Protein Refinement Pipeline Guide

Author: Guowei Qi
Email: guowei-qi@uiowa.edu

Using the Protein Refinement Pipeline

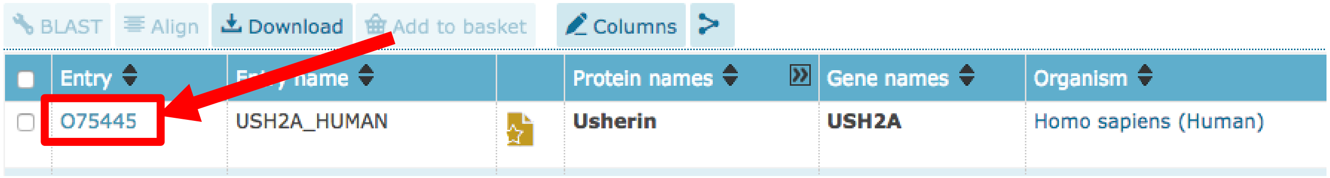
Place **RefinementPipeline.java** in a directory titled **refinementpipeline**. Navigate to **RefinementPipeline.java** within terminal and compile the Java code:

```
javac RefinementPipeline.java
```

Navigate into the directory where you would like the PDB files to be saved and enter the following command:

```
java -cp /Users/… refinementpipeline/RefinementPipeline XXXXXX
```

where **/Users/…/** represents the path to the directory in which the **refinementpipeline** package is stored and "**XXXXXX**" represents the UniProt entry ID corresponding to the protein, found here:



□	Entry	Entry name		Protein names	Gene names	Organism
□	O75445	USH2A_HUMAN		Usherin	USH2A	Homo sapiens (Human)

Enter multiple entry IDs separated by a space to refine multiple proteins with a single command:

java -cp /Users/… refinementpipeline/RefinementPipeline XXXXXX$_1$ XXXXXX$_2$ … XXXXXX$_n$

For example, if a user wanted the PDB files corresponding to the ACTG1 and ADCY1 genes to be saved to their current directory and the **refinementpipeline** directory is stored on the Desktop, they would enter the following command:

```
java -cp /Users/…/Desktop refinementpipeline/RefinementPipeline P63261 Q08828
```

For a shorter command, **vi** into your bash profile:

```
vi ~/.bash_profile
```

and enter insert mode (type **i**) to add the following line:

```
alias pipeline="java -cp /Users/… refinementpipeline/RefinementPipeline"
```

Then save (**esc**, then **:x**, then **enter**) and source your bash profile:

```
source ~/.bash_profile
```

This creates a shortcut for the Refinement Pipeline command. The command can now be entered as following:

pipeline XXXXXX$_1$ XXXXXX$_2$ … XXXXXX$_n$

The final PDB files will be saved in a directory titled **pdbFiles** with the following naming conventions:

/…/pdbFiles/GENE/RESIDUE/GENE_RESIDUE.pdb

where "**GENE**" represents the gene name, "**RESIDUE**" represents the residue range of the PDB file, and "**GENE_RESIDUE.pdb**" represents the final name of the PDB file.

After downloading the PDB files, move **upload.sh** into the same directory that contains **pdbFiles**. Use the **sed** command to replace **gqi1** with your own Hawk ID:

```
sed -i “s/gqi1/HawkID/g” upload.sh
```

Use another **sed** command to replace **gqi** with the name of your home directory on Argon:

```
sed -i “s/gqi/homeDirectory/g” upload.sh
```

Run the shell script:

```
./upload.sh
```

Terminal will prompt you to log in to Argon, which you will have to verify using Duo. This shell script compresses the **pdbFiles** directory and uploads the **tar.gz** file to Argon.

