機械学習とプログラミング教育

箕原 辰夫[*1]
Email: minohara@cuc.ac.jp

*1: 千葉商科大学政策情報学部

◎Key Words　　機械学習、深層学習、プログラミング、Python

1. はじめに

卒業研究などにおいて学部の学生でも、プログラミング教育の成果として機械学習のライブラリを用いて、予測や判断などを行なう研究がされています。また、2019 年現在、書店には、Python を使って機械学習や深層学習をするような書籍が溢れ返っている状況です。しかしながら、scikit-learn[(1)]などのライブラリは統計的な手法による学習ですし、TensorFlow[(2)]などの深層学習のライブラリは、ニューラルネットワークによる学習に過ぎません。最近、人工知能と共に、さも新しい技術のように喧伝されていますが、元々は 1980 年代～1990 年代の人工知能の研究や神経網研究に基づくもので、その時代の成果が、卑近に使えるようになったものだけではないかという印象が拭えません。機械学習や深層学習は、いわゆる常識（Common Sense）という知識ベースがない形で利用した場合、単なる統計解析、画像や音声認識などにしか利用することができません。またプログラミング教育としても、データを集めて、ライブラリで用意されている関数を呼び出すだけになっており、解析結果の正当性について学生が判断することができない状況です。昨年度にそのような卒業研究を行なった学生の例から、機械学習を題材として学生のプログラミング能力・データ解析能力を向上させるには、どうするべきかを考察します。

2. 機械学習と深層学習

2.1 scikit-learn による機械学習

scikit-learn[(1)]は、Python のオープンソースの機械学習のライブラリです。図 1 のような分類・回帰・クラスタリング・データの次元圧縮などを学習させ、学習結果を用いて、別のデータに対して、適用させることが可能になっています。実際に、それぞれのアルゴリズムを適用させるために、様々な書籍が出ていますが、Web 上では、Qiita の「全手法の解説・実装してみた」[(2)]の記事が有名です。

scikit-learn については、例えば、“drink”の動詞の後には“beer”や“wine”が置かれることが多い（確率的に何%以上である）ということを学習し、“drink”が来たときに、その後に“beer”や“wine”を候補として入力するサポートをする[(3)]ということはできます。しかし、“drink”と“beer”の結びつきについて、文法的な知識があるわけでもなく、その動作の意味も理解している訳ではありません。日本語で文章を入力するときに、「しかし」が来たら、「ながら」が後に来る確率が高く、自動的に「ながら」を補完して、「しかしながら」にするという卑近な例を考えれば、統計的な機械学習によってコンピュータが何を学習しているかは、自ずと知れてしまいます。

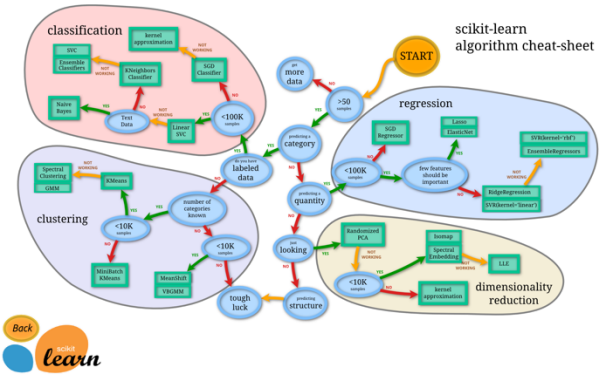

図 1 Scikit-learn の cheat sheet map[(4)]

scikit-learn は、教師データを使って（使わない場合も可能ですが）、目的のデータ間の関係の統計操作により、新たなデータの分析を行なうことしかできません。統計情報から、新たな知見を見つけるのは、統計的因果探索[(5)]を行なう必要があります。たとえば、LiNGAM 分析[(5)]などについても扱い、その有用性を確かめる必要があります。実際に、Qiita では、そのような有用性を確かめる記事[(6)]も出ていますが、学部レベルの学生がこのような分析まで利用して、対象のデータ分析を行なうことは、ほとんどないでしょう。

つまり、scikit-learn を使った機械学習をプログラミングできるようになったとしても、コンピュータは、統計的な情報から学習し、対象となるデータの分析が可

能となりますが、それをプログラミングしている学生は、まったくデータ分析からの知見が得られない状態で留まることになります。また、対象となる現象を分析するのに、どのようなデータを取捨選択して、機械学習させるかについても、その現象についての専門家でないと、見当外れなデータを学習するだけに留まることになります。

2.2 TensorFlow による深層学習

TensorFlow[7]は、Google が開発したオープンソースのニューラルネットの多層モデルを利用した深層学習（deep learning）のためのライブラリです。ニューラルネットによる学習は、メディア認識の学習では有効に用いられています。例えば、顔認識・音声認識・画像認識など、従来のアルゴリズムでは識別できないような用途においては、ニューラルネットに学習させることによって、別のメディアに対して認識させることが可能になります。実際に、TensorFlow をリアルタイムに識別させて使うためには、GPGPU や専用のチップが必要になります。例えば、iPhone に使われている A11 や A12 CPU には、専用のニューラルエンジンと呼ばれる専用のハードウェアが、SoC として組み込まれています。そのため、GPU 性能の低いコンピュータで、TensorFlow を使う場合は、低速で使い物にならないことが多い状況です。所謂ゲーミング PC と呼ばれる高性能 GPU を積んだコンピュータでないと利用することができません。

