universität wien

MASTERARBEIT / MASTER'S THESIS

Titel der Masterarbeit / Title of the Master's Thesis

„Degree of inbreeding in populations of *Arabidopsis halleri* and its correlation with metalliferous soils"

verfasst von / submitted by

Katharina Schneider, BSc

angestrebter akademischer Grad / in partial fulfilment of the requirements for the degree of

Master of Science (MSc)

Wien, 2018 / Vienna 2018

Studienkennzahl lt. Studienblatt / degree programme code as it appears on the student record sheet:	A 066 829
Studienrichtung lt. Studienblatt / degree programme as it appears on the student record sheet:	Masterstudium Evolutionsbiologie
Betreut von / Supervisor:	Univ.-Prof. Dr. Christian Schlötterer Veterinärmedizinische Universität Wien

Acknowledgements

I would like to thank my supervisor Univ.-Prof. Dr.rer.nat. Christian Schlötterer for making this master thesis possible and for his support during the project. I also would like to give a special thanks to Dipl.-Biol. Viola Nolte, who supervised me in the laboratory and always supported me with advice and expertise.

Furthermore, I am deeply grateful for the help of DI Dr.nat.techn. Marlies Dolezal, MSc who taught me scripting and statistics and supervised me during the data analysis. Special thanks also to Dr.rer.nat. Lukas Endler for his scripts and support.

I also would like to acknowledge the collaborating group of Prof. Dr. Katja Tielbörger and Dr. Michal Gruntman at the University of Tübingen. A special thanks to Anubhav Mohiley, MSc for his supportive correspondence concerning the field work of this project.

Finally, I would like to thank my mentor at the University Vienna, Univ.-Prof. Mag. Dr. Christian Lexer, for his advice and support.

Table of contents

Abstract

Heavy metal pollution in soils has an important impact on the physiology and population structure of plants. Due to differences in the tolerance ranges towards cadmium, only a restricted number of individuals can colonize a specific site. It was assumed, that individuals on metalliferous soils will predominantly propagate clonally to ensure the establishment and maintenance of a population. Therefore, an effect of cadmium on the population structure was expected. In this master thesis, the influence of cadmium on the level of relatedness and the degree of inbreeding was investigated in natural populations of *Arabidopsis halleri*. A representative proportion of the whole genome was examined in 260 individuals from 10 populations by applying a two-restriction-enzyme sequencing approach. No clones could be found, neither on metalliferous nor on non-metalliferous sites. Clonal propagation as the predominant reproduction mode on metal-contaminated soils could thus be refuted. Furthermore, no impact of cadmium on the inbreeding coefficient was observed in an overall population comparison. This could indicate a higher flexibility of the metal tolerance levels than expected. The examined populations differed significantly in their inbreeding coefficients. These differences could imply a variation in the strength of the self-incompatibility system and question the definition of *A. halleri* as a self-incompatible and obligatory outcrossing species.

Zusammenfassung

Die Belastung der Böden mit Schwermetallen stellt einen Stressor für Pflanzen in Bezug auf Physiologie und Populationsstruktur dar. Aufgrund von Unterschieden im Toleranzbereich gegenüber Cadmium, kann nur eine limitierte Anzahl an Pflanzenindividuen einen Standort besiedeln. Es wurde angenommen, dass sich Individuen auf metallbelasteten Böden vorherrschend klonal vermehren um eine Population zu etablieren und ihren Erhalt sicherzustellen. Davon ausgehend wurde ein Einfluss von Cadmium auf die Struktur einer Population vermutet. In dieser Masterarbeit wurde der Effekt von Cadmium auf die Verwandtschaftsnähe und das Ausmaß an Inzucht in natürlichen Populationen von *Arabidopsis halleri* überprüft. Unter Verwendung von zwei Restriktionsenzymen wurde ein repräsentativer Anteil des Genoms in 260 Individuen aus 10 Populationen untersucht. Weder an metallbelasteten, noch unbelasteten Standorten konnte klonale Vermehrung nachgewiesen werden. Dadurch konnte klonale Reproduktion als primäre Vermehrungsmethode auf metallbelasteten Böden ausgeschlossen werden. Cadmium hatte darüber hinaus keinen Einfluss auf das Ausmaß an Inzucht. Dies könnte auf flexiblere Metalltoleranzniveaus hinweisen als angenommen wurde. Die untersuchten Populationen unterschieden sich signifikant in ihren Inzuchtkoeffizienten. Diese Unterschiede könnten auf Variationen im Ausmaß der Selbstinkompatibilität hindeuten und dadurch eine neue Charakterisierung des Reproduktionsmodus von *A. halleri* erfordern.

1. Introduction

1.1. Heavy metal pollution in soils

Besides natural deposits of zinc (Zn), iron (Fe), lead (Pb) and cadmium (Cd) (etc.), anthropogenic activities lead to an additional introduction of heavy metals into the soil (Chiang et al. 2006, Chibuike and Obiora 2014, Wuana and Okieimen 2011). An increase in the concentration of heavy metals by mining, fertilizers, sewage waste and other industrial activities (Zhang et al. 2010, Halim et al. 2003, Chiang et al. 2006, Singh et al. 2016, Yang et al. 2005) represents a stressor for many plant species and thus influences their development and evolution (e.g. Chatterjee and Chatterjee 2000, Oancea et al. 2005).

As an immediate effect changes in the physiological processes (Singh et al. 2016, Dubey 2011, Hossain and Fujita, 2009, 2011, Hossain et al. 2009, 2010, 2012, Sandalio et al. 2001, Sharma and Dietz 2009, Tan et al. 2010, Villiers et al. 2011, Bert et al. 2000) and a reduction in growth and consequently in yield (i.a. Sandalio et al. 2001, Onacea et al. 2005, Singh et al. 2016 Keunen et al. 2011) were described for instance in cauliflower (Chatterjee and Chatterjee, 2000). One reason for the toxicity of heavy metals consists in the competition and displacement of nutrients with non-essential metal cations (Hall 2002, Hossain et al. 2012, Sharma and Dietz 2009, Singh et al. 2016). This can result in an alternation or even inhibition of enzymes (Hirata et al. 2005, Hall 2002, Hossain et al. 2012, Sharma and Dietz 2009, Singh et al. 2016). Furthermore, the production of reactive oxygen species (ROS) is accelerated under the influence of heavy metals (Hossain et al. 2010, Navari-Izzo 1998, Romero-Puertas et al. 2002, Barconi et al. 2011, Singh et al. 2016, Hirata et al. 2005). ROS can lead to oxidative damages causing modifications of lipids, proteins, enzymes and nucleic acids (Hossain et al. 2010, Navari-Izzo 1998, Romero-Puertas et al. 2002, Barconi et al. 2011, Singh et al. 2016, Hirata et al. 2005).

Some plant species, like *Arabidopsis halleri*, adapted to these conditions and developed mechanisms to tolerate heavy metals in the soil (Baker 1981). According to Baker heavy metal tolerance can be achieved by exclusion, inclusion or accumulation (Baker 1981, Chibuike and Obiora 2014). In species that tolerate heavy metals via exclusion, a constant concentration in the shoots is maintained through a retention of metals in the roots and a restricted transport into aerial structures (Baker 1981, Chibuike and Obiora 2014). Tolerance through inclusion is achieved by a regulated metal uptake and transport (Baker 1981, Chibuike and Obiora 2014). This results in a shoot concentration that is equal to the level in the surrounding soil (Baker 1981, Chibuike and Obiora 2014). In accumulating species metal ions are aggregated in the shoots and roots, independent of the soils concentration (Baker 1981, Chibuike and Obiora 2014).

1.2. *Arabidopsis halleri* (Linnaeus) O'Kane & Al-Shehbaz

Arabidopsis thaliana is a popular model organism for evolution and development of plants. Its popularity as a research tool mainly bases on a short generation time (6 weeks) and simple cultivation (TAIR 2017; Clauss and Koch 2006). *A. thaliana* possesses a small genome (125Mb; TAIR 2017), which was fully sequenced in 2000 (Pruitt et al. 2003, Koornneef et al. 2004, Tonsor et al. 2005 as cited in Clauss and Koch 2006). Furthermore, many mutant lines are available (TAIR 2017).

A. thaliana has a very efficient mode of reproduction. Besides vegetative and sexual propagation *A. thaliana* increased its reproductive success by overcoming self-incompatibility (Shimizu and Tsuchimatsu 2015).

A close relative of *A. thaliana* is the pseudometallophyte *Arabidopsis halleri* (L.) O'Kane and Al-Shehbaz. *A. halleri* is a herbaceous member of the family *Brassicaceae* which can reach a height between 20-65 cm (Al-Shehbaz and O'Kane 2002). Its blossoms consist of white to purplish petals (Al-Shehbaz and O'Kane 2002).

The perennial species is distributed from France to Japan and Taiwan (Mitchell-Olds 2001, O'Kane and Al-Shehbaz 1997, Al-Shehbaz and O'Kane 2002, Kolník and Marhold, 2006, Kubota and Takenaka 2003) where it colonizes mesic environments like grassy landscapes and forest margins (Al-Shehbaz and O'Kane 2002, Mitchell-Olds 2001). Furthermore it can be found in mountainous areas at altitudes above 600m (Al-Shehbaz and O'Kane 2002)

A. halleri diverged from *A. thaliana* about 3 to 5.8 mya (Koch et al. 2000, Clauss and Koch 2006, Al-Shehbaz and O'Kane 2002). The two species share on average 94% of their nucleotide sequence within coding regions (Vess et al. as cited in Weber et al. 2004, Becher et al. 2004). However, *Arabidopsis halleri* differs in the number of chromosomes (2n=16, *A. thaliana*: 2n=10) and genome size (Briskine et al. 2016, Johnston et al. 2005, Kolník and Marhold 2006). With an estimated size of 250Mb

its genome is approximately 1.4 times larger than the genome of *A. thaliana* (Briskine et al. 2016, Johnston et al. 2005, Kolník and Marhold 2006).

A. halleri is self-incompatible (Clauss and Koch, 2006; Llaurens et al. 2008). Other than in *A. thaliana* sporophytic self-incompatibility prohibits the fertilization with pollen showing a similar genotype at the S-locus (Castric and Vekemans 2004, Kusaba et al. 2002, Clauss und Koch 2006) which facilitates the maintenance of genetic diversity (Castric et al. 2008, Roux et al. 2011).

