업무 인수 인계서(상세)

근무부서	소프트웨어 개발팀	사번	-
성명	김민정	직급	사원
업무인수인계 내용			

1. SeqFF(Enet) 모델 훈련

✓ 관련 문서 위치 : /업무인수인계_김민정/SeqFF(Enet)/

✓ code 위치(server)

- sample 선정 및 준비 과정 code : (albatross) **/BiO/Alba/Enet/code/**
- 훈련 과정 code : (albatross) **/BiO/Alba/Enet/Training/code/**

✓ 훈련 과정

1) DB 출고 문서에서 샘플 선정 및 목록 작성

- DB 출고 문서 위치 : **/SeqFF(Enet)/출고 DB 복사본/**
- 현재 출고 DB는 운영팀 공유폴더에서 복사한 201608~201802 문서와 노트북에 저장되어 있던 201603~05 문서로 이루어져 있습니다.
- 엑셀 필터 기능을 통해 XY, 단태아인 경우만을 선정합니다. 선정된 샘플의 구성은 다음과 같습니다. 시트 간의 참조와 수식을 주의하세요.

< 선정 샘플 정보 >

GenomecareID	Result	RR	UR	Y-FF	Fasta name

* **사용 시트** : 제노맘 분석 종합(결과, XY, FF, UR 정보), Global 분석 결과(fasta name, RR 정보)
* RR or UR, Y-FF, GenomecareID 정보는 반드시 필요합니다. Fasta name이 없는 경우에는 MedicalLAB ID도 함께 필요합니다. 그 밖의 정보는 필요하시다면 추가하시면 됩니다.
* GenomecareID와 Fasta name을 제외한 중간 값들은 위치 변경 가능합니다.
* scp code는 fasta name으로 matching 여부를 결정하므로 Date_name.list file의 **마지막 열은 반드시 Fasta name**이 있어야 합니다.
* fasta name = GenomecareID+MedicalLAB ID+Bacorde
* ex. GC1801N0004_20171230-29303_IonXpress_081
* 2017 상반기, 2016년의 경우 fasta name이 출고 문서에 없습니다. 이 경우 **genomecareID와 MedicalLAB ID**를 이용하여 matching 여부를 결정하므로

MedicalLAB ID도 함께 필요합니다.

2) Backup server에서 목록에 맞는 fastq file을 albatross server로 복사

- scp 사용

- 명령어 : **copy_fastq.py Date_name.list**

Input File : **Date_name.list**(GenomecareID/results/RR/UR/Y-FF/Fastq)

* ex. /BiO/Alba/Enet/201701/에서 실행

../code/copy_fastq.py 201701_match_fastq.list

* Date는 반드시 '201701' 과 같은 형태로 작성되어야 합니다. Ex) 201701_match_fastq.list

* 'Date_' 이후의 이름은 변경 가능합니다.

* 각 열의 구분자는 tab(\t) 입니다.

* ex. GC1709C4616 redraw 5841484 3030193 1.537704777 GC1709C4616_20170911-311217_IonXpress_069

Output file : name.Fastq, Date_except.log

* Date_except.log : 선정 샘플 목록에 존재하지만, backup 폴더에서 찾지 못한 fastq file

* 복사해온 fastq file은 **/BiO/Alba/Enet/Date(ex. 201701)/** 폴더에 생성됩니다.

* Backup 서버에서 fastq 파일을 가져올 때, 2016년, 2017년, 2018년의 파일 위치가 다릅니다. Copy-fastq.py 에서 년도에 따라 location을 지정해주긴 하지만 현 시점을 기준으로 경로를 설정하였으므로 2018년의 경우 backup 서버에서 경로의 변화가 있을 수 있습니다. Copy-fastq.py를 실행하기 전, Backup 서버에서 **2018년 fastq 파일의 경로가 현재 설정 경로와 동일한지 확인**할 필요가 있습니다.

- fasta name이 없는 경우 : copy_fastq-ID.py Date_name.list

Input File : Date_name.list(GenomecareID\tMedicalLabID\t.....)

1열, 2열은 반드시 GenomecareID\tMedicalLabID로 구성되어야 합니다.

