End Joining Signatures - dev

Charlie Soeder

2/15/2019

25 February 2019

Rebuilding, starting with summary stats for the materials/methods section.

Reference genomes

Table 1: Size and Consolidation of Reference Genomes

Reference Genome:	dm6
number_bases	144 M
number_contigs	1.87 k

Sequenced reads

Table 2: Number of Sequenced Samples by Treatment

experimental	sample_count
mutant	30
control	44

Table 3: Sequenced Experimental Samples

name	paired	experimental	source
Mcm5-A7	TRUE	mutant	talia
mcm5-28	TRUE	mutant	talia
mcm5-27	TRUE	mutant	talia
mcm5-26	TRUE	mutant	talia
mcm5-25	TRUE	mutant	talia
mcm5-24	TRUE	mutant	talia
mcm5-23	TRUE	mutant	talia
mcm5-22	TRUE	mutant	talia
mcm5-21	TRUE	mutant	talia
mcm5-20	TRUE	mutant	talia
mcm5-19	TRUE	mutant	talia
mcm5-18	TRUE	mutant	talia
mcm5-17	TRUE	mutant	talia
mcm5-16	TRUE	mutant	talia
mcm5-15	TRUE	mutant	talia
mcm5-14	TRUE	mutant	talia
mcm5-13	TRUE	mutant	talia
mcm5-12	TRUE	mutant	talia
mcm5-11	TRUE	mutant	talia
mcm5-10	TRUE	mutant	talia
mcm5-09	TRUE	mutant	talia

name	paired	experimental	source
mcm5-08	TRUE	mutant	talia
mcm5-07	TRUE	mutant	talia
mcm5-06	TRUE	mutant	talia
mcm5-05	TRUE	mutant	talia
mcm5-04	TRUE	mutant	talia
mcm5-03	TRUE	mutant	talia
mcm5-02	TRUE	mutant	talia
mcm5-01	TRUE	mutant	talia
DfMcm5	TRUE	mutant	talia
w13_4	TRUE	control	danny
cs2_7	TRUE	control	danny
cs2_7	TRUE	control	danny
cs12_3	TRUE	control	danny
w15_7	TRUE	control	danny
w5_17	TRUE	control	danny
w5_17	TRUE	control	danny
w12_5	TRUE	control	danny
w13_1	TRUE	control	danny
cs8_8	TRUE	control	danny
cs8_8	TRUE	control	danny
w15_6	TRUE	control	danny
w3_16	TRUE	control	danny
w3_16	TRUE	control	danny
w4_6	TRUE	control	danny
w4_6	TRUE	control	danny
w12_12	TRUE	control	danny
w12_1	TRUE	control	danny
w11_13	TRUE	control	danny
w4_7	TRUE	control	danny
w4_7	TRUE	control	danny
w3_24	TRUE	control	danny
w3_24	TRUE	control	danny
w12_7	TRUE	control	danny
w5_15	TRUE	control	danny
w5_15	TRUE	control	danny
w13_13	TRUE	control	danny
w15_8	TRUE	control	danny
w4_3	TRUE	control	danny
w4_3	TRUE	control	danny
cs12_5	TRUE	control	danny
w3_12	TRUE	control	danny
w3_12	TRUE	control	danny
w4_17	TRUE	control	danny
w4_17	TRUE	control	danny
cs2_6	TRUE	control	danny
cs2_6	TRUE	control	danny
w4_8	TRUE	control	danny
w4_8	TRUE	control	danny
w15_9	TRUE	control	danny
w5_16	TRUE	control	danny
w5_16	TRUE	control	danny
w1118	TRUE	control	danny

name	paired	experimental	source
CantonS	TRUE	control	danny

Total Starting Reads: $3.88G$ Post-QC Reads: $3.64G$.

Table 4: Read Count and Percent Retention

type	minimum	average	maximum
prefiltered	42.3 M	64.7 M	75.7 M
postfiltered	39.4 M	60.6 M	71.1 M
percent retention	92.4	93.7	95.5

This framework is general-purpose enough that it might be a good template. . . . . .

Dupes:

Table 5: Percentage Duplication

minimum	average	median	maximum
4.1	7.6	5.8	43.5

27 February 2019

Bioinformatics tips on INDEL calling & normalization with DSB background:

https://genome.sph.umich.edu/w/images/b/b4/Variant_Calling_and_Filtering_for_INDELs.pdf

5 March 2019

Going to go ahead and recycle BWA-Uniq but may want to change the algorithm later. . . .

