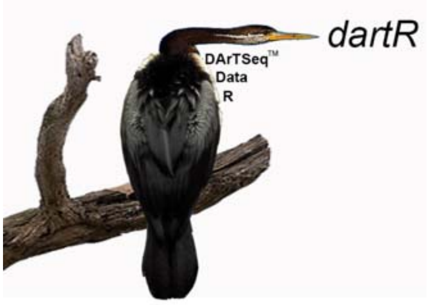
Wildlife Genomics@UC
wildlifegenomics.biomatix.org

UNIVERSITY OF CANBERRA

SNP Analysis using dartR



How dartR stores your data and data input

Version 1.8.3

Copies of these workshop notes are available from:

The Institute for Applied Ecology
University of Canberra ACT 2601
Australia

Email: **georges@aerg.canberra.edu.au**

Contents

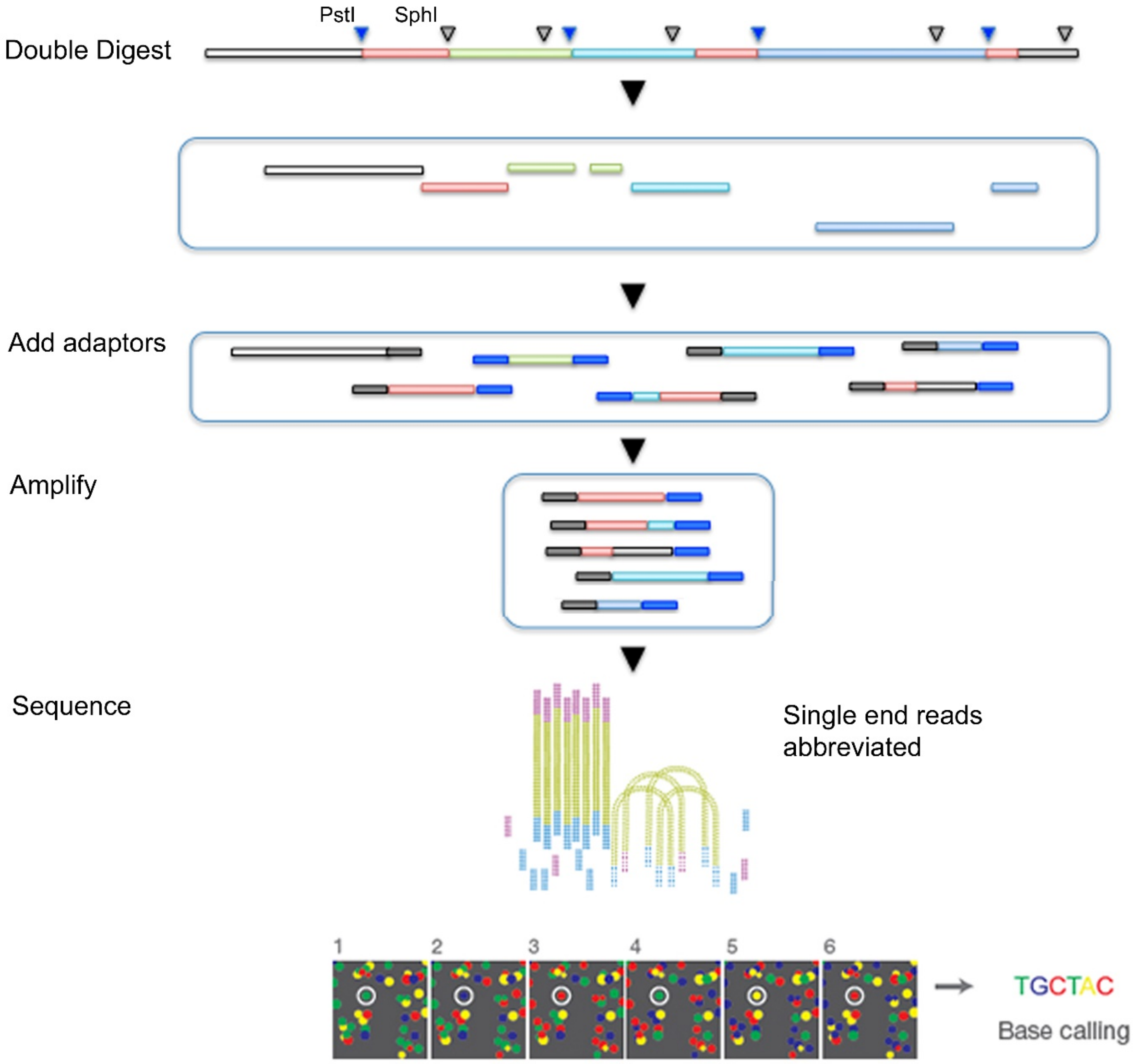
Session 1: Introduction to DArTSeq

Sequencing

Diversity Arrays Technology Pty Ltd (DArT PL) is a private company that specializes in genotyping by sequencing. Their approach is one of genome complexity reduction. But what does this mean?

