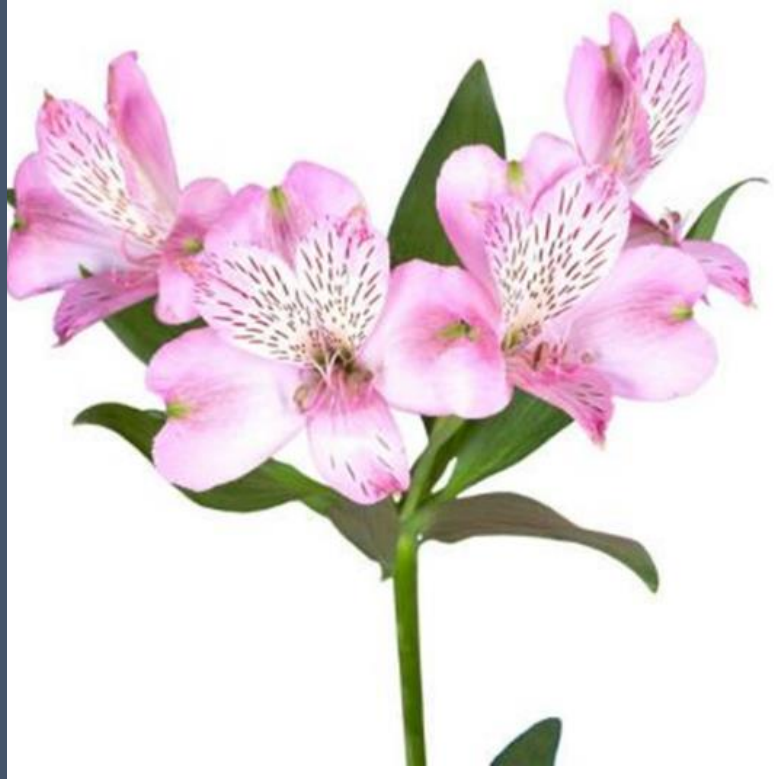
MSc Thesis

SNP markers scoring and linkage maps construction for a triploid Alstroemeria population

Laboratory of Plant breeding, Wageningen University and Research centre, Wageningen the Netherlands



Max van Lieshout

880916-517-130

Student, Plant Sciences, Plant Breeding and Genetic Resources

Wageningen University

May 2016

Supervisors:

Roeland Voorrips

Arwa Shahin

MSc Thesis

Title: SNP markers scoring and linkage maps construction for a triploid Alstroemeria population

Name: Max van Lieshout

Study: MSc Plant Sciences

Specialisation: Plant Breeding and Genetic Resources

Registration number: 880916-517-130

Wageningen University

Laboratory of Plant Breeding

Super visor: Roeland Voorrips & Arwa Shahin

May 16th, 2016

Abstract

The R package fitTetra, and the not yet published fitTri do a good job at genotype calling of tetraploid and triploid samples based on bi-allelic SNP marker assays. However, samples having non-coinciding allele dosages, such as triploid Astroemeria and its diploid and tetraploid parent, cannot be genotyped together, and had to be genotyped separately and combined afterwards. Rejection of a marker in the tetraploid data set, such as having a monomorphic F1 cluster, and genotyping in the triploid F1 data set, results in missing parental genotypes in the combined data set. These missing parental genotypes are a big problem for the construction of polyploid linkage maps, because segregation must be assigned to a parent, and multiple parental genotypes are possible for one segregation pattern. During this thesis we developed methods based on different information sources such as: the information about heterozygosity in the marker name, triploid segregation pattern, arcsine square root of the ratio (ratio between the X and Y signal intensities from marker essay), and XY-signal intensities, to retrieve the missing parental genotypes in the triploid data set due to monomorphic F1 cluster in the tetraploid data set. Out of the 10239 markers with missing parental genotypes in the triploid F1 we were able to retrieve the 3947 parental genotypes. A linkage map was constructed by using heterozygous SNP markers from the diploid parent, containing 988 markers over 11 linkage groups, with a total length of 1015.9cM. Linkage mapping enabled the correction of 372 incorrectly fitted markers having a disomic nulliplex x simplex 1:1 segregation in the triploid F1 from the heterozygous diploid parent.

Keywords: Alstroemeria, fitTetra, fitTri, diploid, triploid, tetraploid, genotyping, allele dosage, linkage mapping.

Table of contents

1: Introduction

1.1: Alstroemeria

Alstroemeria (family *Alstroemeriaceae,* order of *Liliales* and superorder *Liliiflorae*) is an economical important rhizomatous herbaceous monocotyledonous perennial, consisting of approximately 78 species of which most are endemic to Chile (34) and Brazil (44) (Han et al. 1999),(Chacon et al. 2012). Wild Alstroemeria possesses a diploid genome (2n=2x= 16), but new commercial cultivars have originated through interspecific hybridization (Buitendijk et al. 1992), polyploidization and through irradiation treatments (Broertjes and Verboom 1974). Interspecific hybrids, including intra-Chilean species hybrids and hybrids between Chilean and Brazilian species, are often highly sterile but unreduced (2n) gametes occur and formed the basis of the present-day triploid (2n=3x=24) and tetraploid (2n=4x=32) cultivars (Ramanna 1992).

Alstroemeria is mainly cultivated for the production of cut flowers (Buitendijk et al. 1995) but also as bedding or potted plants. Alstroemeria has excellent post-harvest keeping quality, availability of attractive colours and high productivity, which make it a popular flower world-wide. The commercial importance of Alstroemeria has grown rapidly during the last 15 years, and there is still great potential for the future (Lu and Bridgen 1997). Breeding centre are located in the Netherlands but the production area in the Netherlands has decreased over the last few years. Large parts of the production have shifted to growers in South America which have lower production costs and are closer to the important North American market. According to the annual report of Flora Holland, the turnover of Alstroemeria on the Dutch flower auction has reached 29 million euro in 2014, with an increase of 3,7% since 2013. This puts Alstroemeria cut flowers on the 13th position of most sold flowers on the Dutch auction (www.floraholland.nl 2014).

Breeders of Alstroemeria are interested in resistance, novel flower colours, long vase life, yield, big flowers, stem length, and sterility for competitive reasons. Except sterility which is probably achieved by triploidy, many of these traits might have a complex quantitative nature. This together with a long generation cycle of 2 years makes it a challenge to breed for such traits using conventional breeding techniques. The use of molecular assisted breeding (MAB) next to conventional breeding, can shorten and improve the breeding of Alstroemeria. This requires genomic resources such as genomic sequence, molecular markers, and a well-covered linkage map. Despite the fact that a full genomic sequence is not available, there are limited genetic studies performed on Alstroemeria, such as the construction of a linkage map using AFLP markers, study on nuclear DNA content, and chromosome studies (Han et al. 2002; Buitendijk et al. 1997; Hang and Tsuchiya 1988)

Developing genomic resources might be hindered by the fact that Alstroemeria species have very large genomes (1C=26pg, (Kew et al. 2012), and probably both a allo- and autogamous genome structure (Han et al. 2002). For accurate trait mapping and initiating marker assisted breeding (MAB), good marker coverage of linkage maps is required (Shahin et al. 2011; Han et al. 1999). A genetic linkage map using dominant AFLP (Amplified Fragment Length Polymorphism) markers for diploid Alstroemeria has been constructed and several traits expressed in a late stage of plant development such as flower shape, number of flowering stems and vase life have been mapped (Han et al. 2002). However, AFLP markers do not allow the application of this markers in breeding as SNP markers do, and also no linkage is possible between the SNP based map and the AFLP based map. It can be very useful to compare the linear arrangement of markers on this map with the map of the six parental homolog that will be developed in this study and further studies (2 homologs in diploid and 4 homologs in tetraploid).

1.2: Polyploidy in plants

Studies have estimated that approximately 70% of all angiosperms had a polyploid genome at one point in history. The increases in genome size arise predominantly through polyploidy and amplification of non-coding repetitive DNA, especially retrotransposons (Bennetzen et al. 2005). In general, polyploidy leads to increased cell size, flower size, leaf size, and stomatal density, also referred as "Giga" features. These features are of great interest to ornamental and fruit breeders, providing them a platform for commercial exploitation of polyploids. Triploid organisms rarely produce functional fertile gametes, resulting in sterility and absence of seeds, and combine the advantages of polyploidy and hybrid vigour (seedless watermelon, banana, orange)(Dhawan and Lavania 1996; Acquaah 2009).

Many important agronomic and horticultural crops such as potato (*Solanum tuberosum*) (2n=4x=48), bread wheat (*Triticum* spp.) (2n= 6x=42), leek (*Allium porrum*)(2n=4x=32), cotton (*Gossypium hirsutum*), dahlias (*Dahlia variabilisherbaceum*)(2n=4x=52), rose (*Rosa × hybrida*)(2n=3x=21 & 2n=4x=28), petunia (*Petunia hybrida*)(2n=4x=28), and Alstroemeria (2n=3x=24 & 2n=4x=32) possess a polyploid genome. Polyploidy means that organisms possess multiple copies of each chromosome, which are called homologous chromosomes or homologs in their (somatic) cells, instead of the regular two sets of chromosomes. Different numbers of chromosome sets can be present such as: triploid (three sets; 3x), tetraploid (4x), pentaploid (5x), hexaploid (6x), etc. (Fisher 1947), but the focus of this report will be on diploid (2n= 2x = 16), triploid (2n = 3x = 24), and tetraploid (2n = 4x = 32) Alstroemeria.

The origin of duplication of the genome is an important factor to distinguish between polyploids, which can be classified as either autopolyploid or allopolyploid. Generally, Hybridization of different species leads to combination of distinct subgenomes resulting in allopolyploids. An autopolyploid arises from somatic or zygotic chromosome doubling, or by a non-reduction event in the formation of gametes (Gallais 2003; Wendel and Doyle 2005) so that the basic chromosome sets are homologous.

Next to the origin of genome duplication, we can characterize polyploid organisms based on the mode of inheritance: disomic, and polysomic. The exact mode of inheritance affects the segregation of alleles in the offspring and is therefore of great interest, both from an evolutionary perspective as well as for breeding purposes (Stift et al. 2008). In disomic inheritance a chromosome exclusively pairs with one of its homologs which can result in fixed heterozygosity for certain alleles on the homologs.

Polysomic inheritance makes the genetic analyses of polyploids even more complex compared to disomic inheritance, because the homologs pair at random with each other in bivalents or quadrivalents, (Luo et al. 2006). This leads to equal expected frequencies of all the possible allelic combinations excluding the homozygous configurations due to double reduction. During tetravalent homolog pairing a combination of three major events can lead to double reduction: crossing-over between non-sister chromatids, an appropriate pattern of disjunction, and the migration of the sister alleles to the same gamete (Haldane 1930). Offspring of a tetraploid parent having tetrasomic homolog pairing, can have a seemingly impossible allelic combinations appearing inherited from the same parent due to double reduction. So for example, a tetraploid parent with genotype AAAB can produce gametes with genotype BB.

Although most often allotetraploids show disomic inheritance and autotetraploids tetrasomic inheritance they cannot be equated, and Recent studies have shown that there could also be intermediate types of inheritance (segmental allopolyploidy) among different hybrid individuals (Gallais 2003; Stift et al. 2008). For many polyploid crops it is still unknown what the origin of duplication of the genome is and which type(s) of inheritance models they possess. All these complexities make the study of polyploids a great challenge.

In the view of breeding, the mode of inheritance and pairing behaviour is the most important consideration (Doerge and Craig 2000). Despite the fact that most commercial Alstroemeria cultivars have originated through interspecific hybridisation, it is still unknown which types of inheritance models are present. The possible presence of (segmental) polysomic inheritance results in a larger number of possible allelic combinations at a single locus (Meyer et al. 1998). This requires genotyping tools, software, and methodology that enable the genetic analysis of polyploid organisms.

1.3: Genetic analyses of polyploids

Genotyping tools, methodology and software for genetic analysis are well developed in diploid organisms. In polyploids however, genetic studies are not so well represented due to the complex structure of segregation patterns, and the limitation of software to perform linkage mapping and QTL (Quantitative Trait Locus) analysis in polyploids (Bourke et al. 2015). Marker scoring, linkage mapping and QTL analysis are important tools for breeders as a step towards marker assisted breeding(Wu et al. 2004).

This thesis is part of the Alstroemeria project, which is part of the polyploid project, which aims to develop methods for the construction of polyploid linkage maps, necessary for parental haplotype construction and QTL analyses. One of the goals of the Alstroemeria project is to produce a polyploid linkage map for a triploid population, resulted from crossing a tetraploid mother and a diploid father, by creating linkage maps for the diploid and tetraploid parents separately and integrating those maps thereafter (Mann et al. 2011). Triploid Alstroemeria is interesting for breeding companies for competitive reasons.

The availability of the highly abundant co-dominant Single Nucleotide Polymorphisms (SNP) markers enables us to use a new type of information, the allele dosage, which was until recently not used by current methodologies for linkage analysis and QTL mapping (Hackett et al. 2013). For some time genetic analysis in polyploids mostly relied on single-dose (simplex) dominant markers that segregate in a simple 1:1 ratio in mapping populations which require the crossing of a highly heterozygous parent with a parent having a low level of heterozygosity (Wu et al. 1992). Segregation of these single-dose dominant markers in coupling phase is exactly as in a diploid heterozygous × homozygous cross, while also assuming solely random bivalent pairing among homologous chromosomes (Hackett et al. 2001). However, it is desirable to use all the segregating markers in a population since polyploids not only possess nulliplex or simplex allele dosages but also duplex, triplex, quadruplex, etc. allele dosages across the homologous chromosomes, and this may lead to dosage effects and allele interactions not present in diploids (Ripol et al. 1999).

Next generation sequencing technology, high throughput genotyping techniques, and new statistical approaches enable us to generate, genotype, and map huge amounts of molecular markers in a relative short amount of time (Shahin et al. 2012; Blanca et al. 2011). In the Alstroemeria project, bi-allelic SNP markers were designed from RNA-seq sequence assemblies of Alstroemeria parents and ancestors. High throughput genotyping technique Affymetrix Axiom array was used for the scoring of

SNP markers, based on allelic discrimination by direct hybridization of genomic DNA to arrays containing locus- and alleles-specific oligonucleotides. Following a PCR amplification step, the products are end-labelled, hybridized, and stained (Ragoussis 2009), producing two signal intensities for each SNP marker, one for each allele.

Different statistical approaches are available for the estimation of allele dosage using the signal intensities. The correct determination of the allelic configuration in a segregating polyploid population can provide important information about the underlying inheritance pattern and meiosis mechanisms that take place during formation of the progeny(Hackett et al. 2013). All these approaches are developed to find clusters, infer the allele dosage score of each cluster, and assign a genotype dosage to each individual. In the Alstroemeria project the R package fitTetra was used, which uses the ratio of the two signals intensities and fits a mixture model to the distribution of these ratios (Voorrips et al. 2011). The mixture model has five component distributions for the five possible tetraploid genotypes: nulliplex (aaaa), simplex (Aaaa), duplex (AAaa), triplex (AAAa) and quadruplex (AAAA). The three possible allele dosages of diploids (aa, Aa and AA), coincide with the nulliplex, duplex, and quadruplex allele dosages in tetraploids, which enables the simultaneous genotyping of diploid and tetraploid samples. The triploid F1, however, has four different possible dosage classes nulliplex (aaa), simplex (Aaa), duplex (AAa), and triplex (AAA), which do not coincide with the five possible allele dosages for tetraploids, and the 3 possible allele dosages for diploids. This required a new program that can handle the different number of classes, and allele ratios with adaptations specific for triploid samples, called fitTri. The R function “check.F1” evaluated the goodness-of-fit of the assigned F1 genotypes to the parental dosage scores with the assumptions of random or fully preferential pairing, random segregation and no skewed segregation.

Genotyping is followed by the construction of a linkage map by ordering the markers in the (near) optimal position. Several programs have been developed which use different sorts of algorithms to find (near) optimal orders of the markers, while avoiding time consuming calculations. During this thesis JoinMap was used, which uses Maximum Likelihood or Linear Regression algorithm to convert recombination frequency estimates into map distances used for linkage map construction(Van Ooijen 2006). Since joinMap is developed for diploid organisms, it can only distinguish the two phases (two homologs) of a diploid. The mapping of polyploid organisms requires software which can handle the complexities due to multiple possible phase combinations, as estimation of recombination frequencies and LOD scores are mostly not the same in polyploids as in diploids.

The not simultaneous genotyping of the parents and the triploid F1 resulted in many missing parental genotypes for markers where the triploid F1 was genotyped and assigned with an approved segregation. Three possible reasons could lead to missing parental genotypes in the triploid population: SNPs are genotyped by fitTri and fitTetra but parental scores did not explain the segregation type in the triploid population, the total SNP was not genotyped by fitTetra in the tetraploid dataset, or the SNP was genotyped in the tetraploid dataset but parent(s) could not be scored. We focussed on the reason why the total SNP in the tetraploid population was not genotyped. A possible explanation is that fitTetra rejects markers having a monomorphic population pattern because non-segregating markers are not interesting for linkage mapping, and unreliable to genotype. A monomorphic tetraploid population does not mean that the triploid population is also monomorphic, and since the triploid F1 and the parents are genotyped separately this can lead to missing parental genotypes in the triploid population.

1.4: Thesis Goals

In this thesis we are going to study the possibility to genotype the parents of a triploid Alstroemeria population of markers which are rejected by fitTetra because of a monomorphic population pattern, with the use of different sources of information. This is followed by the construction of a diploid linkage map, and the validation of the triploid and parental genotypes of markers segregating 1:1 in the triploid population, using the map construction software joinMap.

2: Material and methods

2.1: Dataset

- Populations

Three different Alstroemeria populations were used in this study with their parents and progenitors. Two tetraploid populations TC and TP (tetra cut and tetra pot), and one triploid population. The triploid population resulted from crossing ALS02 (mother, 2n=4x=32) with ALS01 (father, 2n=2x=16), producing 156 offspring (2n=3x=24). The triploid samples are indicated with the abbreviation Tri- followed by the number of the sample (001 till 156). The TC population resulted from crossing ALS02 (mother, 2n=4x=32) with ALS03 (father, 2n=4x=32), and produced 194 offspring (2n=4x=32). The tetra cut samples are indicated with the abbreviation TC- followed by the number of the sample (001 till 194). The TP population resulted from crossing ALS04 (mother, 2n=4x=32) with ALS05 (father, 2n=4x=32), and resulted in 196 offspring (2n=4x=32). The tetra pot samples are indicated with the abbreviation TP- followed by the number of the sample (001 till 196). The parental samples of ALS1 – ALS5 are replicated and distinguished by a suffixed a, b, or c. The twenty-one progenitors are indicated as ALS06 till ALS27.

- SNP markers

The SNPs were selected out of the RNA-seq (NGS technology RNA sequencing) sequence data of 8 Alstroemeria genotypes (the parents of the three populations: ALS01, ALS02, ALS03, ALS04 and ALS05 and another three progenitors of Alstroemeria: ALS06, ALS07, and ALS08) in order to cover the majority of genetic variation present in Alstroemeria. The origin of the SNP is indicated by the last two numbers in the marker name. As an example: “01” at the end of the marker name “C12345_678_01P” means that the SNP is selected based on the segregation of the SNP in ALS01. In case a SNP is present in more than three of the eight genotypes then “_all” is added to the SNP name “C12345_678_all”. The first numbers in the marker name after the letter “C” and before the first underscore “_” represents the contig number, in which the SNP is located. The numbers between the first and second underscore represents the exact position of the SNP marker in the contig.

SNP markers were pre-selected with quality criteria such as no flanking SNPs, introns, and null alleles at the upstream and downstream side of the SNP. This resulted in a total of 61532 SNP markers, each having two probes, indicated with the letter P or Q at the end of the marker name, which resulted in a total of 123064 markers, developed for the Affymetrix’ Axiom array. Based on fluorescence attachment of the nucleotide incorporated at the SNP locus, the Affymetrix Axiom array produced two signal intensities for each probe, one for each allele.

2.2: R and R packages fitTetra/fitTri and R function check.F1

All of the subsequent simulation work, calculations, data analysis and plotting was performed in R version 3.2.2, and the more user friendly interface RStudio version 0.99.486. SNP markers scoring is conducted by R package fitTetra (Voorrips et al. 2011) with an unpublished extension fitTri used for triploid samples, and is based on the allele signal ratio, which is the fraction of a signal in the total signal. Samples with a small total intensity were not discarded before scoring, as was done in potato, because no clear pattern was found between low signal intensities and the clarity of the pattern of clusters. An arcsine-square root transformation is then used on signal ratios to stabilize the variance, $\gamma = arcsin\sqrt{(y/(x+y)}$ where y and x are two signal intensities and γ is the transformed ratio. Using the expectation maximization (EM) algorithm, a normal mixture model, $(\gamma)=\Sigma\pi i 5 i=1 f i(\gamma)$, is fitted to the transformed allele signal ratios. Five components are expected for tetraploid samples, where triploids are expected to have four components, and diploid samples are expected to have three components, each corresponding to one of the possible genotypic classes. πi (i is 1 to number of

possible genotypic classes) are their mixing proportion and $fi(\gamma)$ are the density functions with different means μi and a common standard deviation σ. To optimize the model fitting, the means μi are constrained as linear or quadratic correlated (Xiao 2015). Using the selected model, the probabilities of belonging to each of the (five for tetraploid and diploid, and four for triploid samples) configurations are calculated for every sample. Only if the probability is above a threshold of 0.95, the corresponding genotype class will be assigned to the sample. Monomorphic markers with a high fraction (above 85%) of samples scored as the same genotype or with a low fraction (below 40%) of scored samples are rejected by fitTetra and fitTri.

The output files ("comb3x.RData" and "comb4x.Rdata") were used as input files for the work in this thesis, and contain the marker name, sample name, X, Y, R, ratio between the XY-intensities $\gamma = \mathrm{y}/(\mathrm{x}+\mathrm{y})$, and the applied genotype.

The R function check.F1 evaluates the goodness-of-fit by Chi-square test of the assigned F1 genotypes to the parental genotypes with the assumptions of random or fully preferential pairing, random segregation and no skewed segregation. However, if no parental genotypes are available, the goodness of fit to the F1 genotypes cannot be performed, and only the validation of skewed segregation was performed, together with the calculation of the fraction of invalid genotypes. Markers where the triploid genotyping passed checkF1 were put in the file "tri_combscores_noRedundant.dat". In this file, a total of 18459 markers have genotyped F1 samples fulfilling the checkF1 requirements. 10239 of these markers miss parental genotypes

The P and Q probes were combined when both probes were genotyped, pass the requirements of checkF1 and were sufficiently similar. The combined P and Q probes were indicated with the letter R at the end of the marker name. Markers where the two segregations coincide without the need of shifting one of the two markers, are indicated with the letters 'nn' at the end of the marker name. Markers where one or both of the probes needed to be shifted to coincide, are indicated with the letters 'ns' (P probe non-shifted, Q probe shifted), 'sn' (P probe shifted, Q probe non-shifted), or 'ss' (both probes shifted) depending which of the two probes required shifting.

2.3: Linkage map construction

JoinMap version 4.1 developed by Kyazma B.V. and Biometris of the Wageningen University (Van Ooijen 2006). During this thesis joinMap was used for mapping and analysing 1350 markers segregating 1:1 in the triploid F1, segregating only from the diploid father (ALS01).

JoinMap allows the user to group markers into linkage groups, and enables the user to quickly inspect the data for every marker and individual for distorted segregation ratios, genotype frequencies, 'double' recombination frequencies, missing values, labelling errors, etc. The parameter Strongest Cross Link (SCL) permits inspection whether the assignment of a marker to a group might be suspicious, and allows the assignment of previously unmapped markers to established linkage groups (Van Ooijen 2006).

Different thresholds are allowed for grouping the SNP markers, and in this way for diploid Alstroemeria, the number of linkage groups should be equal to the number of chromosomes. This means that we expect 8 linkage groups representing the 8 chromosomes. Once the linkage groups are determined using a LOD score between 6 and 8, the linkage map can be calculated for each group. There are now two algorithms to choose from, the original regression mapping algorithm, and the Monte Carlo maximum likelihood (ML) mapping algorithm.

The regression mapping procedure (Stam 1993) is a process of building a map by adding loci one by one, starting from the most informative pair of loci. For each added locus the best position is searched by comparing the goodness-of-fit of the calculated map for each tested position. Both methods should lead to more-or-less the same map orders, however, the ML mapping algorithm allows much faster computation in comparison to the regression mapping algorithm (Van Ooijen 2006). After mapping using ML algorithm we evaluated the maps using several parameters: missing sample values, double recombination estimates, LOD score, and the phase. ML mapping is applied to get a draft version of plausible map positions of the markers, whereas the regression mapping algorithm is used to get a more accurate estimate of the markers on the map, once the ML map is corrected and approved.

2.4: Mapchart

MapChart is a computer programme that displays linkage maps (Voorrips 2002). The linkage maps are projected as vertical bars. Mapchart is incorporated in JoinMap. In this thesis MapChart version 2.2 is used.

3: Results

3.1: Identification of markers that are genotyped in the triploid F1, and not in TC and TP F1

The output files ("comb3x.RData" and "comb4x.Rdata") were used as input files for the work in this thesis, and contain the marker name, sample name, X, Y, R, ratio between the XY-intensities $\gamma = y/(x+y)$, and the applied genotype.

The tetraploid populations TC and TP, combined with the diploid and tetraploid parents and ancestors were genotyped in fitTetra. Out of the 123064 markers, 40645 markers could be genotyped by fitTetra, and 33483 markers out of the 123064 markers in the triploid population could be genotyped by fitTri. However, 15999 genotyped SNP markers in triploid population miss parental genotypes in the output files from fitTri (comb3x) and were not genotyped by fitTetra (comb4x) (the 15999 markers were stored in the file: "rejectedmarkerset.RData") (The script for marker identification, fulfilling the requirements of having genotyped triploid F1 with non-genotyped ALS01, ALS02, TC F1, and TP F1 can be found in appendix scripts 1.2).

3.2: Defining monomorphic tetraploid clusters (XY & ratio)

Presence of a monomorphic tetraploid pattern (consists of a single compact well defined cluster of signal intensities in a tetraploid XY-scatterplot with no segregation into different allele dosages) could be one reason for missing parental genotypes. SNPs were rejected by fitTetra and/or fitTri when the maximum allowed fraction of the scored samples that are in one peak was exceeded (threshold = 0.85) of F1 samples having the same genotype, (even when the parents have different genotypes than the tetraploid F1), because the remaining samples offers too little information for reliable model fitting. The diploid and tetraploid parents of the triploid F1 (ALS1 and ALS2) were genotyped together with the tetraploid F1 populations, and by rejecting markers in fitTetra having a monomorphic pattern in the tetraploid population will lead to missing parental genotypes, even when the triploid F1 could be genotyped by fitTri. To recover the genotypes of ALS1 and ALS2 in these set of markers we need to find method to identify the monomorphic cluster in tetraploid populations, and by comparing the signal intensities of ALS01 and ALS02 to the monomorphic cluster the parental genotypes can then possibly be retrieved (possible parental genotypes indicated in table 3.1).

If the tetraploid TC F1 population is monomorphic, this means that we expect ALS2 (one of the parents of the TC population as well as of the triploid population) to be either homozygous (nulliplex, quadruplex), or non-segregating disomic duplex (AA|aa). If ALS2 has a simplex, duplex (disomic Aa|Aa, or tetrasomic) or triplex genotype the TC population will segregate, and cannot be monomorphic. This means that the triploid F1 population is expected to have limited possible segregation patterns (monomorphic, and 1:1, Table 3.1). Some triploid segregation patterns have two or three possible parental genotypes. Some of those genotypes of ALS02 will lead to segregation in the TC population (ALS02 indicated with an asterisk in Table 3.1). To be able to assign parental genotypes to markers rejected because of a monomorphic tetraploid pattern, we need to find a way to distinguish between segregating and non-segregating tetraploid populations.

Plotting of markers rejected by fitTetra revealed two possible methods to identify monomorphic markers in TC and TP populations. The first way is to use of the standard deviation (SD) of the signal ratio $\gamma = y/(x+y)$, and the second way is to use of the XY-intensities obtained from Affymetrix axiom array.

Table 3.1: The expected segregation patterns in the triploid population for the limited possible allele dosages of ALS02 when we assume the tetraploid population is a monomorphic cluster. To fulfil the monomorphic requirement, ALS02 can have nulliplex, quadruplex, or a disomic duplex (AA|aa) genotype, but the simplex and triplex genotypes (marked with an asterisk) are not possible. The segregation pattern (indicated with an 'S') explain the segregation of a marker into the 4 possible genotypes for the triploid population.

ALS01 (diploid)	**ALS02 (tetraploid)**	**triploid Segregation**
duplex	quadruplex	S0001
nulliplex	quadruplex	S0010
duplex	duplex (disomic AA|aa)	S0010
duplex	nulliplex	S0100
nulliplex	duplex (disomic AA|aa)	S0100
nulliplex	nulliplex	S1000
simplex	quadruplex	S0011
duplex	triplex*	S0011
duplex	simplex*	S0110
simplex	duplex (disomic AA|aa)	S0110
nulliplex	triplex*	S0110
simplex	nulliplex	S1100
nulliplex	simplex*	S1100

3.2.1: SD of the ratio

The first way to identify monomorphic markers in the tetraploid populations is with the use of the SD of the arcsine square root of the ratio $\gamma = y/(x+y)$. First the arcsine square root of the ratio of each tetraploid F1 individual was calculated. Second we used this ratio of each sample to calculate the mean and standard deviation for each of the selected 15999 markers, as shown in table 3.2 (both TC and TP population combined) (the calculation of the 'mr' and 'sr' can be found in appendix scripts: 1.3). We expect monomorphic clusters (as opposed to unclear mixtures of genotypes) to have a low SD. To define the difference between monomorphic and mixed clusters requires a certain threshold. This was done by grouping the markers by increasing SD of the arcsine square root of the ratio, followed by visual inspection of marker XY-plots in different SD clusters.

Table 3.2: The mean and standard deviation of the X and Y signal intensities, and of the arcsine square root of the ratio, for each marker. 'mx' represent the mean and 'sx' the SD of the X-signal intensity, 'my' the mean and 'sy' the SD of the Y-signal intensity, and 'mr' represents the mean and 'sr' the SD of the arcsine square root of the ratio of the tetraploid TC and TP F1 population for each marker. The complete dataframe is found in the file: "gemiddeldemarkerscomb4x.RData".

	MarkerName	mx	sx	my	sy	mr	sr
1	C10002_1330_04P	1010.5998	516.54711	1121.8057	512.40748	0.8216094	0.19279098
2	C10002_1330_04Q	557.8677	163.79225	1161.1496	218.96522	0.9672996	0.09555097
3	C10004_100_allQ	1318.4890	211.22835	1765.8518	296.77153	0.8572183	0.06539906
4	C10006_229_04P	1274.5501	318.77261	1048.0693	274.16981	0.7367345	0.10369293
5	C10006_229_04Q	804.5518	334.56292	654.2957	200.18188	0.7446935	0.13294889

The range of the SD of the arcsine square root of the ratio is between 0.018 and 0.283. We made 28 clusters increasing with 0.01 SD, starting at a SD of 0.01 and ending at a SD of 0.28. We made 20 XY-scatter plots of the combined TC and TP F1 together with the ALS01 - ALS05 parents of each cluster (if enough markers are present in a cluster). By visual inspection of the XY-scatter plots of each clusters, we came up with a threshold excluding markers that do not fulfil the monomorphic ratio pattern. This threshold is set at SD < 0.06 (The grouping of the markers based on the arcsine square root of the ratio of tetraploid F1 is represented in appendix 1.4**).** A total of 4459 markers out of the 15999 markers have an SD <0.06, and were identified as being monomorphic in the tetraploid F1 populations.

3.2.2: XY signal intensities

We assumed fitTetra rejected a lot of markers because of a monomorphic pattern in the tetraploid F1 populations. We expected that these markers have a small SD. During the process of obtaining monomorphic markers with the use of the SD of the arcsine square root of the ratio, we found that a lot of markers have large SD (SD >0.06). Visual inspection of many of the XY-scatter plots however, revealed a monomorphic pattern (nulliplex and quadruplex) in the tetraploid populations. Selection of markers based on a small SD of the arcsine square root of the ratio alone does not identify all monomorphic markers, so we need an additional method to identify monomorphic markers with SD > 0.06.

A lot of homozygous markers share the characteristics of having a large range in the Y direction and background noise in the X direction, or v.v. (background noise is the background signal intensity not explained by the staining of hybridised genomic DNA to arrays containing locus- and alleles-specific oligonucleotides). fitTetra used the ratio of the X and Y signal intensities (ratio = Y/(X+Y)), and perhaps could not filter out the background noise in combination with large ranges of the ratio sufficiently, although many of these markers were presumably rejected by fitTetra because they are monomorphic. However, it is possible to identify these monomorphic markers by comparing the two ranges of the X and Y signal intensities. Figure 3.1 shows the nicely placed nulliplex and quadruplex clusters on the X and Y-axis, showing the difference in range between the X and Y signal intensities.

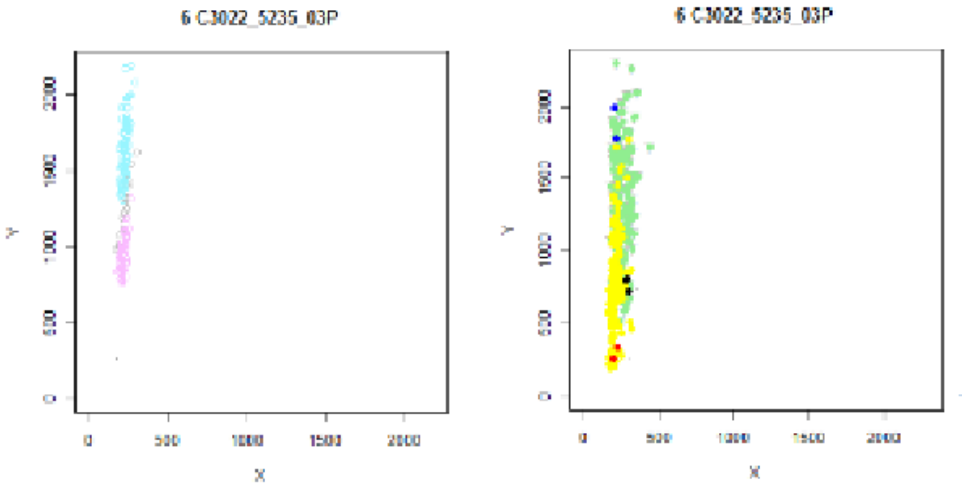


Figure 3.1. Left: monomorphic triploid cluster with a triplex allele dosage, Right: for the same SNP marker, the monomorphic tetraploid cluster with a quadruplex allele dosage and an SD > 0.06. ALS01 (indicated in red) is the diploid parent of the triploid population. ALS02 (indicated in blue) is the tetraploid parent of both the triploid and tetraploid TC population (green). ALS03 indicated in black, is the other parent of the TC population (green).The other tetraploid population (TP) is shown in green.

We calculated for each marker the mean and standard deviation of the X and Y signal intensities, of the combined tetraploid TC and TP F1 populations. We calculated the length of the 95% confidence interval (C.I.) of the X and Y signal intensities to exclude extreme individual outliers (Table 3.3) (The R-script of the calculation of the 95% C.I. of the signal intensity length can be found in Appendix scripts 1.5).

Table 3.3: The mean (mx, my) and SD (sx, sy) were used to calculate the length of the 95% C.I. of the of the XY-signal intensities (Xlength, Ylength) of the tetraploid TC and TP F1. The X and Y length of the 95% C.I. were used to define homozygous monomorphic clusters, which could not be defined with the SD<0.06 of the arcsine square root of the ratio. The file can be found under the name: "msXYmarkers.RData".

	MarkerName	mx	sx	my	sy	Xlength	Ylength
1	C10002_1330_04P	1010.5998	516.54711	1121.8057	512.40748	2024.8275	2008.6004
2	C10002_1330_04Q	557.8677	163.79225	1161.1496	218.96522	642.0538	858.3279
3	C10004_100_allQ	1318.4890	211.22835	1765.8518	296.77153	827.9999	1163.3230
4	C10006_229_04P	1274.5501	318.77261	1048.0693	274.16981	1249.5657	1074.7259
5	C10006_229_04Q	804.5518	334.56292	654.2957	200.18188	1311.4626	784.6986

As with the use of the SD of the arcsine square root of the ratio, a threshold was necessary to distinguish markers with one homozygous cluster from other situations (segregating markers, markers with too much noise). We used the length of the 95% C.I. of the X and Y signal intensities to set the threshold. By trial and error and visual inspection of XY-scatter plots we came with a threshold of a 3x larger length of the X-signal intensities compared to the Y-signal intensities for nulliplex genotypes, and a 3x larger length of the Y-signal intensities compared to the X-signal intensities for quadruplex genotypes.

Out of the 15999 markers, a total of 1141 markers show a nulliplex allele dosage for the TC and TP populations, while 1419 markers show a quadruplex allele dosage in the tetraploid populations. Of these 2560 monomorphic homozygous markers, 1334 markers overlap with the monomorphic markers based on SD of the ratio. This means we were able to select 1226 extra markers which otherwise would not be identified as being monomorphic.

3.3: genotyping ALS01 and ALS02

Once markers were obtained having a monomorphic tetraploid F1 we tried to genotype ALS01 and ALS02. The markers identified with the use of the XY signal intensities were homozygous in the tetraploid F1. This means that the parents of these tetraploid F1 samples (ALS02, ALS03, and ALS04) also must be homozygous for these markers, as shown in Table 3.4 and 3.5 for segregation s10000 and s00001. Genotyping of ALS01 is not so straightforward because it is a parent of the triploid population, and can have a different relative allele dosage than the tetraploid parents. It was possible however, to genotype ALS01 based on the comparison of the parental XY-signal intensities with the 95% C.I. of the tetraploid F1, or with the use of the triploid segregation pattern in combination with the retrieved ALS02 genotype.

The monomorphic markers identified with the arcsine square root of the ratio alone do not provide sufficient information to genotype the parents. It is very hard to define the correct genotype of a monomorphic tetraploid F1, as in principle multiple genotypes are possible for the monomorphic F1 cluster because of the possibility of having both disomic and tetrasomic inheritance in Alstroemeria (Tables 3.4 and 3.5). This required additional sources of information which possibly could help genotyping ALS01 and ALS02 in cases where the tetraploid F1's are monomorphic.

Table 3.4 Expected segregation patterns with tetraploid parents having full tetrasomic inheritance, where the "s" is short for segregation and five numbers behind "s" are the expected ratio of five genotypes. The "I" in "s18I81" stands for 18 so "s18I81" means that there were five expected genotypes in ratio of 1:8:18:8:1.

autotetraploid

	aaaa	aaaA	aaAA	aAAA	AAAA
aaaa	s10000				
aaaA	s11000	s12100			
aaAA	s14100	s15510	s18I81		
aAAA	s01100	s01210	s01551	s00121	
AAAA	s00100	s00110	s00141	s00011	s0001

Table 3.5 Expected segregation pattern with tetraploid parents having full disomic inheritance. In the duplex allele dosage, the homologs show two different cases of preferential pairing: "aa|AA" results in non-segregating gametes (aA), and with preferential pairing "Aa|Aa" the gametes segregate 1:2:1.

allotetraploid

	aaaa	aaaA	aa|AA	aA|aA	aAAA	AAAA
aaaa	s10000					
aaaA	s11000	s12100				
aa|AA	s01000	s01100	s00100			
aA|aA	s12100	s13310	s01210	s14641		
aAAA	s01100	s01210	s00110	s01331	s00121	
AAAA	s00100	s00110	s00010	s00121	s00011	s0001

Two additional sources of information were used in combination with the arcsine sqrt ratio: the triploid segregation pattern, and the information about heterozygosity of the parents in the last part of the marker name. Together with the XY-signal intensities the following three methods were used to genotype ALS01 and ALS02:

- Parental XY intensities compared with 95% C.I of XY intensities of monomorphic tetraploid cluster.
- Information about heterozygosity of the parents provided in the last part of the marker name, in combination with the parental arcsine sqrt of the ratio compared with 95% C.I. of the arcsine sqrt of the ratio of the monomorphic tetraploid cluster.
- Triploid segregation pattern in combination with the parental arcsine sqrt of the ratio compared with 95% C.I. of the arcsine sqrt of the ratio of the monomorphic tetraploid cluster.

These three methods are presented in section 3.3.1 to 3.3.3. To be able to use the assigned triploid segregation type, and triploid F1 genotypes together with the marker names used by fitTetra and fitTri we had to convert the combined markers to from the file "tri_combscores_noRedundant.dat" back in a P and a Q probes, and remove shifting/non-shifting information (the script used for separation of the combined two probes are stored in appendix scripts: 1.7).

This was needed because the R function "check.F1" evaluated the goodness-of-fit by Chi-square test of the assigned F1 genotypes to the parental genotypes with the assumptions of random or fully preferential pairing, random segregation and no skewed segregation. A total of 18459 markers have

genotyped triploid F1 samples fulfilling the checkF1 requirements, from which 10239 markers miss parental genotypes (55.5%). The triploid F1 segregation pattern of the P and Q probes that fulfil the "checkF1" requirement are compared with the function 'compare_probe', and combined as one probe in the file "tri_combscores_noRedundant.dat" if the segregation type was the same. Markers where one or both probes were shifted to let the segregation type coincide were indicated with a combination of the letters "n" (non shifted) and "s" (shifted), were the first letter relates to the P probe, and the second letter to the Q probe (a total of 3249 markers were combined, which are indicated with the letter R at the end of the marker name).

After converting the combined markers into P and Q probes, 18459 + 3249 = 21708 probes have assigned segregation types in the triploid population, from which 10239 + 721 = 10960 probes lack parental genotypes (721 separated probes). We stated that in the R files (comb3x & comb4x before checkF1, and combining the markers), 15999 genotyped markers in the triploid population have no parental genotype. Comparison of the 15999 markers with the 21708 genotyped markers leads to 9936 probes with missing parental genotypes having assigned triploid segregation types by checkF1.

