

# Reads2Map

Practical and reproducible workflows to build polyploid linkage maps from sequencing data

**Cris Taniguti**

Lucas Taniguti

Jeekin Lau

Augusto Garcia

Marcelo Mollinari

David Byrne

Oscar Riera-Lizarazu



**TEXAS A&M**  
UNIVERSITY®



**STATISTICAL  
GENETICS  
·LAB·**  
ESALQ / USP

**NC STATE**  
UNIVERSITY



This project is funded by USDA NIFA Specialty Crop Research Initiative  
Award # 2020-51181-32156 (09/01/20 - 08/31/24)



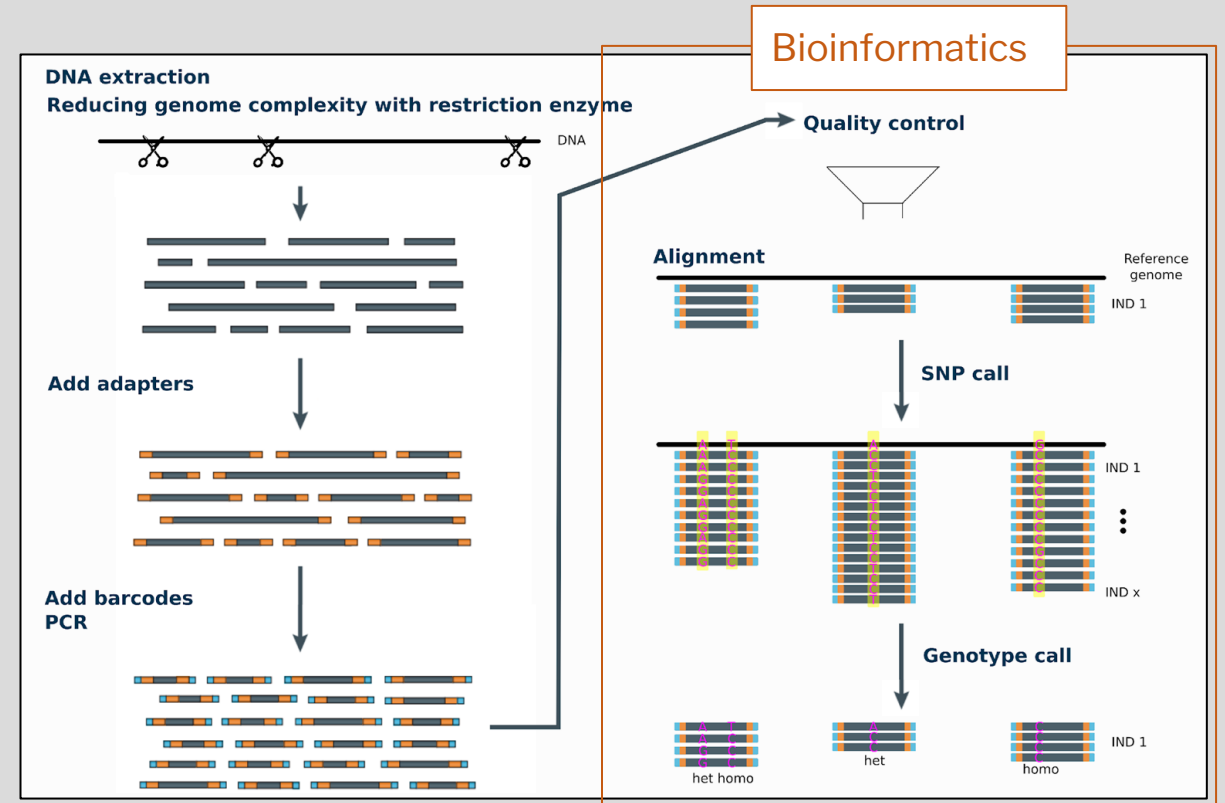
**TEXAS A&M**  
UNIVERSITY®

# Motivation

A genotyping-by-sequencing story



- My first GBS data (2015)
  - Biology background



# Motivation

## A genotyping-by-sequencing story

- My first GBS data (2015)
  - Biology background
  - File formats

```
@SRR6249787.7 7 length=143
AGCTTGAACCAATTCCTACAGTGATGAATATAAGCGCAATTGAAATTCCTGGGGAGTTA
+SRR6249787.7 7 length=143
IIIIIGGIIIIIGIIIIIIIIIIIGIIIIIIIIIIIIIIIIIIIIIIIIIGIIIIIIII
@SRR6249787.8 8 length=143
AGCTTTCAAACGATGTCTATTTACCTCAAACATTGTCATTCTTGGATTCTCAATATC
+SRR6249787.8 8 length=143
IIIIIGIIGIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIGI
@SRR6249787.9 9 length=143
AGCTTGAAACTTTCAGTATCCCCTGCCCTGCAGACACCATTGATTGCAGGATCCAT
+SRR6249787.9 9 length=143
IIIIIIIIIIIGIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII
@SRR6249787.10 10 length=143
AGCTTCCTTTACTGAAAAACAATGCTATTATTATATACATGAGATCGGAAGAGCGGT
+SRR6249787.10 10 length=143
IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIGIIIIIIIIIIIIIIIIIIIIII
```

FASTQ

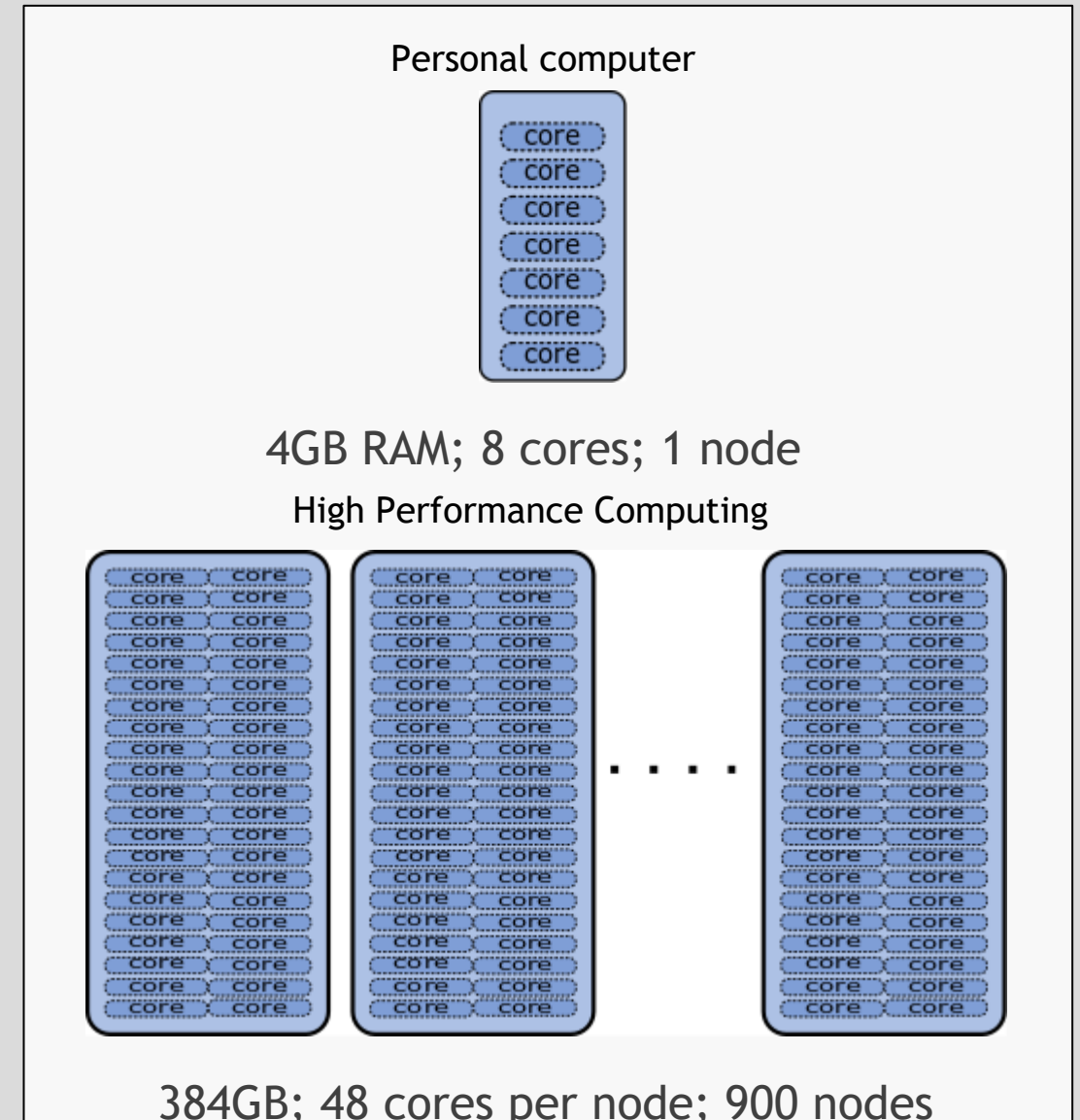
```
##FORMAT=<ID=GQ,Number=1,Type=Float,Description="Genotype Quality">
##FORMAT=<ID=PL,Number=R,Type=Float,Description="Normalized, Phred-scaled likelihoods for AA,AB,BB genotype
es where A=ref and B=alt; not applicable if site is not biallelic">
#CHROM POS ID REF ALT QUAL FILTER INFO FORMAT II_1_37 II_1_81 II_3_08 II_3_09 II
_3_10 II_3_10a II_3_11 II_3_14 II_3_15 II_3_18 II_3_19 II_3_20 II_3_23 II_3_24 II_3_25 II_3_26 II_3
_27 II_3_28 II_3_29 II_3_31 II_3_39 II_3_40 II_3_43 II_3_44 II_3_59 II_3_61 II_3_63 II_3_64 II_3_65 II_3_6
6 II_3_68 II_3_72 II_3_73 II_3_74 II_3_75 II_3_76 II_3_77 II_3_78 II_3_79 II_5_02 II_5_03 II_5_04 II_5_08
II_5_11 II_5_12 II_5_13 II_5_14 II_5_15 II_5_17 II_5_20 II_5_21 II_5_22 II_5_23 II_5_25 II_5_27 II_5_30 II
_5_31 II_5_32 II_5_33 II_5_34 II_5_36 II_5_40 II_5_42 II_5_43 II_5_45 II_5_48 II_5_51 II_5_52 II_5_54 II_5
_59 II_5_61 II_5_62 II_5_72 II_5_78 II_5_80 II_81 I_3_03 I_3_06 I_3_09 I_3_12 I_3_15 I_3_27 I_3_31
I_3_32 I_3_34 I_3_38 I_3_42 I_3_45 I_3_50 I_3_53 I_3_55 I_3_56 I_3_58 I_3_59 I_3_60 I_3_66
I_3_67 I_3_70 I_3_71 I_3_72 I_3_75 I_3_77 I_3_78 I_4_01 I_4_02 48 I_4_12 I_4_15 I_4_19 I
_4_21 I_4_26 I_4_28 I_4_35 I_4_39 I_4_48 I_4_51 I_4_62 I_4_65 I_4_73 I_4_76 I_4_78 I_4_79 I_4
_81 PT_F PT_M
Chr10 99721 Chr10_99721 C A NS=123;DP=2236;LU=2;HH=0.613787693283989
GT:AD:DP:PL:GQ 0/0:1,0:1,0:3,97:3 0/0:11,0:11:0,20,98:20 0/0:2,0:5,97:5 0/0:17,0:1
7:0,26,98:26 0/0:16,0:16:0,25,98:25 0/0:33,0:33:0,99,99:99 0/0:19,0:19:0,26,98:26 0/1:5,5:10:99,0,99:9
9 0/1:4,4:8:99,0,99:99 0/1:11,15:26:99,0,99:99 0/1:13,17:30:99,0,99:99 0/1:7,6:13:99,0,99:99 0/0:18
,0:18:0,26,98:26 0/1:11,14:25:99,0,99:99 0/1:2,12:14:99,0,99:99 0/0:26,0:26:0,29,98:29 0/1:6,7:13:99,0,
99:99 0/0:18,0:18:0,26,98:26 0/1:13,24:37:99,0,99:99 0/1:10,15:25:99,0,99:99 0/0:13,0:13:0,22,98:22 0/
0:25,0:25:0,29,98:29 0/0:21,0:21:0,29,98:29 0/0:7,0:7:0,14,98:14 0/0:9,0:9:0,17,98:17 0/0:6,0:6:0,
13,98:13 0/0:15,0:15:0,23,98:23 0/1:3,13:16:99,0,99:99 0/0:21,0:21:0,29,98:29 0/1:11,3:14:25,0,98:25
0/0:15,0:15:0,23,98:23 0/1:6,4:10:99,0,99:99 0/1:13,11:24:99,0,99:99 0/0:14,0:14:0,22,98:22 0/0:23,0
:23:0,29,98:29 0/1:3,3:6:99,0,99:99 0/1:10,5:15:99,0,99:99 0/1:12,7:19:99,0,99:99 0/1:12,3:15:23,0,9
8:23 0/0:17,0:17:0,26,98:26 0/1:5,5:10:99,0,99:99 0/0:8,0:8:0,16,98:16 0/1:12,10:22:99,0,99:99 0/0:
20,0:20:0,29,98:29 0/1:4,7:11:99,0,99:99 0/1:10,7:17:99,0,99:99 0/1:5,5:10:99,0,99:99 0/0:12,0:12:0
```

