JSDSS 日本ソーシャルデータサイエンス論文誌
第 3 巻　第 1 号（2019 年 3 月）

特集 : データサイエンス人材育成

データサイエンティストが実務を通して経験すべきこと

岩永 二郎[†1]

概要 : データサイエンティストに必要なスキルとして理論，エンジニアリング，サイエンスの素養があり，どれもアカデミックで学ぶことができる．しかし，これらのスキルを実務で実践する際には些か隔たりがある．本稿ではアカデミックで学んだデータサイエンスのスキルを実務で実践する際のギャップについて整理した．また，実務では避けて通れないコミュニケーションについても議論する．本稿を通してアカデミックで育った駆け出しのデータサイエンティストが実務でインパクトを出せるプレイヤーに成長することを，また経験豊富なデータサイエンティストには振り返りの機会を与えられることを期待したい．

キーワード : データサイエンス，データサイエンティスト，育成，理論，エンジニアリング，サイエンス，コミュニケーション

1. はじめに

本稿ではデータサイエンスの実践者をデータサイエンティストと呼ぶ．データサイエンティストに必要なスキルは 3 つある．数理科学やコンピュータサイエンスなどの理論，実装に落とし込むためのエンジニアリング，そしてこれらを高いレベルで実現するためのサイエンスの素養である．ただし，ここでいうサイエンスの素養とは科学的手法を適切に使えることや科学的プロセスを踏むことができる能力を指す．これらのスキルをアカデミックや独学で学んだからといって実務ですぐにインパクトを出すことは難しい．実務でサイエンスを実践するということは世界中で自分だけの問題に取り組むことに他ならない．例えば，実務では事業ドメイン，サービス，ユーザー行動の組み合わせだけ異なる現象があり，必ずしも汎用的な理論を適用できるわけではない．データの形式が同じであれば，全て同じような現象が起きていると期待したくなるが，結局のところデータを生成した背景は千差万別であり，その都度データの裏にある現象を解析する必要がある．このように実務にデータサイエンスを適用するためにはいくつか気をつけなければならないことがある．

本稿はデータサイエンティストに必要な理論，エンジニアリング，サイエンスの素養についてアカデミックと実務のギャップを意識できるように整理した．アカデミックと実務におけるデータサイエンスには共通点があることについては，[1] で議論しているのでこちらを参照されたい．本稿では共通点ではなく，相違点に注目する．本稿の構成を説明する．本節では取り扱うテーマについて述べた．第 2 節ではアカデミックで学んだ理論を実務に応用する際の壁について，第 3 節ではデータサイエンス特有の実務におけるエンジニアリングについて，第 4 節では実務とサイエンスの溝について述べる．第 5 節では実務において重要視されるコミュニケーションについて触れたい．最後に第 6 節でまとめをする．

本稿はアカデミックで育ったデータサイエンティストが実務とのギャップを乗り越えて現場に適応するために気をつけるべきことを議論する．経験豊富なデータサイエンティストからすれば，すでに経験してきたことを言語化したにすぎないが，意識下におくことで自己成長や育成の糧としていただけたら幸いである．

2. 理論と実務の壁

本節では理論を実務に適用する際の壁について焦点を当てて解説する．まず，理論は絶対的なものではないという認識が必要である．研究者は世界に起きている現象を鋭い観察力と豊かな表現力を武器に理論を作りだす．しかし，現象が理論の通りに動くべきだと考えてはならない．[2] で挙げられている例を紹介しよう．経済学者は，様々な要因が複雑に絡み合っている経済現象に大胆な単純化を施し，その本質部分を暴き出して理論をつくる．理論をつくった人たちは，現在の経済現象を十分な精度で説明できるとは限らないことを知っており，理論の現実への応用については十分な謙虚さを備えている．しかし，教科書を通じてこの理論を学んだ人々の中には過信する人が現れ，現実はこの理論通りに動いているに違いないと信じる，という話である．このように現象から生まれた理論を，実務に適用する際には注意が必要である．それは多くの場合，現象特有の条件が，理論が成立する条件とは異なるからである．そのため，実務では“いま取り組んでいる問題は世界に一つだけの問題である”という姿勢は大事である．初めて出会う問題に対してどのようにアプローチするかアカデミックで経験していればそれに越したことはないが，データサイエンティストのキャリアとしては実際の実務で経験するほうが現実的である．このような経験は大きな成長機会の一つであり，実務を通してこそ身につけることができる．