Due to this self-incompatibility *A. halleri* is strictly outcrossing (Clauss and Koch 2006, Llaurens et al. 2008). Butterflies, bees and syrphids are described as pollinators (Clauss and Koch 2006), and the seeds are spread by wind (anemochory) or water (hydrochory) after being released from the dry silique (Dinneny and Yanofsky 2004, Al-Shehbaz and O'Kane 2002).

In addition to sexual reproduction *A. halleri* also propagates clonally by stolons (Al-Shehbaz and O'Kane 2002, Clauss and Koch 2006). In regions with suboptimal conditions (e.g. metalliferous soils) the ability of vegetative clonal propagation can constitute a selective advantage (Gaudeul et al. 2007, Sun et al. 2001, Williams 1975, Sun et al. 2001 as cited in Gaudeul et al. 2007, Salemaa et al. 1999, Salemaa and Sievanen 2002, Bizoux and Mahy 2007). It ensures the maintenance of a population when few or no mating partners are present (Gaudeul et al. 2007, Sun et al. 2001, Williams 1975, Sun et al. 2001 as cited in Gaudeul et al. 2007, Salemaa et al. 1999, Salemaa and Sievanen 2002, Bizoux and Mahy 2007).

Apart from these characteristics *A. halleri* is of major scientific interest due to its ability to tolerate high amounts of heavy metals (Bert et al. 2000, Van Rossum et al. 2004, Clauss and Koch 2006). It is described to be tolerant to high soil contents of zinc (Bert et al. 2000, Van Rossum et al. 2004), cadmium and lead (Van Rossum et al. 2004). Besides, *A. halleri* not only tolerates but also accumulates essential and non-essential metals (hyperaccumulation), especially zinc and cadmium (Van Rossu et al. 2004). This capacity to hyperaccumulate heavy metals made *A. halleri* become of huge interest for the field of phytoremediation (van Rossum et al. 2004, Kubota and Takenaka 2003, Cunningham et al. 1997).

1.3. Metal tolerance and hyperaccumulation in *Arabidopsis halleri*

Metal tolerant plants reduce the toxic effects of heavy metals by chelation and sequestration or by transporting them into the extracellular spaces (Singh et al. 2016). Several proteins (e.g. Phytochelatine and Metallothioneine) are related to these mechanisms (i.a. Guo et al. 2008, Zimeri et al. 2005, Clemens et al. 1999, MC et al. 1998, Singh et al. 2016, Yang et al. 2005). Four classes of transporters, however, are always reported to be associated with metal tolerance and hyperaccumulation: HEAVY METAL ATPASEs (HMAs) are responsible for metal homeostasis and tolerance in plants by transporting several essential and non-essential heavy metals (Williams 2000, Yang et al. 2005). Here, HMA4 is described to be the determining gene for Cd tolerance and accumulation in *A. halleri* (Courbot et al. 2007, Meyer et al. 2015). METAL TOLERANCE PROTEINS (MTPs) regulate the uptake and efflux of heavy metals (Yang et al. 2005, Zimeri et al. 2005, Williams et al. 2000, Grennan 2009, Singh et al. 2016). NATRUAL RESISTANCE ASSOCIATED MACROPHAGE PROTEINS (NRAMPs) are described to be responsible for the uptake and transport of metals with an ionic valence of two, like iron or cadmium (Thomine et al. 2000, Yang et al. 2005). The last class of proteins commonly connected to heavy metal tolerance are ZINK-IRON PERMEASEs (ZIPs) which also transport several metal ions (Weber et al. 2004, Vert et al. 2002, Pence et al. 2000, Guerinot 2000, Yang et al. 2005).

Tolerance and hyperaccumulation of Cd and Zn are supposed to concur with duplications and consequently a higher expression of the genes HMA4 and MTP1 (Briskine 2016, Courbot 2007, Willems 2007, Hanikenne et al. 2008). It is assumed that HMA4 duplicated at the divergence of *A. halleri* and *A. lyrata* 337 kya - 2.5 mya (Roux et al. 2011, Castric et al. 2008). This is supported by the fact, that the latter neither possesses tolerance against heavy metals nor the ability to hyperaccumulate them (Clauss and Koch 2006). An increased copy number of MTP1 could also be observed in populations of A. halleri which grow on mining sites (Shahzad et al. 2010). However, the origination of these duplications could not yet be dated.

Hyperaccumulation as a strategy of metal tolerance could be rejected by Bert and colleagues (2000). They showed that the highest zinc tolerance level was accompanied with reduced accumulation in *A. halleri* (Bert et al. 2000). Bert and colleagues (2003) also showed that the amount of accumulated cadmium did not differ between plants with different tolerance levels. Due to this, hyperaccumulation can also be ruled out as a tolerance strategy towards Cd.

The accumulation of metals in leaves and phloem, however, reduces the feeding damage on plants (Stolpe et al. 2017, Kazemi-Dinan, 2014). Thus, the accumulation of heavy metals plays an important role in the defence mechanisms against herbivores (Stolpe et al. 2017, Kazemi-Dinan, 2014).

1.4. Clonal reproduction and inbreeding on metalliferous soils

The level of heavy metal tolerance can vary among populations and between individuals of *A. halleri* (Meyer et al. 2015, Bert et al. 2000, Van Rossum et al. 2004, Macnair 2002). Therefore, heavy metal pollution can represent a limiting factor even for tolerant plant species (Bert et al. 2000). It is anticipated that only few individuals, which can tolerate the local cadmium concentration, can colonize the respective site and accordingly the number of founder individuals is assumed to be low. To establish a proper population size, even with a very low number of founder individuals, the plants will reproduce clonally to establish a population.

Due to fine scale differences in the metal content, metalliferous soils represent a heterogeneous environment (Linhart and Grant 2006, Mattner et al. 2002 as cited in Bizuox and Mahy 2007). Clonal reproduction is often only successful at short distances (Bizoux and Mahy 2007). Thus, successful clonal propagation is frequently restricted to small patches with the individual optimal metal concentration (Bizoux and Mahy 2007). Higher metal concentrations and especially a high variation in the concentration of a soil presumably result in rarer and smaller patches, at which a plant is able to survive and reproduce. Owing to the restricted patch size, and potentially high distances between the patches, sexual reproduction between individuals of the same patch is expected to be more frequent than outcrossing between plants from different spots (Bizoux and Mahy 2007). Consequently, the number of potential mating partners within a patch and a population is reduced (Gaudeul et al. 2007). Moreover, the number of potential partners is further restricted due to self-incompatibility between individuals of the same genet (group of genetically identical individuals due to clonal propagation (Harper 1977)).

Alternatively, individuals originating from clonal propagation are connected with rhizomes or stolons (Hutchings and Wijesinghe 1997, Alpert 1999, Tielbörger and Gruntman 2016). These interconnections enable an exchange of different resources between the individuals of a genet (clonal integration) and could reduce stress induced by heavy metals (Hutchings and Wijesinghe 1997, Alpert 1999, Tielbörger and Gruntman 2016). As a consequence, plants can also establish larger populations on soils with a very heterogeneous cadmium content. Due to clonal propagation these populations will mainly persist of genetically identical individuals and the number of genetically different mating partners is again low.

In this regard, the mating between closely related individuals (inbreeding) should be high on metalliferous sites. Inbreeding reduces the probability that different alleles will be combined (Russel 2010). This leads to an increase of homozygosity (Russel 2010) which, in turn, enhances the probability of unfavourable phenotypes and reduced resistance (Charlesworth and Willis 2009, Russel 2010). Thus, inbreeding results in a low genetic variability and reduced fitness of a population (inbreeding depression) (Charlesworth and Willis 2009, Russel 2010). On the contrary, low genetic variability can also be beneficial since it ensures the spreading and maintenance of well-adapted genotypes (Salema and Sievanen 2002, Van Rossum et al. 2004).

In conclusion, the heavy metal content of a soil represents an important influential factor for the structure and reproduction mode of a population.

1.5. Restriction-site associated DNA sequencing (RAD-Seq)

Next generation sequencing (NGS) approaches have become increasingly popular and more cost effective (Schlötterer et al. 2014, van Dijk et al. 2014, McCormack et al. 2012, Cao and Sun 2016). Costs and time can moreover be reduced by simultaneously sequencing multiple individuals (pool sequencing) (Schlötterer et al. 2014, Davey et al. 2011, Cao and Sun 2016). However, pool sequencing just gives an average of a pool's genetic information (Goodwin et al. 2016, Mullen et al. 2012) and rare alleles might be underestimated or even remain unrecognized (Mullen et al. 2012). Furthermore, it is not possible to ascribe an allele to a specific individual sample (Cao and Sun 2016). Especially when not sequencing the whole genome the haplotype can only be determined for the given fragment of the sequence. Therefore, it is an insufficient tool to provide haplotype information for each individual (Cao and Sun 2016).

Sequencing a high number of samples individually is expensive, especially since many scientific questions do not require information from the whole genome to be answered (Schlötterer et al. 2014, Gautier et al. 2013, Davey et al. 2011). An established approach to achieve a complexity reduction in these situations is the use of restriction-site associated DNA sequencing (RAD-Seq) (Baird et al. 2008, McCormack et al. 2012, Poland et al. 2012). Here, specific restriction enzymes are used to excise the genomic regions of interest, which then are sequenced representatively for the whole genome (Baird et al. 2008, Davey and Blaxter, 2011, McCormack et al. 2012, Poland et al. 2012). The ascription of a particular, maybe rare, genetic variation to a certain individual can be achieved by the introduction of

individual-specific barcodes (Baird et al. 2008, Poland et al. 2012, Peterson et al. 2012). In RAD-Seq this is done in an early phase of library preparation. A subsequent pooling of the barcoded individual DNAs allows that later steps in the library preparation can be done on a single sample (Baird et al. 2008). Thus sequencing a high number of multiplexed individuals is also applicable for investigations at individual level (Baird et al. 2008, Poland et al. 2012, Peterson et al. 2012).

