00：00：00

田中：ATR脳情報通信総合研究所の田中と申します。

　国際脳では、グループ1の2に在籍、所属しておりまして、データプラットフォームの構築、データの管理、データベースの管理を行っております。

　今日は、国際脳のデータの前に、これもATRが関わっていました脳プロ・革新脳プロトコルでのデータについて、前処理方法の1つを少し紹介したいと思います。いわゆるHCP-styleではないレガシープロトコルといいますが、レガシーデータをHCP-styleで前処理できないかということで、今回紹介をさせていただきます。

　ビデオはこれで切ります。進めていきたいと思います。

　ちなみに、私もこの前処理方法はそこまでエキスパートではないので、エキスパートの皆さまにはもう正直言って聞いていただく必要もないような内容なのですけれども、もし初学者の方から質問があったら、むしろ参加されているエキスパートの方に回答をいただければと思っています。それでは、よろしくお願いします。

　今日の最初のチュートリアルということで、今回対象としているデータは先ほど少し説明をさせていただきました、いわゆるHCP-styleのプロトコルではないデータです。どこが一番違うのかといいますと、HCP-styleでは、T2画像やfieldmapは必須になっていると思います。ただ、レガシーデータはこれらが撮っていないようなデータが結構多いのです。私たちが集めていた脳プロのデータベースもまさにこれで、T2が撮られていなかったのです。ただ、それでもHCP-style、いわゆるsurfaceベースの前処理ができないかということで、今日紹介するやり方でそういうことができるのではないかということです。

　今回、対象とするのは、先ほどから出ています、脳プロ・革新脳プロトコルで撮像されたデータです。こちらは、国際脳でも一部のサイトで撮られていて、国際脳ではSRPBと呼ばれているものです。ですので、今回のチュートリアルが対象としているのが、私たちが集めて公開をしている脳プロのデータも対象になっています。こちらの詳細については、最後に少し紹介させていただきます。

　先ほど小池先生から、今回どういう方々が参加されているのか、一部革新脳に関わっている皆さんも参加されているということですので、お手持ちの革新脳データでも今回のチュートリアルは適用することが可能です。

　また、タイミングが、本当は公開されていれば良かったのですが、今、整備中で、今後近いうちに公開する予定の国際脳で集めたSRPBデータにも、今回のチュートリアルの手法は適用可能です。

　というわけで、いわゆるレガシーデータの解析、前処理についてこれから紹介をしていきますが、国際脳の本丸といいますか、HCP-styleのデータの前処理につきましては、午後にあります理研の林先生のチュートリアルの中で恐らくカバーいただけると思っています。

　それでは、進んでいきます。

　今日、対象とするSRPBプロトコルとはどういうものかということですけれども、ATRはこのSRPBプロトコルの中でもResting-stateのfMRIのプロトコルの策定に関わっています。こちらのRestingの詳しいプロトコルに関しては、こちらにリンクを貼っています。こちらに全て詳細があります。

　このRestingのポリシーがここに書いてあるのですけれども、小脳を含めた全脳を含めたものとします。その上でなるべくTRを短くするというものです。

00：05：03

　精神疾患の多施設・多疾患データベースということですので、精神疾患に関連の深い前頭前野を重視するような撮り方になっています。

　SRPBはそもそも多施設、多施設とはシーメンスマシーンだけではなく、いろいろなベンダーのマシーンも参画できるように、そこまでシビアな条件は設定していません。ただ、TRが2.5ということと、Phase encodingがPからA、PosteriorからAnterior、これだけは守っていただきたいという内容になっています。また詳しい内容についてはこちらのウェブページを参照いただければと思います。

　ちなみに、この「P→A」がなぜ太線になっているかというと、われわれがこういうプロトコルで集めてくださいと皆さんにお願いをして撮ってもらったのですが、やはりヒューマンエラーがどうしても生じてしまい、一部、A→Pで撮ってしまったデータがあります。それらのデータを合わせて解析をする際に、今日、私の後に恐らくハーモナイゼーションの手法について紹介いただくと思うのですが、そのときに施設間差で影響が大きかった要素が、このPhase encodingだったのです。P→AとA→Pでかなり撮れ方が違うということですので、P→Aが非常に重要なファクターであったということが後に分かったのです。国際脳ではP→AとA→Pの両方を撮っています。