2.3 現在の機械学習と深層学習の限界

機械学習を用いて、Python でプログラミングできるようになったとしても、その結果を統計的に分析する能力がなければ、新たな知見は得られませんし、機械学習そのものが与えられたデータに対しての学習・分析に限られています。1980 年代の人工知能の研究でなされてきた、意味ネットワークの構築が可能になる訳ではありません。意味ネットワークという常識がない状態で、与えられた対象データだけを学習しても、人工知能が作れる訳ではありません。また、深層学習もメディアを認識するのが主な用途になっており、視覚・聴覚系の認識はできますが、そこから、意味を抽出する部分は、別途構築する必要があります。意味ネットワークあるいはオントロジーの構築と検索については、長い期間研究されていますが、たとえば、意味ニューラルネットワーク[8]などの研究分野では、ニューラルネットを使って意味ネットワークの修正もできる成果もでていますが、前面の機械学習・深層学習の結果から、後段の意味ネットワークを構築することが可能になって、初めて人工知能をプログラミングで作り出したと言えるのではないでしょうか。

3. 株価予測を行なってみた例

ここでは実際に、Python でのプログラミングもそこそこの学生が、2018 年度に卒業研究として機械学習のライブラリを使って、株価予測をしてみた経緯について報告します。

3.1 pandas によるデータ・スクレイピング

最初は、まず Web 上から株価のデータを取得するところで躓いていました。日経 255 先物取引の株価のデータを欲しかったようなのですが、pandas-datareader[9]という Python 用のデータ・スクレイピングのライブラリのサンプルプログラムが対応していなかったのですが、それを変更して、データを取得するまでに至りました。また、そのデータから、ローソク足データを P Python の matplotlib ライブラリで表示するところまで漕ぎ着けました。

3.2 株価の推移の学習

scikit-learn の機械学習を使って、株価の推移を予測することを始めたのですが、先行した研究の結果[10]があまり芳しくなく、結局 50%～60%の確率でしか予測できないことを学んで、これを使うことを諦めました。

3.3 MCMC 法の利用

意味ネットワークの構築に一役買うのがベイジアン・ネットワーク[11]です。そのため、ベイズ統計を使った推論[12]を行なうことを試みました。ベイズ推論である MCMC（マルコフ連鎖モンテカルロ法）を使った Python のライブラリとして、PyMC[13]があります。このライブラリを使った解説記事[14]を参考にして、株価の予想をしてみましたが、結局、わかった知見は、Nasdaq や Dow などが日経 255 に先行しており、日経 255 が遅れて同じような株価の推移をしているということだけでした。

4. おわりに

現状では学部の学生は、機械学習や深層学習のライブラリをお試しで使って終わりというような研究成果しか挙げていないように思えます。それを前段において、意味ネットワークを構築し、常識を得ていくようなプログラミングまで作り上げないと、真の人工知能の構築・運用には繋がらないでしょう。また、データ解析についても、数学的な素養がないと難しいように思えます。これから、この分野が、更に発展させて、学部の学生でもそのような人工知能を作り上げることができるようなカリキュラムを考えていく必要性があるように思えます。

参考文献

(1) David Cournapeau, scikit-learn, 2007, https://scikit-learn.org/
(2) すぐる (小川雄太郎) @sugulu,「【機械学習初心者向け】scikit-learn「アルゴリズム・チートシート」の全手法を実装・解説してみた」, 2017, https://qiita.com/sugulu/items/e3fc39f2e552f2355209
(3) 岡崎直観,「単語の意味をコンピュータに教える」, IWANAMI Data Science Vol. 2 統計的自然言語処理 ―ことばを扱う機械 , pp.47-61, 岩波書店, 2016.
(4) Scikit-learn, Choosing the right estimator, https://scikit-learn.org/stable/tutorial/machine_learning_map/
(5) 清水昌平,『統計的因果探索』, 機械学習プロフェッショナルシリーズ, 講談社, 2017.
(6) @m__k,「LiNGAM モデルの推定方法について」, 2018, https://qiita.com/m__k/items/bd87c063a7496897ba7c
(7) Google Brain Team, Tensor Flow, 2015, https://www.tensorflow.org
(8) Wikipedia, Semantic neural network, https://en.wikipedia.org/wiki/Semantic_neural_network
(9) pandas-datareader, https://pandas-datareader.readthedocs.io/
(10) Kotaro Kamata,「機械学習で株価予測～scikit-learn で株価予測①～④～」, 2018, https://kkmax-develop.com/machinelearning-scikit-learn-1/
(11) Wikipedia, ベイジアンネットワーク, https://ja.wikipedia.org/wiki/ベイジアンネットワーク
(12) 伊庭幸人,「ベイズ超速習コース」, IWANAMI Data Science Vol. 1 ベイズ推論と MCMC のフリーソフト, pp. 6-16, 岩波書店, 2015.
(13) The PyMC developments team, PyMC3, 2018, https://docs.pymc.io
(14) @crambon,「MCMC 初心者が pymc3 で株価の期待日次リターンを推定してみる」, 2018, https://qiita.com/crambon/items/8af9a52d4e1e91eaafb5