In the original RAD-Seq protocol the DNA is fragmented by one restriction enzyme and physical shearing (Baird et al. 2008, Shirasawa et al. 2015). Elshire et al. (2011) reduced the complexity of the protocol by using a rare cutting restriction enzyme and designing adapters with segments complementary to the restriction overhang (Genotyping-by-sequencing; GBS). By this, the shearing step can be avoided and adapters can directly be ligated to the digested DNA (Elshire et al. 2011). Poland et al. (2012) and Peterson et al. (2012) achieved a further complexity reduction with the introduction of a second restriction enzyme in the GBS protocol (double digest Restriction-site associated DNA sequencing; ddRAD). By applying two restriction enzymes also no shearing and DNA end repair steps are necessary (Peterson et al. 2012). This means that fewer manipulation steps are required and a smaller amount of starting DNA can be used (Etter et al. 2011, Andrews et al. 2016, DaCosta and Sorenson 2014).

1.6. Aim of the project

This master project is part of an ongoing collaboration of Univ.-Prof. Dr.rer.nat. Christian Schlötterer with Prof. Dr. Katja Tielbörger and Dr. Michal Gruntman from the University of Tübingen. The projects' overall objective is the investigation of metal tolerance and hyperaccumulation in the metal tolerant species *Arabidopsis halleri* and *Noccaea* caerulescens (Tielbörger and Gruntman 2016). A special focus lays on the causes and effects of genetic and phenotypic variation in metal tolerance and hyperaccumulation and the interaction with several ecological factors (Tielbörger and Gruntman 2016). Especially the role of clonal integration, which is the sharing of information and resources among individuals of a genet, will be examined (see Dong 1996a, 1996b, 2011).

In this master project the aspects of clonal propagation and inbreeding and their connection with metalliferous soils will be investigated. Since clonal propagation constitutes a selective advantage under harsh environmental conditions, a correlation between heavy metal pollution and the extent of clonal reproduction is assumed. Due to the resulting composition of a population, i.e. genetically identical and closely related individuals, also a correlation between metalliferous soils and inbreeding is expected. It is supposed that the degree of clonal reproduction is higher on contaminated sites. Furthermore, an increased level of homozygosity and of the inbreeding coefficient is presumed for plants from metalliferous soils.

By determining the frequency of clonal propagation on metalliferous soils, clonal integration can be investigated indirectly. Since clonal integration promotes the translocation of heavy metals and thus can reduce the effect of cadmium on an individual a contribution to heavy metal tolerance and hyperaccumulation is expected (Hutchings and Wijesinghe 1997, Alpert 1999, Tielbörger and Gruntman 2016). Based on this, clonal integration is presumed to be high on soils contaminated with heavy metals (Tielbörger and Gruntman 2016). A correlation of frequent clonal propagation on soils with a high cadmium content could thus be used as an indicator for high clonal integration.

2. Material and Methods

2.1. Leaf material

Leaf samples from natural populations of *Arabidopsis halleri* (Linnaeus) O'Kane & Al-Shehbaz were collected from five German metalliferous regions: Littfeld (litt), Vienenburg (vieb), Clausthal Zellerfeld (clau), Lautenthal (laut) and Wulmeringhausen (wulm). Samples from non-metalliferous locations were collected from sites in Blaibach (blai), Bad Rippoldsau (badr) and "Fortfun" (fort) in the municipality Bestwig (Figure 1). Additionally, leaves were collected from one metalliferous (czrc) and two non-metalliferous (czra, czrb) sample sites in the Bohemia forest in the Czech Republic. Former mining sites were chosen as metalliferous sampling sites (in correspondence with the collaborators).

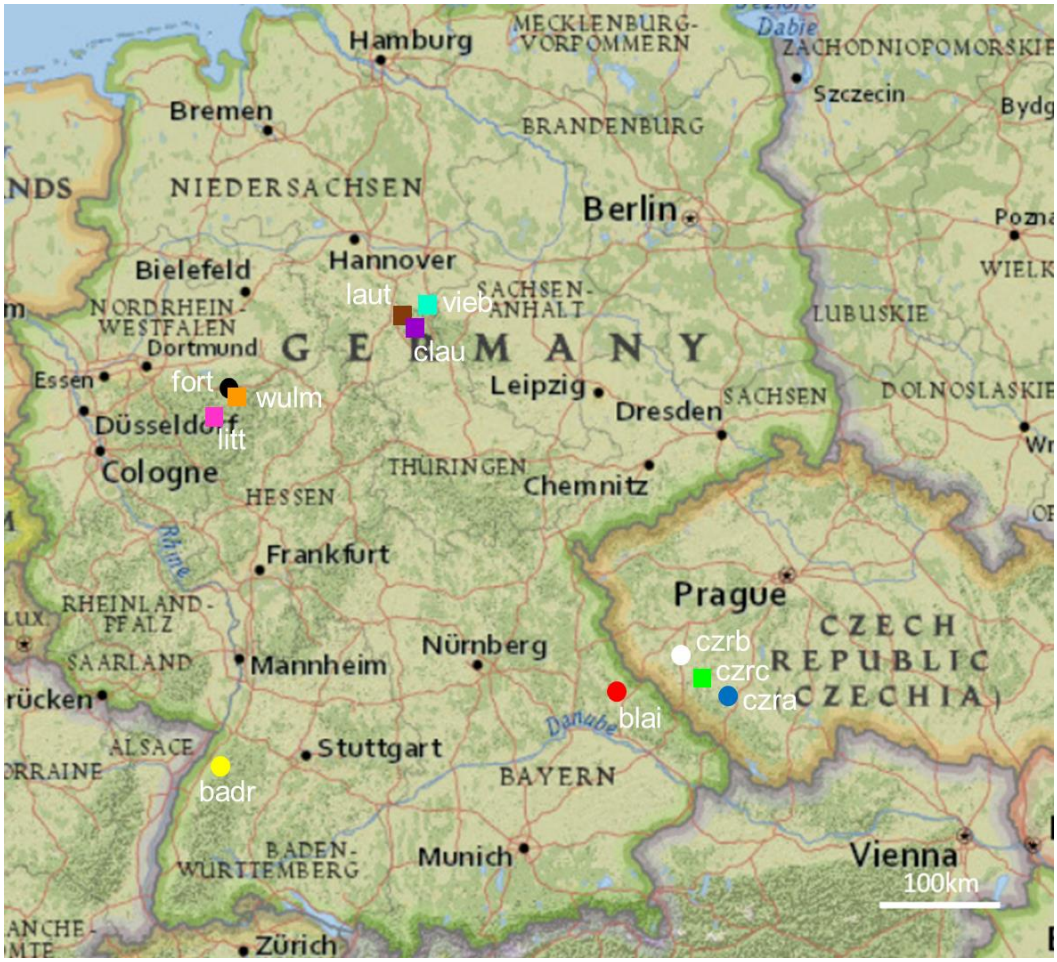

Figure 1. Sampling sites. Locations of the sampling sites in Germany and the Czech Republic are shown. Non-metalliferous sampling sites are marked with circles. Cadmium contaminated locations (former mining sites) are labelled with squares. badr = Bad Rippoldsau, blai = Blaibach, clau = Clausthal Zellfeld, fort = Fortfun/Bestwig, laut= Lautenthal, litt = Littfeld, vieb = Vienenburg, wulm = Wulmeringhausen, czra = Czech Republic population 1, czrb = Czech Republic population 2, czrc = Czech Republic population 3. Scale bar = 100km. Generated with map-maker.education.nationalgeographic.com (14.3.2017).

Along a transect material from 20 plant-pairs was collected per population. Two leaves per individual were sampled from 20 individuals per population with an interindividual distance of 3m (o) (Figure 2) (in correspondence with the collaborators). Furthermore, leaves were collected from adjacent partner-individuals growing 50cm apart from each of the already sampled plants (y) ($n_{(total)}$=433) (in correspondence with the collaborators). The samples were classified according to the distance between two plants. Individuals with a distance of 50cm were denoted as pairs/paired (P).

For drying, leaf samples were placed on silica gel (in correspondence with the collaborators).

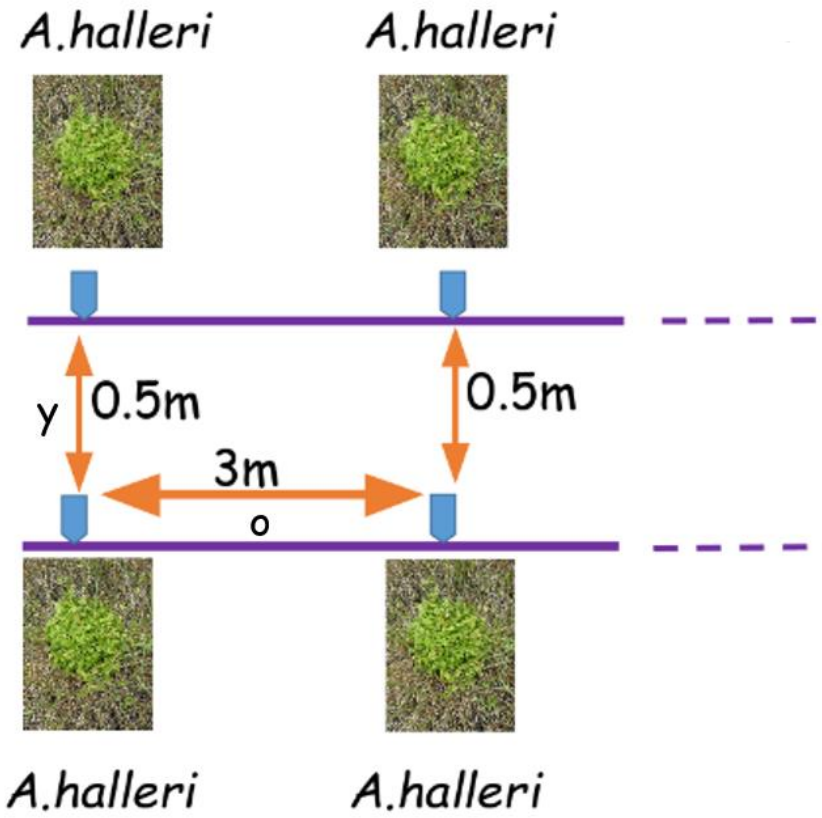

Figure 2. Sampling method (provided by the collaborators). Per population leaves from 20 individuals which grew at a distance of 3m were collected (o). Furthermore, leaves of individuals growing 50cm apart from the respective plant were sampled (y). A detailed description of each sample site can be found in Table 1. (provided by the collaborators)

At the metalliferous locations the soil was contaminated with cadmium (Cd). Moreover also high concentrations of copper (Cu), manganese (Mn), lead (Pb) and zinc (Zn) were measured. For this project only the impact of Cd was investigated (Tielbörger and Gruntman 2016). To take heterogeneities in the soils heavy metal concentration into account, 9-16 soil samples per site were taken at a depth of 15 cm (provided by the collaborators). The concentrations of the individual soil samples of a site can be found in Supplementary Table S1. A microwave digest was done using HCl and HNO_3 (Stein et al. 2017). The concentration of each heavy metal was determined using inductively coupled plasma optical emission spectrometry (ICP-OES) (in correspondence with the collaborators, Stein et al. 2017). The leaf sampling as well as the determination of the soil concentration was done by the collaborators from the University of Tübingen. The exact location of the sampling sites and its mean metal concentrations are summarized in Table 1.

