Introduction to Hail

Cotton Seed, Technical Lead
Tim Poterba, Software Engineer
Hail Team, Neale Lab
Broad Institute and MGH
Why Hail?

- Genetic data is becoming absolutely massive
Broad Genomics, by the numbers

- Number of Genomes:
  - 2009: 0
  - 2010: 0
  - 2011: 22,000
  - 2012: 44,000
  - 2013: 66,000
  - 2014: 63,703

- Number of Exomes:
  - 2009: 0
  - 2010: 0
  - 2011: 0
  - 2012: 0
  - 2013: 0
  - 2014: 84,000
  - 2015: 168,000
  - 2016: 250,771

Graphs showing the increase in the number of genomes and exomes from 2009 to 2016.
Why Hail?

• Genetic data is becoming absolutely massive

• gnomAD: 123K exomes, 15K WGS, 40TB compressed VCF

• UKBB: 500K samples impute 40M variants, 10s of TB BGEN
Why Hail?

- Genetic data is becoming absolutely massive
  - gnomAD: 123K exomes, 15K WGS, 40TB compressed VCF
  - UKBB: 500K samples impute 40M variants, 10s of TB BGEN
  - Power is proportional to $Np(1 - p)$
  - Need massive data, knowledge about the genome, functional annotation, reference datasets, burden methods, etc. to detect association
What is Hail?

Hail is a **scalable** tool for doing **data science** on **genetic data**.
What is Hail?

Hail is a **scalable** tool for doing **data science** on **genetic data**.

- **Scalable**:
  - Add more CPUs, get your answer faster.
  - Add more resources, compute on bigger data.
What is Hail?

Hail is a **scalable** tool for doing **data science** on **genetic data**.

- **Scalable:**
  - From 1 core (laptop) to 10,000 core clusters
  - Use for QC, analysis of gnomAD (20K WGS, 200K exomes), 40TB compressed VCF
What is Hail?

Hail is a **scalable** tool for for doing **data science** on **genetic data**.

![Diagram: Data Science Process]

- **Import** → **Tidy** → **Transform** → **Visualize** → **Model** → **Communicate** → **Understand**

Program
What is Hail?

Hail is a **scalable** tool for
for doing **data science**
on **genetic data**.

**No reads.**
Functionality

**Import/Export**
- VDS
- VCF
- GEN
- BGEN
- PLINK
- TSV
- UCSC BED
- Interval List
- FAM
- synthetic
- JSON
- Python

**Transform**
- Query
- Filter
- Aggregate
- Join/Annotate

**Analyze**
- Concordance
- Fisher Exact Test
- GRM
- IBD
- Impute sex
- Mendel errors
- PCA
- Regressions:
  - linear
  - logistic
  - linear-mixed
- TDT
- QC stats
Architecture

• Interface is Python

• Python functions in turn use Hail expression language

• Two languages! This is the most confusing part.

• Built on Spark, distributed computing framework

• Hail users don’t need to know Spark (but it can be useful... )
hail Architecture

Data shuffling across machines (wide dependencies)
Where can you run Hail?

- Single computer: laptop to big server
- On the cloud: Google and Amazon clouds both have products that can run Hail
- To use multiple machines in HPC cluster you probably need help from your local sysadmin.
Help!

• Extensive documentation: https://hail.is

• Another tutorial! https://hail.is/hail/tutorial.html

• Live chat: https://gitter.im/hail-is/hail

• Discussion forum: http://discuss.hail.is/

• Updates: http://discuss.hail.is/c/updates
Read the docs!

• We’ve worked hard to make them not suck.

• Liberal links in the practicals to the documentation. Explore!
Caveats

- Hail is powerful but complicated.
hail Caveats

- Interface is beta
  - Interface changing (improving!) often
  - Moving towards versioned release next few months
- Does not support all VCF features
  - Fixed genotype schema GT:AD:DP:GQ:PL/GP, diploid genotypes only (but support for sex chromosomes), no phasing, no symbolic alleles, no CNVs, no gVCF support.
- GRCh37 hardcoded.
Main Python objects

- **HailContext**: main entry point for Hail functionality
- **VariantDataset**: Hail’s representation of a dataset
- **KeyTable**: Table-like structure (think data frame)
HailContext

• Main entry point for Hail functionality

• Created once at the beginning of a Hail session or script:

```python
import hail
hc = hail.HailContext()
```

• Calling functions on hc is you how access Hail functionality
Example

In [1]: import hail

In [2]: hc = hail.HailContext()

In [3]: (hc.import_vcf('hail-practical/sample.vcf')
   ...:  .count(genotypes=True))

Out[3]:
{u'callRate': 97.45664739884393,
  u'nCalled': 33720L,
  u'nGenotypes': 34600L,
  u'nSamples': 100,
  u'nVariants': 346L}
Example

In [1]: import hail

In [2]: hc = hail.HailContext()

In [3]: (hc.import_vcf('sample.vcf')
  ...:  .count(genotypes=True))

Out[3]:
{u'callRate': 97.45664739884393,
 u'nCalled': 33720L,
 u'nGenotypes': 34600L,
 u'nSamples': 100,
 u'nVariants': 346L}
In [1]: import hail

In [2]: hc = hail.HailContext()

In [3]: (hc.import_vcf('sample.vcf')
   ...:   .count(genotypes=True))
Out[3]:
{u'callRate': 97.45664739884393,
 u'nCalled': 33720L,
 u'nGenotypes': 34600L,
 u'nSamples': 100,
 u'nVariants': 346L}
Example