VCF

# Motivation

A genotyping-by-sequencing story

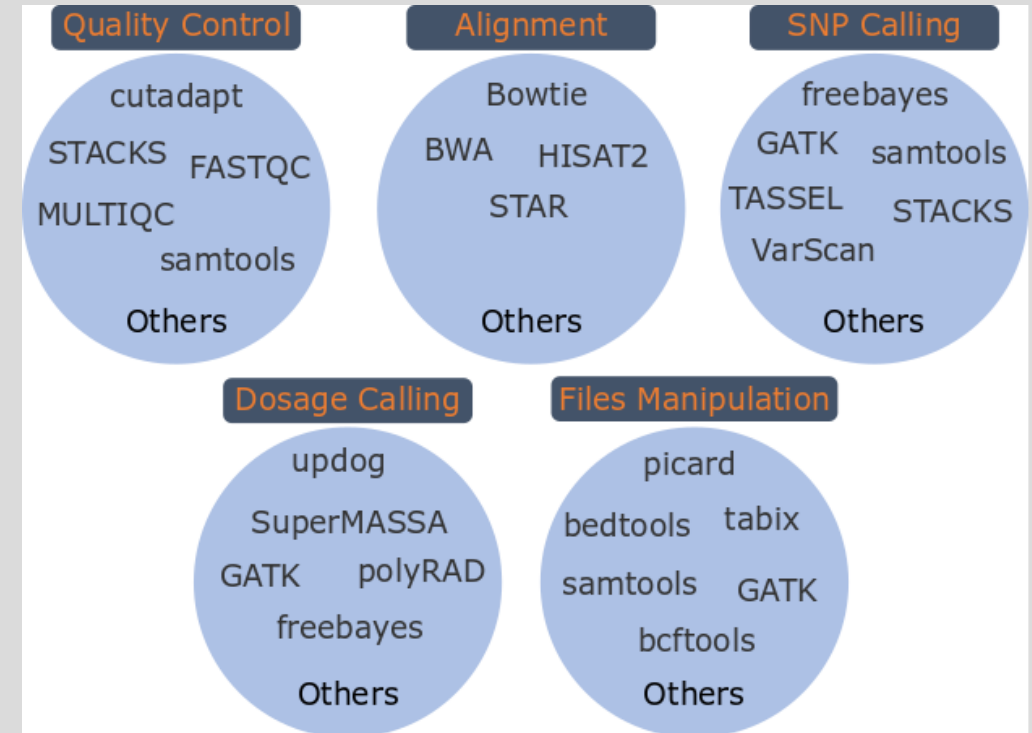
- My first GBS data (2015)
  - Biology background
  - File formats
  - High Performance Computing
  - Linux
  - Surviving without sudo permissions



# Motivation

A genotyping-by-sequencing story

- My first GBS data (2015)
  - Biology background
  - File formats
  - High Performance Computing
  - Linux
  - Surviving without sudo permissions
  - Many software and parameters
  - Many programming languages
  - Different Operational Systems
  - Updates



# Motivation

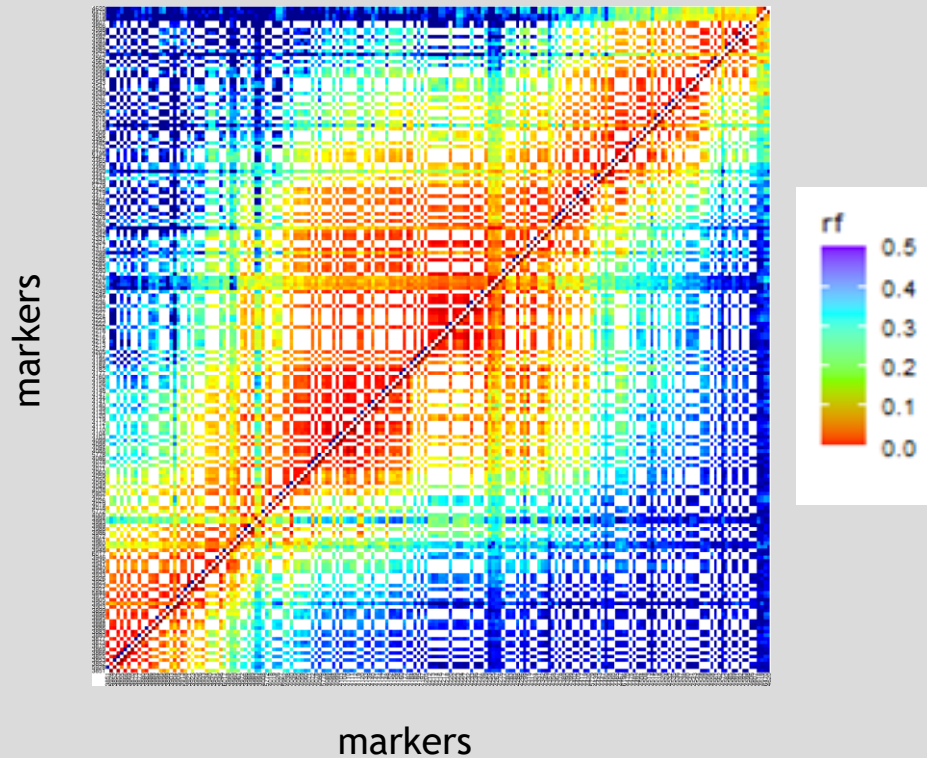
A genotyping-by-sequencing story

- My first diploid outcrossing population linkage map
  - Estimating also phases
  - Hidden Markov Model multipoint approach
  - OneMap
  - R



- Diploid species
- Bi-parental populations
- Backcross, RILs, F2 and outcrossing
- Biallelic and Multiallelic markers

# Recombination fraction matrix



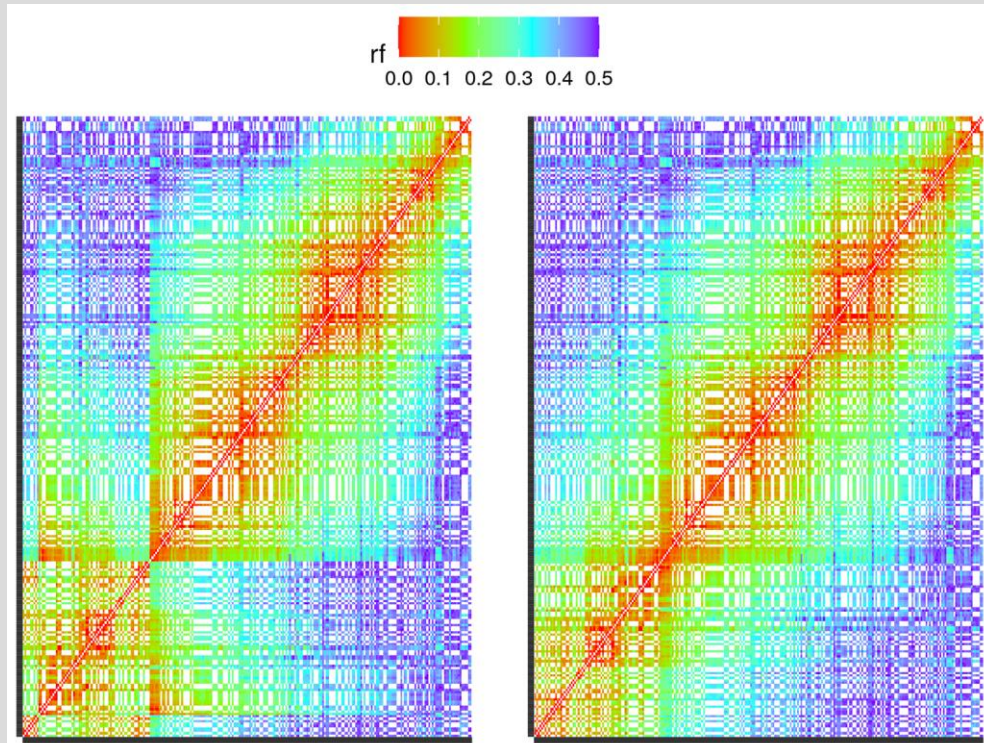
- # recombinations between two markers during meiosis
- Genomic order
  - Example: Eucalyptus chromosome 10

# Recombination fraction matrix

## Inversion

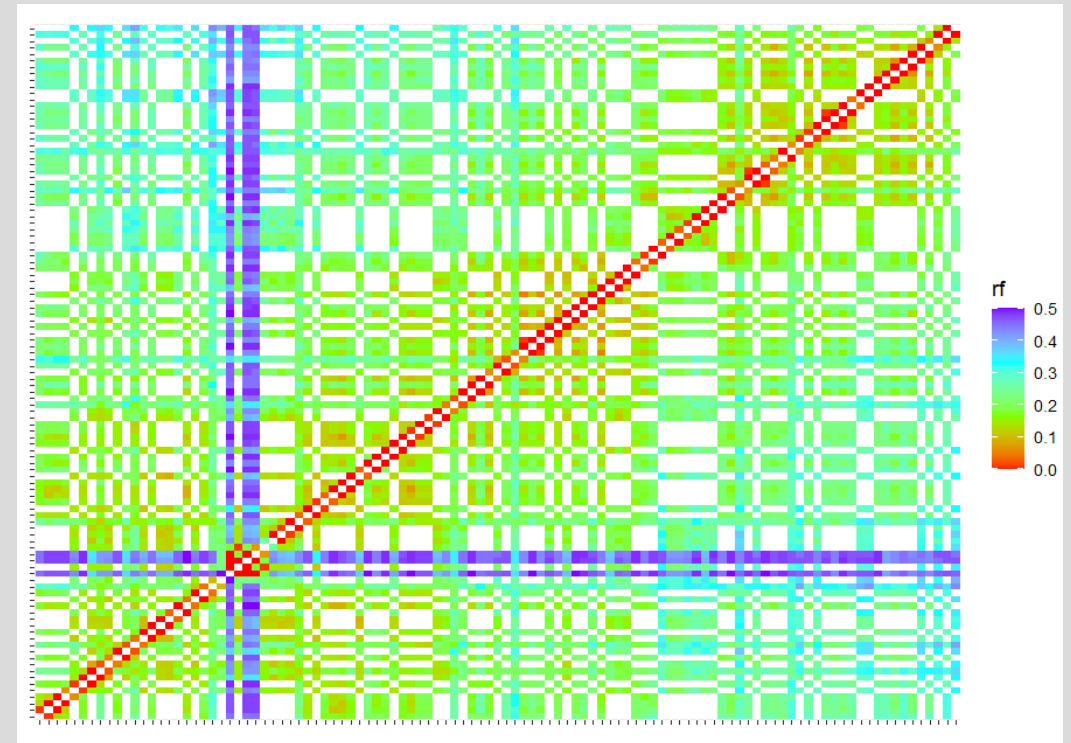
- Aspen chromosome 12

Most likely bad data



Genomic order

Order fixed





# Motivation

## Experiences building Linkage Maps

- Using sequencing markers
  - Recombination fraction matrix not always pretty
  - Slow analysis
  - Inflated map size
- OneMap - Since 2007
- OneMap maintainer - Since 2017