Table 6: Read Counts During Alignment & Filtration

measure	minimum	average	median	maximum
filtered_mapped_count	17.5 M	40.5 M	42.1 M	48.5 M
total_mapped_count	38.1 M	58.6 M	60.3 M	68.9 M
total_read_count	39.4 M	60.6 M	62.7 M	71.1 M

Table 7: Percentage of Reads Retained at Each Step

measure	minimum	average	median	maximum
filter_retention	39.7	68.8	70.1	73.8
mapping_retention	95.8	96.6	96.7	97.3

Depth of coverage:

Table 8: Depth of Coverage Statistics for Raw and Filtered Alignments

step	minimum	average	median	maximum
pre-filtration depth	37.1	57.2	58.6	66.4
post-filtration depth	17.6	40.7	42.1	48.5
depth retention percent	40.9	70.9	72.3	75.5

Breadth of coverage:

Will run the VCF caller on both BWA and BWA-Uniq; reporting will be reworked since we're interested in indels.

5 March 2019

Doing things a little differently, calling variants from both BWA and BWA-Uniq, then compare the two. (whereas before we used reference genome as a variable)

6 March 2019

```
## Warning: Removed 4 rows containing missing values (geom_path).
```

Histogram of SNPs by Number of Samples Called At Site
Number of Sites
1000
750
500
250
0
Number Samples
aligner
bwa
bwaUniq

Jointly Called SNPs Callable per Sample, by Breadth and Depth of Covera
breadth
depth
Fraction of SNPs Callable
0.995
0.990
0.985
0.980
0.975
90
95
20
30
40
50
60
aligner
bwa
bwaUniq

7 March 2019

Might also be good to do a comparison between the two VCFs using vcftools –diff.

```
vcftools --vcf variants/all_samples.vs_dm6.bwa.vcf --diff variants/all_samples.vs_dm6.bwaUniq.vcf --dif
```

This complains: Error: Cannot determine chromosomal ordering of files, both files must contain the same chromosomes to use the diff functions. Found chrUn_DS483679v1 in file 1 and chrUn_DS483680v1 in file 2.

Let's try using the –chr command to limit to main-line chromosomes. . . .

```
vcftools --vcf variants/all_samples.vs_dm6.bwa.vcf --diff variants/all_samples.vs_dm6.bwaUniq.vcf --dif
```

There is also the –diff-site-discordance flag:

"The MATCHING_ALLELES column tells you if the alleles called in file match exactly at that site (i.e the REF and ALT columns are identical in the two files). The N_COMMON_CALLED column tells you the number of individuals at that site that were called in both files (i.e. the individuals in the intersection of the two datasets that don't have missing data ./.). The N_DISCORD column tells you the number of individuals in the intersection that are discordant at that site." -Adam Auton

https://sourceforge.net/p/vcftools/mailman/message/27128665/

also maybe use –diff-indv-discordance then compare individual discordance to e.g. breadth reduction upon BAM filtration

Locii variable in A only:

```
cat dev/meta/VCFs/comparisonTest.diff.sites | tail -n +2 | awk '{if($3 == 1)print;}' | cut -f 1 | uniq

cat dev/meta/VCFs/comparisonTest.diff.sites | tail -n +2 | awk '{if($3 == 1)print;}' | cut -f 1 | uniq
```

in B only:

```
cat dev/meta/VCFs/comparisonTest.diff.sites | tail -n +2 | awk '{if($3 == 2)print;}' | cut -f 1 | uniq

cat dev/meta/VCFs/comparisonTest.diff.sites | tail -n +2 | awk '{if($3 == 2)print;}' | cut -f 1 | uniq
```

Locii variable in both A and B:

```
cat dev/meta/VCFs/comparisonTest.diff.sites | tail -n +2 | awk '{if($3 == "B")print;}' | cut -f 1 | unic

cat dev/meta/VCFs/comparisonTest.diff.sites | tail -n +2 | awk '{if($3 == "B")print;}' | cut -f 1 | unic
```

```
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   X2 = col_character()
## )
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   X2 = col_character()
## )
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   X2 = col_character()
## )
```

https://rstudio-pubs-static.s3.amazonaws.com/13301_6641d73cfac741a59c0a851feb99e98b.html

```
## Loading required package: grid
```

```
## Loading required package: futile.logger
```

```
## (polygon[GRID.polygon.1256], polygon[GRID.polygon.1257], polygon[GRID.polygon.1258], polygon[GRID.pol
```

of the ones in the intersection:

Locii variable in A and B but with at least one discordant individual:

```
## Parsed with column specification:
## cols(
##   X1 = col_double(),
##   X2 = col_character()
## )
```

a total of 7.6637×10^4 sites with at least one discordant individual.

```
## Parsed with column specification:
## cols(
##   X1 = col_character(),
##   X2 = col_double(),
##   X3 = col_character(),
##   X4 = col_double(),
##   X5 = col_double(),
##   X6 = col_double(),
##   X7 = col_double()
## )
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

OK so that's some VCF comparison infrastructure; how to implement? Here we are comparing VCFs from two different alignment/calling strategies but ultimately we'll probably want to compare subgroups of all_samples. Do I use the same rule for both? It's possible that down the line when an alignment/calling strategy is decided upon, I won't want something so general...

8 March 2019

Looking at INDELs within the control subset...

