だと思えて来て、トレードのモチベーションを失った。自動で同じようなトレードが出来ないかという欲求には駆られていた。

1-2 AI や Deep Learning について

2012 年に Google が猫を判定したというのはあまりにも有名なニュースだった。ニューラルネットの中間層に猫の特徴をあらわした画像が出ており、人間の顔の特徴が出てきていたのがとても印象的だった。Deep Learning の利点は人間が気付かないような特徴も分析結果として現れるという事だった。ここから世間で語られていない全く新しい取引方法が見いだせる可能性があると考えた。



私もこの後、機械学習に触れ始めるようになっていたが、チャートの特徴分析を行う事によって、一度諦めたトレードが再度できるようになるのではないかと考え始めた。目で見て値を判定するという事にこだわったので、一番初めに行った検証は、画面上の画像としてのレートを表示して文字情報としてトレード判定条件に利用できると考えて MNist 数字を判定する事で、機械学習の道に入った。

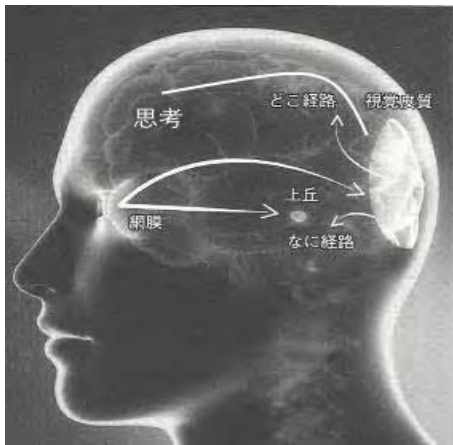
1-3 AI トレーダー

次は勝率の高いトレードを再現するために、レートの数字のみでなくチャートを目で見て判定してい

るように振舞うアプリケーションを開発できないか検討を始めた。

人間の目の働きは網膜に映った情報が神経を経て脳の後ろの視覚野に到達する。その後それは何か？などの判定に入るのだから、認識判断速度としては少なくとも300ミリ秒以上通常400ミリ秒ほどかかる。ただ、それよりもショートカットされた中脳の上丘（じょうきゅう）でも情報は認識される。視覚認識に15ミリ秒ほどと非常に早く、100ミリ秒で反応を行える。その経路は熟練したスキルを持つものが無意識に利用する経路である。

AIトレーダーの実現方法は人間の認知の仕組みと機械学習の利点を組み合わせようと考えた。上丘の動きを模した部分、特徴を認識し学習する前頭葉にあたる部分、前意識と呼ばれる、勘などで反応ができる熟練性である。前頭葉では情報を本などから入力することにより学習し、学習したことを利用することによって熟練する。



2. 準備

2-1 準備 1

・チャート画像取得プログラム MQL4 C++
メタトレーダーのチャートをコマ送りするプログラムを作成することにより、2005年1月から2016年末までのUSD/JPY・15分足チャート画像30万枚をスクリーンショットとして保存した。同時にチャートの値動きと時刻情報をラベリングした。+-5 PIP以上の値動きをそれぞれラベリングし、5 PIP以内は同じラベルとした。

例 12005#01#17#23#00.png -5 以下-
22016#07#27#10#00.png 5 以上+
32009#05#05#14#30.png 5PIP 以内
ラベル年#月#日#時#分

・画像サンプル



・チャート画像加工プログラム

Png形式で保存されたデータに対し、前処理として、ビット深度やサイズ変更をするようなプログラムを作成した

・Chainer CNN 学習プログラム Python

Pythonで畳み込みニューラルネットワーク(3章で後述)を利用し、上記の画像を学習するプログラムを作成した。

・Chainer 学習結果返却プログラム Python

上記で学習した学習済みデータセットに2016年中の画像を入力として与えることで、売りが買いか何もしないかを返却するプログラムを作成した。

・チャート価格等取得プログラム MQL4

対象の画像時刻(チャート左端)チャートの現在時刻(チャートの右端)次の足時刻(チャートの右端+1 右端の終値、右端+1の終値、両者の価格差をすべての15分足で取得した。

・結果検証プログラム Python Postgres Excel

学習済みデータセットが返却した売買情報と上記のデータベースに対してSQLを発行し、Excelにわかりやすいようにまとめるプログラム

・ハードウェア

自作PC i7920 メモリ16G Ubuntu 16.04 GTX1060 6G

2-2 準備 2

ショートカット経路のあるあまり思考することなく反応する経路について、高速に作動するVisualC++を利用し、与えたテンプレートに対して、雰囲気似ている事を検知できるプログラムを作成した。

