

1. 最終提出物と締め切りについて

最終成果物として授業で説明してきた「自作の Jupyter Notebook」を提出すること。

提出先：**takigawa@ist.hokudai.ac.jp** へメール提出

提出〆切：**2018年8月8日(水)午後17:00**まで

内容：Jupyter Notebook 形式で下記例のような内容(自由)をまとめたもの

- 「python」の基本や使い方について自分なりにまとめたもの
- 「プログラミング」について自分なりに分かったことを書いたもの
- 授業中の問題や自分でやってみたことについての python コード
- Q&A でのフィードバックや質問などで得たことのまとめ

2. 成績評価について

シラバスに沿って、出席 2/3(10回)以上を成績評価対象とし、授業や演習への積極性(30%)、レポートや成果物の内容(50%)、ディスカッションへの貢献(20%)を総合的に評価
※ただしディスカッションへの貢献はミニレポートの Q&A やコメントでの授業への貢献とします。(授業の内容はこれによって適宜微修正でフィードバックしてきました)
また授業や演習への積極性はミニレポートおよび最終成果物の内容から評価します。

3. 今日の内容

- フォローアップ Q&A
- この授業の評価について
- 授業のふりかえり：プログラミングとは？
- このあとどうする？
 - 授業でやらなかったこと
 - Python 以外の言語
- 授業のまとめ

先週の質問への回答：

Q. シラバスに書いてある第 12 回目に行われる予定だった「ディスカッション」は一体なんだったのでしょうか。

A. 当初このタイミングで、受講生をあてて発表してもらいディスカッションしようと予定していましたが、進捗にばらつきがあり、つまづいてる人もいたので、それよりも「ここまでの復習」に充てたほうが良さそうということで割愛しました。「ディスカッション」自体は、この Q & A コーナーで個々の進捗や興味に合わせてある程度実施できてるかなということ。

Q. 「Python ができる」とはどの程度をさすのでしょうか。

A. 「英語ができる」とはどの程度をさすのかと同じで、本人ができると思ってるならできると言って良いのではないかと思います。一般的には python で何かを作った経験がある程度ある感じでしょうか。

Q. SE やプログラマーはだいたい何種類くらいの言語をマスターしていますか？

A. できる人なら少なくともコンパイル言語を 1-2 つ、インタプリタ言語を 1-2 つで、4-5 個は使えるのではと思います。Google では C++、Java、Python が 3 大言語で、いまはそれに加えて Go という独自言語って感じだと思います。ただ職種にもよるところが大きく、Web 系なのか基幹システム系なのかによっても全然違いますし、同じ会社の SE やプログラマーといってもピンキリです。もし Python しか読み書きできなくても世の中には仕事はたくさんあるので！

Q. Practice 07 の問題 5 の「数当てゲーム Hit & Blow」のプログラムが作れません。数字があるかないかのみプログラムは作れたのですが、場所が合っていないかのプログラムが作れません。

A. 授業中にも言いましたがこれは典型的な問題なので本当にわからないので知りたいという意味ならエラーメッセージや「Hit & Blow python」でネット検索してみてください。自分で調べる点が大事です。

Q. 複数のプログラミング言語を勉強したいのですが、そういう場合は 1 つをある程度使いこなせるようになってから次の言語を勉強しはじめるのと複数同時進行で勉強していくのはどちらが良いと思いますか？

A. 基本的には前者が良いと思います。言語にもよりますが書き方が似ている言語もとても多いので同時にやると(たとえば ruby と python など)、どっちがどっちなのか混乱するのではと思います。また、基本的にどれか一つ気に入ったものを深掘りする、というのも大事です。どれも浅いとどれもあまり使いこなせませんが、一つ深掘りしておけば、他の言語の習得も圧倒的に早いし、使いこなせると思います。

Q. practice 8, 9 が難しく理解するのに時間がかかってしまっています。

A. この部分は実践的なのを少し体験してもらっただけなのでこの資料だけでそもそも完全に理解できるものではありません。本質的には数値計算や機械学習の知識が必要ですし、外部ライブラリにも習熟する必要があるので、まあこんなもんか、思ったよりコードちょっとで色々できるんだな、くらいが良いと思います。もし興味が沸いたら自分でいろいろ調べるきっかけにしてください。

Q. プログラミングによって様々なものが動作しているのは分かったのですが、プログラミングのソフト自体はどのように作成しているのでしょうか。

A. プログラミングのソフトもプログラムなので当然プログラミング言語で書くのが普通です。たとえば、みなさんが使っている python は Cpython というもので C 言語で書かれています。ソースコードはオープンなので興味あったらのぞいてください。<https://github.com/python/cpython>
一方で、にわとりとたまごみたいにそもそもの最初はどうしたのだ、と気になるとは思いますが、そのあたりは色々込み入ってくるので、最悪 0 と 1 を並べて PC の演算装置のコマンド列を作ればソフトウェアになるので、なんかで 0 と 1 を並べて作ったんだくらいでとらえておいてください。