3) Duplication 제거 및 정렬 작업을 마친 후 최종 sam file 생성

- 명령어 : **enet_analysis.sh flistN**

Input File : flist(fasta file list)

* 데이터 분석을 진행하고자 하는 fastq list(ex. ls *.Fastq > flist).

* Fastq 파일을 N개씩 나누어 qsub 분산 처리 할 수 있습니다(list file 이름은 변경 가능합니다). 50개 권장(4~6시간 소요)

* ex) sed –n '1,50p' flist > flist1; sed –n '51,100p' flist > flist2; ...

* ex) qsub … enet_analysis.sh flist1; qsub … enet_analysis.sh flist2; …

* 기타 qsub 사용 예시는 '/업무인수인계_김민정/SeqFF(Enet)/Code/**qsub명령어 예시.txt**' 에서 확인하실 수 있습니다.

Output file : name.Fastq.sam.bam.sort.bam.rmdup.bam.sam

4) Bininfo file 생성

- ls *. Fastq.sam.bam.sort.bam.rmdup.bam.sam > blist

- 명령어 : **bininfo.sh blist**

Input File : blist(rmdup-sort된 sam 파일의 list. 이름 변경 가능.)

* bininfo를 생성하고자 하는 파일의 list를 input 값으로 넣어줍니다.

* 생성된 bininfo file은 **/BiO/Alba/2017_backup/SeqFF/ENET/bininfo/test_data** 에 저장됩니다.

* 이전까지 생성한 bininfo file은 /BiO/Alba/2017_backup/SeqFF/ENET/bininfo 에 저장되어 있습니다.

Output File : GenomcereID_bininfo

5) 최종적으로 생성된 결과 목록 구성

- 실제로 생성된 bininfo 와 샘플 정보를 matching 하는 과정입니다.

- N차에서 생성하고자 했던 bininfo의 생성이 모두 종료된 후 진행해주세요(ex. 5st의 경우 201601, 201602, 201610, 201801, 201802의 bininfo 생성 작업이 종료 된 후 진행해주시면 됩니다).

- 위치 : **/BiO/Alba/Enet/analysis/**

< 샘플 목록 구성 >

Bininfo	GenomecareID	Results	RR	UR	Y-FF

- 선정 샘플 정보에서 GenomecareID/results/RR/UR/Y-FF 정보 list를 가져옵니다. 구분자는 tab(ex. /BiO/Alba/Enet/analysis/info.4st.list).

- N차에서 생성한 bininfo list(test_data에 저장되어 있는)를 가져옵니다 (ls *_bininfo > infolist)

- 명령어 : **match_2.py info.4st.list infolist**

Input File : info.N.list(ID/results/RR/UR/Y-FF), infolist(ID_bininfo)

Output File : **final.list**(bininfo\tgenomecareID\tresults\tRR\tUR\tY-FF)

* bininfo file이 존재하나, info.list 파일에 matching되는 샘플이 없는 경우 except.list에 출력됩니다(_RD, _rs 등과 같은 ID 문제일 수 있습니다).

* final.list는 N차에서 최종적으로 생성된 bininfo 목록 파일입니다. 위 목록을 이전 회차들에서 생성한 bininfo 목록과 합친 후 최종적으로 훈련/검증에 사용할 샘플을 선정합니다(/업무인수인계_김민정/SeqFF(Enet)/Data/bininfo데이터목록(전체) 참조).

- N차에서 생성한 bininfo 의 sample 정보와 matching이 끝났고, 문제가 없다면 **/BiO/Alba/2017_backup/SeqFF/ENET/bininfo/test_data/에 존재하는 bininfo 파일들을 /BiO/Alba/2017_backup/SeqFF/ENET/bininfo/ 폴더로** 옮겨주셔야 합니다. 훈련 시 bininfo 파일을 불러올 때 bininfo/ 폴더에서 불러오기 때문입니다.

6) 훈련과 검증에 사용할 샘플 목록 작성

- 전체 샘플 중에서 훈련/검증에 사용할 샘플을 선정합니다.
- RR이 400만개 이상이면서 Y-FF이 5% 이하인 경우를 우선적으로 선정 -> 그 후 RR이 높은 순으로 선정합니다.
- 선정된 샘플의 bininfo name, Y-FF만을 추출하여 list로 만듭니다.