データサイエンスの分野において最新の研究成果があったとしても，実務で自分が取り組んでいる問題では再現さ

†1 Retty 株式会社 （連絡先：iwanaga@retty.me）

れないことは頻繁に起こる．論文で取り扱っているデータの形式と自分が取り組んでいる問題のデータの形式が同じ場合，自分の問題でも再現できるのではないか，という期待をする気持ちはよく分かる．しかし，対象とするデータを生成した背景は異なるため，同じことが再現されるかどうかは分からないのである．例えばレコメンドアルゴリズムに関する論文を読んだ際，自分が直面している問題でも性能が良いことを期待してしまう．しかし，論文で利用したデータは対象とする事業ドメイン，サービス，ユーザーの特性に依存しており，その条件下で性能が高いことしか担保していない．もちろん汎用的に精度が高いレコメンドアルゴリズムもあるが，自分が直面している問題の特性を利用すればシンプルなアルゴリズムでも汎用アルゴリズムの精度を超えることは頻繁に起こり得る．また，実務ではアルゴリズムだけでなく，情報設計やデザインなど UI/UX を含めてレコメンドを実装するため，アルゴリズム以外の影響で論文に書かれているような性能が再現しないこともある．

理論を実務に適用する際，成立条件を確認することは重要である．例えば統計的検定を利用する際には，統制された状況で得られたデータに対して検定を行う場合と，WEB のユーザー行動ログのように統制ができない状況で得られたデータに対して検定を行う場合ではまるで異なる．後者の状況はノイズが多いため検定を適用するのは好ましくないが，実務ではノイズを無視しなければ話が進まないという理由で棚上げされることがある．論文とは違い，ある程度のミスリードが許容される実務ではこのようなことが頻繁に起きていて，それが妥当な判断である場合もある．非常に気持ち悪いが，このようなジレンマを経験したデータサイエンティストは多いだろう．

本章の最後にデータサイエンスにおける理論の学び方について触れたい．まず大事なことは理論だけでなく，成立する条件を理解することである．論文に実証実験があればそのデータの形式だけでなく，データの背景にある現象，およびその特性を理解することが重要である．その上でどのような現象から生成されたデータであれば適用可能か，もしくはどのようなサービスでインパクトが出せるかを考えておくと，いざ理論を実務に適用する際に扱いやすい．

3. 実務におけるエンジニアリング

本節ではデータサイエンス特有の実務におけるエンジニアリングの話題に焦点を当てて解説する．一般のエンジニアリングと異なる点に注目して 4 つの話題を取り上げる．探索的データ解析，自動化と高速化，ロギング，保守運用についてである．

3.1 探索的データ解析

探索的データ解析 (exploratory data analysis) [3] は統計学者 J. W. Tukey によって提唱された．これは解析初期のフェーズにおいて多面的にデータの特徴を捉えるために対話的に分析を進める手法である．データの背景にある現象を明らかにすることで，どのようなモデルを適用すればよいか検討できるようになる．データサイエンスでは最も重要な考え方である．実務においてデータサイエンスの技術を用いたシステム開発をする場合，ウォーターフォール型の開発は向いていない．なぜならば自分たちが直面している現象に対して，どのようなモデルを当てはめればよいか，どのようなアルゴリズムを選択すれば性能がよいか未知なためである．データの形式を揃えて，適当なアルゴリズムを選択すれば機械学習モジュールが出来上がると考えているシステム開発者は意外に多い．しかし，データサイエンティストが開発しているものはシステムにおける関数ではなく，モデルを開発しているのである．入力と出力が定義されたブラックボックスとして定義される関数とは異なり，モデルには“構造”が含まれる．単なる関数ではなくモデルを開発するという立場にたてば，探索的データ解析の重要性も理解できるであろう．ここではデータ解析の初期フェーズで行う探索的データ解析を例に話をしたが，実際にはデータ解析初期だけではなく，特徴量エンジニアリングやアルゴリズム選定，パラメータ調整など様々なシーンでチューニングの必要性があり，その都度，対話的なデータ解析が必要になる．また，ここでは触れなかったが特徴量エンジニアリングについては [4]（日本語訳 [5]）によくまとめられているのでこちらも参照いただきたい．