Table 1. Coordinates and description of the sampling sites (provided by the collaborators). Summary of the population's location and heavy metal concentrations. The concentration of Cd, Cu, Mn, Pb and Zn were determined with ICP-OES by pooling 9-16 soil samples per site collected at a depth of 15 cm. A microwave digest was done using HCl and HNO_3 (Stein et al. 2017). According to the threshold values defined in Wuana and Okieimen (2011) the sample sites were classified as metalliferous (m) and non-metalliferous (nm). For better illustration populations from metalliferous sample sites are shaded in grey.

Site	Origin	Area	Area description	GPS coordinate	Altitude [m]	Population size	Cd [µg/g]	Cu [µg/g]	Mn [µg/g]	Pb [µg/g]	Zn [µg/g]
Bad Rippoldsau (badr)	nm	Germany: Freudenstadt	meadows on a slope	N 48° 24‘ 48.6“ E 008° 19‘ 41.7"	650	100-500	0.641	15.742	617.731	66.870	265.054
Blaibach (blai)	nm	Germany: Bayern	Lawn at the Rege river bank	N 49° 09.830' E 012° 47.759'	382	50-500	1.429	37.698	406.77	91.927	147.770
Clausthal Zellerfeld (clau)	m	Germany: Harz Mountains	Meadows next to Ag/Zn/Pb mining heaps	N 51° 48' 08.8" E 010° 18' 11.1"	483	500-5,000	36.915	964.642	2,606.250	19,349.080	8,821.167
Fortfun, Bestwig (fort)	nm	Germany: Sauerland	forest next to the fun-park	N 51° 18' 26.4" E 008° 26' 29.1"	230	>400	2.217	18.618	1,008.892	214.483	426.675
Lautenthal (laut)	m	Germany: Harz Mountains	Meadows next to Ag/Zn/Pb mining heaps	N 51° 51' 45.3" E 010° 18' 00.4"	390	>5,000	34.957	270.194	1,468.45	2,677.708	9,349.833
Littfield (litt)	m	Germany: Siegerland	Old mining area	N 51.00540° E 008.006606°	388	>5,000	18.783	414.391	6,488.250	7,169.833	5,351.083
Vienenburg (vieb)	m	Germany: Harz Mountains	meadows next to the stream	N 51° 57' 29.4" E 010 34' 08.2"	136	100-500	21.673	608.058	2,206.583	4,542.167	2,777.500
Wulmeringhausen (wulm)	m	Germany: Sauerland	Meadows next to railway track	N 51° 18.383' E 008° 29.112'	388	150-500	22.305	295.048	3,110.917	7,648.642	8,911.583
Bohemia Forest CZ1 (czra)	nm	Czech Republic	Meadows next to road	N 48°59′ E 13°46′	1,052	100-500	0.981	37.863	477.456	101.200	78.191
Bohemia Forest CZ2 (czrb)	nm	Czech Republic	Meadows next to forest	N 49°03′55″ E 13°46′3″	1,069	100-500	1.312	27.005	529.367	134.665	211.172
Bohemia Forest CZ3 (czrc)	m	Czech Republic	Meadows next to railway track	N 49°03′35″ E 13°33′31″	405	100-500	21.289	103.912	2,218.417	1,579.667	2,026.917

2.2. Extraction

DNA was extracted from 433 leaf samples following the extraction protocol of Miller et al. (1988). The used tools and reagents are summarized in Supplementary Table S2.

First, a dried leaf was grinded using MiniG™ 1600 (Spex® SamplePrep, USA). For this a leaf was crunched in 200µl 2X CTAB-Buffer (2% Hexadecyltrimethylammonium bromide, 100mM Tris-HCl pH 8, 20mM EDTA pH 8, 1.4M NaCl, 1% PVP (Polyvinylpyrrolidone)) until it was properly homogenized (approximately 5 minutes) using a 2ml Eppendorf tube and 2 metal beads. The crunched samples were filled up with 800µl CTAB-Buffer. Subsequently the samples were incubated for 20 minutes at a temperature of 65°C and 700rpm using PHMT Grant-bio Thermomixer (Grant Instruments Ltd., United Kingdom).

Afterwards the beats were removed and 0.5µl of RNAse A (100mg/ml) (QIAGEN, Germany) was added. The samples were again incubated for 30 minutes at 37°C.

1ml chloroform was added and samples and chloroform were mixed by multiple inverting of the tubes. The samples were centrifuged for 5 minutes at a temperature of 4°C using the Eppendorf Centrifuge 5424R (Eppendorf AG, Germany). The upper phase was transferred into a new 2ml tube. 950µl isopropanol were added and the samples were mixed using IKA® Vortex Genius 3 (IKA®-Werke GmbH & Co. KG, Germany). Subsequently the samples were again centrifuged for 20 minutes at 4°C. The supernatant was decanted and 50µl of 70% ethanol were added to wash the DNA pellet. The pellet and ethanol were mixed and centrifuged for 10 minutes at 4°C. The supernatant was again decanted and the pellet dried over night at room temperature. The dry pellets were then dissolved in 12-24µl Low TE (10mM Tris-HCl pH 8, 0.1mM EDTA pH 8). For proper dissolving the samples were allowed to stand for one day at room temperature. They were mixed and spun down using Eppendorf Centrifuge 5424 (Eppendorf AG, Germany).

The DNA concentration was quantified using the Quant-iT™ dsDNA HS Assay Kit and the Qubit™ fluorometer (Invitrogen™ by Thermo Fisher Scientific, USA). For this, 200µl of the dilution buffer and 1µl assay reagent were mixed. 1µl of the sample was added to 199µl of the buffer-dye-Mastermix. They were mixed, spun down and quantified.

2.3. Restriction digest

DNA was digested using a two-restriction-enzyme approach, following the protocol of Poland et al. (2012). For this, the DNA was first normalized with MilliQ-water to achieve a total concentration of 20ng in 10µl. Due to an insufficient amount of DNA 16 samples needed to be excluded from the further processing steps.

The normalized DNA was digested using the rare cutting PstI-HF (CTGCAG) and the more frequently cutting MspI (CCGG) restriction enzyme (Poland et al. 2012). Both restriction enzymes were ordered from New England Biolabs ® Inc. (NEB, USA). 2µl CutSmart® Buffer (10X, NEB), 0.4µl of PstI-HF restriction enzyme, 0.4µl MspI restriction enzyme and 7.2µl MiliQ-water were mixed with 10µl of the normalized DNA. The material was digested at 37°C for 2 hours. Subsequently the enzymes were inactivated for 20 minutes at 65°C. For this, the ProFlex PCR system (Thermo Fisher scientific, USA) was used.

2.4. Adapter preparation

The adapters were prepared following the protocol of Poland et al. (2012).

220 barcoded forward adapters and a reverse adapter, designed by Poland and colleagues (2012), were ordered from Sigma® Life Science (USA). The barcodes varied in length between 5 and 10bp (Poland et al. 2012). A list of the used barcode and adapter sequences can be found in the Supplementary Table S3. The forward adapter included an Illumina adapter, a binding site for the PCR primer and the individualizing barcode (Poland et al. 2012). The reverse adapter matched to the overhang of the common cutting enzyme (MspI) and was designed as a Y adapter (Poland et al. 2012). This means that it exactly corresponded to the sequence of the reverse primer (Poland et al. 2012). Thus, the complementary reverse primer binding site first needed to be synthesized in the course of the first PCR cycle (Figure 3) (Poland et al. 2012). This enables the production of a uniform library, since only fragments

consisting of a forward adapter (including the barcode), the excised genomic DNA and a reverse adapter were amplified (Poland et al. 2012).

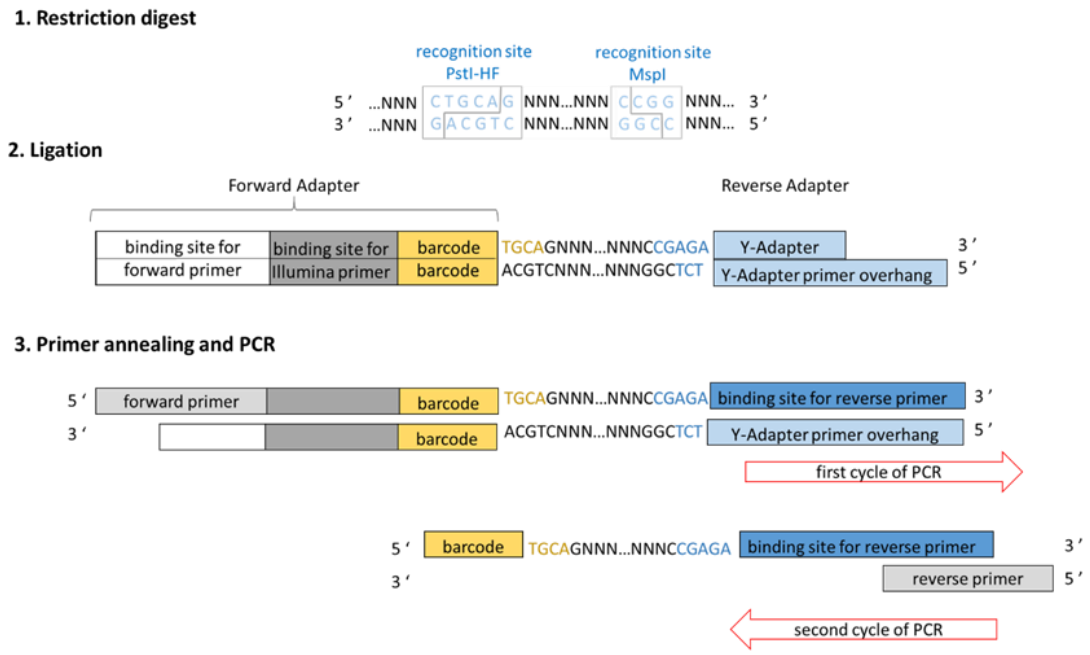

Figure 3. Adapter ligation and fragment amplification (adapted from Poland et al. 2012, Baird et al. 2008, Peterson et al. 2012). After the restriction digest the barcoded adapter was ligated to the restriction overhang of PstI-HF. The reverse adapter was ligated to the restriction overhang of the more frequently cutting enzyme MspI. Since the reverse adapter contained a sequence, which was identical to the reverse PCR primer, the complementary primer binding site first needed to be synthesized in the first cycle of PCR. Subsequently the reverse primer could bind. This design enabled, that no fragments with the sequence motive of the common restriction enzyme (MspI) on both ends were amplified (Poland et al. 2012).

