Summary Part 4

- Location => Look out for Feature
- Object creation from Adaptor or from API object
- No allele strings for structural variations, only coordinates
- VariationFeature already contains a lot of useful information to study consequences BUT:
  - Go further down to TranscriptVariation
    - Location specific information of Variant overlapping a Transcript
  - And TranscriptVariationAllele
    - Allele specific information on Transcript
- PhenotypeFeatures can be obtained from Variation and StructuralVariation objects
Linkage disequilibrium

- Linkage disequilibrium (LD) is a measure of how frequently alleles at two separate loci are inherited together on the same haplotype.

- Two common measures
  - $r^2$, $D'$ ($r^2 = 1 \rightarrow$ perfect LD)

![Diagram showing high and lower LD in two populations](chart.png)

**Population A**
- Individual A1, Locus 1: T, Locus 2: T
- Individual A2, Locus 1: T, Locus 2: G
- Individual A3, Locus 1: T, Locus 2: G
- Individual A4, Locus 1: T, Locus 2: G
- Individual A5, Locus 1: T, Locus 2: G
- Individual A6, Locus 1: T, Locus 2: G

**Population B**
- Individual B1, Locus 3: T, Locus 4: G
- Individual B2, Locus 3: T, Locus 4: G
- Individual B3, Locus 3: T, Locus 4: G
- Individual B4, Locus 3: T, Locus 4: G
- Individual B5, Locus 3: T, Locus 4: G
- Individual B6, Locus 3: T, Locus 4: G

**High LD**
- The alleles at Locus 1 and Locus 2 are more frequently inherited together than expected by chance.

**Lower LD**
- The alleles at Locus 3 and Locus 4 are less frequently inherited together than expected by chance.
Linkage disequilibrium container objects

- **LDFeatureContainer (LDFeatureContainerAdaptor)**
- Contains pairwise LD values in a region
- Can contain values for multiple populations
- Values are calculated on the fly
- Most methods return hash references:

```perl
my $ldf_container = $ldfca->fetch_by_Slice($slice);  # returns listref of hashrefs
my $ld_values = $ldf_container->get_all_ld_values();
foreach my $ld_hash_ref (@$ld_values) {
    my $r2 = $ld_hash_ref->{r2};  # look up type
    my $variation_1 = $ld_hash_ref->{variation1};
    ...
}
```
Variation sets

- Arbitrary collections of Variations (different from the sources)
- Useful to limit your script to important subsets
  - All Variations linked to entries in OMIM
  - Variants on Illumina chips
  - ...
- Sometimes contain millions of Variations, in which case you can fetch an Iterator instead of a list

```perl
my $vs = $variation_set_adaptor->fetch_all_by_name('All phenotype-associated variants');
print $vs->description();
my @variations = @{$vs->get_all_Variations()};
my @sub_sets = @{$vs->get_all_sub_VariationSets()};
```

[Link to variation data description](http://www.ensembl.org/info/docs/variation/data_description.html#variation_sets)
What next

- Ensembl tool to predict the functional consequences of variants, using the Ensembl Variation API
- 2 ways to use it:
  - Through the web interface
  - Using the standalone perl script
- Query the databases or a downloaded cache file
- Several input formats: VEP tabulated format, VCF, Pileup, HGVS identifiers, Variant identifiers
- A lot of options available (data filters, output formats)

http://www.ensembl.org/info/docs/variation/vep/index.html
Summary

  - Data
  - Database
  - Perl API
  - VEP
- Ensembl developers mailing list
  - dev@ensembl.org
Ensembl Acknowledgements

Ensembl Team

• Ensembl Variation
  • William McLaren
  • Laurent Gil
  • Sarah Hunt
  • Fiona Cunningham (head)

• Paul Flicek, Steve Searle and the entire Ensembl Team

Funding

European Commission
Framework Programme 7

Quantomics
From Sequence to Consequence: Tools for the Exploitation of Livestock Genomes
Figures

- Path 2: http://www.photographyblogger.net/wp-content/uploads/2011/05/83594459_70d9688f23.jpg
- Path 3: http://liveholiness.com/wp-content/uploads/2012/02/path.jpg