```
vcftools --vcf variants/all_samples.vs_dm6.bwaUniq.vcf --keep-only-indels --freq --out test
vcftools --vcf variants/all_samples.vs_dm6.bwaUniq.vcf --keep-only-indels --counts --out test
```

hmmm, we're presumably looking for individual-specific alleles; there could potentially be more than one such allele at a given site! (these show up in the VCF as eg 0/2)

The INDELs .... will they be het or hom??? Presumably Het, since the break repair only happens on one strand....

Finding all the variants from the above count tallies with only one chromosome in the population carrying; awk + grep for more filtration.

```
cat test.frq.count | grep -P ':1[$,\t]'
```

10 March 2019

Background: Figure 1 of (McVey and Lee 2008) and https://en.wikipedia.org/wiki/Non-homologous_end_joining

It looks like NHEJ and MMEJ are prone to forming heterozygous indels near the DSB site. (am i right that MMEJ at least will form flanking indels?)

Are the DSBs random across the genome? What are the odds that the same site would be struck by DSB and repair error twice in the population (ie, 30 flies)? if it's very low, we need to look for indels which are heterozygous for an indel in one individual but homozygous in everyone else.

One data subtelty here is that the called variants are vs the ref genome so the ancestral genotype is presumably the one with the higher (f~1.0 in approx isogenic pop) allele frequency. Thus a reversion to het dm6 reference (0/1) from a population of hom variants (1/1) would be a candidate site. Also, since a site which is different from the reference may later be altered via DSB repair (ie, in a 1/1 population a 1/2 genotype with low AF on 2 is a candidate.)

figs to make: * histogram of minor allele frequency for indel variants. Many will have a MinAF of 0 (ie, fixed differences); filter these? others will have a small minAF corresponding to e.g. 1 chromosome in the population.

if there is a flanking behavior this should be easy to pick out: do a histogram of intrachromsomal distance among variants and then to a freq_poly plot binned by genotype (ie, hom vs het). Look for heterozygotes which are neighbors.

From (Miller et al. 2016): "Drosophila oocytes experience ~11–17 DSBs per meiosis that are restricted to the euchromatin . . . How the position of these DSBs is determined and their fate (whether they become COs or NCOs) is poorly understood." also discusses recombinational hotspots; evidence is against them??

Table S2 for crossover sites identified in (Miller et al. 2016); Table S3 for noncrossover sites.

From Danny Miller 11 Jan 2019:

*First, I called SNPs using GATK. From this output I isolated novel deletions for both chromosome X and 2. I re-wrote this script tonight and re-ran it and found four likely novel deletions that I hadn't seen before:

mcm5-12, chrX:12825436, GAAA deletion mcm5-21, chr2R:12737873, A deletion mcm5-22, chr2R:8597548, T deletion mcm5-24, chr2L:21664793, TATATA deletion*

his main finding:

I really expected to find lots of deletions suggesting repair of DSBs via NHEJ, but I didn't. This finding is consistent with the 95-ish single genomes I sequenced from homozygous c3g females where I also failed to find any deletions.

So, maybe another figure would be deletions vs. the major allele? eg, histogram of indel change in nucleotides.

11 March 2019

```
vcftools --vcf variants/all_samples.vs_dm6.bwaUniq.vcf --keep-only-indels --freq --out indelFrq.test

 vcftools --vcf variants/all_samples.vs_dm6.bwaUniq.vcf --keep-only-indels --counts --out indelCount.tes
cat indelCount.test.frq.count  | tail -n +2 | tr ":" "\t" | nl -n ln | head -n 1000 > dev/indelCount.tes
```

importing a CSV with a variable number of columns: https://stackoverflow.com/questions/18922493/how-can-you-read-a-csv-file-in-r-with-different-number-of-columns

Filter to main-line autosomes?

Just some top-level stuff: how many chroms avail per site?

Distribution of variants per variant site?

Minimum allele frequency by site:

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```

Let's start with the simplest case: biallelic sites with all 60 chromosomes available and allele count of 1 (ie, one het indiv.) hmmm, a number of MNPs showing up; let's try to filter those out for a really simple case.

```
grep "#" variants/all_samples.vs_dm6.bwaUniq.vcf > variants/all_samples.vs_dm6.bwaUniq.vcf.noMNP.tmp

vcftools --vcf variants/all_samples.vs_dm6.bwaUniq.vcf --keep-only-indels --recode --recode-INFO-all --s



vcftools --vcf variants/all_samples.vs_dm6.bwaUniq.vcf.noMNP.tmp --keep-only-indels --counts --out indel

cat indelCount.test.frq.count  | tail -n +2 | tr ":" "\t" | nl -n ln | head -n 1000 > dev/indelCount.tes
```

```
## [1] 10332
```

Hmmm, that seems like a lot. Let's get all the mainline chroms and look at them