Basically, DArTSeq is a method that digests genomic DNA using pairs of restriction enzymes (cutters). When the DNA is cut at two locations within a reasonable distance of each other, the fragment is available for sequencing using the Illumina short-read platforms.

The first step in the process involves the selection of restriction enzymes that provide the best balance between getting adequate fraction of the genome represented, an adequate read depth for each fragment, and adequate levels of polymorphism. This is species specific and so requires some initial optimization.



Once the best restriction enzymes are selected, say PstI (recognition sequence 5’-CTGCA|G-3’) and SphI (5’-GCATG|C-3’), then the DNA is digested, and various adaptors added to the sequence fragments to allow Illumina short-read sequencing to proceed. These additional terminal sequences include a barcode to allow disaggregation of the sequences for each sample during later analysis.

The fragments of DNA selected by this process are sequenced in an abbreviated process to yield a set of raw “sequence tags” each of around 75 bp. They are filtered on sequence quality, particularly in the barcode region, truncated to 69 bp and stacked based on sequence similarity. A series of proprietary filters are then applied to select those sequence tags that include a reliable SNP marker.

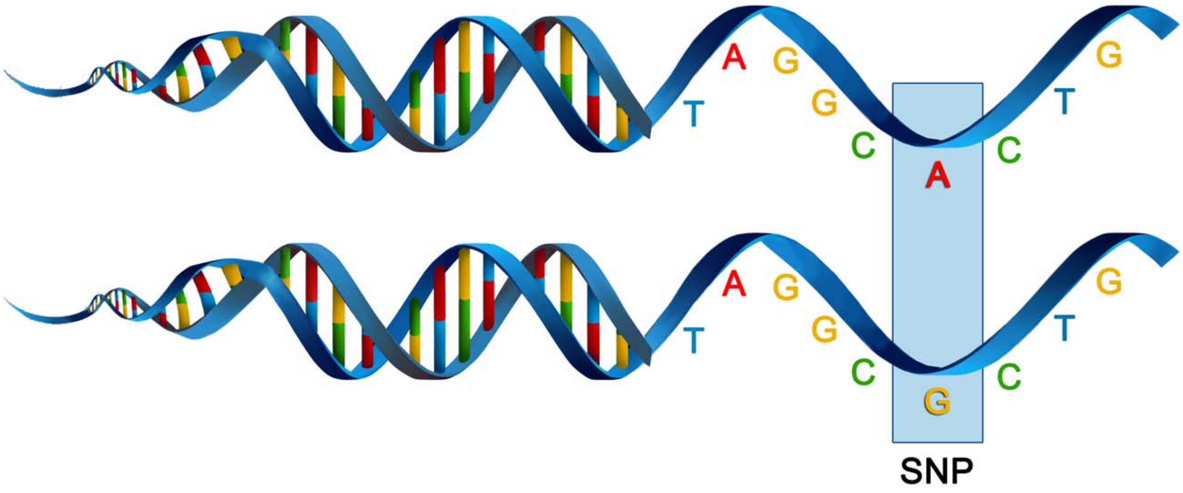
In particular, one third of samples are processed twice as technical replicates, from DNA and using independent adaptors, through to allelic calls. Scoring consistency (repeatability) is used as the main selection criterion for high quality/low error rate markers.

These DArT analysis pipelines have been tested against hundreds of controlled crosses to verify mendelian behaviour of the resultant SNPs as part of their commercial operations.

When you come to publish, you may receive requests to be more elaborative than you are able to, because of the proprietary nature of the pipelines. DArT Pty Ltd is a private company and needs to hold some of its proprietary analyses inhouse. Note that other companies with whom you interact, including Illumina, do the same. The work is reproducible in that using the same service/equipment on the same samples will yield the same result. Most journals accept this.

The SNP dataset

SNPs, or single nucleotide polymorphisms, are single base pair mutations at a nuclear locus. That nuclear locus is represented in the dataset by two sequence tags which, at a heterozygous locus, take on two allelic states, one referred to as the reference state, the other as the alternate or SNP state.



Because it is extremely rare for a mutation to occur twice at the same site in the genome (perhaps with the exception of Eucalypts), the SNP data are effectively biallelic.

The data can be represented in a table of SNP bases (A, T, C or G), with two states for each individual at each locus in a diploid organisms.