3.3.1: Parental XY intensities compared with 95% C.I of XY intensities of monomorphic tetraploid cluster.

A total of 2560 markers of the 15999 markers show either a nulliplex or quadruplex allele dosage for the TC and TP populations. For these homozygous tetraploid clusters we tested whether the parental X or Y signal intensities significantly differed from the 95% C.I. of the X or Y signal intensities of the tetraploid F1. This resulted in a P-value for each parental signal intensity (X and Y) as shown in table 3.6 (The R-script for calculating the P-values can be found in appendix scripts: 1.5).

Table 3.6: The P-values for the ALS01 and ALS02 XY signal intensities are indicated in the columns: 'pALS01aX', 'pALS01bX' 'pALS01aY', pALS01bY', 'pALS02a', etc. The file is stored under the name: "pvaluesXYparents

MarkerName	pALS01aX	pALS01bX	pALS01aY	pALS01bY	pALS02aX	pALS02bX	pALS02aY	pALS02bY
C10002_1330_04P	0.099961401723297	0.088890263138371	0.5642922582	0.43700175737	0.560436278	0.54537898	0.21274389	0.34024319
C10002_1330_04Q	0.316135937917874	0.099524539232184	0.8770375311	0.93985002420	0.515182095	0.61193436	0.37435950	0.22119406
C10004_100_allQ	0.710451129429501	0.750471637386218	0.0706337510	0.14348367945	0.111814770	0.18849199	0.81762716	0.74079617
C10006_229_04P	0.999619248652872	0.999631848177490	0.1957121641	0.24646679044	0.136598785	0.16740912	0.46478544	0.70584010
C10006_229_04Q	0.965964607157291	0.996120820763696	0.2452211100	0.17109503192	0.181470334	0.36463421	0.50777051	0.79816344

Figure 3.2 shows the markers with a homozygous genotype in the tetraploid F1 (TC and TP) that are identified using the XY-signal intensities. Homozygous markers have a one sided C.I. which means that parental samples can only be significantly larger than the 95% C.I. For markers having a monomorphic cluster with a quadruplex genotype (Figure 3.2 top left), we identified parents as significantly different from the tetraploid cluster when P-values of the X signal intensity of both the samples (a and b) were > 0.95. For markers having a tetraploid cluster with a nulliplex genotype, we used the criteria of a P-value > 0.95 of the Y signal intensity to identify the parents as significantly different (Figure 3.2, top right: ALS01). When the p-value of the signal intensity of one or both of the samples was equal to or lower than 0.95, the parent was identified it as not significantly different from the tetraploid cluster, and we were able to genotype it with the same homozygous genotype as the tetraploid cluster. This means for tetraploid cluster identified as having a quadruplex genotype, ALS01 was genotyped as duplex and ALS02 as quadruplex (Figure 3.2, top left), and for nulliplex tetraploid clusters both ALS01 and ALS01 were genotyped as nulliplex (Figure 3.2, bottom left and

right) (The filtering of markers fulfilling the requirements of being a homozygous cluster, and having P-values>0.95 can be found in appendix scripts: 1.8).

The genotypes of ALS01 and ALS02 were stored in the file: “parentscoring.RData”, under the columns ‘genoALS01XY’ (ALS01 genotypes using XY signal intensities), and ‘genoALS02XY’ (ALS02 genotypes using XY signal intensities) (The R-script of genotyping ALS01 and ALS02 can be found in appendix scripts 1.9.1).

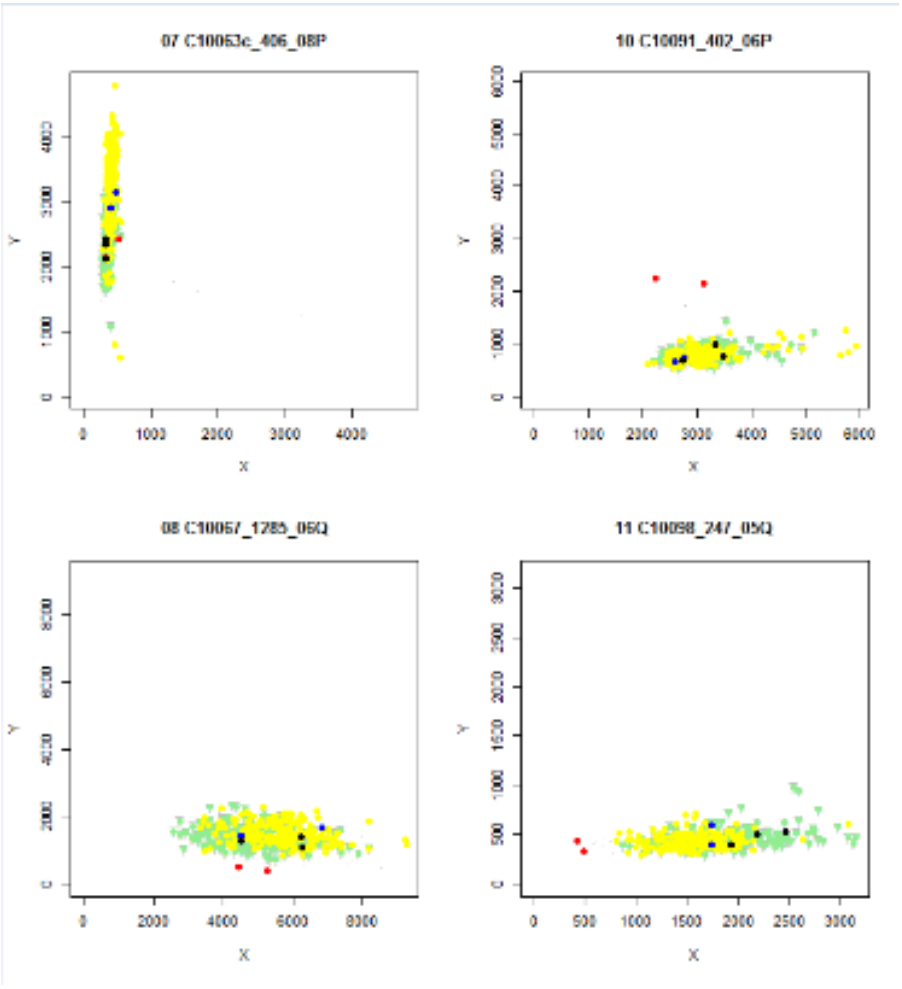


Figure 3.2: Monomorphic markers with a homozygous genotype in the tetraploid F1 that are identified with the XY-signal intensities. TC population (in green), TP (in yellow), ALS01 (in red), ALS02 (in blue), ALS03 (in black). To partially confirm the homozygosity of the tetraploid F1, we plotted ALS03 which is expected to have the same genotype as ALS02, since both are parent of the TC population. Top left: ALS01, ALS02, and ALS03 not significantly differing from the cluster having a homozygous quadruplex allele dosage. Bottom left: ALS01, ALS02, and ALS03 not significantly differing from the cluster having a homozygous nulliplex allele dosage. ALS01 would sign differ if the 95% C.I. would be two sided but this is not expected, since the tetraploid cluster is homozygous. Top right: significantly different ALS01, where ALS02, and ALS03 are not significantly differing from the cluster having a homozygous nulliplex allele dosage. Bottom right: ALS01, ALS02, and ALS03 not significantly differing from the cluster having a homozygous nulliplex genotype. ALS01 and ALS02 were given the same genotype as the tetraploid F1 when both samples of each parent did not significantly differ from the tetraploid cluster.

Table 3.7 shows the amount of parental samples which could be genotyped using the comparison of the XY-signal intensities with the length of the 95% C.I. of the XY signal intensities. The last column shows the total amount of markers having a tetraploid homozygous cluster. For a total of 2477 markers, at least one of the parental samples could be genotyped with the use of this method. As expected, more ALS02 parents were found to be homozygous in comparison to ALS01, partially confirming the homozygosity of the parents of the TC F1. By using the assigned triploid segregation

pattern, in combination with one corresponding homozygous parental genotype, we were able to genotype the parent which was significantly different from the homozygous tetraploid F1.

Table 3.7: Number of markers where ALS01 (No. genotyped ALS01), ALS02 (No. genotyped ALS02), or at least one of could be genotyped of the markers having a homozygous tetraploid cluster pattern.

Tetraploid cluster pattern	No. ALS01 not sign dif from homozygous tetraploid F1	No. ALS02 not sign dif from homozygous tetraploid F1	Total unique monomorphic markers
nulliplex	619	1030	1096
quadruplex	610	1330	1381
total	1229	2360	2477

3.3.2: Triploid segregation pattern in combination with the parental arcsine sqrt of the ratio compared with 95% C.I. of the arcsine sqrt of the ratio of the monomorphic tetraploid cluster.

The second method to genotype ALS01 and ALS02 is based on the assigned segregation type in the triploid F1 population (one of the 20 tetrasomic and/or disomic patterns). As shown in Table 3.8, multiple parental genotypes are possible for one assigned triploid segregation pattern, and the segregation could be explained by the heterozygosity of ALS01 and/or ALS02. This means we had to use the triploid segregation pattern in combination with the relationship between the parental ratio and the monomorphic tetraploid cluster (SD > 0.06) to genotype ALS01 and ALS02.

Table 3.8: The expected segregation pattern in a triploid population with ALS01 (diploid), and ALS02 (tetraploid) having full disomic inheritance, or tetrasomic inheritance.

Triploid F1 segregation

		ALS01		
		aa	aA	AA
ALS02	aaaa	s1000	s1100	S0100
	aaaA	s1100	s1210	S0110
	aaAA	S1410	S1551	S0141
	aa|AA	s0100	s0110	s0010
	aA|aA	s1210	s1331	s0121
	aAAA	s0110	s0121	s0011
	AAAA	s0010	s0011	s0001

One way to identify the correct parental genotypes is to study the relationship between the arcsine square root of the ratio of the parents and the tetraploid F1 cluster. Due to the low number of parental replicates (only 2 individuals, a and b), we could not test whether the parents significantly differed from each other. However, it was possible to use the ratio information to test whether the parents were significantly different from the tetraploid cluster.

This was done by comparing the arcsine square root of the ratio of ALS01 (a and b) and ALS02 (a and b) with the 95% C.I. of the arcsine sqrt of the tetraploid F1 (shown in Table 3.9), with the assumption that arcsine sqrt or the ratio of the tetraploid F1 (TC and TP F1 combined) were normally distributed. This resulted in a probability value (P-value) for each parental sample (shown in Table 3.9), which indicates whether the parental sample were inside or outside the 95% C.I. of the tetraploid cluster.

The P-value of the parents was calculated with the use of the arsine square root of the ratio of the parents, and the mean and standard deviation of the arcsine square root of the ratio of the tetraploid F1. The P-value of each parent is shown in table 3.9 in the columns “PALS01ar”, “PALS01br”, “PALS02ar”, and “PALS02br” (the used script for calculation of the probability values for the parental arcsine sqrt of the ratio compared to the 95% C.I. of the tetraploid F1 are shown in appendix 1.6).

The 95% C.I. was chosen by visual inspection of the XY-scatter plots, where individuals falling outside this 95% C.I. were in general clearly outside the cluster. A higher than 95% C.I gave the false negative result of parents falling clearly outside the cluster but by visual inspection of XY-scatter plots were not significantly different from the tetraploid F1 cluster.

We assume that parents were not significantly different from the tetraploid F1 cluster when the P-values were >0.025 and <0.975. We assumed that parental samples with P-values <0.025 were significantly smaller than the tetraploid cluster, and samples with P-values >0.975 were significantly larger than the tetraploid F1 cluster. For the parent to be significantly different from the tetraploid F1, both the samples (a and b) had to be significantly different from the cluster.

Table 3.9: The mean (mr) and standard deviation (sr)of the arcsine(sqrt(ratio)) of each marker. The arcsine(sqrt(ratio)) of each parent (ALS01a, ALS01b, ALS02a, and ALS02b) is shown in columns (rALS01a … rALS02b), and the corresponding P-value of the parental arcsine(sqrt(ratio)) in relation with the tetraploid F1 cluster arcsine(sqrt(ratio)) is shown by (PALS01ar…PALS02br). The Table is stored under the name: “pvaluesRatioparents.RData”.

	MarkerName	mr	sr	rALS01a	rALS01b	rALS02a	rALS02b	PALS01ar	PALS01br	PALS02ar	PALS02br
1	C10002_1330_04P	0.8216094	0.19279098	1.0773335	1.0681384	0.6804169	0.7452884	9.076516e-01	8.995052e-01	0.231974157	0.3460987694
2	C10002_1330_04Q	0.9672996	0.09555097	1.0436627	1.1223689	0.9473929	0.9082356	7.879090e-01	9.476941e-01	0.417483171	0.2682411463
3	C10004_100_allQ	0.8572183	0.06539906	0.7661555	0.7834360	0.9452981	0.9206662	8.189811e-02	1.296210e-01	0.910978675	0.8340182309
4	C10006_229_04P	0.7367345	0.10369293	0.5319263	0.5440096	0.8106963	0.8385031	2.412604e-02	3.154111e-02	0.762162966	0.8368125673
5	C10006_229_04Q	0.7446935	0.13294889	0.5433743	0.4820732	0.8538197	0.8293900	6.497995e-02	2.411434e-02	0.794123748	0.7379573122

To define the correct parental genotype for an assigned triploid F1 segregation, we used the relationship between the arcsine sqrt of the ratio of the parental samples and the tetraploid F1 cluster. As explained earlier, multiple parental genotypes are possible for one assigned triploid F1 segregation (Table 3.8). This means that the assigned triploid F1 segregation could be explained by segregation from ALS01, and/or ALS02. We expected to find differences between the relative allele dosages of ALS01 and ALS02 for a given triploid segregation which also means that the arcsine sqrt of the ratio between these parents is expected to be different. The relation of the parental arcsine sqrt of the ratio in comparison to the tetraploid F1 is expressed in the previously calculated P-value. These P-values indicate whether one parent has a smaller or larger “relative” allele dosage than the other parent, which enabled us to define the correct genotype for monomorphic and 1:1 triploid segregation patterns (the assignment of parental genotypes using the triploid segregation and arcsine sqrt of the ratio can be found in appendix 1.9.3).

As an example the duplex: triplex segregation (s0011) in the triploid F1 has two possibilities of parental genotypes: ALS01 is duplex and ALS02 is triplex, or ALS01 as simplex and ALS02 as quadruplex. Based on the position of the tetraploid F1 we expect one parent to be significantly smaller or larger than the tetraploid F1, and the other parent to be not significantly different from the tetraploid F1, as seen in Figure 3.3.

This method was also used for the other 1:1 segregations. The S0110 segregation however, has extra third possibility where both parents have the same “relative” genotype (ALS01 is simplex, and ALS02 is duplex having disomic inheritance). In this case we expect both parents to be not significantly differ from the tetraploid F1 cluster, and the parents fulfilling this criteria could be genotyped as simplex (ALS01), and duplex (ALS02).

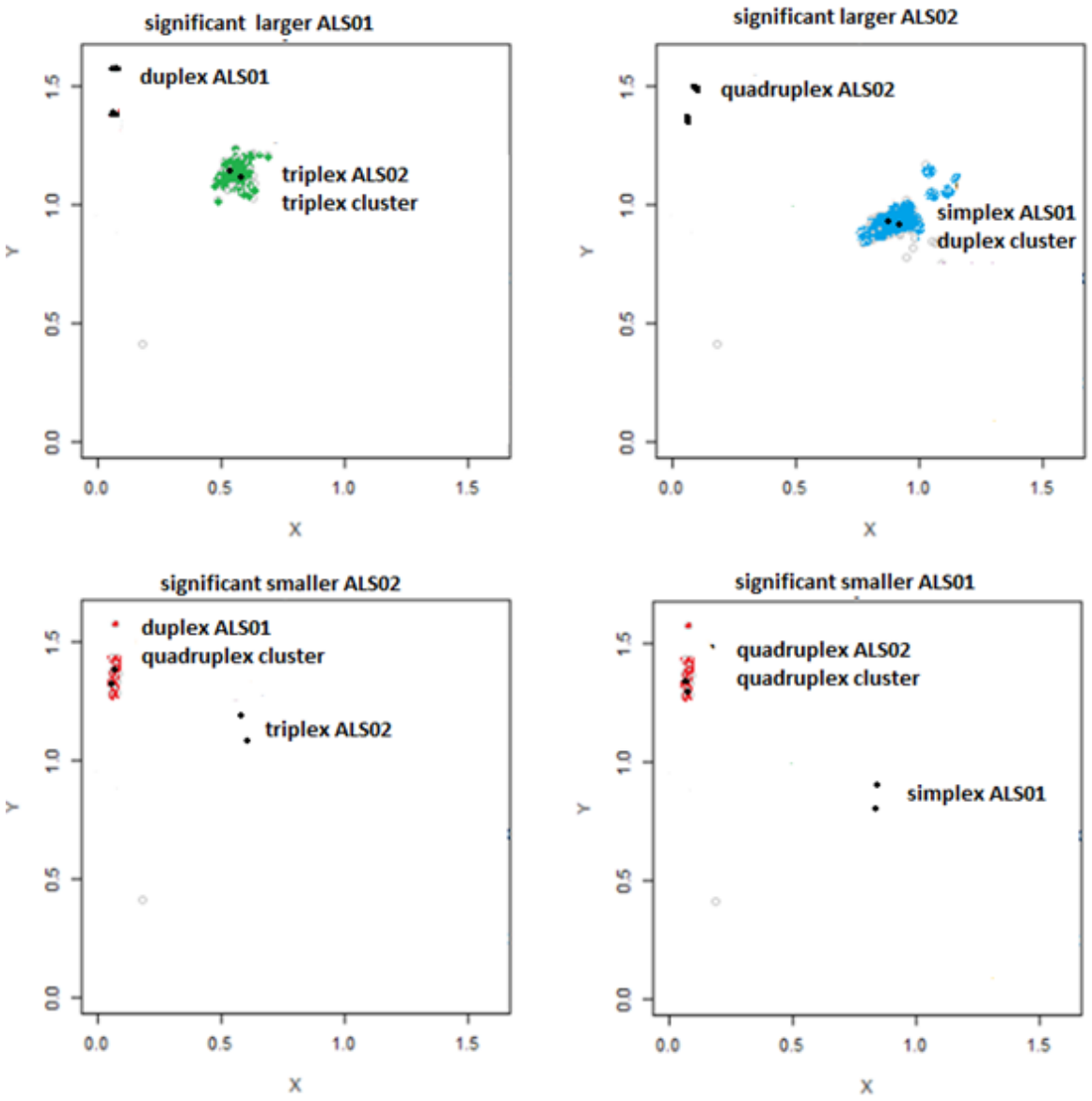


Figure 3.3: Genotyping of ALS01 and ALS02 based on the parental arcsine square root of the ratio in relation to the tetraploid F1 cluster, when the triplex F1 has segregation s0011 (duplex : triplex). Top left: ALS02 is not significantly different from the tetraploid F1, where ALS01 is significantly larger. In this case ALS01 is genotyped as duplex, and ALS02 is genotyped as triplex. Bottom left: ALS01 is not significantly different from the tetraploid F1, where ALS02 is significantly smaller. In this case ALS01 is genotyped as duplex, and ALS02 is genotyped as triplex. Top right: ALS01 is not significantly different from the tetraploid F1, where ALS02 is significantly larger. In this case ALS01 is genotyped as simplex, and ALS02 is genotyped as quadruplex. Bottom right: ALS02 is not significantly different from the tetraploid F1, where ALS01 is significantly smaller. In this case ALS01 is genotyped as simplex, and ALS02 is genotyped as quadruplex.

For the triploid 1:3:3:1 and 1:5:5:1 segregation patterns (resulting from segregating ALS02) only one genotype is possible for the parents, so in theory we do not have to test the parental relationship of the ratio between the two (when we assume the triploid segregation pattern is correct). Also the markers having these segregating patterns are expected to have SD > 0.06 of the arcsine square root of the ratio. For this reason we can ignore the SD > 0.06, and use the methods based on the information about heterozygosity of the parents in the marker name and the arcsine square root of the ratio to confirm a correct triploid segregation or to warn for a possible scoring error (such as a shift).

In total 1216 markers could be genotyped using the triploid segregation in combination with the arcsine sqrt of the ratio of the parents and tetraploid F1. Table 3.10 shows the amount of markers for each possible segregation type that is not in conflict with the monomorphic requirement (SD < 0.06 in the tetraploid F1), which could be genotyped using this method. The genotypes of ALS01 and ALS02 were stored in the file: "parentscoring.RData", under the columns 'genoALS01seg' (genotypes ALS01 using triploid F1 segregation pattern), and 'genoALS02seg' (genotypes ALS02 using triploid segregation pattern).

Table 3.10: amount of genotyped markers in the triploid F1, using the triploid segregation pattern (monomorphic triploid F1, or 1:1 segregation) in combination with the arcsine sqrt of the ratio.

	No. genotyped markers
Monomorphic	404
1:1 segregation triploid F1	812
Total	1216

3.3.3: Information about heterozygosity of the parents provided in the last part of the marker name, in combination with the parental arcsine sqrt of the ratio compared with 95% C.I. of the arcsine sqrt of the ratio of the monomorphic tetraploid cluster

The last method which was used to genotype the parents was using the Information about heterozygosity provided in the last part of the marker name, in combination with the parental arcsine sqrt of the ratio compared with 95% C.I. of the arcsine sqrt of the ratio of the monomorphic tetraploid cluster. The origin of the SNP is indicated by the last two numbers in the marker name. This information is based on the RNA-seq data of the ALS individuals. SNPs having a "1" at the end of the marker name were selected based on a polymorphism in ALS01 for that marker, where SNPs selected from a polymorphism in ALS02 have "2" at the end of the marker name.

To define the genotype of ALS01 and ALS02 where the marker name does not supply information about heterozygosity, we used the relation of the parental arcsine sqrt of the ratios in comparison to the monomorphic F1. This means that we assume that ALS01 and ALS02 have the same "relative" genotype when both parents were not significantly different from the tetraploid F1.

In comparison to ALS01 which has only one possible heterozygous genotype, ALS02 can have three different possible heterozygous genotypes (simplex, duplex, and triplex). This makes it harder to define the parental genotype using the information about heterozygosity in the marker name. Since there is only one heterozygous genotype possible for ALS01 (simplex) we can narrow down the possible genotypes of ALS02 when using the relationship of the arcsine sqrt of the ratio between parents and tetraploid F1. This enabled us to genotype ALS01 and/or ALS02 when only the information about heterozygosity of one parent was provided in the marker name.

For markers having a "1", "2", or "12" at the end of the marker name, ALS01 was genotyped as simplex, and ALS02 as duplex when both ALS01 and ALS02 were not significantly different from the tetraploid monomorphic F1, as seen in Figure 3.4 bottom left, bottom right and top right. We used the same p-value criteria as was used for genotyping using triploid segregation pattern in combination with the relation of arcsine sqrt of the ratio between parents and tetraploid F1. This means that a parent was found significantly different from the 95% C.I. of the arcsine sqrt of the ratio

of the tetraploid F1 when the p-value <0.025 or > 0.975. The assigned parental genotypes for each given situation is shown in Table 3.11 (the applied scripts are found in appendix scripts 1.9.2).

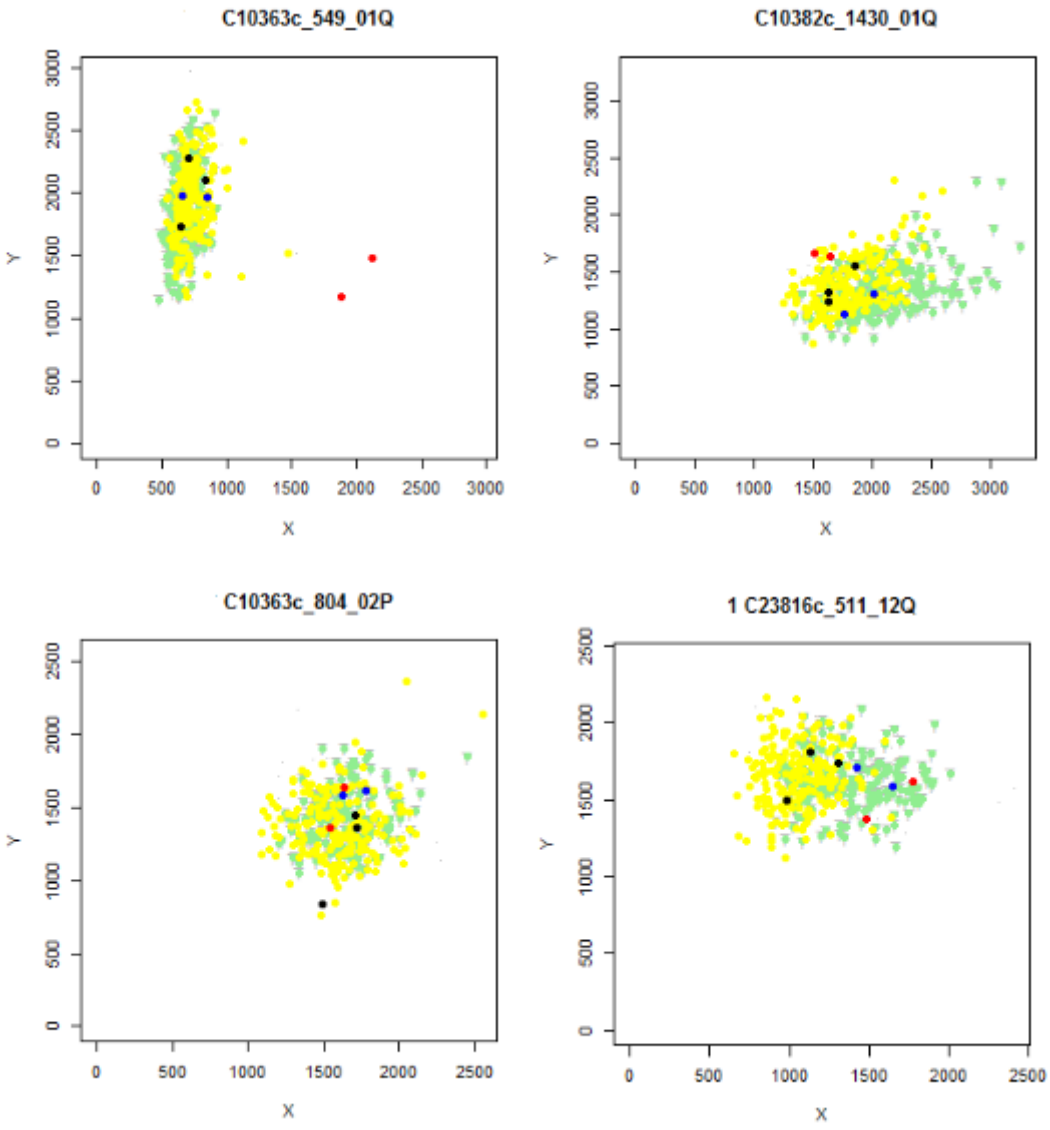


Figure 3.4: Genotyping of ALS01 and ALS02 using the information about heterozygosity in the marker name, in combination with the parental arcsine sqrt of the ratio in relation to the 95% C.I. of the arcsine sqrt of the ratio of the monomorphic tetraploid cluster. Top left: ALS01 is significantly smaller (p-value < 0.025) than the tetraploid F1, where ALS02 is not significantly different from the tetraploid F1. The marker name indicates ALS01 is heterozygous, and can be genotyped as simplex. The genotype of ALS02 is unknown as it could be triplex, or quadruplex. Top right: marker name indicates ALS01 is heterozygous, and because both ALS01 and ALS02 are not significantly different from the tetraploid cluster they could be assigned with the same "relative" genotype (ALS01 simplex, ALS02 duplex). Bottom left: marker name indicates ALS02 is heterozygous, and because both ALS01 and ALS02 are not significantly different from the tetraploid cluster they could be assigned with the same "relative" genotype (ALS01 simplex, ALS02 duplex). Bottom right: marker name indicates that both ALS01 and ALS02 are heterozygous, and because both ALS01 and ALS02 are not significantly different from the tetraploid cluster they could be assigned with the same "relative" genotype (ALS01 simplex, ALS02 duplex).

Table 3.11: Genotyping of ALS01 and ALS02, based on the parental ratio in relation to the arcsine square root of the ratio of the tetraploid cluster, and the information about heterozygosity in the marker name. Assumption that the tetraploid cluster is monomorphic is based on a SD<0.06. The marker name provides information whether a parent is heterozygous. The ratio

of the parents compared to the tetraploid F1 cluster, provides information whether the parents are significantly larger/smaller than the F1 cluster. The combination of both methods help us to genotype the ALS01 and/or ALS02.

		Marker name contains			
		Not 1	1	not 1	1
		Not 2	not 2	2	2
	ALS01=cluster, ALS02=cluster	Both parents unknown	ALS01 simplex, ALS02 duplex	ALS01 simplex, ALS02 duplex	ALS01 simplex, ALS02 duplex
Ratio parents compared with cluster:	ALS01<cluster, ALS02<cluster	Both parents unknown	ALS01 simplex, ALS02 unknown	Both parents unknown	ALS01 simplex, ALS02 simplex or duplex
	ALS01<cluster, ALS02=cluster	ALS01 nulliplex or simplex, ALS02 unknown	ALS01 simplex, ALS02 unknown	Both parents unknown	ALS01 simplex, ALS02 triplex
	ALS01<cluster, ALS02>cluster	ALS01 nulliplex or simplex, ALS02 unknown	ALS01 simplex, ALS02 unknown	Both parents unknown	Not possible
	ALS01>cluster, ALS02<cluster	Both parents unknown	ALS01 simplex, ALS02 nulliplex	Both parents unknown	Not possible
	ALS01>cluster, ALS02=cluster	Both parents unknown	ALS01 simplex, ALS02 unknown	Both parents unknown	ALS01 simplex, ALS02 simplex
	ALS01>cluster, ALS2>cluster	Both parents unknown	ALS01 simplex, ALS02 unknown	Both parents unknown	ALS01 simplex, ALS02 duplex or triplex
	ALS01=cluster, ALS02<cluster	ALS01 simplex, ALS02 nulliplex or simplex	ALS01 simplex, ALS02 unknown	Both parents unknown	ALS01 simplex, ALS02 simplex

In situations where ALS02 should be genotyped as simplex or triplex, only ALS01 was genotyped because of the assumption that the tetraploid F1 is monomorphic (non-segregating). A total of 4251 marker names possess information of ALS01 and/or ALS02 being heterozygous ("01", "12", "02", "23" in the marker name). We were able to genotype 1634 ALS01 parents, and 367 ALS02 parents with the use of the information about heterozygosity in the marker name, the comparison of the parental with the tetraploid F1 arcsine sqrt of the ratio, and the requirement of being a monomorphic tetraploid F1 (arcsine sqrt ratio SD < 0.06). Table 3.12 shows the number of markers with genotyped ALS01 and ALS02.

Table 3.12: number of genotyped parents, based on the information about heterozygosity in the marker name, and the parental arcsine square root of the ratio compared to the tetraploid cluster.

Heterozygosity in name	#No. genotyped ALS01	#No. genotyped ALS02
ALS01 heterozygous name	1306	40
ALS02 heterozygous name	327	327
Total	1634	367

The genotypes of ALS01 and ALS02 were stored in the file: "parentscoring.RData", under the columns 'genoALS01name' (genotypes ALS01 using marker name), and 'genoALS02name' (genotypes ALS02 using marker name).

3.4: Combining the evidence from the different sources into parental genotypes.

In total 5727 markers of the 15999 markers that were genotyped in the triploid F1 but non-genotyped in the tetraploid F1 were identified as having a monomorphic tetraploid F1 pattern. Of these 5727 monomorphic markers, 3947 markers could be genotyped using one or more of the described methods. A lot of these genotyped markers have overlapping or conflicting genotypes between the different used methods. These overlapping genotyped markers enabled us to confirm or reject the applied genotyping method, and gave information about the reliability of the applied methods. However, it was found that all methods were subject to errors.

We wrote a script to construct a final genotype for the parents in which we combine all the parental data together, as seen column “ALS01” and “ALS02” of table 3.13 (the script for assigning final genotypes and conflicting information between used methods can be found in appendix 1.10). We assume that information based on XY-signal intensities was more reliable than the other 2 applied methods which were based on the marker name and triploid segregation pattern, because these 2 methods are more prone to “human” errors (SNP development errors, genotyping errors, shifts, etc.), while the methods based on XY-signal intensities only used primitive data. For this reason the parental genotypes based on the XY- signal intensities were preferred over the genotyping based on the other 2 methods whether or not the genotypes coincide.

Table 3.13: Table with the applied genotypes based on XY-signal intensities (genoALS01XY, and genoALS02XY), triploid segregation pattern (genoALS01seg, and genoALS02seg), and information about heterozygosity in the marker name (genoALS01name, genoALS02name). Final assigned genotypes are shown in columns “ALS01” and “ALS02”, where a genotype of “6” indicates that there is a conflict in the genotype between the methods using information about heterozygosity in marker name and triploid segregation pattern. The column “conflict methods” indicates the methods and parents that are in conflict with each other with the letter “a” for conflict between XY-signal intensity and marker name, “b” for conflict between XY-signal intensity and triploid segregation pattern, or “c” for a conflict between marker name and triploid segregation pattern, followed by the numbers “1” for a conflict in ALS01, “2” for a conflict in ALS02, or “12” for conflict in ALS01 and ALS02. No letter indicates that the final applied genotype is based on only one method, or that the genotypes between the different methods coincide.

MarkerName	segtype	genoALS01XY	genoALS02XY	genoALS01seg	genoALS02seg	genoALS01name	genoALS02name	ALS01	ALS02	conflicting methods
C9977_130_02Q	s1100	0	0	NA	NA	1	2	0	0	a12
C11854_292_03Q	s0110	4	4	1	2	NA	NA	2	4	b12
C11063_469_02P	s0001	NA	4	2	4	1	2	6	4	c1
C19425c_537_01Q	s0100	NA	NA	0	2	1	NA	6	2	c1
C1308_1476_02Q	s0001	NA	NA	2	4	1	2	6	6	c12
C15205c_1011_05Q	s1000	0	0	NA	NA	NA	NA	0	0	NA
C15228c_801_06P	s1100	NA	NA	1	0	NA	NA	1	0	NA
C15268c_182_01Q	s1100	NA	0	1	0	1	NA	1	0	NA
C15268c_902_06Q	s1100	NA	0	1	0	NA	NA	1	0	NA
C152c_646_01Q	NA	NA	0	NA	NA	1	NA	1	0	NA

For the genotyping of parents where the monomorphic markers had been identified with the XY-signal intensity range, we ignored the SD <0.06 threshold. However, when no genotype was available based on the XY-signal intensities, we used the genotypes based on the triploid segregation pattern, and the information about heterozygosity in the marker name. These two methods assumed that the marker is monomorphic in the tetraploid populations because of the SD <0.06 of the arcsine sqrt of the ratio.

We also assumed that the genotyping based on the triploid segregation pattern and the information about heterozygosity in the marker name were equally reliable. When one or more parental genotypes were available, and these genotypes were not in conflict with each other, the genotype was assigned as final genotype genotypes in the columns “ALS01” and/or “ALS02” of table 3.13. Conflicting genotypes between the two methods were rejected, and assigned with a genotype “6”.

The column “conflicting methods” in table 3.13 indicates whether the genotypes based on one of the three methods were in conflict with each other. Conflicts between genotypes based on XY-signal intensities and information about heterozygosity in the marker name were indicated with the letter “a”, where the letter “b” was assigned for a conflict between genotypes based on XY-signal intensities and triploid segregation pattern, the letter “c” for a conflict between genotypes based on information about heterozygosity in the marker name and triploid segregation pattern, or a combination of the letters to indicate that there were multiple conflicts between the methods. The letters were followed by the numbers “1” for conflicting genotypes only in parent ALS01, number “2” for conflicting genotypes only in ALS02 or “12” when both parents have conflicting genotypes between the methods as seen in Table 3.14. No letter in the column “conflicting methods indicates that the final applied genotype is based on only one method, or that the genotypes between the different methods coincide.

Table 3.14: Explanation of the letter indication for the different possible conflicts between the parental genotypes and the different methods.

Conflict between	ALS01	ALS02	Both parents
XY-intensity and marker name	a1	a1	a12
XY-intensity and segregation	b1	b1	b12
Marker name and segregation	c1	c1	c12
XY-intensity and marker name + XY-intensity and segregation	ab1	ab2	ab12
XY-intensity and marker name + marker name and segregation	ac1	ac2	ac12
XY-intensity and segregation + marker name and segregation	bc1	bc2	bc12
XY-intensity and segregation and marker name	abc1	abc2	abc3

Using this method we were able to apply a final genotype to ALS01 for 3261 markers, and to ALS02 for 3365 markers. In total for 3943 markers of the total 15999 markers at least one parent (±25%) could be genotyped, and only 51 of these genotyped markers show conflicting genotypes between at least two of the used methods).

3.5: Construction of a diploid linkage map using 1:1 segregating markers

JoinMap was used for the validation of genotyped markers, and the construction of a diploid linkage map. For the construction of a diploid linkage map, markers were used having a 1:1 segregation in the triploid F1, segregating from the diploid parent (ALS01). This means that we only used markers where ALS01 is heterozygous (simplex), and ALS02 is homozygous (nulliplex or quadruplex) or disomic duplex (aa|AA).

In JoinMap the triploid F1 was treated as a CP (cross pollinator) population that originated from a cross between heterozygous diploid (male), and homozygous tetraploid (female) parents. The tetraploid parent and triploid F1 were treated as diploid samples because we only used markers segregating from the diploid parent (ALS01), having a 1:1 segregation in the triploid F1. The segregation type code for population type CP is <nnxnp> were the locus is heterozygous in the male parent ALS01 (np) and non-segregating from the female parent ALS02 (nn). Depending on the assigned triploid F1 segregation, the triploid F1 genotypes were converted into the genotype codes “nn, or “np”, as shown in Table 3.15. Shifted genotypes deviating from the applied triploid F1, and missing values were converted into “--“, indicating a missing value.

Table 3.15: conversion of applied triploid F1 genotypes into genotype codes necessary for running JoinMap.

		Triploid F1 genotype			
Triploid F1 segregation	<ALS02XALS01>	0	1	2	3
S1100	<nnxnp>	nn	np	--	--
S0110	<nnxnp>	--	nn	np	--
S0011	<nnxnp>	--	--	np	nn

We start with 889 markers having an s1100 (nulliplex x simplex) and s0011 (duplex x triplex) segregation in the triploid F1, where ALS01 was genotyped as simplex (heterozygous), and ALS02 was genotyped as homozygous (nulliplex or quadruplex). Out of these 889 markers, for 262 markers ALS01 and ALS02 were genotyped by fitTetra, and for 627 markers the parents were genotyped using the developed methods in this thesis. The markers genotyped by the methods developed during this thesis were indicated with a letter "M" at the end of the marker name to be able to follow the markers during the linkage mapping.

Linkage maps were calculated using ML (maximum likelihood) mapping of linkage groups >3 markers at a LOD score of 8.0, 879 markers could be assigned to 25 linkage groups.

First results of linkage mapping indicated that the information about heterozygosity in marker name was accurate. We found 13 linkage groups where the majority of the markers had the heterozygous information "01" at the end of the marker name suggesting that most of the parental genotyping was correct, and that the segregation comes from the diploid parent ALS01.

12 linkage groups contained markers having only information about heterozygosity of other parents than ALS01 which might suggest that the parental genotyping was incorrect. To check, XY-scatter plots were drawn of markers were the information about heterozygosity at the end of the marker name suggested that other parents (not ALS01) were heterozygous. Markers or complete linkage groups were removed from the diploid linkage map when ALS02 was found to be heterozygous (except disomic duplex AA|aa), and/or ALS01 homozygous. Also markers having >30 missing values out of the 156 triploid samples were removed. A total of 332 markers were removed from the diploid linkage map, from which at least 195 markers were probably segregating from ALS02, leaving 557 markers that have been assigned to one of the 13 linkage groups.

Since information about heterozygosity in the marker name is found to be accurate, 51 markers with missing parental genotypes showing s1100 or s0011 segregation in the triploid F1 (which were unable to be genotyped by fitTri and the developed methods in this thesis), and having "01" at the end of the marker name were added. After removal of markers having >30 missing values, 46 markers could be mapped into one of the 13 linkage groups. This resulted in 618 markers forming backbone of the diploid linkage map.

After the backbone was formed with markers having s1100 and s0011 segregation in the triploid F1, we added 415 markers showing a S0110 segregation in the triploid F1 segregating from ALS01. Because we relied on the information about heterozygosity in the marker name, we added all markers having "1"at the end of the marker name whether they have parental genotypes or not. The added markers were indicated with "X" at the end of the marker name to be able to follow the markers during the linkage mapping. For these markers, ALS02 is expected to have a non-segregating disomic duplex (AA|aa) genotype and ALS01 is simplex. The genotypes of the triploid F1 assigned

with a s0110 segregation were converted into the genotype codes “nn”, “np”, and “--“ as explained in Table 3.15.

Out of the 1033 markers (618+415), 1020 markers could be assigned to one of the 13 linkage groups. Linkage groups were inspected, and individual markers were evaluated. The two probes were expected to have low recombination frequencies (<1 cM distance on map), and the same phase. However, for a lot of markers this was not the case, as seen in Table 3.16.

Table 3.16: The difference in phase between the P and Q probe of the same SNP marker. Markers having the same contig and SNP position (P and Q probe) were expected to share the same phase and position on the linkage group. The “X” at the end of the marker name indicated that the marker has an s0110 segregation in the triploid F1, and was added to the linkage map whether or not the parents were genotyped. The “M” at the end of the marker name indicated that the parents were genotyped using the methods developed during this thesis.

