Mutation2transcription

白石友一

東京大学　医科学研究所　助教

使用機器：MacBook Pro

1. スクリプト、テストデータ取得

まず適切なディレクトリに移動した後，

git clone <https://github.com/friend1ws/saibo-kogaku-ngs.git>

1. 必要package取得

pip install pytabix

また，tabixを取得する

（例）

wget <http://sourceforge.net/projects/samtools/files/tabix/tabix-0.2.6.tar.bz2>

tar zxvf tabix-0.2.6.tar.bz2

cd tabix-0.2.6

make

その後，必要に応じてpathに登録

1. ゲノム変異と転写異常の関係図のプロット

用意するもの

* sample\_list.txt: サンプル名リスト
* gene\_sig.txt: 表示する遺伝子リスト
* mut2trans.txt: ゲノム変異と転写物異常の関係（一行目: サンプル名、二行目：遺伝子、三行目：ゲノム、転写異常の種類）

cd saibo-kogaku-ngs/script

Rscript visMut2sp.R ../plot\_example\_data/mut2trans.txt ../plot\_example\_data/sample\_list.txt ../plot\_example\_data/gene\_list.txt ../plot\_example\_data/output.jpg

1. ゲノム異常と転写異常の関係を示すファイルの作成例

以下では，そもそもゲノム変異と転写異常の関係を示すデータを作成する例を紹介する．ここでは， splicing異常を引き起こすpoint mutation, indelを抽出し，上記のvisMut2sp.Rにあうフォーマットに直すコマンド例を紹介する．

用意するもの

* ${sample}.mutaiton.vcf.gz, ${sample}.mutaiton.vcf.gz,tbi: (gz圧縮されたvcfファイル、tabixインデックス，圧縮されていない場合はbgzipで圧縮して、tabixでインデックスを付与bgzip ${sample}.mutaiton.vcf > ${sample}.mutaiton.vcf.gz; tabix ${sample}.mutaiton.vcf.gz)
* ${sample}.junction.txt: mapsplice2などで出力したjunctionファイル（bedファイル形式を圧縮したもの、normal ファイルなどがあれば，normalに見られるjunctionパターンを除いておくことが望ましい）

まず，遺伝子，エキソン領域のアノテーションファイルを作成

cd saibo-kogaku-ngs/data/db

sh prepGeneInfo.sh

これにより，

refGene.bed.gz, refGene.bed.gz.tbi, refExon.bed.gz, refExon.bed.gz.tbi

の４ファイルが生成される．

以下junction.txtファイルがmapsplice2の出力によるものだと仮定した場合のコマンド例

mkdir -p ./result

echo -n > ../result/RK.mut2trans.txt

while read sample;

do

# support read < 5はフィルターする

python simpleFilt.py ../data/mapsplice/${sample}C.junctions.txt.gz 5 > ../result/${sample}.junctions.filt.txt

# 各々のsplicing junctionのアノテーションファイルを作成

python annotSplicing.py ../result/${sample}.junctions.filt.txt ../data/db/refGene.bed.gz ../data/db/refExon.bed.gz > ../result/${sample}.annot.txt

# mutation dataとsplicing junctionを比較し，あるsplicing motifに影響を及ぼしている遺伝子変異とそれに対応するsplicing junctionを抽出

python getSplicingMut.py ../result/${sample}.annot.txt ../data/vcf/${sample}.mutation.vcf.gz ../data/db/refExon.bed.gz > ../result/${sample}.mut2sp.txt

# 元々のmutationのデータと上記の結果を統合

python mergeMut.py ../data/vcf/${sample}.mutation.vcf.gz ../result/${sample}.mut2sp.txt ../data/db/refExon.bed.gz ${sample} >> ../result/RK.mut2trans.txt

done < ../data/sample\_list.txt

これにより，../result/RK.mut2trans.txt に結果が出力される．また，こちらをまとめたスクリプトがscript/runall.shにまとめられている．

この後，例えば以下のように興味のある遺伝子リストを作成し

echo -e "TP53\nHNF4A\nBRD7\n\nRPS6KA3\nARID2\nALB" > ../result/RK.gene.txt

visMut2sp.Rスクリプトで図を出力

Rscript visMut2sp.R ../result/RK.mut2trans.txt ../data/sample\_list.txt ../result/RK.gene.txt ../result/plot.jpg