```
cat indelCount.test.frq.count  | tail -n +2 | tr ":" "\t" | nl -n ln | grep -v Un | grep -v rand  > dev/
```

Hmm, yes 10k+ is a lot???

Corbin & Talia suggest a histogram of delta-bp for each indel...

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

spread by multiple columns: https://stackoverflow.com/questions/30592094/r-spreading-multiple-columns-with-tidyr

calculate delta-bp relative to the larger allele count or if it's 50/50, use the reference (allele 1)

ugh who knows, I'll figure it out tomorrow

13 March 2019

Ah, the variants with AC == 0 are examples of fixed differences vs. the dm6 reference.

It looks like the variants with delta_bp == 0 are cases of ??recognized SNPs?? being counted as indels?? what??

Maybe something like

```
cat variants/all_samples.vs_dm6.bwaUniq.vcf | ~/modules/vcflib/bin/vcfallelicprimitives | vcftools --vc
```

hmmm yes, this seems right.

maybe add some depth-filtering to improve genotype calls? avg depth is pretty high so it could be set stringent. The –minDP 10 flag in vcftools will do this (sets the called genotype to ./.; –counts then tallies this GT as uncalled.)

maybe something like

```
cat variants/all_samples.vs_dm6.bwaUniq.vcf | vcftools --vcf -  --minDP 10 --recode --recode-INFO-all --
```

??

Pi Day 2019

Hmmm, looks like vcfallelicprimatives breaks up MNPs and such, possibly unwanted. Maybe shunt the MNPs down a different path? "complex" repeats it doesn't seem to touch.

modding the tomato.count.frq.count file from yesterday...

```
cat variants/all_samples.vs_dm6.bwaUniq.vcf | vcftools --vcf -  --minDP 10 --recode --recode-INFO-all --
```

```
cat tomato.count.frq.count | tail -n +2 | tr ":" "\t" | nl -n ln | grep -v Un | grep -v rand  > dev/toma
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

Ok, I see part of the problem. the unite() pipe segment recasts the column as a string apparently?

15 March 2019

Apparently these are the offspring of a single mating between two flies? That makes the sharp peaks of the histogram more reasonable, but not the lopsided chromosomal distribution:

Getting some weird errors on the proposed pipeline so going through stepwise:

```
cat variants/all_samples.vs_dm6.bwaUniq.vcf | vcftools --vcf -  --minDP 10 --recode --recode-INFO-all --

 cat test.depthFiltered.recode.vcf | grep -v "TYPE=complex" | ~/modules/vcflib/bin/vcfallelicprimitives
```

hmm it looks like the sites which are broken down by vcfallelicprimitives are in a different format than the SNPs.

Also, vcfallelicprimitiives reclassifies some variants as complex so might as well exclude them afterwards. . . .

```
vcftools --vcf test.depthFiltered.simplified.primitive.vcf --keep-only-indels --recode --recode-INFO-al

vcftools --vcf test.depthFiltered.simplified.primitive.indelOnly.vcf   --counts --out test.depthFiltere

cat test.depthFiltered.simplified.primitive.indelOnly.count.frq.count | tail -n +2 | tr ":" "\t" | nl -
```

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

delta_bp
4
0
−4
0
10
20
30
mac
chrom
chr2L
chr2R
chr3L
chr3R
chr4
chrM
chrX
chrY

count
3000
2000
1000
0
0
10
20
30
mac
chrom
chr2L
chr2R
chr3L
chr3R
chr4
chrX

ok so this looks like some data sanity problems are fixed (eg no indels of length 0) but there are still some issues (eg the bizarre lopsided distribution of allele frequency across chromosomes. . . .)

Returning to the Danny Miller calls:

> mcm5-12, chrX:12825436, GAAA deletion mcm5-21, chr2R:12737873, A deletion mcm5-22, chr2R:8597548, T deletion mcm5-24, chr2L:21664793, TATATA deletion

Slopped these locations 100bp either way and intersected with the VCF:

```
cat misc/dannysCalls.prebed | cut -f 2 -d " " | tr -d "," | tr ":" "\t"  | awk '{print $1, $2-100, $2+1

edtools intersect -wa -wb -a misc/dannysCalls.bed -b test.depthFiltered.simplified.primitive.indelOnly.
```

The first, chrX:12825436, appears in the VCF but is called as a homozygote (both BWA an BWA-Uniq alignments agree here!)

The second, an A deletion at chr2R:12737873, appears in the VCF, appears to be heterozygous in two different flies (mcm5-21,mcm5-22). Also, there are two insertions (+A, +AA) at this site as well in the samples. (DfMcm5 and Mcm5-A7, respectively). Kind of a dodgy site even in the BWA-Uniq alignment??

The third, a T deletion at chr2R:8597548, appears in the VCF, heterozygous in a single individual. This is another site with 4 different insertion-deletion alleles called.