　ということで、SRPBプロトコルの紹介はこれで終わりたいと思います。

　早速ですが、今から実際の前処理を進めていきたいと思います。事前に必要な準備は、皆さんへ小池先生からGoogle Docsを経由してお知らせいただいていたと思います。皆さんにはサンプルデータをダウンロードしていただけたと思います。サンプルデータは、Amazonのウェブサービス、AWSのシンプルストレージサービス、S3を経由して皆さんにダウンロードしてもらっています。あちらのリンクですけれども、1週間有効です。確か23日に皆さんにリンクをお知らせしていたと思いますので、今週の水曜日までリンクが有効ですので、まだの方はぜひダウンロードして、実際にプログラムを走らせてみてください（注意：現在このリンクからはダウンロードできません）。

　zipファイルをダウンロードしていただいて、ご自身の計算環境のところに解凍してもらっていると思います。一応1人分のデータになっています。もう一つ、FreeSurferのライセンスを入手して、ご自身の実際に計算をする環境に保存してください。今回はDockerを使っていますので、FreeSurfer自体をインストールしていただく必要は、今回のデモの内容ではありません。ですが、ライセンスが必要ですのでよろしくお願いします。

　では、大きく解析の流れを説明したいと思います。今回はスリーステップを考えていて、1つはBIDS変換というものです。これはまたあとで紹介します。2つ目に、いわゆる前処理になります。3つ目が、2つ目のステップで前処理をしたデータを使ってどういう解析ができるのかという一例として、Resting-stateのFunctional connectivityを実際に算出してみようということを、デモで少しやりたいと思います。

00：09：59

　最初に、BIDSの話をしたいと思います。BIDSとは何ぞやといいますと、いわゆる共通フォーマットの名前だと思ってください。これは脳画像実験、fMRIによらず、最初はfMRIだけだったのですけれども、今はEEGやMEGも多分含まれていると思います。それらで得られるデータの標準フォーマットとして提唱されているものです。

　どういうものかというと、ファイルやフォルダーの名前や構成、メタデータの記載に関する様式になっています。詳しくはBIDSのサイトを見ていただいたら非常に丁寧に紹介されていますし、『Scientific Data』の2016年に出た論文にもBIDSの心や、こういうものであるということが紹介されていますので、興味がありましたら、また参照ください。

　簡単にここを説明しますと、dicomで皆さんデータを取っていると思います。こういうトップディレクトリがあって、その後にpaticipants.tsv、いわゆる被験者のdemographicがtsv形式であって、その下にsub-01、sub-02、sub-03という被験者ごとのディレクトリ、フォルダーがあります。その1人の中にanatomical、functional、dwiがあり、この中にnii.gzとjsonがペアになっている、こういう形式です。

　こういうフォルダー、ディレクトリの構成や名前の付け方をルールに従っていきましょうというものです。

　こちらはBIDSのメリットが何かということですけれども、1つは、標準フォーマットということで管理しやすいということです。

　また、自分の研究室だけでデータを触っている分にはBIDSのメリットはそれほどないのかもしれませんが、例えばデータを集約したり、共有するときに、結構コミュニケーションコストがかかります。そういうコストやヒューマンエラーを減少させるというメリットもあります。

　また、皆がBIDSを使うようになれば、データ解析ソフトウェアの最適化や自動化も安易にできるということです。

　また、4つ目は、皆さんが論文を出されるときに、近年、特にデータを公開せよと要求が結構あると思うのです。fMRI業界の1つの大きなデータ公開プラットフォームにOpenNeuroというものがあります。OpenNeuroは実はBIDSしか取り扱っていません。BIDSでアップしてくださいとしています。また、OpenNeuroだけではなく、いろいろな公開プラットフォームでもBIDSだと簡単にアップできるということで、これらのメリットがBIDSを提唱しているPoldrackたちから言われています。

　今回、なぜBIDSを使うかということですけれども、BIDSでいろいろなアプリが開発されていて、そのアプリの一つ、fMRIPrepを今回使うので、BIDSに変換をしています。

