研究プロジェクト報告

非理工系学生向け機械学習／データマイニング教材開発

稲積 宏誠†,宮治 裕†

抄 録　現在、大学において AI やデータサイエンスを学ぶ環境整備が広く求められている。私たちの取組みでは、大規模データからの知識発見を、現実の問題解決への適用を最終目的としている非理工系学生を対象とした機械学習／データマイニングについての教育プログラムを検討した。その結果、一定のプログラミングの素養をもち、自ら機械学習アルゴリズムの実装が可能な学生向けの教材と、実装することのできるプログラミングの素養をもたない学生向けの教材の開発を試みた。本稿では、その取り組み概要を報告する。

キーワード　機械学習、データマイニング、非理工系学生

Development of Machine Learning/Data Mining Materials for Non-Science and Engineering Students

Hiroshige INAZUMI†, Yutaka MIYAJI†

Summary　Today, universities are being asked to create an environment for learning AI and data science. In our approach, we examined an educational program on machine learning/data mining for non-science and engineering students whose ultimate purpose is to apply knowledge discovery from large-scale data to real problem solving. As a result, we tried to develop teaching materials for students who have certain programming skills and can implement machine learning algorithms on their own, and for students who do not have programming skills to implement them. In this paper, we report the outline of our efforts.

Keywords : Writing center,　Operation,　Quality assurance,　Students' use,　Data analysis

1. はじめに

現在、AI ブームの根幹をなす大規模データからの知識発見への取組は、今ではあらゆる専門領域における必須のテーマとなっている。したがって、機械学習の理論的な理解や具体的な実装が可能な人材のみならず、AI や機械学習を特に専門としないが、それらの考え方を理解して実際の大規模データの解析を行う人材にとって必要な機械学習やそれを応用したデータマイニング技術の修得が可能な教育システムが必要となる。そのためには、文理の枠を超えた取り組みが必要となる。

そこで、大規模データからの知識発見を、社会への適用・応用を最終目的としている非理工系学生を対象とした機械学習／データマイニングについての教育プログラムを検討する。その結果、一定のプログラミングの素養により自ら機械学習アルゴリズムの実装が可能な学生向けの教材と、実装するまでのプログラミングの素養をもたない学生向けの教材の開発を試みる。前者については宮治が、後者については稲積が取組み、本稿では、その概要を紹介する。

† 青山学院大学　社会情報学部

2. プログラミング素養をもつ学生向けの教育プログラム

非理工系学生のなかで、「自ら機械学習アルゴリズムの実装が可能である学生向けの授業」（以下、本節にて本授業と記す）の作成と評価をおこなった。

なお、本取組みに先行して社会情報学部（以降、本学部）における科目の内容や授業環境を検討するプロジェクトがおこなわれ [1]、その検討結果を受ける形でカリキュラム変更が確定した。本取り組みは、この新カリキュラムにおいて、プログラミング素養のある学生向けの授業科目として位置づけられた授業の設定や内容について再検討・詳細設計をおこない、効果検証に関して再試行したものである。

2.1 背景および目的

本授業は、本学部のカリキュラムがより一層充実することを目指して、新カリキュラムにおける新規科目として設置されることを想定した。

本学部は「情報の高度な活用」を、すべての学生に必須の４つの力の内のひとつとして、カリキュラムポリシーの中で掲げている。実際に、「統計入門」や「コンピュータ実習」などが必修科目として設置され、高校時代に「文系」だった学生たちも必ず履修する。また、統計関連科目やデータ分析関連科目、数学関連科目、情報関連科目を履修することができ、データを処理・分析・活用する技能を身につけることもできる。それにより、実際に情報の高度な活用能力を生かす職業に就く学生も輩出している。

しかしながら、本プロジェクトが指導した 2018 年度当時の社会情報学部のカリキュラムでは、近年の機械学習やディープラーニングに関する技術者育成には充分とはいえなかった。「人工知能基礎」では、その時間的な制約から基本的に座学であり、基礎内容を学習するに止まってしまっている。そのため、実際の問題への適用方法や困難な点を理解することが難しい。もう一つの代表的な科目である「データマイニング」およびその演習は、効果的で実践的な内容を扱っているが、既存のソフトウェアの中での情報活用方法に限られてしまっている。これらの授業に不足している「実装」を意識し、機械学習に関する一連の技術的な手法を「経験」する授業が必要といえる。

