実習！トランスオミクスのデータ解析（7日目課題集）

小鍛治　俊也

* 目的

UNIXコマンドを駆使して、オミクスデータ解析を体験する。

* 準備

実習ウェブサイトから各種ファイルをフォルダ”Transomics”にダウンロード＆解凍

* トランスオミクスStep (iii): リン酸化された責任代謝酵素を同定する

phosphoproteome.txt の１列目には、リン酸化された代謝酵素の EC number が含まれている。ここから、責任代謝酵素に関する行のみ抽出する。

1. phosphoproteome.txtをExcelで開き、データの特徴を捉えなさい。予めExcelを起動しておいて、Drag & Dropで開ける。

（1列目：酵素番号 = EC番号、２列目：タンパク質ID、…、８列目：リン酸化残基、９列目以降：数値）

1. リン酸化された代謝酵素の[EC number, IPI ID, リン酸化残基]　の情報を抜き出し、phospho\_ec\_ipi\_residue.txtを作成しなさい。先頭２行は見出し行なので、３行目から出力すること。重複の除去を忘れずに。

（参照：UNIX課題5\_Q2、よく使うコマンド”cut”）

1. （不安であれば）模範解答ファイル‘Day7\_AnswerFiles/Step\_iii/　phospho\_ec\_ipi\_residue.txt’と一致しているかを確認せよ。（ヒント: diffコマンドを使う）
2. phospho\_ec\_ipi\_residue.txt から責任代謝酵素リスト(rme.txt)にも含まれている代謝酵素のみを抜き出しphosphorme\_ec\_ipi\_residue.txtを作成せよ。for文でrme.txtのそれぞれのEC番号をphospho\_ec\_ipi.txtからgrepするとよい。ただしgrepは単語一致を指定するオプションwを用いる（ec:1.1.1.1とec:1.1.1.10を区別するため）。

$ for aEC in $(cat rme.txt) ; do grep -w この空欄を埋める; done > phosphorme\_ec\_ipi\_residue.txt　←わからない人はこの部分をコピペ。時間少しかかる。

1. Q3 で作成したphosphorme\_ec\_ipi\_residue.txtの内容が、模範解答ファイル‘Day7\_AnswerFiles/Step\_iii/ phosphorme\_ec\_ipi\_residue.txt’と一致しているか確認しなさい。
2. phosphorme\_ec\_ipi\_residue.txtからリン酸化された責任代謝酵素のIPI IDとEC番号を抜き出したファイルphosphorme\_ipi\_ec.txtを作成せよ。（ヒント：awkで2列目と1列目を抜き出し、タブ文字を忘れずに、重複削除）
3. Q6 で作成したphosphorme\_ipi\_ec.txtの内容が、模範解答ファイル‘Day7\_AnswerFiles/Step\_iii/ phosphorme\_ipi\_ec.txt’と一致しているか確認しなさい。
* トランスオミクスStep (iv): 責任キナーゼの推定

キナーゼ推定ソフトウェアNetPhorestを用いて、責任代謝酵素をリン酸化したタンパク質キナーゼを推定する。

Q1. まずはお試し用のタンパク質配列ipi\_sample.fastaを使い、以下の手順に則ってNetPhorest の動作を体験してみよう。

* 1. TextEditでipi\_sample.fastaの中身を確認する（TextEditにDrag & Drop）。タンパク質ID\_IPI00197210とそのアミノ酸配列から構成されていることがわかる。このような形式をFASTA形式という（詳しくはQ2で）。
	2. lessなどでresnum\_sample.txtの中身を確認する。タンパク質ID\_IPI00197210とその490番目のアミノ酸であるSerineを表すS490が記されている。このファイルはタンパク質のどの位置がリン酸化されているかの情報を与えるために用いる。
	3. NetPhorest (<http://netphorest.info>) のウェブサイトを開く
	4. FASTA形式のアミノ酸配列ファイルipi\_sample.fastaの内容を問合せフォームに貼り付けて、[Select Phosphosite]を押す。
	5. 配列相同度が最も高いタンパク質を選択し、[Next]を押す
	6. "Load sites from file" にて、resnum\_sample.txt を選択する。これにより、タンパク質のIDと当該タンパク質において実際にリン酸化が確認された残基の位置が与えられる。最後に[Start prediction]ボタンをクリックする。
	7. 今回はキナーゼによるリン酸化のみ使うので、8つあるチェックボックス “KIN”, “SH2”, “PTP”, “PTB”, “14-3-3”, “BRCT”, “WW” ,”WD40” のうち ”KIN” のみチェックする。
	8. [Save] → Full dataset を選択し、キナーゼの予測結果をnetworkin\_predictions\_sample.tsvを保存し、Excelなどで中身を確認する。タンパク質IPI00197210の490番目のSerineのリン酸化の予測キナーゼやその予測スコアが記されている。（その他には、アミノ酸配列やキナーゼのタンパク質IDが記されている。）