Marker	Genotype code	Phase	Position (cM)
C+++93c_402_01Qn_X	<nnxnp>	(-0)	55.079
C+++93c_565_01Pn_M	<nnxnp>	(-1)	55.079
C+++93c_565_01Qn_X	<nnxnp>	(-0)	55.079
C???8c_582_01Pn_M	<nnxnp>	(-0)	60.143
C???8c_582_01Qn_X	<nnxnp>	(-1)	61.750

Visual inspection of 618 XY-scatter plots of markers showing s1100 and s0011 segregation in triploid F1, and 415 XY-scatter plots of markers showing s0110 segregation in the triploid F1 revealed shifted triploid F1 genotypes. These shifted genotypes explained the difference in phase between the two probes of SNP markers. Out of the 618 markers having an s1100 or s0011 segregation, 8 markers were genotyped incorrectly. However, only 42 out of the 415 markers having an assigned s0110 triploid segregation were genotyped correctly in the triploid F1. This means that ± 90% of the triploid markers with an s0110 segregation were incorrectly genotyped and should have an s1100 or s0011 segregation.

An R-script was developed to correct the shifted s0110 markers (appendix scripts 1.11). The script tested whether the arcsine sqrt of the ratio of ALS01 was significantly different from the 95% C.I. of the arcsine sqrt of the ratio of the TC F1. We expected ALS01 and ALS02 to have the same “relative” genotypes (ALS01 simplex, ALS02 disomic duplex) when the marker shows an s0110 segregation in the triploid F1. This means ALS01 and ALS02 were expected to be not significantly different from the TC F1 when the triploid F1 were genotyped correctly in fitTri (Figure 3.5 top left). Markers were expected to have a shifted s1100 segregation in the triploid F1 when ALS01 was significantly larger than the TC F1 (Figure 3.5 bottom left). A shifted s0011 segregation in the triploid F1 was expected when ALS01 was significantly smaller than the TC F1 (Figure 3.5 top right).

We only tested against the TC population because ALS02 is one of the parents of this population. For some markers, the TC and TP F1 do not show the same segregation. Plotting the combined TC and TP F1 would only increase the SD of the arcsine sqrt of the ratio, while plotting of the TC F1 together with ALS01 and ALS02 would be sufficient for this application.

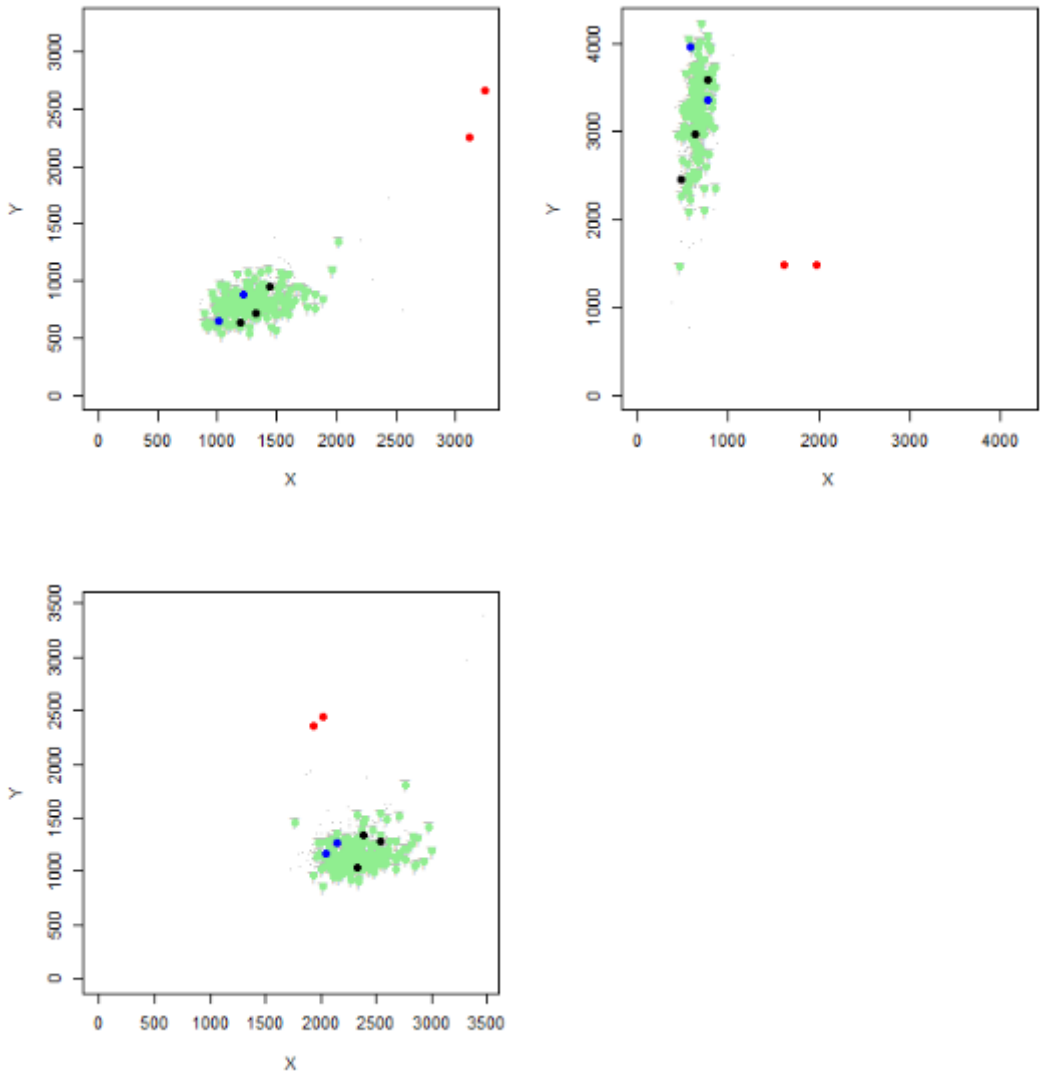


Figure 3.5: XY-scatter plots of the TC F1 (green) together with ALS01 (red), ALS02 (blue), and ALS03 (black). Markers were genotyped as segregating s0110 (duplex, simplex) from ALS01 in the triploid F1 (information about heterozygosity of ALS01 in marker name). Top left: the arcsine sqrt of the ratio of ALS01 and ALS02 is not significantly different from the monomorphic TC F1, as expected for markers segregating s0110 in the triploid F1. Top right: ALS01 is significantly smaller than the monomorphic TC F1, where ALS02 is not significantly different from TC F1. The segregation in the triploid F1 is expected to be s0011. Bottom: ALS01 is significantly larger than the monomorphic TC F1, where ALS02 is not significantly different from TC F1. The segregation in the triploid F1 is expected to be s1100.

Linkage maps were calculated again using ML (maximum likelihood) mapping of linkage groups >3 markers at a LOD score of 6.0 markers having >30 missing values were discarded, and out of the 1033 corrected markers, 988 markers could be assigned to one of the 11 linkage groups from which 668 markers were non-identical, as seen in Table 3.17. The linkage map was recalculated using Regression mapping with Haldane´s mapping function (diploid linkage map can be found in Appendix 3). The total length of the diploid linkage map was estimated at 1015.9 cM, having an average density of ± 1 marker/cM, (0.66 markers/cM without identical markers). The gaps present in the diploid linkage map range between 3 and 50 cM.

Table 3.17: Number of (non-identical) assigned markers and the total length (in cM) of each linkage group on the diploid linkage map of Alstroemeria.

Linkage group	No. markers (non-identical)	Length (cM)
1	169 (104)	228.8
2	122 (55)	112.5
3	60 (35)	9.2
4	51 (46)	98.4
5	29 (21)	44.6
6	5 (5)	12.7
7	183 (102)	90.7
8	75 (69)	141.0
9	14 (13)	20.4
10	177 (146)	101.6
11	103 (72)	156.0
total	988 (668)	1015.9

4: Discussion

During this thesis we studied the possibility to genotype parents of the triploid F1 of markers which are rejected by fitTetra because of a monomorphic F1 cluster by using different sources of information such as: the arcsine sqrt of the ratio, XY-signal intensities, information about heterozygosity in the marker name, and the triploid segregation pattern. This was followed by the evaluation of triploid and parental genotypes of markers segregating 1:1 from the heterozygous diploid parent in the triploid F1 by means of mapping and XY scatter plots, and finally a diploid linkage map was constructed.

Different underlying (sometimes unknown) reasons resulted in missing parental genotypes in the triploid data set. One possible reason is that some SNPs are genotyped in fitTri but rejected in fitTetra. It is also possible that the SNP is genotyped in fitTetra and fitTri, but the parents of the triploid F1 could not be genotyped. Another possible reason is that some SNPs are genotyped by fitTetra and fitTri, but parental scores did not explain the segregation type in the triploid population. We studied the SNPs which were genotyped by fitTri, but rejected in the tetraploid data set by fitTetra. A possible reason for rejection by fitTetra is the presence of a monomorphic cluster, exceeding the maximum fraction of allowed samples in one peak. Normally these monomorphic SNPs are not useful to genotype linkage mapping requires segregating markers. However, the parents of the triploid F1 were genotyped together with the tetraploid F1 because the relative allele dosages of the triploid F1 and the parents do not coincide, making it unable to genotype the triploid F1 together with their parents. Genotyping of the triploid F1 by fitTri, and rejection in the tetraploid data set by fitTetra leads to missing parental values in the triploid data set. We identified 15999 markers which were genotyped in the triploid F1, but rejected in the tetraploid data set. This was followed by the identification of markers with a monomorphic cluster in the tetraploid F1.

4.1: Monomorphic markers

Two methods were used to identify monomorphic markers in the tetraploid data set. The first method uses the SD of the arcsine sqrt of the ratio of each marker. We came with a threshold of SD < 0.06 to identify monomorphic markers by evaluating XY-scatter plots with increasing SD of the arcsine sqrt of the ratio of the 15999 rejected markers in the tetraploid data set. This method was subject to errors because we did not evaluate all the markers in a SD group, and there is no clear threshold to distinguish between monomorphic and segregating markers. Falsely identified monomorphic markers, while it actually was segregating, results in wrong parental genotypes. We assume that parental samples and the tetraploid F1 have the same genotype when they were not significantly different from each other, but when the tetraploid F1 was actually segregating in multiple genotypes, it was impossible to identify the correct parental genotype. The SD threshold was selected based on groups from which we thought that more than 90% of the markers were monomorphic, and resulted in 4459 monomorphic markers. However we found a lot of homozygous markers in groups with SD > 0.06 which showed big ranges of either the X or Y-signal intensity resulting in a large SD. These markers could not be identified by using the SD of the arcsine sqrt of the ratio, so we used the 95% C.I. of the length of the XY signal intensities. This enabled us to identify 2560 monomorphic homozygous markers, from which 1226 markers have a SD > 0.06. In total, 5685 of the 15999 markers were identified as having a monomorphic cluster in the tetraploid F1.

4.2: Genotyping

The parents of monomorphic markers identified with the XY signal intensities were genotyped as having a homozygous genotype when the X or Y-signal intensity did not significantly differ from the 95% C.I. of the length of the X or Y signal intensity of the tetraploid F1. When a triploid segregation

pattern was available, and the homozygous genotype of the genotyped parent matched the expected corresponding genotype, we were able to genotype the other parent which was significantly different from the homozygous tetraploid F1. In most cases ALS02 was not significantly different from the homozygous tetraploid cluster, as expected because ALS02 is also a parent of the TC population. When both ALS01 and ALS02 were not significantly different from the homozygous tetraploid cluster, which was the case for 1248 markers, we expected the triploid F1 to be homozygous. This was unexpected because fitTri should reject markers which have a monomorphic triploid cluster. When we look at the assigned segregations in the file "tri_combscores_noRedundant.DAT", we see that monomorphic clusters in the triploid F1 were not rejected by fitTri, so the outcome confirms this observation. In total we were able to genotype 2477 markers by using the XY-signal intensities.

The second method applied for genotyping the parents was based on the triploid segregation pattern, and used the arcsine sqrt of the ratio to distinguish between the multiple possible parental genotypes. As shown in table 3.4, and 3.5, we expect ALS02 to have a homozygous, or disomic duplex genotype to enable a monomorphic cluster in the tetraploid F1. This means that only non-segregating (s1000, s0100, s0010, and s0001) and 1:1 segregations were expected in the triploid F1 (table 3.8). In total 1216 monomorphic and 1:1 segregating markers could be genotyped by using the triploid segregation pattern. However, the reliability of this method is questionable because we have to rely on the correctness of the assigned triploid segregation pattern. Unfortunately this was not the case as linkage mapping and visual inspection of XY-scatter plots indicated that ± 90% (373 out of 415) of markers genotyped with a 1:1 duplex x simplex segregation in the triploid F1 were actually shifted nulliplex x simplex markers. Markers assigned with 1:1 nulliplex x simplex, and duplex x triplex segregation had a much lower amount of shifted genotypes, with only ± 2% (8 out of 618) incorrect genotyped triploid samples.

Another questionable aspect is the assignment of parental genotypes of markers with a 1:1 segregation in the triploid F1 by using the arcsine sqrt of the ratio to distinguish between the multiple parental genotypes. The 1:1 segregation in the triploid F1 is caused by a heterozygous genotype of ALS01, ALS02, or both in cases where ALS02 is disomic duplex (table 3.8). We assumed that either ALS01, ALS02, or both parents were not significantly different from the tetraploid F1, and that the other parent was significantly larger/smaller than the tetraploid F1 (Figure 3.3). In some cases this would mean that ALS02 has a simplex or triplex genotype, which contradicts the assumption that the marker is monomorphic in the tetraploid F1 (SD < 0.06). This means that the tetraploid F1 was not monomorphic, or that the comparison between the parental and tetraploid F1 arcsine sqrt of the ratio was not correct.

Visual inspection of XY-scatter plots showed that the TC and TP F1 behave differently. A monomorphic cluster in the TC F1 does not mean that the TP F1 is also monomorphic, and v.v. Especially for SNPs selected based on heterozygosity of the parents of the tetraploid F1 (ALS02, ALS03, ALS04 and ALS05). In the beginning of this thesis we thought that it would be better to keep the TC and TP F1 together because it was proven useful in other species such as potato. However during the end of this thesis we moved away from this idea, and treated the populations separately. This means that calculations were done over using the TC F1 only. Visual inspection of XY-scatter plots also revealed that some markers had big differences in parental XY signal intensities (Figure 3.5 top left). Although we used the arcsine sqrt of the ratio, it was possible that the parental genotype was different than the arcsine sqrt of the ratio does seem occur. Especially for markers with low parental XY-signal intensities, it was almost impossible to identify the correct genotypes (even by visual inspection of XY scatter plots).

The third method applied to retrieve the parental genotypes was based on the information about heterozygosity in the marker name in combination with the arcsine sqrt of the ratio. This method used the information about heterozygosity in the marker name to indicate from which heterozygous parent the SNP was selected. By using the arcsine sqrt of the ratio and the possible heterozygous genotype of ALS01, we were able to retrieve the genotype of ALS01 and/or ALS02 of markers having a “1”, “12” or “2” at the end of the marker name. At the beginning of this thesis it was unclear how reliable this information was, and we assumed it was as reliable as the triploid segregation pattern. However, linkage mapping revealed that the information about heterozygosity in the marker name was a more reliable source of information then the triploid segregation pattern. At the end of this thesis we genotyped all diploid parents with a heterozygous genotype when the marker name indicated that the SNP was selected from ALS01, and evaluated the parental and triploid F1 genotypes by linkage mapping. This was impossible for markers selected from ALS02, because multiple genotypes were possible.

4.3: Linkage mapping

Linkage mapping was used for the validation of parental and triploid F1 genotypes, and the calculation of a diploid linkage map. We started with nulliplex x simplex and duplex x triplex segregating markers, with parents genotyped by fitTetra and the methods developed during this thesis. By using visual inspection of XY-scatter plots of linkage groups having more than 90% markers with information about heterozygosity in the marker name of other parents then ALS01, we were able to identify wrong genotyped markers. This clarified the accuracy of the information about heterozygosity in the marker name, and enabled us to use even more SNP markers selected from ALS01 with missing parental genotypes. The double recombination frequencies were used to identify 10 triploid samples with deviating triploid genotypes. A possible reason for this abnormality could be aneuploidy of the triploid samples.

After we calculated a linkage map with 618 markers with a nulliplex x simplex or duplex x triplex segregation having information about heterozygosity of ALS01 in the marker name, we added duplex x simplex markers with information about heterozygosity of ALS01 in the marker name. Linkage mapping in combination with visual inspection of XY-scatter plots enabled the identification of shifted nulliplex x simplex and duplex x triplex markers. Adding of markers with a duplex x simplex segregation to the diploid backbone with nulliplex x simplex and duplex x triplex segregating markers resulted in non-coinciding phases between the two probes, indicating a shifted marker. For duplex x simplex segregating markers in the triploid F1, the segregation could be caused by ALS01 resulting in the same genotypes of ALS01 (simplex) and ALS02 (disomic duplex), or by ALS02 which would result in a simplex or triplex genotype of ALS02, and a homozygous genotype of ALS01. The information about heterozygosity in the marker name suggests that the segregation is caused by ALS01, so we expected ALS01 and ALS02 to have the same genotypes. Visual inspection of the XY-scatter plots of the TC F1 revealed that the segregation was indeed caused by ALS01, but also showed homozygous TC clusters of >90% of the duplex x simplex markers. This confirmed that the triploid F1 was shifted, as a homozygous genotype of ALS02 would result in a nulliplex x simplex or duplex x triplex genotype in the triploid F1, and not in a duplex x simplex genotype.

The large amount of wrong genotyped markers with a duplex x simplex segregation and information about heterozygosity from ALS01 was a reason for us to look if at the other markers with duplex x simplex segregation without information about heterozygosity of ALS01. However, visual inspection of XY-scatter plots revealed no clear pattern as was found with the markers having information about heterozygosity of ALS01, making it not possible to use the same R-script.

An R-script was made for the correction of the shifted triploid F1 with their parents, resulting in 373 (out of 415) corrected markers. The diploid linkage map calculated during this thesis contained 668 non-identical markers divided over 11 linkage groups, with a maximum gap of 50cM, and a total length of 1016cM. We expected 8 large linkage groups representing the 8 chromosomes of Alstroemeria, but were unable to link the linkage groups at this point. Possible reasons for the big gaps and the unlinked linkage groups could be the presence of recombination Hot Spots, or the lack of linking markers.

Literature

Acquaah G (2009) Principles of plant genetics and breeding. John Wiley & Sons,

Bennetzen JL, Ma J, Devos KM (2005) Mechanisms of recent genome size variation in flowering plants. Annals of botany 95 (1):127-132

Blanca J, Canizares J, Roig C, Ziarsolo P, Nuez F, Pico B (2011) Transcriptome characterization and high throughput SSRs and SNPs discovery in Cucurbita pepo (Cucurbitaceae). BMC Genomics 12. doi:10.1186/1471-2164-12-104

Bourke PM, Voorrips RE, Visser RG, Maliepaard C (2015) The Double Reduction Landscape in Tetraploid Potato as Revealed by a High-Density Linkage Map. Genetics:genetics. 115.181008

Broertjes C, Verboom H (1974) Mutation breeding of Alstroemeria. Euphytica 23 (1):39-44

Buitendijk J, Ramanna M, Jacobsen E Micropropagation ability: towards a selection criterion in Alstroemeria breeding. In: VI International Symposium on Flower Bulbs 325, 1992. pp 493-498

Buitendijk JH, Boon EJ, Ramanna MS (1997) Nuclear DNA Content in Twelve Species of Alstroemeria L. and Some of their Hybrids. Annals of Botany 79 (4):343-353. doi:10.1006/anbo.1996.0345

Buitendijk JH, Pinsonneaux N, van Donk AC, Ramanna MS, van Lammeren AAM (1995) Embryo rescue by half-ovule culture for the production of interspecific hybrids in Alstroemeria. Scientia Horticulturae 64 (1–2):65-75. doi:http://dx.doi.org/10.1016/0304-4238(95)00827-2

Chacon J, Sousa A, Baeza CM, Renner SS (2012) Ribosomal DNA distribution and a genus-wide phylogeny reveal patterns of chromosomal evolution in Alstroemeria (Alstroemeriaceae). Am J Bot 99 (9):1501-1512. doi:10.3732/ajb.1200104

Dhawan O, Lavania U (1996) Enhancing the productivity of secondary metabolites via induced polyploidy: a review. Euphytica 87 (2):81-89

Doerge R, Craig BA (2000) Model selection for quantitative trait locus analysis in polyploids. Proceedings of the National Academy of Sciences 97 (14):7951-7956

Fisher RA (1947) The Theory of Linkage in Polysomic Inheritance. Philosophical Transactions of the Royal Society of London B: Biological Sciences 233 (594):55-87. doi:10.1098/rstb.1947.0006

Gallais A (2003) Quantitative genetics and breeding methods in autopolyploid plants. Quae,

Hackett C, Bradshaw J, McNicol J (2001) Interval mapping of quantitative trait loci in autotetraploid species. Genetics 159 (4):1819-1832

Hackett CA, McLean K, Bryan GJ (2013) Linkage Analysis and QTL Mapping Using SNP Dosage Data in a Tetraploid Potato Mapping Population. PLoS ONE 8 (5):e63939. doi:10.1371/journal.pone.0063939

Haldane JBS (1930) Theoretical genetics of autopolyploids. Journ of Gen 22 (3):359-372. doi:10.1007/BF02984197

Han T-H, van Eck H, de Jeu M, Jacobsen E (2002) The construction of a linkage map of Alstroemeria aurea by AFLP markers. Euphytica 128 (2):153-164. doi:10.1023/A:1020921103374

Han TH, van Eck HJ, De Jeu MJ, Jacobsen E (1999) Optimization of AFLP fingerprinting of organisms with a large-sized genome: a study on Alstroemeria spp. Theoret Appl Genetics 98 (3-4):465-471. doi:10.1007/s001220051093

Hang A, Tsuchiya T (1988) Chromosome Studies in the Genus Alstroemeria. Plant Breeding 100 (4):273-279. doi:10.1111/j.1439-0523.1988.tb00253.x

Kew RBG, Gardens VK, Wakehurst V (2012) Plant DNA C-values Database.

Lu C, Bridgen MP (1997) Chromosome doubling and fertility study of Alstroemeria aurea× A. caryophyllaea. Euphytica 94 (1):75-81

Luo ZW, Zhang Z, Zhang RM, Pandey M, Gailing O, Hattemer HH, Finkeldey R (2006) Modeling Population Genetic Data in Autotetraploid Species. Genetics 172 (1):639-646. doi:10.1534/genetics.105.044974

Mann H, Iorizzo M, Gao L, D'Agostino N, Carputo D, Chiusano ML, Bradeen JM (2011) Molecular linkage maps: strategies, resources and achievements. Genetics, Genomics and Breeding of Crop Plants: Potato:68-89

Meyer RC, Milbourne D, Hackett CA, Bradshaw JE, McNichol JW, Waugh R (1998) Linkage analysis in tetraploid potato and association of markers with quantitative resistance to late blight (Phytophthora infestans). Mol Gen Genet 259 (2):150-160. doi:10.1007/s004380050800

Ragoussis J (2009) Genotyping technologies for genetic research. Annual review of genomics and human genetics 10:117-133

Ramanna MS The role of sexual polyploidization in the origins of horticultural crops: Alstroemeria as an example. In: Proc. Workshop Gametes with somatic chromosome number in the evolution of and breeding of polyploid polysomic species: achievements and perspectives. Perugia, Italy (1992) 83-90., 1992.

Ripol M, Churchill G, Da Silva J, Sorrells M (1999) Statistical aspects of genetic mapping in autopolyploids. Gene 235 (1):31-41

Shahin A, Arens P, Van Heusden AW, Van Der Linden G, Van Kaauwen M, Khan N, Schouten HJ, De Weg V, Eric W, Visser RG (2011) Genetic mapping in Lilium: mapping of major genes and quantitative trait loci for several ornamental traits and disease resistances. Plant breeding 130 (3):372-382

Shahin A, van Kaauwen M, Esselink D, Bargsten JW, van Tuyl JM, Visser RG, Arens P (2012) Generation and analysis of expressed sequence tags in the extreme large genomes Lilium and Tulipa. BMC Genomics 13 (1):1-13. doi:10.1186/1471-2164-13-640

Stam P (1993) Construction of integrated genetic linkage maps by means of a new computer package: Join Map. The plant journal 3 (5):739-744

Stift M, Berenos C, Kuperus P, van Tienderen PH (2008) Segregation Models for Disomic, Tetrasomic and Intermediate Inheritance in Tetraploids: A General Procedure Applied to Rorippa (Yellow Cress) Microsatellite Data. Genetics 179 (4):2113-2123. doi:10.1534/genetics.107.085027

Van Ooijen J (2006) JoinMap 4. Software for the calculation of genetic linkage maps in experimental populations Kyazma BV, Wageningen, Netherlands

Voorrips R (2002) MapChart: software for the graphical presentation of linkage maps and QTLs. Journal of heredity 93 (1):77-78

Voorrips RE, Gort G, Vosman B (2011) Genotype calling in tetraploid species from bi-allelic marker data using mixture models. BMC bioinformatics 12 (1):1

Wendel J, Doyle J (2005) Polyploidy and evolution in plants. Plant diversity and evolution: genotypic and phenotypic variation in higher plants:97

Wu KK, Burnquist W, Sorrells ME, Tew TL, Moore PH, Tanksley SD (1992) The detection and estimation of linkage in polyploids using single-dose restriction fragments. Theoret Appl Genetics 83 (3):294-300. doi:10.1007/BF00224274

Wu R, Ma C-X, Casella G (2004) A bivalent polyploid model for mapping quantitative trait loci in outcrossing tetraploids. Genetics 166 (1):581-595

www.floraholland.nl (2014) Flora Holland Auction Kengetallen 2014. https://www.floraholland.com/media/4213134/Floraholland_Kengetallen_2014_NL.pdf. Accessed 25/10 2015

Xiao D (2015) Quality criteria for SNP markers in tetraploid species. Laboratory of Plant Breeding, Wageningen University and Research Centre,, Wageningen, the Netherlands

Appendix 1: R scripts

1.1: Creating a list for identification of sample names (parents and F1) in data files

```
setwd("~/alstroemeria")
source("fitTetra_preparation_20141103.r")

samples <- readDatfile("2177A_Arens_SamplesheetPL_20140918-rv.dat")
pop <- list()
pop$TC <- list()
pop$TC$F1 <- sort(setdiff(samples$sampcode[samples$material=="tetra-cut"], #198
 "TC-134")) #occurs in samplefile but not in score file
pop$TC$P1 <- sort(as.character(samples$sampcode[substring(samples$sampcode,1,5)
 == "ALS03"])) #ALS03a ALS03b ALS03c
pop$TC$P2 <- sort(as.character(samples$sampcode[substring(samples$sampcode,1,5)
 == "ALS02"])) #ALS02a ALS02b
#20150309: Arwa Shahin reported that P1 and P2 of the TC population are reversed:
#actually ALS02 is parent 1 (mother) and ALS03 is parent 2 (father)
#In our analyses we stick to the original, incorrect order to avoid confusion
#between versions

pop$TP <- list()
pop$TP$F1 <- sort(setdiff(samples$sampcode[samples$material %in%
 c("tetra-pot-short", "tetra-pot-tall")], #193
 "TP-047")) #occurs in samplefile but not in score file
#pop$TP$P1 <- samples$sampcode[substring(samples$sampcode,1,5) == "ALS04"] #ALS04a
ALS04b
pop$TP$P1 <- "ALS04a" #not ALS04b: incorrect
#pop$TP$P2 <- samples$sampcode[substring(samples$sampcode,1,5) == "ALS05"] #ALS05a
ALS05b ALS05c
pop$TP$P2 <- c("ALS05b", "ALS05c") # not ALS05a: incorrect

pop$tri <- list()
pop$tri$F1 <- sort(as.character(samples$sampcode[samples$material=="triploid"]))
#151
pop$tri$P1 <- sort(as.character(samples$sampcode[substring(samples$sampcode,1,5)
 == "ALS01"])) #ALS01a ALS01b
pop$tri$P2 <- sort(as.character(samples$sampcode[substring(samples$sampcode,1,5)
 == "ALS02"])) #ALS02a ALS02b
pop$di <- c("ALS01a", "ALS01b", "ALS06", "ALS07", "ALS24") #the diploid samples
# make list called “pop” containing the 4 population types (diploid, triploid, TC
and TP), which contain the sample names stored in the DAT file
"2177A_Arens_SamplesheetPL_20140918-rv.dat". This list enables the identification
of TC (pop$TC$F1), TP (pop$TC$F1), and triploid (pop$tri$F1) F1 samples, and the
parents ALS01 (pop$tri$P1), ALS02 (pop$tri$P2) or (pop$TC$P2), ALS03 (pop$TC$P1),
ALS04 (pop$TP$P1), ALS05 (pop$TP$P2).
```

1.2: Identify markers where the triploid F1 were genotyped by fitTri, but were the tetraploid F1 could not be genotyped by fitTetra.

```
load("~/alstroemeria/triploid data/comb3x.Rdata")
genotyped3xmarkers <- unique(as.character(comb3x$MarkerName[comb3x$SampleName %in%
c(pop$tri$F1) & comb3x$geno %in% c(1, 2, 3, 4, 5),])) #33483 genotyped markers
save(genotyped3xmarkers, file= "genotyped3xmarkers.RData")
#make and store vector with marker names from comb3x file, where triploid F1
samples were genotyped (genotype = 1, 2, 3, 4, or 5)


load("E:/comb4x.RData")
TCgenotyped4xmarkers <- unique(as.character(comb4x$MarkerName[comb4x$MarkerName
%in% genotyped3xmarkers & comb4x$geno %in% c(1, 2, 3, 4, 5) & comb4x$SampleName
%in% c(pop$TC$F1)]))

TPgenotyped4xmarkers <- unique(as.character(comb4x$MarkerName[comb4x$MarkerName
%in% genotyped3xmarkers & comb4x$geno %in% c(1, 2, 3, 4, 5) & comb4x$SampleName
%in% c(pop$TP$F1)]))
#make vectors of marker names from comb4x file, where at least one tetraploid F1
was genotyped (genotype = 1, 2, 3, 4, or 5), both for TC “TCgenotyped4xmarkers” and
TP populations “TPgenotyped4xmarkers” separately.

genotypedTCTPmarkers <- unique(as.character(comb4xD$MarkerName[comb4xD$MarkerName
%in% TCgenotyped4xmarkers & comb4x$MarkerName %in% TPgenotyped4xmarkers]))
# combine markers where both TC and TP F1 were genotyped from vectors
“TCgenotyped4xmarkers”, and “TPgenotyped4xmarkers”.

rejectedmarkers <- unique(as.character(comb4x$MarkerName[comb4xB$MarkerName %in%
genotyped3xmarkers & !(comb4xB$MarkerName %in% genotypedTCTPmarkers)]))
```

```
save(rejectedmarkers, file= "rejectedmarkers.RData")

#make and save vector “rejectedmarkers” of marker names (15999 markers names) where
both TC and TP F1 are not genotyped, but triploid F1 were genotyped, by selecting
marker names from “comb4x” file that are in vector “genotyped3xmarkers” but not in
the vector “genotypedTCTPmarkers”

rejectedmarkerset <- comb4x[comb4x$MarkerName %in% c(rejectedmarkers),]
rejectedmarkerset$asinsqr <- asin(sqrt(rejectedmarkerset$ratio))
#make reduced new data.frame called “rejectedmarkerset” from file “comb4x”
selecting only marker names present in vector “rejectedmarkers”, and create new
column “asinsgr” containing the calculated arcsine square root of the ratio of each
sample.
```

1.3: Calculation of mean and SD of arcsine square root of the ratio, and XY-signal intensities

```
pvaluesXYparents = data.frame(rejectedmarkers,
mx=rep(0, length(rejectedmarkers)),
sx=rep(0,length(rejectedmarkers)),
my=rep(0, length(rejectedmarkers)),
sy=rep(0,length(rejectedmarkers)))

pvaluesRatioparents = data.frame(rejectedmarkers,
mr=rep(0, length(rejectedmarkers)),
sr=rep(0,length(rejectedmarkers)))

for(mrk in 1:length(rejectedmarkers)) {
 ar <- rejectedmarkerset$X[rejectedmarkerset$MarkerName == rejectedmarkers[mrk] &
rejectedmarkerset$SampleName %in% c(pop$TP$F1, pop$TC$F1)]

 ak <- rejectedmarkerset$Y[rejectedmarkerset$MarkerName == rejectedmarkers[mrk] &
rejectedmarkerset$SampleName %in% c(pop$TP$F1, pop$TC$F1)]

 al <- rejectedmarkerset$asinsqr[rejectedmarkerset$MarkerName ==
rejectedmarkers[mrk] &
rejectedmarkerset$SampleName %in% c(pop$TP$F1, pop$TC$F1)]


 mx <- mean(ar)
 my <- mean(ak)
 sx <- sd(ar)
 sy <- sd(ak)
 mr <- mean(al)
 sr <- sd(al)

 pvaluesXYparents [mrk,2] <- mx
 pvaluesXYparents [mrk,3] <- sx
 pvaluesXYparents [mrk,4] <- my
 pvaluesXYparents [mrk,5] <- sy
 pvaluesRatioparents[mrk,6] <- mr
 pvaluesRatioparents[mrk,7] <- sr
}
Make data.frame named “pvaluesXYparents”, containing marker name, and calculated
mean, and SD of arcsine sqrt of the ratio of combined tetraploid (TC and TP) F1,
for each marker name present in data.frame “rejectedmarkerset”, and vector
“rejectedmarkers”.
Make data.frame named “pvaluesXYparents”, containing marker name, and calculated
mean, and SD of X and Y signal intensities of combined tetraploid (TC and TP) F1,
for each marker name present in data.frame “rejectedmarkerset”, and vector
“rejectedmarkers”.
```

1.4: Setting SD of the arcsine sqrt of the ratio threshold for identification monomorphic tetraploid markers.

```
a<- as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$sr< 0.02])
b<- as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$sr>0.02 &
pvaluesRatioparents$sr<0.03][1:6])
.
.
.
zz<- as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$sr>0.27 &
pvaluesRatioparents$sr<0.28][1:6])
zzz<- as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$sr>0.28 ])
```

```
#creates vectors “a” till “zzz” with maximum 6 marker names from data.frame
“pvaluesRatioparents” having increasing SD of arcsine sqrt of the ratio.

a<- rejectedmarkerset[rejectedmarkerset$MarkerName %in% a &
rejectedmarkerset$SampleName %in% c(pop$TC$F1, pop$TP$F1, pop$TC$P1, pop$TC$P2,
pop$tri$P1),]
b<- rejectedmarkerset[rejectedmarkerset$MarkerName %in% b &
rejectedmarkerset$SampleName %in% c(pop$TC$F1, pop$TP$F1, pop$TC$P1, pop$TC$P2,
pop$tri$P1),]
.
.
.
zz<- rejectedmarkerset[rejectedmarkerset$MarkerName %in% zz &
rejectedmarkerset$SampleName %in% c(pop$TC$F1, pop$TP$F1, pop$TC$P1, pop$TC$P2,
pop$tri$P1),]
zzz<- rejectedmarkerset[rejectedmarkerset$MarkerName %in% zzz &
rejectedmarkerset$SampleName %in% c(pop$TC$F1, pop$TP$F1, pop$TC$P1, pop$TC$P2,
pop$tri$P1),]
#creates data.frames with selected marker names and tetraploid F1 and parents, at
increasing SD clusters, necessary for plotting XY-scatter plots.


drawXYplots(dat= a,
 markernames=levels(factor(a$MarkerName)),
 out="a",
 genocol=get.genocol(),
 sel.samples=pop$TC$F1, pop$TP$F1,
 sample.groups=list(pop$TC$F1, pop$TP$F1, P1=pop$tri$P1, P2=pop$tri$P2, pop$TC$P1),
 groups.col=c("lightgreen", "orange"), groups.pch=16)
#plotting of XY-scatter plots of samples (TC, TP, parents) in data.frames “a” till
“zzz”, using function “drawXYplots” from “fitTetra_preparation_20141103.r”.
```

1.5: Calculation of the probability values for the parental XY-signal intensities compared to the normal distribution of the tetraploid F1

```
names(pvaluesXYparents)[1]<- c("MarkerName")
#renames column “rejectedmarkers” in data.frame “pvaluesXYparents” into
“MarkerName”

pvaluesXYparents$rangX <- qnorm(0.975, pvaluesXYparents$mx, pvaluesXYparents$sx) -
qnorm(0.025, pvaluesXYparents$mx, pvaluesXYparents$sx)

pvaluesXYparents$rangY <- qnorm(0.975, pvaluesXYparents$my, pvaluesXYparents$sy) -
qnorm(0.025, pvaluesXYparents$my, pvaluesXYparents$sy)

# makes new column names “rangX”, and “rangY”, and stores the calculated length of
95% C.I. of the XY-signal intensities using the calculated mean (mx and my) and the
SD (sx and sy).

pvaluesXYparents$XALS01a <-
rejectedmarkerset$X[rejectedmarkerset$SampleName=="ALS01a"]
pvaluesXYparents$XALS01b <-
rejectedmarkerset$X[rejectedmarkerset$SampleName=="ALS01b"]
pvaluesXYparents$XALS02a <-
rejectedmarkerset$X[rejectedmarkerset$SampleName=="ALS02a"]
pvaluesXYparents$XALS02b <-
rejectedmarkerset$X[rejectedmarkerset$SampleName=="ALS02b"]

pvaluesXYparents$YALS01a <-
rejectedmarkerset$Y[rejectedmarkerset$SampleName=="ALS01a"]
pvaluesXYparents$YALS01b <-
rejectedmarkerset$Y[rejectedmarkerset$SampleName=="ALS01b"]
pvaluesXYparents$YALS02a <-
rejectedmarkerset$Y[rejectedmarkerset$SampleName=="ALS02a"]
pvaluesXYparents$YALS02b <-
rejectedmarkerset$Y[rejectedmarkerset$SampleName=="ALS02b"]
#Adds new columns “XALS01a till YALS02b” and stores X and Y signal intensities from
data.frame “rejectedmarkerset” of ALS01 and ALS02 in data.frame “pvaluesXYparents”

pvaluesXYparents$pALS01aX <- pnorm(pvaluesXYparents$XALS01a, pvaluesXYparents$mx,
pvaluesXYparents$sx)
pvaluesXYparents$pALS01bX <- pnorm(pvaluesXYparents$XALS01b, pvaluesXYparents$mx,
pvaluesXYparents$sx)
pvaluesXYparents$pALS01aY <- pnorm(pvaluesXYparents$YALS01a, pvaluesXYparents$my,
pvaluesXYparents$sy)
```

```
pvaluesXYparents$pALS01bY <- pnorm(pvaluesXYparents$YALS01b, pvaluesXYparents$my,
pvaluesXYparents$sy)


pvaluesXYparents$pALS02aX <- pnorm(pvaluesXYparents$XALS02a, pvaluesXYparents$mx,
pvaluesXYparents$sx)
pvaluesXYparents$pALS02bX <- pnorm(pvaluesXYparents$XALS02b, pvaluesXYparents$mx,
pvaluesXYparents$sx)
pvaluesXYparents$pALS02aY <- pnorm(pvaluesXYparents$YALS02a, pvaluesXYparents$my,
pvaluesXYparents$sy)
pvaluesXYparents$pALS02bY <- pnorm(pvaluesXYparents$YALS02b, pvaluesXYparents$my,
pvaluesXYparents$sy)
#calculates and stores the probability values for the parental XY-signal
intensities compared to the normal distribution of the tetraploid F1 in the new
columns “pALS01aX” till “pALS02bY” in the data.frame “pvaluesXYparents”.
```

1.6: Calculation of the probability values for the parental arcsine sqrt of the ratio compared to the 95% C.I. of the tetraploid F1

```
pvaluesRatioparents$rALS01a <-
rejectedmarkerset$asinsqr[rejectedmarkerset$SampleName=="ALS01a"]
pvaluesRatioparents$rALS01b <-
rejectedmarkerset$asinsqr[rejectedmarkerset$SampleName=="ALS01b"]
pvaluesRatioparents$rALS02a <-
rejectedmarkerset$asinsqr[rejectedmarkerset$SampleName=="ALS02a"]
pvaluesRatioparents$rALS02b <-
rejectedmarkerset$asinsqr[rejectedmarkerset$SampleName=="ALS02b"]
#Calculate arcsine square root of the ratio of ALS01 and ALS02 of marker names in
data.frame “rejectedmarkerset”, and store it in data.frame “pvaluesRatioparents”.


names(pvaluesRatioparents)[1]<- c("MarkerName")
save(pvaluesRatioparents, file = " pvaluesRatioparents.RData")

#renames column “rejectedmarkers” in data.frame “pvaluesRatioparents” into
“MarkerName” and saves the data.frame


pvaluesRatioparents$PALS01ar <- pnorm(pvaluesRatioparents$rALS01a,
pvaluesRatioparents$mr, pvaluesRatioparents$sr)
pvaluesRatioparents$PALS01br <- pnorm(pvaluesRatioparents$rALS01b,
pvaluesRatioparents$mr, pvaluesRatioparents$sr)
pvaluesRatioparents$PALS02ar <- pnorm(pvaluesRatioparents$rALS02a,
pvaluesRatioparents$mr, pvaluesRatioparents$sr)
pvaluesRatioparents$PALS02br <- pnorm(pvaluesRatioparents$rALS02b,
pvaluesRatioparents$mr, pvaluesRatioparents$sr)

#calculates and stores the probability values for the parental arcsine sqrt of the
ratio compared to the normal distribution of the tetraploid F1 in the new columns
“pALS01ar” till “pALS02br” in the data.frame “pvaluesRatioparents”.
```