In [1]: import hail

In [2]: hc = hail.HailContext()

In [3]: (hc.import_vcf('sample.vcf')
    ...:    .count(genotypes=True))
Out[3]:
{u'callRate': 97.45664739884393,
  u'nCalled': 33720L,
  u'nGenotypes': 34600L,
  u'nSamples': 100,
  u'nVariants': 346L}
Example 2

In [4]: (hc.import_vcf('hail-practical/sample.vcf'))
   ...:  .filter_genotypes('g.gq > 20')
   ...:  .count(genotypes=True))
Out[4]:
{u'callRate': 89.09537572254335,
   u'nCalled': 30827L,
   u'nGenotypes': 34600L,
   u'nSamples': 100,
   u'nVariants': 346L}
Example 2

In [4]: (hc.import_vcf('hail-practical/sample.vcf')
   ...:   .filter_genotypes(["g.gq > 20"])
   ...:   .count(genotypes=True))
Out[4]:
{u'callRate': 89.09537572254335,
 u'nCalled': 30827L,
 u'nGenotypes': 34600L,
 u'nSamples': 100,
 u'nVariants': 346L}
Example 2

In [4]: (hc.import_vcf('hail-practical/sample.vcf')
    ...:  .filter_genotypes('g.gq > 20')
    ...:  .count(genotypes=True))
Out[4]:
{u'callRate': 89.09537572254335,
 u'nCalled': 30827L,
 u'nGenotypes': 34600L,
 u'nSamples': 100,
 u'nVariants': 346L}
Types

• The Hail expression language is **typed**.

• What is the type of 3?
Types

- The Hail expression language is **typed**.
- What is the type of 3? **Int**
Types

• The Hail expression language is typed.

• What is the type of 3? Int

• What is the type of 3.14?
Types

• The Hail expression language is typed.

• What is the type of 3? Int

• What is the type of 3.14? Double

• What is the type of “Hello, world!”?
Types

• The Hail expression language is **typed**.

• What is the type of 3? **Int**

• What is the type of 3.14? **Double**

• What is the type of “Hello, world!”? **String**
Types

- The Hail expression language is **typed**.
- What is the type of 3? **Int**
- What is the type of 3.14? **Double**
- What is the type of “Hello, world!”? **String**
- We write 3: **Int** to indicate that 3 has type **Int**. Similarly for 3.14: **Double** and “Hello, world!”: **String**.
Types

- The Hail expression language is typed.
- What is the type of 3? Int
- What is the type of 3.14? Double
- What is the type of “Hello, world!”? String
- 5 and “5” and 5.0 all have different types!
Types

- Int, Double and String are **primitive** types.

- What is the type of [1, 2, 3]?
Types

- Int, Double and String are **primitive** types.
- What is the type of [1, 2, 3]? Array
Types

• *Int*, *Double* and *String* are *primitive* types.

• What is the type of \([1, 2, 3]\)？ *Array[Int]*
Types

• Int, Double and String are **primitive** types.

• What is the type of [1, 2, 3]? Array[Int]

• What is the type of [1, 3.14, “foo”]?
Types

• **Int**, **Double** and **String** are *primitive* types.

• What is the type of `[1, 2, 3]`? Array[Int]

• What is the type of `[1, 3.14, “foo”]`? **No.**

• You can also have Array[Double], Array[Array[Double]], ... Array[T]
Types

• Int, Double and String are **primitive** types.

• Array[T] is a **compound** type, since it contains types. We will learn about more compound types later.
Types

- Int, Double and String are **primitive** types.

- Array\[T\] is a **compound** type, since it contains types. We will learn about more compound types later.

- Hail also has (primitive) types for genetic concepts like Variant, Genotype, Interval, etc. A genotype is printed like this:
  \[
  \text{Genotype}(\text{GT}=0, \text{AD}=[21, 0], \text{DP}=21, \\
  \text{GQ}=60, \text{PL}=[0, 60, 759])
  \]
Main Python objects

- **HailContext**: main entry point for Hail functionality
- **VariantDataset**: Hail’s representation of a dataset
- **KeyTable**: Table-like structure (think data frame)
From VCF ...

