2019-5-6

Diagnostic Variant Analysis Pipeline Workflow

Contact:
Evan Baugh <eb3142@cumc.columbia.edu>
Joe Hostyk <jh3958@cumc.columbia.edu>

~~~~~~

Legend:

- Hardcoded filenames
- Variable filenames - make sure to consistently use the same name (!)
- Nick's Scripts (/nfs/projects/Diagnostic/diagnostic_pipeline_scripts/nick_stong_scripts)
- Evan's Scripts (/nfs/projects/Diagnostic/diagnostic_pipeline_scripts)
- Server on which you should run these commands

I. **General set-up/locations**

A. Scripts are in /nfs/projects/Diagnostic/diagnostic_pipeline_scripts
   - `PATH=$PATH:/nfs/projects/Diagnostic/diagnostic_pipeline_scripts/nick_stong_scripts` will let you easily run the commands below.

B. You may need to create a ".my.cnf" file in your home directory to provide proper mysql database access, the contents are not copied here (involves a databases password) but you can copy the version in /nfs/projects/Diagnostic/diagnostic_pipeline_scripts into your home directory (cp /nfs/projects/Diagnostic/diagnostic_pipeline_scripts/.my.cnf /home/me)

C. The commands below don't take absolute paths, so you should `cd` into the directory where you wish to run the analysis.

II. **Obtaining Unanalyzed Samples**

Official IGM Diagnostic Pipeline usage: Evan will have run the scrip, so the user can check /nfs/projects/Diagnostic/unanalyzed_sample_stage for an unanalyzed input file corresponding to your samples/project of interest.

Alternatively see "Obtaining Unanalyzed Samples Directly" below.

III. **Preparing Input for ATAV (trios and non-trios)**

dev1 (10.73.50.104)

Official IGM Diagnostic Pipeline usage:

`step1_prepare_unanalyzed_samples.sh unanalyzed.out`

This script automates a series of Nick's scripts AND adds automatic filtering for specific sample and family failure criteria

Alternatively you may run all of the individual steps, detailed below

---

Breakdown of official IGM pipeline (Nick's individualized scripts, in detail):
~~~~~~

- Extract sample information
    a) Determine family IDs
        (1) `famsFromPro.sh unanalyzed.out`
        (2) Automatically outputs "Fams.txt"
    b) Determine all family members
        (1) `famMemFromFams.sh Fams.txt`
        (2) Automatically outputs "allFamilyMembers.txt"
    c) Extract sample information
        (1) `seqDBoutputFromCHGVIDs.sh allFamilyMembers.txt > seqDBoutput.csv`
        (2) Input file from last step, output filename can be named as desired
- Filter samples for properly complete families
    a) Convert the output .CSV to .TSV
        (1) `python convert_csv_to_tsv.py seqDBoutput.csv`
    b) Filter out incomplete families
        (1) `python filter_samples_from_seqDB.py seqDBoutput.csv seqDBoutput.filtered_fams.csv`
        (2) Automatically outputs a .CSV and individual files for the different filter 'incomplete' cases (gender mismatch, missing data, etc.)
- Determine the trios and non-trio families
    a) `makeTrios.R seqDBoutput.filtered_fams.csv trios.ped | cut -d ' ' -f 8|sed 's/"//g' > nonTrioFams.txt`
    b) Generate input for non-trio families
        (1) `makeNonTrioFams.R seqDBoutput.filtered_fams.csv | grep --ignore-case -wf nonTrioFams.txt > nonTrioFams.ped`

IV. Extract sample data using ATAV

qs1 (10.73.50.79 - ATAV job-submission node)

`runTrio.sh [controlFilePath]`

generates 4 ATAV submissions:

- nonTrio (list-var-geno)
- Trio
- Parental mosaic
- Coverage

Wait for those to finish running. (~2 hours.)