	Ind 01	Ind 02	Ind 03	Ind 04	Ind 05	Ind 06	Ind 07	Ind 08	Ind 09	Ind 10
Locus 1	A/A	A/A	A/A	A/A	A/G	A/A	A/A	A/A	A/A	-/-
Locus 2	C/C	C/C	C/C	C/C	C/C	C/C	C/T	C/C	C/C	C/C
Locus 3	C/G	G/G	G/G	G/G	G/G	C/C	C/C	C/C	C/C	C/C
Locus 4	A/A	A/T	A/A	A/T	T/T	A/A	A/A	A/A	A/A	A/A
Locus 5	A/A	A/A	A/A	A/A	-/-	A/G	A/A	A/A	A/A	A/A
Locus 6	C/C	C/C	C/C	C/C	C/C	C/C	C/T	C/C	C/C	C/C
Locus 7	C/G	G/G	G/G	G/G	G/G	C/C	C/C	C/C	C/C	C/C
Locus 8	A/A	A/T	A/A	A/T	T/T	A/A	A/A	A/A	A/A	A/A
Locus 9	A/A	A/A	A/A	A/A	A/A	A/A	A/A	A/A	A/A	A/A
Locus 10	C/C	C/C	C/C	C/C	C/C	C/C	C/T	C/C	C/C	C/C
Locus 11	C/G	G/G	G/G	G/G	G/G	C/C	C/C	C/C	C/C	C/C

Alternatively, because the data are biallelic, it is convenient to code the data as coded 0 for homozyogotes for one allele, 1 for heterozygotes, and 2 for homozygotes of the other allele.

The reference allele is arbitrarily taken to be the most common allele, so 0 is the score for homozygous reference, and 2 is the score for homozygous alternate or SNP state. NA indicates that the SNP could not be scored.

	Ind01	Ind02	Ind03	Ind04	Ind05	Ind06	Ind07	Ind08	Ind09	Ind10
Locus 1	0	0	0	0	1	0	0	0	0	NA
Locus 2	0	0	0	0	0	0	1	0	0	0
Locus 3	1	2	2	2	2	0	0	0	0	0
Locus 4	0	1	0	1	2	0	0	0	0	0
Locus 5	0	0	0	0	NA	1	0	0	0	0
Locus 6	0	0	0	0	0	0	1	0	0	0
Locus 7	1	2	2	2	2	0	0	0	0	0
Locus 8	0	1	0	1	2	0	0	0	0	0
Locus 9	0	0	0	0	0	0	0	0	0	0
Locus 10	0	0	0	0	0	0	1	0	0	0
Locus 11	1	2	2	2	2	0	0	0	0	0

This is the form the data are stored in in dartR, though note that it departs from the coding arrangement used by DArT PL.

Some sequence tags might contain more than one SNP, in which case they are likely to be closely linked when passed from parent to offspring. These may need consideration when preparing your data for analysis.

The SNP data are provided in two forms by DArT, which will be described later.

SilicoDArT

As well as individuals varying in allelic composition at SNP sites, they can vary at the restriction sites used to pull the representation from the genome. A mutation at one or both of the restriction sites will result in allelic drop-out or null alleles. The presence or absence of particular sequence tags across individuals provides a source of information additional to the SNP data.

Broadly, SilicoDArT markers can be considered analogous to AFLPs (Amplified Fragment Length Polymorphisms).

DArT PL provide this second dataset, the presence or absence of scored sequence tags across individuals in what it calls the SilicoDArT dataset. The filtering pipeline applied to generate these data has been highly optimized for reliability, so do not be tempted to use the null alleles (missing data) present in the SNP dataset.

	Ind01	Ind02	Ind03	Ind04	Ind05	Ind06	Ind07	Ind08	Ind09	Ind10
Locus 01	0	1	0	0	0	0	1	1	0	1
Locus 02	1	0	1	1	1	1	1	0	0	1
Locus 03	1	1	0	1	0	0	0	0	1	0
Locus 04	1	1	0	1	1	0	1	0	0	0
Locus 05	0	0	0	0	1	NA	1	0	1	0
Locus 06	1	1	1	0	1	1	1	0	1	1
Locus 07	1	1	NA	0	0	1	1	0	1	0
Locus 08	1	0	1	0	0	1	1	0	1	1
Locus 09	1	1	0	0	0	1	1	1	0	1
Locus 10	0	0	1	1	0	1	1	1	NA	1
Locus 11	0	1	1	0	1	1	0	0	1	1

Note that unlike the SNP data, NA represents a truly missing value, in that it could not be determined if a sequence tag was present or absent.

Where have we come?

The above Session was designed to give you a very brief overview to the Diversity Arrays Technology pipelines for producing SNP and associated data. Having completed this Session, you should now be familiar the following concepts.

- The concept of a SNP marker and how they are generated.
- The distinction between DArTSeq and SilicoDArT datasets.
- The coding used for SNP genotypes – 0 for homozygous reference, 2 for homozygous alternate, 1 for heterozygous, and NA for 'missing'.
- The coding used for SilicoDArT genotypes – 0 for absent, 1 for present, and NA for missing.

