Mendel Genetic Counseling

Installation Instructions
using the Julia Icon (REPL)

Before the Workshop

- It is important to install Julia
- Install Open Mendel modules:
  - SnpArrays.jl
  - MendelSearch.jl
  - MendelBase.jl
  - MendelGeneticCounseling.jl#ASHG2019
- And test Open Mendel with a small test example

Notebook() or Commands within the Icon?

- The Open Mendel team prefers to use Jupyter notebooks whenever possible. See for example MendelGeneticCounselingTutorial.ipynb at https://github.com/OpenMendel/GeneticCounseling_ASHG2019. We prefer them primarily because they make reproducibility easier. However you might find it easier to run Julia by just clicking on the Julia Icon.
- This presentation is designed to assist users who can’t or don’t want to use the IJulia- Jupyter notebook option.

Step 1: Install Julia and Open Julia

- Go to the Julia download site, https://julialang.org/downloads/ and download the latest stable release of julia (version 1.2.0 on 9/19/2019).
- Once Julia is downloaded click on the julia icon to open the Icon

```
Last login: Wed Sep 18 18:56:25 on ttys001
hg-js-mbp:~ janets$ exec '/Applications/Julia-1.2.app/Contents/Resources/julia/b
in/julia'

               _
   _       _ _(_)_     |  Documentation: https://docs.julialang.org
  (_)     | (_) (_)    |
   _ _   _| |_  __ _   |  Type "?" for help, "]?" for Pkg help.
  | | | | | | |/ _` |  |
  | | |_| | | | (_| |  |  Version 1.2.0 (2019-08-20)
 _/ |\__'_|_|_|\__'_|  |  Official https://julialang.org/ release
|__/                   |

julia> 
```

Step 2: Installing MendelGeneticCounseling

- Use the package manager in Julia.
- Invoke by typing ”]” at the Julia prompt
- Add **in order** the modules: SnpArrays.jl, MendelSearch.jl, MendelBase.jl and MendelGeneticCounseling.jl#ASHG2019

```
pkg>  add https://github.com/OpenMendel/SnpArrays.jl
pkg>  add https://github.com/OpenMendel/MendelSearch.jl
pkg>  add https://github.com/OpenMendel/MendelBase.jl
pkg>add https://github.com/OpenMendel/MendelGeneticCounseling.jl#ASHG2019
```

- Then give the command: build SpecialFunctions

```
pkg>  build SpecialFunctions
```

- To exit the package manager press the delete key

Step 3: Check for Input files

- Check that working directory contains your files

```
julia> pwd()
julia> readdir()
```

- If the files aren't there change to the appropriate directory :

```
julia> cd("/Users/janets/GeneticCounseling/SmallSample")
```

- Then check for the input files by typing

```
julia> readdir()
 "ControlSmallParametric.txt"
 "HeterozygousRisk.txt"
 "LocusSmall.txt"
 "PedSmall.csv"
 "PhenoSmall.txt"
 "SampleOutput"
```

Step 4: Run MendelGenetiCounseling

- First compile MendelGeneticCounseling

```
julia> using MendelGeneticCounseling
```

- Then run the test problem by typing:

```
julia> GeneticCounseling("ControlSmallParametric.txt")
```

- Result:

```
The risk =  0.03892.
```

If you Decide you want to use the Jupyter Notebooks

- To use Jupyter notebooks within Julia you want to install IJulia within the package manager

```
] add IJulia
```

- then leave the package manager by hitting the delete key
- To run IJulia

```
julia>using IJulia
julia>notebook()
```

- When finished type “control C”

Step 5: Report Any Difficulties

- We’ve tried hard to test the installation and implementation of MendelGeneticCounseling on both Mac and PCs but you may have problems we haven’t encountered before.
- So please report any problems on the open mendel github site or by email jsinshei@g.ucla.edu as soon as you have them (don’t wait until the week of the workshop because we might not be able to help you in time).
- Be sure to provide as much information as you can on the commands you tried and the error messages you received. Also give us an email address that you check on a regular basis so the Open Mendel team can help you solve any problems.

Mendel Genetic Counseling

Tutorial

using the Julia Icon (REPL)

Before the Workshop

- Be sure you have downloaded all the input files in Open Mendel github subdirectories:

github.com/OpenMendel/GeneticCounseling_ASHG2019/tree/master/BRCA

github.com/OpenMendel/GeneticCounseling_ASHG2019/tree/master/Cholestrol

Jupyter Notebook or Julia REPL (Icon)?