1.7: Combining of (non-) combined marker names, triploid segregation pattern, SD of ratio and genotypes into one data frame.

```
tricombscores <- readDatfile("tri_combscores_noRedundant.dat")
tri <- tricombscores[,c(1, 2)]
namen <- tricombscores$MarkerName
tri$namen <- namen
#creates new data.frame called “tri” from "tri_combscores_noRedundant.dat"
containing 2 columns “markerName”, and “namen” with combined marker names
(containing P, Q, and R marker names), and one column with assigned triploid
degregation type “segtype”. The dataframe "tri_combscores_noRedundant.dat" contains
apllied triploid genotypes and segregation type, and combined markers names (P and
Q probes).

tri$MarkerName <- gsub("n", "", tri$MarkerName)
tri$MarkerName <- gsub("Qs", "Q", tri$MarkerName)
tri$MarkerName <- gsub("Ps", "P", tri$MarkerName)
tri$MarkerName <- gsub("Rs", "R", tri$MarkerName)
markersR <- as.character(tri$MarkerName)
#removes the letters “n”, and “s” from the marker names in the column “MarkerName”
from the data.frame “tri”, and stores these marker names in the vector “markersR”.

rmarkerspq <- tri[substr(markersR, nchar(markersR), nchar(markersR)) %in% c("P",
"Q"),]
```

```
rmarkersp <- tri[substr(markersR, nchar(markersR), nchar(markersR)) %in% c("R"),]
rmarkersq <- tri[substr(markersR, nchar(markersR), nchar(markersR)) %in% c("R"),]
rmarkersp$MarkerName <- gsub("R", "P", rmarkersp$MarkerName)
rmarkersq$MarkerName <- gsub("R", "Q", rmarkersq$MarkerName)
tri2 <- rbind(rmarkersp, rmarkersq, rmarkerspq)
#creates 3 different data.frames "rmarkerspq", "rmarkersp", and "rmarkersq", from
the data.frame "tri" (two contain marker names with "R", and one with "P" and "Q").
In the 2 data.frames containing "R" at the end of the marker name, the "R" is
converted into the letter "P" ("rmarkersp"), and "Q" ("rmarkersq").The 3
data.frames are binded together into the data.frame "tri2".

mrknamesegtype <- pvaluesRatioparents[,c(1, 2)]
#the data.frame "mrknamesegtype" is the downsized data.frame "pvaluesRatioparents",
only containing the 15999 marker names, and the SD of the arcsine sqrt of the
ratio.
parentscoring<- merge(tri2, mrknamesegtype, by = "MarkerName", all.y = T)
rejected <- parentscoring$MarkerName
parentscoring <- data.frame(parentscoring, genoALS01XY=rep(0, length(rejected)),
genoALS02XY=rep(0, length(rejected)), genoALS01seg=rep(0, length(rejected)),
genoALS02seg=rep(0, length(rejected)), genoALS01name=rep(0, length(rejected)),
genoALS02name=rep(0, length(rejected)), ALS01=rep(0, length(rejected)),
ALS02=rep(0, length(rejected)), "conflicting methods"=rep(0, length(rejected)))
#creates data.frame called "parentscoring" for storage of the triploid segregation,
SD of the ratio, parental genotypes obtained from the 3 methods, the final
genotypes, and information about conflicts between methods(columns: genoALS01XY,
genoALS02XY, genoALS01seg, genoALS02seg, genoALS01name, genoALS02name, ALS01,
ALS02, "conflicting methods), by merging the data.frames "tri2" and
"mrknamesegtype" by MarkerName, for the situation where the markername is present
in the dataframe "mrknamesegtype".
```

1.8: Identifying marker names where the parents are/are not significantly different/larger/smaller than the 95% C.I. of the tetraploid F1.

```
mrkALS01samenulli <-
unique(as.character(pvaluesXYparents$MarkerName[(pvaluesXYparents$rangX >
3*pvaluesXYparents$rangY & pvaluesXYparents$pALS01aY < 0.95) &
pvaluesXYparents$pALS01bY < 0.95]))

mrkALS02samenulli <-
unique(as.character(pvaluesXYparents$MarkerName[(pvaluesXYparents$rangX >
3*pvaluesXYparents$rangY & pvaluesXYparents$pALS02aY < 0.95) &
pvaluesXYparents$pALS02bY < 0.95]))

mrkALS01samequad <-
unique(as.character(pvaluesXYparents$MarkerName[(pvaluesXYparents$rangY >
3*pvaluesXYparents$rangX & pvaluesXYparents$pALS01aX < 0.95) &
pvaluesXYparents$pALS01bX < 0.95]))

mrkALS02samequad <-
unique(as.character(pvaluesXYparents$MarkerName[(pvaluesXYparents$rangY >
3*pvaluesXYparents$rangX & pvaluesXYparents$pALS02aX < 0.95) &
pvaluesXYparents$pALS02bX < 0.95]))
#produces vectors ("mrkALS01samenulli" till "mrkALS02samequad") containing marker
names where both parental samples (a and b) are not significantly different from
homozygous tetraploid cluster difined with XY signal intensities, and having at
least a 3 times difference between the X and Y length.

ALS01sameclusterratio <-
unique(as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$PALS01ar >
0.025 & pvaluesRatioparents$PALS01br > 0.025 & pvaluesRatioparents$PALS01ar < 0.975
& pvaluesRatioparents$PALS01br < 0.975]))

ALS02sameclusterratio <-
unique(as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$PALS02ar >
0.025 & pvaluesRatioparents$PALS02br > 0.025 & pvaluesRatioparents$PALS02ar < 0.975
& pvaluesRatioparents$PALS02br < 0.975])))

ALS01difclusterratio<-
unique(as.character(pvaluesRatioparents$MarkerName(![pvaluesRatioparents$PALS01ar >
0.025 & pvaluesRatioparents$PALS01br > 0.025 & pvaluesRatioparents$PALS01ar < 0.975
& pvaluesRatioparents$PALS01br < 0.975])))

ALS02difclusterratio <-
unique(as.character(pvaluesRatioparents$MarkerName(![pvaluesRatioparents$PALS02ar >
```

```
0.025 & pvaluesRatioparents$PALS02br > 0.025 & pvaluesRatioparents$PALS02ar < 0.975
& pvaluesRatioparents$PALS02br < 0.975]))
#creates vectors called "ALS01sameclusterratio" till "ALS02difclusterratio" with
marker names from data.frame "pvaluesRatioparent" where both samples of ALS01 or
ALS02 are (not) significantly different from the 95% C.I. of the arcsine sqrt of
the ratio of the tetraploid cluster

bothALSinclusterratio <-
unique(as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$MarkerName
%in% ALS01sameclusterratio & pvaluesRatioparents$MarkerName %in%
ALS02sameclusterratio]))
#creates vectors called "bothALSinclusterratio" with marker names from data.frame
"pvaluesRatioparent" where both samples of ALS01 and ALS02 are not significantly
different from the 95% C.I. of the arcsine sqrt of the ratio of the tetraploid
cluster.

smallerALS01markers <-
as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$MarkerName %in%
ALS02sameclusterratio & pvaluesRatioparents$MarkerName %in%
ALS01difclusterratio & pvaluesRatioparents$mr > pvaluesRatioparents$rALS01a &
pvaluesRatioparents$mr > pvaluesRatioparents$rALS01b])

largerALS01markers <-
as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$MarkerName %in%
ALS02sameclusterratio & pvaluesRatioparents$MarkerName %in%
ALS01difclusterratio & pvaluesRatioparents$mr < pvaluesRatioparents$rALS01a &
pvaluesRatioparents$mr < pvaluesRatioparents$rALS01b])

largerALS02markers <-
as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$MarkerName %in%
ALS01sameclusterratio & pvaluesRatioparents$MarkerName %in%
ALS02difclusterratio & pvaluesRatioparents$mr < pvaluesRatioparents$rALS02a &
pvaluesRatioparents$mr < pvaluesRatioparents$rALS02b])

smallerALS02markers <-
as.character(pvaluesRatioparents$MarkerName[pvaluesRatioparents$MarkerName %in%
ALS01sameclusterratio & pvaluesRatioparents$MarkerName %in%
ALS02difclusterratio & pvaluesRatioparents$mr > pvaluesRatioparents$rALS02a &
pvaluesRatioparents$mr > pvaluesRatioparents$rALS02b])
#creates vectors called "smallerALS01markers" till "largerALS02markers" with marker
names from data.frame "pvaluesRatioparent" where ALS01 or ALS02 is not
significantly different and the other parents is significantly smaller/larger from
the 95% C.I. of the arcsine sqrt of the ratio of the tetraploid cluster, using
vectors ALS01sameclusterratio" till "ALS02difclusterratio".
```

1.9: Genotyping

1.9.1: Genotyping of marker having homozygous tetraploid cluster using 95% C.I. of XY signal intensity lengths

```
parentscoring$genoALS01XY[parentscoring$MarkerName %in% mrkALS01samenulli]<- 0
parentscoring$genoALS02XY[parentscoring$MarkerName %in% mrkALS02samenulli]<- 0
parentscoring$genoALS01XY[parentscoring$MarkerName %in% mrkALS01samequad]<- 2
parentscoring$genoALS02XY[parentscoring$MarkerName %in% mrkALS02samequad]<- 4
#genotyping of ALS01 and ALS02 with homozygous genotypes in the columns
"genoALS01XY" and "genoALS02XY" of the data.frame "parentscoring", when the marker
name is present in the either of the vectors ("mrkALS01samenulli" till
"mrkALS02samequad").

parentscoring$genoALS01XY[genoALS02XY=="0" & parentscoring$segtype=="s1000" &!
parentscoring$MarkerName %in% mrkALS01samenulli]<- 0
parentscoring$genoALS02XY[genoALS02XY=="0" & parentscoring$segtype=="s0100" &!
parentscoring$MarkerName %in% mrkALS01samenulli]<-2

parentscoring$genoALS02XY[genoALS01XY=="0" & parentscoring$segtype=="s1000" &!
parentscoring$MarkerName %in% mrkALS02samenulli]<-0
parentscoring$genoALS02XY[genoALS01XY=="0" & parentscoring$segtype=="s0100" &!
parentscoring$MarkerName %in% mrkALS02samenulli]<-2
parentscoring$genoALS02XY[genoALS01XY=="0" & parentscoring$segtype=="s0010" &!
parentscoring$MarkerName %in% mrkALS02samenulli]<-4

parentscoring$genoALS01XY[genoALS02XY=="0" & parentscoring$segtype=="s1100" &!
parentscoring$MarkerName %in% mrkALS01samenulli]<- 1

parentscoring$genoALS02XY[genoALS01XY=="0" & parentscoring$segtype=="s1100" &!
parentscoring$MarkerName %in% mrkALS02samenulli]<- 1
```

```
parentscoring$genoALS02XY[genoALS01XY=="0" & parentscoring$segtype=="s0110" &!
parentscoring$MarkerName %in% mrkALS02samenulli]<- 3


parentscoring$genoALS01XY[genoALS02XY=="4" & parentscoring$segtype=="s0001" &!
parentscoring$MarkerName %in% mrkALS01samequad]<-2
parentscoring$genoALS01XY[genoALS02XY=="4" & parentscoring$segtype=="s0010" &!
parentscoring$MarkerName %in% mrkALS01samequad]<-0

parentscoring$genoALS02XY[genoALS01XY=="2" & parentscoring$segtype=="s0001" &!
parentscoring$MarkerName %in% mrkALS02samequad]<-4
parentscoring$genoALS02XY[genoALS01XY=="2" & parentscoring$segtype=="s0010" &!
parentscoring$MarkerName %in% mrkALS02samequad]<-2
parentscoring$genoALS02XY[genoALS01XY=="2" & parentscoring$segtype=="s0100" &!
parentscoring$MarkerName %in% mrkALS02samequad]<-0

parentscoring$genoALS01XY[genoALS02XY=="4" & parentscoring$segtype=="s0011" &!
parentscoring$MarkerName %in% mrkALS01samequad]<-1

parentscoring$genoALS02XY[genoALS01XY=="2" & parentscoring$segtype=="s0011" &!
parentscoring$MarkerName %in% mrkALS02samequad]<-3
parentscoring$genoALS02XY[genoALS01XY=="2" & parentscoring$segtype=="s0110" &!
parentscoring$MarkerName %in% mrkALS02samequad]<-1
#genotyping of significantly different ALS01 or ALS02 by using triploid segregation
pattern in combination with genotype of not significantly different parent from the
columns "genoALS01XY" and "genoALS02XY" of the data.frame "parentscoring", when the
marker name is present in the either of the vectors ("mrkALS01samenulli" till
"mrkALS02samequad").
```

1.9.2: Genotyping of parental samples using the heterozygous information in the marker name, and the arcsine sqrt of the ratio

```
markers <- parentscoring$MarkerName
hetrALS01 <- as.character(parentscoring$MarkerName[substr(markers, nchar(markers)-
2, nchar(markers)) %in% c("01P", "01Q", "12P", "12Q") & parentscoring$SDratio <
0.06])
#makes vector "hetrALS01" with marker names that end with 01P, 01Q, 12P, 12Q
(meaning ALS01=heterozygous) in the column "markername" from the data.frame
"parentscoring", and having a SD <0.06 in the column name "SDratio".
parentscoring$genoALS01name[parentscoring$MarkerName %in% hetrALS01]<- 1
#genotypes ALS01 with hetreoygous genotype of markers present in vector
"hetrALS01".
parentscoring$genoALS02name[parentscoring$MarkerName %in% hetrALS01 &
parentscoring$MarkerName %in% bothALSinclusterratio]<- 2
#genotypes ALS02 with disomic duplex genotype in column "genoALS02name" in
data.frame "parentscoring", when marker name is present in vector "hetrALS01" and
in "bothALSinclusterratio".

hetrALS02 <- as.character(parentscoring$MarkerName[substr(markers, nchar(markers)-
2, nchar(markers)) %in% c("02P", "02Q", "23P", "23Q") & parentscoring$SDratio <
0.06])
#creates vector with marker names from data.frame "parentscoring" column
"MarkerName" that have "02P", or "02Q" in the last part of the marker name, and
having an SDratio < 0.06.
parentscoring$genoALS01name[parentscoring$MarkerName %in% hetrALS02 &
parentscoring$MarkerName %in% bothALSinclusterratio]<- 1
parentscoring$genoALS02name[parentscoring$MarkerName %in% hetrALS02 &
parentscoring$MarkerName %in% bothALSinclusterratio]<- 2
#genotypes ALS01 and ALS02 in data.frame "parentscoring" columns "genoALS01name",
and "genoALS01name"when marker name is present in vector "hetrALS02" and
"bothALSinclusterratio".
```

1.9.3: Genotyping of parental samples using the triploid segregation information, and the arcsine sqrt of the ratio

```
#s1000
parentscoring$genoALS01seg[parentscoring$segtype=="s1000" & parentscoring$SDratio <
0.06]<- 0
parentscoring$genoALS02seg[parentscoring$segtype=="s1000" & parentscoring$SDratio <
0.06]<- 0

#S0001
parentscoring$genoALS01seg[parentscoring$segtype=="s0001" & parentscoring$SDratio <
0.06]<- 2
parentscoring$genoALS02seg[parentscoring$segtype=="s0001" & parentscoring$SDratio <
0.06]<- 4
```

```
#s0100
parentscoring$genoALS01seg[parentscoring$segtype=="s0100" &
parentscoring$MarkerName %in% smallerALS01markers & parentscoring$SDratio<0.06]<- 0
parentscoring$genoALS02seg[parentscoring$segtype=="s0100" &
parentscoring$MarkerName %in% smallerALS01markers & parentscoring$SDratio<0.06]<- 2

#s0010
parentscoring$genoALS01seg[parentscoring$segtype=="s0010" &
parentscoring$MarkerName %in% largerALS01markers & parentscoring$SDratio<0.06]<- 2
parentscoring$genoALS02seg[parentscoring$segtype=="s0010" &
parentscoring$MarkerName %in% largerALS01markers & parentscoring$SDratio<0.06]<- 2

#s1100
parentscoring$genoALS01seg[parentscoring$segtype=="s1100" &
parentscoring$MarkerName %in% largerALS01markers & parentscoring$SDratio<0.06]<- 1
parentscoring$genoALS02seg[parentscoring$segtype=="s1100" &
parentscoring$MarkerName %in% largerALS01markers & parentscoring$SDratio<0.06]<- 0

parentscoring$genoALS01seg[parentscoring$segtype=="s1100" &
parentscoring$MarkerName %in% smallerALS02markers & parentscoring$SDratio<0.06]<- 1
parentscoring$genoALS02seg[parentscoring$segtype=="s1100" &
parentscoring$MarkerName %in% smallerALS02markers & parentscoring$SDratio<0.06]<- 0

#s0011
parentscoring$genoALS01seg[parentscoring$segtype=="s0011" &
parentscoring$MarkerName %in% smallerALS01markers & parentscoring$SDratio<0.06]<- 1
parentscoring$genoALS02seg[parentscoring$segtype=="s0011" &
parentscoring$MarkerName %in% smallerALS01markers & parentscoring$SDratio<0.06]<- 4

parentscoring$genoALS01seg[parentscoring$segtype=="s0011" &
parentscoring$MarkerName %in% largerALS02markers & parentscoring$SDratio <0.06]<- 1
parentscoring$genoALS02seg[parentscoring$segtype=="s0011" &
parentscoring$MarkerName %in% largerALS02markers & parentscoring$SDratio <0.06]<- 4

#s0110
parentscoring$genoALS01seg[parentscoring$segtype=="s0110" &
parentscoring$MarkerName %in% bothALSinclusterratio & parentscoring$SDratio <
0.06]<-1
parentscoring$genoALS02seg[parentscoring$segtype=="s0110" &
parentscoring$MarkerName %in% bothALSinclusterratio & parentscoring$SDratio <
0.06]<-2

parentscoring$genoALS01seg[parentscoring$segtype=="s0110" &
parentscoring$MarkerName %in% largerALS01markers]<-2
parentscoring$genoALS02seg[parentscoring$segtype=="s0110" &
parentscoring$MarkerName %in% largerALS01markers]<-1

parentscoring$genoALS01seg[parentscoring$segtype=="s0110" &
parentscoring$MarkerName %in% smallerALS02markers]<-2
parentscoring$genoALS02seg[parentscoring$segtype=="s0110" &
parentscoring$MarkerName %in% smallerALS02markers]<-1

#1210
parentscoring$genoALS01seg[parentscoring$segtype=="s1210" &
parentscoring$MarkerName %in% largerALS02markers]<-0
parentscoring$genoALS02seg[parentscoring$segtype=="s1210" &
parentscoring$MarkerName %in% largerALS02markers]<-2

parentscoring$genoALS01seg[parentscoring$segtype=="s1210" &
parentscoring$MarkerName %in% smallerALS01markers]<-0
parentscoring$genoALS02seg[parentscoring$segtype=="s1210" &
parentscoring$MarkerName %in% smallerALS01markers]<-2

#0121
parentscoring$genoALS01seg[parentscoring$segtype=="s0121" &
parentscoring$MarkerName %in% largerALS01markers]<-2
parentscoring$genoALS02seg[parentscoring$segtype=="s0121" &
parentscoring$MarkerName %in% largerALS01markers]<-2

parentscoring$genoALS01seg[parentscoring$segtype=="s0121" &
parentscoring$MarkerName %in% smallerALS02markers]<-2
parentscoring$genoALS02seg[parentscoring$segtype=="s0121" &
parentscoring$MarkerName %in% smallerALS02markers]<-2

#1331
```

```
parentscoring$genoALS01seg[parentscoring$segtype=="s1331" &
parentscoring$MarkerName %in% bothALSinclusterratio]<-1
parentscoring$genoALS02seg[parentscoring$segtype=="s1331" &
parentscoring$MarkerName %in% bothALSinclusterratio]<-2

#s1551
parentscoring$genoALS01seg[parentscoring$segtype=="s1551" &
parentscoring$MarkerName %in% bothALSinclusterratio]<-1
parentscoring$genoALS02seg[parentscoring$segtype=="s1551" &
parentscoring$MarkerName %in% bothALSinclusterratio]<-2

#s1410
parentscoring$genoALS01seg[parentscoring$segtype=="s1410" &
parentscoring$MarkerName %in% largerALS02markers] <- 0
parentscoring$genoALS02seg[parentscoring$segtype=="s1410" &
parentscoring$MarkerName %in% largerALS02markers] <- 2

parentscoring$genoALS01seg[parentscoring$segtype=="s1410" &
parentscoring$MarkerName %in% smallerALS01markers] <- 0
parentscoring$genoALS02seg[parentscoring$segtype=="s1410" &
parentscoring$MarkerName %in% smallerALS01markers] <- 2

parentscoring$genoALS01seg[parentscoring$segtype=="s0141" &
parentscoring$MarkerName %in% largerALS01markers] <- 2
parentscoring$genoALS02seg[parentscoring$segtype=="s0141" &
parentscoring$MarkerName %in% largerALS01markers] <- 2

parentscoring$genoALS01seg[parentscoring$segtype=="s0141" &
parentscoring$MarkerName %in% smallerALS02markers] <- 2
parentscoring$genoALS02seg[parentscoring$segtype=="s0141" &
parentscoring$MarkerName %in% smallerALS02markers] <- 2

#genotyping of ALS01 and ALS02 in columns “genoALS01seg”, and “genoALS02seg” of the
data.frame “parentscoring”, using the triploid segregation pattern in the column
“segtype”, and the marker names in the vectors “smallerALS01markers”,
“smallerALS02markers”, “largerALS01markers”, “largerALS02markers”, and
“bothALSinclusterratio”.
```

1.10: Assignment of final parental genotypes, and conflicting information, followed by combining of triploid genotypes and segregation type with assigned parental genotypes.

```
parentscoring$ALS01[parentscoring$genoALS01XY==0 ]<-0
parentscoring$ALS01[parentscoring$genoALS01XY==2 ]<-2
parentscoring$ALS01[!parentscoring$genoALS01XY %in% c(0, 2) &
parentscoring$genoALS01seg==0]<-0
parentscoring$ALS01[!parentscoring$genoALS01XY %in% c(0, 2) &
parentscoring$genoALS01seg==parentscoring$genoALS01name]<-1
parentscoring$ALS01[!parentscoring$genoALS01XY %in% c(0, 2) &
parentscoring$genoALS01seg!=parentscoring$genoALS01name]<-6
parentscoring$ALS01[!parentscoring$genoALS01XY %in% c(0, 2) &!
parentscoring$genoALS01name %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS01seg==1]<-1
parentscoring$ALS01[!parentscoring$genoALS01XY %in% c(0, 2) &!
parentscoring$genoALS01seg %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS01name==1]<-1
parentscoring$ALS01[!parentscoring$genoALS01XY %in% c(0, 2) &!
parentscoring$genoALS01name %in% c(0, 1, 2, 3, 4)& parentscoring$genoALS01seg==2]<-
2

parentscoring$ALS02[parentscoring$genoALS02XY==0 ]<-0
parentscoring$ALS02[parentscoring$genoALS02XY==4 ]<-4
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &
parentscoring$genoALS02seg!=parentscoring$genoALS02name]<-6
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02seg %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02name==1]<- 1
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02seg %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02name==2]<- 2
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02seg %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02name==3]<- 3
```

```
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02name %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02seg==0]<- 0
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02name %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02seg==1]<- 1
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02name %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02seg==2]<- 2
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02name %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02seg==3]<- 3
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &!
parentscoring$genoALS02name %in% c(0, 1, 2, 3, 4) &
parentscoring$genoALS02seg==4]<- 4

parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &
parentscoring$genoALS02seg==parentscoring$genoALS02name &
parentscoring$genoALS01seg==0]<- 0
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &
parentscoring$genoALS02seg==parentscoring$genoALS02name &
parentscoring$genoALS01seg==1]<- 1
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &
parentscoring$genoALS02seg==parentscoring$genoALS02name &
parentscoring$genoALS01seg==2]<- 2
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &
parentscoring$genoALS02seg==parentscoring$genoALS02name &
parentscoring$genoALS01seg==3]<- 3
parentscoring$ALS02[!parentscoring$genoALS02XY %in% c(0, 4) &
parentscoring$genoALS02seg==parentscoring$genoALS02name &
parentscoring$genoALS01seg==4]<- 4
#assignment of final genotype in columns “ALS01” and “ALS02”, by comparing assigned
genotypes between methods. Genotype in columns “genoALS01XY”, and “genoALS02XY” has
priority over other methods. Conflicting genotypes between columns
“genoALS01/2name” and “genoALS01/2” results in assigned Genotype “6” in columns
“ALS01” and/or “ALS02”.

parentscoring$`conflicting-
methods`[parentscoring$genoALS01name!=parentscoring$genoALS01XY &
parentscoring$genoALS02name!=parentscoring$genoALS02XY]<- "a12"
parentscoring$`conflicting-
methods`[parentscoring$genoALS01seg!=parentscoring$genoALS01XY &
parentscoring$genoALS02seg!=parentscoring$genoALS02XY]<- "b12"
parentscoring$`conflicting-
methods`[parentscoring$genoALS01name!=parentscoring$genoALS01XY &
parentscoring$genoALS02name!=parentscoring$genoALS02XY &
parentscoring$genoALS01seg!=parentscoring$genoALS01XY &
parentscoring$genoALS02seg!=parentscoring$genoALS02XY]<- "ab12"

parentscoring$`conflicting-
methods`[parentscoring$genoALS01name!=parentscoring$genoALS01XY &
parentscoring$genoALS02name==parentscoring$genoALS02XY]<- "a1"
parentscoring$`conflicting-
methods`[parentscoring$genoALS01seg!=parentscoring$genoALS01XY &
parentscoring$genoALS02seg==parentscoring$genoALS02XY]<- "b1"
parentscoring$`conflicting-
methods`[parentscoring$genoALS01seg!=parentscoring$genoALS01XY &
parentscoring$genoALS02seg==parentscoring$genoALS02XY &
parentscoring$genoALS01name!=parentscoring$genoALS01XY &
parentscoring$genoALS02name==parentscoring$genoALS02XY]<- "ab1"

parentscoring$`conflicting-
methods`[parentscoring$genoALS01name==parentscoring$genoALS01XY &
parentscoring$genoALS02name!=parentscoring$genoALS02XY]<- "a2"
parentscoring$`conflicting-
methods`[parentscoring$genoALS01seg==parentscoring$genoALS01XY &
parentscoring$genoALS02seg!=parentscoring$genoALS02XY]<- "b2"
parentscoring$`conflicting-
methods`[parentscoring$genoALS01seg==parentscoring$genoALS01XY &
parentscoring$genoALS02seg!=parentscoring$genoALS02XY &
parentscoring$genoALS01name==parentscoring$genoALS01XY &
parentscoring$genoALS02name!=parentscoring$genoALS02XY]<- "ab2"

parentscoring$`conflicting-methods`[parentscoring$ALS01==6 &!
parentscoring$ALS02==6]<- "c1"
parentscoring$`conflicting-methods`[parentscoring$ALS02==6 &!
parentscoring$ALS01==6]<- "c2"
```

```
parentscoring$`conflicting-methods`[parentscoring$ALS01==6 &
parentscoring$ALS02==6]<- "c12"

parentscoring$`conflicting-methods`[parentscoring$ALS01==6 &!
parentscoring$ALS02==6]<- "c1"
parentscoring$`conflicting-methods`[parentscoring$ALS02==6 &!
parentscoring$ALS01==6]<- "c2"
parentscoring$`conflicting-methods`[parentscoring$ALS01==6 &
parentscoring$ALS02==6]<- "c12"

parentscoring$`conflicting-
methods`[parentscoring$genoALS01seg!=parentscoring$genoALS01XY &
parentscoring$genoALS01XY!=parentscoring$genoALS01name &
parentscoring$genoALS01name!=parentscoring$genoALS01seg
& parentscoring$genoALS02seg!=parentscoring$genoALS02XY &
parentscoring$genoALS02XY!=parentscoring$genoALS02name &
parentscoring$genoALS02name!=parentscoring$genoALS02seg]<- "abc12"
#assigns conflicting information ("a", "b", "c" for ALS01=1 and ALS02=2) in column
"conflicting-methods" to indicate whether the assigned genotypes between the
methods are in conflict with each other.

tricombscores <- readDatfile("tri_combscores_noRedundant.dat")
names(tricombscores)[1] <- c("namen")
Tricombscores <- merge(parentscoring, tricombscores, by = "namen)
writeDatfile(tricombscores ,file = "tricombscores.DAT")

#combines data.frames "parentscoring" and "tri_combscores_noRedundant.dat" by
marker names.
```

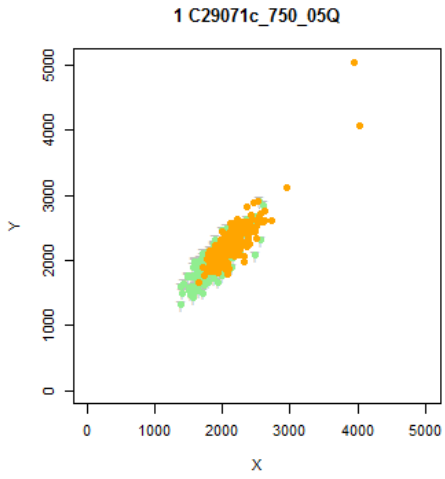
1.11: Correcting parental genotypes of shifted triploid F1s showing an s0110 segregation, using the relation between the arcsine sqrt of the ratio of ALS01 and the 95% C.I. of the arcsine sqrt of the ratio of the TC F1.

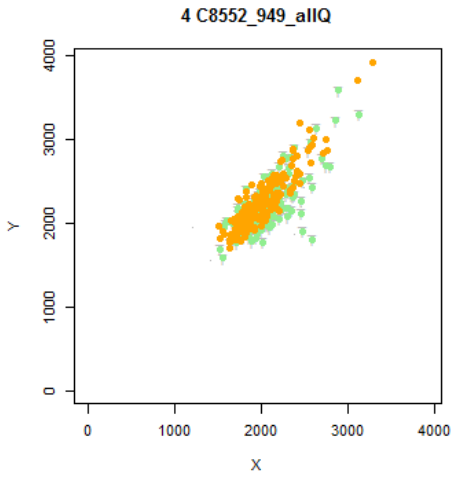
```
s0110TC01 <- c(…)
#vector "s0110TC01" containing marker names from "tricombscores" having segtype
"s0110", and "01" heterozygous information at the end of the marker name.
s0110TC01 <- comb4x[comb4x$MarkerName %in% s0110markers01 & comb4x$SampleName %in%
c(pop$TC$F1, pop$TC$P1, pop$TC$P2, pop$tri$P1),]

markers01 <- unique(as.character(s0110TC01$MarkerName))

s0110TC01$asinsqr <- asin(sqrt(s0110TC01$ratio))

gemratios0110 = data.frame(markers01, mr=rep(0, length(markers01)),
sr=rep(0,length(markers01)))

for(mrk in 1:length(markers01)) {

 al <- s0110TC01$asinsqr[s0110TC01$MarkerName == markers01[mrk] &
 s0110TC01$SampleName %in% c(pop$TC$F1)]

 mr <- mean(al)
 sr <- sd(al)

 gemratios0110 [mrk,2] <- mr
 gemratios0110 [mrk,3] <- sr
}

gemratios0110$rALS01a <- s0110TC01$asinsqr[s0110TC01$SampleName=="ALS01a"]
gemratios0110$rALS01b <- s0110TC01$asinsqr[s0110TC01$SampleName=="ALS01b"]
gemratios0110$rALS02a <- s0110TC01$asinsqr[s0110TC01$SampleName=="ALS02a"]
gemratios0110$rALS02b <- s0110TC01$asinsqr[s0110TC01$SampleName=="ALS02b"]
gemratios0110$rALS03a <- s0110TC01$asinsqr[s0110TC01$SampleName=="ALS03a"]
gemratios0110$rALS03b <- s0110TC01$asinsqr[s0110TC01$SampleName=="ALS03b"]
gemratios0110$rALS03c <- s0110TC01$asinsqr[s0110TC01$SampleName=="ALS03c"]

gemratios0110$pALS01a <- pnorm(gemratios0110$rALS01a, gemratios0110$mr,
gemratios0110$sr)
gemratios0110$pALS01b <- pnorm(gemratios0110$rALS01b, gemratios0110$mr,
gemratios0110$sr)
gemratios0110$pALS02a <- pnorm(gemratios0110$rALS02a, gemratios0110$mr,
gemratios0110$sr)
gemratios0110$pALS02b <- pnorm(gemratios0110$rALS02b, gemratios0110$mr,
gemratios0110$sr)
```

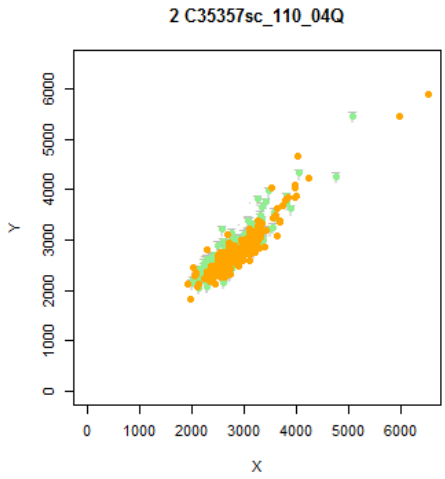
```
gemratios0110$pALS03a <- pnorm(gemratios0110$rALS03a, gemratios0110$mr,
gemratios0110$sr)
gemratios0110$pALS03b <- pnorm(gemratios0110$rALS03b, gemratios0110$mr,
gemratios0110$sr)
gemratios0110$pALS03c <- pnorm(gemratios0110$rALS03c, gemratios0110$mr,
gemratios0110$sr)

als01smallercluster <-
unique(as.character(gemratios0110$markers01[gemratios0110$pALS02a > 0.025 &
gemratios0110$pALS02b > 0.025 & gemratios0110$pALS03a > 0.025 &
gemratios0110$pALS03b > 0.025 & gemratios0110$pALS03c > 0.025 &
 gemratios0110$pALS02a < 0.975 & gemratios0110$pALS02b < 0.975 &
gemratios0110$pALS03a < 0.975 & gemratios0110$pALS03b < 0.975 &
gemratios0110$pALS03c < 0.975]))
alcluster <- unique(as.character(gemratios0110$markers01[!(gemratios0110$pALS02a >
0.025 & gemratios0110$pALS02b > 0.025 & gemratios0110$pALS03a > 0.025 &
gemratios0110$pALS03b > 0.025 & gemratios0110$pALS03c > 0.025 &
 gemratios0110$pALS02a < 0.975 & gemratios0110$pALS02b < 0.975 &
gemratios0110$pALS03a < 0.975 | gemratios0110$pALS03b < 0.975 |
gemratios0110$pALS03c < 0.975)]))

als01largercluster <-
unique(as.character(gemratios0110$markers01[gemratios0110$pALS01a > 0.99 &
gemratios0110$pALS01b > 0.99]))
als01smallercluster <-
unique(as.character(gemratios0110$markers01[gemratios0110$pALS01a < 0.01 &
gemratios0110$pALS01b < 0.01]))
als01sameclustera <-
unique(as.character(gemratios0110$markers01[gemratios0110$pALS01a > 0.01 |
gemratios0110$pALS01b > 0.01]))
als01sameclusterb <-
unique(as.character(gemratios0110$markers01[gemratios0110$pALS01a < 0.99 |
gemratios0110$pALS01b < 0.99]))
als01samecluster <-
unique(as.character(gemratios0110$markers01[gemratios0110$markers01 %in%
als01sameclustera & gemratios0110$markers01 %in% als01sameclusterb]))

tab = data.frame(markers01, segregation=rep(0, length(markers01)), ALS01=rep(0,
length(markers01)), ALS02=rep(0, length(markers01)))
tab$segregation[tab$markers01 %in% als01samecluster] <- "s0110"
tab$segregation[tab$markers01 %in% als01largercluster] <- "s1100"
tab$segregation[tab$markers01 %in% als01smallercluster] <- "s0011"
tab$ALS01[tab$markers01 %in% als01samecluster] <- 1
tab$ALS02[tab$markers01 %in% als01samecluster] <- 2
tab$ALS01[tab$markers01 %in% als01largercluster] <- 1
tab$ALS02[tab$markers01 %in% als01largercluster] <- 0
tab$ALS01[tab$markers01 %in% als01smallercluster] <- 1
tab$ALS02[tab$markers01 %in% als01smallercluster] <- 4

correctedsegregations01<- tab
save(correctedsegregations01, file = "correctedseg01.RData")

kkk <- merge(combmarkers, correctedsegregations01 , by= "MarkerName", all.y = T)
vvv <- merge(kkk, tricombscores, by= "Markername", all.x = T)
names(kkk)[2]<- c("Markername")
save(vvv, file = "corrected0110seg01markers.RData")
writeDatfile(vvv, file = "corrected0110seg01markers.Dat")
```

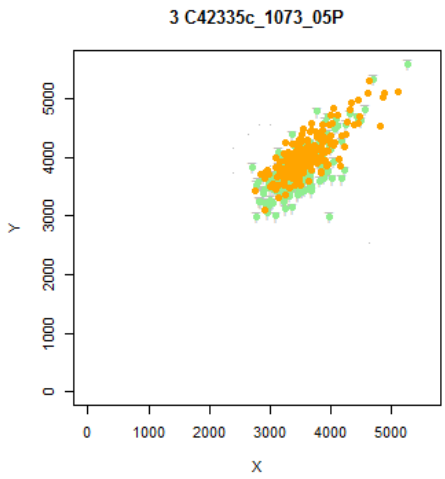
Appendix 2: grouping of markers based on SD of the arcsine square root of the ratio of tetraploid F1.