Further reading

Sansaloni, C., Petroli, C., Jaccoud, D., Carling, J., Detering, F., Grattapaglia, D., & Kilian, A. (2011). Diversity Arrays Technology (DArT) and next-generation sequencing combined: genome-wide, high throughput, highly informative genotyping for molecular breeding of *Eucalyptus*. BMC Proceedings 5(Suppl 7), P54. doi:10.1186/1753-6561-5-S7-P54.

Kilian, A., Wenzl, P., Huttner, E., Carling, J., Xia, L., Blois, H., … Uszynski, G. (2012). Diversity arrays technology: a generic genome profiling technology on open platforms. *Methods in Molecular Biology 888*:67–89.

Session 2: Getting data into dartR

If you are coming back to us, create or load a project, set a working directory, and do not forget to set the default directory for files, the outpath, to getwd(). Refer to the previous Session.

A sensible pipeline

Let us begin by jumping the gun and defining a sensible pipeline for entering your data, as a context for the material in this and subsequent Sessions.

1. Examine the data provided by DArT PL in Excel to confirm that it conforms to expectations of the dartR package.

   For the SNP data, there needs to be a `AlleleID` column, and the locus metadata needs to finish with the column `RepAvg`. The row with the locus metadata labels needs to be the same row that holds the individual (= specimen or sample labels). This is usually the case, but some older datasets may need a little modification.

   For the SilicoDArt data, there needs to be a `CloneID` column and the locus metadata needs to finish with the column `Reproducibility`.

2. Prepare the metadata associated with each individual. This dataset, stored in csv format, contains at a minimum the individual/specimen labels in a column headed `id`, and a population column that assigns individuals to groups or populations in a column headed `pop`. Other columns are optional, but might include latitude, longitude of capture, sex, or other possible groupings of the individuals.

3. Read the data into dartR

How dartR stores SNP data

The R package dartR relies on the SNP data being stored in a compact form using a bit-level coding scheme. SNP data coded in this way are held in a genlight object that is defined in R package adegenet (Jombart, 2008; Jombart and Ahmed, 2011). Refer to the tutorial prepared by Jombart and Collinson (2015), called *Analysing genome-wide SNP data using adegenet 2.0.0*, if you require further information.

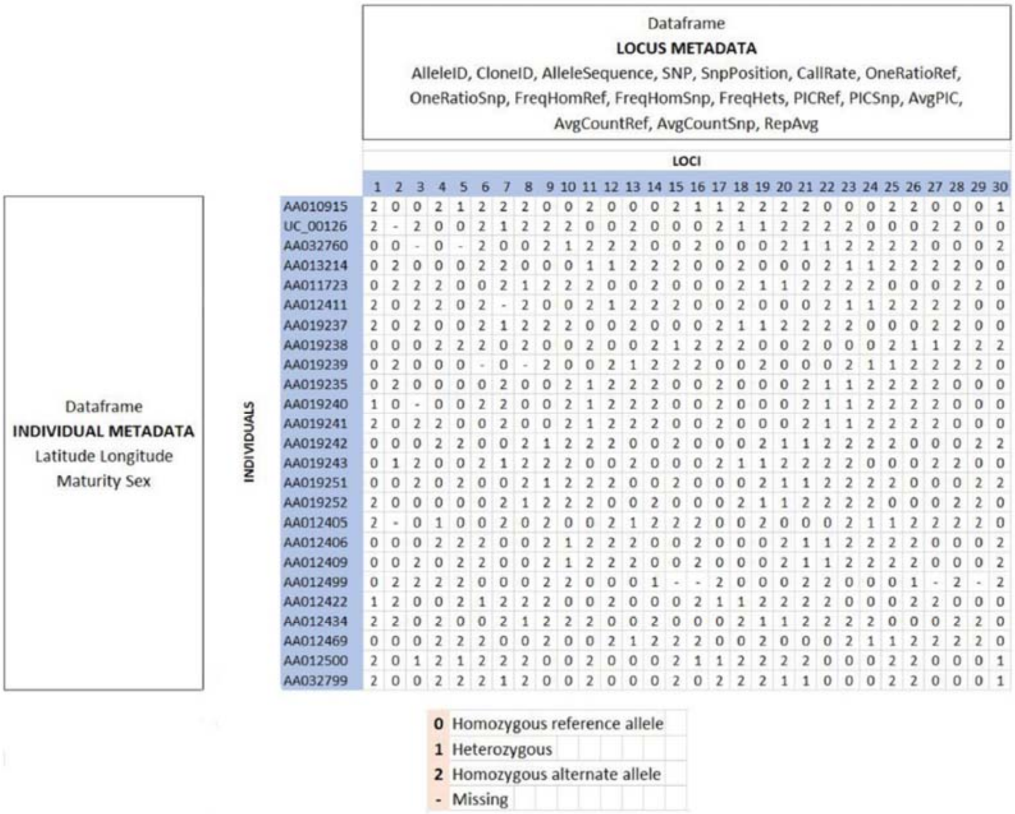
The complex storage arrangement of genlight objects is hidden from the user because it is accompanied by a number of “accessors”. These allow the data to be accessed in a way similar to the manipulation of standard objects in R, such as lists, vectors and matrices.