< **bininfo_N**(ex. bininfo_4000) >

Bininfo	Y-FF

- ex. C1706E2953_bininfo 10.08413332
- input data 위치 : /BiO/Alba/Enet/Training/input-data/
- 생성된 list를 input-data 폴더로 이동

*** Training 준비 완료 ***

7) Fold 별로 나누어 모델 훈련 및 검증

- 명령어 : **Enet.py (or Enet.sh)**

Input file : bininfo_N(GenomecareID_bininfo\tY-FF)

Output file : foldN.csv, foldN_enet.csv

- code 위치 : /BiO/Alba/Enet/Training/code/Enet.py
- 관련 code : Enet.py(Enet.sh), glmnet_v3.R, cal_enet_v2.R
- 실행 전 확인 사항 :

* /input-data/ 에 반드시 bininfo_N 파일이 존재하여야 합니다.

* **input file 이름 설정** 필요(36 line, file name 변경). Input file의 이름을 code에 명시해 주세요.

* **f_num은 총 fold의 개수**이며, **f_range는 Test sample의 개수**입니다(38 line의 변수). 원하시는 값으로 변경하셔야 합니다(f_num의 default 값은 10 입니다).

* ex. f_num=10; f_range=500 -> 10 fold, test 500, 나머지 training

- code 설명 :

*** Enet.py** : 훈련/검증 샘플 목록 생성 및 fold 설정, glmnet_v3.R, cal_enet_v2.R 호출(input : bininfo_N)

*** glmnet_v3.R** : 훈련을 통해 parameter 생성(input : Training, bininfo / output : result_v2.csv, foldN.csv)

*** cal_enet_v2.R** : 훈련시킨 모델을 사용하여 FF 추정(input : Test, bininfo, RData, result_v2.csv / output : foldN_enet.csv)

* 훈련 순서 : **Enet.py** 실행 -> fold1 Training, Test 파일 생성 -> **glmnet_v3.R** 호출 -> Training 파일을 읽어 들인 후 훈련 -> parameter 생성(result_v2.csv) -> **cal_enet_v2.R** 호출 -> Test 파일, result_v2.csv 파일을 read -> Test 샘플 FF 추정 -> 결과 파일 생성(fold1_enet.csv) -> fold2 Training, Test 파일 생성 -> 반복

* glmnet_v3.R을 통해 생성된 모델 계수는 foldN.csv(fold1.csv, fold2.csv, ...)로 따로 저장됩니다. result_v2.csv는 fold 진행 시마다 매번 각 fold의 모델 계수가 쓰여지므로, 각 fold에서 사용된 모델 계수를 확인하고 싶으실 때는 foldN.csv 파일을 보시면 됩니다.

* 이미 만들어진 모델을 이용하여 Test 샘플의 FF만을 추정하고자 할 때는 사용하고자 하는 모델(result_v2.csv)과 Test 샘플 목록(Test)을 fold_result 폴더에 위치시킨 후 cal_enet_v2.R을 실행시키면 됩니다. 단, 이름은 반드시 'result_v2.csv', 'Test' 형식이어야만 합니다.

8) 결과 분석

- 결과 파일 위치 : /BiO/Alba/Enet/Training**/fold_result/**

- **foldN_enet.csv** : FF에 대한 최종 결과 파일은 **csv(쉼표로 구분)**로 저장됩니다. 구성은 아래와 같습니다.

index	final_ID	final_enet_ori	**final_enet**	y

* final_enet_ori : original 모델로 추정한 FF

* **final_enet : 새로 훈련시킨 모델로 추정한 FF**

* y : y 기반 FF

* 윈도우로 결과 파일들을 옮긴 후 엑셀에서 분석을 진행하시면 됩니다.

* /업무인수인계_김민정/SeqFF(Enet)/Data/4000/model, /업무인수인계_김민정/SeqFF(Enet)/Data/4000/3600개 훈련, 400개 검증 최종 결과.xlsx 참조

9) 전체 명령어 정리

'/업무인수인계_김민정/SeqFF(Enet)/Code/전체 명령어 정리.txt' 참조

* 훈련 과정에서 사용하는 명령어들을 정리해 놓았습니다.