V. Generate initial Variant Report

dev1 (10.73.50.104)

A. Prepare ATAV output for generating the Variant Report
    - `makeAtavlinks.sh`
      creates symlinks, hardcoded for future steps

- `splitAllNonTrios.sh`
  creates output directories, .ped file, and _genotypes file

B. Generate the initial trio Variant Report and final/only non-trio Variant Report.
   - `generateVariantReportDragen -o variant_report_DATE_PROJECT_trio.xlsx -p seqDBoutput.filtered_fams.csv -s`
   - `generateNonTrioVariantReportDragen -o variant_report_DATE_PROJECT_non_trio.xlsx -p seqDBoutput.filtered_fams.csv -s`

VI. Manually inspect variants using IGV

A. Confirm that you have permissions to access all BAMs required to inspect in IGV
   - python check_bam_path_permissions.py PATH_TO_OUTPUT_DIRECTORY
   - This script simply compares the filenames in "IGV/dnm.IGV.bamloc" and "IGV/dnm.IGV.batch" to find paths that could not be accessed, and by default it prints this list of paths to screen 'formatted' to be copied into a RedMine ticket requesting file access
   - Once you have permissions, re-run the Variant Report generation steps

B. Create a "denovo.IGV.csv" with manually annotated variants which Pass
   - Open "denovo.csv" or create a copy named "denovo.IGV.csv"
   - Excel reformats the "Exons" and "X.Transcript" columns as dates. (E.g. Exon 2/3 becomes Februuary 3.) To change back, right click the columns -> "Format cells" -> "Custom". Type: "m/d". Then it will go back to being numbers
   - Create an "IGV.Pass" to the left of "MGI.Essential"
   - Go through the IGV pictures. For each variant, if it looks bad or if it's real but not de novo, mark as "FALSE"
   - Save as "denovo.IGV.csv"

VII. Now can generate final trio Variant Report

A. `generateVariantReportDragen -o variant_report_DATE_PROJECT_trio.xlsx -p seqDBoutput.filtered_fams.csv -s --noVEP --denovoIGV denovo.IGV.csv`

VIII. If generating a variant report for GCs or collaborators, send them the results

A. Copy (rysnc) the entire output directory to its associated project directory
   - /nfs/projects/CUMC/Neuro/Candidates2
   - /nfs/projects/CUMC/DiagSeq/Candidates
   - /nfs/projects/CUMC/picu

B. Email the output Variant Report (.XLSX) and sample summaries (.DOCX) to your corresponding collaborators or GCs

<u>Obtaining Unanalyzed Samples Directly</u>

Nick provided a script for obtaining unanalyzed samples from Sequence, however it appears to require a consistent list of ALL previously run input...yet this shared a filename with one of the hardcoded filenames used during the Variant

Report generation...requiring careful attention to the run directory with the potential danger of considering unanalyzed samples as previously analyzed samples.
For simplicity, a separate file has been maintained that will include ALL IDs of previously run analyses (/nfs/projects/Diagnostic/unanalyzed_sample_stage/all_previous_run_CHGVIDs.txt) and an accompanying script (/nfs/projects/Diagnostic/unanalyzed_sample_stage/check_all_sequencing_unanlyzed.py) to be run with a bulk export of ALL unanalyzed samples, determining new samples by comparing to the list of previously run CHGVIDs and separating samples into projects.
If instead you wish to produce your own commands for interacting with Sequence, please see Nick's script "getUnanalyzedSamples.sh" which can be run with an input argument of (I think) a CHGVID substring to filter by project. It outputs the obtained samples to stdout so an example run command looks like

getUnanalyzedSamples.sh neuro > unanalyzed_neuro.out

However, keep in mind that several projects use "diagseq" and "neuro" as project substrings even though they are part of distinc projects, such as "PICU" which lists this distinction in the "AKA" column - something Nick's script cannot currently handle, requiring removal of PICU samples after generating this output.

For simplicity, we intend moving forward to periodically pull down the full list of output from Sequence and filter novel samples by comparing to the list of previously run IDs (which itself will be continuously updated as analyses are completed) and then sort output samples into the desired projects, such that the entire Varint Analysis Report generation Pipeline above can be run individually on those unanalyzed output files.