1: SD<0.02

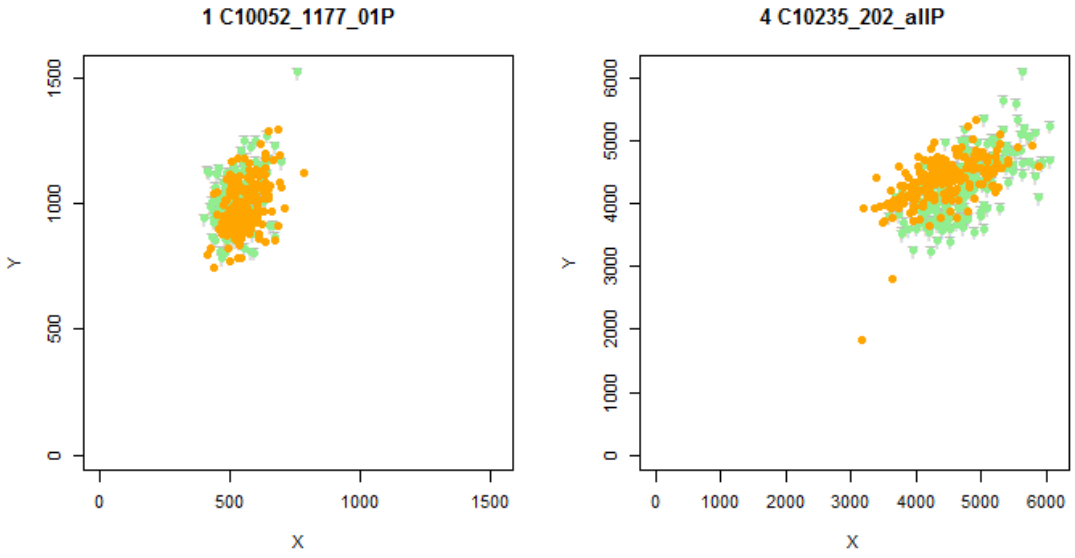




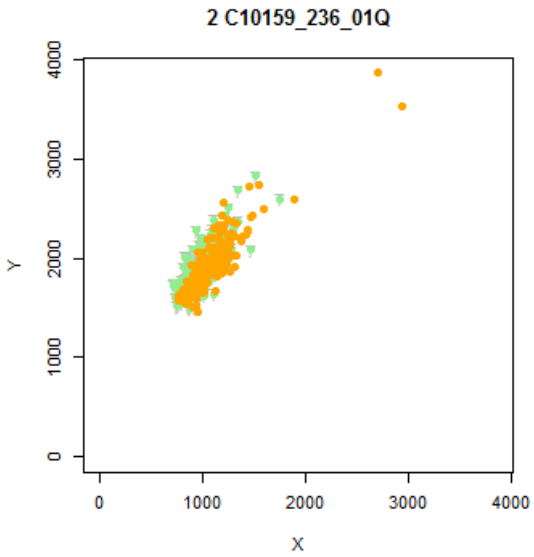


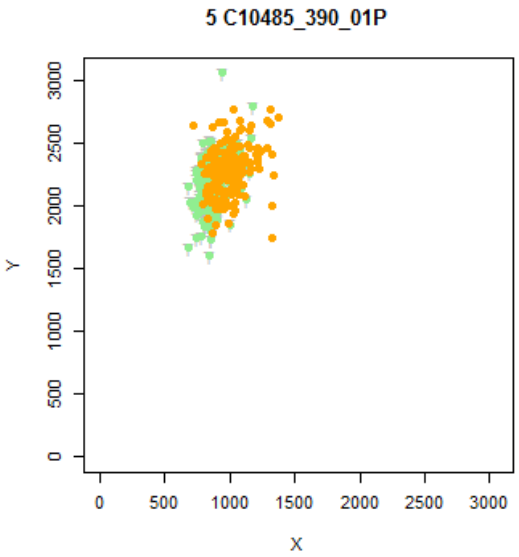


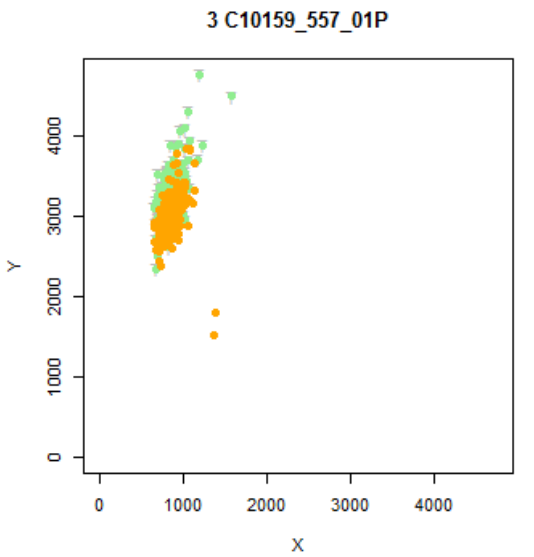
2: 0.02<SD<0.03

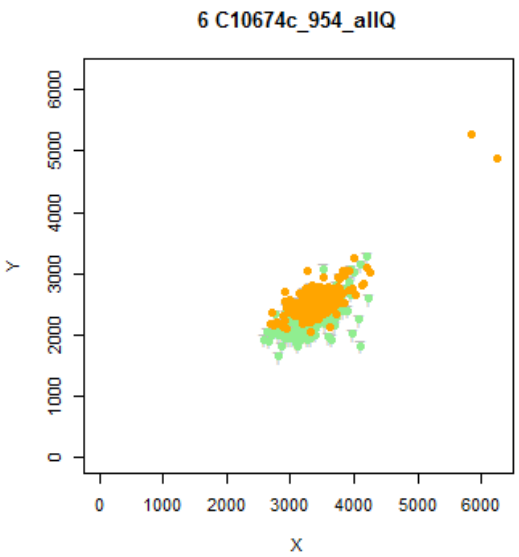

1 C10052_1177_01P
Y
X
0
500
1000
1500

4 C10235_202_allP
Y
X
0
1000
2000
3000
4000
5000
6000

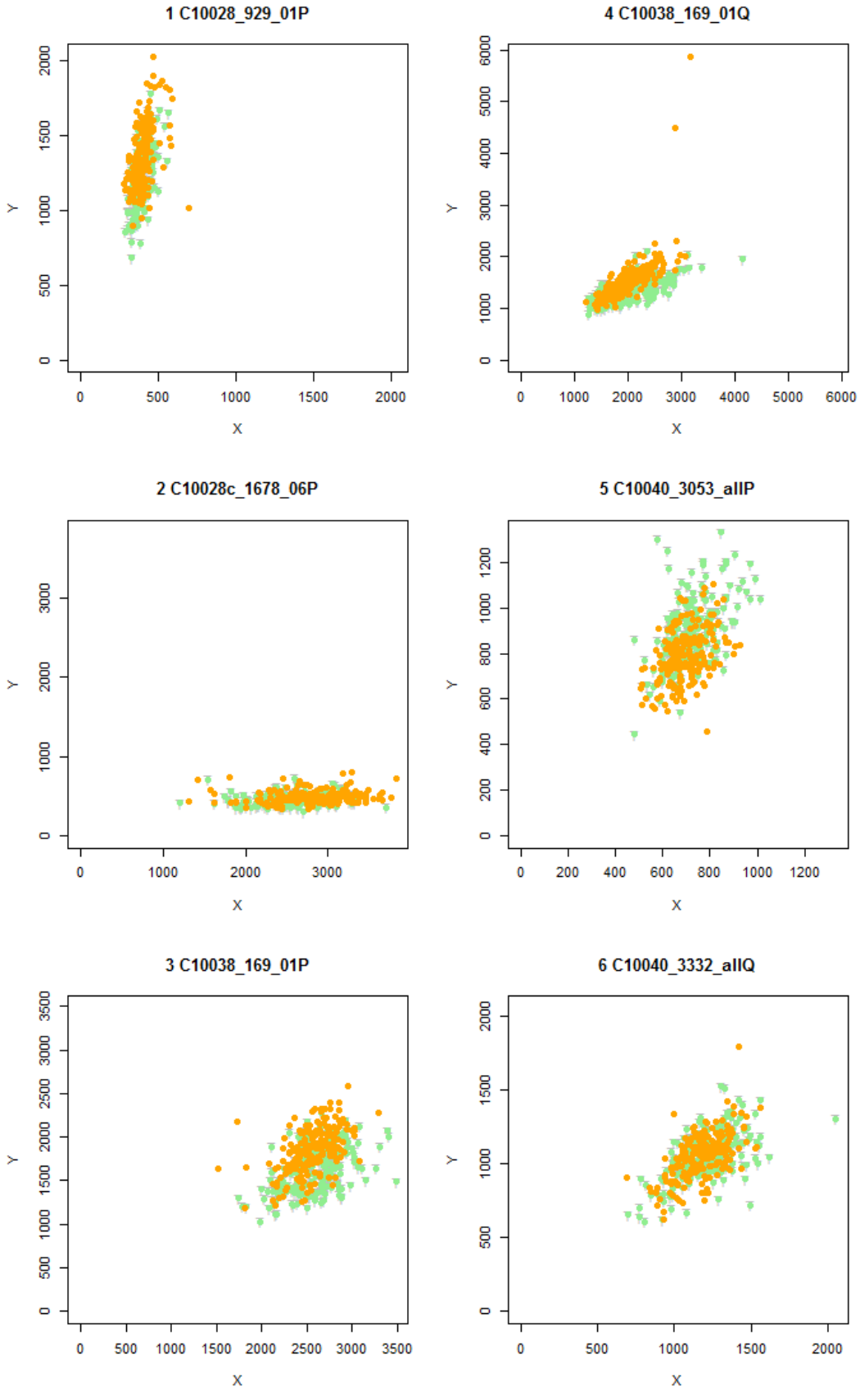

2 C10159_236_01Q
Y
X
0
1000
2000
3000
4000


5 C10485_390_01P
Y
X
0
500
1000
1500
2000
2500
3000

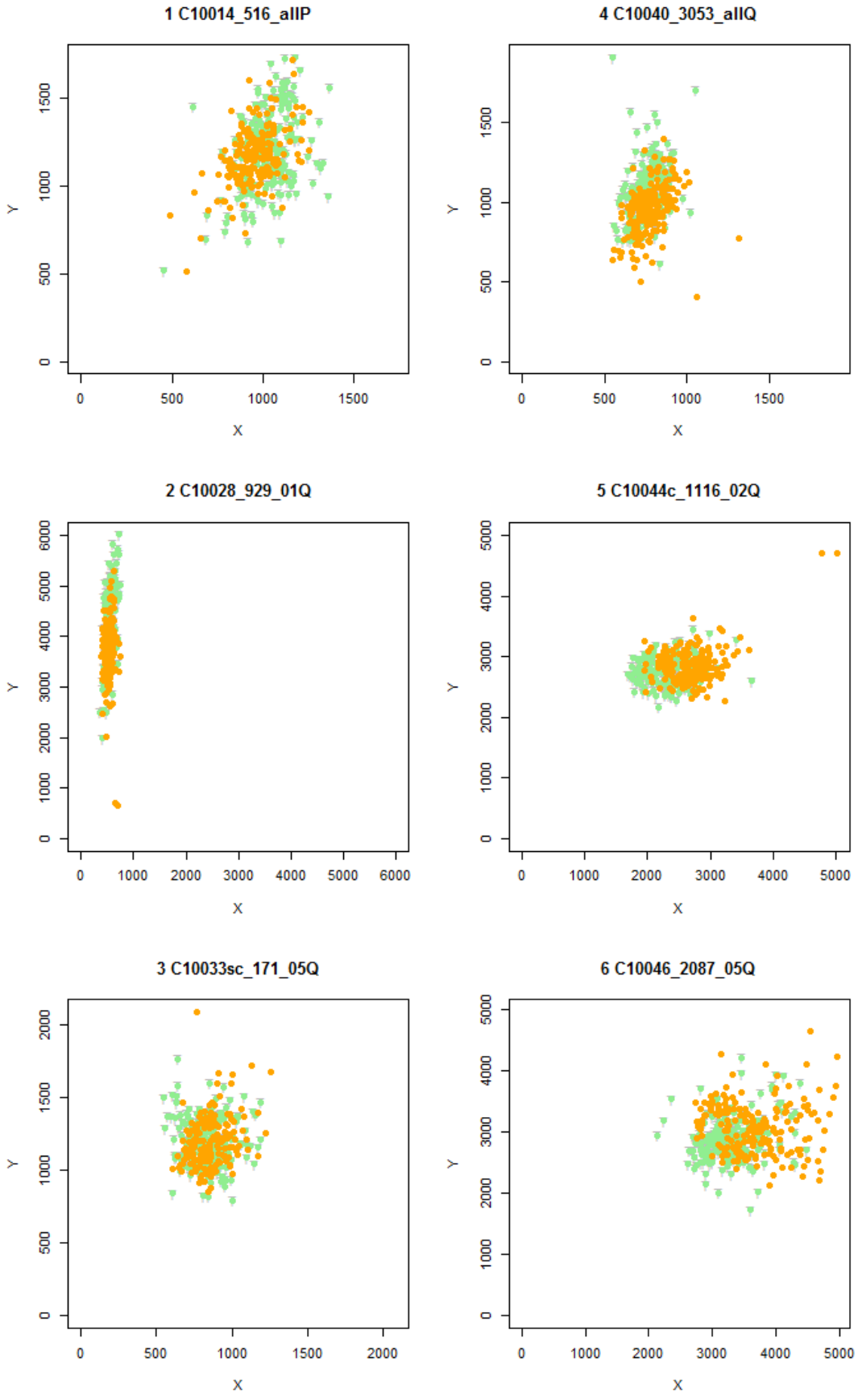

3 C10159_557_01P
Y
X
0
1000
2000
3000
4000


6 C10674c_954_allQ
Y
X
0
1000
2000
3000
4000
5000
6000

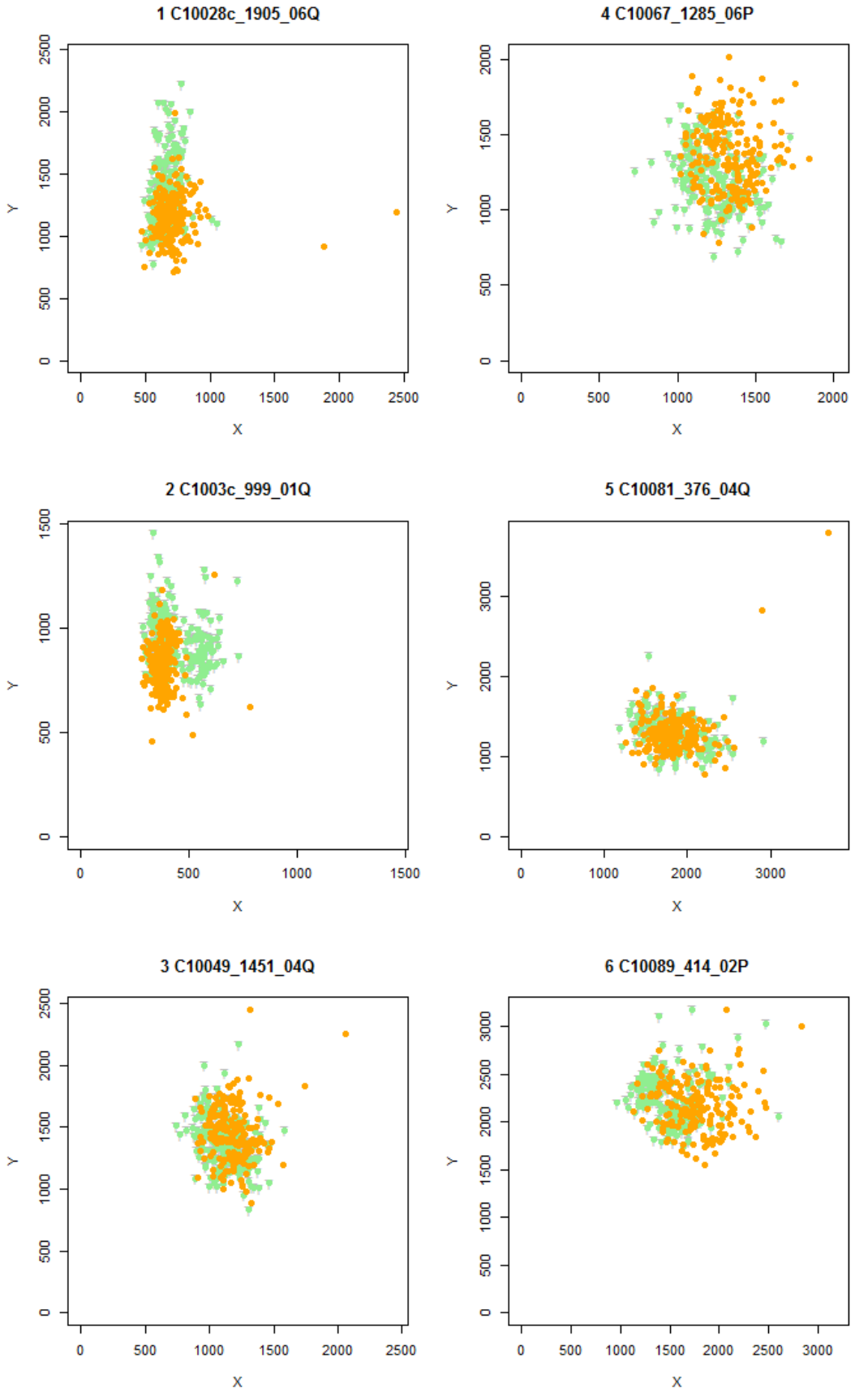
3: 0.03<SD<0.04


1 C10028_929_01P
4 C10038_169_01Q
2 C10028c_1678_06P
5 C10040_3053_allP
3 C10038_169_01P
6 C10040_3332_allQ
X
Y

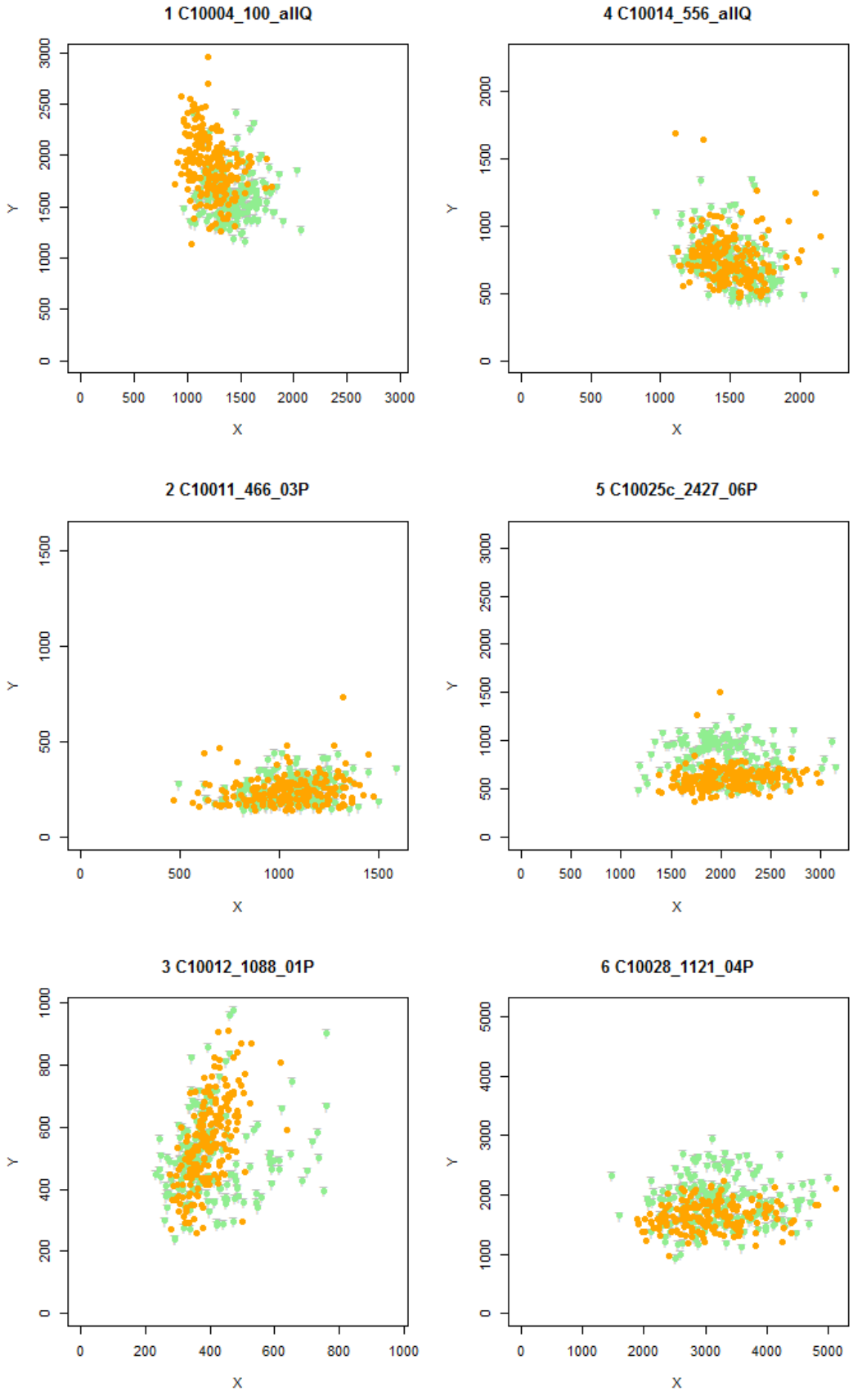
4: 0.04<SD<0.05


1 C10014_516_allP
4 C10040_3053_allQ
2 C10028_929_01Q
5 C10044c_1116_02Q
3 C10033sc_171_05Q
6 C10046_2087_05Q
X
Y

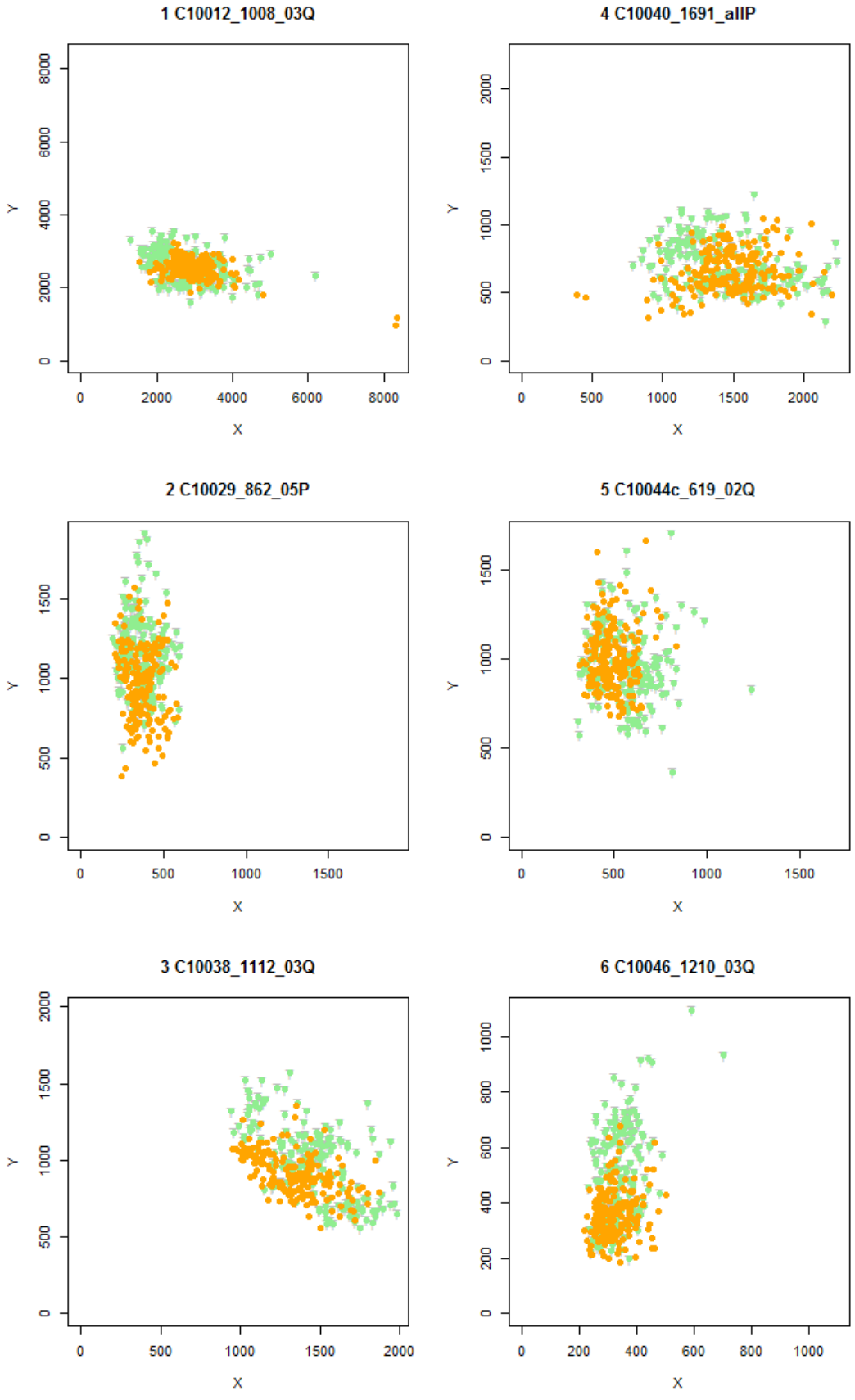
5: 0.05<SD<0.06


1 C10028c_1905_06Q
4 C10067_1285_06P
2 C1003c_999_01Q
5 C10081_376_04Q
3 C10049_1451_04Q
6 C10089_414_02P
X
Y

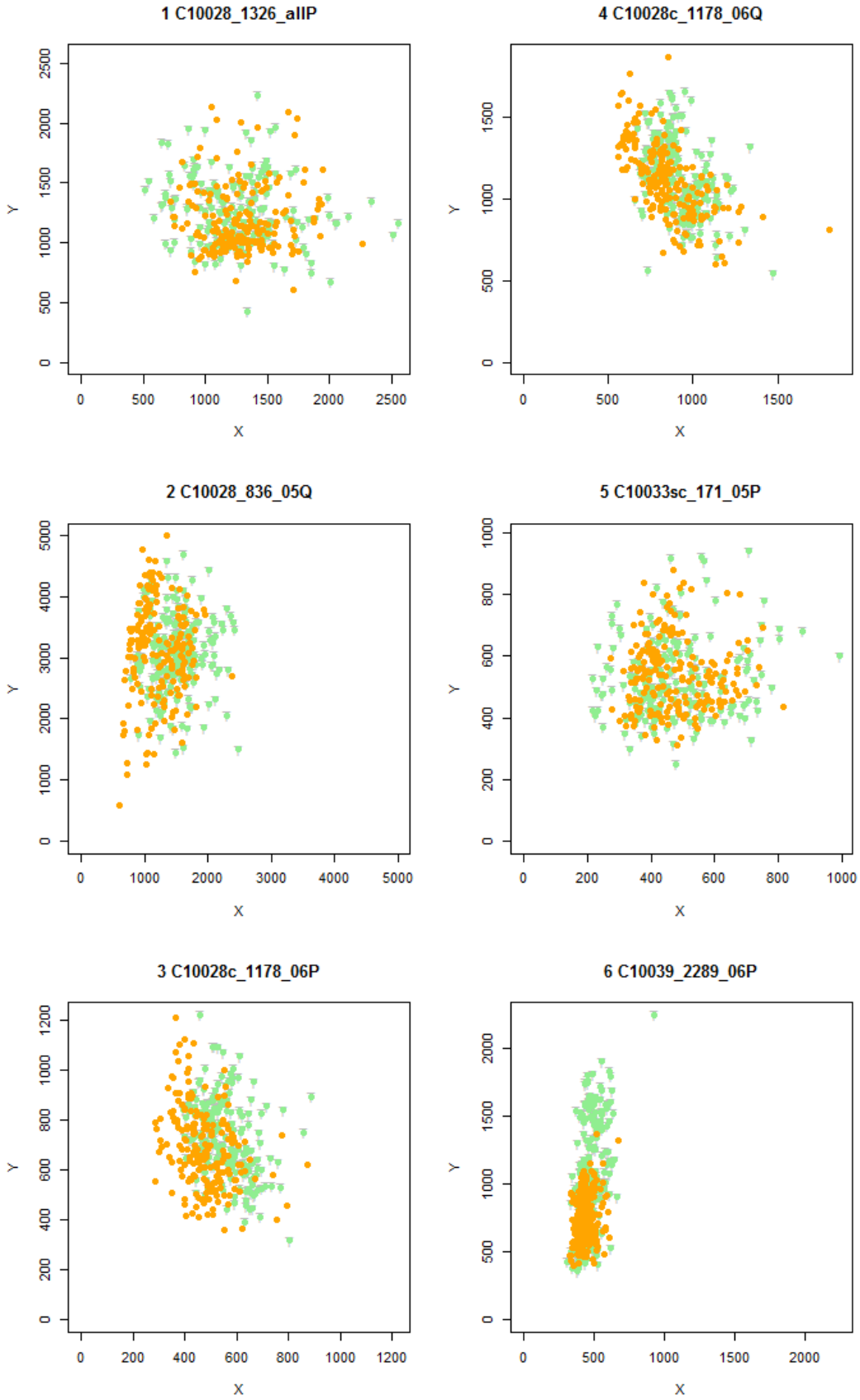
6: 0.06<SD<0.07


1 C10004_100_allQ
4 C10014_556_allQ
2 C10011_466_03P
5 C10025c_2427_06P
3 C10012_1088_01P
6 C10028_1121_04P
X
Y

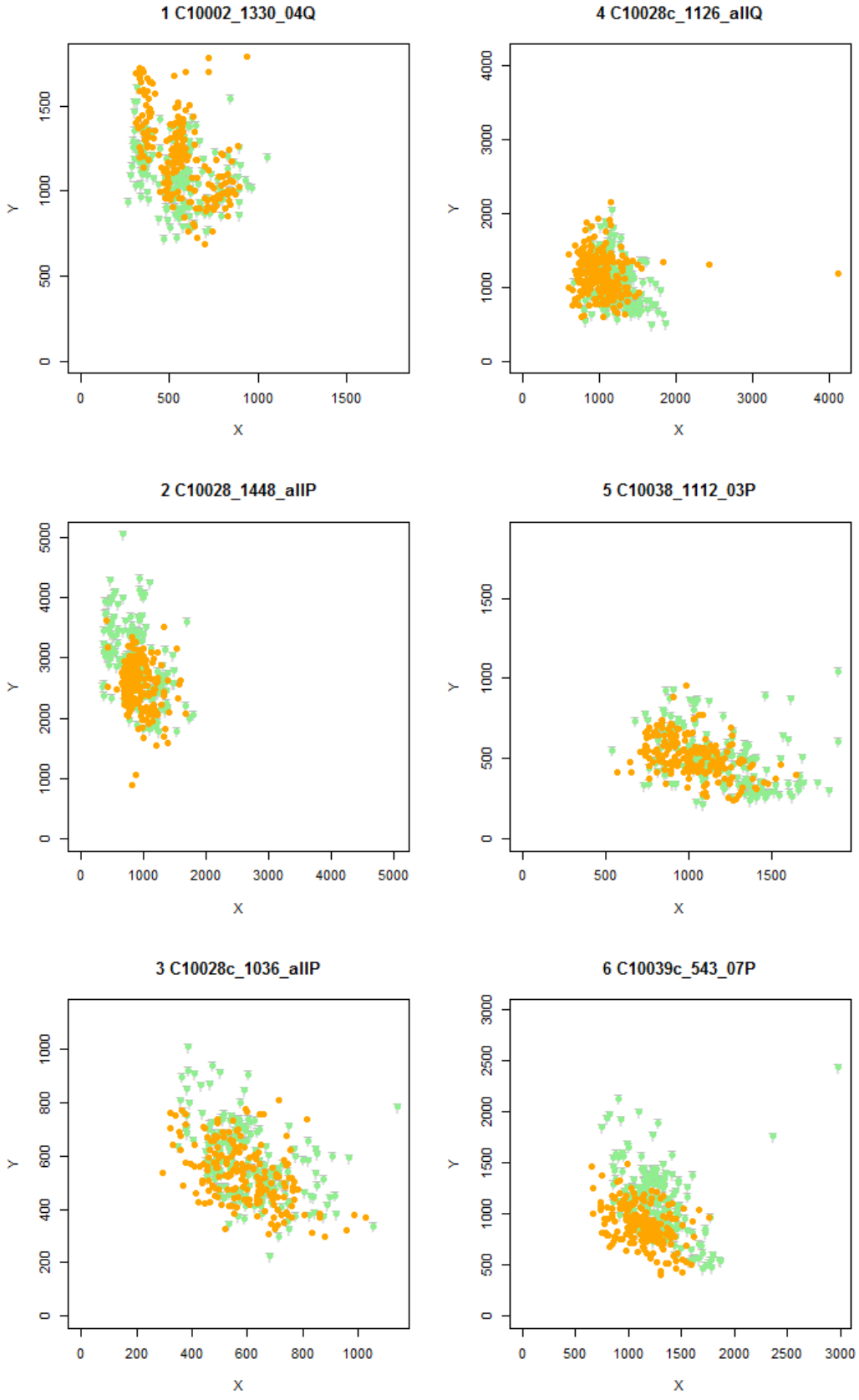
7: 0.07<SD<0.08


1 C10012_1008_03Q
4 C10040_1691_aIIP
2 C10029_862_05P
5 C10044c_619_02Q
3 C10038_1112_03Q
6 C10046_1210_03Q
X
Y

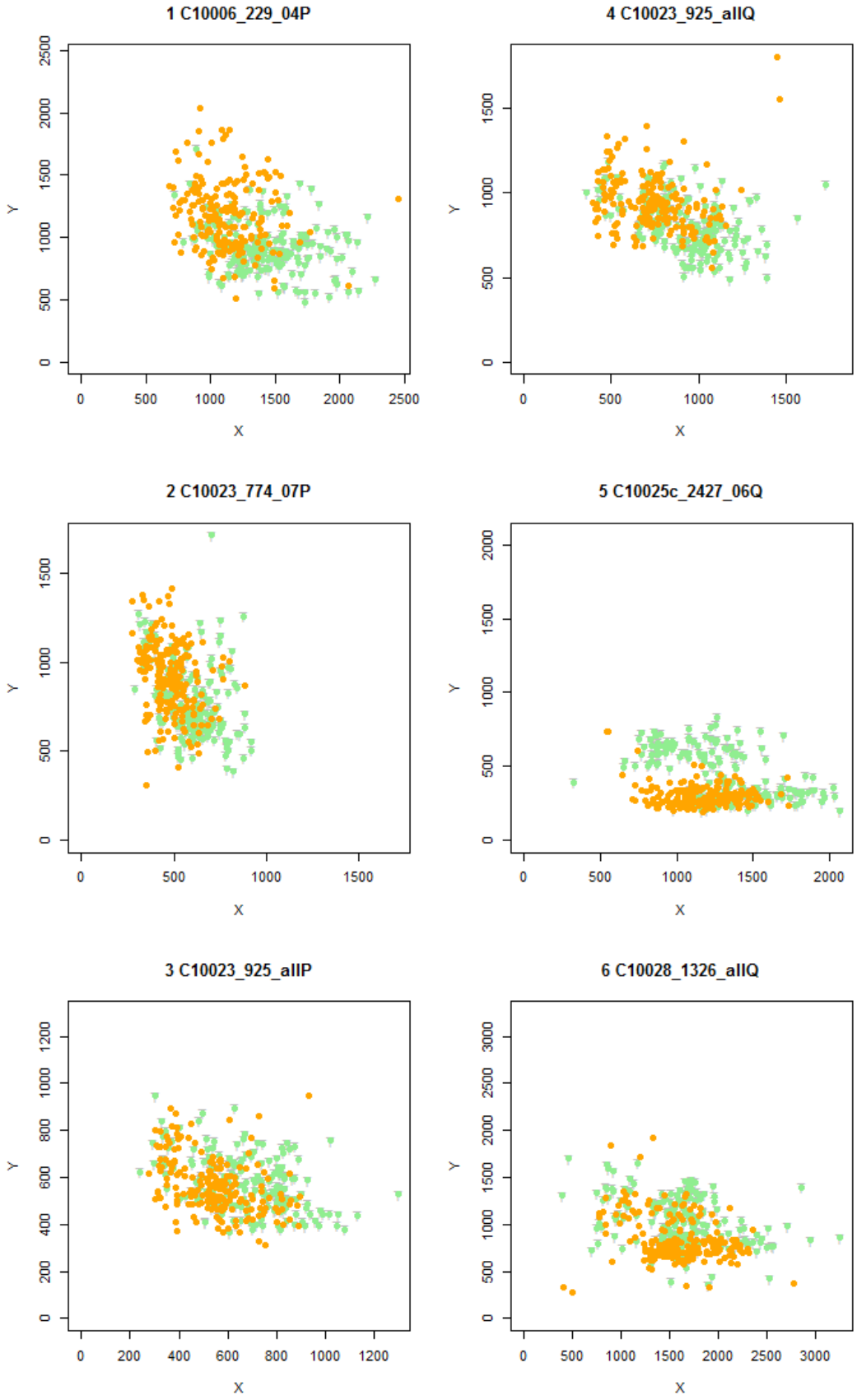
8: 0.08<SD<0.09


1 C10028_1326_aIIP
4 C10028c_1178_06Q
2 C10028_836_05Q
5 C10033sc_171_05P
3 C10028c_1178_06P
6 C10039_2289_06P
X
Y

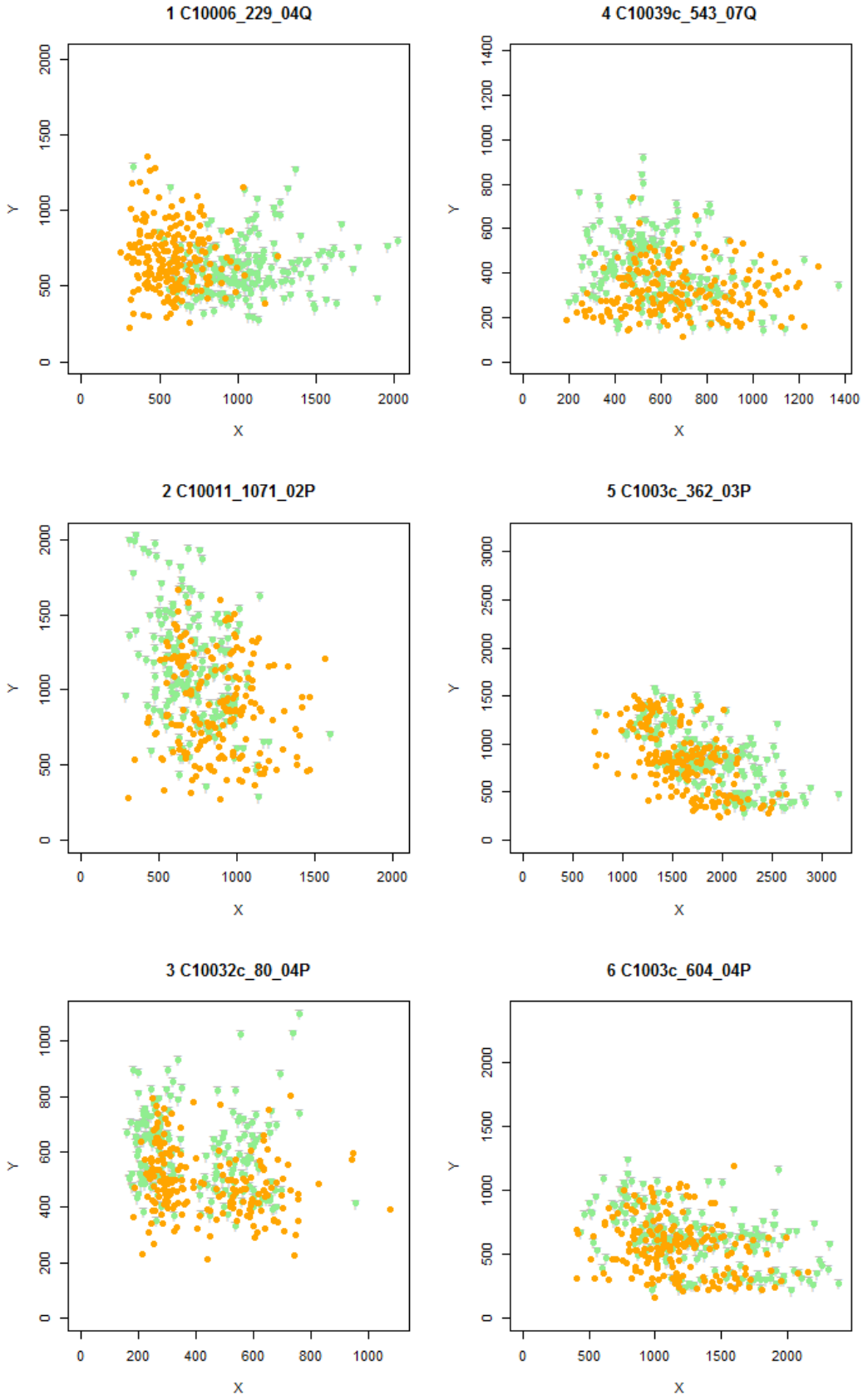
9: 0.09<SD<0.10


1 C10002_1330_04Q
4 C10028c_1126_aIIQ
2 C10028_1448_aIIP
5 C10038_1112_03P
3 C10028c_1036_aIIP
6 C10039c_543_07P
X
Y

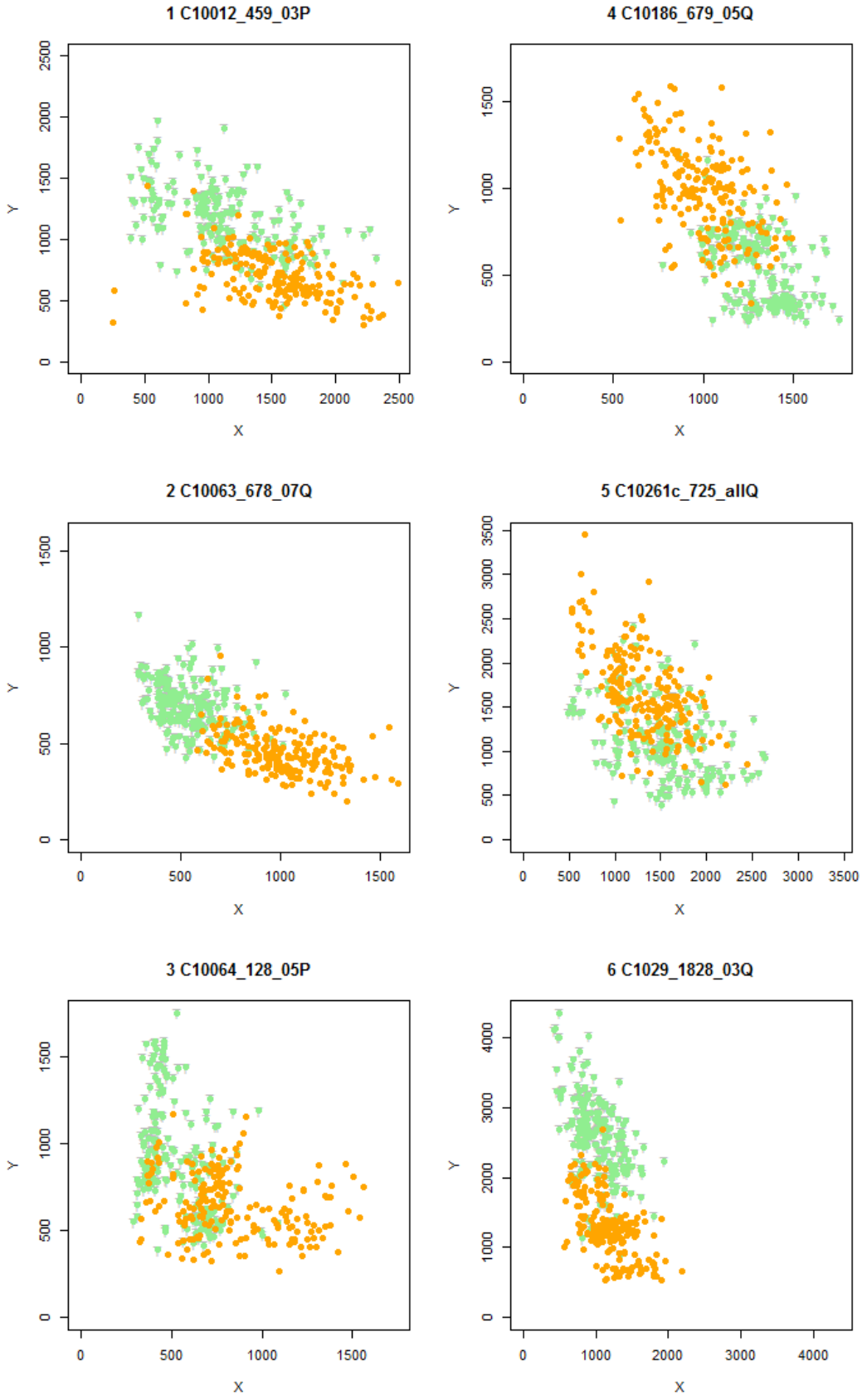
10: 0.10<SD<0.12


1 C10006_229_04P
4 C10023_925_allQ
2 C10023_774_07P
5 C10025c_2427_06Q
3 C10023_925_allP
6 C10028_1326_allQ
X
Y

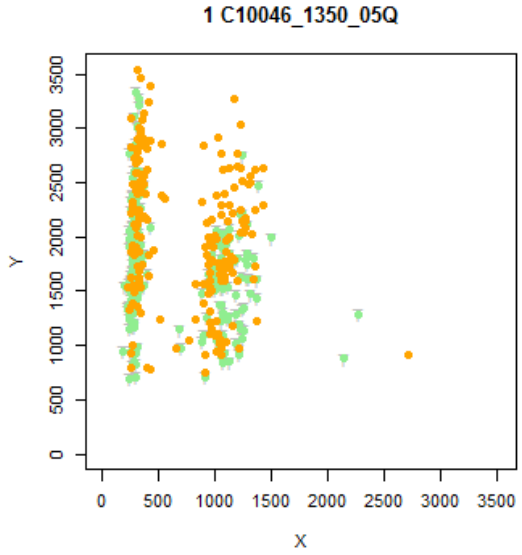
11: 0.12<SD<0.14

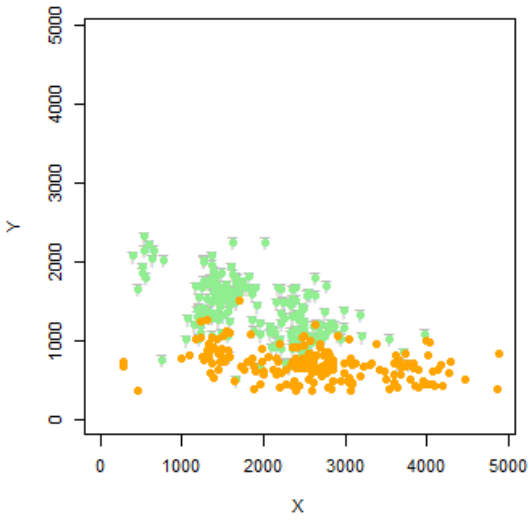

1 C10006_229_04Q
4 C10039c_543_07Q
2 C10011_1071_02P
5 C1003c_362_03P
3 C10032c_80_04P
6 C1003c_604_04P
X
Y

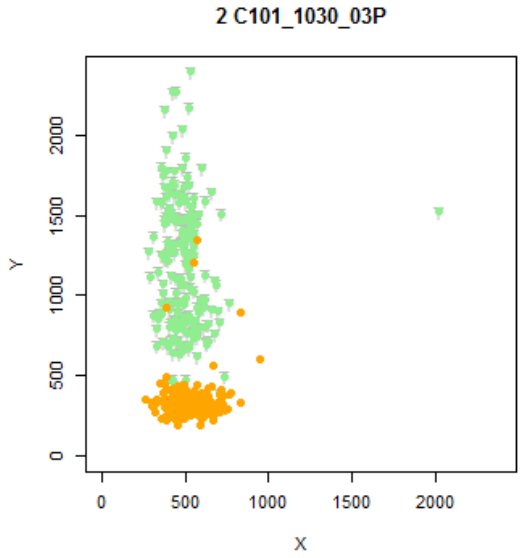
12: 0.14<SD<0.16

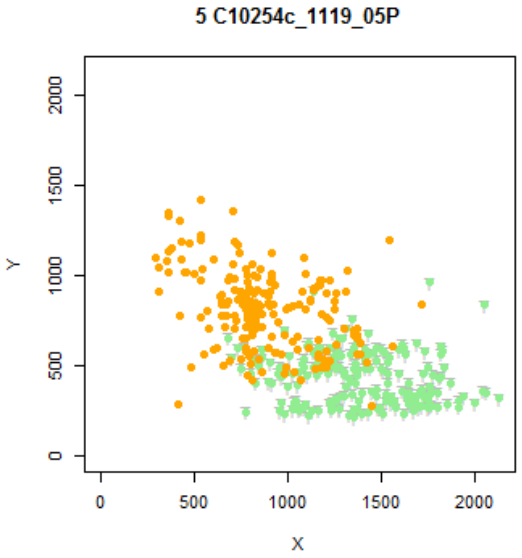

1 C10012_459_03P
4 C10186_679_05Q
2 C10063_678_07Q
5 C10261c_725_allQ
3 C10064_128_05P
6 C1029_1828_03Q
X
Y

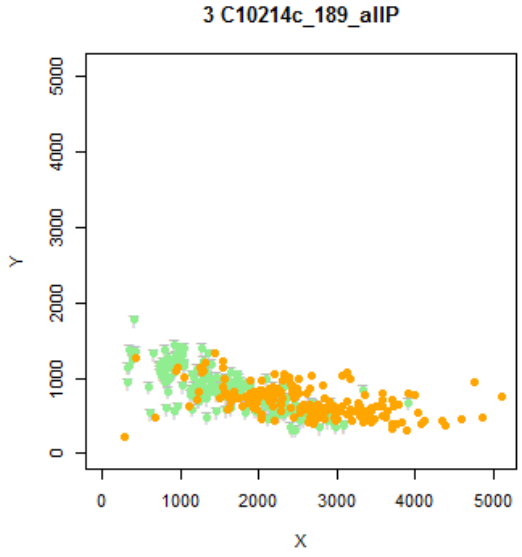
13: 0.16<SD<0.18

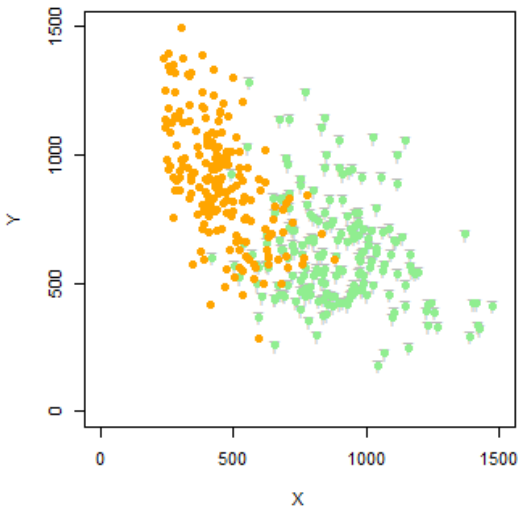

1 C10046_1350_05Q
Y
X
0
500
1000
1500
2000
2500
3000
3500


4 C10221_1674_23Q
Y
X
0
1000
2000
3000
4000
5000

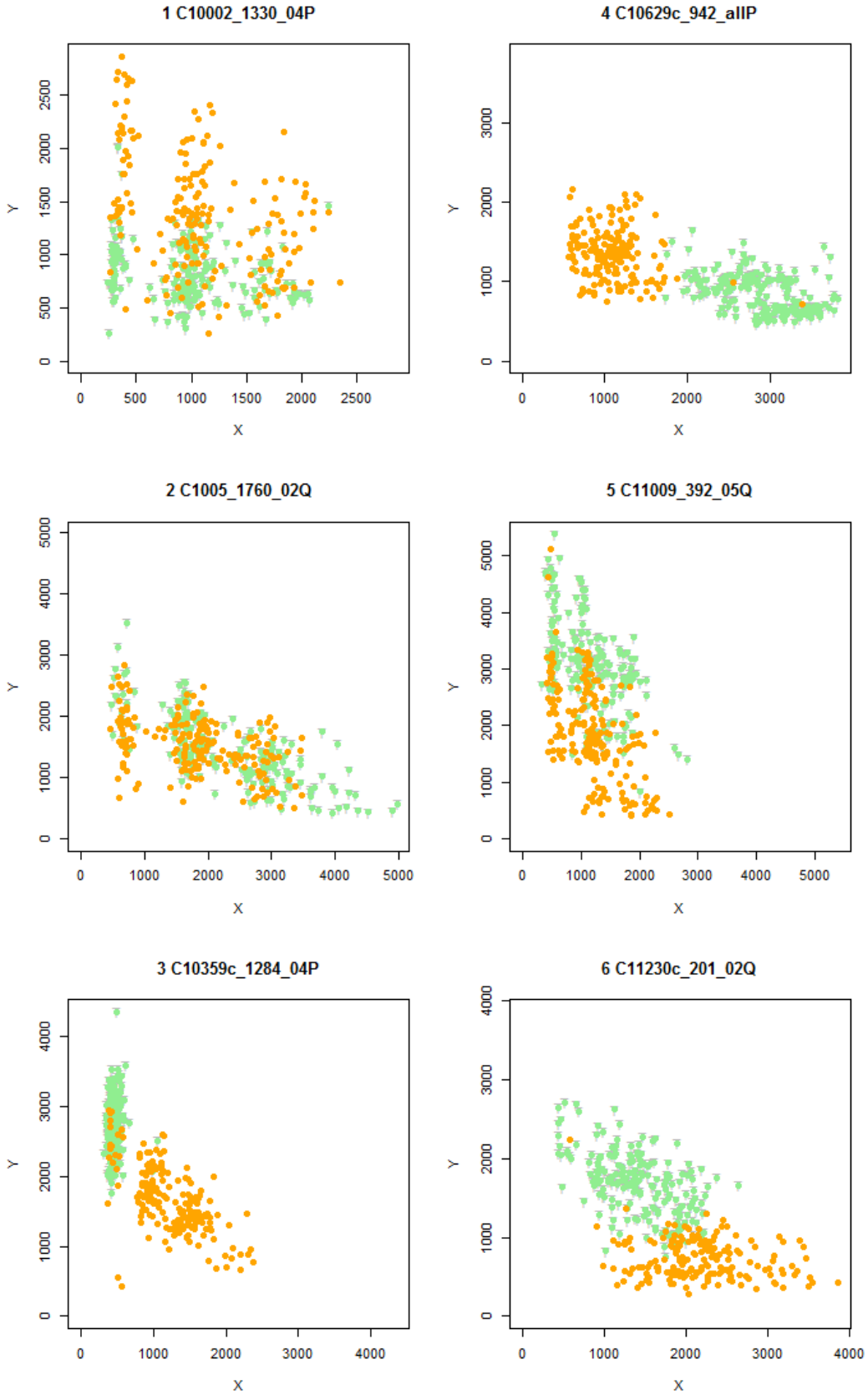

2 C101_1030_03P
Y
X
0
500
1000
1500
2000


5 C10254c_1119_05P
Y
X
0
500
1000
1500
2000

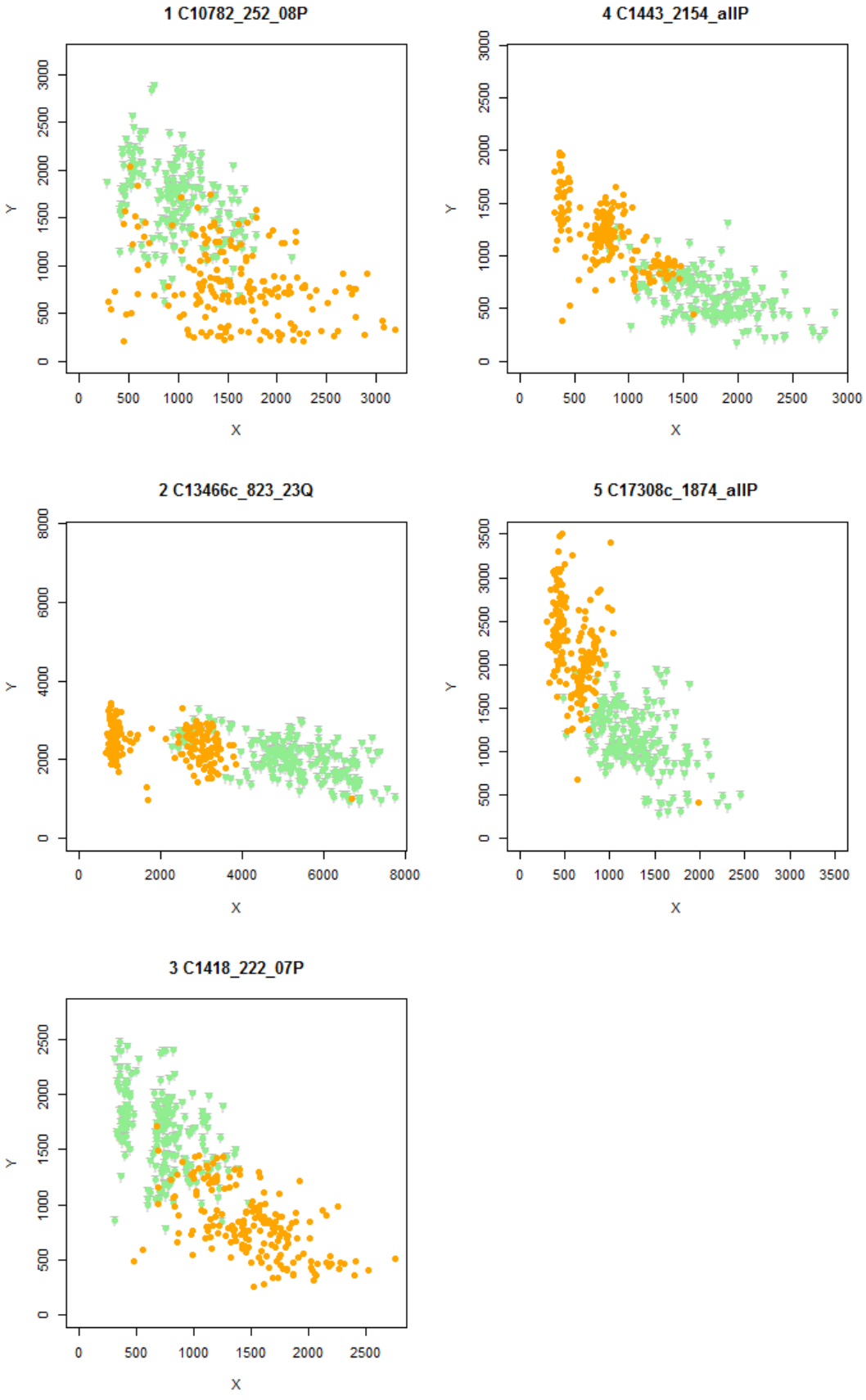

3 C10214c_189_allP
Y
X
0
1000
2000
3000
4000
5000


6 C1029_939_03Q
Y
X
0
500
1000
1500

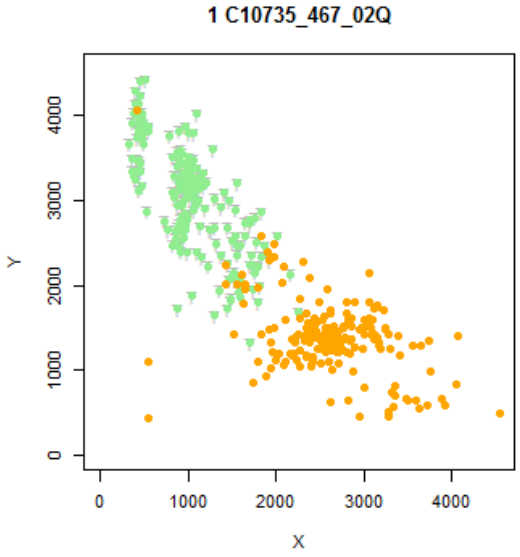
14: 0.18<SD<0.20

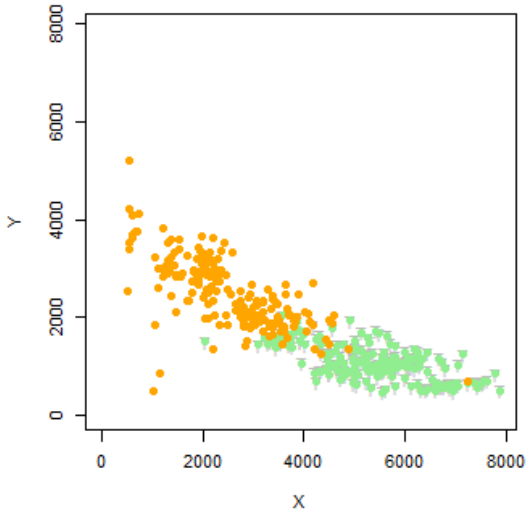

1 C10002_1330_04P
4 C10629c_942_aIIP
2 C1005_1760_02Q
5 C11009_392_05Q
3 C10359c_1284_04P
6 C11230c_201_02Q
X
Y

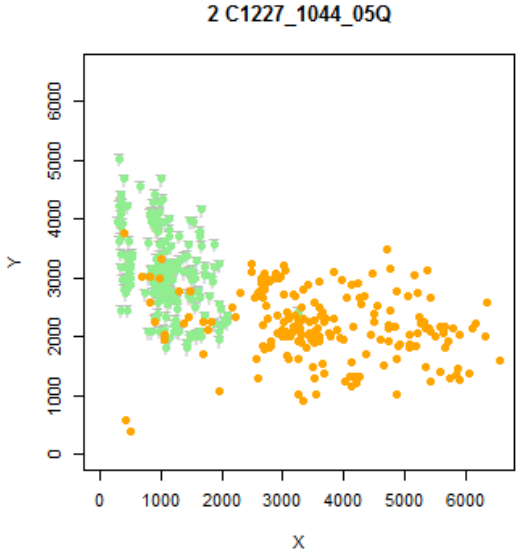
15: 0.20<SD<0.22

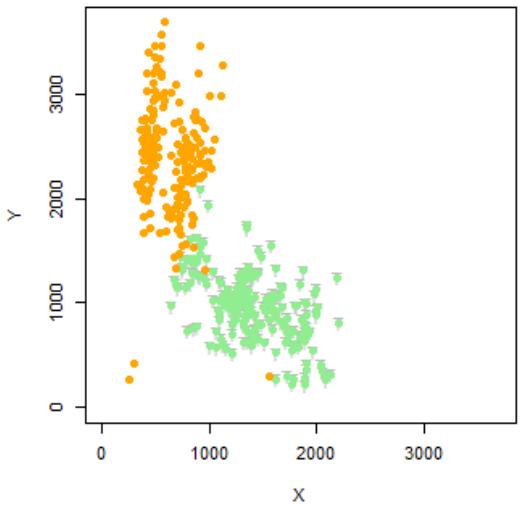

1 C10782_252_08P
4 C1443_2154_allP
2 C13466c_823_23Q
5 C17308c_1874_allP
3 C1418_222_07P
X
Y

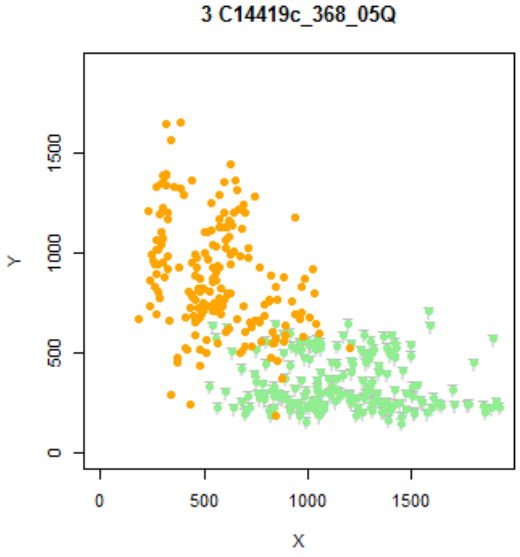
16: 0.22<SD<0.24


1 C10735_467_02Q
Y
X
0
1000
2000
3000
4000

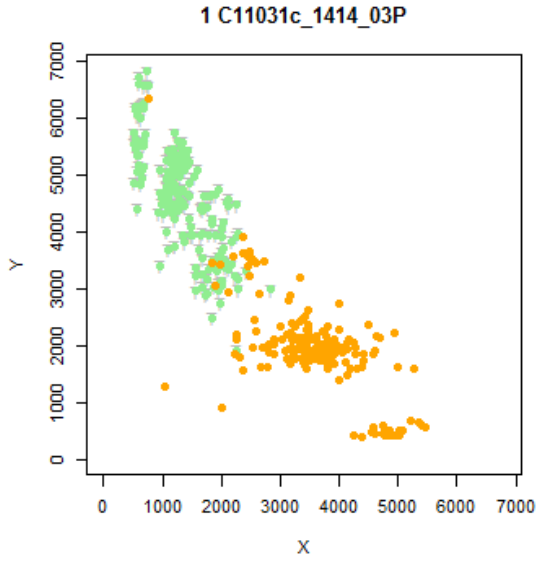

4 C15384c_1429_aIIP