2-3 準備 3

通常人間は、本やその他の情報を入力として勉強し、スキルを身につけていく。今回はMT4のハーモニックパターン検出システムを調査した

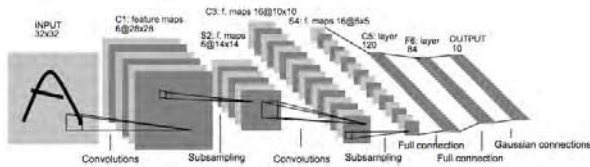
3. 実証分析

Deep Learning の利点は人間が気付かないような特徴も分析することが出来て結果にあらわれるということ。思考部分であり、今回の研究メインである。

3-1 畳み込みニューラルネットワークでのチャート
チャートを見て、視覚情報をもとに売買するトレードを模するために、畳み込みニューラルネットワークモデル(CNN)を採用した。

CNN は視覚野の構造を取り入れた多層パーセプトロンモデルであり、今回の分析目的に最も適していると考えられる。

局所領域（受容野）の畳み込みとプーリングを繰り返す多層ネットワークであり、段階的にスケールを変えながら、局所的な相関パターンを抽出することが可能である。



3-1-1 各種ネットワークでの実行結果

Chainer-imaget というカラー画像を分類するプログラムがあるが、優秀な各種ニューラルネットワークが定義されており入れ替えて利用できる。

画像入力数も何種類か実施したが、10年分30万画像に関してはボリュームがありすぎて、現状の構成ではメモリエラーがでたり、lossの値が収束しなかったりなどで、あまり多すぎる画像は控えるようにした。

・実施結果表

	Net	入力数	Epoch	収益 (PIP)	取引回数	勝率
1	NIN	32000	10	112	52	52
2	GoogLeNet	300000	10	15	2	100
3	Alex	300000	10	1300.50	5764	50
4	Alex	2015年データ 24194	70	1350	5098	52.5

4に関しては accuracy（学習率）97%に到達

いずれのケースも負け越すことは無いので多少の特徴分析は行えているようだが、明確に表れていないためこの後のフェーズでは全体の検証と工夫を行った。

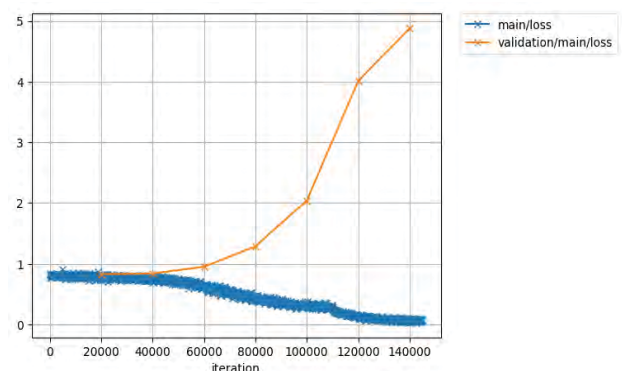
3-1-2 各年度での検証結果

2006年中の画像を入力し、2007年のデータをテストサンプルとしてモデルを学習した。2008年から2016年の期間のデータでバックテストを実行した。

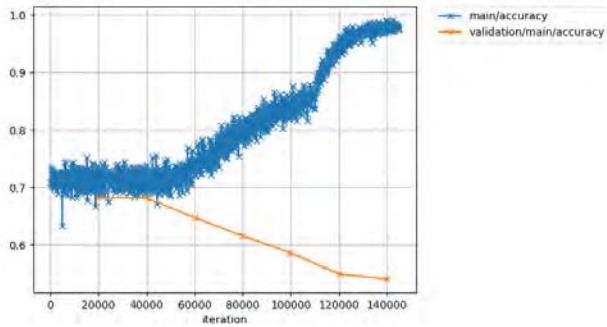
検証年度	取引数	勝率	損益 (PIPS)
2008年	7835	52.1%	-2.4
2009年	7726	52.4%	-4.84
2010年	7690	53.7%	3.659
2011年	7581	51.0%	-0.836
2012年	7817	50.1%	-3.194
2013年	7675	49.7%	-9.862
2014年	7555	49.6%	-10.802
2015年	7544	50.7%	5.107
2016年	6710	49.1%	-14.271
合計	68133	51.0%	-37.439