3.2 自動化と高速化

探索的データ解析は対話的にデータを分析していく手法であるが，ある程度モデルの予想がつけば，網羅的な実験を計画することができる．さらに計画された実験の自動化と高速化をすることで分析効率を上げることが可能である．まず，網羅的に実験する理由は抜けもれなく性能のよいモデルを見つけることである．実験を自動化し，高速化するのは決められた期間内で最も良い手法を見つけるためである．ロジック，特徴量，ハイパーパラメータなど可能な限り網羅的に実験することが望ましい．適切な実験項目を列挙できれば，あとは自動化と高速化である．実験の自動化は初歩的なコーディングスキルがあればできるが，コードをべた書きしかしたことがないデータサイエンティストには少々難しいかもしれない．最低限のクラス設計や関数化はできたほうがよい．また，高速化も重要である．高価なハードを利用することだけで高速化の問題を解決しようとする人も多いが，データ構造やアルゴリズムなどのコンピュータサイエンス，データベースやプログラミング言語の背景にある実装の理解があると実務の節々で役に立つ．データサイエンティストが最もパフォーマンスを出せる分析環境を構築できるかどうかは，その後の分析効率に大きな

影響を与える．決められた期間内で最も多くの試行錯誤ができる環境をつくる経験は非常に重要である．データ解析系のコンペに参加したことがある人はこの重要性が身に染みてわかるだろう．納期が決められたプロジェクトであればなおさら重要である．

3.3 ロギング

データサイエンティストの中で分析結果の検算を習慣的に行っている人はどのくらいいるであろうか．分析結果の正しさを保証するためには検算をしなければならないが，検算を疎かにする人は多い．特に独学でスキルを身に着けた人や多忙な人は検算を怠る習性があるため肝に銘じておおきたい．

ここで検算に触れたのはデータサイエンスの技術を利用したシステムを構築する場合，検算を設計するスキルがテストやエラー処理に関するエンジニアリングの質に影響するからである．例えば，複数のデータセット（Python や R 言語でいうデータフレームなど）を入力とする関数を実装しなければならないケースがあるとしよう．本来は，各データの型だけでなくデータの定義域やデータ同士の整合性もチェックしなければならない．さらには，欠損値や想定外の入力に対してどのように動作させるか事細かに関数設計することも必要である．入力とするデータセットが自身で管理している CSV ファイルであればエラーが起きるケースも想定できるが，自身が管理していない DB を利用する場合には注意が必要である．ましてや自社で管理している DB ですら予告なしに仕様変更されるのに，パブリックに公開されている API だとなおさら何が起きるかわからない．ユーザーが投稿した情報を DB に入れる際に適切なバリデーションが実装されていない場合も想定外のことが起こりやすい．これらを考慮すると正しい入力を期待することは不可能であるため，どのようなデータが入力されても大雑把にエラー処理をしてそのまま処理を続けるような実装にたどり着く．このとき，エラー処理の方法を間違えるとリスクが生じる．例えば，レコメンドエンジンでは意図と異なる商品を配信してしまうこともあるし，契約が絡んでいる案件では大きな問題になることは容易に想像できる．また，想定外のデータが入力されても動作するように実装した場合，誤りであるにもかかわらず 1 年，2 年そのまま運用されることもざらにあるだろう．これはデータサイエンティストとして非常に格好が悪い．

上記のリスクを避けるためにはロギングの仕組みを実装することをお薦めする．高度なアルゴリズムを利用する場合，プログラムが正しく動作しているにもかかわらず，入力データの異常のために想像と異なる動作をすることは起き得る．これは通常のシステムよりも複雑で高度な処理をしているため，異常ケースを潰しきれない，またはその工数が確保できないという理由が大きい．そのためプログラムが正しく動作していることをモニタリングするだけでなく，その処理が意図通りの処理になっているかを検証できる仕組みがあるとよい．例えば，データの前処理のフェーズで常にデータ数をカウントしておくだけでもよいし，気が利いた統計値を出力しておくのでもよい．

本節ではロギングという言葉を利用したが，検証という意味合いもあるのでレポーティングに近いのかもしれない．出力されたログ/レポートはデータサイエンティストが目視で確認してもよいし，異常検知を実装して毎日目に触れる媒体に自動通知させる仕組みもスマートである．定期的にモニタリングすることで健全かつ頑健な運用をすることができる．