Q. アプリのミニゲームのような簡単なゲームを作るのに先生方でどのくらいかかりますか。

A. 完成度とゲームの内容に依ります。たとえポゲームでも商売上リリースするなら何週間かそれなりに時間をかけるでしょうし、遊びでいいなら小一時間あればコアはできるのではないのでしょうか。ちなみに Scratch でも問題なくミニゲーム作れるので興味あったら挑戦してみてください。

Q. 職としてプログラマーを選んだ場合はブラックなところが多いのでしょうか。プログラマーという職へのつき方もよくわからないのですが、必要なスキルなどはあるのでしょうか。

A. ブラックなところが多いかは(入ってみないと)分からないしどんなプログラムを開発しているかという業種にもよると思います。必要なスキルはプログラミングと対象の基礎知識は入りそうですが、「プログラマー」はスキルがあるなら基本的に誰でもなれます。何か資格が必要なわけでもないし理系出身の人も文系出身の人もいます。対象の知識というのはコンピュータの仕組み、インターネットの仕組み、Web の仕組み、から始まって、例えば自動車のエンジン制御のプログラムなら、自動車やエンジンや制御理論の知識が必要、アドテクのプログラムならアドテクの関連知識が必要という感じ。

Q. `import sklearn.datasets` で例題のデータセットをインポートしているとのことでしたがこれは sklearn にもとからサンプルのデータセットが入っているのでしょうか？

A. 基本的には YES なのだけど、実際には `import` した段階で裏でダウンロードする形で実装してあるデータが多いです。これらのプリインストールのデータセットに用がない人も多いので最初から含めておくと、ファイルが大きくなってしまいます。

Q. 発展課題をやってみたら、`k-NeighborsClassifier` だけ訓練データの正解率が 100% になりませんでした。これはどういうこと？

A. 機械学習の世界へようこそ。K-NeighborsClassifier は訓練データと近いものを取ってきて多数決で決めるというシンプルな方式です。ただし、訓練データの正解率が 100% になることが良いとも限らないのが機械学習の難しいところだし、訓練データが内在する統計的傾向にも依存します。もし興味があったら「発展」すべく機械学習の入門書などを見てみてください。

Q. 今後自分でプログラミングに取り組んでいくためのモチベーションとしてこんなものを作ってみたいというのがまだ漠然としています。第 12 回の配布プリントで家計簿ソフトやそのほかいくつか例をいただいたのですが、今までの授業でやったことの組み合わせで作成でき、とっつきやすいものとして、どんなソフトがあるか、もう少し教えていただきたいです。

A. 「何を作りたいか」は個人が興味持てるものをやるのが良いので、特に作りたいものがないなら授業中の課題を元に発展させるとか、スキルアップで今日紹介した「プロコン」の練習問題とかをやるとかで良いのかも。無理してなんか作らなければいけないってこともないし、何ができるかまだわかりきっていないのでイメージが漠然としているのもあると思うので、専門の雑誌みてみたり、気になるアプリや Web サービスを探してみるとかも良いと思います。あとは身近なことが良いと思います。スポーツ好きなら自分の体調管理(Fitbit みたいな活動量計つかって?)するとか、食事と体重管理するとか、音楽が好きなら、音楽を自動生成してみるとか、音楽ファイルを整理してみるとか、本が好きなら、自分が読んだ本と感想を残しておくシステムとか、何もないなら受けている授業や課題の処理状況を管理するとか。もっと一般的にとっつきやすいのはやはりゲームだと思います。入門書にあると思うので、3 目並べとかを作ってみると良いかも。Scratch でもかなりできますし、すでに色々な人が作ったものがサイトで見られるのでコードは参考にできます。また Unity などを使えば結構見栄えのするものも作れます。

Q. 先生なら機械学習をどのように現実社会に反映させますか？イマイチ機械学習のいいところがわかりません。アマゾンのオススメとか？

A. アマゾンのオススメも、Facebook の知り合い suggest もそうですし、Google 検索、スマホの画面フック、音声認識や画像認識(カメラかざしたら人の顔に四角つくとか)をはじめ、ネットやスマホは機械

学習応用の塊なので、実は知らないところでもう機械学習のお世話になってるかも?? 興味わいたらぜひ機械学習の入門書とか雑誌記事とかあるいは私のホームページのスライドを check してみてください。http://art.ist.hokudai.ac.jp/~takigawa/index_ja.html