　さて、BIDS変換の流れを簡単に説明させていただきます。ただ、今回はもうBIDSに変換をした後のデータを、皆さんにダウンロードいただいたデモデータの中に入れてあります。大きく流れを説明しますと、dcm2bidsを使います。これはPythonのモジュールになっています。dcm2bidsのページに行っていただいたら、これも丁寧に説明があります。まず、dcm2bids\_scaffoldというコマンドで、先ほど紹介をしたサブジェクトの下に、anatomical、functional、dwiという構造があるのですけれども、この構造を作成します。そこにそれぞれ必要なjsonファイルを作成していきます。dicomdataの中にDICOM形式のデータを置きます。そうしますと、その後にまだあります。paticipants.tsv、jsonファイルを作成します。

00：15：00

　その状態まで準備をして、dcm2bidsが実行できます。その後にbids-validatorによるBIDSデータの整合性をチェックします。

　これで画面全体が見られますか。レスポンスがないのですが、見えているとしていきたいと思います。チャット、ありがとうございます。

　今回、皆さんにもうダウンロードして解凍をいただいているdemoというフォルダーの中の構造になっています。ここにsourcedataがあります。ここが既にBIDSになっている状態です。では、本当にBIDSになっているのかを、bids-validatorを使って検証したいと思います。

　今、ブラウザ画面になっているのですけれども、「bids-validator」とGoogleで探していただいて最初に出てくるのが、bids-validatorのページです。ここでファイルを選択して、先ほどのsourcedataをアップすると、このようにきちんとBIDSかどうかを見てくれます。1件だけWarningが出ていますけれども、Warningはそれほど気にしなくてもいいです。ここに「Ignore Warnings」というのもありますので、この状態でもう一回先ほどのsourcedataを選んでいただくと、このように「This is a valid BIDS dataset!」となります。このように非常に簡単にBIDSになっているかどうかの評価ができます。皆さんもこれをやってみてください。ちなみに、このデータは別にbids-validatorを管理しているところに行くわけではありませんので、ご安心ください。

　bids-validatorはウェブベースではなく、Pythonのアプリもありますので、これでやるのは少し心配だと思っている方は、そちらでやっていただいてもいいかと思います。

　では、そのように皆さんにお渡ししているこちらのsourcedataが、きちんとBIDS形式になっているということで先に進みたいと思います。

　これもそうです。BIDSの中身です。Anatomicalの中にsub-0001のT1wのnii.gzとjsonがあります。fmapの中にmagnitudeとphasediffがあり、magnitudeは2つあります。phasediffはそれぞれのnii.gzとjsonです。functionalの中には今回はRestingのデータだけですので、task-rest\_boldのnii.gzとjsonが入っている状態です。

　では、この状態でfmriprep\_ciftifyによる前処理を紹介したいと思います。前処理なのですけれども、冒頭で説明したように、今回はいわゆるsurfaceベース、脳表面の座標系に変換する前処理を紹介したいと思います。脳表面の座標系に変換しない前処理も当然あります。ボリュームベースの前処理というものです。恐らくfMRIの研究でこれまで一番多く行われてきたのが、SPMによるボリュームベースの前処理だったと思います。そちらで前処理していただいても当然大丈夫ですし、絶対これをやらなくてはいけないというわけではなく、今回は前処理の1つを紹介するということです。

　前処理なのですけれども、今回、fmriprepとciftify toolboxを実行します。fmriprepを用いて脳画像の一般的な前処理を実行し、その後にciftify toolboxで前処理後のデータを脳表面の座標系に変換します。

00：20：04

　今回はfmriprepとciftify toolboxをパッケージ化した、Dockerのコマンドでありますfmriprep\_ciftifyを利用します。

　ただ、多分詳しい方はご存じだと思うのですけれども、実は現存のfmriprepの中にsurface preprocessingが含まれています。ですので、実はciftifyをわざわざ使わなくても、fmriprepのみでも脳表面の座標系への変換は実施可能です。ただ、ATRで開発していたのがfmriprep\_ciftifyを使った前処理ですので、今回はこれで紹介をさせていただきます。

　fMRIPrepはどういうものかといいますと、それを説明しますとこれだけで多分時間が終わってしまうので、ご存じの方は当然スキップしていただいて大丈夫ですし、どういうものかを知らない方は、こちらの論文でまとめられていますし、ウェブページにも非常に丁寧に説明がありますので、またご確認ください。