以上のことから本プロジェクトでは、本学部における統計系科目や情報系・プログラミング系の科目が比較的得意な学生を対象として、実践的な授業を開発することを目的とした。この実践的とは、柔らかい言葉でいえば「自分でひととおり動かせる」ことを体験し、「必要な技術を自分で調べる」ことができる能力を身に着けることを意味する。つまり、履修後には機械学習に関する発展的な内容に対して、自力で取り組むことができる基礎力を養成する授業を開発する。

2.2 授業設定および設計

プロジェクトの目的を達成すべく、授業に関する条件を設定し、授業の具体的な設計をおこなった。その詳細について述べる。

2.2.1 本講義の位置づけと対象の設定

本講義は、学部カリキュラムにおける情報系科目群の発展的科目として配置し、その履修には条件を設けることとした。

まず、発展的な科目ではあるが、履修には強制力の低い選択科目として設置することを想定した。これによって興味を有し必要性を感じている者だけが履修することになる。このことは、単純に履修者数を増やすということではなく、モチベーションが低い受講生を減らすことを意味している。本プロジェクト開始前にもゼミナール内で参加自由の機械学習勉強会を開催していたが、その際に必要性をあまり感じていない学生は最後まで継続することが難しいことが判明していたためである。

また、多くの前提となる知識や技能が必要となるため、発展的科目として本学部の 3 年次に配置することとした。4 年次履修科目としなかった理由は、本授業での経験を卒業研究などで活用することを想定したためである。さらに、「プログラミング基礎」「オブジェクト指向プログラミング」「統計入門」「社会統計」「確率統計」「数理情報」「社会数理」「人工知能基礎」の科目を履修済みであることを条件として設定することを想定した。これにより、基礎事項の確認に多くの時間を割く必要がなくなると考えた。

2.2.2 到達目標と内容および難易度設定

到達目標・取り扱う内容・難易度は、従来の試行[1]を元に設定した。具体的な数値として示すことができないため、説明が困難ではあるが、学部の情報系の成績上位 10-20%の者が授業時間内だけでは試行錯誤を必要とし、

授業後の課題によって習得する程度を想定した。

到達目標は、「履修後に機械学習の発展的内容に自力で取り組める力を習得」することとした。つまり、機械学習に関しての理論や調べ方の基礎力が身につくこと、実装を試行することのできる力がつくことを目指す。

また、内容の設定としては、機械学習の「理論」と「実装」の両面のバランスに注意を払うこととした。理論がわからなくともライブラリが利用できれば、ある程度のアプリケーションは作成できる。しかし、これではライブラリに組み込まれていない新手法に対応することができず、精度をあげるための手法選定やパラメータチューニングもシステム任せになってしまう。これでは単に表計算ソフトが利用できると言っているのと大差ないことになってしまう。一方で、理論に偏ると実社会への適用の側面が疎かになりがちである。機械学習を扱う際には、問題設定などの俯瞰した視点も必要である一方で、データ取得や前処理などを効果的におこなえなければならない。

取り扱う範囲は、基礎的な内容から応用的な内容まで幅広くカバーすることとした。様々な手法の紹介、様々な分野での利用例を目にしてキッカケを作ることを意識している。自学実装の実力がついていれば、あとは各自が深く勉強してくれるようになることも期待している。

広い範囲を取り扱い、実装（実習）もあることから、当然難易度は高くできない。基礎的な内容を端的にしめし、不明な点については質問を受け付ける形を想定した。

2.2.3 授業内容

上記設定に基づき 15 回分の授業内容を設定した。以下に、その概略を示す。

1. ガイダンスおよびシステム環境・構築について
2. 回帰
3. 分類
4. 決定木・ランダムフォレスト
5. 検証・性能評価
6. 前処理
7. データ取得
8. 演習試験
9. 深層学習
10. 深層学習・画像処理
11. 画像処理
12. 時系列データ
13. 自然言語処理（基礎）
14. 自然言語処理（深層学習）
15. 教師なし学習・まとめ

各授業回のタイトルだけをみると、至極一般的な内容にうつるのではないだろうか。

第 5 回検証・性能評価、第 6 回前処理および第 7 回データ取得は、時間をかけて解説することとした。書籍などでは、これほどの分量を割いているものは少ないが、自分の手で機械学習のシステムを実装する際には非常に重要な部分となるため、本講義では力を入れている。