Y
X
0
2000
4000
6000
8000

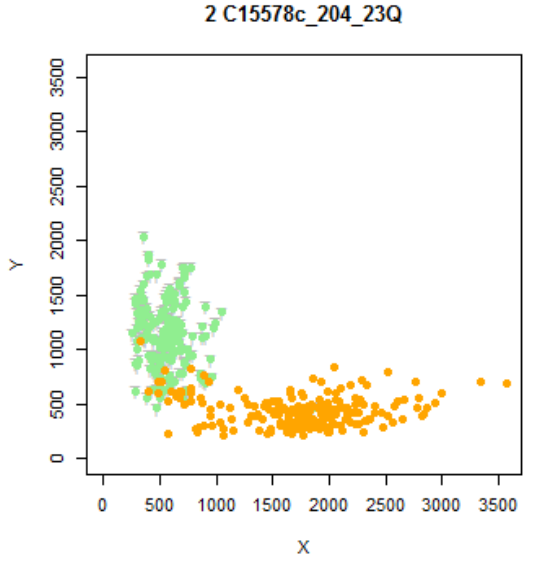

2 C1227_1044_05Q
Y
X
0
1000
2000
3000
4000
5000
6000

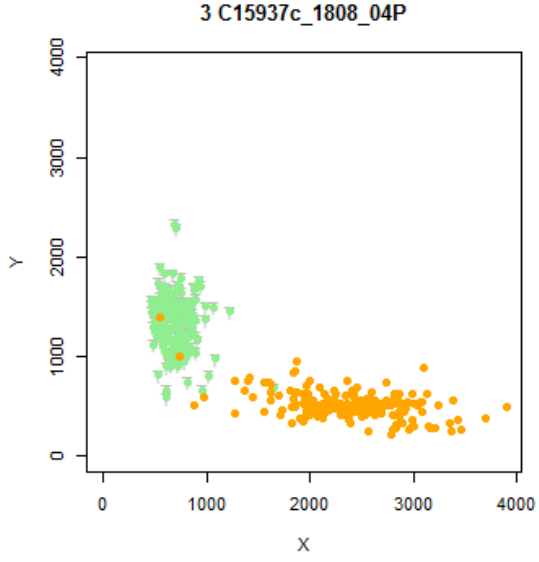

5 C16738c_2501_aIIP
Y
X
0
1000
2000
3000


3 C14419c_368_05Q
Y
X
0
500
1000
1500

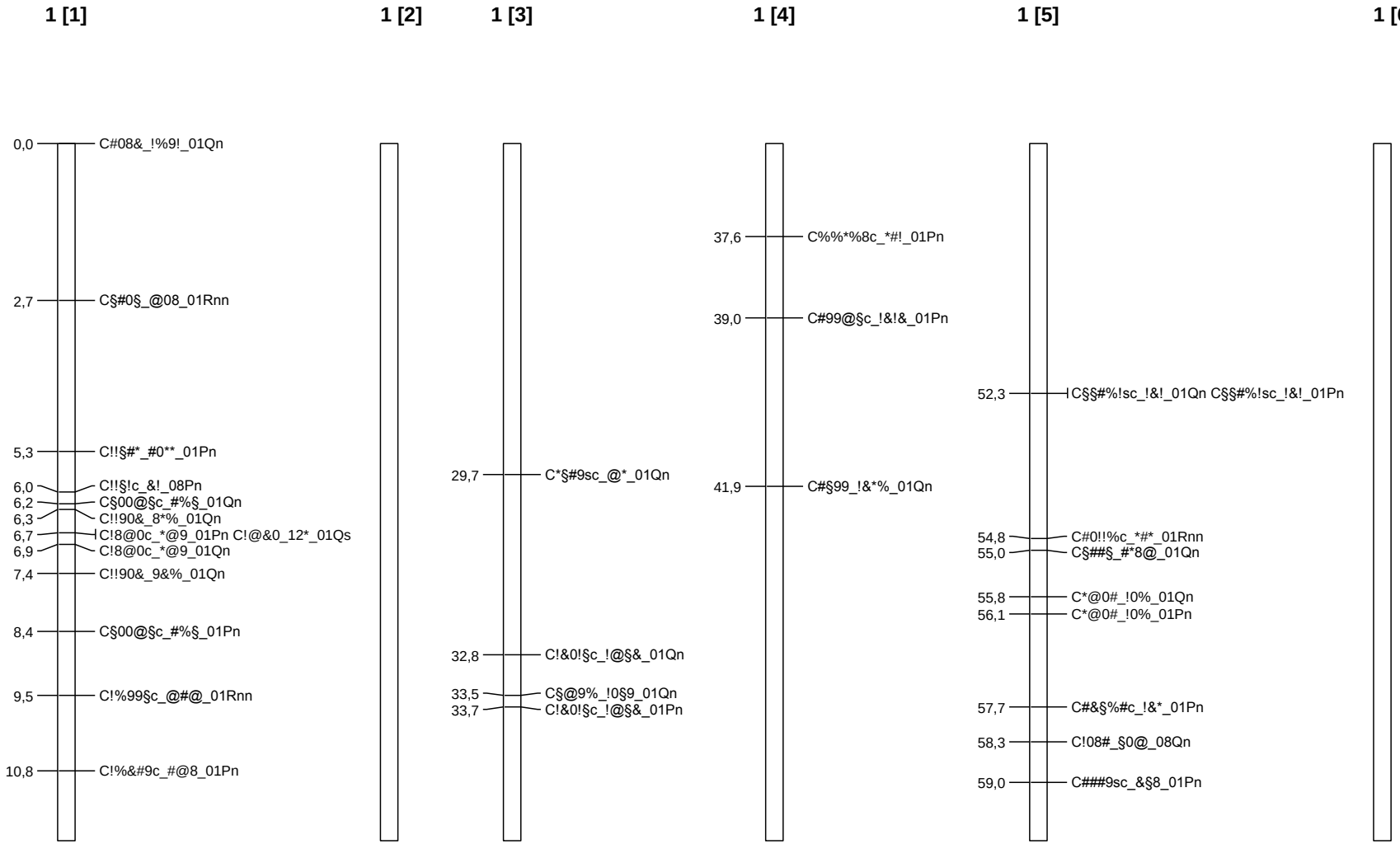
17: SD>0.26

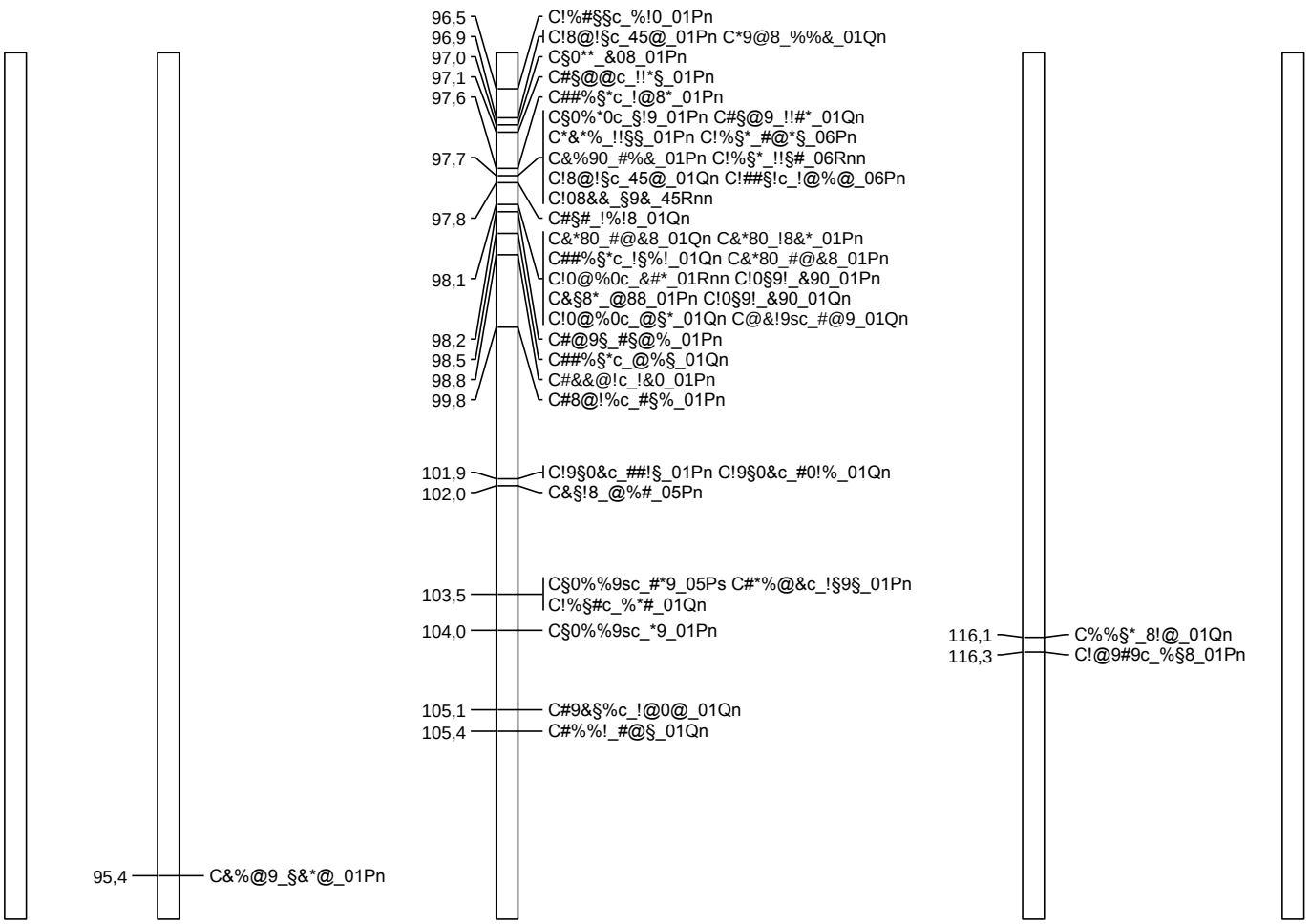

1 C11031c_1414_03P
Y
X
0
1000
2000
3000
4000
5000
6000
7000

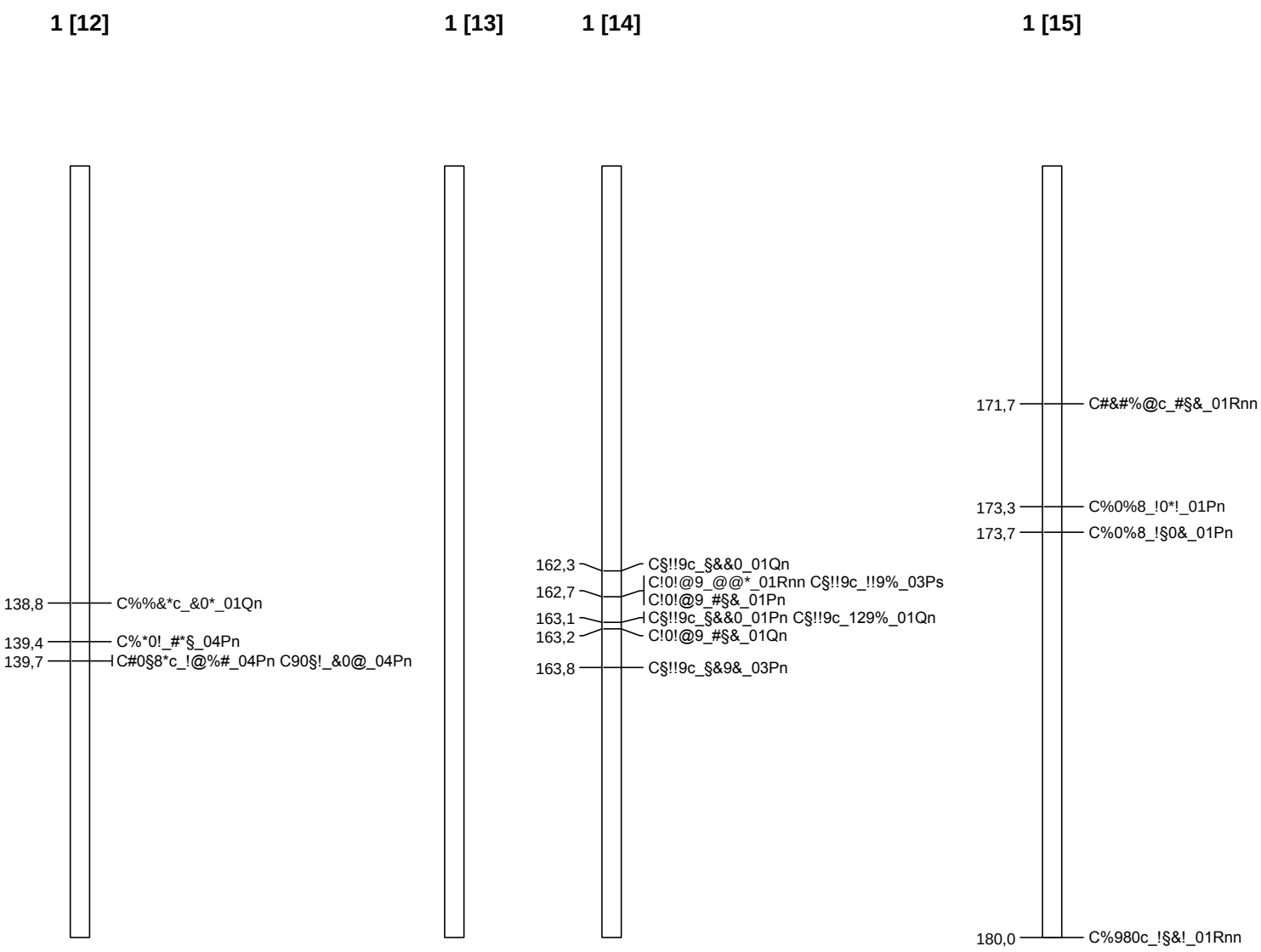

2 C15578c_204_23Q
Y
X
0
500
1000
1500
2000
2500
3000
3500

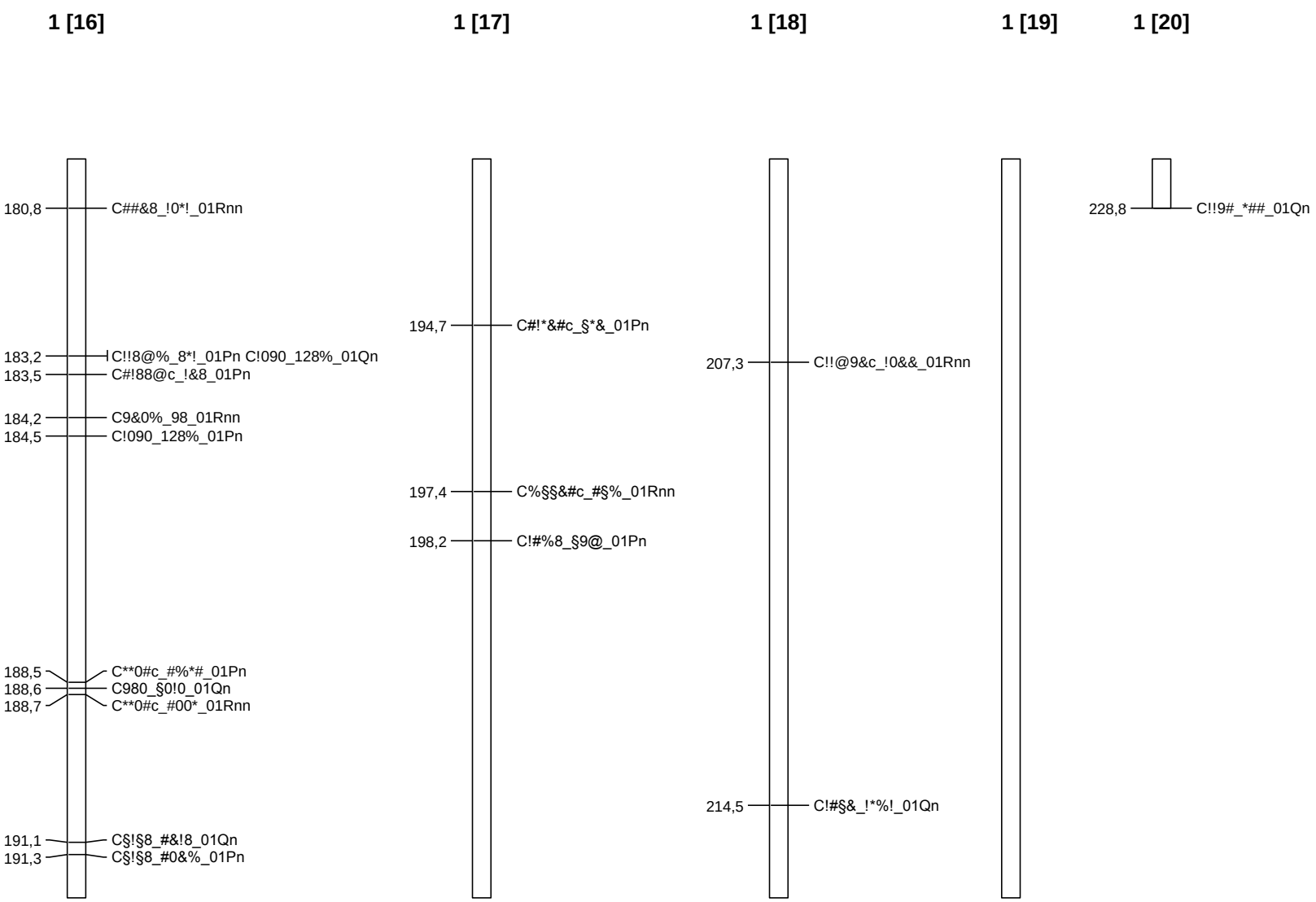

3 C15937c_1808_04P
Y
X
0
1000
2000
3000
4000

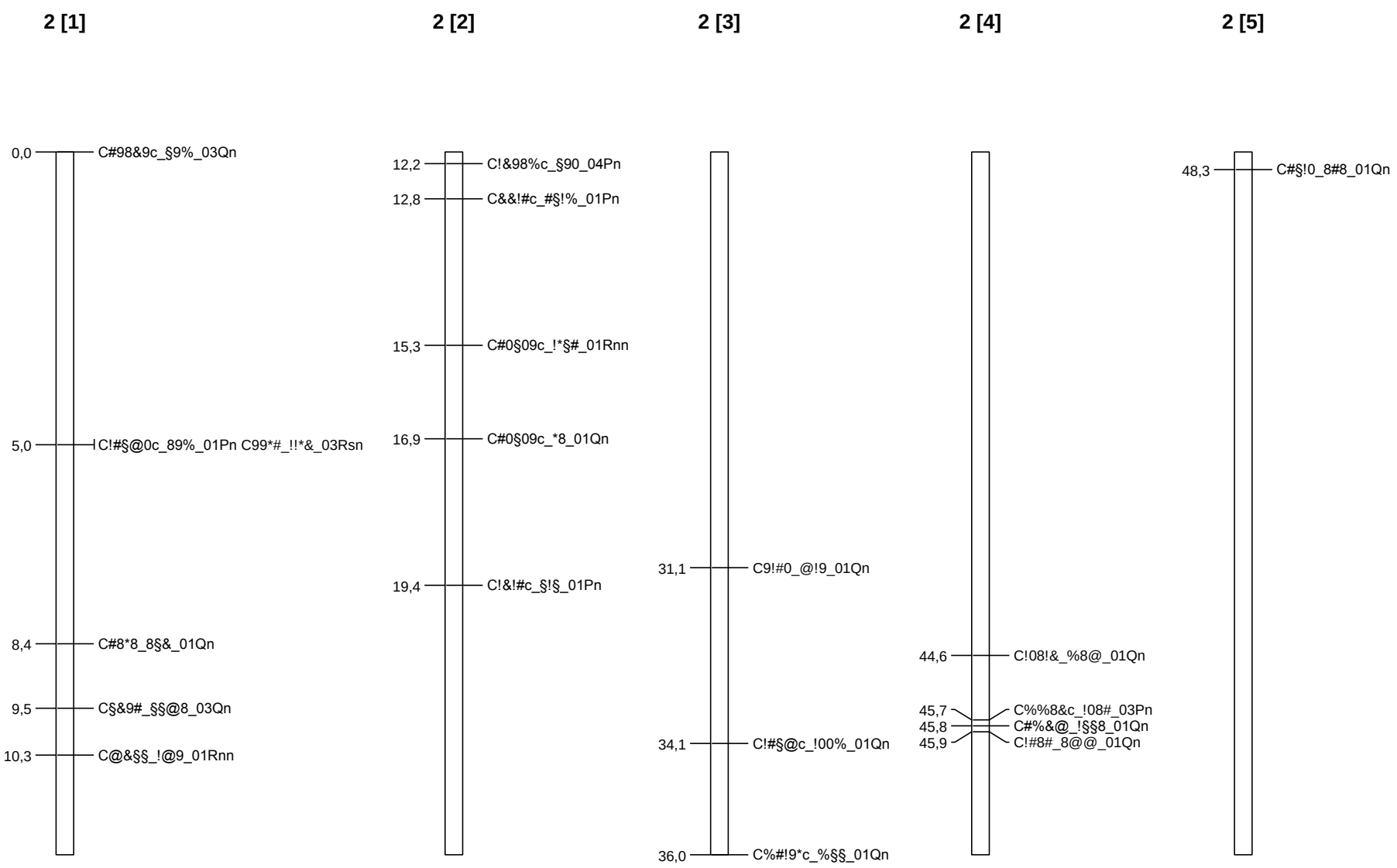
Appendix 3: Diploid linkage map

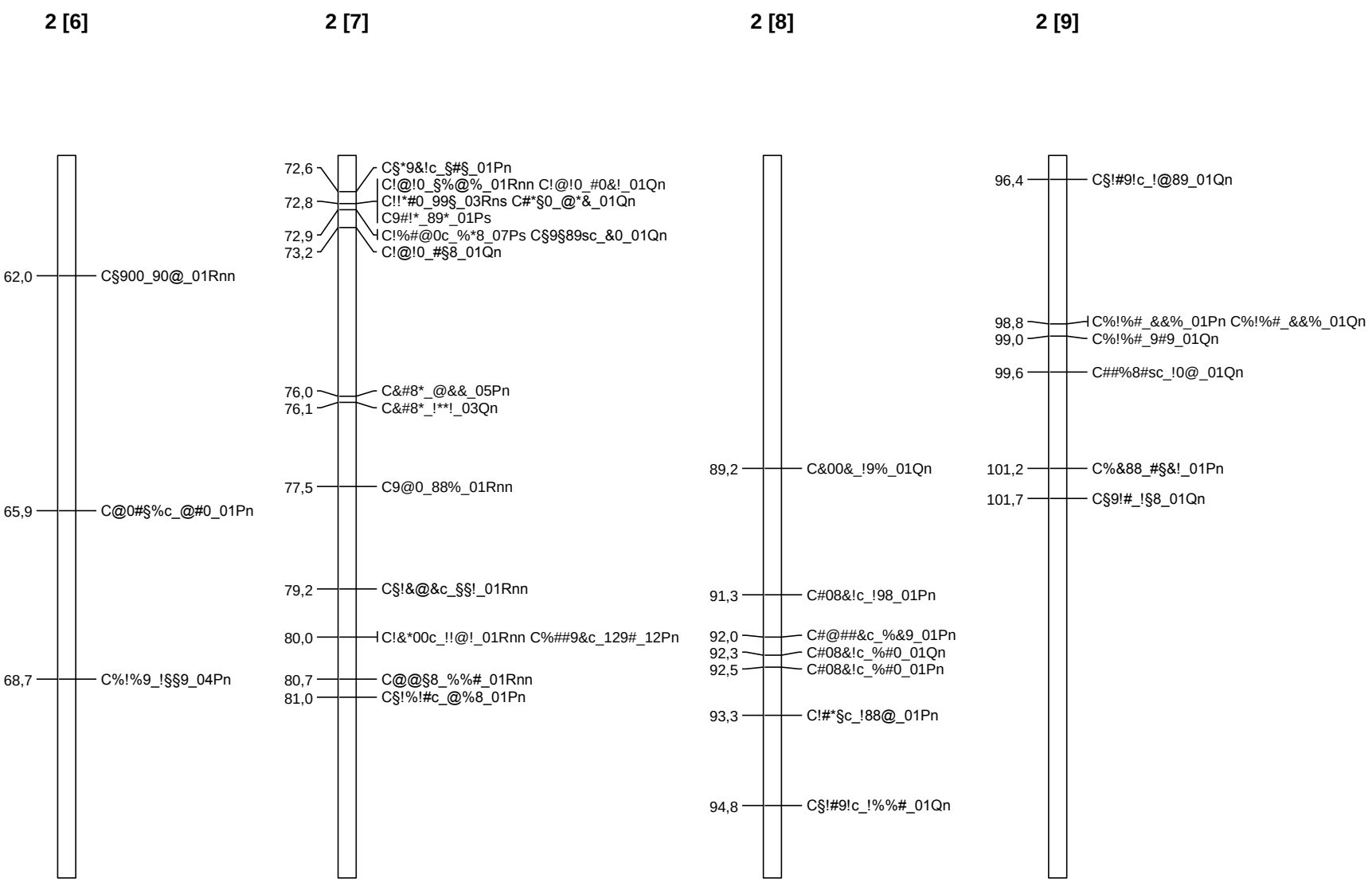


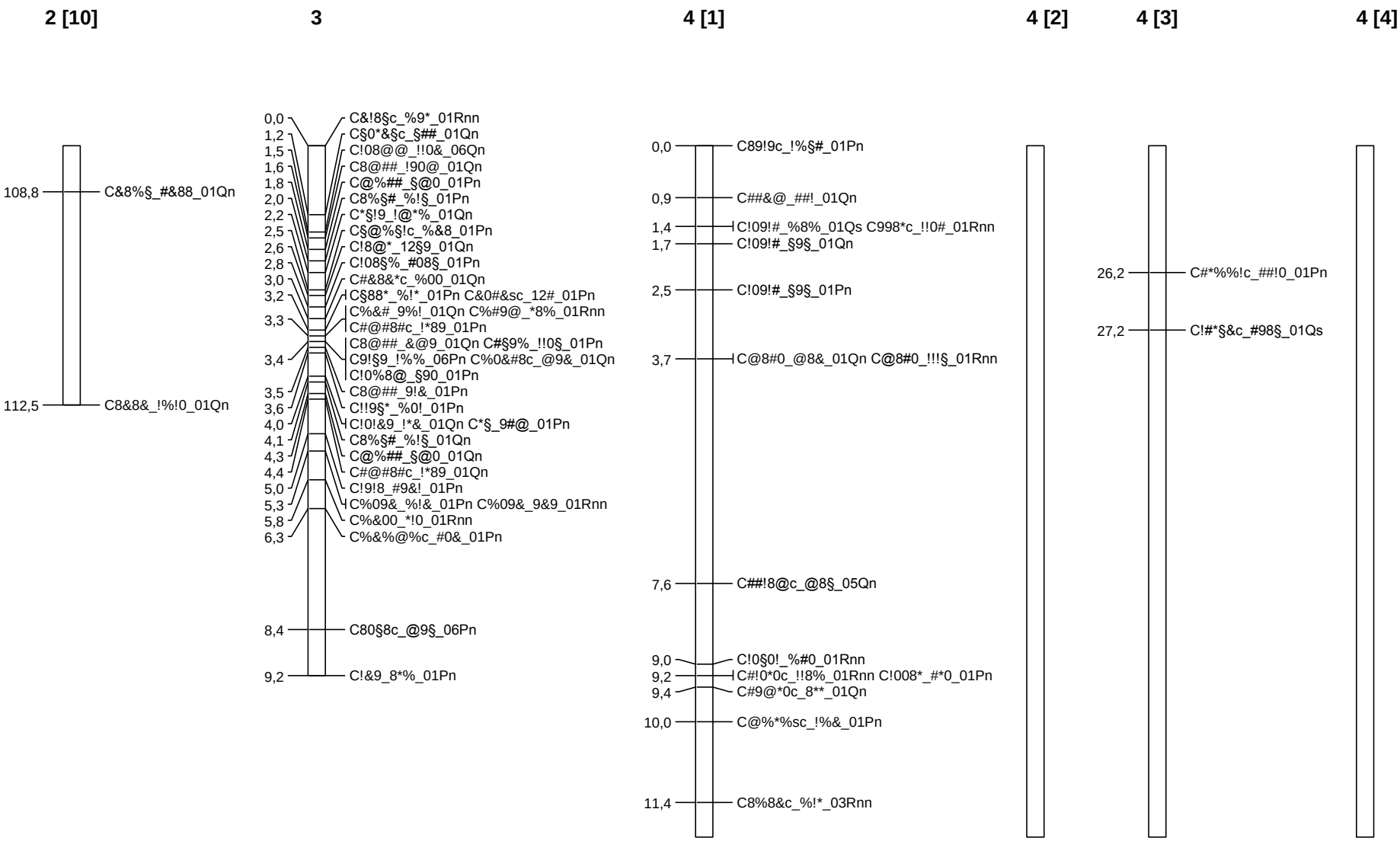

1 [7]
1 [8]
1 [9]
1 [10]
1 [11]
95,4 C&%@9_§&*@_01Pn
96,5 C!%#§§c_%!0_01Pn
96,9 C!8@!§c_45@_01Pn C*9@8_%%&_01Qn
97,0 C§0**_&08_01Pn
97,1 C#§@@c_!!*§_01Pn
97,6 C##%§*c_!@8*_01Pn
C§0%*0c_§!9_01Pn C#§@9_!!#*_01Qn
C*&*%_!!§§_01Pn C!%§*_#@*§_06Pn
97,7 C&%90_#%&_01Pn C!%§*_!!§#_06Rnn
C!8@!§c_45@_01Qn C!##§!c_!@%@_06Pn
C!08&&_§9&_45Rnn
97,8 C#§#_!%!8_01Qn
C&*80_#@&8_01Qn C&*80_!8&*_01Pn
C##%§*c_!§%!_01Qn C&*80_#@&8_01Pn
98,1 C!0@%0c_&#*_01Rnn C!0§9!_&90_01Pn
C&§8*_@88_01Pn C!0§9!_&90_01Qn
C!0@%0c_@§*_01Qn C@&!9sc_#@9_01Qn
98,2 C#@9§_#§@%_01Pn
98,5 C##%§*c_@%§_01Qn
98,8 C#&&@!c_!&0_01Pn
99,8 C#8@!%c_#§%_01Pn
101,9 C!9§0&c_##!§_01Pn C!9§0&c_#0!%_01Qn
102,0 C&§!8_@%#_05Pn
103,5 C§0%%9sc_#*9_05Ps C#*%@&c_!§9§_01Pn
C!%§#c_%*#_01Qn
104,0 C§0%%9sc_*9_01Pn
105,1 C#9&§%c_!@0@_01Qn
105,4 C#%%!_#@§_01Qn
116,1 C%%§*_8!@_01Qn
116,3 C!@9#9c_%§8_01Pn


1 [12]
1 [13]
1 [14]
1 [15]
138,8
C%%&*c_&0*_01Qn
139,4
C%*0!_#*§_04Pn
139,7
C#0§8*c_!@%#_04Pn C90§!_&0@_04Pn
162,3
C§!!9c_§&&0_01Qn
162,7
C!0!@9_@@*_01Rnn C§!!9c_!!9%_03Ps
C!0!@9_#§&_01Pn
163,1
C§!!9c_§&&0_01Pn C§!!9c_129%_01Qn
163,2
C!0!@9_#§&_01Qn
163,8
C§!!9c_§&9&_03Pn
171,7
C#&#%@c_#§&_01Rnn
173,3
C%0%8_!0*!_01Pn
173,7
C%0%8_!§0&_01Pn
180,0
C%980c_!§&!_01Rnn

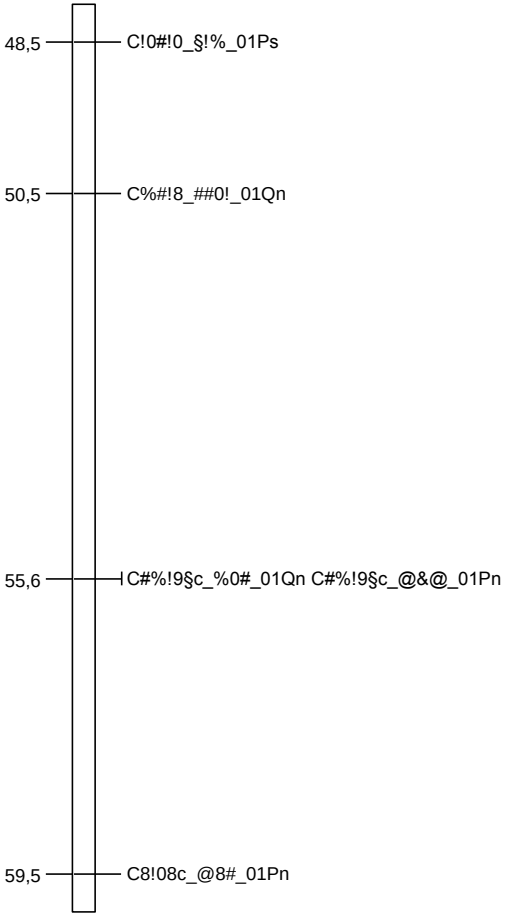

1 [16]
1 [17]
1 [18]
1 [19]
1 [20]
180,8
C##&8_!0*!_01Rnn
183,2
C!!8@%_8*!_01Pn C!090_128%_01Qn
183,5
C#!88@c_!&8_01Pn
184,2
C9&0%_98_01Rnn
184,5
C!090_128%_01Pn
188,5
C**0#c_#%*#_01Pn
188,6
C980_§0!0_01Qn
188,7
C**0#c_#00*_01Rnn
191,1
C§!§8_#&!8_01Qn
191,3
C§!§8_#0&%_01Pn
194,7
C#!*&#c_§*&_01Pn
197,4
C%§§&#c_#§%_01Rnn
198,2
C!#%8_§9@_01Pn
207,3
C!!@9&c_!0&&_01Rnn
214,5
C!#§&_!*%!_01Qn
228,8
C!!9#_*##_01Qn


2 [1]
0,0
C#98&9c_§9%_03Qn
5,0
C!#§@0c_89%_01Pn C99*#_!!*&_03Rsn
8,4
C#8*8_8§&_01Qn
9,5
C§&9#_§§@8_03Qn
10,3
C@&§§_!@9_01Rnn
2 [2]
12,2
C!&98%c_§90_04Pn
12,8
C&&!#c_#§!%_01Pn
15,3
C#0§09c_!*§#_01Rnn
16,9
C#0§09c_*8_01Qn
19,4
C!&!#c_§!§_01Pn
2 [3]
31,1
C9!#0_@!9_01Qn
34,1
C!#§@c_!00%_01Qn
36,0
C%#!9*c_%§§_01Qn
2 [4]
44,6
C!08!&_%8@_01Qn
45,7
C%%8&c_!08#_03Pn
45,8
C#%&@_!§§8_01Qn
45,9
C!#8#_8@@_01Qn
2 [5]
48,3
C#§!0_8#8_01Qn

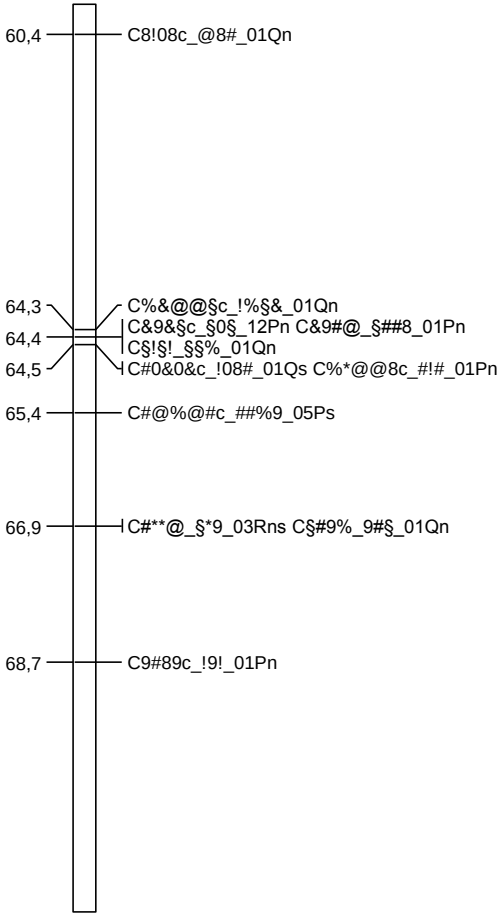

2 [6]
62,0
C§900_90@_01Rnn
65,9
C@0#§%c_@#0_01Pn
68,7
C%!%9_!§§9_04Pn
2 [7]
72,6
C§*9&!c_§#§_01Pn
C!@!0_§%@%_01Rnn C!@!0_#0&!_01Qn
72,8
C!!*#0_99§_03Rns C#*§0_@*&_01Qn
C9#!*_89*_01Ps
72,9
C!%#@0c_%*8_07Ps C§9§89sc_&0_01Qn
73,2
C!@!0_#§8_01Qn
76,0
C&#8*_@&&_05Pn
76,1
C&#8*_!**!_03Qn
77,5
C9@0_88%_01Rnn
79,2
C§!&@&c_§§!_01Rnn
80,0
C!&*00c_!!@!_01Rnn C%##9&c_129#_12Pn
80,7
C@@§8_%%#_01Rnn
81,0
C§!%!#c_@%8_01Pn
2 [8]
89,2
C&00&_!9%_01Qn
91,3
C#08&!c_!98_01Pn
92,0
C#@##&c_%&9_01Pn
92,3
C#08&!c_%#0_01Qn
92,5
C#08&!c_%#0_01Pn
93,3
C!#*§c_!88@_01Pn
94,8
C§!#9!c_!%%#_01Qn
2 [9]