- Updates in version 3.0

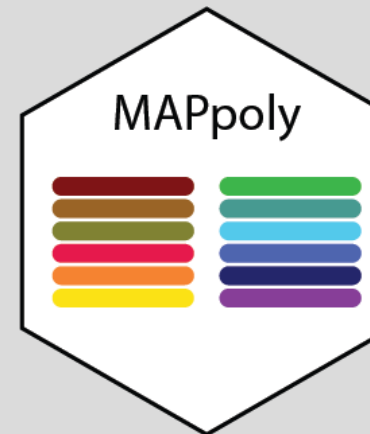
# Motivation

## Experiences building Linkage Maps

- Using sequencing markers
  - Recombination fraction matrix not always pretty
  - Slow analysis
  - Inflated map size
- OneMap - Since 2007
- OneMap maintainer - Since 2017
- MAPpoly – Since 2018



- Updates in version 3.0



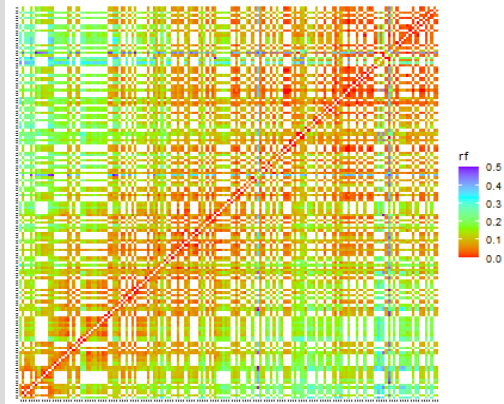
- Diploid and polyploid species
- Bi-parental populations
- Outcrossing
- All dosages markers
- Updates - Marcelo and Gabriel talk

# Diploid roses

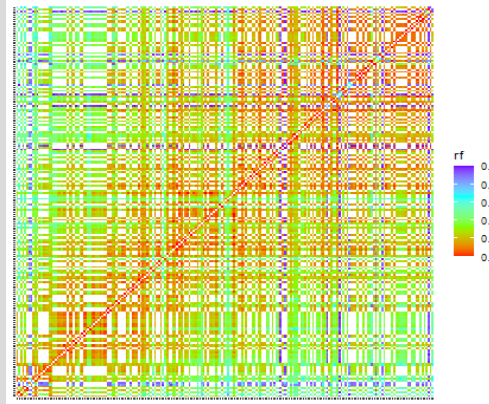
- 37% of chromosome 1
- ~ 38 cM
- Sequencing depth ~ 94X



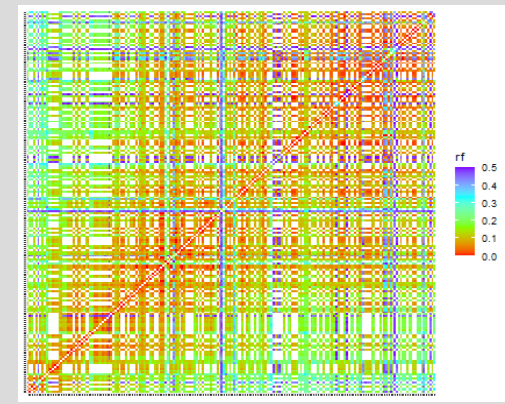
GATK



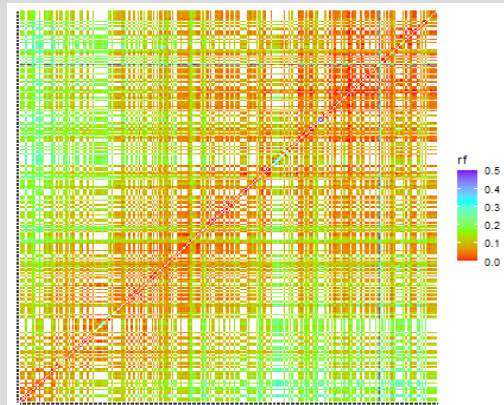
GATK + polyRAD



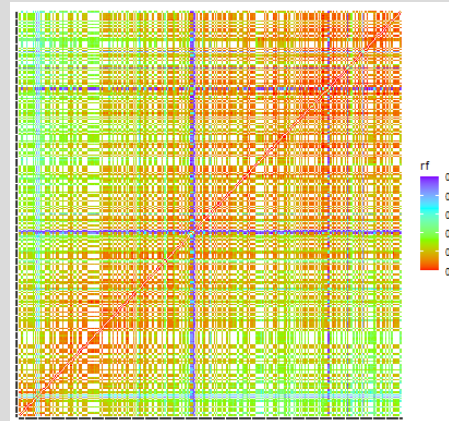
GATK + updog



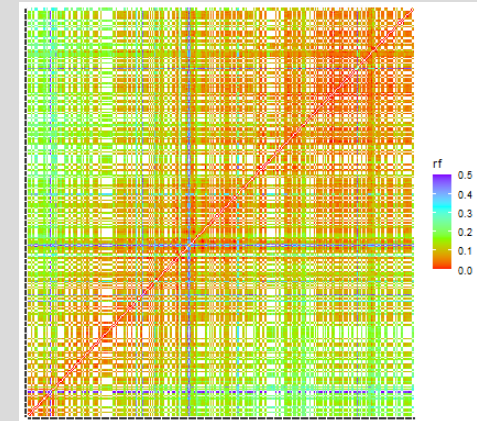
freebayes



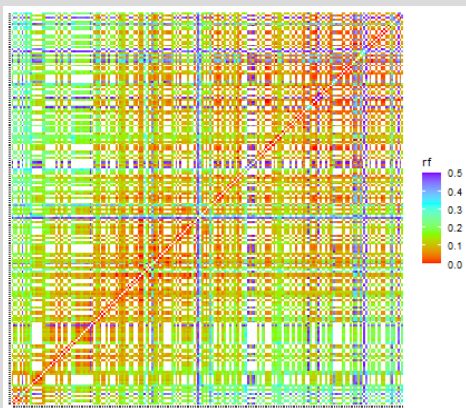
freebayes + polyRAD



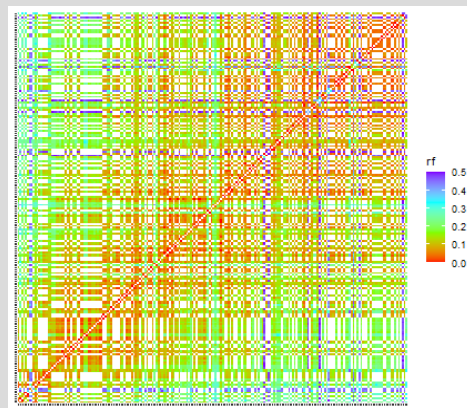
freebayes + updog



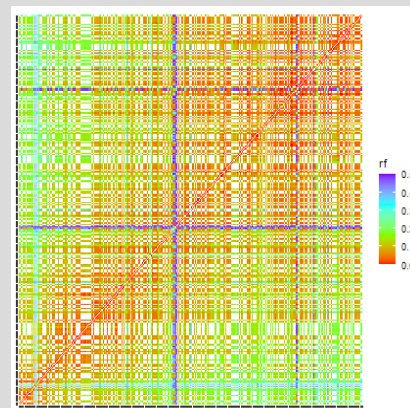
GATK + updog



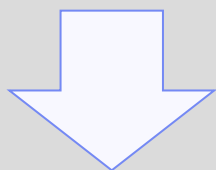
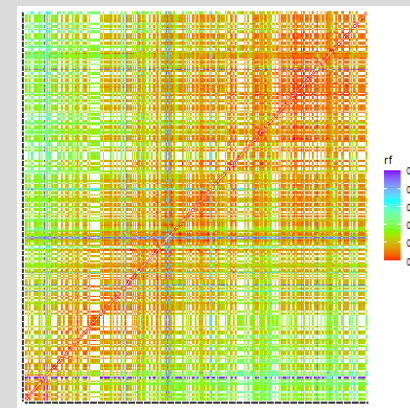
GATK + polyRAD



freebayes + polyRAD



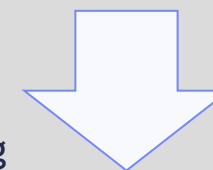
freebayes + updog



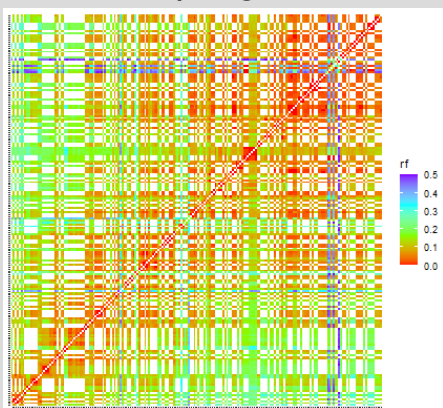
Simulation studies: 10,880 maps  
Empirical studies: 816 maps

Filters:

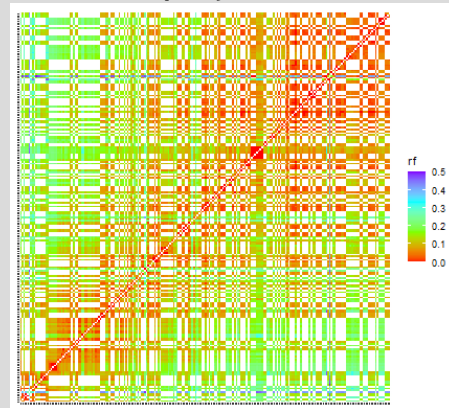
- genotype probabilities
- non-informative markers
- replace AD by missing when GT is missing