3.4 テストコード

データサイエンティストでテストコードを書いている人は少ない印象を受ける．システムを開発するにあたりモデルが未知のフェーズでは試行錯誤が多いので，いちいちテストをするのは非効率である．しかし，分析や実験の試行錯誤に時間を費やして，テストコードを書く時間がとれなかった，というのは良くない．少なくともリスクが発生する箇所のアサーションや単体テスト，可能であればリグレッションテストまで含めて実装しておくとよい．著者自身の経験であるが，先日，2 地点の緯度経度から距離を計算する関数を実装したのだが，関数呼び出しの際に緯度経度の引数の順番を間違えており，後々単体テストを書いて誤りを見つけたことがある．緯度経度の順序を逆にしてもそれらしい距離が出力されてしまうため気付かなかったのである．おそらく単体テストを書かなければ一生見つからなかった不具合であろう．

3.5 保守運用

保守運用で重要なことはドキュメントを残すことである．納品を経験したことがあるデータサイエンティストであれば README やプログラムの利用説明書を書いたことがあるだろう．しっかりしたドキュメントを書かなければ納品後に問い合わせが殺到し，多くの時間を無駄にする．メール対応に追われて難しい問題を考える時間がとれなくなるのはデータサイエンティストにとって最も不幸なことである．アカデミックで論文を書く訓練をしっかり受けたデータサイエンティストはドキュメントを書く素養があるだろう．しかし，そのようなデータサイエンティストでも自分が開発したプログラムについてのドキュメントを書きにくいと感じたことはないだろうか．そのような場合，そもそもプログラムの設計が悪いのではないかと疑ってみるとよい．運用する際に直感よりも手順が多いと感じるケースや保守する際に変更を加えるファイルが多いと感じるケースである．情報系出身ではないデータサイエンティストには苦手な人も多いかもしれないが，よく考慮して設計されたプログラムのドキュメントはシンプルになる傾向がある．

もう一つ保守運用に関して大事な話題がある．機械学習エンジンの開発をした場合は，データの管理，アノテーションの実施，モデルの管理などもデータサイエンティストが意識しなければならない．ここではデータの管理について触れる．まず，データを管理するときに気をつけなければならないのはデータの品質を保証することである．例えば，EC サイトの購買履歴を用いて予測モデルを構築するケースを考えてみる．EC サイトでは運営側の施策やキャンペーンが絶えず実施されるため，その影響を受けたユーザーの購買履歴から予測モデルを学習すると意図とは異なるモデルが構築されてしまう．データサイエンティストはこのようにデータにノイズが入ることも考慮してデータを管理しなければならない．また，モデルを学習するにあたってバージョニングされたデータが必要な場合もある．ここでは詳細に触れないが時系列で更新されていくデータを利用して学習をする場合，時点ごとのデータのスナップショットを利用しなければ意図したモデルが構築できないことがある．適切な分析をするためにはデータの管理から意識する必要がある．

4. 実務におけるサイエンス

本節ではサイエンスと実務の溝に焦点を当てて解説する．データサイエンティストがまず実務で意識しなければならないのは，インパクトを出す責任があるということである．データサイエンスが生み出すアウトプットの価値やどのような変化をもたらすかを考える必要がある．研究の意義を重要視するアカデミックと異なり，実務ではどのような変化をもたらし，どのようなインパクトを与えたのかが重要となる．

ここでは実務でサイエンスを実践するにあたり溝を感じやすい話題として課題設定，解決方法の設計，検証について触れたい．アカデミックでサイエンスするのと異なり，実務でサイエンスする際にはギャップを感じることが多い．それはアカデミックでは学べない難しさである．これは現場の経験を通して学ぶべきことであるが，多くのデータサイエンティストがアカデミックと実務との溝を克服できずに挫折していくのを目の当たりにすると，予め心構えをしておく必要があるのだと思う．

4.1 課題設定

適切な課題を設定することは，実務でサイエンスしていけるかどうかが決まる重要なタスクである．プロジェクトを成功に導くために大事なことは，ゴールの状態を明確にして適切な課題を設定することである．アカデミックで研究テーマなどの課題設定をしたことがあるデータサイエンティストであっても，実務で課題設定を行うのに苦労することは多い．実務では研究とは異なり合理的で理想的な課題設定をできない都合があるためである．例えば，課題設定できない理由として，他の部署とのカニバリを起こすだとか，既存メンバーでは難易度が高すぎるという理由もあるし，その課題を設定すると不快に思う人がいるという感情的な話もある．実務で課題を設定するのは想像以上に難しいのである．外圧により明確な課題を設定できなかった時点でデータサイエンティストがそのプロジェクトでインパクトを出すことは難しいと覚悟したほうが良い．