　ここにありますように、fMRIPrep自体もpreprocessingのPipelineです。Pipelineというのはつまりいろいろな前処理のツールから、ライブラリーといったら変でしょうか。いろいろなパーツを組み合わせて、一連の前処理の流れにしているというのがPipelineになります。

　今回はPipelineを付けて構成しているということで、少し冗長なところがひょっとしたらあるのかもしれませんし、ややこしいのですが進めていきたいと思います。

　ciftifyですけれども、こちらのほうが概念的には分かりやすいかと思っています。つまり、左にあるのがいわゆるHCPで行われている前処理で、右側がciftifyで行う前処理になっています。冒頭に言いましたが、fieldmapやT2があるという前提でHCPは前処理をしていきますが、fieldmapやT2がなくても、無理やりという言い方は変ですが、surfaceベースに落とし込もうというのがciftifyです。

　ciftifyも論文がありますので、またこちらを参照しておいてください。

　では、今回fmriprepとciftifyをDockerで一気にやってしまおうという内容になっています。ここで皆さんのお手元にサンプルスクリプトをご準備ください。私の左側にVS Codeというエディターを使って、サンプルコードを見ていきたいと思います。

　こちらはシェルスクリプトになっていますので、実行はコマンドラインにターミナルで行ってください。私が今日使っているVS Codeは、この下にターミナルが付いていて、ここの部分がターミナルでここがエディターになっています。ですので、実行は下のターミナルで行うようにします。

　こちらはサンプルスクリプトなのですけれども、ざっと見ていただくといいと思いますが、環境に合わせて変えていただくのが、27行目から29行目になります。あとfmriprepとciftifyのバージョンもあります。すみません、ここも変えてください。ここは皆さんの使いたいバージョンに合わせて変更してください。

　まず、トップディレクトリをサンプルスクリプト一式、サンプルデータ一式のデモが置いてあるところに変更をしてください。ちなみに私はMacでデモを行っています。28行目をFreeSurferのライセンスが置いてあるパスに変更してください。

00：25：02

　29行目なのですけれども、これは計算機で使うCPUのコア数になっています。一応デフォルト8にしてありますが、これも皆さんが使うマシーンに合わせて変えてください。

　事前準備で皆さんにお知らせしたかと思いますが、こちらはDockerを入れておいていただく必要があります。Dockerの入れ方からやると時間がかかってしまいますので、もう皆さんのマシーンにDockerが入っているという前提で進めていきたいと思います。ちなみにDockerがきちんと動いているかどうかに関しては、Macですとここです。ツールバーというのですか。上にくじらのマークがありますので、ここを見ていただくと、「Docker Desktop is running」と書いてありますので、今きちんとDockerが走っている状態ですということです。Windowsは私も検証していませんが、恐らくこれと同じような形でDockerが走っているというお知らせがどこかにあると思います。

　さて、ここまで来たら、あとはDockerがインストールしてある計算機でこれを実行するだけです。ですので、今ターミナルに移っていますが、fmriprep\_ciftify\_demo.sh、このコマンドをやっていただいたら大丈夫です。ちなみにこのsourcedataがBIDS形式のディレクトリの名前を指定していて、0001がサブジェクトのIDです。最後のATRはサイト情報なのですけれども、これは書かなくても大丈夫です。

　これを実行していただくと……、少し待ってください。すみません。これはdemoでした。スライドのほうが間違っていました。このような感じで走りました。これでざっと出てきまして今走っている状態です。この状態になったら大丈夫です。無事に走っているということになります。

　ちなみに、処理にはかなりの時間がかかります。私が今デモを行っているのがMacプロで、一応DockerでCPUのコア数やメモリが設定できるのですけれども、この条件で1人分に約20時間かかりました。恐らくDockerの設定をうまくすれば、もう少し早く終わるのではないかと思います。ちなみにATRのクラスターでは1人8時間ぐらいという話だったので、ハイスペックマシーンを使えば、その辺りまで恐らく時間は削減できるかと思っています。

　こちらのサンプルスクリプトですけれども、皆さんのサンプルデータ一式をダウンロードしていただいて、このsourcedataや被験者のIDの名前、そういうものをそのままにしていただいていたら、走るようになっています。