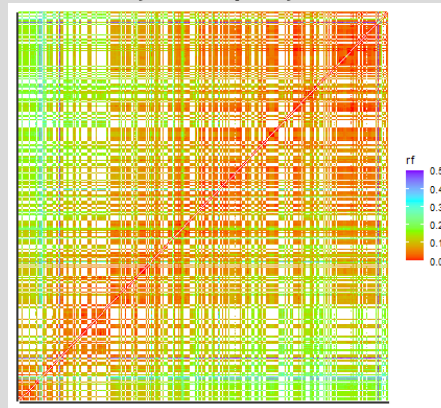
GATK + updog



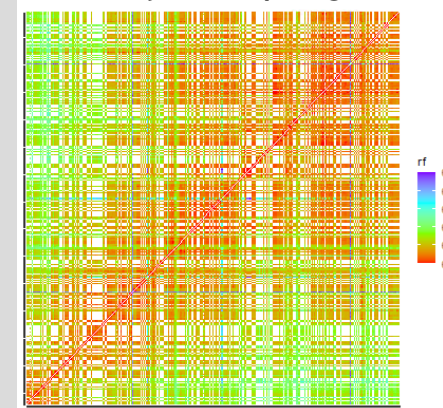
GATK + polyRAD



freebayes + polyRAD

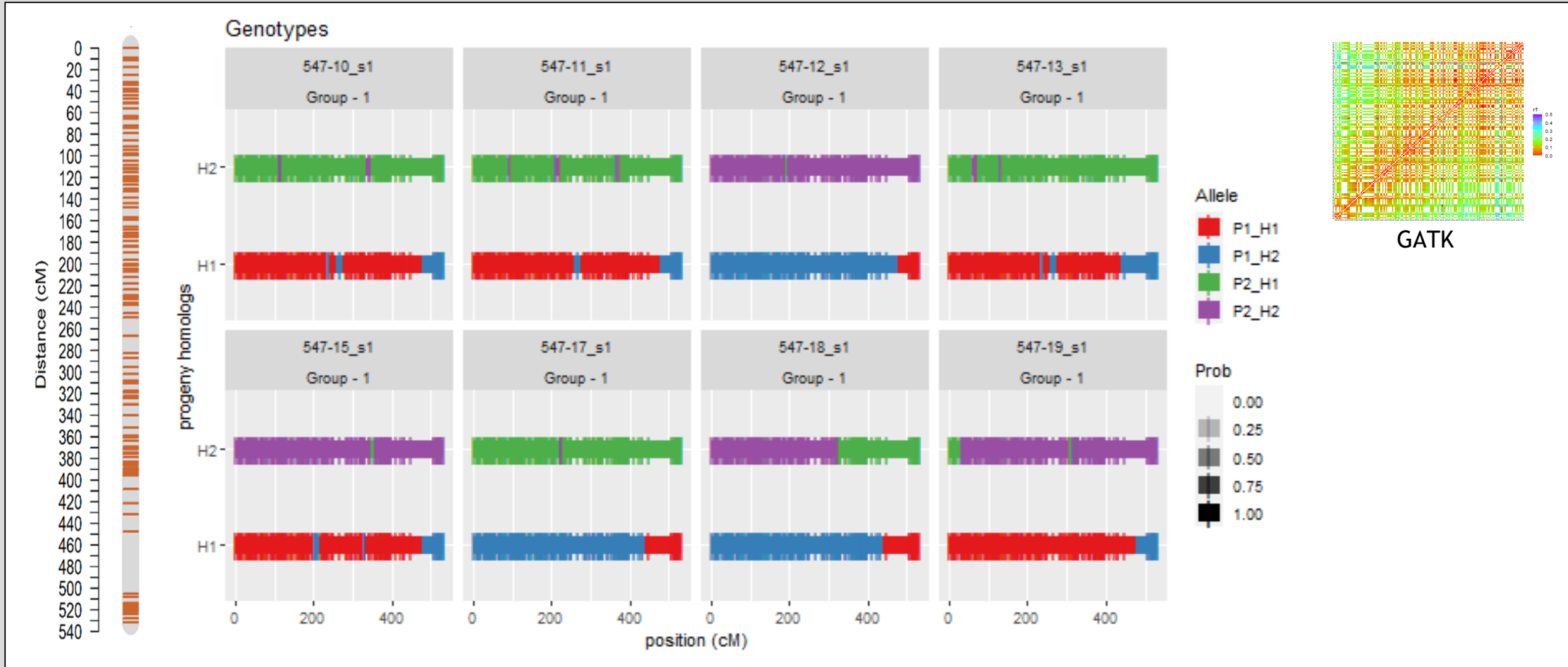


freebayes + updog



# Map size

37% of chromosome 1 ~38cM



# Map size

- Hidden Markov Model Emission Function
  - global error rate
  - genotype probabilities (PL)

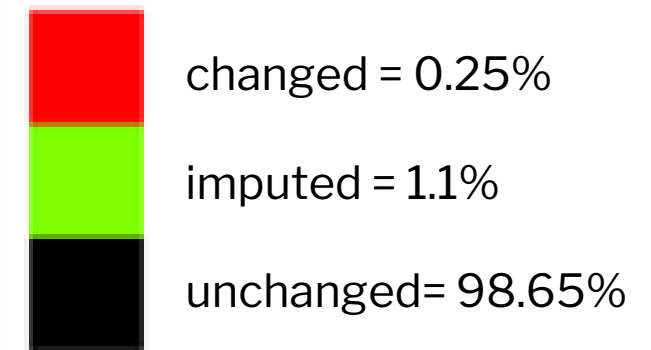
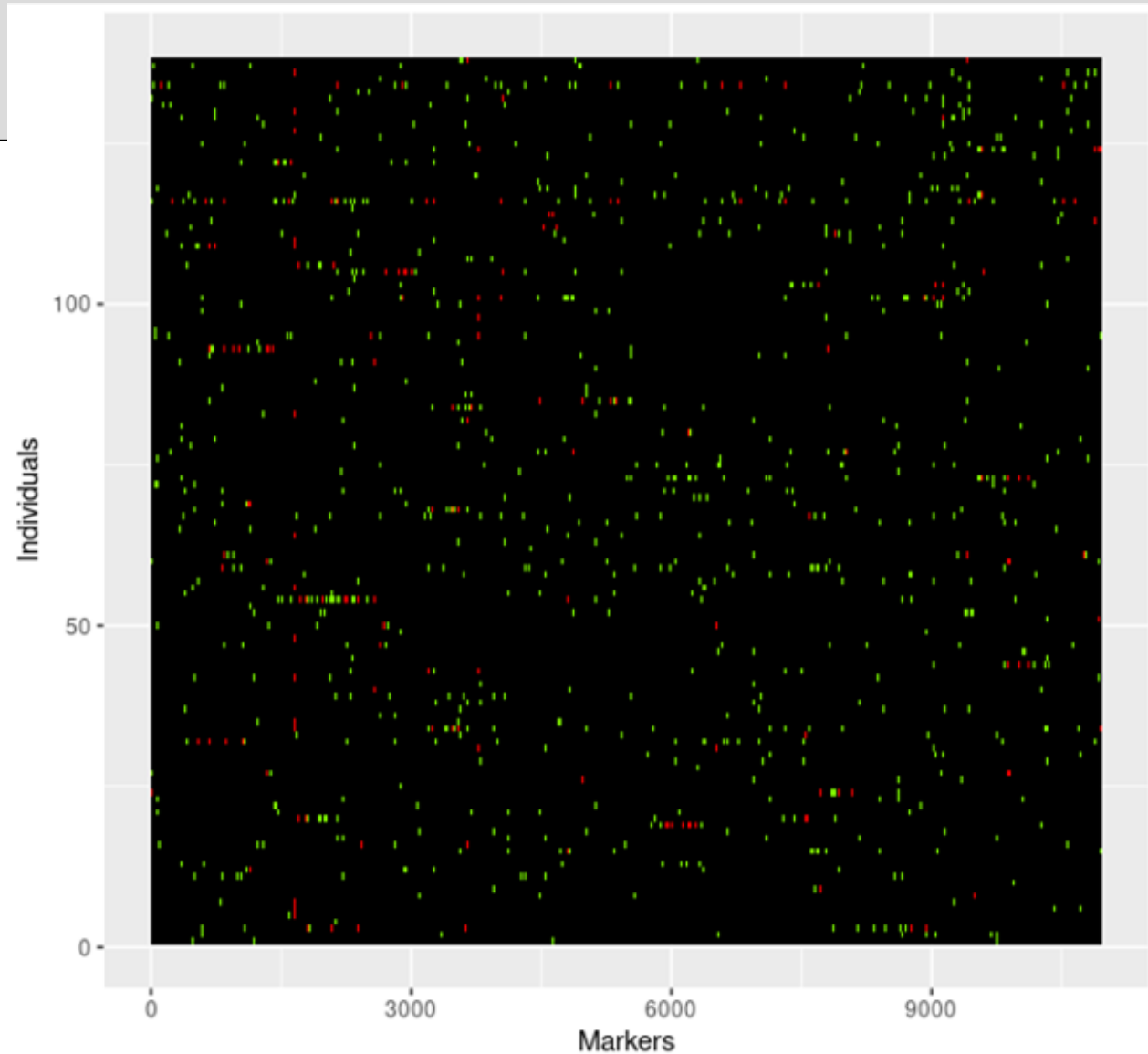


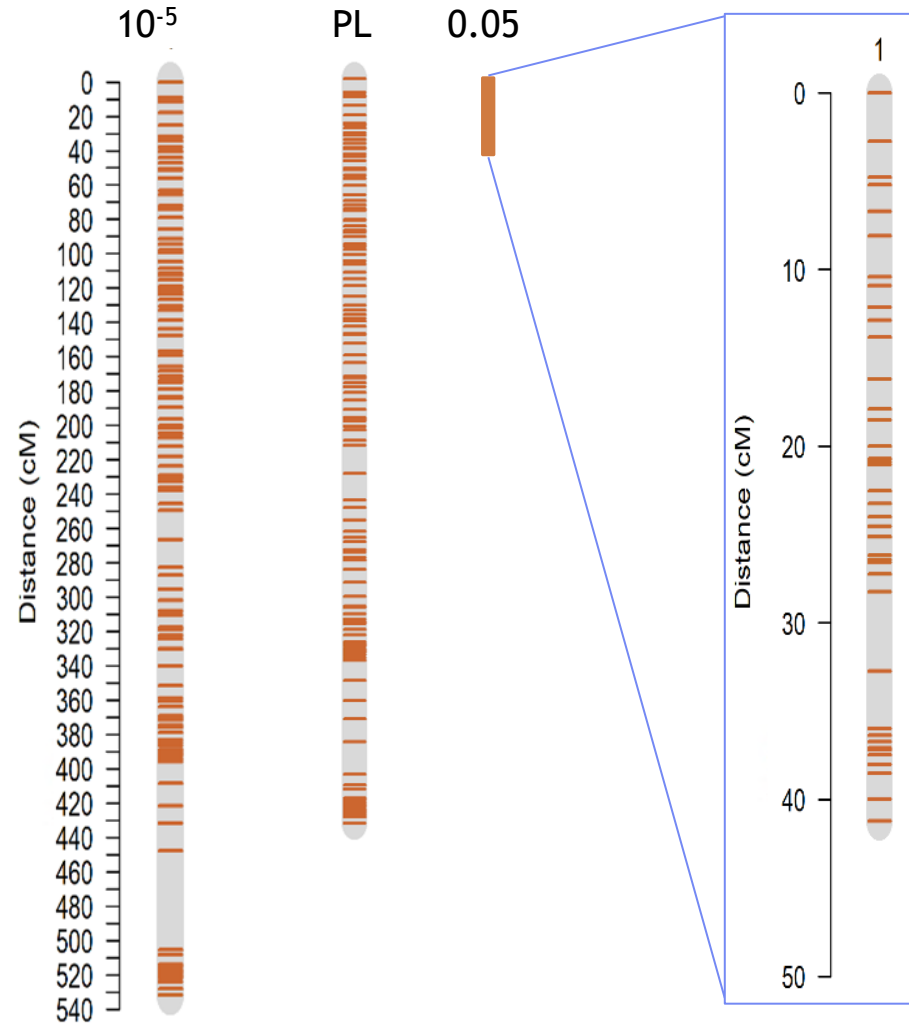
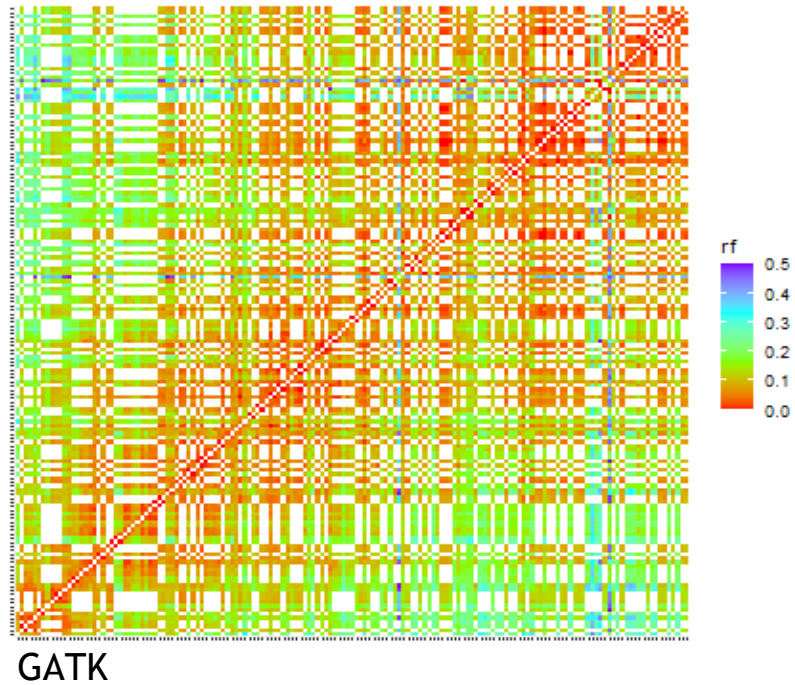
Figure by Jeekin Lau