Q. 瀧川先生と横山さんにとって機械学習はどんなものだと考えていらっしゃるか知りたいです。研究者の立場でのことは学生の私との差があると思うので気になります。

A. 一言で言うと「研究対象」。「研究対象」だということは、まだわからないことがたくさんあるし、技術としてもこれで完成というものではないと思っているということで、まだまだ発展途上の技術とってます。もう少し込み入った答えは私のホームページのスライドを check してみてください。

http://art.ist.hokudai.ac.jp/~takigawa/index_ja.html

Q. 人工知能にプログラミングをさせることは可能でしょうか。

A. 人工知能はそもそも人間がプログラミングしなければいけないがやり方がわからないタスクを解く技術だ、と考えれば、それ自体新しいプログラミングの形と言われています。Software 2.0 と呼ぶ人もいます。人工知能でたとえば python のコードを書けるかという意味なら、あまりできてませんが、素朴なのはできる例が知られていて、現在ホットな研究分野(program induction)です。

Q. 将来的にプログラマーの仕事は AI に奪われると思いますか? 先生の個人的な意見を聞きたいです。

A. そもそもプログラマーの仕事の内容が変わると思います。そもそも AI 動かすならそれは少なくともプログラムしないとダメだし当然メンテも必要だし、AI が関与するシステムやインフラにもプログラムが必要でそこにもプログラマー必要なのでプログラマー自体がいなくなるってことは当面はないと思います。が、たとえば会計処理専任のプログラマーとか、ホームページ作成だけできる人とか、そういう人の仕事は減っていくかもね。それはしかし、AI のせい云々ではなく、技術は進展していくものなので、古い技術依存の仕事(たとえば昔の電話の交換手とか? 公衆電話のメンテする人とか?)は徐々に自然に減っていくので、そういう流れが少し早いというだけなのかなとも思います。

Q. 今まで簡単な構文が現実にどのような所で、どのように機能をしているか想像がつかないです。何か Python で作られているもので具体的な例があれば見たいです。

A. いままで挙げた例では、instagram、youtube、dropbox などの Web サービスがわかりやすいかな? もちろん、簡単な構文がわかればこういうものが作れるわけではなくて、Web とは? インターネットとか? という基本知識が必要だし、Web サービスを稼働するには Web サーバーが必要なので、その仕組みや設定や使い方の知識も必要だし、とその間の「プログラミング以外の知識」がもう少しついてこないとなかなか具体的な想像は難しいのかな? とも思います。高校で習った物理学がわかったところで、車のエンジンの仕組みやエアコンの冷暖房の仕組みが想像できるかというのと、そうでもないかも? というのと似ているような気はしますね。

Q. 先生は自分の cheat sheet をどれくらい持っていますか?

A. 常時でいえば一つも持っていません。何か新しいことを覚えるときは作って机においておいたりしますがずっと使っているとそのうち要らなくなります。

Q. プログラミングでわからないことがあったら、とぼして次の内容をやったほうが良いですか、それともわかるようになるまで勉強したほうが良いですか。

A. その「わからないこと」の内容によります。基本的なことなら飛ばせないだろうし、今の目的で使わない部分なら飛ばしてください。

Q. プログラミングで解けない問題はありますか。

A. あります。たとえば、あなたの人生の目的を見つけること。

Q. 実際どれくらいの量のデータがあれば使い物になる機械学習が可能なのでしょうか？ practice9 で 91% でしたが 1700 程度でこんなに精度がでるんですね。

A. まず、機械学習手法を工夫すればデータ数はそのままでもっと精度は出ます。で、量についてなのですが、機械学習は「統計学的しくみ」なので、量だけあってもゴミデータなら精度は出ません。逆に限られた設定での良いデータなら数十個でも十分精度が出る場合もあります。このあたりは機械学習のしくみの根幹にかかわるのでもし興味があったら機械学習の本や私のホームページのスライドを check してみてください。 http://art.ist.hokudai.ac.jp/~takigawa/index_ja.html

Q. 自分の PC で YouTube を開くと最初 1 回は必ず北大のシングルサインオンシステムのページに飛んでしまいます。原因はなんですか。なんとか直したいです。

A. この情報だけでは考えられる可能性が多すぎてなんとも言えません。ブラウザハイジャッカーみたいなマルウェア(コンピュータウイルス)だとしたら、北大のシングルサインオンシステムには飛ばさないと思うので、ウイルスではなさそうだけど。まず、Web ブラウザを変えても再現するのか(ブラウザの問題かそれ以外の問題かを判定)を確認してからだろうか。別のブラウザなら再現されないなら明らかにそのブラウザの設定の問題だと思います。一方、他のブラウザでも再現するなら PC 設定やネットワーク設定、使っているネットワークの問題とかかな？ まあ、なんとも言えませんが、こういうのは一個一個どこが原因かを絞っていくのが大事です。