The adapters were ordered in form of two single stranded oligos. The oligos were in solution, whereby 15nmol of an adapter were diluted in 150µl water (100µM). For the annealing 10µl of the single stranded oligos were mixed with 10µl of 10X Adapter Buffer (500mM NaCl, 100mM Tris-Cl) and 70µl MiliQ-water. This results in total volume of 100µl double stranded adapter with a concentration of 10µM (Poland et al. 2012). A ProFlex PCR System was used to anneal the adapters. The annealing started at 95°C and was programmed to cool down to 30°C with a temperature reduction of 1°C per minute (Poland et al. 2012). The barcoded adapters were diluted 3:10 with 1X Elution Buffer (10mM Tris-Cl, pH 8.5 (QIAGEN, Germany)). (for details see Supplementary Table S4 and S5). Subsequently they were quantified using Qubit ™ fluorometer.

To get a normalized concentration, all adapters were diluted to 0.1pmol in 100µl (0.1µM) using MiliQ-water (Supplementary Table S4 and S5). The reverse Y-Adapter was not diluted and was kept at 10µM (Poland et al. 2012). A separate working stock (WS) was prepared for each forward adapter. For this, 50µl of 1X Adapter Buffer, 20µl of the barcoded Adapter and 30 µl of the reverse adapter were mixed. The final working stock consisted of 0.02µM barcoded and 3µM reverse adapter (Poland et al. 2012).

2.5. Ligation

After the restriction digest the DNA fragments were immediately ligated to the adapters. To ligate the adapters to the samples 20µl of the restriction digest were mixed with 2µl CutSmart® Buffer, 0.5µl T4 DNA ligase (NEB, USA), 4µl ATP (NEB, USA), 8.5µl MiliQ water and 5µl of the adapter working stock. The samples were incubated for 2 hours at 22°C.

Following this, the ligase was inactivated for 20 minutes at 65°C. For ligation and ligase inactivation the ProFlex PCR system was used.

2.6. Pooling and purification

6.2 -20µl of 55, respectively 8 samples were taken from the restriction digest. The samples were multiplexed and purified using the QIAquick® PCR Purification Kit (QIAGEN GmbH, Germany). Here the fivefold volume of Buffer PB (5M Gu-HCl, 0.9M potassium acetate, pH 4.8) was added to the pooled samples. The mixture was loaded onto a column placed in a 2ml tube. The filled column was centrifuged for one minute at 13000rpm. The flow-through was discarded. 750µl of PE washing Buffer (10mM Tris-HCl pH 6.6, 80% ethanol) were added and the column was again centrifuged for one minute. Again the flow-through was discarded and the empty column was centrifuged a second time for one minute to ensure that residual wash buffer was removed. The provided tube was discarded and the column was placed in a new 1,5ml tube. Subsequently the column was eluted with 41-81µl Buffer EB (10mM Tris-Cl, pH 8.5). The elution was done in two steps, whereby 20 + 21µl, 40 + 41µl respectively were added and the column was centrifuged for one minute.

The eluate was quantified using Qubit.

2.7. Amplification

The forward primer (5' - AATGATACGGCGACCACGAGATCTACACTCTTTCCCTACACGACGCTC TTCCGATCT - 3') consisted of 58bp. The reverse primer (5' - CAAGCAGAAGACGGCATACGAGATCGGTCTCGGCATTCCTGCTGAA - 3') had a length of 46bp. Both primers were ordered from Invitrogen (Thermo Fisher scientific, USA) and diluted to 10mM using MilliQ-water.

To amplify the pooled samples 13.3-15µl DNA were mixed with 25µl Phusion polymerase (NEB, USA), 2µl forward primer, 2µl reverse primer and 6-7.7µl MiliQ water. For this PCR tube-strips were used. To avoid preferential amplification 3-6 PCRs à 50µl were run in parallel. The amplification was done using ProFlex PCR systems (Thermo Fisher scientific). For this an initial denaturation step for 30 sec at 98°C, followed by 13 cycles of denaturation (10 sec at 98°C), annealing (30 sec at 65°C) and elongation (30 sec at 72°C) was programmed. For the final extension 5 minutes at 72°C were set. After termination the samples were held at 4°C.

2.8. Size selection and gel extraction

In the original protocol a short extension time was used to enrich for fragments within a range of 200 - 500bp (Poland et al. 2012). This length range is optimal for bridge amplification during sequencing (Poland et al. 2012). Due to the high efficiency of the used polymerase, a short elongation time was not sufficient for the preferential amplification of short fragments (Poland et al. 2012). Therefore, the fragment size needed to be selected manually. The pooled amplified samples were loaded onto a 2% agarose gel with ethidiumbromide which run for at least 2.5 hours at 700mA and 90V. The size was selected by cutting out a slice of gel between the range 450 and 650bp (insert size: 320 - 520bp) using the Molecular Imager® Gel Doc™ XR+ system (USA) and a ChromaLight blue Conversion Screen (distributed by Biozym Scientific GmbH, Germany). This range was chosen to avoid that, with a read length of 125bp, the adapter at the opposite end of the fragment was sequenced as well.

After size selection the DNA was extracted from the gel using the MinElute® Gel Extraction Kit (QIAGEN, Germany). The gel slice was weighed in a 2ml tube and dissolved in a threefold volume of Buffer QG (5.5 M guanidine thiocyanate, 20mM Tris-HCl pH 6.6). Afterwards 1 gel volume of isopropanol was added and mixed through inverting the tube multiple times. The sample was load onto the column, which was placed into a 2ml tube. After loading the column, it was centrifuged for one minute at 13000rpm. The flow-through was discarded. Subsequently 500µl of Buffer QG were added and the column was centrifuged for one minute. The flow-through was again discarded. To wash the column 750µl Buffer PE were

added and the column was centrifuged for one minute. Afterwards the flow-through was discarded and the empty tube was centrifuged again to remove residual washing buffer. The column was placed into a new 1.5ml tube and eluted with 41µl Buffer EB (10 mM Tris-Cl, pH 8.5). The elution was done in two steps (20 +21µl).

2.9. Quantification and sequencing

The eluate of the purified sample batches was quantified using the Qubit™ fluorometer. Different numbers of samples were combined to a pool in the multiplexing-step. As a result, the amount of DNA per sample differed between the batches. To ensure a normalized DNA concentration the respective volume was calculated for each batch according to the number of pooled samples. The normalized batches were combined into two separate pools consisting of DNA material from 220 and 197 individuals (Table 2). The pools were sequenced in two separate runs. Paired-end Illumina sequencing (HiSeq 2500) with a read length of 125bp was performed at the VBCF NGS Unit (www.vbcf.ac.at). Due to the use of individual barcodes a pool of samples could be sequenced on a single lane (Poland et al. 2012).

Table 2. Total number of samples. The total number of sequenced samples (n = 417) is given for each population. Populations from metalliferous sites are shaded in grey.

Population	Abbreviation	Number of samples
Bad Rippoldsau	badr	38
Blaibach	blai	36
Clausthal Zellerfeld	clau	40
Fortfun/Bestwig	fort	38
Lautenthal	laut	38
Littfeld	litt	39
Vienenburg	vieb	38
Wulmeringhausen	wulm	40
Bohemia Forest CZ1	czra	37
Bohemia Forest CZ2	czrb	38
Bohemia Forest CZ3	czrc	35

2.10. Processing NGS reads

2.10.1. De-multiplexing

In total 417 samples from 11 populations were used for data analysis. The sequencing data were de-multiplexed using an in house custom Python script (provided by Lukas Endler). For this, the sequences were first filtered by checking for the presence of a barcode and the expected restriction overhangs at the 5´- and 3´-end of the fragments. All sequences which did not possess a barcode or the restriction sites for both enzymes (PstI and MspI), were excluded. The remaining sequences were split and sorted on the basis of the unique barcodes. Subsequently the restriction overhangs and adapters were cut.

The further data processing was performed employing in house custom shell and R scripts (provided and supported by Marlies Dolezal, R Core Team 2017) (Supplementary Table S6 and S7).

2.10.2. Quality control

For quality control the remaining BAM-files were first converted into fastq-files using bedtools v2.25.0 (option: bamtofastq; Quinlan and Hall 2010). Subsequently the quality of the raw sequences was assessed using FastQC v0.11.5 (Andrews 2011). To identify overrepresented sequences the sequences were determined using the NCBI Web-BLAST (Altschul et al. 1990, https://blast.ncbi.nlm.nih.gov/Blast.cgi?PROGRAM=blastn&PAGE_TYPE=BlastSearch&LINK_LOC=blasthome).

2.10.3. Mapping

The FASTA sequence of *Arabidopsis halleri* provided by Briskine et al. (2016a) was used as a reference genome. The genome assembly comprised of short contigs (for details see Results and Briskine et al. 2016a). The reference genome was prepared by creating an index file applying samtools v1.4.1 (Li et al. 2009). A sequence directory was created using Picard v1.136 (java: v1.8.0_131, http://broadinstitute.github.io/picard). Reads were mapped to the reference genome using BWA mem v0.7.15 (parameters: -R -V, Li 2013).

A summary of the sequence statistics was generated using samtools Flagstat v1.4.1 (Li et al. 2009).