# Map size

37% of chromosome 1 ~38cM

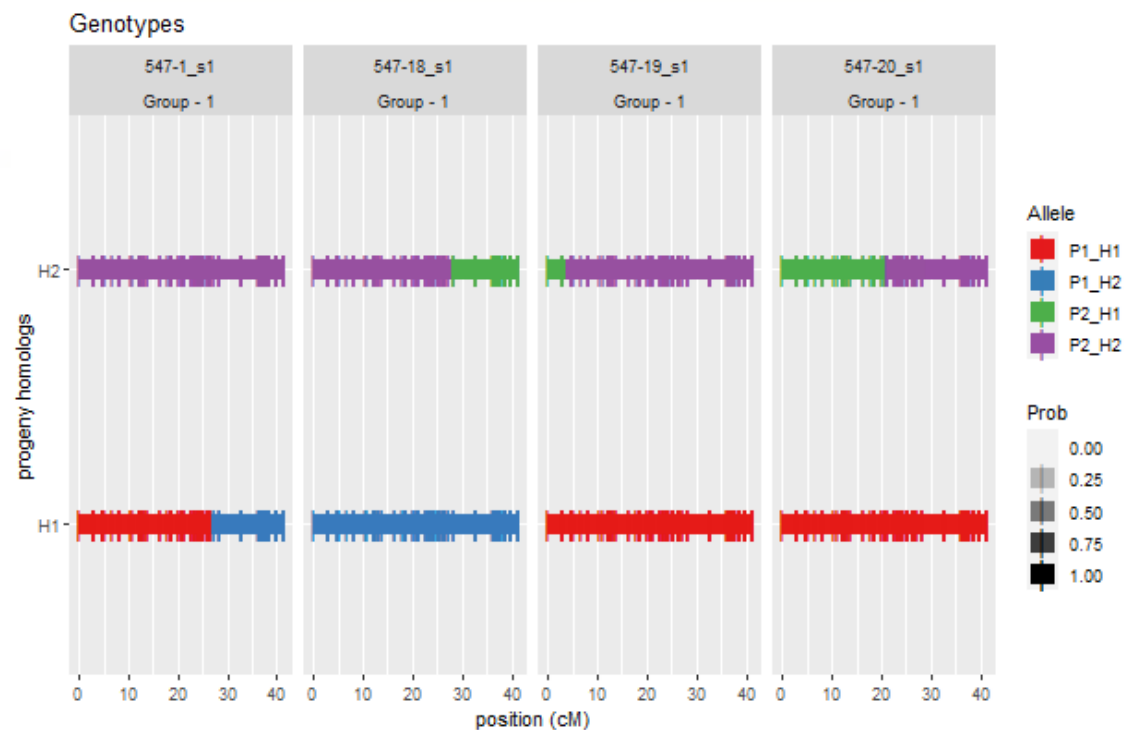
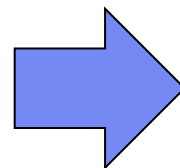
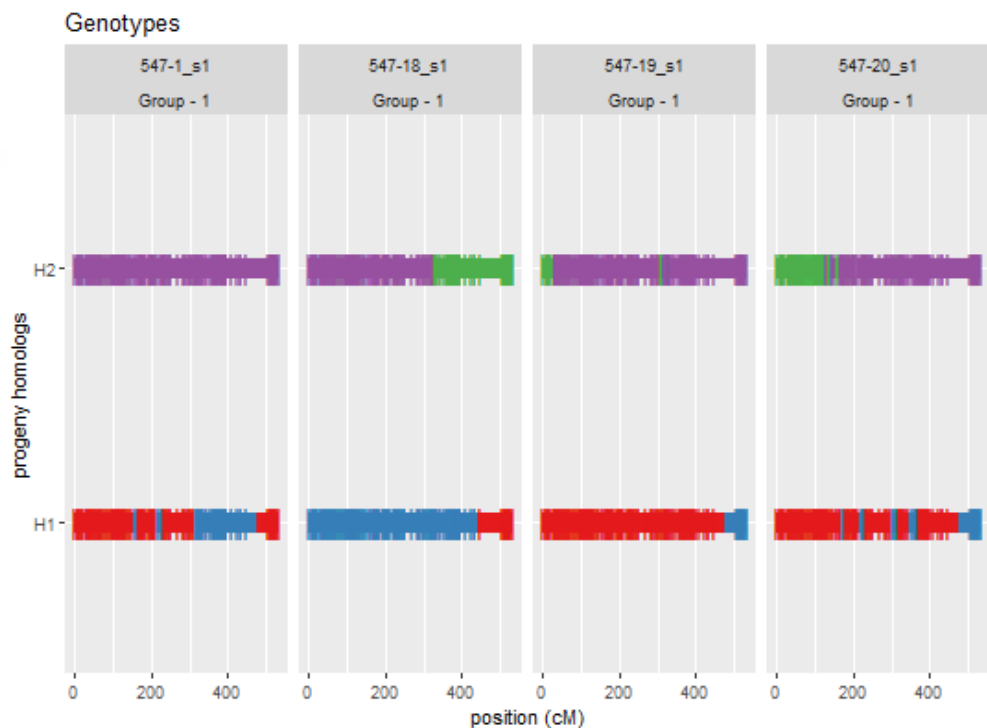
- Hidden Markov Model Emission Function

- global error rate
- genotype probabilities (PL)



# Map size

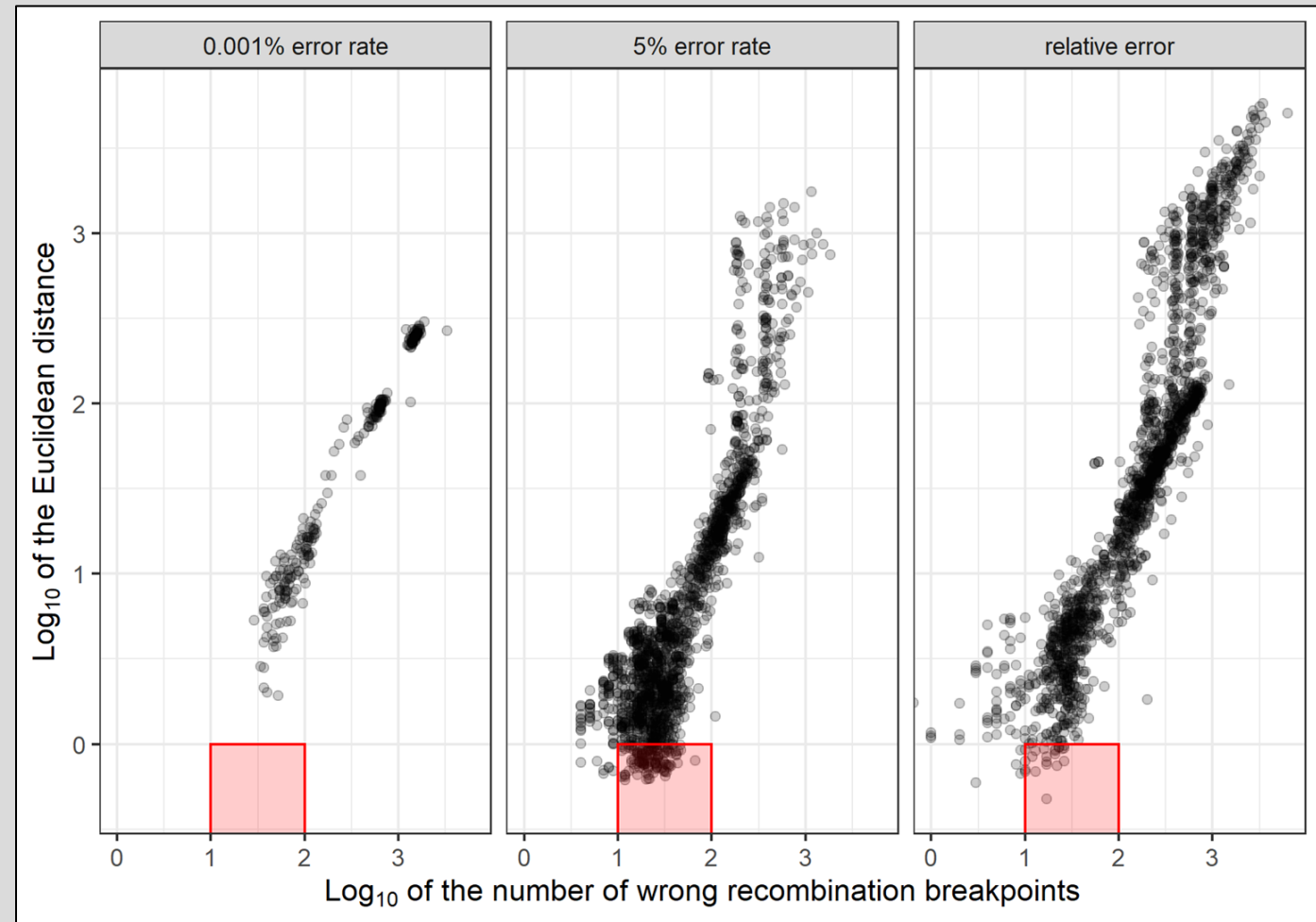
37% of chromosome 1 ~38cM





# Simulation study

- Match recombinations breakpoints
  - Large maps - always bad
  - Small maps - not always good
- Other tested scenarios with:
  - Segregation distortion
  - Contaminants samples
  - Multiallelic markers



Red square: no inflated size (1 or less Euclidean distance) but have from 10 to 100 wrong recombination breakpoints

# Preprint



bioRxiv

THE PREPRINT SERVER FOR BIOLOGY

bioRxiv posts many COVID19-related papers. A reminder: they have not been formally peer-reviewed and should not guide health-related behavior or be reported in the press as conclusive.

New Results

[Follow this preprint](#)

## Developing best practices for genotyping-by-sequencing analysis using linkage maps as benchmarks

Cristiane Hayumi Taniguti, Lucas Mitsuo Taniguti, Rodrigo Rampazo Amadeu, Jeekin Lau, Gabriel de Siqueira Gesteira, Thiago de Paula Oliveira, Getulio Caixeta Ferreira, Guilherme da Silva Pereira, David Byrne, Marcelo Mollinari, Oscar Riera-Lizarazu, Antonio Augusto Franco Garcia

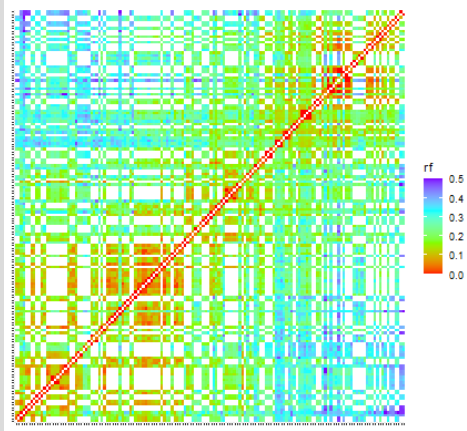
doi: <https://doi.org/10.1101/2022.11.24.517847>

# Diploid Aspen

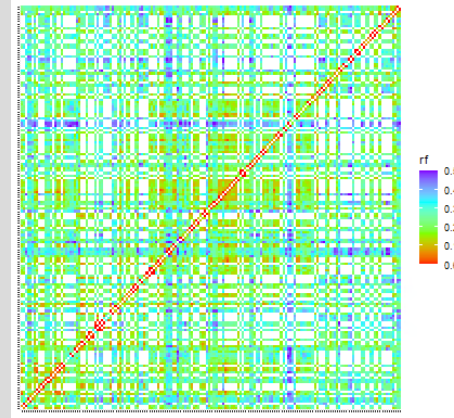
- 37% of chromosome 10
- Sequencing depth ~ 6X