✓ Enet 관련 문서

- Training data 관련 문서 위치 : /SeqFF(Enet)/Data/
- /SeqFF(Enet)/Data/샘플개수/**N훈련, N검증 최종 결과.xlsx** : 최종 결과 분석 문서

(ex. /SeqFF(Enet)/Data/1000/900개 훈련, 100개 검증 최종결과.xlsx)

- /SeqFF(Enet)/Data/샘플개수/N 데이터.xlsx : N개 훈련에 사용한 샘플

(ex. 2000 데이터.xlsx)

- foldN.csv, foldN_enet.csv 파일 위치 :

/SeqFF(Enet)/Data/샘플개수/model/

- 출고 DB 위치 : /SeqFF(Enet)/출고 DB 복사본/
- 샘플 정보 위치 : /업무인수인계_김민정/SeqFF(Enet)/Data/샘플정보/
- 1, 2, 3, 4 차까지 생성한 bininfo 목록 :

/업무인수인계_김민정/SeqFF(Enet)/Data/**bininfo데이터목록(전체).xlsx**

- code 위치 : /SeqFF(Enet)/Code/
- seqff_sample_info.xlsx : 월별 사용 샘플 개수 정보

✓ N차 생성 목록

	1차	2차	3차	4차	5차(예상)
생성한 bininfo	1044	1108	1298	993	966

1) 1차~4차에서 생성한 bininfo 총 개수 : **4443**개
2) 5st에 사용될 샘플 : **201601,201602,201610,201801,201802**
3) 5st 진행 상태

- 201601, 201602 : albatross 서버에 fastq 복사 완료
- 201610 : bininfo 까지 생성 완료
- 201801, 201802 : 선정 샘플 목록 생성 완료

✓ 기타 참고 사항

- **201601, 201602** 샘플의 경우 입, 출고 DB 파일이 없습니다. Storage에 존

재하는 201601, 201602에 해당하는 fastq 파일을 albatross server의 /BiO/Alba/Enet/201601-02/ 에 복사해 두었습니다. Y-FF를 측정하여 남 태아인 샘플만을 사용하시면 될 것 같습니다(Y-FF와 RR을 계산하는 과정이 추가로 필요, copy_fsatq 과정은 건너뛰시면 됩니다)

- 201801, 201802 의 경우 출고 DB에서 샘플을 선정하여 선정 샘플 목록을 생성해 놓았습니다. /BiO/Alba/Enet/201801, /BiO/Alba/Enet/201802 폴더에 있는 목록을 이용하여 2번 과정(copy fastq)부터 진행하시면 됩니다.

- 201610 의 경우 bininfo 까지 생성 완료하였습니다 (/BiO/Alba/2017_backup/SeqFF/ENET/bininfo/test_data/ 위치). 2018년 1~2월 샘플과 201601, 201602 샘플의 bininfo를 생성하셔서 5000개 훈련을 진행하시면 됩니다(5st : 201601,201602,201610(완료),201801,201802).

- 훈련 전 bininfo_N 파일을 shuffle 하고 싶으시면 /BiO/Alba/Enet/Training/code/shuffle.py 를 사용하시면 됩니다. 코드 사용 시 input file의 마지막 line에 '\n' 문자가 포함되어 있지 않을 시, 셔플 후에 두 개의 샘플이 한 line에 출력될 수 있습니다(ex. 3000개 셔플 후 2999개 line write). 이는 개행문자가 없는 샘플이 중간에 위치하면서 샘플 두 개가 한 줄에 출력되면서 생기는 문제입니다. Input file의 마지막 줄에 개행 문자를 추가해 주시거나, 셔플 후 개행 문자가 미포함 되었던 샘플을 찾아서 수정해 주시면 됩니다.

- 용량 문제로 생성하였던 sam file은 보관하고 있지 않습니다. 대신 사용했던 fastq file과 bam file을 각 Date 폴더의 fastq, samfile 폴더(ex. /BiO/Alba/Enet/201712/samfile/)에 보관하고 있습니다. Sam file이 필요하시면 bam file에서 생성하시면 됩니다.