2.10.4. Identifying loci mapping to chloroplast, mitochondria and repeat-masked regions

To determine the number of reads mapping to the nuclear genome, reads from chloroplast regions were excluded. Furthermore, loci mapping to repeat-masked regions were also removed. To identify which loci mapped to chloroplast or to repeat-masked regions the reads were mapped to the chloroplast genome of *Arabidopsis halleri* subsp. *halleri* (Novikova et al. 2016) and to the RepeatMasker output provided by Briskine and Colleagues (2016b). Moreover, the reads were mapped to a mitochondrial genome of *Arabidopsis thaliana* (Unseld et al. 1997). For the mapping BLAST v2.2.26 (Altschul et al. 1990) was used.

2.10.5. Polymorphism calling

Polymorphisms were called using Freebayes v1.1.0 (parameters: --use-best-n-alleles 4, Garrison and Marth 2012).

The coverage of each genotype was calculated with VCFtools (parameter: --geno-depth). Variations in the coverage were investigated and illustrated using the R-packages ‘plotly’ v4.7.1 (Sievert et al. 2017), ‘car’ v2.1-5 (Fox and Weisberg 2011), ‘nortest’ v1.0-4 (Gross and Ligges 2015), ‘scatterplot3d’ v0.3-40 (Ligges and Maechler 2003), ‘rgl’ v0.98.1 (Adler et al 2017), ‘corrplot’ v0.77 (Wei and Simko 2016) and ‘corrgram’ v1.12 (Wrigth 2017). Spearman correlations were performed to investigate correlations between the coverage of different loci and individuals. Furthermore, the correlations were PCA-ordered and illustrated using the R-package ‘seriation’ v1.2-2 (Hahsler et al. 2008, 2017, methods: ‘OLO’, ‘GW’, ‘HC’).

2.10.6. Filtering

As recommended by the authors SNPs and indels were called together (Garrison and Marth 2012). Indels were later filtered with VCFtools v0.1.13 (parameters: --remove-indel; Danecek et al. 2011). Reads mapping to the chloroplast genome or to repeat-masked regions were excluded (VCFtools, parameters: --exclude-positions).

Only fully diploid biallelic SNPs were used for the analysis. Therefore, all sites, with more than two alleles (one reference and one alternative allele) were removed by filtering the data using GATK (-restrictAllelesTo BIALLELIC). Furthermore, only SNPs with a quality-score (QUAL) equal to or higher than 100 were used. The QUAL-score is a Phred-scaled assessment of the calls’ confidence (http://samtools.github.io/hts-specs/VCFv4.3.pdf), whereby a high sequencing accuracy is indicated by high quality scores (https://software.broadinstitute.org/gatk/documentation/article.php?id=4260).

A set of SNPs, which were common to all populations (fully diploid SNPs), was created using GATK v3.7-0 (McKenna et al. 2010, parameters: -T CombineVariants, -T SelectVariants, -select ‘set==”intersection”’). Furthermore the loci of the populations were compared (--concordance, --discordance).

To increase the number of common usable fully diploid SNPs, individuals with a high number of missing loci were excluded. To do so, a subset of samples with less than 50% missing loci was created in R (VCFtools, parameters: --keep).

2.11. Population genetics

2.11.1. Population analysis

To compare the populations the number of private sites per population was calculated with GATK (-T SelectVariants, --discordance). Furthermore VCFtools was used to determine the number of loci only present in one individual (--singletons). These comparisons were performed on the basis of all biallelic loci present in samples with less than 50% missing data.

To investigate if private sites could be ascribed to technical artefacts a multi-sample calling was done with Freebayes using individuals with less than 50% missing loci from all populations. VCF-files only containing the private sites of a population were combined into a single file using GATK (-T CombineVariants). Loci common to the output of multi-sample calling and to the file with private sites from all populations were determined with GATK (-T SelectVariants --concordance) resulting in a file with 29,553 sites. Subsequently the coverage of each genotype was measured with VCFtools (--geno-depth). The number and affiliation of called SNPs to a particular population was examined in R. For genotypes with missing entries a value of -1 is provided by the VCFtools function --geno-depth (http://vcftools.sourceforge.net/man_latest.html). Thus, loci which possessed a positive coverage record could easily be identified and were denoted as called SNPs. The quantity of loci, which were exclusively found in one individual, was calculated.

Principal component analyses (PCA) were performed with the packages 'prcomp' v3.4.0 (R Core Team 2017) and 'adegenet' v2.0.1 (Jombart 2008, Jombart and Ahmed 2011). For this, the final filtered VCF-file, with 21,711 loci, was converted into a genlight object using 'vcfR' v1.5.0 (Knaus and Grünwald 2017, Knaus and Grünwald 2016). The genlight object was converted into a matrix. Subsequently all loci with missing data and a variance equal to zero were excluded using 'matrixStats' v0.52.2 (Bengtsson 2017). A scaled and centered PCA was performed with 'prcomp' on the basis of 3,154 remaining loci present in all 260 individuals.

Since most loci possessed F_{ST}-values below 0.4 additional PCAs were performed to investigate how much variance could be explained by loci with top F_{ST}-values. For this PCAs were done only using loci with F_{ST}-values above 0.4, 0.5, and 0.6 respectively. To do so, the pairwise F_{ST} was calculated over all populations using VCFtools (parameter: --weir-fst-pop). The subset of data with F_{ST}-values above 0.4 (n_{loci}=264), 0.5 (n_{loci}=88) or 0.6 (n_{loci}=36) was created in R. The PCAs were performed using 'prcomp'.

The pairwise F_{ST} was calculated using the R-package 'hierfstat' v0.04-22 (Goudet and Jombart 2015). For this, the filtered VCF-file was converted into a genind-object using the R-package 'vcfR'. The effect of distance between two populations on the pairwise F_{ST} was calculated with the function 'lm' (R Core Team 2017).

To identify which of the common SNPs (n = 3,961) strongly contributed to the differentiation of the populations, the F_{ST}-values (calculated with VCFtools, parameters --weir-fst-pop) and the squared loadings of the first principal component (calculated with R package 'adegenet') of each locus were compared. Furthermore, the top 10% of the loci (n = 396) possessing the highest F_{ST}-values and squared loadings respectively, were tested for a correlation (Spearman).

To estimate the genetic variation among the populations their level of heterozygosity was determined. On the basis of fully diploid SNPs present in samples with less than 50% missing loci heterozygosity was calculated for each locus in a population using VCFtools (--site-pi). Subsequently a mean value was computed for each population.

2.11.2. Impact of metal contamination

The inbreeding coefficient, F_{IS}, of each individual was determined using VCFtools (--het). The function 'lmer' from the R package 'lme4' v1.1-13 (Bates et al. 2015) was used to model the covariance with "population" as random effects for normal distribution. The residuals were checked for normal distribution and variance homogeneity. Differences in the extent of inbreeding between different populations were calculated with linear models using the R-package 'lsmeans' v2.26-3 (Lenth 2016). After correction for multiple testing p-values < 0.05 were considered significant. The impact of cadmium on the degree of inbreeding was calculated using 'lmer' and 'lmerTest' v2.0-36 (Kuznetsova et al. 2017) with "cadmium contamination" as

a fixed and “population” as random effect and ‘lsmeans’. The same was done for longitude, latitude, and altitude to investigate the influence of of a population’s geographic location. Furthermore, ‘lmer’ and ‘lmerTest’ were used to test the influence of the actual cadmium concentration on the inbreeding coefficient. Since heterogeneities in the cadmium content of a sample site could affect the number of genetically different individuals colonizing a location, also the impact of variance in the cadmium concentration was tested with ‘lmer’ and ‘lmerTest’. In addition, the influence of the number of individuals on a population site was tested with ‘lmer’ and ‘lmerTest’ to investigate if the size of a population affects the extent of inbreeding. Since the population size was specified as a range the effect of the minimal, mean and maximal number of individuals was measured. The population size at each sample site is stated in Table 1.

To estimate the extent of clonal propagation the kinship coefficient was determined by calculating the relatedness between the individuals using VCFtools (--relatedness2). The kinship coefficients range from 0 for unrelated individuals to 0.5 for monozygous twins and clones (Manichaikul et al. 2010). Kinship calculations for an individual with itself were excluded. Moreover, repeated calculations for the same sample combination (e.g. 42 and 111, and 111 and 42) were removed. This ensured that only unique kinship values were kept per individual pair. Again ‘lmer’ was used to model the covariance (“population” = random effects for normal distribution) and the residuals were checked to validate the model. Differences in the kinship coefficients were determined by performing a Kruskal-Wallis test over all populations. To determine which populations differed significantly Wilcoxon tests were performed for each population pair. The results of the Wilcoxon tests were corrected for multiple testing by using the R-function p.adjust v3.4.0 (R Core Team 2017, method: ‘holm’). The approach of pairwise Wilcoxon tests represents only a suboptimal model since a Wilcoxon test assumes dependent samples (Wilcoxon 1945). The differences of the kinship coefficients, however, were determined by pairwise comparisons of the populations, which represent independent samples and the use of pairwise Wilcoxon tests can thus lead to false negative significant effects. However, since the linear model did not meet the assumptions, this simpler approach needed to be used.

3. Results

3.1. Quality control and basic statistics

3.1.1. Reference genome

The genome assembly of Briskine et al. (2016) was used as a reference genome. The reference genome assembly consisted of 2,339 contigs. These possessed a total length of about 196Mb and a median contig length of 4,624b (Figure 4). 68,173 repeat-masked regions were found which made up a total length of 26Mb.

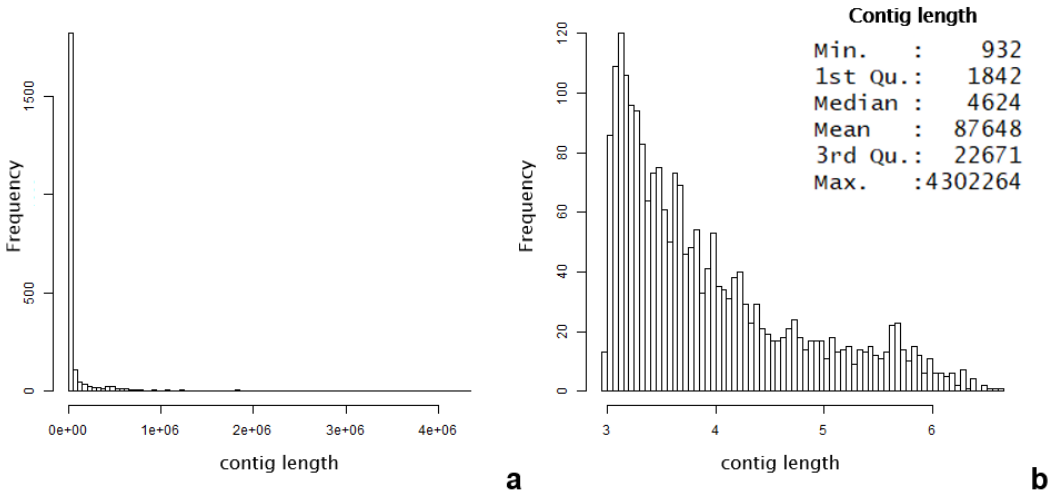

Figure 4. Reference genome properties. Size and length distribution are given for the 2,339 contigs of the reference genome assembly (a). A $\log_{10}$ -transformed distribution is also shown for better illustration (b). Furthermore, a tabular summary of the contig lengths is given.