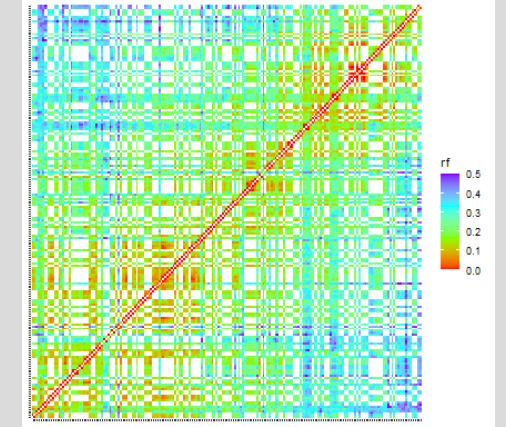
GATK



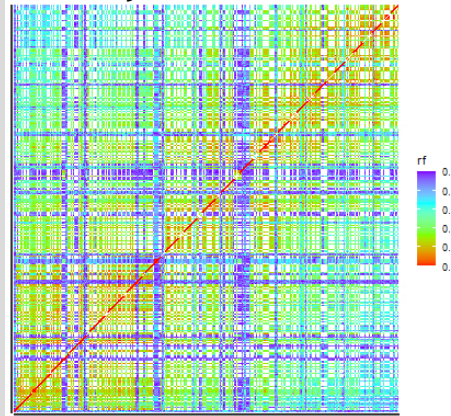
GATK + polyRAD



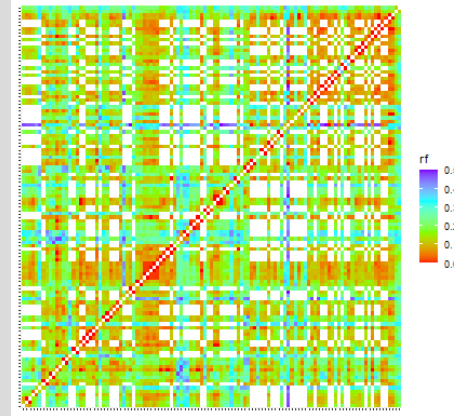
GATK + updog



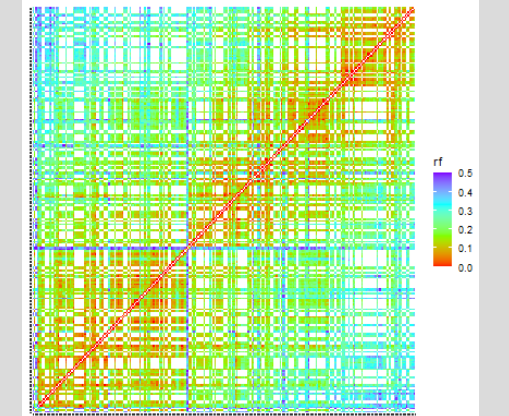
freebayes



freebayes + polyRAD

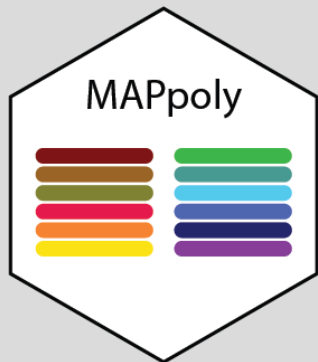


freebayes + updog

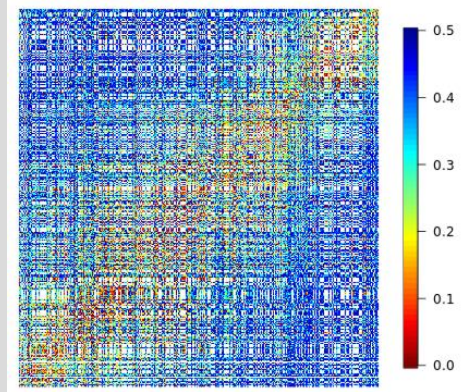


# Tetraploid rose

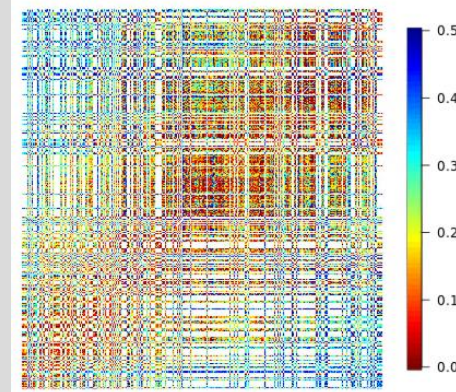
- Chromosome 2
- Sequencing depth ~ 50X



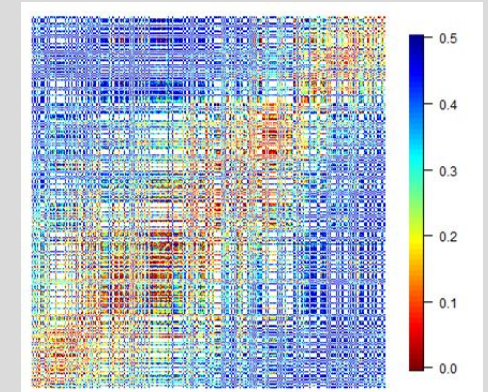
GATK



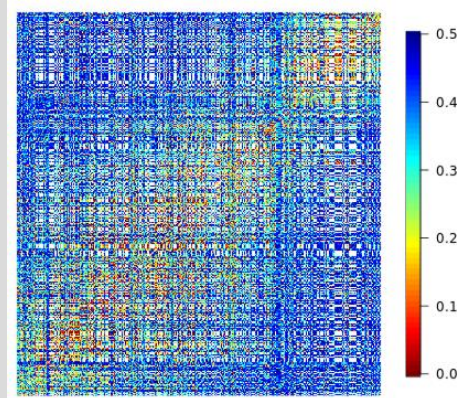
GATK + polyRAD



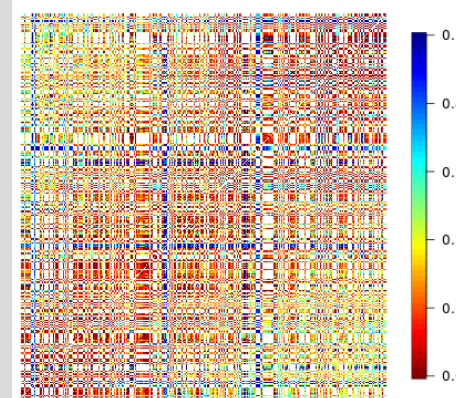
GATK + updog



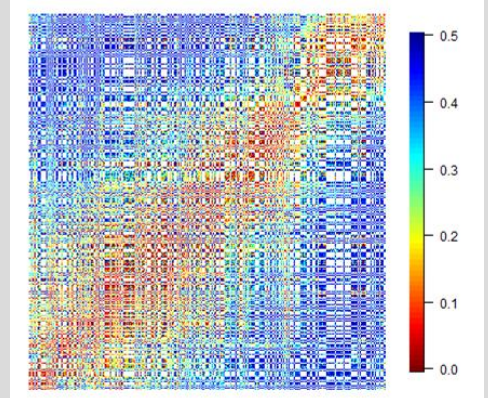
freebayes



freebayes + polyRAD

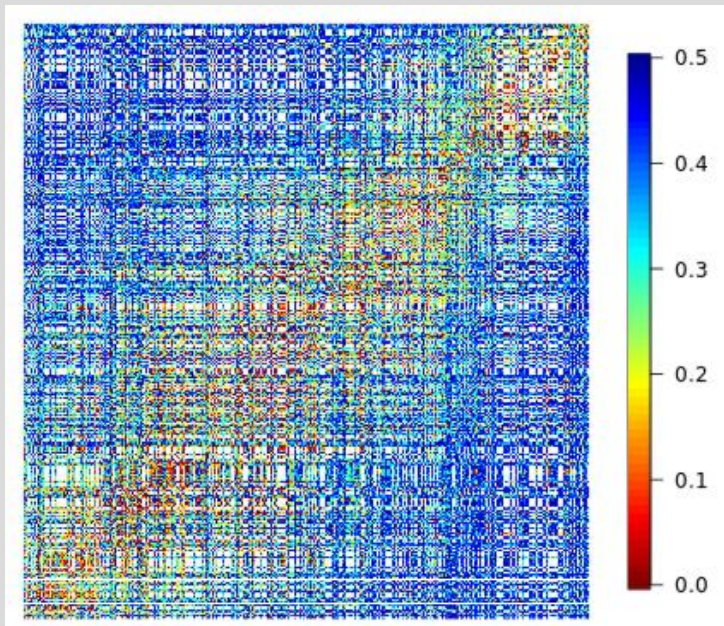


freebayes + updog

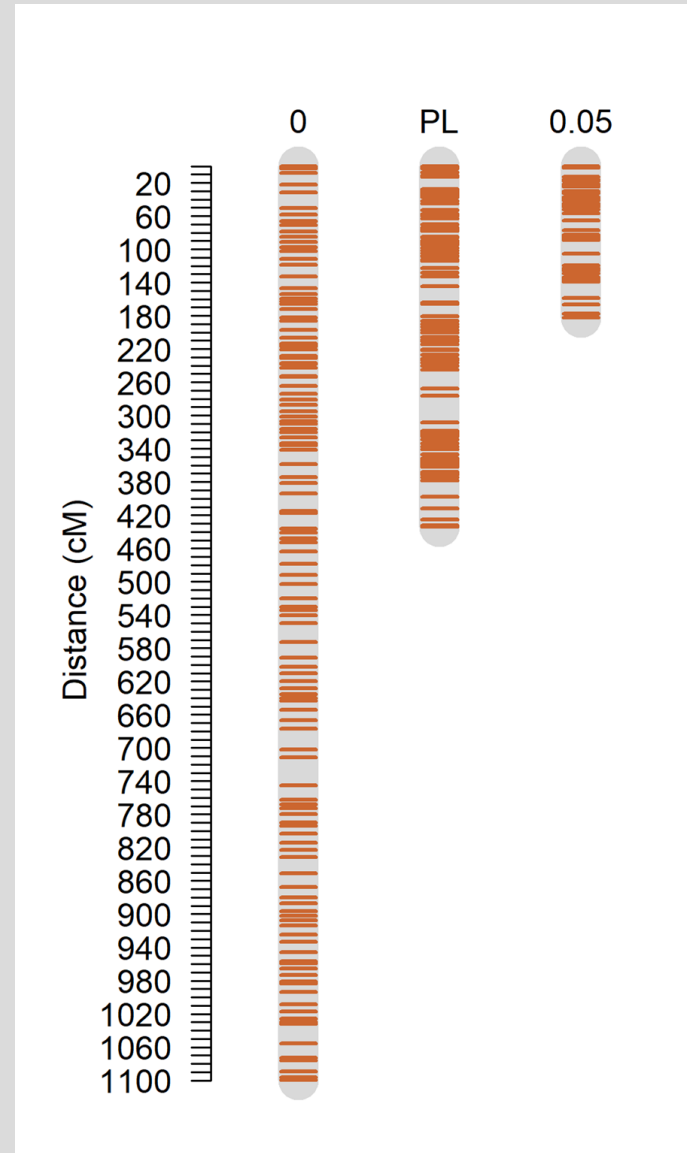


# Map size

- Chromosome 2
- Sequencing depth ~ 50X

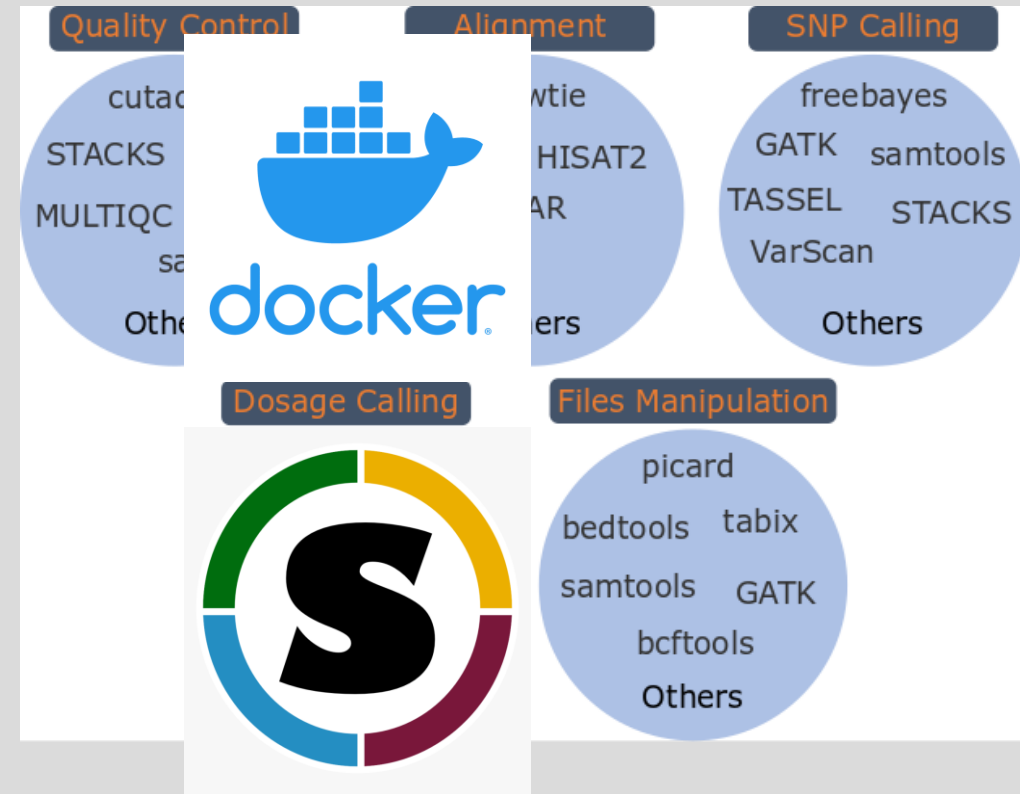


GATK



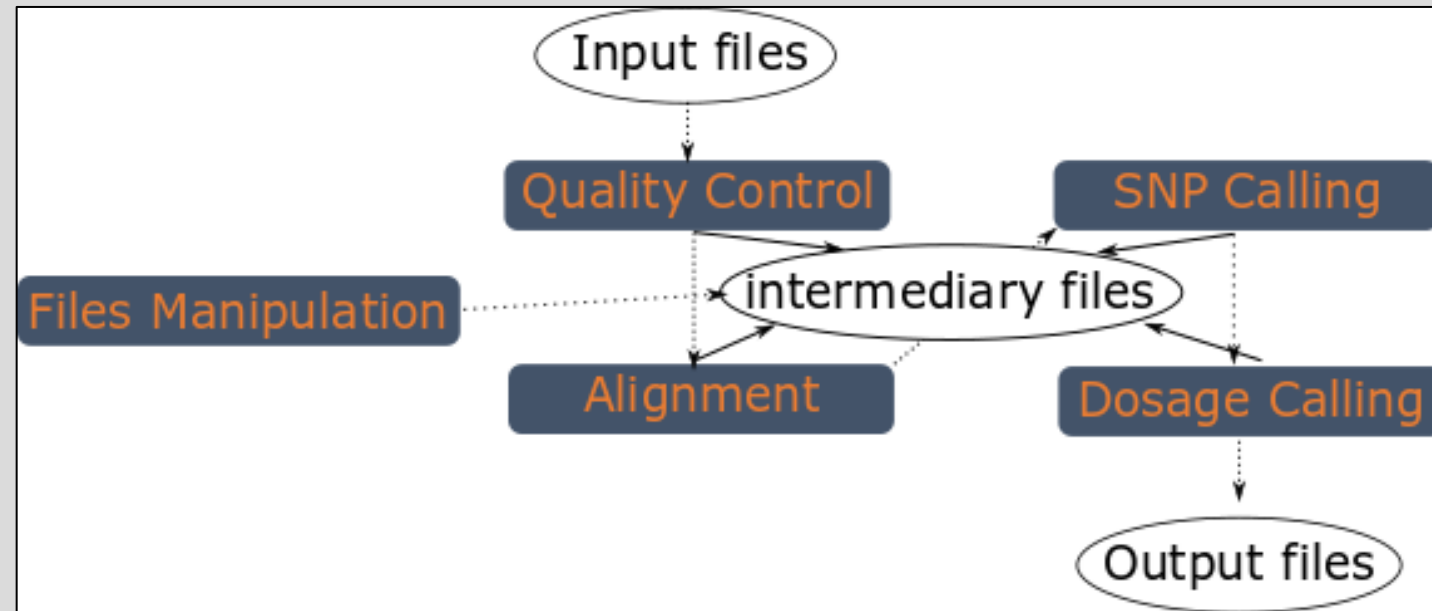
# Overcoming Technical Difficulties

- Large files
  - High Performance Computing (HPC)
  - Management systems (SLURM, SGE)
  - Cloud (Google, Amazon)
- Many software
- Many programming languages
- Different Operational Systems
- Updates
  - Containers
    - Docker
    - Singularity (usually available in HPC)
    - BioContainers



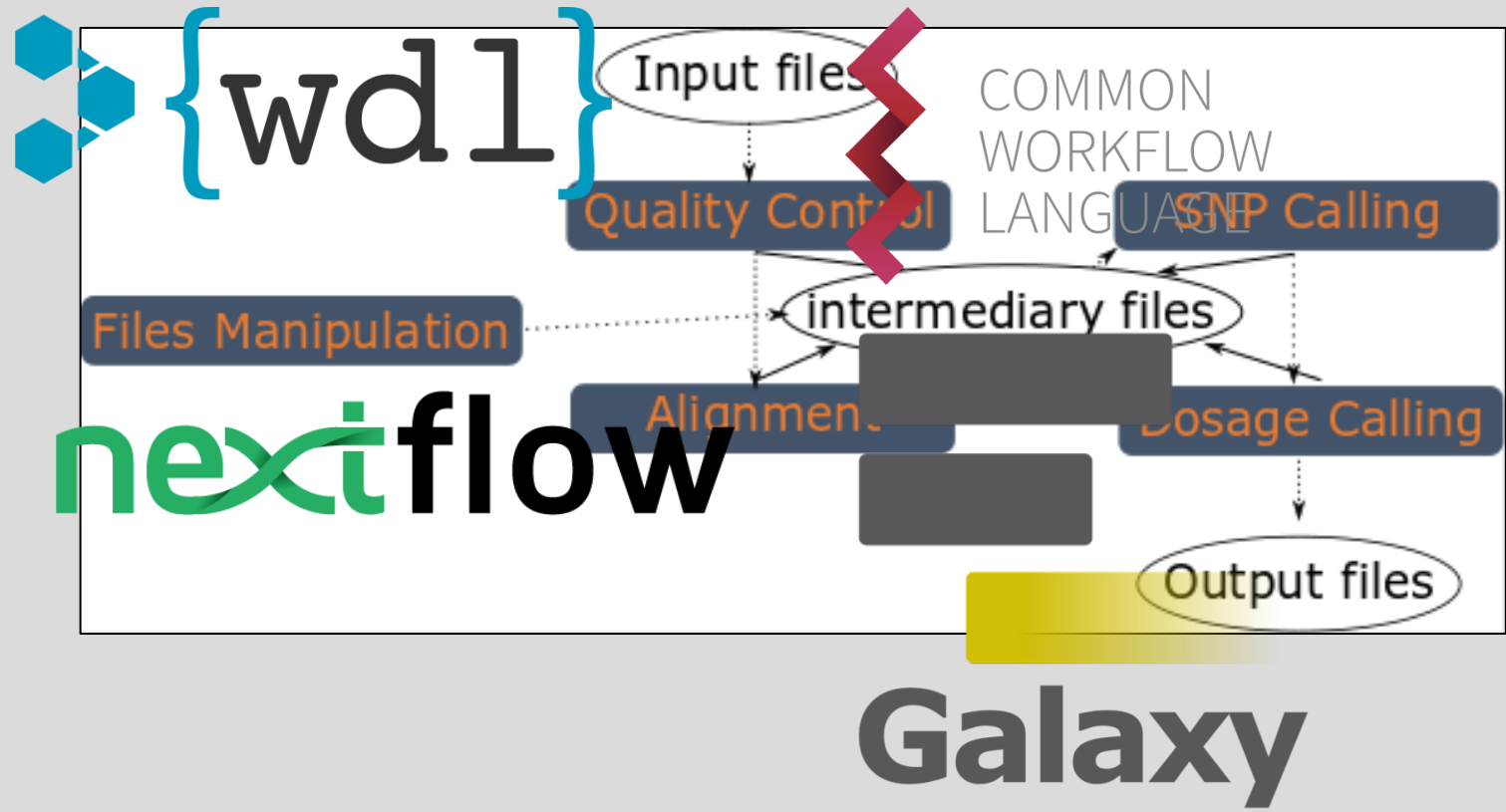
# Overcoming Technical Difficulties

- Many steps
- Many file formats



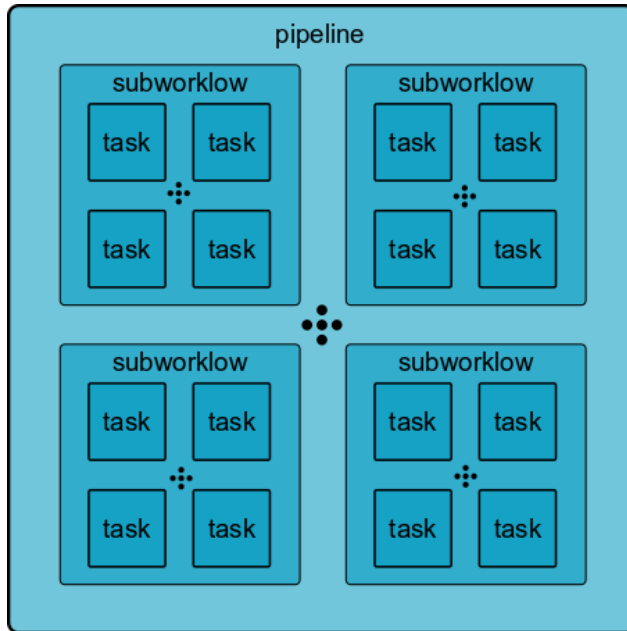
# Overcoming Technical Difficulties

- Many steps
- Many file formats
  - Workflows systems
    - Galaxy; Nextflow; Snakemake; CWL; **WDL**
  - Workflows repositories
    - Dockerstore; WorkflowHub
  - Run workflows on Cloud
    - Galaxy; DNAnexus; Terra; AnVIL; SevenBridges





# Reads2Map



Cristianetaniguti / Reads2Map Public

<> Code Issues Pull requests Actions Projects 1 Wiki Security

main 6 branches 13 tags Go to file