The fourth, a TATATA deletion at chr2L:21664793, doesn't appear in this VCF (filtered for depth, variants simplified, complex variants removed, indels only). However, it is picked up in the unfiltered BWA-Uniq VCF as a complex variant: TAC -> CAT. It is called as heterozygous in five individuals: mcm5-04,mcm5-03,mcm5-13,mcm5-18 mcm5-27,mcm5-19. mcm-24, instead of the TATATA deletion, is called as homozygous for the reference (the unfiltered BWA alignment for mcm5-24 has some 5 and 7bp indels but this still gets resolved as an MNP in the BWA-derived VCF. These reads are gone in BWA-Uniq.). The alignments give weak support for the existence of the TAC->CAT variant: the variant sites are there in the reads, but coverage is pretty low and the variation always seems to occur near the ends of the reads. In some cases the complex variant has been imputed from a single SNP near the end of a read.

18 March 2019

Nice! adds a help rule to a snakefile: https://www.biostars.org/p/220268/

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

total number of sites: 41143

```
## # A tibble: 5 x 2
##   chrom count
##   <fct> <int>
## 1 chr2L 11545
## 2 chr2R 10639
## 3 chr3L  9479
## 4 chr3R  9183
## 5 chr4    297
```

Logarithmic color scale in legend: https://stackoverflow.com/questions/8069837/is-there-a-built-in-way-to-do-a-logarithmic-col

Thought about a ggMarginal histogram but it wasn't doing much for me

INDEL size & type profile similar between the singletons and the sites at large.

20 March 2019

OKAY so it turns out that the reads I have been using are actually Talia's mcm5 mutants which explains some things I guess.

Updating the YAML to reflect this; they are now in the subgroup "mutants".

The flies from (Miller et al. 2016) are on NCBI as PRJNA285112 (isogenized parental flies, w1118 and Canton S) and PRJNA307070 (~200 individual male offspring). Since these are parent/offspring comparisons, maybe add a new tag (pedigree: parent/child)?

Per NCBI, the mating was "mother w5 and isogenic w[1118] father"

Hmm, looks like the offspring were run on 4 lanes and thus 4 different SRAs will need to be downloaded and merged per sample :(

Uh.... fastq-dump says

> fastq-dump –split-3 SRR2044310 Rejected 431205 READS because of filtering out non-biological READS Read 25541271 spots for SRR2044310 Written 25541271 spots for SRR2044310

... which is great and everything but I didn't set –skip-technical or –split-spot? Are they automatically invoked by –split-3????? >Filters for individual reads Applied only with –split-spot set –skip-technical Dump only biological reads

Anyway the outcome is three files, two of paired reads and one of unpaired: >$ ls SRR2044310_1.fastq SRR2044310_2.fastq SRR2044310.fastq

21 March 2019

All of the DM flies are males except for one of the parents. What about Talia's flies? Looking at the breadth of coverage of the chrY, the median is ~30% covered, with two exceptions ~2%. By contrast, chrX was consistently ~90% covered. Talia says they are all Males.

The DM flies were subsampled down to 28 for a balanced comparison.