<table>
<thead>
<tr>
<th>#CHROM</th>
<th>POS</th>
<th>REF</th>
<th>ALT</th>
<th>INFO</th>
<th>C1046::HG02024</th>
<th>C1046::HG02025</th>
<th>C1046::HG02026</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>10019093</td>
<td>A</td>
<td>G</td>
<td>AF=0.582</td>
<td>0/0:30,0:30:72:0,72,1080</td>
<td>0/1:49,45:94:99:1</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026348</td>
<td>A</td>
<td>G</td>
<td>AF=0.005172</td>
<td>0/0:23,0:23:60:0,60,900 0/0:22,0:22:60:0,60,900 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026357</td>
<td>T</td>
<td>C</td>
<td>AF=0.23</td>
<td>0/0:23,0:23:60:0,60,900 0/0:22,0:22:60:0,60,900 0/0:26,0:26:63:0,63,945 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030188</td>
<td>T</td>
<td>A</td>
<td>AF=0.219</td>
<td>0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030452</td>
<td>G</td>
<td>A</td>
<td>AF=0.216</td>
<td>0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030508</td>
<td>T</td>
<td>C</td>
<td>AF=0.002874</td>
<td>0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030573</td>
<td>G</td>
<td>A</td>
<td>AF=0.002874</td>
<td>0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10032413</td>
<td>T</td>
<td>G</td>
<td>AF=0.221</td>
<td>0/0:21,0:21:60:0,60,849 0/0:23,0:23:60:0,60,821 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036107</td>
<td>T</td>
<td>G</td>
<td>AF=0.032</td>
<td>0/0:26,0:26:66:0,66,990 0/0:16,0:16:39:0,39,585 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036141</td>
<td>C</td>
<td>T</td>
<td>AF=0.024</td>
<td>0/0:29,0:29:81:0,81,1215</td>
<td>0/0:16,0:16:39:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036202</td>
<td>G</td>
<td>A</td>
<td>AF=0.047</td>
<td>0/0:29,0:29:81:0,81,1215</td>
<td>0/0:22,0:22:63:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10256252</td>
<td>G</td>
<td>T</td>
<td>AF=0.118</td>
<td>0/1:2,5:7:60:166:0,60 0/0:6,0:6:15:0,15,219 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>C</td>
<td>CT</td>
<td>AF=0.097</td>
<td>0/0:33,4:41:32:0,32,830 0/0:63,0:63:0:0,1271 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>CT</td>
<td>C</td>
<td>AF=0.187</td>
<td>0/0:33,4:41:43:0,43,947 0/0:63,0:63:0:0,1271 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10277621</td>
<td>C</td>
<td>T</td>
<td>AF=0.132</td>
<td>0/0:35,34:69:90:1040,0,994</td>
<td>0/0:31,0:31:60:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280082</td>
<td>A</td>
<td>G</td>
<td>AF=0.747E-4</td>
<td>0/0:28,0:28:69:0,69,1035</td>
<td>0/0:15,0:15:39:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280083</td>
<td>G</td>
<td>A</td>
<td>AF=0.136</td>
<td>0/0:28,0:28:45:0,45,931 0/0:15,0:15:39:0,39,527 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10286773</td>
<td>C</td>
<td>T</td>
<td>AF=0.027</td>
<td>0/0:26,0:26:75:0,75,1052</td>
<td>0/0:24,0:24:60:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385849</td>
<td>C</td>
<td>A</td>
<td>AF=0.021</td>
<td>0/0:7,0:7:21:0,21,257 0/0:8,0:8:21:0,21,315 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385857</td>
<td>T</td>
<td>C</td>
<td>AF=0.021</td>
<td>0/0:7,0:7:15:0,15,225 0/0:8,0:8:21:0,21,315 0</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386013</td>
<td>C</td>
<td>A</td>
<td>AF=0.189</td>
<td>0/1:37,38:75:99:1036,0,1133</td>
<td>0/1:45,35:80:99:1</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386059</td>
<td>G</td>
<td>A</td>
<td>AF=0.19</td>
<td>0/1:48,38:86:99:1079,0,1318</td>
<td>0/1:52,33:85:99:949,0,149</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389422</td>
<td>T</td>
<td>C</td>
<td>AF=0.001724</td>
<td>0/0:27,0:27:72:0,72,1080</td>
<td>0/0:32,0:32:32:87:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389480</td>
<td>T</td>
<td>A</td>
<td>AF=0.244</td>
<td>0/1:10,17:27:99:550,0,285</td>
<td>0/1:19,17:36:99:4</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393145</td>
<td>C</td>
<td>G</td>
<td>AF=0.285</td>
<td>0/0:86,0:86:99:0,120,1800</td>
<td>0/0:89,0:89:99:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393162</td>
<td>A</td>
<td>C</td>
<td>AF=0.189</td>
<td>0/1:87,71:158:99:2011,0,2500</td>
<td>0/1:72,56:128:99:</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393439</td>
<td>C</td>
<td>A</td>
<td>AF=0.001149</td>
<td>0/0:56,0:56:99:0,120,1800</td>
<td>0/0:62,0:62:99:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393629</td>
<td>G</td>
<td>A</td>
<td>AF=0.187</td>
<td>0/1:52,63:115:99:1758,0,1651</td>
<td>0/1:50,60:110:99:</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393689</td>
<td>T</td>
<td>C</td>
<td>AF=0.004023</td>
<td>0/0:55,0:55:00:0,120,1800</td>
<td>0/0:57,0:57:00:0,</td>
<td></td>
</tr>
<tr>
<td>#CHROM</td>
<td>POS</td>
<td>REF</td>
<td>ALT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>-----</td>
<td>-----</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10019093</td>
<td>A</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026348</td>
<td>A</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026357</td>
<td>T</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030188</td>
<td>T</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030452</td>
<td>G</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030508</td>
<td>T</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030573</td>
<td>G</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10032413</td>
<td>T</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036107</td>
<td>T</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036141</td>
<td>C</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036202</td>
<td>G</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10256252</td>
<td>G</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>C</td>
<td>CT</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>CT</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10277621</td>
<td>C</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280082</td>
<td>A</td>
<td>G</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280083</td>
<td>G</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10286773</td>
<td>C</td>
<td>T</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385849</td>
<td>C</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385857</td>
<td>T</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386013</td>
<td>C</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386059</td>
<td>G</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389422</td>
<td>T</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389480</td>
<td>T</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393145</td>
<td>G</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393162</td>
<td>A</td>
<td>C</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393439</td>
<td>C</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393629</td>
<td>G</td>
<td>A</td>
<td></td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