また、単純にツールの利用方法／使い方（Python やそのライブラリの使い方）を学ぶ時間は最小限とし（別の授業にてカバーする必要がある）、機械学習のプログラムを「実際に動かす」ことを想定している。

なお、学生の自宅またはクラウド環境での演習が厳しい深層学習の回をのぞき、すべて授業時間外におこなう課題を設定することとした。これは実際に手を動かす実習時間を確保するためである。

また、過去の試行[1]の改善点を受けて、課題は一般的なものではなく、学生にとって身近に感じられるオリジナルなものをできるだけ用意した。

過去の試行[1]にて、分量や範囲、理論と実習のバランスなどが適当な書籍がないことが判明していたため、今回はすべての教材を作成した。

2.2.4 授業環境・学習環境

現在のところ、授業の環境としては Google Collaboratory 上で Python およびそのライブラリを活用することを想定している。

本プロジェクト開始時および従来の試行[1]の際には、教室および学生の環境として以下のものを想定していた。どの計算機環境でも同様の作業ができるよう仮想化のために Docker を利用する。プログラミングおよびライブラリとしては、Python およびそのライブラリを活用する。プログラミング環境および実行には、Jupyter Notebook を利用する。また、この Notebook を配布することによって、学生は教員とまったく同じ環境下でプログラムの実行がおこなえ、その説明書きを参照することもできる。

しかしながら、様々な授業において Docker を利用する際、学生の自宅環境での設定が困難なケースが散見される。仮想環境をもちいない場合には、バージョン不一致や原因不明のエラー表示など、さらなる問題もある。

そのため、現時点では無料で利用することができるJupyter Notebook 環境として Google Collaboratory を利用することを想定した。その利用の継続性が不明である点に問題があるが、実際に授業が開講される 2 年後には、その他の方法も含めて状況が変わっている可能性もあり、その際には再検討することとした。

2.2.5 授業構成

授業は大きく 3 パートに分けておこなう。1 番目のパートが約 10 分、2 番と 3 番目のパートがそれぞれ約 40 分である。

2 番目のパートは、当日授業回の講義部分である。講義は進行の部分はプレゼンソフトにておこなう。また、機械学習のプログラムの動作をさせながら、その操作方法や手法を説明する。その際には、すでにプログラムが書かれている Google Collaboratory の Notebook を実行しながら動作の様子を提示する。また、実際にパラメータなどを変更しながら実行し、理解を深めてもらう形式とした。

3 番目のパートは、授業内演習である。その日に学習した内容を用いて、実際の演習問題のプログラミングをするとともにパラメータ調整などもおこなう。この時点で、その授業回の基本的な部分や動作部分に関しては、ここですべて把握することを想定している。

1 番目のパートは、授業終了時に出題された授業外の課題の解説をおこなう。基本的には、授業内課題と同レベルのものであり、不明な点はないはずだが、多くのものが間違えた点を中心に解説する。

2.3 試行検証

検証は、学部 3 年生の有志に対しておこない、2018 年度は 8 名、2019 年度は 7 名が参加した。

授業外課題の正解具合、講義への集中度合いなど過去の試行と比較して良好であった。また、無記名自由記述式のアンケートでは、特に演習部分についての不満はなかった。

これらの実験に参加した学生の 8 割が卒業研究にて機械学習手法を利用したことから、本取り組み内容には効果があり、目的が達成されたといえる。

一方で一部の学生からは、理論面での説明を増やして欲しいという要望が挙げられた。これに関しては、過去の試行の改善として現在の分量としているため、直接の対応が難しい。理論面での解説が丁寧な書籍は存在しているため、それらを参考図書として上げる形で対応したい。

3. プログラミング素養をもたない学生向けの教育プログラム

3.1 基本的な考え方

機械学習を扱う授業科目について、WEB 上に公開されている大学について調査を行った。調査の詳細は省略するが、理工系学部で、情報分野の学科がある場合にはほぼ必須の内容として設置されているのに対して、人文社会系学部で設置されている事例は非常に少ない。また、前者の場合にはプログラミングに関する授業が別途設置されており、それを前提として運営されている。すなわち、授業において理論中心の講義がなされたうえで、受講生がプログラミングに基づく演習を行っていくという形態が中心である。これに対して、後者については、授業計画の多くはプログラミング言語習得による演習という形態が中心となる。その際、プログラミング言語の中心は R 言語である。R 言語は、非理工系学部における統計教育等で広く利用されているという背景もあり、このような学びに抵抗感のない学生を対象としているものと想像できる。ただし、授業内でプログラミングそのものの学習を行うことを前提としていることが多いため、理論よりも事例・体験型の授業内容となっている。