- Backup server는 이전에 사용하던 genomecare 서버입니다. Backup 서버의 fastq 위치는 다음과 같습니다.

2016 : /Backup/NIPT/2016/

2017 : /Backup/GENOMOM/FASTQ/2017/

2018 : /Backup/GENOMOM/FASTQ/

2. 연구마을 과제

✓ 관련 code 상세 설명

- tensorflow 설치 관련 : 연구노트#4 참조

- tensorflow 가상환경 실행 및 종료

source activate tensorflow : tensorflow 가상환경 실행

source deactivate : 종료

- tensorflow 가상환경에 설치된 라이브러리 : sklearn, scipy, matplotlib, numpy, tensorflow, tensorboard

- Code 설명 :

* back.py : backpropagation 사용 code.

* CNN.py : CNN 예제 코드

* MLR.py : 다중선형회귀 사용 코드

- input data 생성 명령어 : input_data_v2.py training_4000

위치 : /BiO/Alba/FF_Deep/data/create-list/

Input file : training_4000(이름 변경 가능합니다. genomecareID\tY-FF\n)

* bininfo_4000을 사용하셔도 무관합니다.

Output file : training_4000_input(Y-ff,X1,X2,X3,X4,...,Xn)

* 구분자는 쉼표(,) 입니다.

* 생성 파일의 Y-FF 값에 ^M 문자가 포함되어 있을 경우에는 제거해주셔야 합니다.

File open -> : %s/^M//g -> enter -> ^M 제거

^M 을 입력 하실 때는 반드시 ctrl+V, ctrl+M 으로 입력하셔야 합니다.

* chr13, 18, 21의 경우 bincounts를 0으로 설정하였습니다.

* chrX, chrY는 사용하지 않습니다(총 57634 bins)

* output file 생성 위치 : /BiO/Alba/FF_Deep/data/

- /Saver/ 폴더는 훈련한 모델 정보를 저장하는 폴더입니다.

- /log_N/ 폴더는 log 정보를 저장하는 폴더입니다.

* log는 tensorboard에 사용됩니다. 한 폴더에 하나의 log가 존재해야 tensorboard에서 확인이 가능합니다. Log를 한번 쓴 후에는 다른 log 폴더를 사용하시거나, 직전의 log 기록을 지워주세요. Log 경로 수정은 코드 내의 tf.summary.FileWriter()에서 가능합니다.

* log에 기록할 변수 설정이 가능합니다. 시각적으로 확인하고자 하는 변수를 코드 내에서 지정해 주세요(ex. tf.summary.histogram(), tf.summary.scalar() 함수 사용).

- tensorboard 사용법 : tensorboard --logdir=./logs_1

* 원하는 log 파일이 위치한 경로를 입력하시면 됩니다.

* log의 이름을 지정하여 동시에 여러 개의 log를 tensorboard에서 확인 할 수도 있

습니다(ex. tensorboard --logdir logname1:.\logs_1, logname2:.\logs_3).

✓ 한 샘플에 여러 개의 FF 추정 실험(Deep Leaning+Enet)

- 관련 code 위치 : /BiO/Alba/Enet/Training/code/
- 관련 code : Enet-DL.py, glmnet_v3-DL.R, cal_enet_v2-DL.R
- 명령어 : Enet-DL.py

* 기본적인 사용법은 Enet.py와 동일합니다.
* bininfo_3000 파일을 읽어 들인 후, 3000개의 sample을 무작위로 shuffle 합니다. 그 후, 순서대로 2000개의 샘플을 training_2000으로 저장하고 나머지 1000개의 샘플을 Test로 저장합니다.
* fold별로 반복적으로 훈련을 시작합니다. 훈련을 위한 샘플은 training_2000에서 fold 마다 1500개를 무작위로 선정합니다. Test 샘플은 동일합니다.
* f_num 변수를 조정하여 fold의 개수를 변경할 수 있습니다(default=20).

- 결과 파일 저장 위치 : /BiO/Alba/Enet/Training/DL/
- 결과 파일은 Enet.py의 결과와 동일한 형식입니다(foldN_enet.csv, foldN.csv).
- combine.sh : 실행하시면 fold 20개에 대한 파라미터 별 FF 결과를 하나의 문서(Enet-1000.list)로 통합해줍니다. 컬럼 구성은 다음과 같습니다.