INFO  
C1046::HG02024  C1046::HG02025  C1046::HG02026  
AF=0.582  0/0:30,0:30:72:0,72,1080  0/1:49,45:94:99:1  
AF=0.005172  0/0:23,0:23:60:0,60,900  0/0:22,0:22:60:0,60,900  0/0:26,0:26:
AF=0.23  0/0:23,0:23:60:0,60,900  0/0:22,0:22:60:0,60,900  0/0:26,0:
AF=0.219  0/0:35,0:35:60:0,60,900  0/0:26,0:26:63:0,63,945  0/0:26,0:26:
AF=0.216  0/0:35,0:35:60:0,60,900  0/0:26,0:26:63:0,63,945  0/0:26,0:26:
AF=0.002874  0/0:35,0:35:60:0,60,900  0/0:26,0:26:63:0,63,945  0/0:26,0:26:
AF=0.221  0/0:21,0:21:60:0,60,849  0/0:23,0:23:60:0,60,821  0/0:26,0:26:
AF=0.032  0/0:26,0:26:66:0,66,990  0/0:16,0:16:39:0,39,585  0/0:16,0:16:
AF=0.024  0/0:29,0:29:81:0,81,1215  0/0:16,0:16:39:0,39,585  0/0:16,0:16:
AF=0.047  0/0:29,0:29:81:0,81,1215  0/0:22,0:22:63:0,63,945  0/0:22,0:22:
AF=0.118  0/1:2,5:7:60:166:0,60  0/0:6,0:6:15:0,15,219  0/0:6,0:6:15:
AF=0.097  0/0:33,4:41:32:0,32,830  0/0:63,63:0:0,1271  0/0:63,63:0:
AF=0.187  0/0:33,4:41:43:0,43,947  0/0:63,63:0:0,1271  0/0:63,63:0:
AF=0.132  0/1:35,34:69:99:1040,0,994  0/0:31,0:31:60:0,31:
AF=5.747E-4  0/0:28,0:28:69:0,69,1035  0/0:15,0:15:39:0,39,527  0/0:15,0:15:
AF=0.136  0/0:28,0:28:45:0,45,931  0/0:15,0:15:39:0,39,527  0/0:15,0:15:
AF=0.027  0/0:26,0:26:75:0,75,1052  0/0:24,0:24:60:0,24:
AF=0.021  0/0:7,0:7:21:0,21,257  0/0:8,0:8:21:0,21,315  0/0:8,0:8:21:
AF=0.021  0/0:7,0:7:15:0,15,225  0/0:8,0:8:21:0,21,315  0/0:8,0:8:21:
AF=0.189  0/1:37,38:75:99:1036,0,1133  0/1:45,38:80:99:1  0/1:45,38:
AF=0.19  0/1:48,38:86:99:1079,0,1318  0/1:52,33:85:99:949,0,149  0/1:52,33:
AF=0.001724  0/0:27,0:27:72:0,72,1080  0/0:32,0:32:87:0,32:
AF=0.244  0/1:10,17:27:99:550,0,285  0/1:19,17:36:99:4  0/1:19,17:
AF=0.285  0/0:86,0:86:99:120,1800  0/0:89,0:89:99:0,89:
AF=0.189  0/1:87,71:158:99:2011,0,2500  0/1:72,56:128:99:1  0/1:72,56:
AF=0.001149  0/0:56,0:56:99:120,1800  0/0:62,0:62:99:0,62:
AF=0.187  0/1:52,63:115:99:1758,0,1651  0/1:50,60:110:99:9  0/1:50,60:
AF=0.004023  0/0:55,0:55:00:0,120,1800  0/0:67,0:67:00:0,67:
<table>
<thead>
<tr>
<th>#CHROM</th>
<th>POS</th>
<th>REF</th>
<th>ALT</th>
<th>INFO</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>10019093</td>
<td>A</td>
<td>G</td>
<td>AF=0.582 0:0:30,0:30:72:0,72,1080 0/1:49,45:94:99:1</td>
</tr>
<tr>
<td>20</td>
<td>10026348</td>
<td>A</td>
<td>G</td>
<td>AF=0.005172 0/0:23,0:23:60:0,60,900 0/0:22,0:22:60:0,60,900</td>
</tr>
<tr>
<td>20</td>
<td>10026357</td>
<td>T</td>
<td>C</td>
<td>AF=0.23 0/0:23,0:23:60:0,60,900 0/0:22,0:22:60:0,60,900</td>
</tr>
<tr>
<td>20</td>
<td>10030188</td>
<td>T</td>
<td>A</td>
<td>AF=0.219 0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945</td>
</tr>
<tr>
<td>20</td>
<td>10030452</td>
<td>G</td>
<td>A</td>
<td>AF=0.216 0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945</td>
</tr>
<tr>
<td>20</td>
<td>10030508</td>
<td>T</td>
<td>C</td>
<td>AF=0.002874 0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945</td>
</tr>
<tr>
<td>20</td>
<td>10030573</td>