　結局Dockerのコマンドをキックしているのはここだけです。ここの部分になりますので、ここは触らないで大丈夫かと思います。

00：29：59

　というわけで、ここは皆さんに実際に実行していただくのが難しい内容になっているのですけれども、皆さん、また後日試していただいて、走らなかったなど、ここはどうですかという質問があったら、またなるべく受け付けたいと思いますので、ぜひ試してみてください。

　私のほうは走らせておこうと思います。走らせているときに非常にWarningが出てきたり、コントロールCで終わってくれなどと出てくるのですけれども、終わっては駄目です。走らせたらひたすら待ってください。20時間ぐらいたつと、終わりましたと終わります。1週間ぐらいたっても終わらなかったら、それはもしや何か問題があるかもしれませんので、またご相談ください。

　これで取りあえず待ったら結果ができます。皆さんにダウンロードいただいているこのデータ一式の中に、実は1人分の前処理済みのデータが納まっています。demoの中のderivativesの中に、これは今走らせてできているフォルダーなのですけれども、こちらの終わりに「\_0」と書いてあるのが、もう実際に実行して完了した前処理のデータになります。

　この中身ですけれども、このような感じでたくさん出てきます。では、前処理がうまくいったかどうかを確認するのですけれども、それぞれfmriprepとciftifyでこのように解析が終わりましたというQCのためのファイルが出来上がっています。ですので、それを確認していきたいと思います。

　まず、fmriprepですけれども、こちらを見ていただきますと、「sub-0001.html」というのがあります。こちらをクリックしていただくと、このように前処理のSummaryが出てきます。これを見ていただくと、きちんと前処理ができているのが分かるかと思います。

　上に書いてありますSummaryがこの被験者IDで、Structural imageとFunctional imageを解析しましたと。こちらを見ていきましょうか。Anatomicalではこのようなパラメーターの画像でしたと、このように前処理のSummaryが出てきます。きれいにできているのではないかと思います。surface reconstructionもきちんとfmriprepが実はやってくれているのです。

　ここまでがAnatomicalで、以下はFunctionalになります。Functionalも、判断は難しいのですけれども、このように結果が出てきていると思います。きっときれいにできています。ここがEPIとT1のレジストレーションで、きちんと位置合わせが合っているかということです。これもきれいにいっていると思います。

　その下に、いわゆるBOLD、fMRIのQCの一覧が出てきます。BOLDもSummaryもこのような感じで出てきます。最近これらの値も、データがきちんと取れているのかどうか、QCのアウトプットとしてデータにまとめて出す場合もあります。特にたくさんデータを取ってデータベースを公開するような場合には、これらの指標は表にまとめて出すことを最近は要求されます。これは1人分なのですけれども、このように目視で確認できると思います。

　この辺りはすごいです。Methodにこれをコピペしたら論文が書けることになっています。最後にNo errors to report!　ということで、エラーもなくきれいに終わりましたということです。

00：35：05

　ここがfmriprepのQC、前処理のSummaryになります。

　では次に、fmriprep\_ ciftifyのQCを見ていきたいと思います。ciftify のほうです。ciftifyは構造画像の処理と機能画像の処理のSummaryが別々になっていますので、まずは構造画像から見たいと思います。derivativesの中のfmriprep、これはバージョンの名前なのですけれども、この下にciftifyがあって、qc\_recon\_allの下にサブジェクトのディレクトリがあって、ここにqc\_htmlというのがあります。これを見てみるとこのような感じです。

　チャットに質問をいただいていますが、これは私のほうでは回答できかねます。私も正直に言うとよく分からないので、もし詳しい方がおられたら代わりに回答いただけると大変助かります。

　こちらはciftifyのsurfaceの結果になっています。これを見ますと、きれいにできているのではないかと思います。こちらは構造です。

　次にfMRIですが、qc\_fmriの中の被験者のタスクの中にqcがありますので、こちらも見ていきたいと思います。このような感じです。これもどうなのでしょうか。きっときれいにできているのだろうと思います。fMRIは私も実は知らなかったのですけれども、Correlationの結果も出てきているのです。これは個人的に非常にいいと思っています。Dorsal Attention NetworkとDefault Mode Networkと、Salience Networkで、seedベースのCorrelation（functional connectivity）をプロットしてくれているのです。これで例えばDefault Mode Networkがきちんと出てきているのであれば、取れているのではないかと判断ができるかと思います。