UGHHHH adding the SRA tags gives each of up to four SRAs its own line!!!

```
##       name paired                    path    readsfile1    readsfile2
## 1   DfMcm5   TRUE  FASTQ/mutants/Df-Mcm5/ Df-Mcm5.R1.fq Df-Mcm5.R2.fq
## 2  mcm5-01   TRUE  FASTQ/mutants/mcm5-01/ mcm5-01.R1.fq mcm5-01.R2.fq
## 3  mcm5-02   TRUE  FASTQ/mutants/mcm5-02/ mcm5-02.R1.fq mcm5-02.R2.fq
## 4  mcm5-03   TRUE  FASTQ/mutants/mcm5-03/ mcm5-03.R1.fq mcm5-03.R2.fq
## 5  mcm5-04   TRUE  FASTQ/mutants/mcm5-04/ mcm5-04.R1.fq mcm5-04.R2.fq
## 6  mcm5-05   TRUE  FASTQ/mutants/mcm5-05/ mcm5-05.R1.fq mcm5-05.R2.fq
## 7  mcm5-06   TRUE  FASTQ/mutants/mcm5-06/ mcm5-06.R1.fq mcm5-06.R2.fq
## 8  mcm5-07   TRUE  FASTQ/mutants/mcm5-07/ mcm5-07.R1.fq mcm5-07.R2.fq
## 9  mcm5-08   TRUE  FASTQ/mutants/mcm5-08/ mcm5-08.R1.fq mcm5-08.R2.fq
## 10 mcm5-09   TRUE  FASTQ/mutants/mcm5-09/ mcm5-09.R1.fq mcm5-09.R2.fq
## 11 mcm5-10   TRUE  FASTQ/mutants/mcm5-10/ mcm5-10.R1.fq mcm5-10.R2.fq
## 12 mcm5-11   TRUE  FASTQ/mutants/mcm5-11/ mcm5-11.R1.fq mcm5-11.R2.fq
## 13 mcm5-12   TRUE  FASTQ/mutants/mcm5-12/ mcm5-12.R1.fq mcm5-12.R2.fq
## 14 mcm5-13   TRUE  FASTQ/mutants/mcm5-13/ mcm5-13.R1.fq mcm5-13.R2.fq
## 15 mcm5-14   TRUE  FASTQ/mutants/mcm5-14/ mcm5-14.R1.fq mcm5-14.R2.fq
## 16 mcm5-15   TRUE  FASTQ/mutants/mcm5-15/ mcm5-15.R1.fq mcm5-15.R2.fq
## 17 mcm5-16   TRUE  FASTQ/mutants/mcm5-16/ mcm5-16.R1.fq mcm5-16.R2.fq
## 18 mcm5-17   TRUE  FASTQ/mutants/mcm5-17/ mcm5-17.R1.fq mcm5-17.R2.fq
## 19 mcm5-18   TRUE  FASTQ/mutants/mcm5-18/ mcm5-18.R1.fq mcm5-18.R2.fq
## 20 mcm5-19   TRUE  FASTQ/mutants/mcm5-19/ mcm5-19.R1.fq mcm5-19.R2.fq
## 21 mcm5-20   TRUE  FASTQ/mutants/mcm5-20/ mcm5-20.R1.fq mcm5-20.R2.fq
## 22 mcm5-21   TRUE  FASTQ/mutants/mcm5-21/ mcm5-21.R1.fq mcm5-21.R2.fq
## 23 mcm5-22   TRUE  FASTQ/mutants/mcm5-22/ mcm5-22.R1.fq mcm5-22.R2.fq
## 24 mcm5-23   TRUE  FASTQ/mutants/mcm5-23/ mcm5-23.R1.fq mcm5-23.R2.fq
## 25 mcm5-24   TRUE  FASTQ/mutants/mcm5-24/ mcm5-24.R1.fq mcm5-24.R2.fq
## 26 mcm5-25   TRUE  FASTQ/mutants/mcm5-25/ mcm5-25.R1.fq mcm5-25.R2.fq
## 27 mcm5-26   TRUE  FASTQ/mutants/mcm5-26/ mcm5-26.R1.fq mcm5-26.R2.fq
## 28 mcm5-27   TRUE  FASTQ/mutants/mcm5-27/ mcm5-27.R1.fq mcm5-27.R2.fq
## 29 mcm5-28   TRUE  FASTQ/mutants/mcm5-28/ mcm5-28.R1.fq mcm5-28.R2.fq
## 30 Mcm5-A7   TRUE  FASTQ/mutants/Mcm5-A7/ Mcm5-A7.R1.fq Mcm5-A7.R2.fq
## 31 CantonS   TRUE FASTQ/controls/CantonS/ CantonS.R1.fq CantonS.R2.fq
## 32   w1118   TRUE   FASTQ/controls/w1118/   w1118.R1.fq   w1118.R2.fq
## 33   w5_16   TRUE   FASTQ/controls/w5_16/   w5_16.R1.fq   w5_16.R2.fq
## 34   w15_9   TRUE   FASTQ/controls/w15_9/   w15_9.R1.fq   w15_9.R2.fq
## 35    w4_8   TRUE    FASTQ/controls/w4_8/    w4_8.R1.fq    w4_8.R2.fq
## 36   cs2_6   TRUE   FASTQ/controls/cs2_6/   cs2_6.R1.fq   cs2_6.R2.fq
## 37   w4_17   TRUE   FASTQ/controls/w4_17/   w4_17.R1.fq   w4_17.R2.fq
## 38   w3_12   TRUE   FASTQ/controls/w3_12/   w3_12.R1.fq   w3_12.R2.fq
## 39  cs12_5   TRUE  FASTQ/controls/cs12_5/  cs12_5.R1.fq  cs12_5.R2.fq
## 40    w4_3   TRUE    FASTQ/controls/w4_3/    w4_3.R1.fq    w4_3.R2.fq
## 41   w15_8   TRUE   FASTQ/controls/w15_8/   w15_8.R1.fq   w15_8.R2.fq
## 42  w13_13   TRUE  FASTQ/controls/w13_13/  w13_13.R1.fq  w13_13.R2.fq
## 43   w5_15   TRUE   FASTQ/controls/w5_15/   w5_15.R1.fq   w5_15.R2.fq
## 44   w12_7   TRUE   FASTQ/controls/w12_7/   w12_7.R1.fq   w12_7.R2.fq
```

```