以上で見たように、NetPhorestへの入力として必要な情報は、

1. FASTA形式のアミノ酸配列 (例：ipi\_sample.fasta)
2. リン酸化が実際に確認された残基の位置 (例：resnum\_sample.txt)

の2つである。

　以下のQ2およびQ3では、手持ちのデータを組み合わせて、上記「1」および「2」に相当するデータファイルをそれぞれ作成する。

Q2. ipi.RAT.fastaは、ラット全タンパク質のアミノ酸配列をFASTA形式で格納しているファイルである。FASTA形式とは、以下のようなファイル形式である。

>[IPI]

[60文字ごとに改行したアミノ酸配列]

[60文字ごとに改行したアミノ酸配列]

 （以下、配列の末尾まで同様）

IPIはタンパク質を識別するID体系のひとつ、International Protein Indexを指す。

1. ipi.RAT.fastaをless等で閲覧せよ。見出しとアミノ酸配列が繰り返されていることがわかる。
2. fasta形式のファイルの操作にはいくつかのツールが開発されている。今回はそのうちのseqkitを使用する。以下のコマンドを実行し、その動作を確認せよ。

$ ./seqkit grep ipi.RAT.fasta -p IPI00197210

>IPI00197210

MPLELTQSRVQKIWVPVDHRPSLPRSCGPKLTNSPTVIVMVGLPARGKTYISKKLTRYLN

WIGVPTKVFNVGEYRREAVKQYSSYNFFRPDNEEAMRVRKQCALAALRDVKSYLTKEGGQ

IAVFDATNTTRERRHMILHFAKENDFKAFFIESVCDDPTVVASNIMEVKISSPDYKDCNS

AEAMDDFMKRINCYEASYQPLDPDKCDRDLSFIKVIDVGRRFLVNRVQDHIQSRIVYYLM

NIHVQPRTIYLCRHGENEYNVQGKIGGDSGLSSRGKKFANALSKFVEEQNLKDLRVWTSQ

LKSTIQTAEALRLPYEQWKALNEIDAGVCEELTYEEIRDTYPEEYALREQDKYYYRYPTG

ESYQDLVQRLEPVIMELERQENVLVICHQAVLRCLLAYFLDKSAEEMPYLKCPLHTVLKL

TPVAYGCRVESIYLNVESVSTHRERSEAVKIQHFASVVRPSSYTELDFLSVESAKQDAKK

GPNPLMRRNSVTPLASPEPTKKPRINSFEEHVASTSAALPSCLPPEVPTQLPGQNMRSPR

SGAESSQKH

$ ./seqkit grep ipi.RAT.fasta -p IPI00197241

>IPI00197241

MPLELTQSRVQKIWVPVDHRPSLPRSCGPKLTNSPTVIVMVGLPARGKTYISKKLTRYLN

WIGVPTKVFNVGEYRREAVKQYSSYNFFRPDNEEAMRVRKQCALAALRDVKSYLTKEGGQ

IAVFDATNTTRERRHMILHFAKENDFKAFFIESVCDDPTVVASNIMEVKISSPDYKDCNS

AEAMDDFMKRINCYEASYQPLDPDKCDRDLSFIKVIDVGRRFLVNRVQDHIQSRIVYYLM

NIHVQPRTIYLCRHGENEYNVQGKIGGDSGLSSRGKKFANALSKFVEEQNLKDLRVWTSQ

LKSTIQTAEALRLPYEQWKALNEIDAGVCEELTYEEIRDTYPEEYALREQDKYYYRYPTG

ESYQDLVQRLEPVIMELERQENVLVICHQAVLRCLLAYFLDKSAEEMPYLKCPLHTVLKL

TPVAYGCRVESIYLNVESVSTHRERSEDAKKGPNPLMRRNSVTPLASPEPTKKPRINSFE

EHVASTSAALPSCLPPEVPTQLPGQPLLGKACLRTVCHIFSKFSPY

引用：seqkit

Webサイト：https://bioinf.shenwei.me/seqkit/