96,4
C§!#9!c_!@89_01Qn
98,8
C%!%#_&&%_01Pn C%!%#_&&%_01Qn
99,0
C%!%#_9#9_01Qn
99,6
C##%8#sc_!0@_01Qn
101,2
C%&88_#§&!_01Pn
101,7
C§9!#_!§8_01Qn


2 [10]
3
4 [1]
4 [2]
4 [3]
4 [4]
108,8 C&8%§_#&88_01Qn
112,5 C8&8&_!%!0_01Qn
0,0 C&!8§c_%9*_01Rnn
1,2 C§0*&§c_§##_01Qn
1,5 C!08@@_!!0&_06Qn
1,6 C8@##_!90@_01Qn
1,8 C@%##_§@0_01Pn
2,0 C8%§#_%!§_01Pn
2,2 C*§!9_!@*%_01Qn
2,5 C§@%§!c_%&8_01Pn
2,6 C!8@*_12§9_01Qn
2,8 C!08§%_#08§_01Pn
3,0 C#&8&*c_%00_01Qn
3,2 C§88*_%!*_01Pn C&0#&sc_12#_01Pn
3,3 C%&#_9%!_01Qn C%#9@_*8%_01Rnn
C#@#8#c_!*89_01Pn
C8@##_&@9_01Qn C#§9%_!!0§_01Pn
3,4 C9!§9_!%%_06Pn C%0&#8c_@9&_01Qn
C!0%8@_§90_01Pn
3,5 C8@##_9!&_01Pn
3,6 C!!9§*_%0!_01Pn
4,0 C!0!&9_!*&_01Qn C*§_9#@_01Pn
4,1 C8%§#_%!§_01Qn
4,3 C@%##_§@0_01Qn
4,4 C#@#8#c_!*89_01Qn
5,0 C!9!8_#9&!_01Pn
5,3 C%09&_%!&_01Pn C%09&_9&9_01Rnn
5,8 C%&00_*!0_01Rnn
6,3 C%&%@%c_#0&_01Pn
8,4 C80§8c_@9§_06Pn
9,2 C!&9_8*%_01Pn
0,0 C89!9c_!%§#_01Pn
0,9 C##&@_##!_01Qn
1,4 C!09!#_%8%_01Qs C998*c_!!0#_01Rnn
1,7 C!09!#_§9§_01Qn
2,5 C!09!#_§9§_01Pn
3,7 C@8#0_@8&_01Qn C@8#0_!!!§_01Rnn
7,6 C##!8@c_@8§_05Qn
9,0 C!0§0!_%#0_01Rnn
9,2 C#!0*0c_!!8%_01Rnn C!008*_#*0_01Pn
9,4 C#9@*0c_8**_01Qn
10,0 C@%*%sc_!%&_01Pn
11,4 C8%8&c_%!*_03Rnn
26,2 C#*%%!c_##!0_01Pn
27,2 C!#*§&c_#98§_01Qs

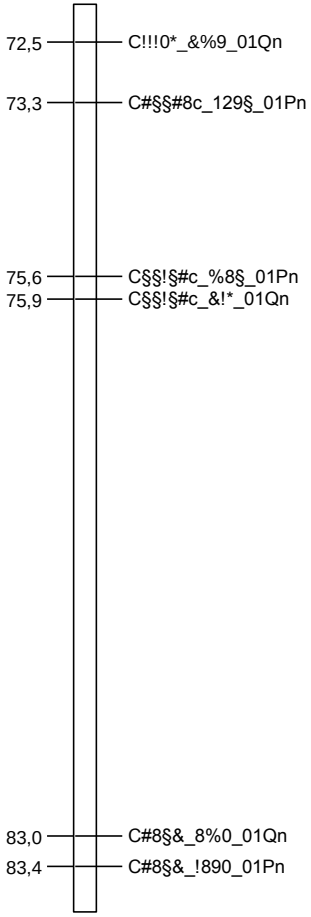
4 [5]


48,5
C!0#!0_§!%_01Ps
50,5
C%#!8_##0!_01Qn
55,6
C#%!9§c_%0#_01Qn C#%!9§c_@&@_01Pn
59,5
C8!08c_@8#_01Pn

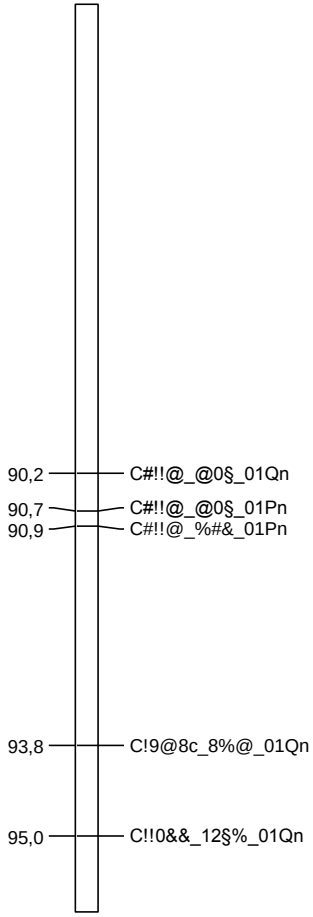
4 [6]


60,4
C8!08c_@8#_01Qn
64,3
C%&@@§c_!%§&_01Qn
64,4
C&9&§c_§0§_12Pn C&9#@_§##8_01Pn
64,4
C§!§!_§§%_01Qn
64,5
C#0&0&c_!08#_01Qs C%*@@8c_#!#_01Pn
65,4
C#@%@#c_##%9_05Ps
66,9
C#**@_§*9_03Rns C§#9%_9#§_01Qn
68,7
C9#89c_!9!_01Pn

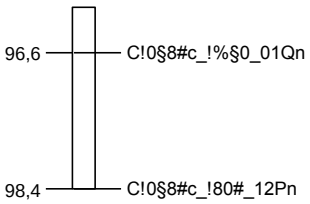
4 [7]


72,5
C!!!0*_&%9_01Qn
73,3
C#§§#8c_129§_01Pn
75,6
C§§!§#c_%8§_01Pn
75,9
C§§!§#c_&!*_01Qn
83,0
C#8§&_8%0_01Qn
83,4
C#8§&_!890_01Pn

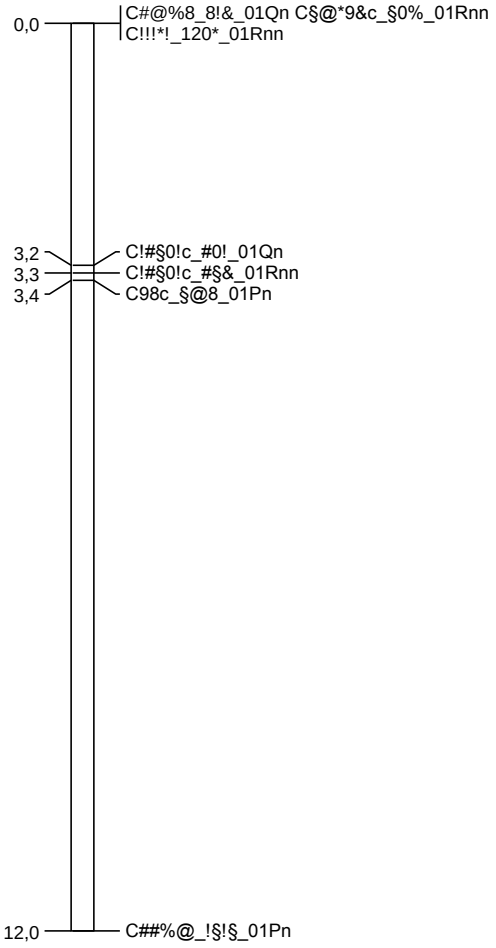
4 [8]


90,2
C#!!@_@0§_01Qn
90,7
C#!!@_@0§_01Pn
90,9
C#!!@_%#&_01Pn
93,8
C!9@8c_8%@_01Qn
95,0
C!!0&&_12§%_01Qn

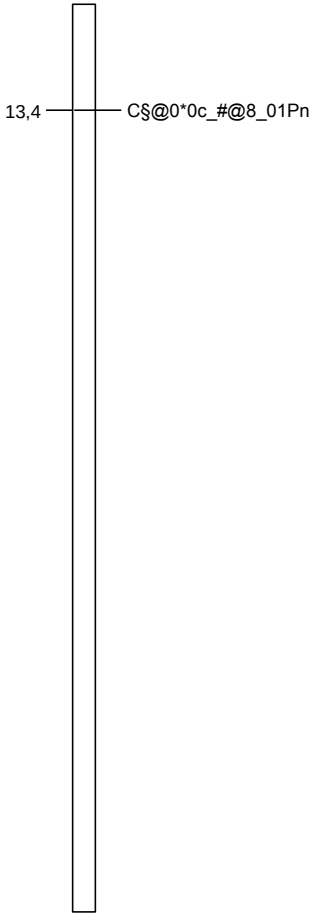
4 [9]


96,6
C!0§8#c_!%§0_01Qn
98,4
C!0§8#c_!80#_12Pn

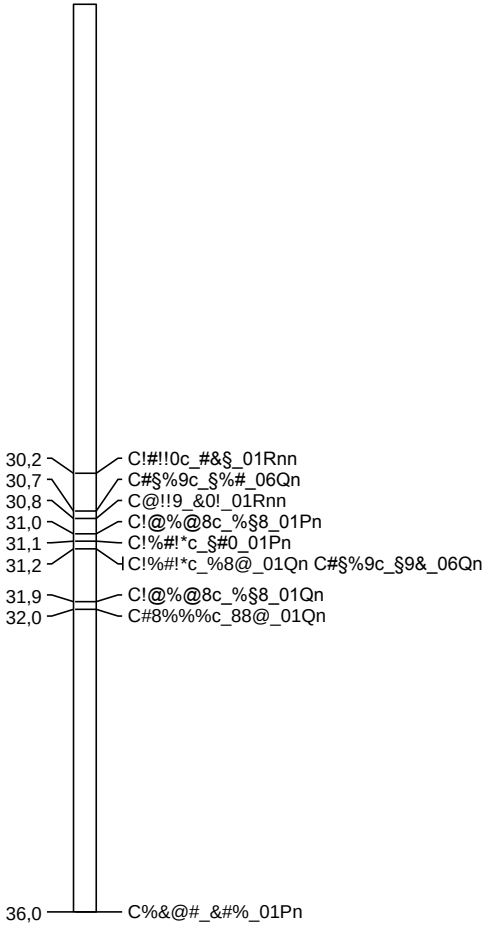
5 [1]


0,0
C#@%8_8!&_01Qn C§@*9&c_§0%_01Rnn
C!!!*!_120*_01Rnn
3,2
C!#§0!c_#0!_01Qn
3,3
C!#§0!c_#§&_01Rnn
3,4
C98c_§@8_01Pn
12,0
C##%@_!§!§_01Pn

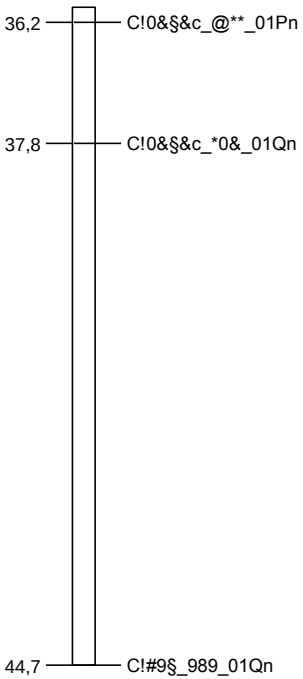
5 [2]


13,4
C§@0*0c_#@8_01Pn

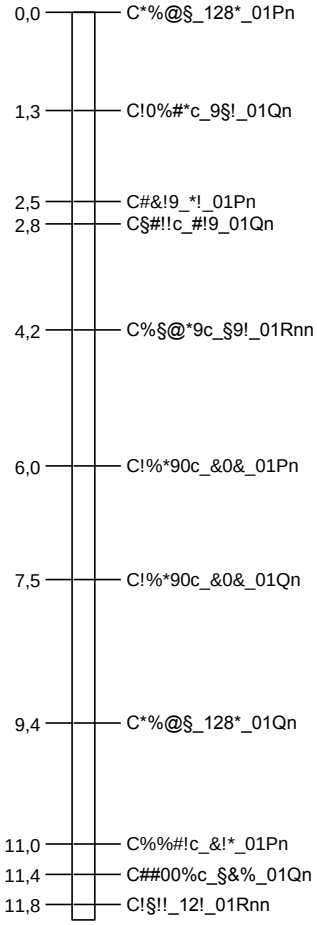
5 [3]


30,2
C!#!!0c_#&§_01Rnn
30,7
C#§%9c_§%#_06Qn
30,8
C@!!9_&0!_01Rnn
31,0
C!@%@8c_%§8_01Pn
31,1
C!%#!*c_§#0_01Pn
31,2
C!%#!*c_%8@_01Qn C#§%9c_§9&_06Qn
31,9
C!@%@8c_%§8_01Qn
32,0
C#8%%%c_88@_01Qn
36,0
C%&@#_&#%_01Pn

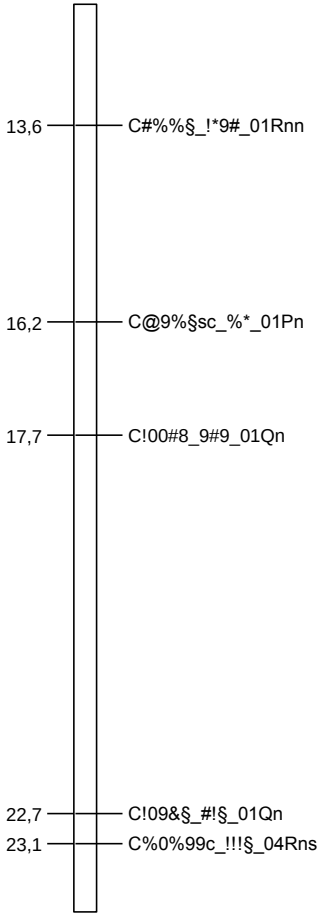
5 [4]



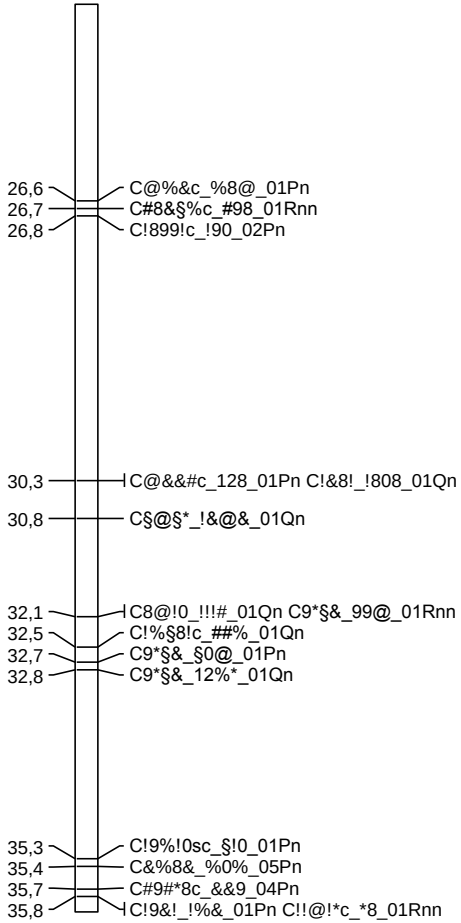
10 [1]



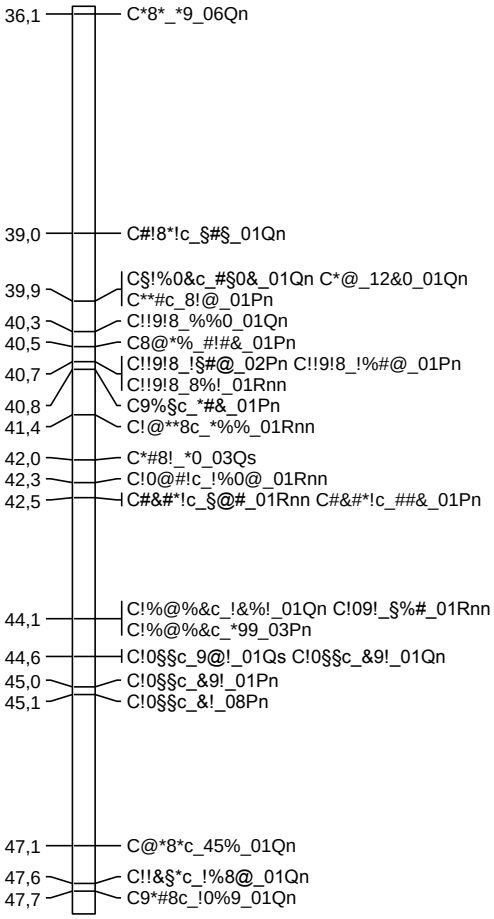
10 [2]



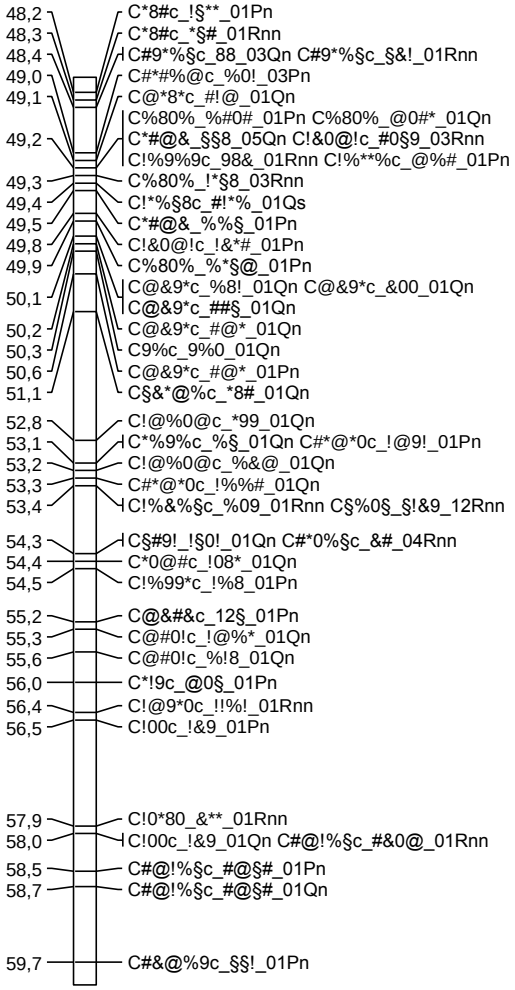
10 [3]



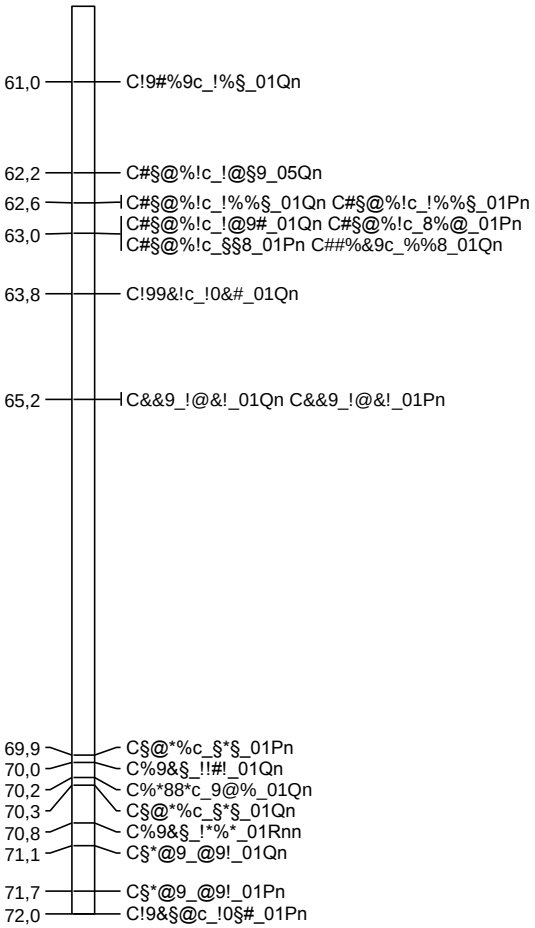
10 [4]

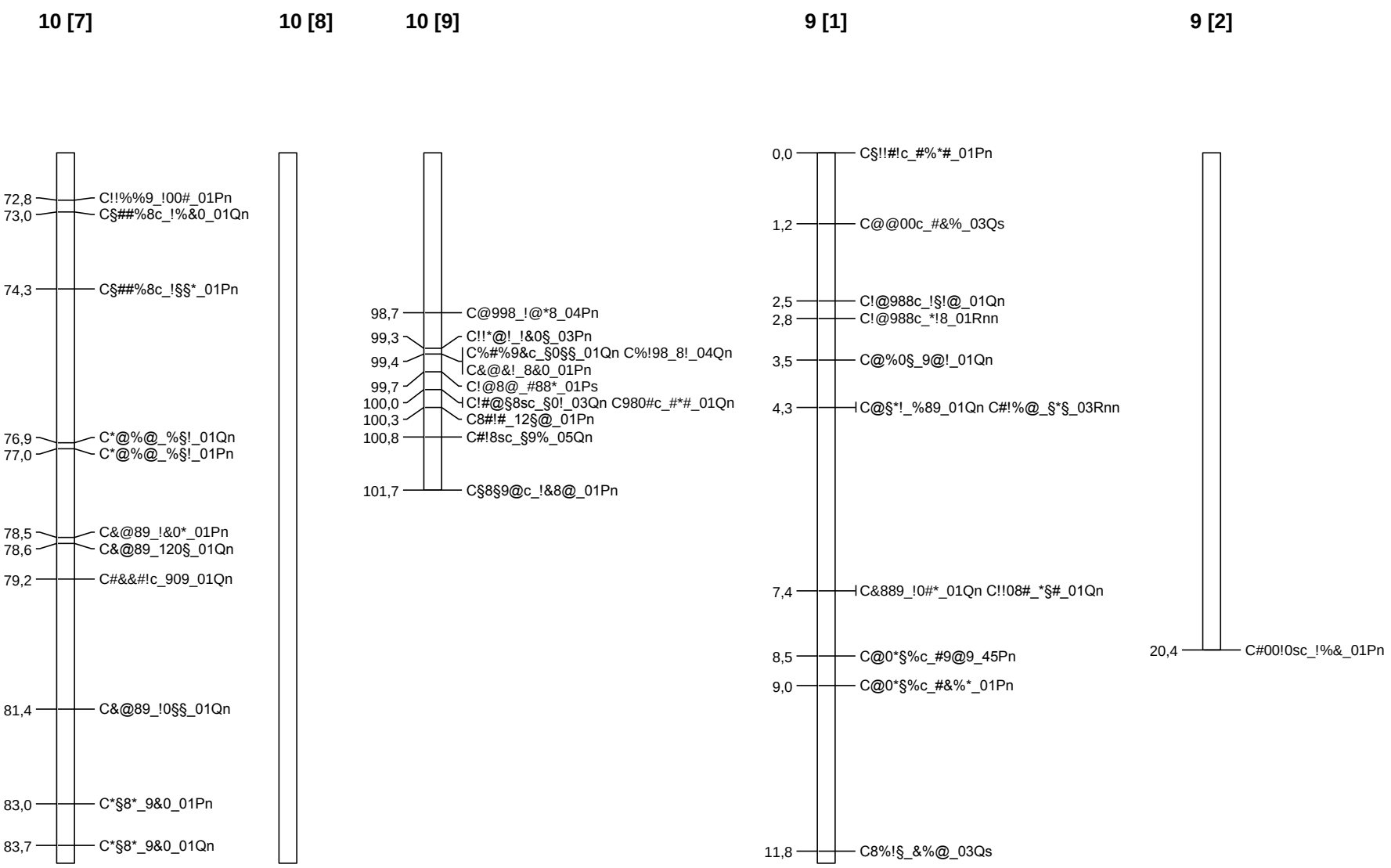

36,1 C*8*_*9_06Qn
39,0 C#!8*!c_§#§_01Qn
39,9 C§!%0&c_#§0&_01Qn C*@_12&0_01Qn
C**#c_8!@_01Pn
40,3 C!!9!8_%%0_01Qn
40,5 C8@*%_#!#&_01Pn
40,7 C!!9!8_!§#@_02Pn C!!9!8_!%#@_01Pn
C!!9!8_8%!_01Rnn
40,8 C9%§c_*#&_01Pn
41,4 C!@**8c_*%%_01Rnn
42,0 C*#8!_*0_03Qs
42,3 C!0@#!c_!%0@_01Rnn
42,5 C#&#*!c_§@#_01Rnn C#&#*!c_##&_01Pn
44,1 C!%@%&c_!&%!_01Qn C!09!_§%#_01Rnn
C!%@%&c_*99_03Pn
44,6 C!0§§c_9@!_01Qs C!0§§c_&9!_01Qn
45,0 C!0§§c_&9!_01Pn
45,1 C!0§§c_&!_08Pn
47,1 C@*8*c_45%_01Qn
47,6 C!!&§*c_!%8@_01Qn
47,7 C9*#8c_!0%9_01Qn

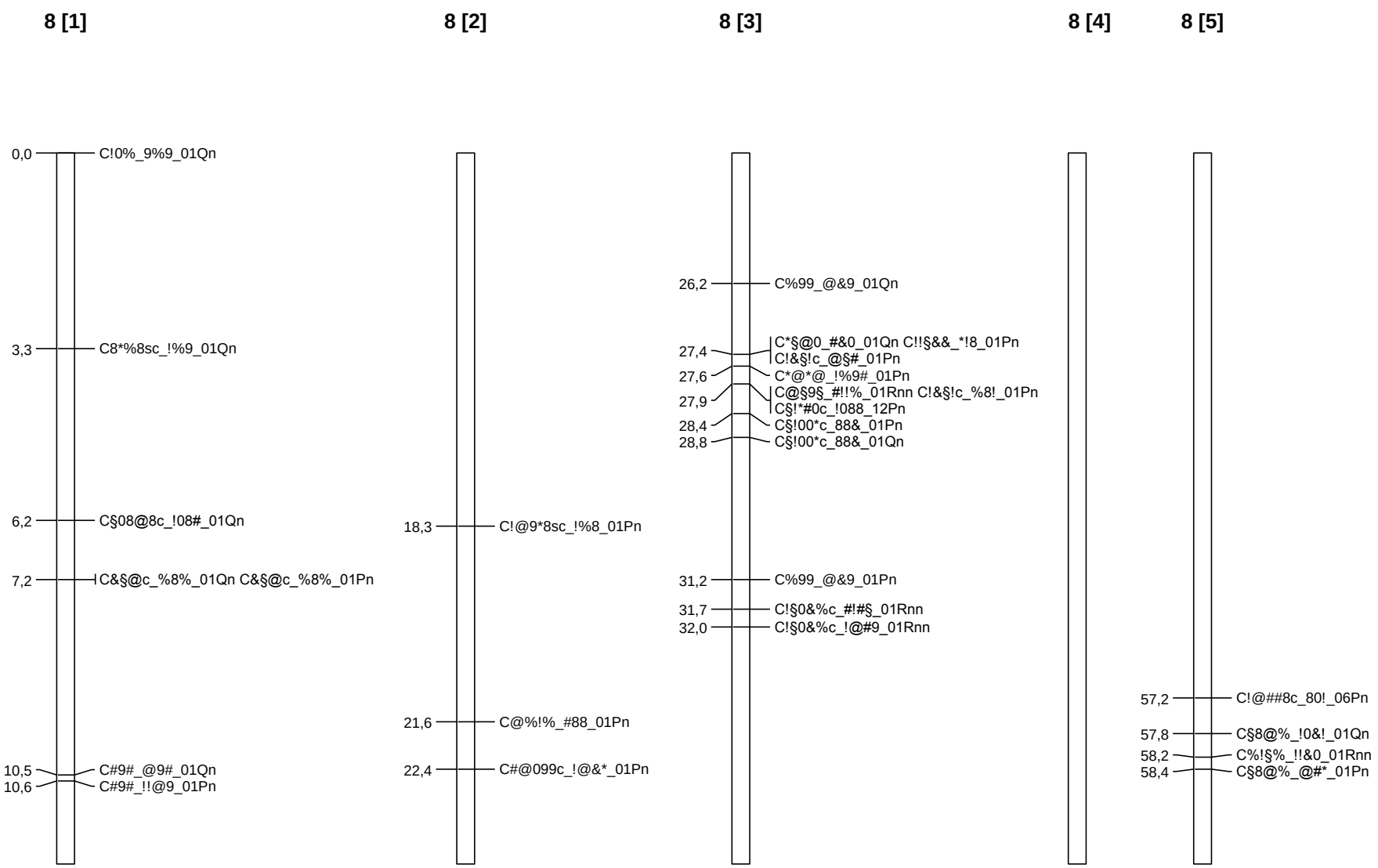
10 [5]