## 45   w3_24   TRUE   FASTQ/controls/w3_24/   w3_24.R1.fq   w3_24.R2.fq
## 46    w4_7   TRUE    FASTQ/controls/w4_7/    w4_7.R1.fq    w4_7.R2.fq
## 47  w11_13   TRUE  FASTQ/controls/w11_13/  w11_13.R1.fq  w11_13.R2.fq
## 48   w12_1   TRUE   FASTQ/controls/w12_1/   w12_1.R1.fq   w12_1.R2.fq
## 49  w12_12   TRUE  FASTQ/controls/w12_12/  w12_12.R1.fq  w12_12.R2.fq
## 50    w4_6   TRUE    FASTQ/controls/w4_6/    w4_6.R1.fq    w4_6.R2.fq
## 51   w3_16   TRUE   FASTQ/controls/w3_16/   w3_16.R1.fq   w3_16.R2.fq
## 52   w15_6   TRUE   FASTQ/controls/w15_6/   w15_6.R1.fq   w15_6.R2.fq
## 53   cs8_8   TRUE   FASTQ/controls/cs8_8/   cs8_8.R1.fq   cs8_8.R2.fq
## 54   w13_1   TRUE   FASTQ/controls/w13_1/   w13_1.R1.fq   w13_1.R2.fq
## 55   w12_5   TRUE   FASTQ/controls/w12_5/   w12_5.R1.fq   w12_5.R2.fq
## 56   w5_17   TRUE   FASTQ/controls/w5_17/   w5_17.R1.fq   w5_17.R2.fq
## 57   w15_7   TRUE   FASTQ/controls/w15_7/   w15_7.R1.fq   w15_7.R2.fq
## 58  cs12_3   TRUE  FASTQ/controls/cs12_3/  cs12_3.R1.fq  cs12_3.R2.fq
## 59   cs2_7   TRUE   FASTQ/controls/cs2_7/   cs2_7.R1.fq   cs2_7.R2.fq
## 60   w13_4   TRUE   FASTQ/controls/w13_4/   w13_4.R1.fq   w13_4.R2.fq
##    experimental source subgroups pedigree  sex
## 1        mutant  talia       all   parent <NA>
## 2        mutant  talia       all    child    M
## 3        mutant  talia       all    child    M
## 4        mutant  talia       all    child    M
## 5        mutant  talia       all    child    M
## 6        mutant  talia       all    child    M
## 7        mutant  talia       all    child    M
## 8        mutant  talia       all    child    M
## 9        mutant  talia       all    child    M
## 10       mutant  talia       all    child    M
## 11       mutant  talia       all    child    M
## 12       mutant  talia       all    child    M
## 13       mutant  talia       all    child    M
## 14       mutant  talia       all    child    M
## 15       mutant  talia       all    child    M
## 16       mutant  talia       all    child    M
## 17       mutant  talia       all    child    M
## 18       mutant  talia       all    child    M
## 19       mutant  talia       all    child    M
## 20       mutant  talia       all    child    M
## 21       mutant  talia       all    child    M
## 22       mutant  talia       all    child    M
## 23       mutant  talia       all    child    M
## 24       mutant  talia       all    child    M
## 25       mutant  talia       all    child    M
## 26       mutant  talia       all    child    M
## 27       mutant  talia       all    child    M
## 28       mutant  talia       all    child    M
## 29       mutant  talia       all    child    M
## 30       mutant  talia       all   parent <NA>
## 31      control  danny       all   parent    F
## 32      control  danny       all   parent    M
## 33      control  danny       all    child    M
## 34      control  danny       all    child    M
## 35      control  danny       all    child    M
## 36      control  danny       all    child    M
## 37      control  danny       all    child    M
```

```
## 38      control  danny       all    child    M
## 39      control  danny       all    child    M
## 40      control  danny       all    child    M
## 41      control  danny       all    child    M
## 42      control  danny       all    child    M
## 43      control  danny       all    child    M
## 44      control  danny       all    child    M
## 45      control  danny       all    child    M
## 46      control  danny       all    child    M
## 47      control  danny       all    child    M
## 48      control  danny       all    child    M
## 49      control  danny       all    child    M
## 50      control  danny       all    child    M
## 51      control  danny       all    child    M
## 52      control  danny       all    child    M
## 53      control  danny       all    child    M
## 54      control  danny       all    child    M
## 55      control  danny       all    child    M
## 56      control  danny       all    child    M
## 57      control  danny       all    child    M
## 58      control  danny       all    child    M
## 59      control  danny       all    child    M
## 60      control  danny       all    child    M
```