A genlight object can be considered to be a matrix containing the SNP data encoded in a particular way. The matrix entities (rows) are the individuals, and the attributes (columns) are the SNP loci. In the body of this individual x locus matrix are the SNP data, coded as 0 for homozygous reference state, 1 for heterozygous, and 2 for homozygous alternate (or SNP) state.

The genelight objects used in dartR not only have the SNP data, but also allow for attachment of locus metadata to the loci, and attachment of individual metadata to the individuals/samples. This is represented diagrammatically below.

Locus metadata

The locus metadata included in the genlight object are those provided as part of your DArT PL report. These metadata are obtained from the DArT PL csv file when it is read in to the genlight object. The locus metadata are held in an R data.frame that is associated with the SNP data as part of the genlight object.


Dataframe
LOCUS METADATA
AlleleID, CloneID, AlleleSequence, SNP, SnpPosition, CallRate, OneRatioRef, OneRatioSnp, FreqHomRef, FreqHomSnp, FreqHets, PICRef, PICSnp, AvgPIC, AvgCountRef, AvgCountSnp, RepAvg
Dataframe
INDIVIDUAL METADATA
Latitude Longitude
Maturity Sex
INDIVIDUALS
LOCI

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
AA010915	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0	0	0	1
UC_00126	2	-	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0	0
AA032760	0	0	-	0	-	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2
AA013214	0	2	0	0	0	2	2	0	0	0	1	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0
AA011723	0	2	2	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0
AA012411	2	0	2	2	0	2	-	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0
AA019237	2	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0	0
AA019238	0	0	0	2	2	2	0	2	0	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2
AA019239	0	2	0	0	0	-	0	-	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0
AA019235	0	2	0	0	0	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0
AA019240	1	0	-	0	0	2	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0
AA019241	2	0	2	2	0	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0
AA019242	0	0	0	2	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2
AA019243	0	1	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0	0
AA019251	0	0	2	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2
AA019252	2	0	0	0	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0
AA012405	2	-	0	1	0	0	2	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0
AA012406	0	0	0	2	2	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2
AA012409	0	0	2	0	2	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2
AA012499	0	2	2	2	2	0	0	0	2	2	0	0	0	1	-	-	2	0	0	0	2	2	0	0	0	1	-	2	-	2
AA012422	1	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0	0	0
AA012434	2	2	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0
AA012469	0	0	0	2	2	2	0	0	2	0	0	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0
AA012500	2	0	1	2	1	2	2	2	0	0	2	0	0	0	2	1	1	2	2	2	2	0	0	0	2	2	0	0	0	1
AA032799	2	0	0	2	2	2	1	2	0	0	2	0	0	0	2	0	2	2	2	1	1	0	0	0	2	2	0	0	0	1

0	Homozygous reference allele
1	Heterozygous
2	Homozygous alternate allele
-	Missing

The locus metadata would typically include:

SNP the mutational change and its position in the sequence tag, referenced from zero

SnpPosition position (zero is position 1) in the sequence tag of the defined SNP variant base

TrimmedSequence (optional)The sequence containing the SNP or SNPs (the sequence tag), trimmed of adaptors.

CallRate proportion of samples for which the genotype call is non-missing (that is, not “-”)

OneRatioRef proportion of samples for which the genotype score is 0

OneRatioSnp proportion of samples for which the genotype score is 2

FreqHomRef proportion of samples homozygous for the Reference allele

FreqHomSnp proportion of samples homozygous for the Alternate (SNP) allele

FreqHets	proportion of samples which score as heterozygous, that is, scored as 1
PICRef	polymorphism information content (PIC) for the Reference allele
PICSnp	polymorphism information content (PIC) for the SNP
AvgPIC	average of the polymorphism information content (PIC) of the Reference and SNP alleles
AvgCountRef	sum of the tag read counts for all samples, divided by the number of samples with non-zero tag read counts, for the Reference allele row
AvgCountSnp	sum of the tag read counts for all samples, divided by the number of samples with non-zero tag read counts, for the Alternate (SNP) allele row
RepAvg	proportion of technical replicate assay pairs for which the marker score is consistent

In addition, dartR calculates the minor allele frequency and an estimate of read depth, and stores it in the locus metadata.