(구성) GenomecareID,Y-FF,para1 FF,para2 FF,para3 FF,.....,para20 FF

* 구분자 쉼표(,)

- 1000개에 대한 20 fold 모델 parameter 문서 저장 위치:
/업무인수인계_김민정/연구마을/SeqFF(Enet)/

* foldN.csv : 각 fold 별 모델 계수
* foldN_enet.csv : foldN.csv 모델 계수로 추정한 FF
* Enet-1000.list : 1000개 샘플에 대해 20개의 모델 계수로 추정한 FF 정보
* Test : 사용한 1000개 검증 샘플
* training_2000 : 사용한 2000개의 훈련 샘플. 이 중에서 1500개를 무작위 선정하여 모델을 훈련시켰습니다.

✓ 연구노트, 기술지도요청서, 회의록 관련 서류

1) 연구노트

- 위치 : /업무인수인계_김민정/연구마을/연구노트/
- 20180112 ~ 20180307 까지 작성되었습니다.

- 2주씩 묶어서 총 5개의 PowerPoint 문서로 이루어져 있습니다.

2) 회의록 & 기술지도요청서

- 위치 : /업무인수인계_김민정/연구마을/회의록&기술지도요청서/
- 회의록 회의 일자 : 20180123, 20180213
- 기술지도요청서와 기술지도보고서는 기술지도 신청자와 기술지도 일자를 제외하고 작성해두었습니다.

✓ 기타 참고 사항

- 다음은 공인성적서 관련으로 문의 드렸을 때 받은 답변입니다.

1) 공인성적서 비용은 연구 개발 서비스 활용 비(시험 · 분석 · 검사) 항목에 포함될 수 있음
2) 사업비 조정 시 재료비에서 가져오는 경우는 주관기관의 승인이 필요한 사항이므로 변경 전 승인 요청이 필요
3) 연구과제 추진비에서 변경하는 경우는 자체 변경사항으로 보고만 하면 됨(서류 작성이 있을 수도 있다고 하셨습니다)
4) 내부 인건비 인력의 참여율, 참여기간의 변경은 자체변경 후 보고사항

대략적인 사항이므로 한번 더 문의해보시고 진행하시면 될 것 같습니다.

✓ 기타 관련 문서 위치

/업무인수인계_김민정/연구마을/기타서류/

1) 연구마을 사업설명회 자료
2) 연구마을 사업비 집행 자료
3) 연구마을 지원제출서류
4) 자문증빙서류
5) 물품검수조서
6) 관리지침

3. 데이터 분석 결과 자료

✓ SNP 기반 FF

1) Genotyped samples 90개 위치 : /BiO/Alba/2017_backup/ngs/1~10_temp

* male fetus는 male 폴더에 있습니다.

* 쌍둥이 samples은 11_twin/ 폴더에 있습니다.

* 9번 폴더는 9, 9-1, male로 구성되어 있습니다.

2) PED/MAP file 위치:

/BiO/Alba/2017_backup/PLINK/PLINK/nGenomeCare_final.ped(& map)

3) SNP 기반 FF 계산 code :

- Code 위치(서버) : /BiO/Alba/2017_backup/final_SNP/

- Code 위치 : /업무인수인계_김민정/SNP 90/code/

- 명령어 : snp_v4.py ID.pileup

Input file : pileup file

Output file : ID_A, ID.count

* ID.count : homozygous, heterozygous alleles count 정보

- 명령어 : for aa in ls *.count ;do ./count_all.py $aa;done

Input file : ID.count

Output file : all_count

* list에 있는 ID.count 정보를 통합해서 하나의 파일에 써주는 코드

- B_list.py : non maternal alleles의 list를 뽑아주는 코드. 참조용.

Output file : ID_B.list

- 코드에 대한 자세한 사항은 /업무인수인계_김민정/기타 자료 /code_info.xlsx 에서 확인해주세요.