両者の取り組みに共通するのは、演習要素の重要性である。理工系分野の授業においては、数学を前提とする理論的な理解が必須であることから、演習の目的は、学んだ理論そのものの理解という要素が大きい。一方、非理工系分野の授業では、数学を前提とした理論的な理解には限界があるため、演習の目的は、具体的な課題解決に対してどのように機能するのかということの理解であり、事例学習に重点があるといえる。このことから、非理工系学生向けの教育プログラムにおける演習要素は、理工系学生向けのそれよりも重要であり必須であることがわかる。

本取り組みでは、非理工系分野の学生に対して、プログラミング言語の習得を前提とせずに機械学習／データマイニングを学ばせる有効な取り組み、すなわち、プログラミング言語の習得を前提としない演習が行える教育プログラムの実現を目指す。そこで、機械学習のためのフリーの GUI ツールである RapidMiner[2] [3]を使用することで、非理工系の専門分野を問わず、「データハン

ドリング→前処理→モデリング→評価・解釈」のデータ分析プロセスを体系的に学ぶことができるような演習中心の授業を設計する。そして、多くの学生が「機械学習で何ができるか」ということを理解し、データの観察力・分析力を身に着けることを目的とする。

3.2 授業計画と特徴

授業は、入門段階として 8 回の授業で、教師あり学習の全容と個別演習テーマとして決定木・回帰を、教師なし学習の全容と個別演習テーマとして K-means クラスタリングとアソシエーションルールを取り上げる。これにより、機械学習アルゴリズムにより実現される問題を理解したうえで、応用段階として 6 回の授業で、他の学習アルゴリズムを取り上げながら、事例学習を継続して実施していくこととする。それぞれの授業での理論の解説の後に、演習を行う。

各演習は①チュートリアル②応用演習の 2 部構成とする。チュートリアルでは各アルゴリズムの解説を中心に演習を進める。応用演習では、チュートリアルとは別のデータセットを用いた演習を行うことで、チュートリアルで学習したアルゴリズムを用いて、データ分析プロセスを体系的に学習する。演習の中で、データ観察時のポイントや前処理の手法、パラメータ、評価指標についての解説を行う。また、授業で扱う項目ごとに到達目標を設定した。この到達目標は暫定的なもので、図 1 に、本プログラムを施行した際に設計段階で協力した学生から提案されたものを示す。

図 1 到達目標

次に、本プログラムの特徴を示す。

3.2.1 GUI ツールの利用

設計した授業内の演習はすべて機械学習の GUI ツールである RapidMiner を使用する。前述のように機械学習の演習を行うためにはツール・言語の習得が必須である。RapidMiner はノンプログラミングで視覚的にデータの流れを理解できるため、理解が容易で効率よく演習が行える。ツールの習得は①指導教員がすべての演習のデモを行い、それに従って演習を進めること②授業資料にサンプルプロセスを載せることを原則とする。図 2 に、決定木学習における交差検定を行う演習画面の例を示す。