These metadata variables are held in the genlight object as part of a data.frame called loc.metrics, which can be accessed in the following form:

```
# Make a genlight object to work with
  gl <- testset.gl
# Only entries for the first 10 individuals are shown
  gl@other$loc.metrics$RepAvg[1:10]
```

```
## [1] 1.000000 1.000000 1.000000 1.000000 0.989950 1.000000 0.993274
## [8] 1.000000 1.000000 0.980000
```

You can check the names of all available loc.metrics via:

```
names(gl@other$loc.metrics)
```

```
## [1] "AlleleID" "CloneID" "AlleleSequence" "SNP"
## [5] "SnpPosition" "CallRate" "OneRatioRef" "OneRatioSnp"
## [9] "FreqHomRef" "FreqHomSnp" "FreqHets" "PICRef"
## [13] "PICSnp" "AvgPIC" "AvgCountRef" "AvgCountSnp"
## [17] "RepAvg" "clone" "uid" "rdepth" "maf"
```

Examine the first 10 values of RepAvg, CallRate and some other listed locus metadata in `testset.gl` and your own dataset.

Depending on the report from DarT you may have additional (fewer) `loc.metrics` (e.g. `TrimmedSequence` is available on request).

These metadata are used by the {dartR} package for various purposes, so if any are missing from your dataset, then there will be some analyses that will not be possible. For example, `TrimmedSequence` is used to generate output for subsequent phylogenetic analyses that require estimates of base frequencies and transition and transversion ratios.

`AlleleID` is essential (with its very special format), and dartR scripts for loading your data sets will terminate with an error message if this is not present.

Individual metadata

Individual (=specimen/sample) metadata are user specified, and do not come from DArT. Individual metadata are held in a second dataframe associated with the SNP data in the genlight object. See the figure above.

Two special individual metrics are:

`id`	Unique identifier for the individual or specimen that links back to the physical sample
`pop`	A label for the biological population from which the individual was drawn

Individual metrics are supplied by the user by way of a metafile, provided at the time of inputting the SNP data to the genlight object. A metafile is a comma-delimited file, usually named ind_metrics.csv or similar, that contains labelled columns. The file must have a column headed id, which contains the individual (=specimen or sample labels) and a column headed pop, which contains the populations to which individuals are assigned.

These special metrics can be accessed using:

```
pop(gl)
```

```
popNames(gl)
```

```
indNames(gl)
```

Try these for yourself to see the output they produce.

A number of other user-defined metrics can be included in the metadata file. Examples of user-defined metadata for individuals include:

sex	Sex of the individual (Male, Female)
maturity	Maturity of the individual (Adult, Subadult, juvenile)
lat	Latitude of the location of collection
long	Longitude of the location of collection

These optional data are provided by the user in the same metafile used to assign id labels and assign individuals to populations at the time of reading in the data.

The individual metadata are held in the genlight object as a dataframe named `ind.metrics`. You can check the names of all available `ind.metrics` via:

```
names(gl@other$ind.metrics)
```

```
[1] "id"      "pop"     "lat"     "lon"     "sex"     "maturity"  "collector" "location"  "basin"     "drainage"
```

and can be accessed using the following form:

```
# Only first 10 entries shown
  gl@other$ind.metrics$sex[1:10]
```

```
[1] Male    Male    Male    Male    Unknown Male    Female  Female  Male    Female
Levels: Female Male Unknown
```

Try these for yourself to see the output they produce.

Flags

The genlight object used by dartR has some additional information not normally accessed by the user. If these data are not in the genlight object, various functions may throw an error.

To ensure your genlight object is compliant, use

```
gl <- gl.compliance.check(gl)
```

How dartR stores SilicoDArT data

dartR also stores SilicoDArT presence/absence data in a genlight object, but distinguishes the data from SNP data by setting ploidy=1.

The locus metadata would typically include:

AlleleSequence Sequence of the tag which is present in samples with genotype score "1"

TrimmedSequence Same as the full sequence, but with removed adapters in short marker tags

AvgReadDepth Sum of the tag read counts for all samples, divided by the number of samples with non-zero tag read counts.

StDevReadDepth Standard deviation of the number of tag reads for all samples with non-zero tag read counts

CallRate Proportion of samples for which the genotype call is either "1" or "0", rather than "-"

CloneID Unique identifier of the sequence tag

OneRatio Proportion of samples for which the genotype score is "1"

PIC Polymorphism Information Content

Qpmr Average of the normalized non-zero tag read counts divided by the standard deviation of the normalized non-zero tag read counts (If standard deviation is zero or near zero, the Qpmr value will be 100).