Cristianetaniguti Merge pull request #58 from Cristianetaniguti/config\_do... ✓ acca101 on 1

folder	.circleci	build: correct env var for prod env
folder	.configurations	fix typo
folder	.dockerfiles	add more tests
folder	.scripts	build: configure to also release to main
folder	pipelines	wf specific name
folder	structs	refactor: remove not used struct
folder	subworkflows	freebayes split ok
folder	tasks	joint wf
folder	tests	refactor: move test files to tests module
file	.dockstore.yml	fix path

# Reads2Map

- Cloud environments
  - [terra.bio](#)
- HPC
  - [Cromwell](#)
  - [MiniWDL](#)
  - [dxWDL](#)


inputs.json

```
{
  "SNPCalling.max_cores": 2,
  "SNPCalling.ploidy": 4,
  "SNPCalling.rm_dupli": false,
  "SNPCalling.replaceAD": false,
  "SNPCalling.run_gatk": true,
  "SNPCalling.run_freebayes": true,
  "SNPCalling.hardfilters": true,
  "SNPCalling.n_chrom": 1,
  "SNPCalling.chunk_size": 2,
  "SNPCalling.samples_info": "tests/data/polyploid/fastq/samples_info.txt",
  "SNPCalling.gatk_mchap": false,
  "SNPCalling.references": {
    "ref_fasta": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.fasta",
    "ref_dict": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.dict",
    "ref_ann": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.fasta.ann",
    "ref_sa": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.fasta.sa",
    "ref_amb": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.fasta.amb",
    "ref_pac": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.fasta.pac",
    "ref_bwt": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.fasta.bwt",
    "ref_fasta_index": "tests/data/polyploid/RchinensisV1.0/Chr04_sub.fasta.fai"
  }
}
```











```
$ java -jar /path/to/cromwell.jar run -i EmpiricalSNPCalling/inputs.json EmpiricalSNPCalling.wdl
```












# task

main ▾ Reads2Map / tasks /

 Cristianetaniguti joint wf

..