<td>G</td>
<td>A</td>
<td>AF=0.002874 0/0:35,0:35:60:0,60,900 0/0:26,0:26:63:0,63,945</td>
</tr>
<tr>
<td>20</td>
<td>10032413</td>
<td>T</td>
<td>G</td>
<td>AF=0.221 0/0:21,0:21:60:0,60,849 0/0:23,0:23:60:0,60,821</td>
</tr>
<tr>
<td>20</td>
<td>10036107</td>
<td>T</td>
<td>G</td>
<td>AF=0.032 0/0:26,0:26:66:0,66,990 0/0:16,0:16:39:0,39,585</td>
</tr>
<tr>
<td>20</td>
<td>10036141</td>
<td>C</td>
<td>T</td>
<td>AF=0.024 0/0:29,0:29:81:0,81,1215 0/0:16,0:16:39:0,</td>
</tr>
<tr>
<td>20</td>
<td>10036202</td>
<td>G</td>
<td>A</td>
<td>AF=0.047 0/0:29,0:29:81:0,81,1215 0/0:22,0:22:63:0,</td>
</tr>
<tr>
<td>20</td>
<td>10256252</td>
<td>G</td>
<td>T</td>
<td>AF=0.118 0/1:2,5:7:60:166:0,60 0/0:6,0:6:15:0,15,219</td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>C</td>
<td>C</td>
<td>AF=0.097 0/0:33,4:41:32:0,32,830 0/0:63,0:63:0,0,1271</td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>CT</td>
<td>C</td>
<td>AF=0.187 0/0:33,4:41:43:0,43,947 0/0:63,0:63:0,0,1271</td>
</tr>
<tr>
<td>20</td>
<td>10277621</td>
<td>C</td>
<td>T</td>
<td>AF=0.132 0/1:35,34:69:69:1040,0,994 0/0:31,0:31:60:0,</td>
</tr>
<tr>
<td>20</td>
<td>10280082</td>
<td>A</td>
<td>G</td>
<td>AF=5.747E-4 0/0:28,0:28:69:0,69,1035 0/0:15,0:15:39:0,</td>
</tr>
<tr>
<td>20</td>
<td>10280083</td>
<td>G</td>
<td>A</td>
<td>AF=0.136 0/0:28,0:28:45:0,45,931 0/0:15,0:15:39:0,39,527</td>
</tr>
<tr>
<td>20</td>
<td>10286773</td>
<td>C</td>
<td>T</td>
<td>AF=0.027 0/0:26,0:26:75:0,75,1052 0/0:24,0:24:60:0,</td>
</tr>
<tr>
<td>20</td>
<td>10385849</td>
<td>C</td>
<td>A</td>
<td>AF=0.021 0/0:7,0:7:21:0,21,257 0/0:8,0:8:21:0,21,315</td>
</tr>
<tr>
<td>20</td>
<td>10385857</td>
<td>T</td>
<td>C</td>
<td>AF=0.021 0/0:7,0:7:15:0,15,225 0/0:8,0:8:21:0,21,315</td>
</tr>
<tr>
<td>20</td>
<td>10386013</td>
<td>C</td>
<td>A</td>
<td>AF=0.189 0/1:37,38:75:99:1036,0,1133 0/1:45,35:80:99:1,</td>
</tr>
<tr>
<td>20</td>
<td>10386059</td>
<td>G</td>
<td>A</td>
<td>AF=0.19 0/1:48,38:86:99:1079,0,1318 0/1:52,33:85:99:949,0,149</td>
</tr>
<tr>
<td>20</td>
<td>10389422</td>
<td>T</td>
<td>C</td>
<td>AF=0.001724 0/0:27,0:27:72:0,72,1080 0/0:32,0:32:87:0,</td>
</tr>
<tr>
<td>20</td>
<td>10389480</td>
<td>T</td>
<td>A</td>
<td>AF=0.244 0/1:10,17:27:99:550,0,285 0/1:19,17:36:99:4,</td>
</tr>
<tr>
<td>20</td>
<td>10393145</td>
<td>C</td>
<td>G</td>
<td>AF=0.285 0/0:86,0:86:99:0,120,1800 0/0:89,0:89:99:0,</td>
</tr>
<tr>
<td>20</td>
<td>10393162</td>
<td>A</td>
<td>C</td>
<td>AF=0.189 0/1:87,71:158:99:2011,0,2500 0/1:72,56:128:99:</td>
</tr>
<tr>
<td>20</td>
<td>10393439</td>
<td>C</td>
<td>A</td>
<td>AF=0.001149 0/0:56,0:56:99:0,120,1800 0/0:62,0:62:99:0,</td>
</tr>
<tr>
<td>20</td>
<td>10393629</td>
<td>G</td>
<td>A</td>
<td>AF=0.187 0/1:52,63:115:99:1758,0,1651 0/1:50,60:110:99:</td>
</tr>
<tr>
<td>20</td>
<td>10393690</td>
<td>T</td>
<td>C</td>
<td>AF=0.004023 0/0:55,0:55:99:0,120,1800 0/0:57,0:57:99:0,</td>
</tr>
</tbody>
</table>
From VCF ...