Known contributors: Slavé Petrovski, Quanli Wang, and Xiaolin Zhu established the framework. Nick Stong developed and improved the pipeline from 2016-2019. Joe Hostyk and Evan Baugh authored this workflow in May 2019.

The wiki is listed here: https://redmine.igm.cumc.columbia.edu/projects/atav/wiki/Diagnostic_Analysis_Framework
.
The following list is taken from there.

2016

Shashi V, Pena LDM, Kim K, Burton B, et al. De novo truncating variants in ASXL2 are associated with a unique and recognizable clinical phenotype: further involvement of the ASXL gene family in neurodevelopmental genetic syndromes. American Journal of Human Genetics 2016; 6;99(4):991-999.
https://www.ncbi.nlm.nih.gov/pubmed/27693232

Kim J-H, Shinde DN, Reijnders MRF, Hauser NS, et al. De novo loss-of-function mutations in SON disrupt RNA-splicing of genes essential for brain development and metabolism, causing an intellectual disability syndrome. American Journal of Human Genetics 2016; pii: S0002-9297(16)30267-1.
https://www.ncbi.nlm.nih.gov/pubmed/27545680

Petrovski S, Parrott RE, Roberts JL, Huang H, et al. Dominant Splice Site Mutations in PIK3R1 Cause Hyper IgM Syndrome, Lymphadenopathy and Short Stature. Journal of Clinical Immunology 2016; 36(5):462-71.
https://www.ncbi.nlm.nih.gov/pubmed/27076228

Petrovski S, Küry S, Myers CT, Anyane-Yeboa K, et al. Germline de novo mutations in GNB1 cause severe neurodevelopmental disability, hypotonia and seizures. American Journal of Human Genetics 2016; 98 (5), 1001-1010.
https://www.ncbi.nlm.nih.gov/pubmed/27108799

2015

Halvorsen M, Petrovski S, Shellhaas R, Tang Y, Crandall L, Goldstein DB, Devinsky O. Mosaic mutations in early-onset genetic diseases. Genetics in Medicine 2015; doi: 10.1038/gim.2015.155.
https://www.ncbi.nlm.nih.gov/pubmed/26716362

Williams C, Jiang YH, Shashi V, Crimian R, Schoch K, McHale D, Goldstein DB, Petrovski S. Additional Evidence that PGAP1 Complete Loss of Function Causes Autosomal Recessive Global Developmental Delay and Encephalopathy. Clinical Genetics 2015; doi: 10.1111/cge.12581
https://www.ncbi.nlm.nih.gov/pubmed/25823418

Petrovski S, Shashi V, Schoch K, Petrou S, et al. Exome sequencing results in successful riboflavin treatment of a rapidly progressive neurological condition. Molecular Case Studies 2015; doi:10.1101/mcs.a000257
http://molecularcasestudies.cshlp.org/content/1/1/a000257.short

Dobbs K, Conde CD, Zhang SY, Parolini S, et al. DOCK2 and recessive immunodeficiency with early-onset invasive invasive infections. New England Journal of Medicine 2015; 372(25):2409-22
https://www.ncbi.nlm.nih.gov/pubmed/26083206

Zhu X, Petrovski S, Xie P, Ruzzo EK, et al. Whole exome sequencing in undiagnosed genetic diseases: Interpreting 119 trios. Genetics in Medicine 2015; 17,774–781.
https://www.ncbi.nlm.nih.gov/pubmed/25590979

Pre-2015

Shashi V, Xie P, Schoch K, Goldstein DB, et al. The RBMX gene as a candidate for the Shashi X-linked intellectual disability syndrome. Clinical Genetics 2014; doi: 10.1111/cge.12511
https://www.ncbi.nlm.nih.gov/pubmed/25256757

Need AC, Shashi V, Hitomi Y, Schoch K, Shianna KV, McDonald MT, Meisler MH, Goldstein DB. Clinical application of exome sequencing in undiagnosed genetic conditions. Journal of Medical Genetics 2012; doi:10.1136/jmedgenet-2012-100819
https://www.ncbi.nlm.nih.gov/pubmed/22581936