　こちらのQCに関して、確かにどこまでどういう基準でやればいいのかが私も分からず、例えばATRで前処理した方に、うまくいかなかった例を見せてもらったのです。なぜうまくいかなかったのかを聞きそびれているのですが、これはciftifyのfMRIのデータです。これは先ほどお見せしたうまくいった例なのですけれども、これはうまくいかなかったらしいです。これを見ると明らかです。そもそもずれています。

　このように、その方が言うには、まずはciftifyのfMRIのデータからQCをして、それで何が失敗だったかをさかのぼって見ていくということです。この辺りの技は、ひょっとしたらエキスパートの先生方もご存じかもしれませんので、こうやってやるといいということがあれば、共有いただけると助かります。

　ここまででfmriprep\_ciftifyを使った前処理の紹介をさせていただきました。

　では、前処理したデータを使って具体的にどのように解析をしていくのかという話に、移りたいと思います。前処理というのは、いわゆる前処理ですので、それだけでは研究になりません。そういう研究もありますが、例えばRestの解析をするのであれば、そこからどのようにデータを解析していくのかということが本番かと思っています。

　今回、安静時のfMRIのデータを使っていますので、安静時のfMRIで一番一般的に行われているFunctional connectivityの解析方法について、1つ紹介したいと思っています。

00：39：56

　根本先生から質問をいただいていますが、すみません。私はまだここをフォローしていないので、詳しい方がおられたらご回答ください。今日、回答がいただけなかった場合は、私のほうで調べて、また根本先生に回答をさせていただきたいと思います。いろいろフォローできていなくて申し訳ありません。

　さて、RestingのBOLDデータを使ってやっていこうということなのですけれども、Functional connectivityの算出については、オーソドックスにはこの3つのステップかと思っています。まず、RestingのfMRIから時系列を抽出します。Functional connectivityはいわゆる領域間の時系列の相関というものなのですけれども、この2領域が、例えばvoxel to voxelの場合もありますし、voxel to ROIの場合もありますし、ROI同士のものもあると思います。

　今日紹介するのは、ROI同士のFunctional connectivityになります。Restingというのは、5分間や10分間、ひたすらぼーっとしていただく課題になっていますので、やはりノイズの影響が大きいということで、体動等の交絡因子を取り除くことが非常に重要になってきています。この辺りの交絡因子の取り除き方だけで、多分論文が1本書けるぐらいの非常に深い世界ですので、今回は飛ばします。

　でも、Restingの論文を出すと、最近、レビュー側からこの辺りは非常に鋭く突っ込まれますので、最新の動向などを見つつ、こういう手法がどうもいいらしいということを、皆さんも調べていただけたらいいかと思います。

　最後にFunctional connectivityを計算します。ここも相関を見るのですけれども、相関のPartial Correlationなど、いろいろあると思いますが、今回は素直にPearson’s Correlationを出しています。

　ただ、surfaceベースの前処理をしていたにもかかわらず、今回、FCを計算するのはボリュームベースになります。その辺りはご了承ください。チャットでも議論が進んでいてありがとうございます。こういうのをきっと集合知と言うのですね。

　では、皆さんにお渡ししているサンプルスクリプトをご確認ください。今回はPythonで作っているスクリプトです。Pythonも私は初心者であまり詳しくなく、教えてもらっているような状態なのですけれども、Pythonはいろいろなモジュールがあり、今回はその中のnilearnを使っています。詳しい方は多分どんどん使っていると思うのですけれども、nilearn自体がNeuro imagingに特化しているモジュールですので、本当にいろいろなことができます。その中で今回、一部のライブラリーを使っています。FCの計算に使う前処理済みのNiftiファイルをまず指定します。これはfmriprepのoutputファイルになります。交絡因子が大変多く出力されます。今回、ここが交絡因子の内容になっています。35行目と36行目です。これがなぜいいのかというのも私はよく知らないので、取りあえずこれだけ入っているということです。

　FCの計算に使うNiftiファイルですけれども、先ほどのderivativesの中にfmriprepのsub-0001の中のfunctional の中にあります、これもたくさんあるのですけれども、これです。