3.1.2. Sequence data

The sequences of 417 individual samples were analysed and evaluated (Table 3). After removing the adapters a median read length of 120bp remained. A median number of 937,626 reads per sample (Min: 24, Max: 8949,446) was determined (Figure 5). Only in few individuals less than 40% of the reads mapped successfully to the reference genome. In the majority of the individuals more than 70% of the reads could be mapped to the reference genome (Figure 6). Most of them possessed 75-85% mapped reads. In some samples even more than 85% of the reads could be mapped. After excluding reads with incorrect orientation, a too large insert size between the reads of a pair or reads for which the second read of the pair was missing the number of properly paired reads was still high (Figure 6d). In most of the individuals more than 60% of the reads mapped as proper pairs. By comparing the total amount of mapped reads to the percentage of properly paired reads a reduction of 10% could be observed. This indicates that 10% of the mapped reads were lost due to improper pairing.

In all samples a good quality with per base and per sequences Phred scores above 20 were observed. For all samples a bias in the sequence composition could be found in both reads. However, a bias in the first bases was anticipated from the applied approach (restriction enzymes + adapters). In the first read of the read pair the bias was measured within the first 13-15bp and a high G-content (around 33%) was found at the tenth to twelfth position of a read. Similar high contents were determined for all four bases in the proceeding positions. In the second read a strong bias was found within the first three bases. At the first three positions a high per base sequence content of C, respectively G (up to 100%) was measured. The differences in sequence content of the first basepairs of both reads can be explained by the overhangs of both restriction enzymes and the subsequent adapter sequences and are thus congruent with the expectations.

Table 3. Sequenced samples. The number of samples (n = 417) is given for each population. To roughly estimate the maximal number of pairs (P), individuals with a distance of 3m (o) and 50cm (y) (see sampling method) are given separately (subdivision). Populations from metalliferous sample sites are shaded in grey.

Population	Abbreviation	Subdivision	number of samples
Bad Rippoldsau	badr	Y	18
		O	20
Blaibach	blai	Y	18
		O	18
Clausthal Zellerfeld	clau	Y	20
		O	20
Fortfun/Bestwig	fort	Y	18
		O	20
Lautenthal	laut	Y	18
		O	20
Littfeld	litt	Y	20
		O	19
Vienenburg	vieb	Y	19
		O	19
Wulmeringhausen	wulm	Y	20
		O	20
CZ1	czra	Y	19
		O	18
CZ2	czrb	Y	20
		O	18
CZ3	czrc	Y	18
		O	17

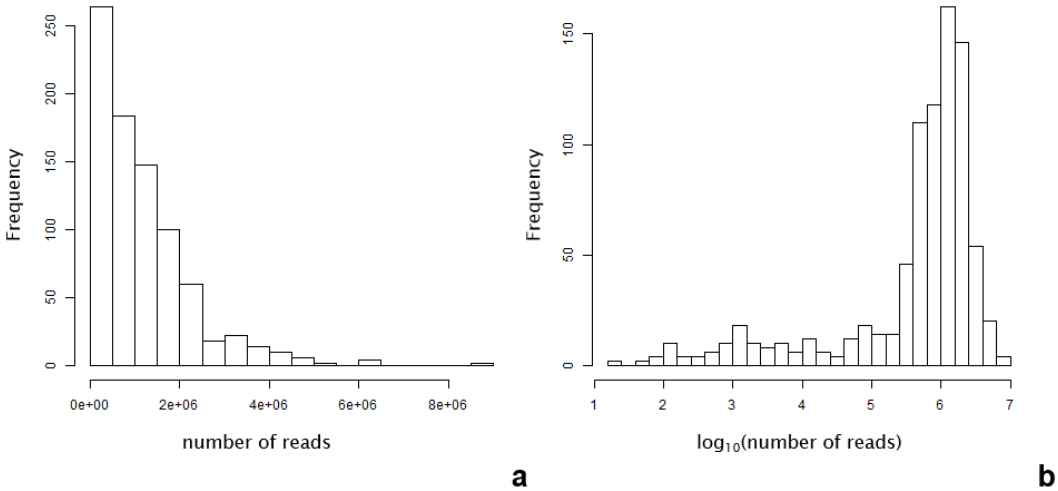

Figure 5. Total number of reads in 417 samples. The distribution of the number of reads in all 417 samples (n = 417) is given (a). Furthermore, a logarithmic (log_{10}) distribution of the read number in all individuals is presented (b).

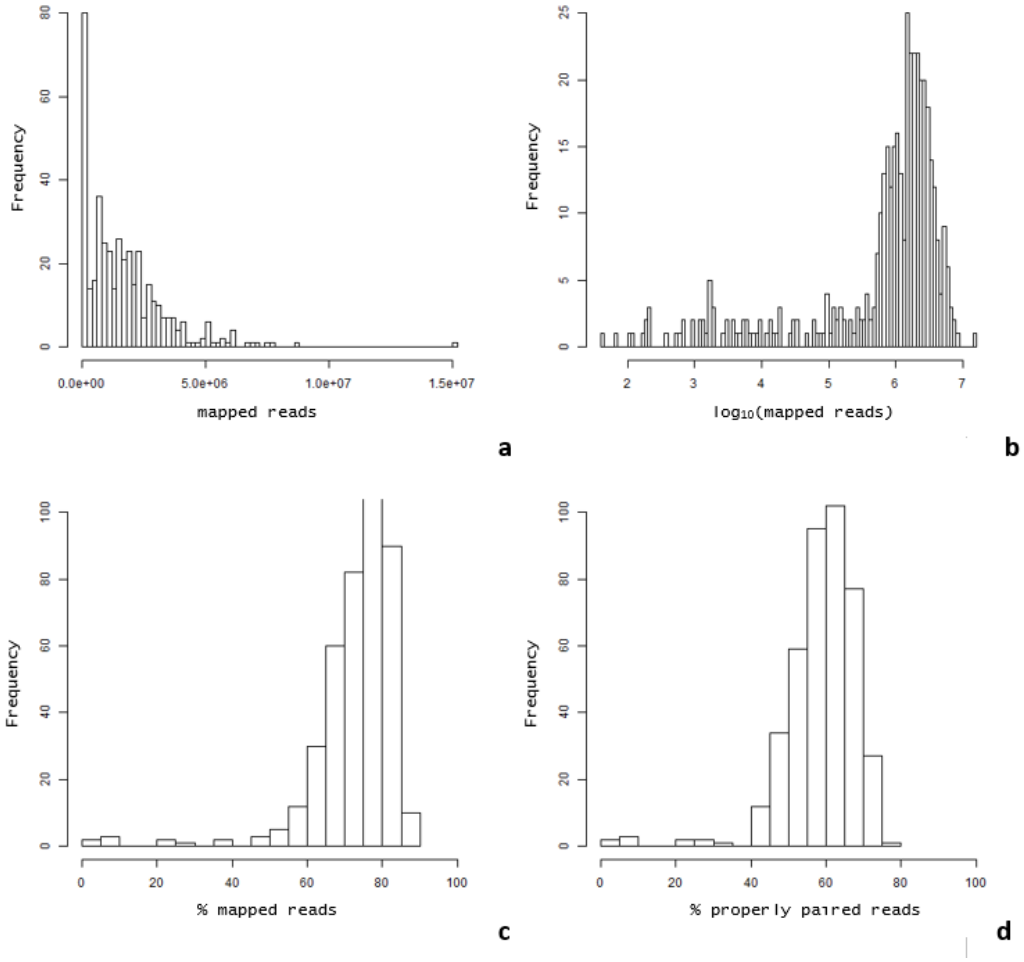

Figure 6. Mapped and properly paired reads in all 417 individuals generated with Flagstat. The number of reads which mapped to the reference genome assembly is given for all samples (n = 417) (a). Again, a logarithmic (log_{10}) (b) and also a percental distribution (c) of the number of mapped reads is given. Furthermore, the portion of properly paired reads in percent is presented for all individuals (d).

As expected from the ddRAD-Seq approach a large number of duplicated reads was found. In almost all samples the total sequence consisted of more than 80% non-unique sequences (Figure 7). The proportion of duplicated reads correlated positively (Spearman) with the number of reads per sample (Figure 7b). A saturation could be observed after approximately 100,000 reads, meaning that after these 100,000 reads almost only duplicated reads were produced.

Furthermore, overrepresented sequences, which made up more than 1% of an individual's total sequence, were found in all samples. All of these sequences could be ascribed to chloroplast and mitochondrial regions.

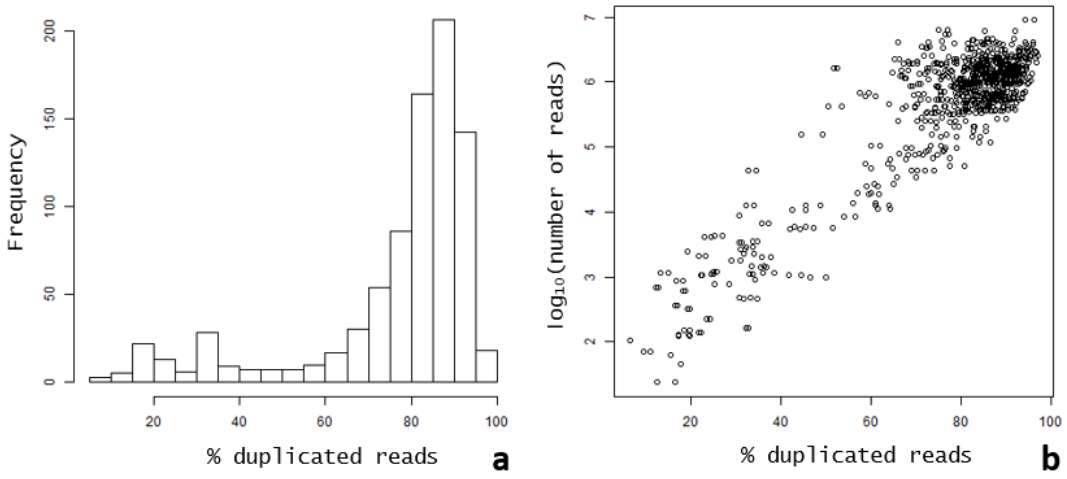

Figure 7. Proportion of duplicated reads in all 417 individuals. The percental proportion of non-unique sequences per sample is given for all individuals (n = 417) (a). Furthermore, the correlation (Spearman) between the number of duplicated reads and the total number of reads in a sample is presented in percent (b).