論文：W Shen, S Le, Y Li\*, F Hu\*. SeqKit: a cross-platform and ultrafast toolkit for FASTA/Q file manipulation. PLOS ONE. doi:10.1371/journal.pone.0163962.

1. phosphorme\_ec\_ipi\_residue.txtに含まれるタンパク質の配列をipi.RAT.fastaから抜き出し、phosphorme\_ipi.fastaを作成しなさい。IPI IDはphosphorme\_ec\_ipi\_residue.txtの2列目から抜き出し、重複を除いた上でfor文に用いるとよい。

$ for aIPI in $(この空欄を埋める phosphorme\_ec\_ipi\_residue.txt | sort | uniq) ; do ./seqkit grep ipi.RAT.fasta -p この空欄を埋める; done > phosphorme\_ipi.fasta　←わからない人はこの部分をコピペ

1. (1)の結果出力されるファイル、phosphorme\_ipi.fastaを同名の模範解答ファイルと比較しなさい。diffコマンドを使うこと。

Q3.責任代謝酵素についてこの情報を取り出し、resnum\_sample.txtと同様の

[IPI][tab][リン酸化部位]

という形式で保存したい。

1. phosphorme\_ec\_ipi\_residue.txtからリン酸化された責任代謝酵素のリン酸化部位（残基番号）を抜き出したファイルphosphorme\_ipi\_residue.txtを作成せよ。（ヒント：cutで2列目と3列目を抜き出し、重複削除）
2. phosphorme\_ipi\_residue.txtを同名の模範解答ファイルと比較しなさい。diffコマンドを使うこと。

Q4. リン酸化された責任代謝酵素のFASTA形式ファイルphosphorme\_ipi.fasta (Q2で作成)およびリン酸化部位ファイルphosphorme\_ipi\_residue.txt (Q3で作成)をNetPhorestに入力し、キナーゼの予測結果をnetworkin\_predictions.tsvに保存しなさい。出力に時間がかかる場合は、Q2(4).Q3(2)で正解を確認した上で、Day7\_AnswerFiles/Step\_ivフォルダ内のnetworkin\_predictions.tsvを以下の問で用いても良い。出力を待つ必要はない、休憩していてもよい。

Q5. （不安であれば）networkin\_predictions.tsvの内容が、模範解答ファイル‘Day7\_AnswerFiles/Step\_iv/networkin\_predictions.tsv’と一致しているか確認しなさい。

Q6. networkin\_predictions.tsv は1,6列目にそれぞれ基質のIPI、責任キナーゼ名が含まれている。これらの列のみ抽出して重複を取り除き、phosphorme\_ipi\_kinase.txtに保存し、Excel等で中身を確認せよ。１行目は見出しなので、除いておくこと。sortコマンドにおいては、辞書順にするために-fオプションを用いること。

Q7. 最後に、phosphorme\_ipi\_ec.txtとphosphorme\_ipi\_kinase.txtを統合し、代謝酵素とキナーゼの対応関係phosphorme\_ec\_kinase.txtを作成し、less等で確認せよ。join(後述)を用いて、IPI\_ID, EC番号, キナーゼの表を作成後、2列目と3列目を抽出し（\*）、重複を削除すれば良い。ただし、cutコマンドにおいては、区切り文字を” ”にする必要があるため（joinの仕様）、-d “ ”を付加すること。またsortコマンドにおいては、辞書順にするために-fオプションを用いること。

\*2列目と3列目の間は”\t”タブ文字ではなく、” ”空白を用いてください（joinの仕様）。そのため、awkを用いる場合は$2と$3の間は” ”を用いる必要がある。