Next, navigate to the directory containing **refine.sh**. Again, use a **sed** command to replace **gqi** with the name of your home directory on Argon:

```
sed -i “s/gqi/homeDirectory/g” refine.sh
```

Upload **refine.sh** to your home directory on Argon:

```
scp refine.sh argon.hpc.uiowa.edu:/Dedicated/…/homeDirectory
```

After uploading, log in to Argon and go to your home directory (which should now contain **pdbFiles.tar.gz** and **refine.sh**). Create a directory titled **jobFiles** and make sure it contains **minimize.job**, **secondminimize.job**, **finalminimize.job, minimize.properties**, and **rotamer.job**. Then run the shell script:

```
./refine.sh > refine.log 2> error.log
```

This script unpacks the directory containing the PDB files, runs **phenix.molprobity** on each PDB and records the data, submits **minimize.job** for each PDB, records the total potential energy after minimization to both 0.8 and 0.1, submits **rotamer.job** for each minimized PDB, runs **phenix.molprobity** on each refined PDB and records the data, and records the total potential energy after refinement. The output of the script is written to **refine.log,** any errors are written to **error.log**, and the MolProbity data is written to **finalrefinementdata.csv**.

Additional Features

Experimental Structures

For some genes, no homology models exist as available PDB files on SwissModel or ModBase. In this case, the Refinement Pipeline downloads experimental structures, which have a **-expt** extension at the end of their directory names. This is currently the case only for GJB2, but if there are more genes where this is the case, the gene name can be added to the list **getExperimental** declared at the beginning of **RefinementPipeline.java**.

Checking Against the DVD GitHub

To have the Refinement Pipeline script check against the DVD GitHub and only download structures that haven't already been uploaded, use the **-g** flag:

pipeline -g XXXXXX$_1$ XXXXXX$_2$ … XXXXXX$_n$

Potential Errors

The **refine.sh** script occupies the terminal until each initial minimization has been submitted. You can continue to access Argon on your terminal to monitor the progress of the script by logging into Argon in another tab.

Make sure to periodically check **refine.log** and **error.log** for errors in the shell script. Occasionally, the following error will occur when running **refine.sh**:

Unable to run job: master got unknown command from JSV: "ERROR"
Exiting.

This error will directly follow the **phenix.molprobity** output within the log file and signals that **minimize.job** could not be submitted. In this case, enter the directory of the PDB file that caused the error and submit the job manually.

Missing data in **finalrefinementdata.csv** can oftentimes be attributed to errors in the original PDB files. Review the contents of the PDB files and the log files for abnormalities when data is missing.

Certain PDB files (often heteromers) may be too large or cause an error that keeps any step in the refinement process from finishing, which will lead to some structures that do not finish refining, as well as some incomplete data. Heteromers that are too large to refine can sometimes be split into their monomer units using PyMOL, refined individually, and spliced back together before refining the entire structure.

Terminating and rerunning the script with the same structures will lead to repeated data in **finalrefinementdata.csv**. To be careful in avoiding this, delete the previous directory containing the PDB structures and **finalrefinementdata.csv** and rerun the script using the original **pdbFiles.tar.gz** file.

The UniProt, Protein Model Portal, SwissModel, and ModBase websites may update over time and affect the functionality of the script. If the Refinement Pipeline script suddenly stops working, look for changes to the source codes of the online protein model repositories and update the script accordingly.

Gene Names and Entry IDs (Deafness Variation Database)