> Males were numbered based on whether their father was homozygous w1118 or Canton-S and the number of their het- erozygous mother. For example, male cs12.3 had a Canton-S father, its mother was female number 12, and it was the third male selected for DNA extraction. Sibling numbers may not be continuous, as males with low DNA concentrations after DNA extraction were not selected for sequencing.

What does this mean as far as sample subset choice????

Need to rewrite the joint VCF rule to call variants by group; can always summon the whole-sample joint call via the "all" group

Will the VCF comparison & contrast section be at all relevant when the two experimental treatments are unrelated? (If I'm using BWA-Uniq, I'll still want to compare to BWA, even if I ultimately don't use the BWA alignment for variant calling)

Also, what about using parental variants to validate/mask child variants?

22 March 2019

> can you look at the indel frequency distribution without chromosome 3 for mcm21-28 samples? do not include mcm5-A7 and DfMcm5.

To answer this: generalize the current VCF_winnower rule, with preprocessing the input vcf to restrict the sample?

```
cp variants/mutant.vs_dm6.bwaUniq.vcf variants/potato.vs_dm6.bwaUniq.vcf

snakemake -n -p analysis/potato.alleleCounts.simpleIndels.dpthFilt.biallelic.universal.out
```

> Building DAG of jobs... InputFunctionException in line 250 of /proj/cdjones_lab/csoeder/EJgrepper/Snakefile: KeyError: 'potato'

Hmm, looks like snakemake actually needs it to be a listed group in the config.yaml. But will it accept the extant variants/potato.vs_dm6.bwaUniq.vcf or try to force an new calling for the new group?

Using an updated config file:

```
keepers=$(for i in {1..8}; do echo --indv mcm5-2"$i"; done  | tr "\n" " ")

vcftools --vcf variants/mutant.vs_dm6.bwaUniq.vcf $keepers --recode --recode-INFO-all --stdout >  varia

snakemake  --configfile config.tmp.yaml -n -p analysis/potato.alleleCounts.simpleIndels.dpthFilt.biallel
```

Works… well actually no the sample number filter is currently hardcoded into the Winnower rule. But copypasting the snakemake verboss output and modding (also means we can filter the chroms now rather than later)

```
vcftools --chr chr2L --chr chr2R --chr chrX --vcf variants/potato.vs_dm6.bwaUniq.vcf --minDP 10 --recod
```

okay!!

```
## Warning: attributes are not identical across measure variables;
## they will be dropped
```

Filtering as before, there are 25325 sites:

Table 9: Quality Biallaeleic INDELs per Chromosome

chrom	count
chr2L	12004
chr2R	11040
chrX	2281

The sudden appearance of variants with a MAC of 0 is because when the starting VCF is subsetted by individual, variant sites are retained even if all subsetted individuals are 0/0. Kicking out all sites with an alternate allele count of 0. . . .

With non-variable sites removed there are 15717 sites remaining:

Table 10: Quality Biallaeleic INDELs per Chromosome

chrom	count
chr2L	7094
chr2R	6486
chrX	2137

Heatmap of INDEL Count, by Frequency and Size
Nucleotides Inserted or Deleted
4
0
−4
0
1
2
3
4
5
6
7
8
Minor Allele Count (chromosomes)
Site Count
1000
100
10
1

Number of Quality INDELs, by Minor Allele Count and Chromosome
Number of Sites
12000
9000
6000
3000
0
0
1
2
3
4
5
6
7
8
Minor Allele Count (chromosomes)
chrom
chr2L
chr2R
chrX

Density of Quality INDELs, by Minor Allele Count and Chromosome
Site Density
0.0
0.2
0.4
0.6
0
1
2
3
4
5
6
7
8
Minor Allele Count (chromosomes)
chrom
chr2L
chr2R
chrX

INDEL Count by Size
Number of Sites
0
1000
2000
3000
4000
5000
−8
−4
0
4
8
Nucleotides Inserted/Deleted
chrom
chr2L
chr2R
chrX

Singletons:

Table 11: Distribution of Singleton INDELs

chrom	count
chr2L	400
chr2R	284
chrX	162

But, I'm not sure why there are singletons appearing on the X? Being males there is only one chrX so it would be read as homozygous at all points (and, the MAC for chrX is exaggerated by 2x).

Looking at the allele counts mod 2 will give a lower bound to the number of sites on chrX with a heterozygote called (it won't reveal cases in which there are an even number of heterozygotes)

```
## # A tibble: 2 x 2
##     mod count
##   <dbl> <int>
## 1     0  1593
## 2     1   544
```

Using the –het flag with vcftools to take another look??

```
vcftools --chr chrX --vcf variants/potato.vs_dm6.bwaUniq.vcf --minDP 10 --recode --recode-INFO-all --st
```

Bibliography

McVey, Mitch, and Sang Eun Lee. 2008. "MMEJ repair of double-strand breaks (director's cut): deleted sequences and alternative endings." *Trends in Genetics* 24 (11): 529–38. doi:10.1016/j.tig.2008.08.007.

Miller, Danny E., Clarissa B. Smith, Nazanin Yeganeh Kazemi, Alexandria J. Cockrell, Alexandra V. Arvanitakis, Justin P. Blumenstiel, Sue L. Jaspersen, and R. Scott Hawley. 2016. "Whole-genome analysis of individual meiotic events in Drosophila melanogaster reveals that noncrossover gene conversions are insensitive to interference and the centromere effect." *Genetics* 203 (1): 159–71. doi:10.1534/genetics.115.186486.