勝率50%ほどの結果になっている。また負け越している年度もあり有効なトレードの結果ではないと思われる。勝率が51%なので、誤差の範囲かもしれないが、少しは特徴の分析が出来たと思われる。

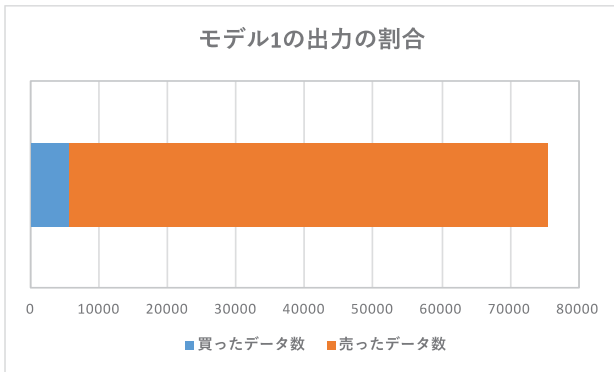
・損失関数の返却値推移



・ 正答率の推移



モデル1の出力の割合



・ 値上がりしているデータと値下りしているデータの割合はほぼ 1:1 にも関わらず、モデルの出力に大きな偏りが見られる。

・ 買いデータの特徴が上手く抽出されていないと考えられる。

3-1-3 ラベリング変更と畳み込み対応

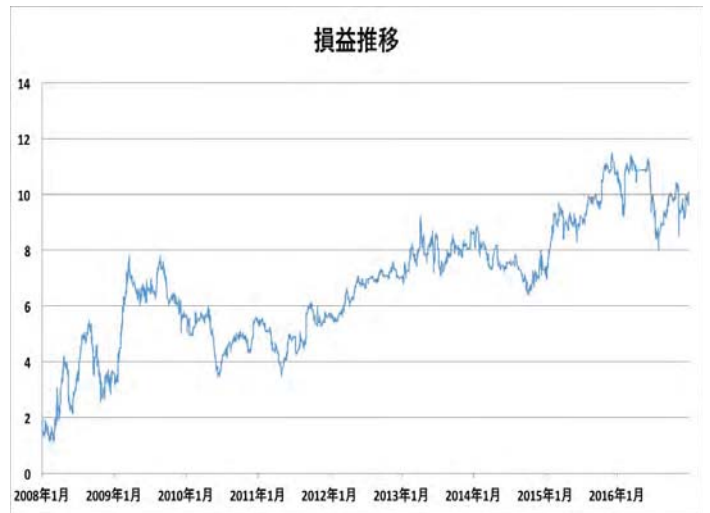
今の画像とラベリングでは違いをもたらす違いが無かったと仮定した場合、準備時に 5PIP 以上の差をラベリングしたが、それを倍の 10PIP に変更した場合どのようなことになるのか検証した。

結果としては、10PIPS の値幅画像での検証は勝率 50.1%で損益合計の推移も順調に増えている。もう少し勝率が上がればもしかしたら実用化も視野に入ると考えた。

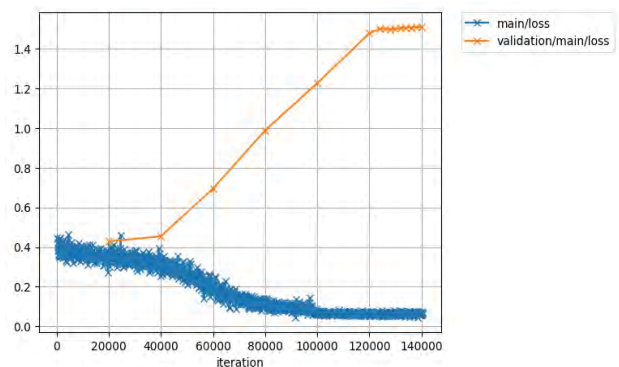
実施結果表

検証年度	取引数	勝率	損益
2008 年	680	51.5%	1.55
2009 年	646	49.5%	2.02
2010 年	593	50.1%	-0.20
2011 年	706	50.7%	0.28
2012 年	666	52.3%	1.31
2013 年	619	52.0%	1.70
2014 年	600	46.3%	-1.56
2015 年	613	48.5%	3.53
2016 年	611	49.3%	-1.05
合計	5734	50.1%	7.59