4.2 解決方法の設計

解決方法を設計し，実装に落としこむタスクはデータサイエンティストの花形の仕事である．研究と異なり実務で大事なことは，必ずしも自分の得意分野の方法で解決する必要はなく，むしろすべての分野の手法を自由に選択して，自然なモデリングの元に適切な解法を組むことである．実際の実務では取り扱っている問題を数理モデルに表現するが，まずは問題が解けるように，解法を実装できるようにモデリングしなければならない．もちろん保守運用も考慮する必要がある．また，近似の精度も重要である．近似の精度が期待する精度を担保できなければ実用に耐えられない高価な玩具となってしまう．環境にも注意が必要である．理論的に実装可能だとしても環境要因のために実装できないケースは多い．例えば実装者の技術力，ソフトウェアの利用可否，インフラなどの環境要因にも注意が必要である．予算内でかつ実時間内に計算が終わるように解法を設計できるかどうかは経験の差が如実にあらわれるところであろう．実務でもう一つ大事なことは，どうしても解決方法を実装に落とし込めない場合である．そのようなときは，業務や仕組み自体を変えてしまうことも検討する必要がある．データサイエンティストとして難しい問題を解く能力は重要であるが，問題を簡単にする能力もまた同じくらい重宝される．仕組みや業務を変えることで技術的に自然で無理をしない実装を実現できるのならば，それがベストプラクティスである．

最後にいくつかアドバンストな話題を提供したい．1 つ目は拡張性の話である．数理科学やコンピュータサイエンスの色が強い案件ではプロジェクト初期では判断がつかないことや現場特有の暗黙知が非常に多い特徴がある．常に要件・仕様変更が起きるため拡張性がない解法を選択してしまうと変更が起きた場合に対応できないこともある．データサイエンスの技術が取り入れられているシステムの中で，世に使えないシステムがあふれるのはこのことも一因である．また，要件・仕様変更に対応すると解法の設計からやり直す必要があり，システムも大きく変更することになる．そうならないためには PoC (Proof of Concept) を実施し，小さめのプロジェクトからはじめて現実的に問題解決が可能か判断する進め方と相性がよい．これらのリスクも考慮した解決方法をはじめから設計できるデータサイエンティストは貴重である．2 つ目は実務ではシンプルな解決

方法で十分であるという話である．データサイエンティストになる人の多くはデータサイエンスに対して過度な期待と夢を持っている人が多いが，実務の多くシーンではシンプルな解法が好まれるだけでなく，実際に費用対効果もよい．意思決定に必要な分析技術は集計だけでよい，という話もよく聞くだろう．実際にデータサイエンティストのキャリアの中で多くの人が経験する洗礼である．私はデータサイエンティストのキャリアはここからがスタートだと思っている．つまり，実務の具体的な問題で単なる集計を超える分析を実施できるかどうか，そのような問題を見つけられるかどうかでデータサイエンティストの深みが分かる．データサイエンティストのキャリアを進む人には，シンプルな解決方法も高度な解決方法も常に選択できるように用意していて，状況に応じて判断できるレベルまでやりきってほしい．

4.3 検証

実務におけるサイエンスで最も重要なタスクは検証である．実務では様々な場面で検証を行うが，ここでは施策を実施した際の検証をイメージしていただきたい．検証は重要であるという認識はされつつも多くの現場で軽視されているのも事実である．時間がないため検証をスキップすることもあるし，モノづくりは楽しいがその後は興味が無いという人も多い．また，現場によっては正確な検証は不都合なことさえある．既存の業務，評価に満足している場合，検証を実施する明確なメリットがなければ検証が現場に定着することはない．検証を行うメリットは意思決定や評価の際に得られるが，ここでは意思決定におけるメリットについて解説する．意思決定における検証のメリットは知見が貯蓄されて判断が効率化されることであり，検証をしないことのデメリットはミスリードが増えることである．しかし，前者については知見を貯めるよりもアクションを増やしたほうが短期的に効率がよいという意見もあるし，後者についてはミスリードの1つや2つを恐れずにアクション数を増やしてインパクトを出せばよいというのも1つの考え方である．検証のメリットを享受したことがない，もしくはそのデメリットと関係のない立場の人にとって，検証が無用のタスクに見えるのは自然なことである．検証を怠っていると意思決定の効率が徐々に悪くなり，ミスリードが増えていく．ミスリードが起きていることに気づくことができれば対応もできるが，検証を怠っているとミスリードすらも検知できないことが最も大きな問題である．このことからも中長期的な観点では検証が重要であることは言うまでもない．データサイエンティストは如何なる圧力があろうとも検証の実施を推奨するべきである．ただし，どの程度の検証の正確さを要求するかはプロジェクトごと，意思決定の大きさによって異なるため，そのバランス感覚には気をつけたい．