<table>
<thead>
<tr>
<th>#CHROM</th>
<th>POS</th>
<th>REF</th>
<th>ALT</th>
<th>INFO</th>
<th>C1046::HG02024</th>
<th>C1046::HG02025</th>
<th>C1046::HG02026</th>
</tr>
</thead>
<tbody>
<tr>
<td>20</td>
<td>10019093</td>
<td>A</td>
<td>G</td>
<td>AF=0.582</td>
<td>0/0:30,0:30:72:0,72,1080</td>
<td>0/1:49,45:94:99:1</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026348</td>
<td>A</td>
<td>G</td>
<td>AF=0.005172</td>
<td>0/23:0,0:23:60:0,60,900</td>
<td>0/0:22,0:22:60:0,60,900</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026357</td>
<td>T</td>
<td>C</td>
<td>AF=0.23</td>
<td>0:23:60:0,60,900</td>
<td>0/0:22,0:22:60:0,60,900</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030188</td>
<td>T</td>
<td>A</td>
<td>AF=0.219</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030452</td>
<td>G</td>
<td>A</td>
<td>AF=0.216</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030508</td>
<td>T</td>
<td>C</td>
<td>AF=0.002874</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030573</td>
<td>G</td>
<td>A</td>
<td>AF=0.002874</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10032413</td>
<td>T</td>
<td>G</td>
<td>AF=0.221</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036107</td>
<td>T</td>
<td>G</td>
<td>AF=0.032</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036141</td>
<td>C</td>
<td>T</td>
<td>AF=0.024</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036202</td>
<td>G</td>
<td>A</td>
<td>AF=0.047</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10256252</td>
<td>G</td>
<td>T</td>
<td>AF=0.118</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>C</td>
<td>CT</td>
<td>AF=0.097</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>CT</td>
<td>C</td>
<td>AF=0.187</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10277621</td>
<td>C</td>
<td>T</td>
<td>AF=0.132</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280082</td>
<td>A</td>
<td>G</td>
<td>AF=5.747E-4</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280082</td>
<td>G</td>
<td>A</td>
<td>AF=0.136</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10286773</td>
<td>C</td>
<td>T</td>
<td>AF=0.027</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385849</td>
<td>C</td>
<td>A</td>
<td>AF=0.021</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385857</td>
<td>T</td>
<td>C</td>
<td>AF=0.021</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386013</td>
<td>C</td>
<td>A</td>
<td>AF=0.189</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386059</td>
<td>G</td>
<td>A</td>
<td>AF=0.19</td>
<td>0/1:48:38:86:99:179,0,1318</td>
<td>0/1:52,33:85:99:949,0,149</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389422</td>
<td>T</td>
<td>C</td>
<td>AF=0.001724</td>
<td>0/0:27,0:27:72:0,72,1080</td>
<td>0/0:32,0:32:87:0,82,1080</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389480</td>
<td>T</td>
<td>A</td>
<td>AF=0.244</td>
<td>0/1:10,17:27:99:550,0,285</td>
<td>0/1:19,17:36:99:4,285</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393145</td>
<td>C</td>
<td>G</td>
<td>AF=0.285</td>
<td>0/0:86,0:86:99:0,120,1800</td>
<td>0/0:89,0:89:99:0,120,1800</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393162</td>
<td>A</td>
<td>C</td>
<td>AF=0.189</td>
<td>0/1:87,71:158:99:2011,0,2500</td>
<td>0/1:72,56:128:99:2,2500</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393439</td>
<td>C</td>
<td>A</td>
<td>AF=0.001149</td>
<td>0/0:56,0:56:99:0,120,1800</td>
<td>0/0:62,0:62:99:0,120,1800</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393629</td>
<td>G</td>
<td>A</td>
<td>AF=0.187</td>
<td>0/0:55,0:55:99:0,120,1800</td>
<td>0/0:67,0:67:99:0,120,1800</td>
<td></td>
</tr>
<tr>
<td>#CHROM</td>
<td>POS</td>
<td>REF</td>
<td>ALT</td>
<td>INFO</td>
<td>C1045::HG02024</td>
<td>C1046::HG02025</td>
<td>C1046::HG02026</td>
</tr>
<tr>
<td>--------</td>
<td>-------</td>
<td>-----</td>
<td>-----</td>
<td>------------------</td>
<td>----------------</td>
<td>----------------</td>
<td>----------------</td>
</tr>
<tr>
<td>20</td>
<td>10019093</td>
<td>A</td>
<td>G</td>
<td>AF=0.582</td>
<td>0/0:30,0:30:72:0,72,1080</td>
<td>0/1:49,45:94:99:1</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026348</td>
<td>A</td>
<td>G</td>
<td>AF=0.005172</td>
<td>0/0:23,0:23:60:0,60,900</td>
<td>0/0:22,0:22:60:0,60,900</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10026357</td>
<td>T</td>
<td>C</td>
<td>AF=0.23 0/0:23,0:23:60:0,60,900</td>
<td>0/0:22,0:22:60:0,60,900</td>
<td></td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030188</td>
<td>T</td>
<td>A</td>
<td>AF=0.219</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030452</td>
<td>G</td>
<td>A</td>
<td>AF=0.216</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030508</td>
<td>T</td>
<td>C</td>
<td>AF=0.002874</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10030573</td>
<td>G</td>
<td>A</td>
<td>AF=0.002874</td>
<td>0/0:35,0:35:60:0,60,900</td>
<td>0/0:26,0:26:63:0,63,945</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10032413</td>
<td>T</td>
<td>G</td>
<td>AF=0.221</td>
<td>0/0:21,0:21:60:0,60,849</td>
<td>0/0:23,0:23:60:0,60,821</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036107</td>
<td>T</td>
<td>G</td>
<td>AF=0.032</td>
<td>0/0:26,0:26:66:0,66,990</td>
<td>0/0:16,0:16:39:0,39,585</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036141</td>
<td>C</td>
<td>T</td>
<td>AF=0.024</td>
<td>0/0:29,0:29:81:0,81,1215</td>
<td>0/0:16,0:16:39:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10036202</td>
<td>G</td>
<td>A</td>
<td>AF=0.047</td>
<td>0/0:29,0:29:81:0,81,1215</td>
<td>0/0:22,0:22:63:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10256252</td>
<td>G</td>
<td>T</td>
<td>AF=0.118</td>
<td>0/1:2,5:7:60:166:0,60</td>
<td>0/0:6,0:6:15:0,15,219</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>C</td>
<td>CT</td>
<td>AF=0.097</td>
<td>0/0:33,4:41:32:0,32,830</td>
<td>0/0:63,0:63:0:0,1271</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10273694</td>
<td>CT</td>
<td>C</td>
<td>AF=0.187</td>
<td>0/0:33,4:41:43:0,43,947</td>
<td>0/0:63,0:63:0,0,1271</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10277621</td>
<td>C</td>
<td>T</td>
<td>AF=0.132</td>
<td>0/1:35,34:69:99:1040,0,994</td>
<td>0/0:31,0:31:60:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280082</td>
<td>A</td>
<td>G</td>
<td>AF=5.747E-4</td>
<td>0/0:28,0:28:69:0,69,1035</td>
<td>0/0:15,0:15:39:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10280083</td>
<td>G</td>
<td>A</td>
<td>AF=0.136</td>
<td>0/0:28,0:28:45:0,45,931</td>
<td>0/0:15,0:15:39:0,39,527</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10286773</td>
<td>C</td>
<td>T</td>
<td>AF=0.027</td>
<td>0/0:26,0:26:75:0,75,1052</td>
<td>0/0:24,0:24:60:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385849</td>
<td>C</td>
<td>A</td>
<td>AF=0.021</td>
<td>0/0:7,0:7:21:0,21,257</td>
<td>0/0:8,0:8:21:0,21,315</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10385857</td>
<td>T</td>
<td>C</td>
<td>AF=0.021</td>
<td>0/0:7,0:7:15:0,15,225</td>
<td>0/0:8,0:8:21:0,21,315</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386013</td>
<td>C</td>
<td>A</td>
<td>AF=0.189</td>
<td>0/1:37,38:75:99:1036,0,1133</td>
<td>0/1:45,35:80:99:1</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10386059</td>
<td>G</td>
<td>A</td>
<td>AF=0.19</td>
<td>0/1:48,38:86:99:1079,0,1318</td>
<td>0/1:52,33:85:99:949,0,149</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389422</td>
<td>T</td>
<td>C</td>
<td>AF=0.001724</td>
<td>0/0:27,0:27:72:0,72,1080</td>
<td>0/0:32,0:32:87:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10389480</td>
<td>T</td>
<td>A</td>
<td>AF=0.244</td>
<td>0/1:10,17:27:99:550,0,285</td>
<td>0/1:19,17:36:99:4</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393145</td>
<td>C</td>
<td>G</td>
<td>AF=0.285</td>
<td>0/0:86,0:86:99:0,120,1800</td>
<td>0/0:89,0:89:99:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393162</td>
<td>A</td>
<td>C</td>
<td>AF=0.189</td>
<td>0/1:87,71:158:99:2011,0,2500</td>
<td>0/1:72,56:128:99:</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393439</td>
<td>C</td>
<td>A</td>
<td>AF=0.001149</td>
<td>0/0:56,0:56:99:0,120,1800</td>
<td>0/0:62,0:62:99:0,</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393629</td>
<td>G</td>
<td>A</td>
<td>AF=0.187</td>
<td>0/1:52,63:115:99:1758,0,1651</td>
<td>0/1:50,60:110:99:</td>
<td></td>
</tr>
<tr>
<td>20</td>
<td>10393690</td>
<td>T</td>
<td>C</td>
<td>AF=0.094023</td>
<td>0/0:55,0:55:0:0,120,1800</td>
<td>0/0:57,0:57:00:0,</td>
<td></td>
</tr>
</tbody>
</table>
... To Variant Dataset