48,2 C*8#c_!§**_01Pn
48,3 C*8#c_*§#_01Rnn
48,4 C#9*%§c_88_03Qn C#9*%§c_§&!_01Rnn
49,0 C#*#%@c_%0!_03Pn
49,1 C@*8*c_#!@_01Qn
C%80%_%#0#_01Pn C%80%_@0#*_01Qn
49,2 C*#@&_§§8_05Qn C!&0@!c_#0§9_03Rnn
C!%9%9c_98&_01Rnn C!%**%c_@%#_01Pn
49,3 C%80%_!*§8_03Rnn
49,4 C!*%§8c_#!*%_01Qs
49,5 C*#@&_%%§_01Pn
49,8 C!&0@!c_!&*#_01Pn
49,9 C%80%_%*§@_01Pn
50,1 C@&9*c_%8!_01Qn C@&9*c_&00_01Qn
C@&9*c_##§_01Qn
50,2 C@&9*c_#@*_01Qn
50,3 C9%c_9%0_01Qn
50,6 C@&9*c_#@*_01Pn
51,1 C§&*@%c_*8#_01Qn
52,8 C!@%0@c_*99_01Qn
53,1 C*%9%c_%§_01Qn C#*@*0c_!@9!_01Pn
53,2 C!@%0@c_%&@_01Qn
53,3 C#*@*0c_!%%#_01Qn
53,4 C!%&%§c_%09_01Rnn C§%0§_§!&9_12Rnn
54,3 C§#9!_!§0!_01Qn C#*0%§c_&#_04Rnn
54,4 C*0@#c_!08*_01Qn
54,5 C!%99*c_!%8_01Pn
55,2 C@&#&c_12§_01Pn
55,3 C@#0!c_!@%*_01Qn
55,6 C@#0!c_%!8_01Qn
56,0 C*!9c_@0§_01Pn
56,4 C!@9*0c_!!%!_01Rnn
56,5 C!00c_!&9_01Pn
57,9 C!0*80_&**_01Rnn
58,0 C!00c_!&9_01Qn C#@!%§c_#&0@_01Rnn
58,5 C#@!%§c_#@§#_01Pn
58,7 C#@!%§c_#@§#_01Qn
59,7 C#&@%9c_§§!_01Pn

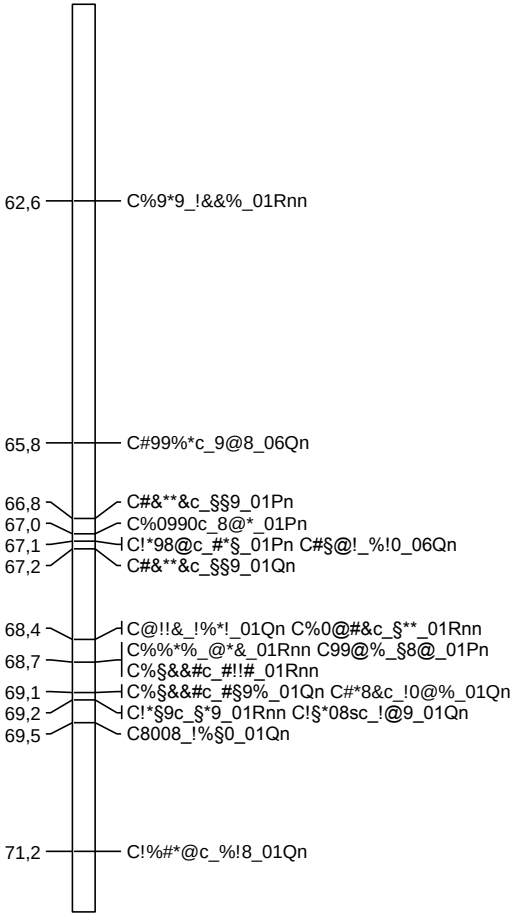
10 [6]


61,0 C!9#%9c_!%§_01Qn
62,2 C#§@%!c_!@§9_05Qn
62,6 C#§@%!c_!%%§_01Qn C#§@%!c_!%%§_01Pn
63,0 C#§@%!c_!@9#_01Qn C#§@%!c_8%@_01Pn
C#§@%!c_§§8_01Pn C##%&9c_%%8_01Qn
63,8 C!99&!c_!0&#_01Qn
65,2 C&&9_!@&!_01Qn C&&9_!@&!_01Pn
69,9 C§@*%c_§*§_01Pn
70,0 C%9&§_!!#!_01Qn
70,2 C%*88*c_9@%_01Qn
70,3 C§@*%c_§*§_01Qn
70,8 C%9&§_!*%*_01Rnn
71,1 C§*@9_@9!_01Qn
71,7 C§*@9_@9!_01Pn
72,0 C!9&§@c_!0§#_01Pn

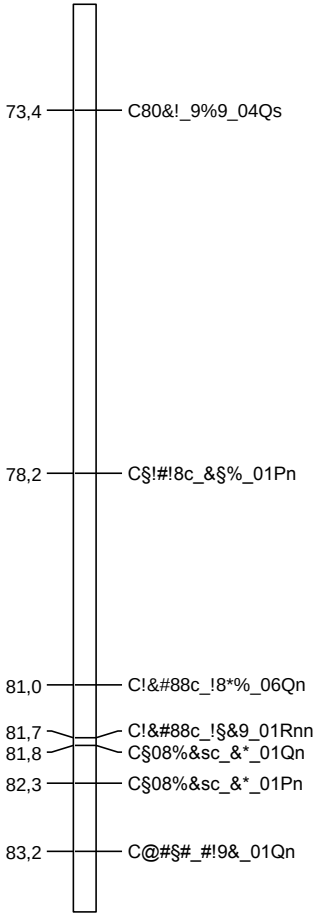

10 [7]
10 [8]
10 [9]
9 [1]
9 [2]
72,8 C!!%%9_!00#_01Pn
73,0 C§##%8c_!%&0_01Qn
74,3 C§##%8c_!§§*_01Pn
76,9 C*@%@_%§!_01Qn
77,0 C*@%@_%§!_01Pn
78,5 C&@89_!&0*_01Pn
78,6 C&@89_120§_01Qn
79,2 C#&&#!c_909_01Qn
81,4 C&@89_!0§§_01Qn
83,0 C*§8*_9&0_01Pn
83,7 C*§8*_9&0_01Qn
98,7 C@998_!@*8_04Pn
99,3 C!!*@!_!&0§_03Pn
99,4 C%#%9&c_§0§§_01Qn C%!98_8!_04Qn
C&@&!_8&0_01Pn
99,7 C!@8@_#88*_01Ps
100,0 C!#@§8sc_§0!_03Qn C980#c_#*#_01Qn
100,3 C8#!#_12§@_01Pn
100,8 C#!8sc_§9%_05Qn
101,7 C§8§9@c_!&8@_01Pn
0,0 C§!!#!c_#%*#_01Pn
1,2 C@@00c_#&%_03Qs
2,5 C!@988c_!§!@_01Qn
2,8 C!@988c_*!8_01Rnn
3,5 C@%0§_9@!_01Qn
4,3 C@§*!_%89_01Qn C#!%@_§*§_03Rnn
7,4 C&889_!0#*_01Qn C!!08#_*§#_01Qn
8,5 C@0*§%c_#9@9_45Pn
9,0 C@0*§%c_#&%*_01Pn
11,8 C8%!§_&%@_03Qs
20,4 C#00!0sc_!%&_01Pn


8 [1]
8 [2]
8 [3]
8 [4]
8 [5]
0,0
C!0%_9%9_01Qn
3,3
C8*%8sc_!%9_01Qn
6,2
C§08@8c_!08#_01Qn
7,2
C&§@c_%8%_01Qn C&§@c_%8%_01Pn
10,5
C#9#_@9#_01Qn
10,6
C#9#_!!@9_01Pn
18,3
C!@9*8sc_!%8_01Pn
21,6
C@%!%_#88_01Pn
22,4
C#@099c_!@&*_01Pn
26,2
C%99_@&9_01Qn
27,4
C*§@0_#&0_01Qn C!!§&&_*!8_01Pn
C!&§!c_@§#_01Pn
27,6
C*@*@_!%9#_01Pn
27,9
C@§9§_#!!%_01Rnn C!&§!c_%8!_01Pn
C§!*#0c_!088_12Pn
28,4
C§!00*c_88&_01Pn
28,8
C§!00*c_88&_01Qn
31,2
C%99_@&9_01Pn
31,7
C!§0&%c_#!#§_01Rnn
32,0
C!§0&%c_!@#9_01Rnn
57,2
C!@##8c_80!_06Pn
57,8
C§8@%_!0&!_01Qn
58,2
C%!§%_!!&0_01Rnn
58,4
C§8@%_@#*_01Pn

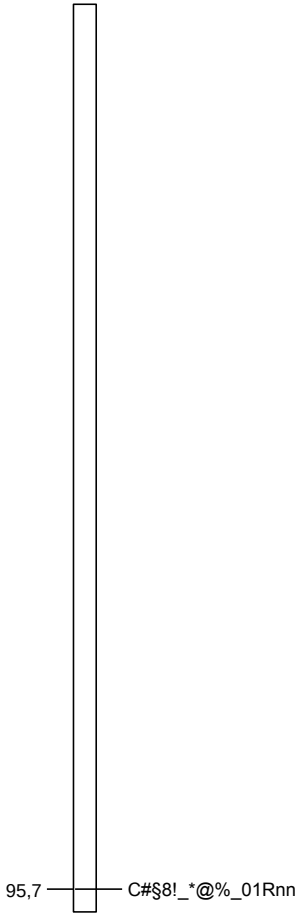
8 [6]


62,6
C%9*9_!&&%_01Rnn
65,8
C#99%*c_9@8_06Qn
66,8
C#&**&c_§§9_01Pn
67,0
C%0990c_8@*_01Pn
67,1
C!*98@c_#*§_01Pn C#§@!_%!0_06Qn
67,2
C#&**&c_§§9_01Qn
68,4
C@!!&_!%*!_01Qn C%0@#&c_§**_01Rnn
68,7
C%%*%_@*&_01Rnn C99@%_§8@_01Pn
C%§&&#c_#!!#_01Rnn
69,1
C%§&&#c_#§9%_01Qn C#*8&c_!0@%_01Qn
69,2
C!*§9c_§*9_01Rnn C!§*08sc_!@9_01Qn
69,5
C8008_!%§0_01Qn
71,2
C!%#*@c_%!8_01Qn

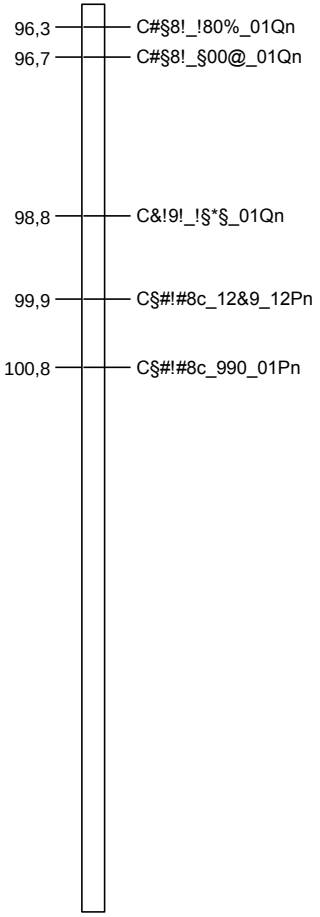
8 [7]

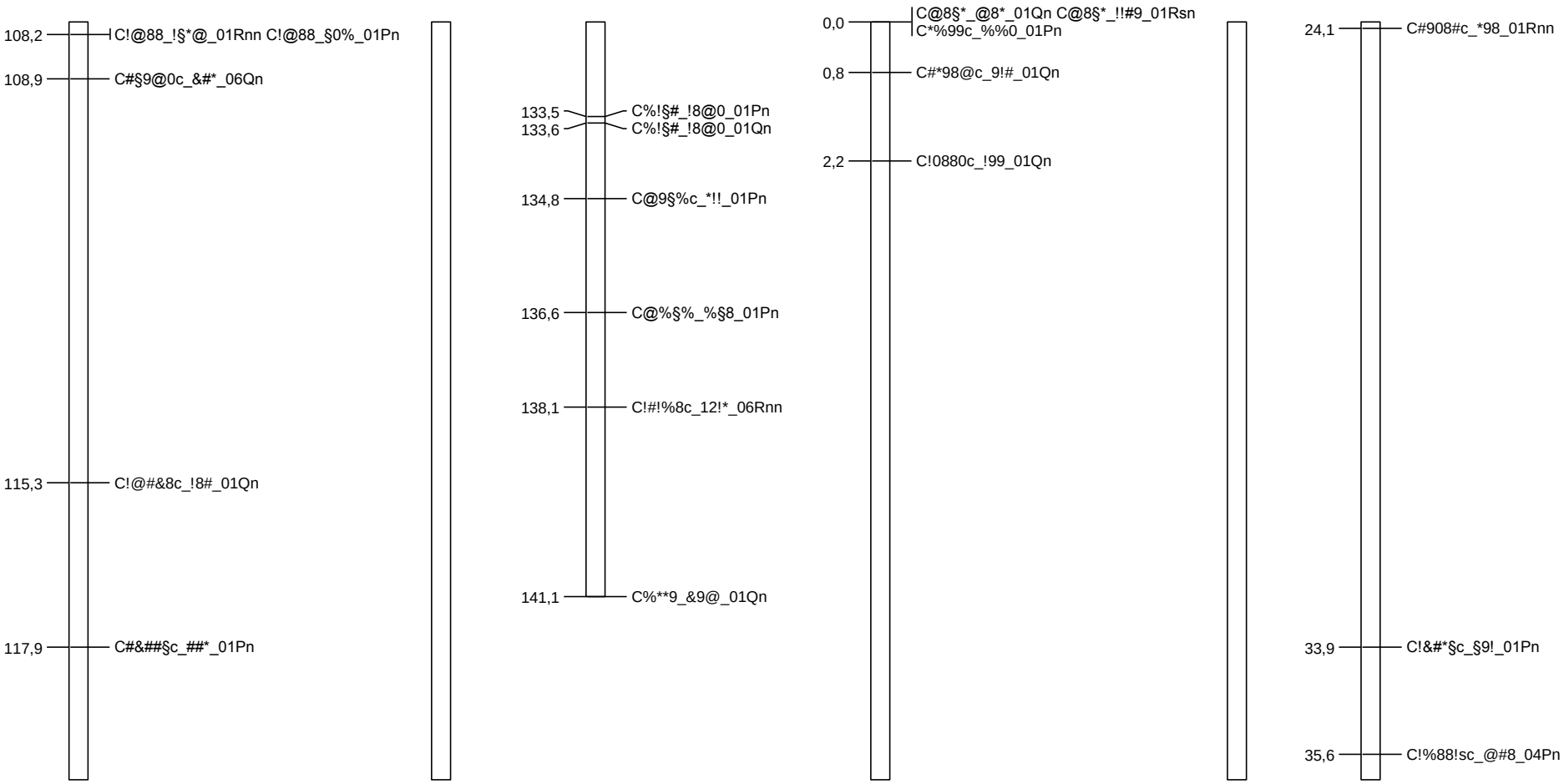

73,4
C80&!_9%9_04Qs
78,2
C§!#!8c_&§%_01Pn
81,0
C!&#88c_!8*%_06Qn
81,7
C!&#88c_!§&9_01Rnn
81,8
C§08%&sc_&*_01Qn
82,3
C§08%&sc_&*_01Pn
83,2
C@#§#_#!9&_01Qn

8 [8]

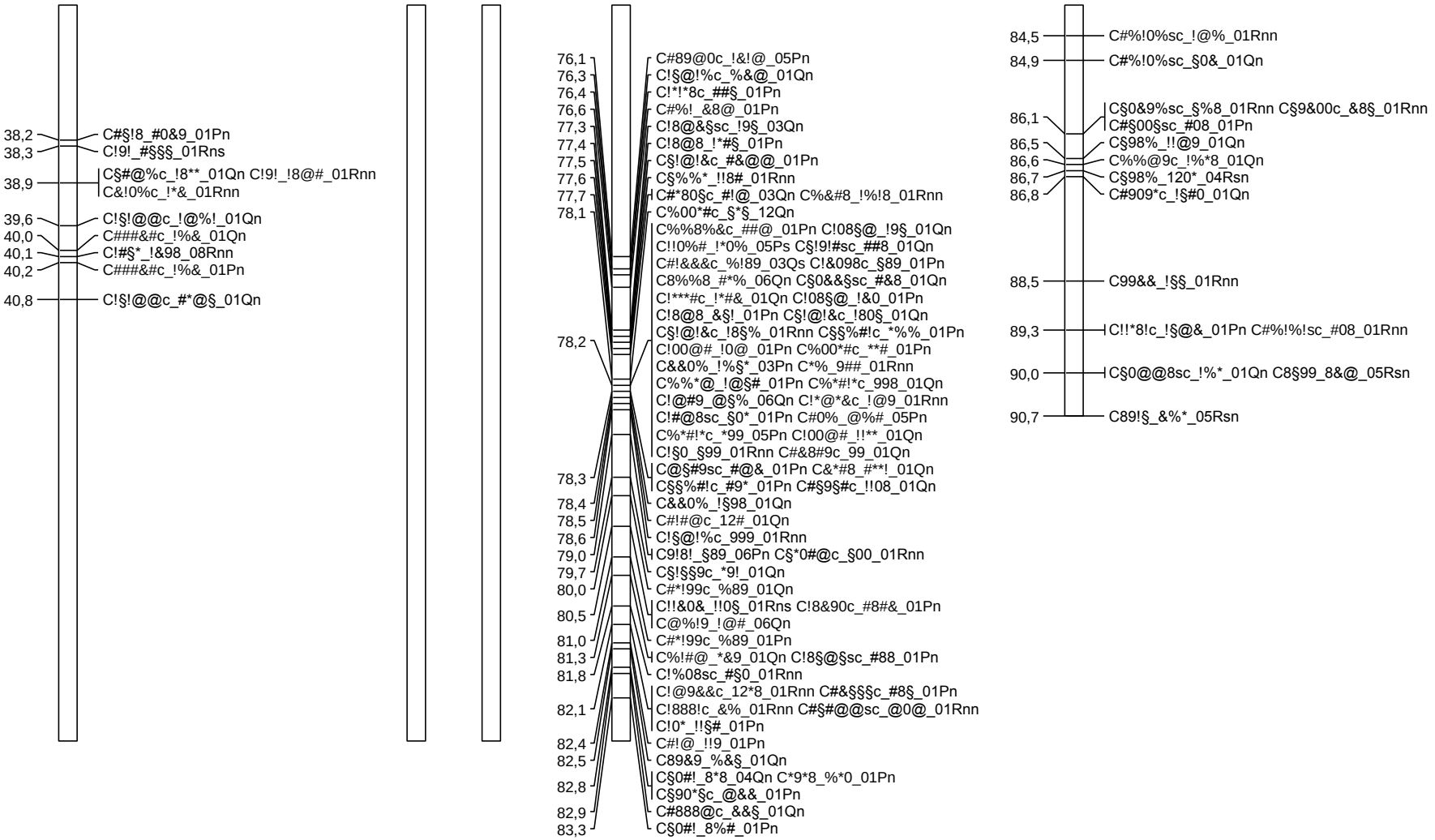

95,7
C#§8!_*@%_01Rnn

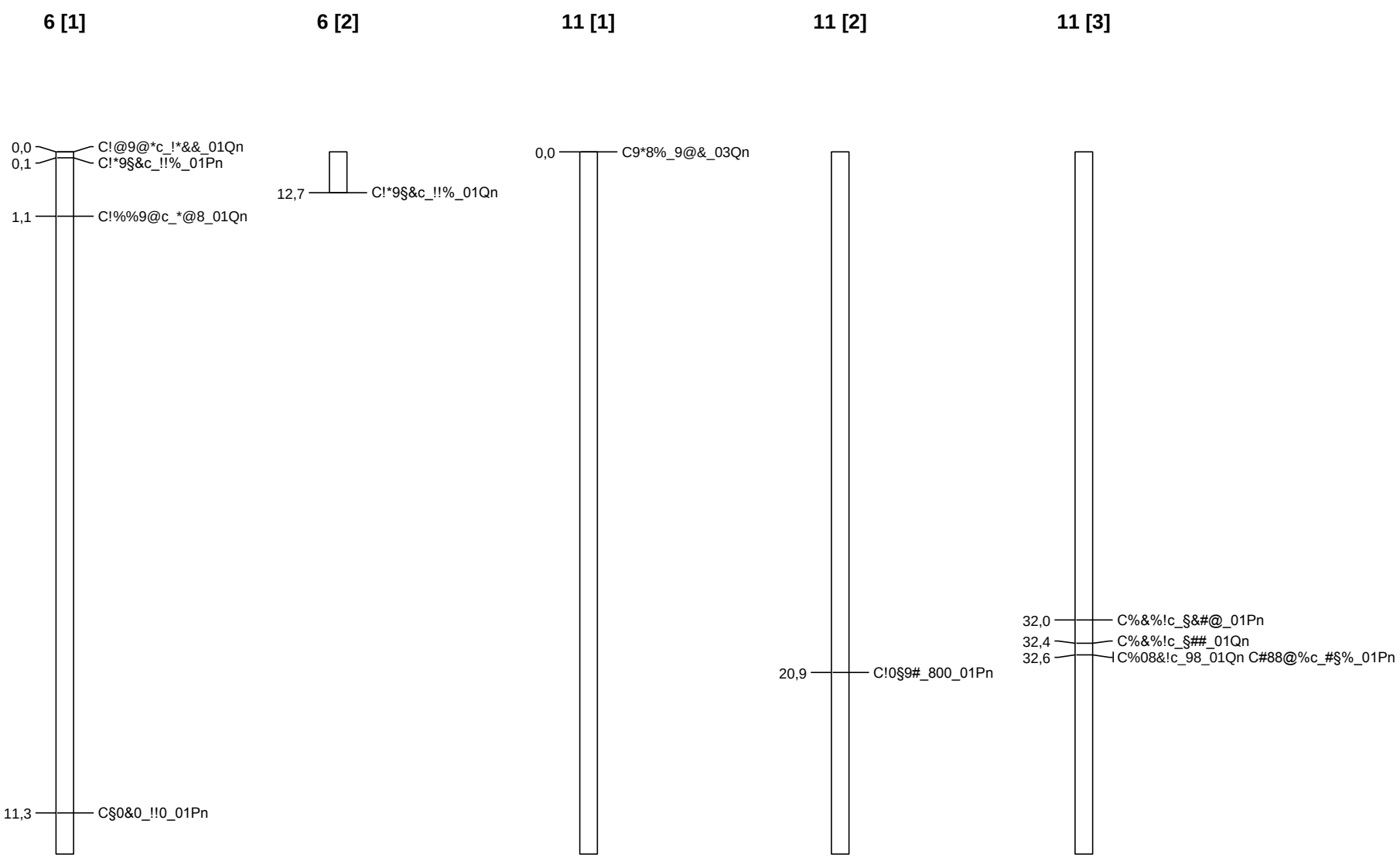
8 [9]

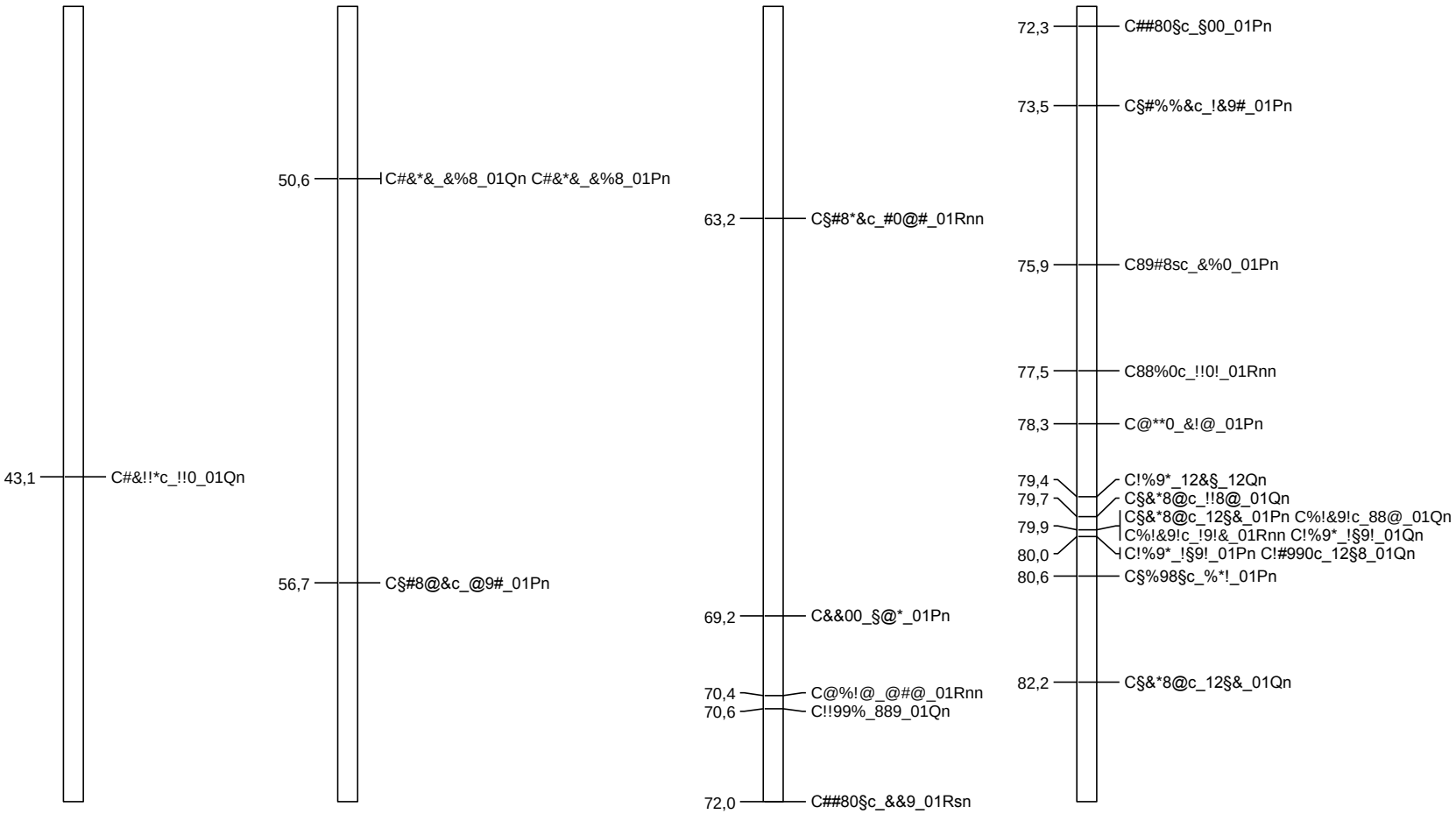

96,3
C#§8!_!80%_01Qn
96,7
C#§8!_§00@_01Qn
98,8
C&!9!_!§*§_01Qn
99,9
C§#!#8c_12&9_12Pn
100,8
C§#!#8c_990_01Pn

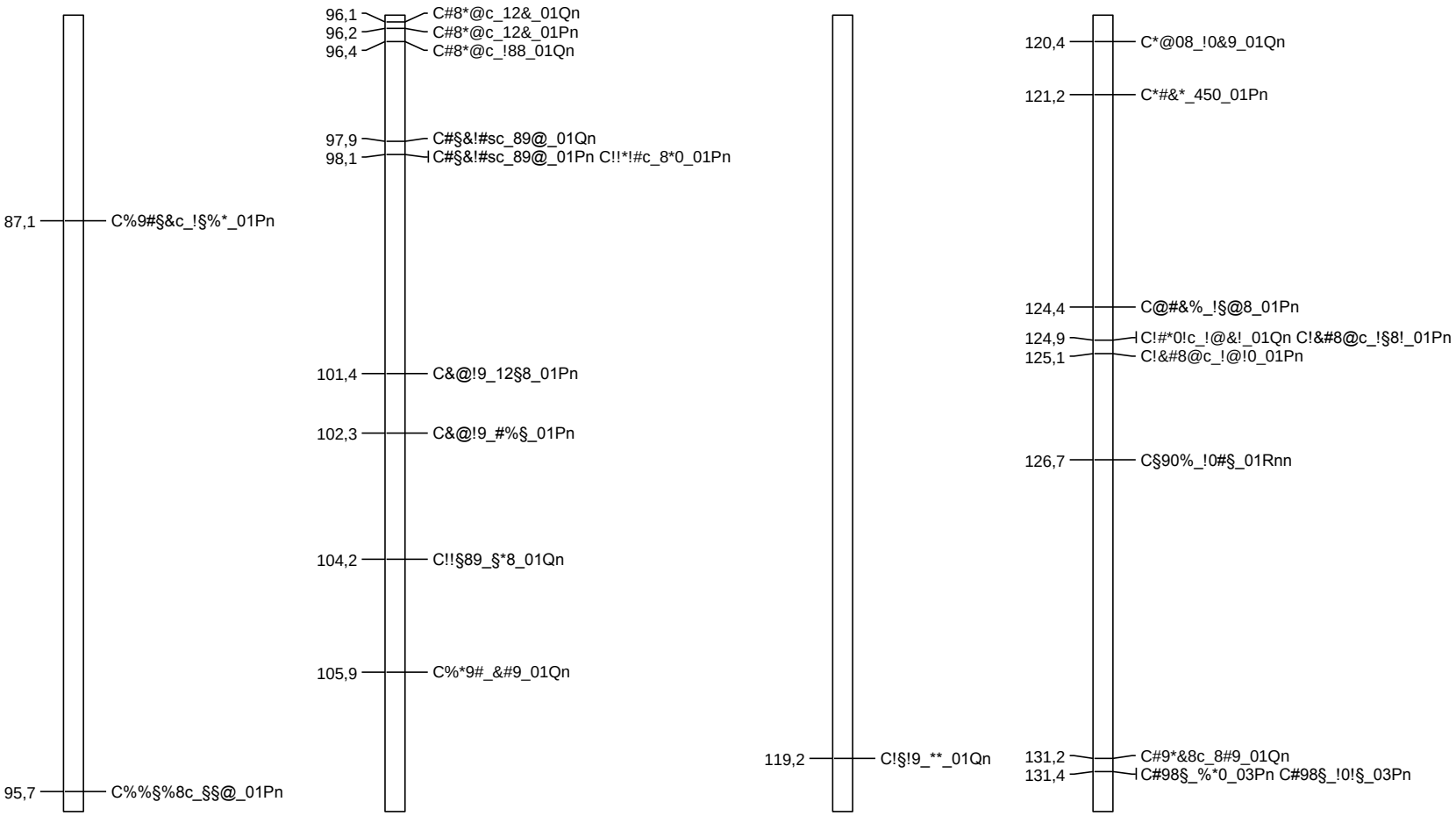

8 [10]
8 [11]
8 [12]
7 [1]
7 [2]
7 [3]
108,2
C!@88_!§*@_01Rnn C!@88_§0%_01Pn
108,9
C#§9@0c_&#*_06Qn
115,3
C!@#&8c_!8#_01Qn
117,9
C#&##§c_##*_01Pn
133,5
C%!§#_!8@0_01Pn
133,6
C%!§#_!8@0_01Qn
134,8
C@9§%c_*!!_01Pn
136,6
C@%§%_%§8_01Pn
138,1
C!#!%8c_12!*_06Rnn
141,1
C%**9_&9@_01Qn
0,0
C@8§*_@8*_01Qn C@8§*_!!#9_01Rsn
C*%99c_%%0_01Pn
0,8
C#*98@c_9!#_01Qn
2,2
C!0880c_!99_01Qn
24,1
C#908#c_*98_01Rnn
33,9
C!&#*§c_§9!_01Pn
35,6
C!%88!sc_@#8_04Pn

7 [4] **7 [5]** **7 [6]** **7 [7]** **7 [8]**

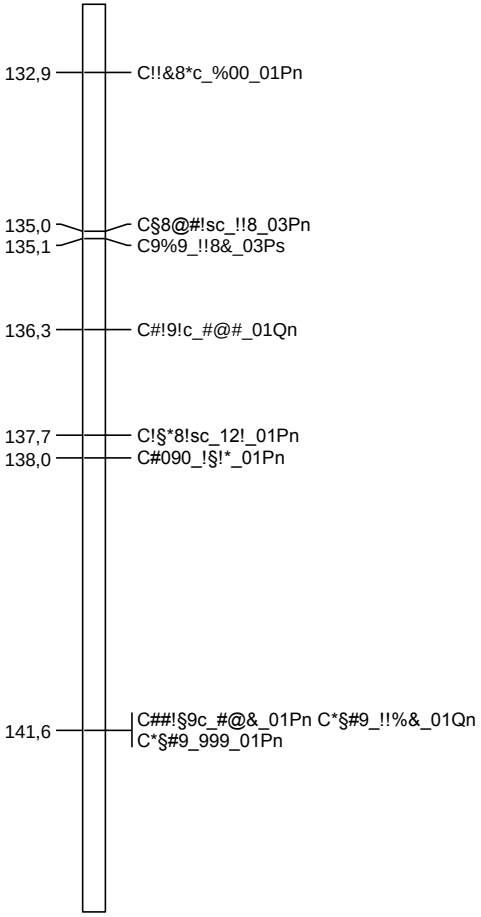

38,2 C#§!8_#0&9_01Pn
38,3 C!9!_#§§§_01Rns
38,9 C§#@%c_!8**_01Qn C!9!_!8@#_01Rnn
C&!0%c_!*&_01Rnn
39,6 C!§!@@c_!@%!_01Qn
40,0 C###&#c_!%&_01Qn
40,1 C!#§*_!&98_08Rnn
40,2 C###&#c_!%&_01Pn
40,8 C!§!@@c_#*@§_01Qn
76,1 C#89@0c_!&!@_05Pn
76,3 C!§@!%c_%&@_01Qn
76,4 C!*!*8c_##§_01Pn
76,6 C#%!_&8@_01Pn
77,3 C!8@&§sc_!9§_03Qn
77,4 C!8@8_!*#§_01Pn
77,5 C§!@!&c_#&@@_01Pn
77,6 C§%%*_!!8#_01Rnn
77,7 C#*80§c_#!@_03Qn C%&#8_!%!8_01Rnn
78,1 C%00*#c_§*§_12Qn
C%%8%&c_##@_01Pn C!08§@_!9§_01Qn
C!!0%#_!*0%_05Ps C§!9!#sc_##8_01Qn
C#!&&&c_%!89_03Qs C!&098c_§89_01Pn
C8%%8_#*%_06Qn C§0&&§sc_#&8_01Qn
C!***#c_!*#&_01Qn C!08§@_!&0_01Pn
C!8@8_&§!_01Pn C§!@!&c_!80§_01Qn
78,2 C§!@!&c_!8§%_01Rnn C§§%#!c_*%%_01Pn
C!00@#_!0@_01Pn C%00*#c_**#_01Pn
C&&0%_!%§*_03Pn C*%_9##_01Rnn
C%%*@_!@§#_01Pn C%*#!*c_998_01Qn
C!@#9_@§%_06Qn C!*@*&c_!@9_01Rnn
C!#@8sc_§0*_01Pn C#0%_@%#_05Pn
C%*#!*c_*99_05Pn C!00@#_!!**_01Qn
C!§0_§99_01Rnn C#&8#9c_99_01Qn
78,3 C@§#9sc_#@&_01Pn C&*#8_#**!_01Qn
C§§%#!c_#9*_01Pn C#§9§#c_!!08_01Qn
78,4 C&&0%_!§98_01Qn
78,5 C#!#@c_12#_01Qn
78,6 C!§@!%c_999_01Rnn
79,0 C9!8!_§89_06Pn C§*0#@c_§00_01Rnn
79,7 C§!§§9c_*9!_01Qn
80,0 C#*!99c_%89_01Qn
80,5 C!!&0&_!!0§_01Rns C!8&90c_#8#&_01Pn
C@%!9_!@#_06Qn
81,0 C#*!99c_%89_01Pn
81,3 C%!#@_*&9_01Qn C!8§@§sc_#88_01Pn
81,8 C!%08sc_#§0_01Rnn
C!@9&&c_12*8_01Rnn C#&§§§c_#8§_01Pn
82,1 C!888!c_&%_01Rnn C#§#@@sc_@0@_01Rnn
C!0*_!!§#_01Pn
82,4 C#!@_!!9_01Pn
82,5 C89&9_%&§_01Qn
82,8 C§0#!_8*8_04Qn C*9*8_%*0_01Pn
C§90*§c_@&&_01Pn
82,9 C#888@c_&&§_01Qn
83,3 C§0#!_8%#_01Pn
84,5 C#%!0%sc_!@%_01Rnn
84,9 C#%!0%sc_§0&_01Qn
86,1 C§0&9%sc_§%8_01Rnn C§9&00c_&8§_01Rnn
C#§00§sc_#08_01Pn
86,5 C§98%_!!@9_01Qn
86,6 C%%@9c_!%*8_01Qn
86,7 C§98%_120*_04Rsn
86,8 C#909*c_!§#0_01Qn
88,5 C99&&_!§§_01Rnn
89,3 C!!*8!c_!§@&_01Pn C#%!%!sc_#08_01Rnn
90,0 C§0@@8sc_!%*_01Qn C8§99_8&@_05Rsn
90,7 C89!§_&%*_05Rsn


6 [1]
6 [2]
11 [1]
11 [2]
11 [3]
0,0
C!@9@*c_!*&&_01Qn
0,1
C!*9§&c_!!%_01Pn
1,1
C!%%9@c_*@8_01Qn
11,3
C§0&0_!!0_01Pn
12,7
C!*9§&c_!!%_01Qn
0,0
C9*8%_9@&_03Qn
20,9
C!0§9#_800_01Pn
32,0
C%&%!c_§&#@_01Pn
32,4
C%&%!c_§##_01Qn
32,6
C%08&!c_98_01Qn C#88@%c_#§%_01Pn

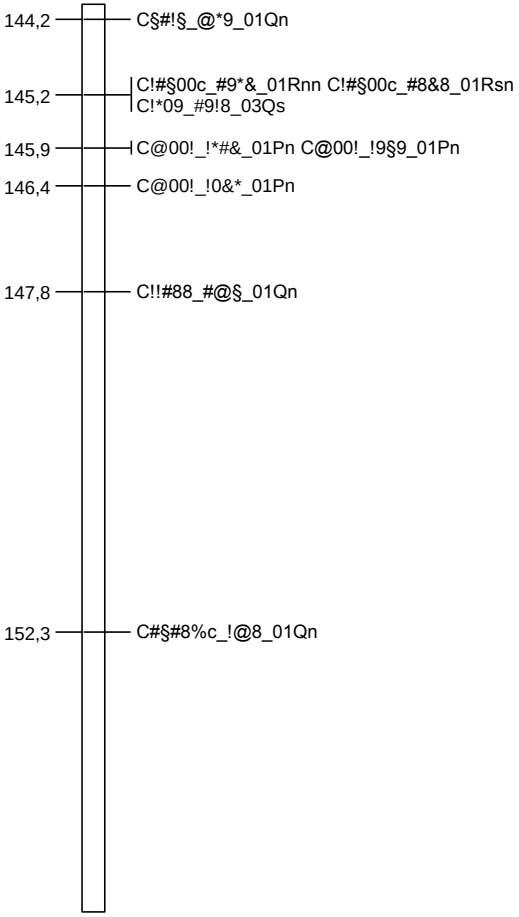

11 [4]
11 [5]
11 [6]
11 [7]
43,1
C#&!!*c_!!0_01Qn
50,6
C#&*&_&%8_01Qn C#&*&_&%8_01Pn
56,7
C§#8@&c_@9#_01Pn
63,2
C§#8*&c_#0@#_01Rnn
69,2
C&&00_§@*_01Pn
70,4
C@%!@_@#@_01Rnn
70,6
C!!99%_889_01Qn
72,0
C##80§c_&&9_01Rsn
72,3
C##80§c_§00_01Pn
73,5
C§#%%&c_!&9#_01Pn
75,9
C89#8sc_&%0_01Pn
77,5
C88%0c_!!0!_01Rnn
78,3
C@**0_&!@_01Pn
79,4
C!%9*_12&§_12Qn
79,7
C§&*8@c_!!8@_01Qn
79,9
C§&*8@c_12§&_01Pn C%!&9!c_88@_01Qn
C%!&9!c_!9!&_01Rnn C!%9*_!§9!_01Qn
80,0
C!%9*_!§9!_01Pn C!#990c_12§8_01Qn
80,6
C§%98§c_%*!_01Pn
82,2
C§&*8@c_12§&_01Qn


11 [8]
11 [9]
11 [10]
11 [11]
87,1
C%9#§&c_!§%*_01Pn
95,7
C%%§%8c_§§@_01Pn
96,1
C#8*@c_12&_01Qn
96,2
C#8*@c_12&_01Pn
96,4
C#8*@c_!88_01Qn
97,9
C#§&!#sc_89@_01Qn
98,1
C#§&!#sc_89@_01Pn C!!*!#c_8*0_01Pn
101,4
C&@!9_12§8_01Pn
102,3
C&@!9_#%§_01Pn
104,2
C!!§89_§*8_01Qn
105,9
C%*9#_&#9_01Qn
119,2
C!§!9_**_01Qn
120,4
C*@08_!0&9_01Qn
121,2
C*#&*_450_01Pn
124,4
C@#&%_!§@8_01Pn
124,9
C!#*0!c_!@&!_01Qn C!&#8@c_!§8!_01Pn
125,1
C!&#8@c_!@!0_01Pn
126,7
C§90%_!0#§_01Rnn
131,2
C#9*&8c_8#9_01Qn
131,4
C#98§_%*0_03Pn C#98§_!0!§_03Pn

11 [12]



11 [13]



11 [14]