コマンドjoinの動作　～join fileA fileBを例に～

file Aの中身 file Bの中身 $ join fileA fileB

A 1 A 11 A　1　11

B 2 B 12 B　2　12

C 3 C 13 C　3　13

C 4 C 14 C　3　14

 C　4　13

 C　4　14

 $ join phosphorme\_ipi\_ec.txt phosphorme\_ipi\_kinase.txt | cut -d " " この空欄を埋める | sort -f | uniq > phosphorme\_ec\_kinase.txt　←わからない人はこの部分をコピペ

Q8. Q7で作成したphosphorme\_ec\_kinase.txtの内容が、模範解答ファイル‘Day7\_AnswerFiles/Step\_iv/ phosphorme\_ec\_kinase.txt’と一致しているか確認しなさい。

* まとめ：経路抜き出し

ここまでに、以下の経路ファイルが作成された

・metab\_rme.txt： 変動代謝物　　～責任代謝酵素

 cpd:C00021 ec:1.16.1.8

 cpd:C00021 ec:2.1.1.1

 cpd:C00021 ec:2.1.1.10

 cpd:C00021 ec:2.1.1.100

 … …

・phosphorme\_ec\_kinase.txt 代謝酵素　～ キナーゼ

 ec:1.1.1.95 AuroraA

 ec:1.1.1.95 CLK\_group

 ec:1.1.1.95 DMPK\_group

 ec:1.1.1.95 PKC\_group

 ec:1.1.1.95 RCK\_group

 ec:1.1.1.95 RSK\_group

 … …

最後にこれらの情報を用いて、ネットワークの可視化を試みる。

* トランスオミクスStep (v): シグナル伝達経路と代謝経路を視覚化する

インスリンシグナル伝達経路と中心炭素代謝経路の間をつないで立体的に視覚化する。

Q1. まず、phosphorme\_ec\_kinase.txtのうち、インスリンシグナル伝達経路と中心炭素代謝経路の間をつなぐリン酸化の情報のみを取り出す。

1. NetPhorestで推定したキナーゼのうち、KEGGのインスリンシグナル伝達経路に存在するキナーゼのIDはPKA\_group, PKB\_group, PKC\_group, p70S6K\_group, GSK3\_group, MAPK3\_MAPK1\_MAPK7\_NLK\_groupの6つである。これら6つを含む行のみをphosphorme\_ec\_kinase.txtから抜き出し、phosphorme\_ec\_inskinase.txt という名前のファイルに保存しなさい。

（ヒント：「PKA\_groupまたはPKB\_group」を検索したいときは、 "PKA\_group\|PKB\_group" のように書く、UNIX課題２参照。）

1. （不安であれば）phosphorme\_ec\_inskinase.txtを同名の模範解答ファイルと比較しなさい。diffコマンドを使うこと。
2. 中心炭素代謝の酵素リストをつくる。そのために再びKEGG API ”Link”を用いる。

$ curl http://rest.kegg.jp/link/ec/[生化学パスウェイのID]

と入力すると、当該生化学パスウェイに含まれる全酵素のEC numberが表示される。解糖系(map00010)、TCAサイクル(map00020)、ペントースリン酸経路(map00030)に含まれる全酵素のEC number一覧をそれぞれglycolysis\_ec.txt、tca\_ec.txt、ppp\_ec.txtというファイル名で保存しなさい。１列目はmapIDであることに注意せよ。

1. glycolysis\_ec.txt、tca\_ec.txt、ppp\_ec.txtに含まれる全EC numberを重複なく統合し、中心炭素代謝経路のEC number一覧ccm\_ec.txt というファイル名で保存しなさい。（ヒント：ファイルの統合にはcatを用いる。ex. cat fileA fileB fileC > file ABC）
2. （不安であれば）ccm\_ec.txtを同名の模範解答ファイルと比較しなさい。diffコマンドを使うこと。
3. EC numberリストccm\_ec.txtとphophorme\_ec\_inskinase.txtを用いて、「インスリンシグナル伝達経路の6つのキナーゼにリン酸化された中心炭素代謝の責任代謝酵素」phosphorme\_ccmec\_inskinase.txtを作成せよ。（ヒント：for文とgrep -w。Step ( iii)Q4参照）

$ for aEC in $(cat この空欄を埋める); do grep -w この空欄を埋める; done > phosphorme\_ccmec\_inskinase.txt