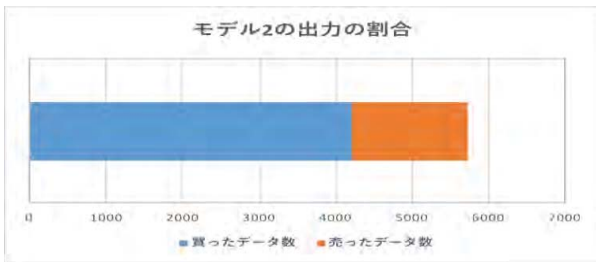
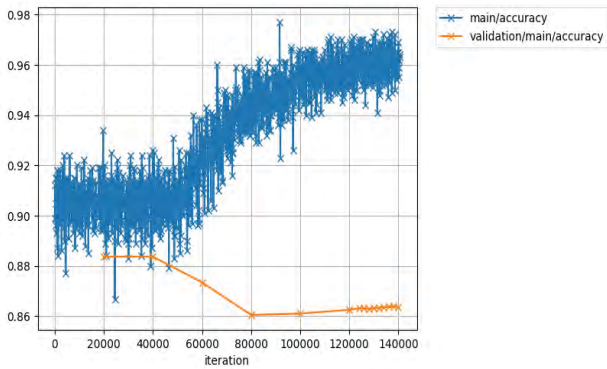
損益推移



・ 損失関数の返却値推移



・ 正答率の推移



出力の偏りはある程度改善されたが、理想の出力の割合は 1:1 のため、まだ改善の余地が見込まれる。より性能の高い GPU を用意し、学習データ数と中間層のフィルタ枚数を増やす必要がある。

3-2 ショートカット回路

ショートカット回路に入力した画像は以下のダブルトップ画像で、各種パラメータ設定を行う事である程度の精度で似ている画像の抽出が行えた。

テンプレート画像



選択後の様子



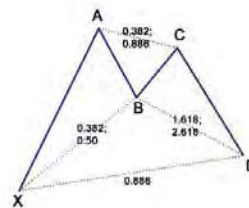
3-3. MT4 のハーモニックパターン検出 EA

ハーモニックパターンの種類 “三尊”・“ダブルトップ”・“フラッグ”などと近い、“ガートレー”、“バット”と呼ばれる各種パターンが成立したところからの反転をトレードする。

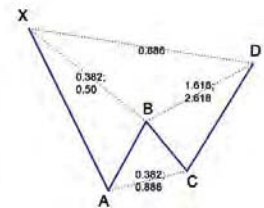
例 バットパターン

The Bat Pattern

Bullish



Bearish



メタトレーダーを利用した EA でハーモニックパターンが成立した時刻がわかるものがあったのだが、私の方でうまく成績がだせず、学習素材として利用できなかったのも、利用できるものを探して利用する。今回のメインテーマではないので、こちらはここまでとしたい。

4. 結論

人間の仕組みを模した人工知能を利用し、Deep Learningの特徴分析能力を利用することによりAIトレーダーは構築できるかどうかを検証した。

1. 前頭葉を模した特徴分析

勝率 53% 勝ち PIP は少ないがある程度、特徴は分析できたと考えている。

過去の研究会では学者をツールとして使ってくださいというお話もあり、お言葉に甘えて本研究会に出席されている、諸先生方のお知恵をお借り出来れば、もっと素晴らしいものになると確信している。

2. 上丘（前意識）を模した判定

VC++で開発することにより、高速動作し、ぱっと見似ているという判定までは行う事が出来た。

3. チャートの形による学習

ハーモニックパターンの各分類でのトレード実行結果が芳しくないので、もう少し相応しい学習素材をさがし学習させることでAIトレーダーに利益が出るトレードを学習させていきたい。

4. 熟練行程でのフィードバック

1で解析した中間層の画像を2の入力判定として利用することにより素早く判定できる仕組みとできることが分かった。

現在の技術でも実際に目で見て考えるような、AIトレーダーは構築出来るのではないかと考えております。また別の分野への転用の幅が広い技術だと思います。

謝辞

東工大の学生の皆様、色々教えていただきありがとうございました。またSIG-FINの先生方にあたたかいお言葉を頂き、私のような至らぬものがこのような研究を行えたこと、非常に誇りに思っています。

参考文献

- [1] ジョン・コーツ, 訳 小野木明恵, トレーダーの生理学
- [2] 新納 浩幸 Chainerによる実践深層学習
- [3] ラリーペサント フィボナッチ逆張り売買法
- [4] Y. LeCun, L. Bottou, Y. Bengio and P. Haffner: Gradient-

Based Learning Applied to Document Recognition,
Proceedings of the IEEE, 86(11):2278-2324, 1998.

- [5] 株お得意ベストセレクション 普遊舎