Reproducibility Proportion of technical replicate assay pairs for which the marker score is consistent

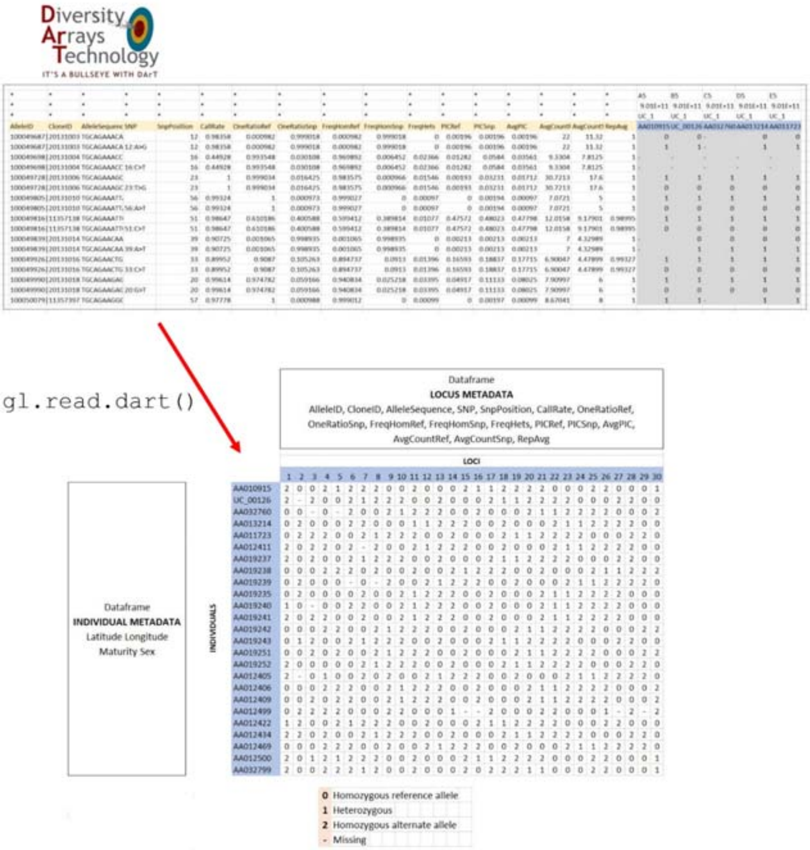
The SilicoDarT data and associated metadata can be accessed in the same way as for SNP data, as described above.

Reading DArT files into a genlight object

SNP genotypes

SNP data can be read into a genlight object using `gl.read.dart()`. This function intelligently interrogates the input csv file to determine

- if the file is a 1-row or 2-row format, as supplied by Diversity Arrays Technology Pty Ltd.
- the number of locus metadata columns to be input (the first typically being `AlleleID` and the last `RepAvg`).
- the number of lines to skip at the top of the csv file before reading the specimen IDs and then the SNP data themselves.
- if there are any errors in the data.


Diversity Arrays Technology
IT'S A BULLSEYE WITH DArT
gl.read.dart()
Dataframe
LOCUS METADATA
AlleleID, CloneID, AlleleSequence, SNP, SnpPosition, CallRate, OneRatioRef, OneRatioSnp, FreqHomRef, FreqHomSnp, FreqHets, PICRef, PICSnp, AvgPIC, AvgCountRef, AvgCountSnp, RepAvg
LOCI
Dataframe
INDIVIDUAL METADATA
Latitude Longitude
Maturity Sex
INDIVIDUALS
0 Homozygous reference allele
1 Heterozygous
2 Homozygous alternate allele
- Missing

An example of the function used to input data is as follows:

```
gl <- gl.read.dart(
  filename="sample_data_2Row.csv",
  ind.metafile="sample_metadata.csv")
```

The filename specifies the csv file provided by Diversity Arrays Technology, and the `ind.metafile` parameter specifies the csv file which contains metrics associated with each individual (id, pop, sex, environmental data, etc).

The resultant genlight object contains the SNP genotypes, the individual metadata and the locus metadata.

If you have saved the sample files provided into your working directory, read `sample_data_2Row.csv` into a genlight object using `DFwt16-sample_metadata.csv` to assign individual metics.

Verify that the genlight object contains the right number of loci, individuals and populations.

SilicoDArT genotypes

SNP data can be read into a genlight object using `gl.read.silicodart()`. This function intelligently interrogates the input csv file to determine

- the number of locus metadata columns to be input (the first typically being `CloneID` and the last `Reproducibility`).
- the number of lines to skip at the top of the csv file before reading the specimen IDs and then the SNP data themselves.
- if there are any errors in the data.

An example of the function used to input data is as follows:

```
gl <- gl.read.silicodart(
  filename="sample_data_silicodart.csv",
  ind.metafile="sample_metadata.csv")
```

The filename specifies the csv file provided by Diversity Arrays Technology, and the `ind.metafile` parameter specifies the csv file which contains metrics associated with each individual (id, pop, sex, environmental data, etc).

The resultant genlight object contains the SilicoDArT presence/absence genotypes, the individual metadata and the locus metadata.

If you have saved the sample files provided into your working directory, read `sample_data_silicodart.csv` into a genlight object using `sample_metadata.csv` to assign individual metics.

Verify that the genlight object contains the right number of loci, individuals and populations.