Annotations

Variant

Sample

Genotypes
... To Variant Dataset

- **v**: Variant
- **va**: v
- **s**: String
- **sa**: s
- **g**: Genotype
- **gs**: Aggregable[Genotype]
Balding-Nichols model

Ancestral population

Pop 1
0.2

Pop 2
0.3

Pop 3
0.5

Fixation Index: $F_{ST}$
- K populations, N samples, M variants.
- \( \pi \) is the population distribution of samples
- \( P_0 \) is ancestral frequency spectrum (uniform distribution from 0.1 to 0.9)

\[
\begin{align*}
    k_n & \sim \pi \\
    p_{0,m} & \sim P_0 \\
    p_{k,m} \mid p_{0,m} & \sim \text{Beta}(\mu = p_{0,m}, \sigma^2 = F_k p_{0,m} (1 - p_{0,m})) \\
    g_{n,m} \mid k_n, p_{k,m} & \sim \text{Binomial}(2, p_{k_n,m}).
\end{align*}
\]
Outline of Hail Practicals

1. Importing, schemas, simulated data
2. The Hail expression language
3. Annotation, query and plotting
4. Aggregables: working with massive data
5. Understanding GQ and DP in sequence data
6. Unmasking ancestry
7. Basic association analysis
Practical 1: What did we learn?

• Hail has its own file format, VDS. Why?

• **VariantDatasets** have three schemas. What are they?