5. コミュニケーション

実務においてコミュニケーションスキルは重要である．チームで業務を遂行する場合にはデータサイエンティストのコミュニケーションスキルがプロジェクトのインパクトを決めることすらある．本節ではデータサイエンティストがコミュニケーションをとる相手として，データサイエンティストを含むエンジニアと非エンジニアに分けて議論する．

エンジニアに対しては，要件や仕様を正しく言語化する能力があれば十分に業務を遂行することができるが，ここでは一歩進んだコミュニケーションの話をしたい．データサイエンティストはステイトメントを正しく述べる能力が重要である．ここでステイトメントとは，明確に定義された言葉を利用して，簡潔に，誤解のない形で表現された言明である．その言明の中で定義が閉じていればその言明に対して真偽を議論することができる．正しくステイトメントが述べられると自分だけでは解けない問題があったときに，他のデータサイエンティストやエンジニアに相談し，チームで問題解決に取り組むことができる．自身の専門分野でも細かい理論まで把握している人は少ないだろうし，専門外の知識を含むような問題を自分だけで問題解決することは難しい．正しくステイトメントを述べることができることでより難しい問題にチャレンジできるようになる．

一方，非エンジニアに対しては，相手の知識に合わせて会話する能力も要求される．そのため，データサイエンティストが相手側の業務を知っておくことが望ましい．また，非エンジニアとのコミュニケーションは相手の言語化能力にも大きく依存する．そのため必要な情報を引き出すのに多大なコミュニケーションコストが発生するケースがある．しかし，非エンジニアとのコミュニケーションがうまくいかないからといって悲観することはない．プロジェクトを通して少しずつコミュニケーションの質を上げていけばよい．両者に歩み寄る意思があれば少しずつ質の高いコミュニケーションが積み上がり，チームワークが醸成するのである．

非エンジニアの中にはデータサイエンティストとクリエイティブなディスカッションをできる人がいる．著者も何人か出会ったことがあるが，そのような人は決まって言語化能力が高く，経験が豊富である．また，データサイエンスで何ができて，何ができないかを直感的に理解しているように思える．そのようなパートナーと出会えたならばデータサイエンスの技術で大きなインパクトを狙うチャンスである．しかし，いつもそのようなパートナーに出会えるとは限らない．そのため，データサイエンティスト自身でプランニングやマネジメント，ディレクションをしたほうが，コミュニケーションコストは少なく済むというのも一つの選択である．

6. おわりに

本稿ではデータサイエンティストが実務において経験すべきことについて議論した．特に，データサイエンティストに必要なスキルである理論，エンジニアリング，サイエンスの素養に注目してアカデミックと実務のギャップを意識できるように解説することを心がけた．また，実務では避けられない話題としてコミュニケーションについて解説した．

データサイエンティストという職業は日本国内でまだ定着していない．これはマーケットの規模が小さいためデータサイエンスの技術でインパクトが出せる現場が少ないということもあるが，データサイエンティスト自身が自律していないことも原因である．理論やエンジニアリングのスキルを学ぶだけでなく，これらのスキルを目的ベースで活用できるデータサイエンティストが必要とされている．もちろん自分自身でプランニングしてもよいし，様々な提案をしていけるのであればなお良い．そのようなデータサイエンティストが増え，継続的に社会でインパクトを出すようになれば職業として定着していくのではないだろうか．