1. phosphorme\_ccmec\_inskinase.txtの中身を読み、最終的に得られた中心炭素代謝系酵素とその責任キナーゼが次のペアであることを確認しなさい。

責任キナーゼ 酵素番号(酵素名)

GSK3\_group 5.4.2.2 (Pgm1)

p70S6K\_group 2.3.3.8 (Acly)

PKA\_group 1.2.1.3 (Aldh)

PKB\_group 2.7.1.11 (Pfkl)

PKC\_group

Q2. VANTED with HIVEを使って、インスリンシグナル伝達経路と中心炭素代謝経路の間を立体的に視覚化する。

0. ソフトウェアの準備のために、以下のコマンドを実行せよ。特に意味は考えないくて良い（JAVA 3D関連のライブラリを導入している。）

$ ./makeJava3Denv.sh

1. HIVE with VANTEDをデスクトップにダウンロードし、解凍しなさい。以下のURLより、[Download] 🡪 ”1. latest release v1.3” の下の [Zip file] の順にクリックする。

https://immersive-analytics.infotech.monash.edu/hive/

1. デスクトップで解凍したHIVE\_with\_VANTED.jarをダブルクリックしてHIVE with VANTED を起動しなさい。“HIVE\_with\_VANTED.jar”は、開発元が未確認のため開けません。」と表示されて開けない場合はcontrolキーを押しながらクリック→開く。（JDKがインストールされていない場合は開発元であるOracle社のサイトが開き、インストールを促されるのでその通りにする）
2. “File” 🡪 “Open” より、中心炭素代謝とインスリンシグナル経路を統合したinssigpath\_ccm\_gmlを開き、全体を概観せよ。
3. insulin\_trans\_omic.gmlという名前で保存しなさい
4. insulin\_trans\_omic.gmlの内、右側のインスリンシグナル伝達ネットワークを範囲選択し、“Cluster”🡪”Enter Cluster ID”🡪”Enter and set cluster ID”を選択し、Cluster IDに’ signaling’と入力しなさい。同様に左側の中心炭素代謝ネットワークを範囲選択し、Cluster IDに’ metabolism’と入力しなさい
5. Q1 (7) にて同定した責任キナーゼと代謝酵素の間のネットワーク（下図）に基づき、insulin\_trans\_omic.gmlの中の対応するノード同士をエッジで結びなさい（ノード同士をエッジで結ぶ方法はスライド参照）。また、下の注意書きをよく読むこと。

（面倒ならば一部のみでもよい）

責任キナーゼ 代謝酵素

GSK3\_group (GSK3b) Pgm1

p70S6K\_group (Rps6kb2) Acly

PKA\_group (Prkar2b) Aldh3a2 (無視してよい)

PKB\_group (Akt1) Pfkl

PKC\_group (Prkcz)

注意書き：

* 代謝酵素のうち、Aldh3a2はマップにないので無視してよい
* insulin\_trans\_omic.gmlのネットワークにおいて、代謝酵素の名前は上図そのままであるが、キナーゼの名前は(~~)のようになっている。
1. insulin\_trans\_omic.gmlを保存しなさい。
2. “File”🡪”New View”🡪”3D Graph View”で3D表示ウィンドウが開く。左ボタンが回転、右ボタンが並進、中ボタンのダイヤル回転がズームである。トラックパッドの場合、１本指でクリックしたまま１本指で swipe すると回転、２本指でクリックし、１本指だけクリックしたまま swipe すると並進、２本指で上から下／下から上に swipe するとズームイン／ズームアウトである。
3. 2D表示ウィンドウにて、“Edit”🡪”Selection”🡪”Select Cluster”でmetabolismまたはsignalingを選ぶと、代謝層またはシグナル層を選択できる。一方の層を選択し、両者が上下にくるように位置を調整しなさい。2D表示ウィンドウで位置を調整してファイルを保存すると、3D表示ウィンドウに反映される。
4. 代謝層とシグナル層の間隔を変えたいときは次のようにする。(1) “Edit”🡪”Selection”🡪”Select Cluster”でmetabolismまたはsignalingを選択 (2)右側の”Network”タブ🡪”Node”🡪”Node Attributes”🡪”Position”の3つある欄のうち、一番右の欄がZ座標なので、ここに適当な値を入れる。Z座標の初期設定は、metabolismが-800、signalingが0である。