00：45：19

　こういうものです。これが前処理後のNiftiファイルになっていますので、これを指定してください。このスクリプトの中では指定はしてあります。

　あとはconfoundsファイル、交絡因子です。こちらもconfounds\_regressors.tsvの中に入っています、同じディレクトリの中のこれです。

　ちなみに、ここにお名前を入れているのですけれども、fmriprepのバージョンで、微妙に交絡因子の名前にアンダーバーが入っていたり、入っていなかったり、大文字だったり、小文字だったりするので、もし動かなかったらそこをご確認ください。

　ですから、tsvファイルを実際に開いていただいて、ここにそれぞれのconfoundsの名前が書いてあるのですけれども、これを実際に出てきたファイルの、confoundsの変数名とこのスクリプトの変数名を合わせていただく必要があります。

　あとはROIの定義ファイルですけれども、今回、nilearnで定義されていますアトラスを使用していますので、スクリプトの中では本当に読み込んでいるだけになっています。ここです。PowerとDosenbachのROIというか、アトラスを使っていますので、ここで呼び出しているだけですけれども、例えばお手元にアトラスがある場合でもこちらは対応可能です。その場合の例を出していないので申し訳ないのですけれども。

　ただ、nilearnの中にこのアトラスもたくさん用意されていますので、好みといいますか、目的に沿って、どのアトラスを使うのかというのは選ばれたら良いかと思います。このスクリプトでは例として、この2つのPowerとDosenbachのROIを使ったFCの算出をしています。

　では、サンプルスクリプトを見ていきたいと思います。これも変えていただくのはここからになるかと思います。31行目です。トップディレクトリをまた先ほどと同じように、デモのディレクトリを解凍したところのパスを通してください。デモの中の構造を変えなければ、この辺りはそのままで大丈夫です。

　今、「\_0」になっているのが、皆さんの手元のもう前処理済みの結果が入っているところを指定していますので、ここは皆さんご自身で、もし前処理をされたら変えていただく必要があるかと思います。

　derivativesの中のFCというディレクトリだけを作っていただいたら大丈夫です。この下にそれぞれのアトラスの名前で、FCの結果が出るようになっています。

　今回、実はFCの中にPowerたちの264個のROIを使ったアトラスの結果が入っていますので、今回はDosenbachを走らせてみたいと思います。この辺りは、trが今回の脳プロ・革新脳プロトコル、SRPBでは2.5ですので、ここでもう2.5と定義をしています。あとはもしサンプルスクリプトを触っていただくとすると、high pass、low passもこちらで合わせていますので、ご自身のデータのプロトコルに合わせて変えていただければいいかと思います。

　FCの図を書くときも、thresholdも適時変えていただければと思います。

00：50：04

　あとはこれをPythonで実行していただいたら大丈夫です。

　では、私のほうはVS Codeのターミナルから実行したいと思います。少し時間がかかっています。出てきました。このような感じです。実行していただくとこのように、上がいわゆるconnectivity matrixです。この1つが、ですから、1辺が、ROIの数160で、それぞれのCorrelationがプロットされます。マイナス1から1までのカラーバーで出てきます。こちらがDosenbachのアトラスにこのCorrelationの値を乗せたものです。これは先ほどのthreshold定義していました99.8で、thresholdを切ると、このような感じで出てきます。

　99.8で切っているので、線が通っているのは少しだけです。ですから、丸が160個あるということになると思います。こちらのスライドにあるのがPowerのアトラスになります。Powerのほうは264個ありますので、Dosenbachよりも少し密な感じになっています。

　先ほど、このスクリプトを走らせていただきますと、このようにsub-0001\_rest\_matというmatファイルが出来上がります。これをMATLABで読んでいただくようになっています。Correlationの値がマットファイルに入っているという、それだけです。フィギュアに関してもスクリプトを走らせるだけではセーブされないのですけれども、ここでセーブボタンをカチッと押していただいたら、今Dosenbachを走らせたのでセーブ、これもセーブという感じでセーブされます。

　このようにFCの計算と可視化ができるようなスクリプトをサンプルスクリプトとして配布していますので、皆さんも試してみてください。

　Functional connectivityの解析がいろいろできますというのが、見づらくて申し訳ありませんが、nilearnのサイトで紹介されています。今回スクリプトにある内容もnilearnのサンプルから取ってきたものです。nilearnでいろいろな解析および可視化ができますので、Pythonに詳しい方もそうでない方もいろいろ試してみるといいかと思います。