Reading non-DArT files into a genlight object

If you are working with data that have not been prepared by Diversity Arrays Technology, you can still input the data to dartR provided you can get it into the appropriate format.

The SNP data need to be in one of two forms. SNPs can be coded 0 for homozygous reference, 2 for homozygous alternate, and 1 for heterozygous with NA for missing values; or it can be coded A/A, A/C, C/T, G/A etc, with -/- as missing. Other formats will thow an error.

SilicoDArT data need to be coded as 0 for absent, 1 for present and NA for missing.

The first column of the matrix should be the individual labels, and the first row should be the locus labels.

The individual metadata needs to be in a csv file, with headings, with a mandatory id column corresponding exactly to the identities provided with the SNP data.

The locus metadata needs to be in a csv file with headings. Note that the locus metadata will be complemented by calculable statistics corresponding to those that would be provided by Diversity Arrays Technology (e.g. CallRate).

The script will then calculate the locus metrics that can be calculated, such as Call Rate. Others locus metrics, such as the trimmed sequences need to be provided in the locus metrics file if you wish to use them later.

Once you have the data in this form, you can read it into a genlight object using

```
gl <- gl.read.csv(filename="snp_matrix.csv",
      ind.metafile = "ind_metrics.csv",
      loc.metafile = "loc.metrics.csv")
```

It is prudent to then do a compliance check with

```
gl <- gl.compliance.check(gl)
```

Saving a genlight object

Reading the data in from an Excel spreadsheet and converting to a genlight object takes a lot of computation, and so time. You will also have done some tidying up of the data. It is sensible to save your genlight object in binary form using

```
saveRDS(gl,file="tmp.Rdata")
```

and then read it in again with

```
gl.new <- readRDS("tmp.Rdata")
```

Try saving `gl` or your own genlight object to your workspace, and verify that it has been saved to the appropriate directory. Then restore it to a new genlight object.

Exercises

Exercise 1

2-Row Format

Open the file sample_data_2Row.csv in Excel. This is a set of SNP data for *Emydura*, a freshwater turtle, in 2-row format as would be supplied by DArT Pty Ltd.

Refer to the documentation on the Diversity Arrays Technology web page to understand the scoring of SNPs in the 2-row format.

Also refer to the MetaDataDefinition file provided by DArT Pl as part of their report. In this case, a definition file is provided as sample_metadata.xlsx.

Now examine the individual metadata in the file sample_metadata.csv. Note the two mandatory columns id and pop.

Read the SNP data in to dartR as a genlight object called gl.2row and check the contents.

Exercise 2

1-Row Format

Open the file sample_data_1Row.csv in Excel. This is a set of SNP data provided in 1-row format by DArT PL.

Refer to the documentation on the Diversity Arrays Technology web page to understand the scoring of SNPs in the 1-row format.

Read the SNP data in to dartR as a genlight object called gl.1row and check the contents.

Exercise 3

SilicoDArT

Open the file sample_data_silicodart.csv in Excel. This is a set of marker presence/absence data for Cherax destructor provided in SilicoDArT format by DArT Pty Ltd.

Refer to the documentation on the Diversity Arrays Technology web page to understand the scoring of the data in the SilicoDArT format.

At this stage, dartR does not support analysis of SilicoDArt data.

Tidy up the workspace

We have created files that we will not use again, so they should be removed from the workspace.

```
rm(gl.new, gl.1row, gl.2row)
```

Where have we come?

In this Session, we have covered a range of topics on data entry, the storage of data and some preliminary approaches to examining those data. Having completed the Session, you should understand

- What is a sensible pipeline for preliminary handling of your SNP data.
- How a genlight object is organised in terms of the SNP scores (which are different from the scores used by DArT PL) and how locus and sample metadata are associated with the genotypes.
- The different types of locus metadata generated by DArT PL, and how to look up what each metric means.
- How to read data from DArT Pty Ltd into a genlight object.
- How to interrogate the locus and individual (specimen/sample) metadata.

References

Jombart T. and Caitlin Collins, C. (2015). Analysing genome-wide SNP data using adegenet 2.0.0. http://adegenet.r-forge.r-project.org/files/tutorial-genomics.pdf

Jombart T. and Ahmed, I. (2011). *adegenet 1.3-1*: new tools for the analysis of genome-wide SNP data. *Bioinformatics*, 27: 3070–3071.

Jombart, T., Kamvar, Z.N., Collins, C., Lustrik, R., Beugin, M.P., Knaus, B.J., Solymos, P., Mikryukov, V., Schliep, K., Maié, T., Morkovsky, L., Ahmed, I., Cori, A., Calboli, F. and Ewing, R.J. (2018). Package ‘adegenet’. Version 2.1.1. Exploratory Analysis of Genetic and Genomic Data. https://github.com/thibautjombart/adegenet

Ende