本稿の読者にはデータサイエンティストを目指している人も多くいるだろう．そのような人は広い意味でのサイエンティストを目指すとよい．実務で科学的な問題解決をする際にはデータサイエンスだけでは解決できないことが多いからである．データサイエンティストという流行りの職業に乗っかることもよいが，実務でサイエンスを活用できるプレイヤーが本質的には重要であることを忘れてはならない．

最後になるが，本稿を通してアカデミックで育った駆け出しのデータサイエンティストが実務でインパクトを出せるプレイヤーに成長することを，また経験豊富なデータサイエンティストには振り返りの機会を与えられることを期待したい．

参考文献

[1] 岩永二郎，“インパクトがだせるデータサイエンティストになるには（小特集 ビジネス現場におけるデータサイエンス）”，経営システム，Vol. 28，No. 2，pp. 127-132 (2019).
[2] 今野浩，「金融工学 20 年〜20 世紀エンジニアの冒険」，東洋経済新聞社 (2005).
[3] Tukey, J. W., *Exploratory Data Analysis*, *Preliminary Edition*, Addison-Wesley (1970).
[4] Zheng, A. and Casari, A., “Feature Engineering for Machine Learning: Principles and Techniques for Data Scientists,” *O’REILLY* (2018).
[5] Zheng, A. and Casari, A, 株式会社ホクソエム（訳），「機械学習のための特徴量エンジニアリング：その原理と Python よる実践」，オライリー・ジャパン (2018).
[6] 水野信也，“データサイエンス分野における人材育成のあり方〜オープンデータ活用研究部会の活動を通して〜”，日本ソーシャルデータサイエンス論文誌，Vol. 2，No. 1，pp. 39-42 (2018).
[7] 中川慶一郎，生田目崇，“ビジネス・アナリティクスの現状と将来（特集：データサイエンスの現状と将来）”，日本ソーシャルデータサイエンス論文誌，Vol. 1，No. 1，pp. 9-14 (2017).
[8] 須江雅彦，“我が国の未来を担うデータサイエンティストの育成―政策の動向と滋賀大学の挑戦―（特集：データサイエンスの現状と将来）”，ソーシャルデータサイエンス学会誌，Vol. 1，No. 1，pp. 3-8 (2017).
[9] 生田目崇，“ビッグデータ分析の最近の潮流（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 224-249 (2014).
[10] 宍倉剛，“ビッグデータ活用の実際とデータアナリティクスによる価値創造（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 230-236 (2014).
[11] 佐々木宏，“ビッグデータ・アナリティクスの組織適用とデータサイエンティスト（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 237-241 (2014).
[12] 佐々木良一，菊池浩明，“ビッグデータ時代のプライバシー保護（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 262-267 (2014).
[13] 牧野剛士，“BI コンサルタントから見た「ビジネスアナリティクス」の勘所（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 230-236 (2014).
[14] 青柳憲治，西郷彰，“インターネット・マーケティングにおけるデータ活用（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 247-251 (2014).
[15] 樋口進，“本当は難しいビッグデータのマーケティング活用（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 242-246 (2014).
[16] 小林元，“Web サービスにおけるデータ活用の課題と可能性（特集 ビッグデータ時代のアナリティクス）”，経営システム，Vol. 23，No. 4，pp. 258-261 (2014).
[17] Iwanaga, J., Nishimura, N., Sukegawa, N., Takano, Y., “Estimating Product-choice Probabilities from Recency and Frequency of Page Views,” *Knowledge-Based Systems*, Vol. 99, pp. 157-167 (2016).
[18] 岩永二郎，鍋谷昴一，梶原悠，五十嵐健太，“関心度と忘却度に基づくレコメンド手法：単調性制約付きレコメンドモデルの構築（特集 データ解析コンペティション: インフォミディアリ・データの分析）”，オペレーションズ・リサーチ，Vol. 59，No. 2，pp. 72-80 (2014).
[19] 西村直樹，鮭川矩義，高野祐一，岩永二郎，水野眞治，“EC サイトの商品特性を考慮した 2 次元確率表による購買予測（特集 データ解析コンペティション: 20 周年）”，オペレーションズ・リサーチ，Vol. 60，No. 2，pp. 69-74 (2015).
[20] Nishimura, N., Sukegawa, N., Takano, Y. and Iwanaga, J., “A Latent-class Model for Estimating Product-choice Probabilities from Clickstream Data,” *Information Sciences*, Vol. 429, pp. 406-420 (2018).