-  BWA.wdl
-  JointReports.wdl
-  bcftools.wdl
-  chunk\_lists.wdl
-  cutadapt.wdl
-  freebayes.wdl
-  gatk.wdl
-  gusmap.wdl
-  mchap.wdl
-  pedigree\_simulator.wdl

-  gusmap.wdl
-  mchap.wdl
-  pedigree\_simulator.wdl
-  pedigree\_simulator\_utils.wdl
-  pirs.wdl
-  radinitio.wdl
-  simuscop.wdl
-  stacks.wdl
-  utils.wdl
-  utilsR.wdl
-  vcf2diploid.wdl

# task

example: freebayes.wdl

```
task RunFreebayes {  
  > input { ...  
  }  
  
  Int disk_size = ceil(size(reference, "GiB") + size(bam, "GiB") + 50)  
  Int memory_size = ceil(size(bam, "MiB") * 25 * max_cores + 10000)  
  
  > command <<< ...  
  >>>  
  
  > runtime { ...  
  }  
  
  > meta { ...  
  }  
  
  > output { ...  
  }  
}
```

```
input {  
  File reference  
  File reference_idx  
  File bam  
  File bai  
  Int max_cores  
  Int ploidy  
}  
  
Int disk_size = ceil(size(reference, "GiB") + size(bam, "GiB") + 50)  
Int memory_size = ceil(size(bam, "MiB") * 25 * max_cores + 10000)
```

inputs.json

```
{  
  "reference": "tests/data/PtrichocarpaV3.0/Chr10.2M.fa",  
  "reference_idx": "tests/data/PtrichocarpaV3.0/Chr10.2M.fa.fai",  
  "bam": "tests/data/Ptremula_PRJNA395596_subset/merged.bam",  
  "bai": "tests/data/Ptremula_PRJNA395596_subset/merged.bam.bai",  
  "max_cores": 2,  
  "ploidy": 4  
}
```

# task

example: freebayes.wdl

```
task RunFreebayes {
```

```
> input { ...
}

Int disk_size = ceil(size(reference, "GiB") + size(bam, "G:
Int memory_size = ceil(size(bam, "MiB") * 25 * max_cores +
> command <<< ...
>>>

> runtime { ...
}

> meta { ...
}

> output { ...
}
}
```

```
command <<<
```

```
ln -s ~{bam} .
ln -s ~{bai} .
```

```
freebayes-parallel <(fasta_generate_regions.py ~{reference_idx} 100000) ~{max_cores} \
--genotype-qualities --ploidy ~{ploidy} -f ~{reference} *bam > "freebayes.vcf"
```

```
>>>
```

# task

example: freebayes.wdl

```
task RunFreebayes {  
  > input { ...  
  }  
  
  Int disk_size = ceil(size(reference, "GiB") + size(bam, "GiB") + 50)  
  Int memory_size = ceil(size(bam, "MiB") * 25 * max_cores + 10000)  
  
  > command <<< ...  
  >>>  
  
  > runtime { ...  
  }  
  
  > meta { ...  
  }  
  
  > output { ...  
  }  
}
```

```
runtime {  
  docker: "cristaniguti/freebayes:0.0.1"  
  cpu: max_cores  
  # Cloud  
  memory: "~{memory_size} MiB"  
  disks: "local-disk " + disk_size + " HDD"  
  # Slurm  
  job_name: "RunFreebayes"  
  mem: "~{memory_size}M"  
  time: "48:00:00"  
}
```

# task

example: freebayes.wdl

```
task RunFreebayes {  
  > input { ...  
  }  
  
  Int disk_size = ceil(size(reference, "GiB") + size(bam, "GiB") + 50)  
  Int memory_size = ceil(size(bam, "MiB") * 25 * max_cores + 10000)  
  
  > command <<< ...  
  >>>  
  
  > runtime { ...  
  }  
  
  > meta { ...  
  }  
  
  > output { ...  
  }  
}
```

```
meta {  
  author: "Cristiane Taniguti"  
  email: "chtaniguti@tamu.edu"  
  description: "Split genomic regions and runs [freebayes](https://github.com/freebayes/freebayes) parallelized."  
}
```

# task

example: freebayes.wdl

```
task RunFreebayes {  
  > input { ...  
  }  
  
  Int disk_size = ceil(size(reference, "GiB") + size(bam, "GiB") + 50)  
  Int memory_size = ceil(size(bam, "MiB") * 25 * max_cores + 10000)  
  
  > command <<< ...  
  >>>  
  
  > runtime { ...  
  }  
  
  > meta { ...  
  }  
  
  > output { ...  
  }  
}
```

inputs.json

```
{  
  "reference": "tests/data/PtrichocarpaV3.0/Chr10.2M.fa",  
  "reference_idx": "tests/data/PtrichocarpaV3.0/Chr10.2M.fa.fai",  
  "bam": "tests/data/Ptremula_PRJNA395596_subset/merged.bam",  
  "bai": "tests/data/Ptremula_PRJNA395596_subset/merged.bam.bai",  
  "max_cores": 2,  
  "ploidy": 4  
}
```

```
output {  
  File vcf = "freebayes.vcf"  
}
```

```
miniwdl run --task RunFreebayes -i tests/tasks/freebayes/inputs.json tasks/freebayes.wdl
```



# subworkflow example

```
workflow FreebayesGenotyping {
>   input { ...
>   }

>   call chunk_lists.CreateChunksBamByChr { ...
>   }

   scatter (chunk in zip(CreateChunksBamByChr.bams_chunks, CreateChunksBamByChr.bais_chunks)) {
>     call freebayes.RunFreebayes { ...
>     }
>   }

>   call utils.mergeVCFs { ...
>   }

>   call norm_filt.Normalization { ...
>   }

   Map[String, Array[File]] map_bams = {"bam": CreateChunksBamByChr.bams_chunks, "bai": CreateChunk
>   if(replaceAD){ ...
>   }

   Array[File] freebayes_vcfs = select_all([Normalization.vcf_norm, ReplaceAD.bam_vcf])
   Array[String] freebayes_software = select_all([Normalization.software, ReplaceAD.software])
   Array[String] freebayes_counts_source = select_all([Normalization.source, ReplaceAD.source])

>   output { ...
>   }
}
```

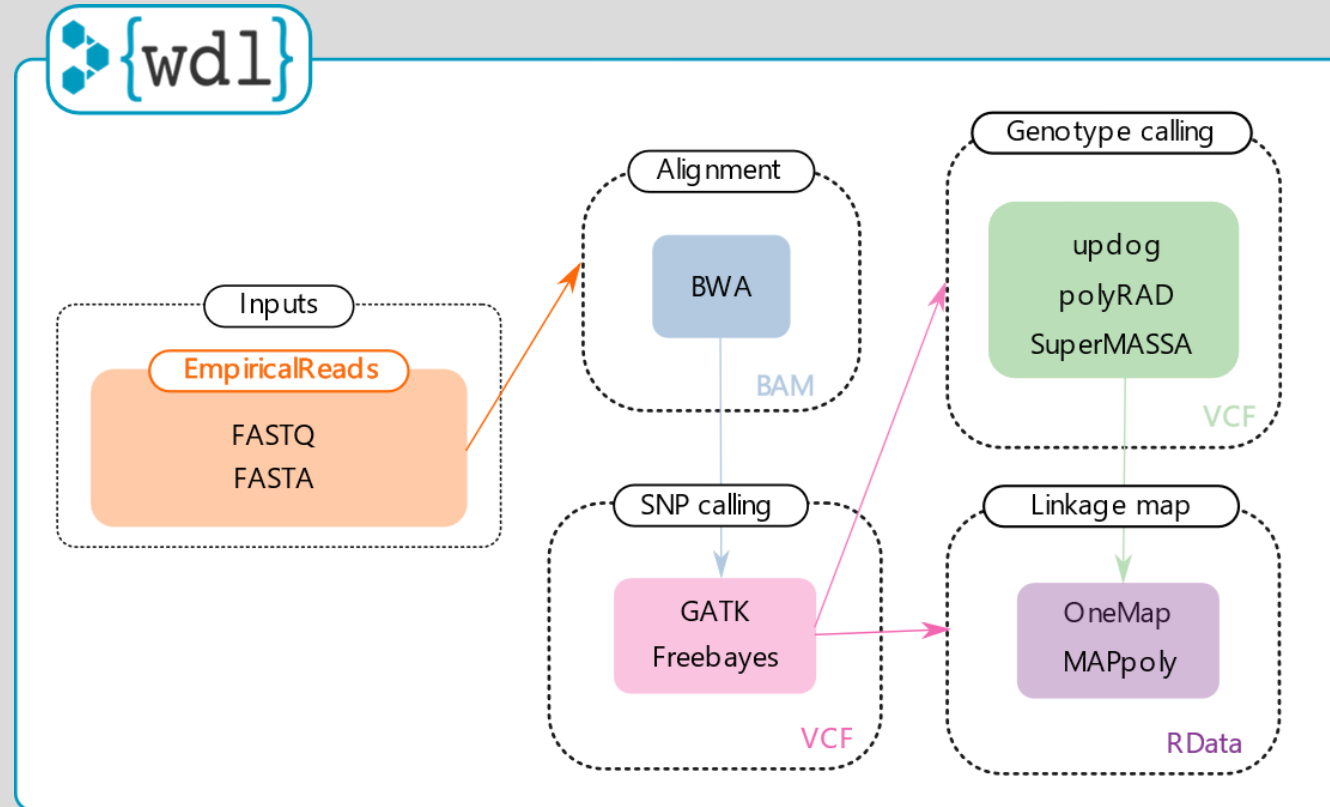
```
$ java -jar /path/to/cromwell.jar run -i freebayes_genotyping/inputs.json freebayes_genotyping.wdl
```

# pipeline example

```
workflow SNPCalling {  
  > input { ...  
  }  
  > call fam.CreateAlignmentFromFamilies { ...  
  }  
  if(run_gatk){  
  >   call gatk.GatkGenotyping { ...  
  }  
  }  
  if(run_freebayes){  
  >   call freebayes.FreebayesGenotyping { ...  
  }  
  }  
  Array[Array[File]] vcfs_sele = select_all([GatkGenotyping.vcfs, FreebayesGenotyping.vcfs])  
  Array[Array[String]] software_sele = select_all([GatkGenotyping.vcfs_software, FreebayesGenotyping.vcfs_software])  
  Array[Array[String]] source_sele = select_all([GatkGenotyping.vcfs_counts_source, FreebayesGenotyping.vcfs_counts_source])  
  > output { ...  
  }  
}
```

```
$ java -jar /path/to/cromwell.jar run -i EmpiricalSNPCalling/inputs.json EmpiricalSNPCalling.wdl
```

# Reads2Map



Available in [Github](#), [Dockerstore](#) and [WorkflowHub](#)

# Summary

- Linkage map quality is very sensitive to upstream choices
- There is no single suggestion for every data set
- Reads2Map uses software's default/tutorials parameters
- Developers, sorry if I used any of the parameters wrongly (let me know)
- Easy to test updates or modification of parameters
- Reads2Map is an organized structure for reproducibility and scalable production

# Thank you



Oscar Riera-Lizarazu  
David Byrne  
Jeekin Lau  
Tessa Hochhaus



Augusto Garcia  
Rodrigo Amadeu  
Getulio Caixeta



Marcelo Mollinari  
Gabriel Gesteira



Lucas Taniguti



Guilherme Pereira



Thiago Oliveira

# Project Members



This project is funded by USDA NIFA Specialty Crop Research Initiative  
Award # 2020-51181-32156 (09/01/20 - 08/31/24)



# Other Collaborators



Neuhouse  
Farms



Wolf Roses  
L.L.C.



This project is funded by USDA NIFA Specialty Crop Research Initiative  
Award # 2020-51181-32156 (09/01/20 - 08/31/24)