4) 90개 샘플 SNP FF 결과 파일 : 업무인수인계_김민정/SNP 90/SNP_FF_result(90).xlsx

5) Positive sample 4개 위치 : /BiO/Alba/2017_backup/positive/sample/

✓ Fetus reads

1) Code 위치 :

(서버) /BiO/Alba/2017_backup/fetus_read/code

2) Code 설명

명령어 : analysis_fetus_read_v2.py ID.sort.redup.sam

* fetus_read_v2.1.py : sam_info file에서 fetus list 뽑아내는 코드(!!서버로 염색체 22개의 qsub을 동시에 요청합니다).

* make_saminfo_v2.py : sam info file 생성

(chr/position/mapping_read/mapping_length/MAP_QUAL/CIGAR/read/length/flag).

* sam_info 정보 : Sam file에서 필요한 정보만 빼서 용량을 줄인 파일입니다. Flag(default 0)는 중복으로 read count 하는 것을 방지하기 위한 상태 정보입니다(위치 : /BiO/Alba/2017_backup/fetus_read/sam_info/)

* **final.py** : chromosome 별로 뽑아낸 list를 합쳐서 all_ID.Fetus_list 생성. Chromosome 별 read의 총 개수는 total_info file에 입력됩니다.

명령어 : freq_fetus_newread.py all_ID.Fetus_list

* fetus_list로부터 bp별 frequency를 count. (output file: ID.freq)

- 코드에 대한 자세한 사항은 /업무인수인계_김민정/기타 자료 /code_info.xlsx 에서 확인해주세요.

3) Fetus reads list 위치 : /BiO/Alba/2017_backup/fetus_read/out/

- all_ID.Fetus_list :
chr/position/read/length/mapping_read/mapping_length/target position
- all_ID.log : 뽑힌 fetus reads의 정보
- total_info : chromosome 별 fetus reads의 총 수를 나타낸 파일. 마지막 열은 chr1-22 fetus read 총 수.

4) Fetus read 분석 결과 파일 :
/업무인수인계_김민정/Fetus reads/analysis.xlsx
/업무인수인계_김민정/Fetus reads/freq_analysis.xlsx

✓ Imputation

- Code 위치(서버) : **/BiO/Alba/2017_backup/ngs/final_test/**final-code/
- Code 위치 : /업무인수인계_김민정/imputation/code/
- 코드에 대한 자세한 사항은 /업무인수인계_김민정/기타 자료 /code_info.xlsx 에서 확인해주세요.
- Imputation 결과 파일 위치 :
/업무인수인계_김민정/imputation/imputation_snp_1000.xlsx
/업무인수인계_김민정/imputation/imputation_snp_hrc.xlsx
(서버) HRC : /BiO/Alba/2017_backup/hrc_impu_test/
(서버) 1000 genome : /BiO/Alba/2017_backup/impu_test/
- imputation PED/MAP file 위치

(서버) HRC : /BiO/Alba/2017_backup/PLINK/hrc/ped/

(서버) 1000 genome :

/BiO/Alba/2017_backup/PLINK/re_imputation/ped/final/

✓ **기타 분석 결과 위치**

/업무인수인계_김민정/기타 분석/

1) Chr별 GC 평균 그래프/

: chromosome 별 GC의 평균을 나타낸 그래프(.pdf)

2) FF-ratio 그래프/

: read length ratio(140, 150, 160)와 FF 간의 상관관계를 나타낸 그래프(.pdf)와 관련 code

3) GC frequency/

: GC frequency 그래프(.pdf)

4) Nucleosome/

5) read frequency/

: read length에 따른 frequency 그래프(.pdf)와 관련 code

6) read length quality/

: read length에 따른 quality 평균 그래프(.pdf)와 read slice 후 coverage 측정 결과(slice coverage)

7) read ratio FF/

: read ratio를 이용하여 추정한 FF 결과 파일(readratio_ff.xlsx)과 관련 code

✓ **이전 인수인계 문서**

다음 문서는 제가 노트북을 인수 받았을 때 노트북에 남아있던 문서입니다. 결과지 양식, 의뢰서 양식, 분석 보고서, 의뢰서와 출고 파일입니다.

위치: /업무인수인계_김민정/이전 인수인계 문서/