• You can simulate genotypes and phenotypes in Hail.
Simulating data

```python
In [22]:
(hc.balding_nichols_model(3, 2000, 2000,
   pop_dist = [0.2, 0.3, 0.5],
   fst = [.07, .11, .13])
   .annotate_samples_expr(['sa.cov1 = rnorm(0, 1)',
   'sa.cov2 = rnorm(0, 1)'])
   .annotate_samples_expr(
   'sa.pheno = rnorm(1, 1) + 2 * sa.cov1 - sa.cov2 + .1 * sa.pop')
   .write('synth.vds', overwrite=True))

synth_vds = hc.read('synth.vds')
```
Hail Expression Language

- Used all over! Filtering, export, annotating, calculating, covariates, …

- Syntax a mishmash styles. We apologize in advance.

- Built-in support for missing values: `NA`.

- Expression language. No user-defined functions, no loops.

- Typed language. All expressions are statically typed.

- Functional. Modifying makes a copy.
# Hail Types

<table>
<thead>
<tr>
<th>Primitive</th>
<th>Compound</th>
<th>Genetic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Boolean</td>
<td>Array[T]</td>
<td>Locus</td>
</tr>
<tr>
<td>Int</td>
<td>Set[T]</td>
<td>AltAllele</td>
</tr>
<tr>
<td>Long</td>
<td>Dict[K, V]</td>
<td>Variant</td>
</tr>
<tr>
<td>Float</td>
<td>Struct {</td>
<td>Interval</td>
</tr>
<tr>
<td></td>
<td>f1: T1,</td>
<td></td>
</tr>
<tr>
<td></td>
<td>f2: T2, ... }</td>
<td>IntervalList</td>
</tr>
<tr>
<td>Double</td>
<td>Aggregable[T]*</td>
<td>Genotype</td>
</tr>
<tr>
<td>String</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

*This is the other most confusing part.*
Hail Expression Language

• Don’t confuse the Hail expression language with Python!

• Hail expressions are written as **strings** in Python and passed to Hail python functions.
Hail Expression Language

- To evaluate a Python expression, you enter it in the Python interpreter:
  In [5]: 1 + 1
  Out[5]: 2

- To evaluate a Hail expression, you pass it as a string in Python to `hc.eval_expr_expr_typed`:
  In [6]: hc.eval_expr_expr_typed('1 + 1')
  Out[6]: (2, Int)
Hail Expression Language

- To evaluate a Hail expression, you pass it as a string in Python to `hc.eval_expr_typed`:
  
  In [6]: hc.eval_expr_typed('1 + 1')
  Out[6]: (2, Int)

- What is the return value?
Hail Expression Language

• `[1, 5, 10].filter(x => x < 10)`

• The `=>` syntax describes a unnamed function. `x` refers to the elements of the array.
Outline of Hail Practicals

1. Importing, schemas, simulated data

2. The Hail expression language

3. Annotation, query and plotting

4. Aggregables: working with massive data

5. Understanding GQ and DP in sequence data

6. Unmasking ancestry

7. Basic association analysis
Practical 2: What did we learn?

- Hail expression syntax is weird and annoying.
- Hail naturally handles missing values like R.
- You can transform Arrays with functional operators like `map`, `filter`.
- You can reduce Arrays with operators like `max` and `mean`.
- Hail supports Structs. You had already seen this before. Where?
Aggregables

• Is that even a word?

• This is the hardest part of Hail. Once you get this, you’re golden.

• How do you manipulate datasets that are bigger than one computer?

• How do you understand, say, the distribution of DP in a dataset with 100T genotypes?
Aggregables

- Aggregable[T] is an unordered, distributed collection of T.

- Aggregable[Int] is an distributed collection of Ints.

- The interface for Aggregable is modeled Array
Aggregables

- `gs: Aggregable[Genotype]`

- `gs.map(g => g.dp)` is an `Aggregable[Int]`

- `gs.filter(g => g.gq > 20)` is a (smaller) `Aggregable[Genotype]`

- `gs.map(g => g.dp).max()` is an `Int`

- Reduction operations like `max` are called **aggregators**. Arrays and Aggregables support a slightly different set of reduction operators.
Aggregable Context

- Aggregables have contexts. This is the second way they differ from Arrays.
VariantDataset

global

v: Variant

va

s: String

sa

g: Genotype

gs: Aggregable[Genotype]
Aggregable Context

- **Aggregables** have contexts. This is the second way they differ from Arrays.

- **map**, **filter** manipulate aggregable elements, **not** context.

- Examples:
  ```javascript
  gs.map(g => va.callRate)...
  (gs.map(g => g.dp)
   .filter(dp => g.gq > 20)...
  )
  ```

- Aggregable context documented with the aggregagable. They can all be figured out from the previous diagram.
Genotype Context

- global
- g: Genotype
- v: Variant
- s: Sample
- va
- sa
Variant Context

- global
- v: Variant
- va
- gs: Aggregable[Genotype]
Outline of Hail Practicals

1. Importing, schemas, simulated data

2. The Hail expression language

3. Annotation, query and plotting

4. **Aggregables: working with massive data**

5. Understanding GQ and DP in sequence data

6. Unmasking ancestry

7. Basic association analysis
Practical 4: What did we learn?
Practical 4: What did we learn?

• Aggregables are a convenient and elegant way to manipulate large, distributed data objects.

• Aggregables can be manipulated similarly to arrays.

• Aggregables carry a natural context that is not changed by \texttt{map} and \texttt{filter}. 