Gene Name	Entry ID
ACTG1	P63261
ADCY1	Q08828
ADGRV1 (GPR98)	Q8WXG9
AIFM1	O95831
ALMS1	Q8TCU4
ATP2B2	Q01814
ATP6V1B1	P15313
BDP1	A6H8Y1
BSND	Q8WZ55
CABP2	Q9NPB3
CACNA1D	Q01668
CCDC50	Q8IVM0
CD164	Q04900
CDC14A	Q9UNH5
CDH23	Q9H251
CEACAM16	Q2WEN9
CIB2	O75838
CISD2	Q8N5K1
CLDN14	O95500
CLIC5	Q9NZA1
CLPP	Q16740
CLRN1	P58418
COCH	O43405
COL11A1	P12107
COL11A2	P13942
COL2A1	P02458
COL4A3	Q01955
COL4A4	P53420
COL4A5	P29400
COL4A6	Q14031
COL9A1	P20849
COL9A2	Q14055
CRYM	Q14894
DCDC2	Q9UHG0
DFNA5 (GSDME)	O60443
WHRN (DFNB31)	Q9P202
PJVK (DFNB59)	Q0ZLH3
DIABLO	Q9NR28
DIAPH1	O60610
DIAPH3	Q9NSV4
DSPP	Q9NZW4
EDN3	P14138
EDNRB	P24530
ELMOD3	Q96FG2
EPS8	Q12929
EPS8L2	Q9H6S3

ESPN	B1AK53
ESRRB	O95718
EYA1	Q99502
EYA4	O95677
FAM65B (RIPOR2)	Q9Y4F9
FGF3	P11487
FGFR1	P11362
FGFR2	P21802
FOXI1	Q12951
GATA3	P23771
GIPC3	Q8TF64
GJB2	P29033
GJB3	O75712
GJB6	O95452
GPSM2	P81274
GRHL2	Q6ISB3
GRXCR1	A8MXD5
GRXCR2	A6NFK2
HARS2	P49590
HGF	P14210
HOMER2	Q9NSB8
HSD17B4	P51659
ILDR1	Q86SU0
KARS	Q15046
KCNE1	P15382
KCNJ10	P78508
KCNQ1	P51787
KCNQ4	P56696
KITLG	P21583
LARS2	Q15031
LHFPL5	Q8TAF8
LOXHD1	Q8IVV2
LOXL3	P58215
LRTOMT	Q8WZ04
MARVELD2	Q8N4S9
MCM2	P49736
MET	P08581
MIR96	
MITF	O75030
MSRB3	Q8IXL7
MT-RNR1	
MT-TL1	
MT-TS1	
MYH14	Q7Z406
MYH9	P35579
MYO15A	Q9UKN7
MYO3A	Q8NEV4
MYO6	Q9UM54
MYO7A	Q13402

NARS2	Q96I59
NLRP3	Q96P20
OPA1	O60313
OSBPL2	Q9H1P3
OTOA	Q7RTW8
OTOF	Q9HC10
OTOG	Q6ZRI0
OTOGL	Q3ZCN5
P2RX2	Q9UBL9
PAX3	P23760
PCDH15	Q96QU1
PDZD7	Q9H5P4
PEX1	O43933
PEX6	Q13608
PNPT1	Q8TCS8
POLR1C	O15160
POLR1D	P0DPB6
POU3F4	P49335
POU4F3	Q15319
PRPS1	P60891
PTPRQ	Q9UMZ3
RDX	P35241
ROR1	Q01973
S1PR2	O95136
SERPINB6	P35237
SIX1	Q15475
SIX5	Q8N196
SLC17A8	Q8NDX2
SLC22A4	Q9H015
SLC26A4	O43511
SLC26A5	P58743
SLITRK6	Q9H5Y7
SMPX	Q9UHP9
SNAI2	O43623
SOX10	P56693
STRC	Q7RTU9
SYNE4	Q8N205
TBC1D24	Q9ULP9
TBX1	O43435
TCOF1	Q13428
TECTA	O75443
TECTB	Q96PL2
TIMM8A	O60220
TJP2	Q9UDY2
TMC1	Q8TDI8
TMEM132E	Q6IEE7
TMIE	Q8NEW7
TMPRSS3	P57727
TNC	P24821

TPRN	Q4KMQ1
TRIOBP	Q9H2D6
TSPEAR	Q8WU66
TWNK	Q96RR1
USH1C	Q9Y6N9
USH1G	Q495M9
USH2A	O75445
WFS1	O76024