- The Open Mendel team prefers to use Jupyter notebooks whenever possible, primarily because they make reproducibility easier. However you might find it easier to run Julia by just clicking on the Julia Icon.
- This version of the tutorial is designed to assist users who can't or don't want to use the IJulia-jupyter notebooks option.

The Purpose of the Tutorial

- To demonstrate how to calculate genetic risks for individuals using their family histories and covariate information.
- The pdf of the jupyter notebook, MendelGeneticCounselingTutorial.pdf, has details and more background than this version. Please look at it.

Checking Julia Version and Updates

- To insure reproducibility it is important to determine the version of Julia being used and check for any updates to the program or the relevant modules.

```
julia> versioninfo()
Julia Version 1.2.0
Commit c6da87ff4b (2019-08-20 00:03 UTC)
Platform Info:
  OS: macOS (x86_64-apple-darwin18.6.0)
  CPU: Intel(R) Core(TM) i7-6567U CPU @ 3.30GHz
  WORD_SIZE: 64
  LIBM: libopenlibm
  LLVM: libLLVM-6.0.1 (ORCJIT, skylake)

(v1.2) pkg> update
  Updating registry at `~/.julia/registries/General`
```

Examples:

- Example 1:
  - Uses one of the available glm models, the Gamma distribution, with the link function = log.
  - The pedigree and cholesterol phenotypes are originally from Schrott et al. (1972) Ann Int Med 76:711–720.
  - Determine the risk that a child (IV11) of a women with abnormally high cholesterol has a heterozygous genotype.
- Example 2:
  - Uses a penetrance file where values are stratified by sex and age.
  - The pedigree and penetrance are for BRCA1 and breast cancer and come from analyze.myvariant.org
  - Determine how likely a currently unaffected 38 year old woman (individual 18) has a heterozygous genotype.

Example 1:

- First check that the desired files are in your directory

```
julia> pwd()
julia> readdir()
```

- If you need to change to a different directory, you can use the cd command
  - Example:

```
julia> cd("/Users/janets/GeneticCounseling/Chol")
```

Ex 1, Step 1: Examine the Pedigree File

- Examine the pedigree files, Note that the pedigree is there twice.

```
julia> readlines("PedChol.csv")
"Pedigree,Person,Father,Mother,Sex,HC,Age,lnChol,Chol,H
igh_Chol"
" TOP , III11 , II2 , II1 ,2,,22,6.39,595,1"
" TOP , II16 , I2 , I1 ,1,,38,6.17,479,1"
⋮
"BOTTOM, III30 ,,,1,,20,NA,NA,NA"
```

- In the first pedigree (TOP) individual IV11's genotype is present. In the second pedigree (BOTTOM) individual IV11’s genotype is absent.
- The program determines the likelihood of each pedigree to calculate the conditional probability of IV11’s genotype given the rest of data from IV11 and the pedigree members.

Ex 1, Step 2: Examine the Control File

```
julia> readlines("ControlParametricPenetranceExample.txt")
16-element Array{String,1}:
"#"
"# Input and Output files."
"#"
"locus_file = LocusChol.txt"
"pedigree_file = PedChol.csv"
"phenotype_file = PhenoChol.txt"
"output_file = CholHeterozygousRisk.txt"
"#"
"# Analysis parameters for Genetic Counseling option."
"#"
"glm_mean = 4.691+0.562(max(allele1,allele2))+0.00194Age+0.036Sex"
"glm_response = GammaDist"
"glm_link = LogLink"
"glm_trait = Chol"
"glm_scale = 44.68"
"glm_trials = 1”
```

- The control file provides the input file names, the mean parameters and variables, the scale parameters, the distribution and the link function.

Ex 1, Step 3: Examine the Locus and Penetrance Files

```
julia> readlines("LocusChol.txt")
3-element Array{String,1}:
"Locus,Allele,Chromosome,European"
"HC,-,autosome,0.9600"
"HC,+,autosome,0.0400”
julia> readlines(”PhenoChol.txt")
 4-element Array{String,1}:
"Locus,Phenotype,Genotypes"
"HC,Homozygous_Normal,\"-/-\""
"HC,Homozygous_Mutant,\"+/+\""
"HC,Heterozygous,\"+/-\""
```

- The locus file provides the putative disease locus name, the allele names and frequencies in specific populations (can be more than one). The phenotype file provides the locus phenotype that corresponds to a genotype. Note that this is useful when genotypes are partially observed (e.g. the classic ABO locus).