　では、ここまでのところでデモは一通り終わりになります。最後に、SRPBのデータ一式を皆さんにもダウンロードして使っていただけるようになっていますので、これを紹介して終わりたいと思います。

　このデータはAMEDの国際脳の前に走っていました、脳プロで集めたデータになっています。脳プロのBMI課題というものがあり、そこで、疾患が今ここにありますとおり、自閉症スペクトラム、うつ、統合失調症、強迫性障害、慢性疼痛、および腰痛、あとはその他で、多疾患のデータを集めていました。健常者も同じぐらいの数を多施設で集めています。

00：55：04

　多施設は、ここで紹介していますけれども、東日本、西日本、および関西、真ん中辺といろいろな所で取っています。スキャナーもシーメンスだけではなく、GEやフィリップス、いろいろなベンダーのスキャナーで収集したデータを集約しています。

　2018年3月までがAMED脳プロBMI課題の期間だったのですけれども、そこまでで2,409例を収集して、多施設・多精神疾患データベースとして構築しています。

　この中の一部の患者さんおよび被験者さんに、公開していいという同意を取っていますので、そのデータの制限公開および非制限公開を終えています。

　こちらですけれども、大きく4種類のRestingの多施設・多精神疾患のデータベースが入手可能になっています。非制限公開と制限公開という2つのくくりで公開をしています。制限公開は、簡単にメールアドレスと名前と所属さえ登録していただいたら、こちらで何か承認をせずともデータが使えるという状態になっています。

　こちらが2種類ありまして、11施設、1,410名分の疾患および健常者の脳画像が入手可能です。この場合の脳画像は、T1構造画像と安静時のfMRI、および、オプションになるのですけれども、fieldmapの画像になっています。こちらは全てNifti形式で使用できるようになっています。

　また、この後にお話があると思うのですけれども、こういう多施設で撮ったデータを合わせて解析するときに、施設間差というものが非常に大きくなってきます。それを補正して合わせて解析ができるようにするのが、ハーモナイゼーションといわれている手法です。そのためには、同じ人がハーモナイズするサイトを巡ってデータを取得する、トラベリングサブジェクトが有効であるといわれています。ですので、脳プロBMI課題でも国際脳に先駆けてトラベリングサブジェクトを行っていまして、9施設、9名分のデータが入手可能になっています。

　また、制限公開のデータがこれより少し多くなっています。こちらの制限公開に関しては、フォームに入力していただいて、アカデミック・ユース・オンリー、商用利用は2つとも不可になっているのですけれども、制限公開は条件がいろいろありますので、それをクリアしていただいて、こちらで承認した後にデータがダウンロードできるという仕組みになっています。

　こちらは、前の画像に加えて、前処理して、先ほど紹介したようにFunctional connectivityのmatrixの形になったデータも公開しています。ですので、前処理も全部飛ばしてFunctional connectivityだけを使って、例えば機械学習のデータとして使いたいというときは有効かと思っています。

　ここにありました。すみません。トラベリングサブジェクトのデータなのですけれども、ハーモナイゼーションという手法に使うことができます。私たちATRでも2019年『PLOS Biology』に山下歩さんという方が中心になって、トラベリングサブジェクトのデータを使って、ハーモナイゼーションの手法を開発した成果を発表しています。

01：00：00

　このハーモナイゼーションを使って施設間差を補正することで、多施設・多疾患データを使ったバイオマーカー、回路マーカーの精度向上もできたということになっています。国際脳でもハーモナイゼーションの手法の開発を行っていく予定です。

　この脳プロBMI課題で集めたデータを使って、たくさん今のところ論文が出ているということですので、また興味がありましたら見ていただけるといいかと思います。

　これで終わるのですけれども、ただ、最後に、今日紹介したBIDSからBIDSアプリを使って一連の前処理を行ったのですけれども、今現在、データベースでダウンロードできるデータがBIDS形式になっていません。申し訳ないのですけれども、今回のサンプルスクリプトを脳プロBMIのデータに使う場合は、BIDSに変換していただく必要がありますので、その点だけご了承ください。

　では、ここまで、スムーズにできなかったところも多々ありますが、私の担当を終わりたいと思います。

01：03：11