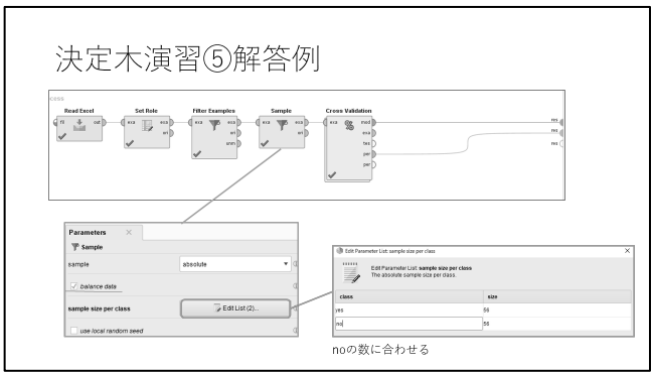

図 2 決定木応用演習の画面イメージ

3.2.2 学生にとって身近な演習内容

演習に用いる機械学習用のデータセットは多く公開されている。機械学習を学習する際によく使用される土地データや医療データ、画像データなどのデータセットは学生にとって必ずしも取り組みやすいデータではない。しばしば用いられる UCI リポジトリ[4]のように理系向きのものやビジネスなどのビジネス向けのものが中心であり、一般の学生にとって馴染みのないものが多い。したがって、学生にとって理解しやすい身近なデータセットが必要となる。演習では本学で使用している学生意識調査データの仕様に基づいたデータセットを作成した。学生意識調査とは、毎年度全学年を対象として実施されており、入学時と卒業時の調査項目を使用し、演習問題を作成した。演習問題は、物語形式に設計することにより、学生にとって取り組みやすい設定となるように配慮した。

3.2.3 受講可能な数学知識レベル

数学の知識を要する概念は、ジニ係数,混合行列,回帰式,RMSE,支持度,確信度,リフト値などが存在する。これらについて、高校 1 年までの教育課程で学習する数学知識で容易に理解することができる解説を施した。

3.2.4 実際の問題解決に沿った授業内容

本プログラムでは、基本に加えて実際の問題解決現場における重要な概念の学習、例えば、過学習や不均衡データ、ダミーエンコーディング、交差検証などの概念理解についても積極的に取り入れた。これらは、一般的に機械学習の参考書では応用領域と位置付けられることが多いが、実践を意識した幅広いデータ分析力を身につけるためには必須の概念として、演習の中でこれらをわかりやすく解説することとした。

3.2.5 具体的事例から学習

必要とされる場面について詳細に説明しなければならない用語や概念の説明は、多くの学生にとって理解が難しい。そのような場合には、特に具体例を提示した後に一般論の説明をするという流れに重点を置いた。たとえば、フィルタリングの解説スライドを示す。ここでは、“すべての学部から特定の学部を抽出するために”行う処理をフィルタリングと定義し、演習内容に沿って用語や概念の解説をする。

4. まとめ

社会情報学部の新カリキュラムとして、非理工系で一定のプログラミング素養をもつ学生向けの教育プログラムである「機械学習基礎」に関し、設定および設計をおこなった。また、その授業コンテンツおよび課題を作成した。これらの効果を確認するために試行検証をおこなった。授業にて実際に活用可能であり、目的が達成されたことが確認された。なお、一部のコンテンツについては新規のより良い手法に置き換えた方が良いことが判明している。またエッジコンピューティングに関する授業コンテンツの試行が充分ではない。これらについては、本講義が開講されるまでに、改善する予定である。

一方、実装するまでのプログラミング素養を持たない学生を想定した学生を対象とした教育プログラムの開発については、いかにして理論修得をと実践への展開を図るかという課題が存在した。また、そのような前提においても、演習中心の授業設計が必須であることも確認された。本取組みでは、このような問題の解決を図るために、1) GUI ツール RapidMiner の活用、2)学生にとって身近な演習内容、3) 高校レベルの数学の知識で受講可能 4)実際の問題解決に沿った授業内容 5)具体的事例から学習の 5 点に留意して授業資料、演習内容を作成した。さらに、実問題への対応可能なデータ観察力・分析力が身につくようなデータ分析プロセスの体系的な学習のための授業設計をめざした。

今後は、授業資料、演習問題についての評価実験を進め、改善点を明らかにすることや、他大学の状況や大学以外での研修等の内容を調査することで、教材全体の改良を図っていきたい。その結果、非理工系学生を対象とした学部横断的な全学的なカリキュラムとなることを期待したい。

最後に、本取り組みのコンテンツ作成および試行実験に、本学社会情報学研究科博士前期課程の酒匂暁史氏、中本昌吾氏、丸山拓己氏をはじめ、本学部学生には試行実験への対象として参加協力していただいた。これらの学生に感謝する。

参考文献

[1] 宮治裕，吉田葵，機械学習に関する演習教材開発および研究，pp. 114-116. 青山社会情報研究，青山学院大学社会情報研究センター，2018.
[2] Data Science Platform | RapidMiner https://rapidminer.com/（2020 年 8 月）
[3] Operator Manual - RapidMiner Documentation
[4] https://docs.rapidminer.com/latest/studio/operators/ （2020 年 8 月）
[5] 「UCI Machine Learning Repository」https://archive.ics.uci.edu/ml/index.php （2020 年 8 月）