Step 4: Run MendelGeneticCounseling

- First compile MendelGeneticCounseling

```
julia> using MendelGeneticCounseling
```

- Then run the test problem by typing:

```
julia> GeneticCounseling("ControlParametricPenetranceExample.txt")
```

- Result (displayed on screen and to a file):

```
The risk = 0.27557
```

- Try changing IV11’s genotype to homozygous normal and rerun.
- Try making III13 the individual of interest and determine the probability that her genotype is heterozygous.

Example 2:

- Again check that the desired files are in your directory

```
julia> pwd()
julia> readdir()
```

- If needed, change to a different directory using the cd command

Ex 2, Step 1: Examine the Penetrance File

```
julia> readlines("PenBRCAExample.csv")
15-element Array{String,1}:
"Homozygous_Normal,Heterozygous,Homozygous_Mutant,Sex,Risk_decade"
"0.000000885,0.001025896,0.001025896,female,1"
"0.000040997,0.047524,0.047524,female,2"
"0.00189916,0.18042,0.18042,female,3"
"0.00878848,0.3736,0.3736,female,4"
"0.0275136,0.5752,0.5752,female,5”
⋮
"0.000085,0.003,0.003,male,4"
"0.00027,0.0062,0.0062,male,5"
"0.00067,0.012,0.012,male,6"
"0.0012,0.018,0.018,male,7
```

- The penetrance is stratified by sex and age group (1 = 0 to 20, 2 = 20 to 30 etc).
- The penetrance is assumed to follow a dominant model.

Ex 2, Step 2: Examine the Pedigree File

```
julia> readlines("PedBRCAExample.csv")

41-element Array{String,1}:
"Pedigree,Person,Father,Mother,Sex,BRCA,Age,Risk_decade,Proband,Cancer"

"TOP,1,0,0,male,,79,7,0,-1"

"TOP,2,0,0,female,,78,7,0,-1"
"TOP,3,1,2,female,Heterozygous,40,4,0,1"

⋮

"BOTTOM,16,9,10,female,Heterozygous,60,6,0,0"
"BOTTOM,17,11,12,female,Heterozygous,49,4,1,1"
"BOTTOM,18,11,12,female,,38,3,0,0"
"BOTTOM,19,11,12,male,Heterozygous,36,3,0,0"
"BOTTOM,20,11,12,female,Heterozygous,48,4,0,1”
```

- The pedigree is again represented twice with the first pedigree providing the likelihood that individual 18 has a heterozygous genotype along with the rest of her and her family’s data and the second pedigree providing the likelihood of the rest of her and her family’s data.

Ex 1, Step 3: Examine the Locus and Penetrance Files

```
julia> readlines("LocusBRCAExample.txt")

3-element Array{String,1}:
"Locus,Allele,Chromosome,European"
"BRCA,\"1\",Autosome,0.998" "BRCA,\"2\",Autosome,0.002"

julia> readlines("PhenoBRCAExample.txt")

4-element Array{String,1}: "Locus,Phenotype,Genotypes"
"BRCA,Homz_rare,\"2/2\"" "BRCA,Heterozygous,\"1/2\""
"BRCA,Homz_common,\"1/1\""
```

Ex 2, Step 4: Examine the Control Files

```
julia> readlines("ControlBRCAExample.txt")
10-element Array{String,1}:
 "#"
 "# Input and Output files."
 "#"
 "locus_file = LocusBRCAExample.txt  "
 "pedigree_file = PedBRCAExample.csv"
 "phenotype_file = PhenoBRCAExample.txt"
 "penetrance_file = PenBRCAExample.csv"
 "output_file = BRCAExampleOut.txt"
 "#"
 "disease_status = Cancer"
```

- This control file is simpler than example 1’s control file
  - The penetrance class information replaces the GLM information.
  - Disease_status = Cancer provides the column in the pedigree file with the affection status of each individual.

Step 5: Run MendelGeneticCounseling

- Compiling MendelGeneticCounseling is only necessary if you didn’t run the first example:

```
julia> using MendelGeneticCounseling
```

- Run the test problem:

```
julia> GeneticCounseling("ControlBRCAExample.txt")
```

- Result (displayed on screen and to a file):

```
The risk = 0.45091
```

- Try changing 18’s genotype to homozygous normal and rerun.
- Try changing 18’s age to 20 or 68 and calculate the probability that she has a heterozygous genotype.

If you Decide you want to use the Jupyter Notebooks

- To use Jupyter notebooks within Julia you want to install IJulia within the package manager

```
] add IJulia
```

- then leave the package manager by hitting the delete key
- To run IJulia

```
julia>using IJulia
julia>notebook()
```

- When finished type “control C”