3.2. Filtering and variant calling

SNPs were called in each of the eleven populations. The total number of SNPs per population ranged from 161,006 to 331,784, whereby differences up to 171,000 loci were found (Table 4). The highest number of SNPs was called in a population of Czechia (czrb) (m), the lowest number of SNPs was found in the population from Bestwig (fort) (nm).

To only use loci from the nuclear genome for the population analysis, all polymorphisms mapping to the chloroplast genome and to repeat-masked regions were removed. Only 1-4 polymorphisms per population were mapped to the chloroplasts. In the populations from Bestwig (fort), Littfeld (litt) and in two populations from the Czech Republic (czra, czrc) only one SNP mapped in the chloroplast region. The highest number of SNPs, 4, on the chloroplasts were found in the populations of Bad Rippoldsau (badr), Clausthal Zellerfeld (CZ), Lautenthal (laut) and in the Czech Republic (czrb). The number of SNPs in repeated masked regions was similar among the populations and ranged from 39 in the Czech Republic (czra) to 59 in the north-eastern Blaibach (blai). The numbers of SNPs mapping to the chloroplast and repeat-masked regions are summarized in Table 5.

In total only four loci mapped to the mitochondrial genome. The four loci were found in the south-western population from Bad Rippoldsau (badr; nm), in the north-eastern population from Blaibach (blai; nm), and one in the Czech populations czra (nm) and czrc (m) respectively. Since only these four loci were found in the mitochondrial region, reads mapping to the mitochondrial genome were not excluded

After removing SNPs which mapped to on chloroplasts and repeat-masked regions, the mean number of loci per population was not very different from the original mean number of polymorphisms (Table 4). A subsequent exclusion of multiallelic loci and SNPs with quality scores below 100 lead to a reduction of about more than 50%. The number of biallelic SNPs ranged from 83,829 in the population from Bestwig (fort) to 172,345 in the population czrb from the Czech Republic. This removing of all multiallelic sites and sites with low quality reduced the distinction in the number of loci between the populations from 171,000 to a maximal difference of 88,516 SNPs.

Table 4. Number of SNPs per population. The number of loci called with Freebayes is given for each population. Only polymorphisms mapping to the nuclear genome were used for analysis (excluding chloroplast and repeat-masked regions). Furthermore, the number of SNPs after exclusion of loci with more than one alternative allele (multiallelic) and a quality-score below 100 is given. Populations from metalliferous sample sites are shaded in grey.

Population	Abbre-viation	Total loci	Number of loci not in chloroplasts or repeat masked regions	Number of biallelic SNPs with QUAL > 100.0
Bad Rippoldsau	badr	210,586	210,528	110,134
Blaibach	blai	234,719	234,657	117,532
Clausthal Zellerfeld	clau	236,461	236,408	117,599
Fortfun/Bestwig	fort	161,006	160,963	83,829
Lautenthal	laut	217,239	217,190	112,344
Littfeld	litt	181,328	181,288	93,692
Vienenburg	vieb	221,443	221,396	111,384
Wulmeringhausen	wulm	167,430	167,386	87,339
Bohemia Forest CZ1	czar	208,171	208,130	104,890
Bohemia Forest CZ2	czrb	331,784	331,732	172,345
Bohemia Forest CZ3	czrc	181,046	180,996	91,383

Table 5. Number of polymorphisms on chloroplasts and in repeat-masked regions. The number of polymorphisms which mapped to the chloroplast genome or to repeat-masked regions is given for each population. Again the populations from metalliferous sample sites are shaded in grey

Population	Abbreviation	Number of SNPs on chloroplasts	Number of SNPs in repeat-masked regions
Bad Rippoldsau	badr	4	54
Blaibach	blai	3	59
Clausthal Zellerfeld	clau	4	49
Fortfun/Bestwig	fort	1	40
Lautenthal	laut	4	48
Littfeld	litt	1	49
Vienenburg	vieb	3	40
Wulmeringhausen	wulm	2	47
Bohemia Forest CZ1	czra	1	39
Bohemia Forest CZ2	czrb	4	43
Bohemia Forest CZ3	czrc	1	43

For the analysis a set of loci common to all populations was created. To achieve an increase in the number of common loci individuals with more than 50% missing loci were excluded.

The number of private loci was determined for each population on the basis of these individuals with less than 50% missing loci (n=277). In three cases more than 10% of the loci were exclusively found in one of the populations (Table 6). In the populations from Bad Rippoldsau (badr; nm) and Blaibach (blai; nm), which grow in western and eastern of southern Germany 33.2% respectively 13.3% private sites were found. In the population czrb (nm) from the Czech Republic almost half (46.5%) of the loci were exclusively found in this population. Due to this high number of private sites the number of loci common to all populations was low. As a consequence the population czrb was excepted from the further analysis.

4,565 -74,243 loci could only be found in one individual (singleton) of a population. The population in Wulmeringhausen (wulm; m) possessed the lowest number of singletons. With

43.1% of its total number of loci, the highest number of singletons was found in the Czech population czrb. In the other populations singletons only constituted 5-10% of the total loci. This amount of singletons confirmed the outlier character of population czrb.

After combining the file from multi-sample calling with the one including the private sites of all populations 29,553 common loci could be remained. From these 1,249 were found in exclusively one individual and might thus arise from technical artefacts.

After the quality control and the filtering steps 260 individuals from 10 populations with 21,711 common loci remained for the population analysis. The number of samples before and after filtering is summarized in Table 7.

Table 6. Number of private sites in each population. The frequency as well as the percentage of private sites are presented (calculations were made on the basis of biallelic SNPs present in individuals with less than 50% missing loci (n=277)). Furthermore, the number and relative frequency of singletons (loci which were exclusively found in one individual) and doubletons (loci of one individual which were homozygous for the minor allele) are given. For better illustration populations from metalliferous sample sites are shaded in grey.

Population	Abbreviation	total number of biallelic SNPs	Number of private sites	% private sites	Number of singletons and doubletons	% singletons and doubletons	Number of singletons	% singletons
Bad Rippoldsau	badr	110,134	36,575	33.2	10,336	9.4	5,924	5.4
Blaibach	blai	117,532	15,587	13.3	12,640	10.8	7,608	6.5
Clausthal Zellerfeld	clau	117,599	7,007	6.0	12,639	10.7	7,717	6.6
Fortfun/Bestwig	fort	83,829	2,928	3.5	14,566	17.4	8,747	10.4
Lautenthal	laut	112,344	5,920	5.3	12,485	11.1	7,232	6.4
Littfeld	litt	93,692	2,527	2.7	8,850	9.4	5,186	5.5
Vienenburg	vieb	111,384	5,412	4.9	14,046	12.6	8,436	7.6
Wulmering-hausen	wulm	87,339	4,783	5.5	7,682	8.8	4,565	5.2
Bohemia Forest CZ1	czra	104,890	7,402	7.1	11,241	10.7	6,630	6.3
Bohemia Forest CZ2	czrb	172,345	80,125	46.5	92,729	53.8	74,243	43.1
Bohemia Forest CZ3	czrc	91,383	3,743	4.1	10,748	11.8	5,933	6.5

Table 7. Number of samples before and after filtering. Summarized are the number of samples in each population before (n = 379) and after (n = 260) the exclusion of samples with more than 50% missing loci (population czrb is already removed). To roughly estimate the maximal possible number of pairs in a population the number of the original and remaining samples are separated into individuals with a distance of 3m (o) and their corresponding partners (y) (see sampling method). Populations from metalliferous soils are shaded in grey.

Population	Abbreviation	Total number of samples	Total number of samples after filtering	Sub-division	Number of samples	Number of samples after filtering
Bad Rippoldsau	badr	38	29	y	18	14
				o	20	15
Blaibach	blai	36	28	y	18	15
				o	18	13
Clausthal Zellerfeld	clau	40	31	y	20	17
				o	20	14
Fortfun/ Bestwig	fort	38	13	y	18	5
				o	20	8
Lautenthal	laut	39	28	y	18	12
				o	20	16
Littfeld	litt	39	28	y	20	13
				o	19	15
Vienenburg	vieb	38	27	y	19	17
				o	19	10
Wulmering-hausen	wulm	40	28	y	20	13
				o	20	15
Bohemia Forest CZ1	czra	36	28	y	18	14
				o	18	14
Bohema Forest CZ3	czrc	35	20	y	18	9
				o	17	11

3.3. Coverage

The coverage of each biallelic locus was calculated for every individual. Between the individuals of a population a high variability in the coverage was found (see exemplary Figures 8a and b from population badr). The median coverage of the individuals from the population from Bad Rippoldsau (nm), for instance, ranged from around 9 to 90.

Despite an equally assumed amplification of loci within an individual a high variability in the coverage was also found among sites of the same sample. Within some individuals the coverage of the loci ranged from below 10 to above 6,000. For some loci even a coverage of above 8,000 and 10,000 was observed. However, it was noticed that the differences in the per locus coverage appeared to be consistent among the samples, meaning that for the same locus a similar coverage was found in different individuals (shown for the first 1000 loci in Figure 8c). By testing for a correlation between the loci of different samples the genome wide Spearman correlation coffiecients were rather high (exemplary Figure 9 from the population badr). This implies an actual correlation between the loci of different individuals and confirms the observed consistency in the coverage differences between the loci of different individuals. Due to this correlation the data remained comparable and could be used for the analysis.

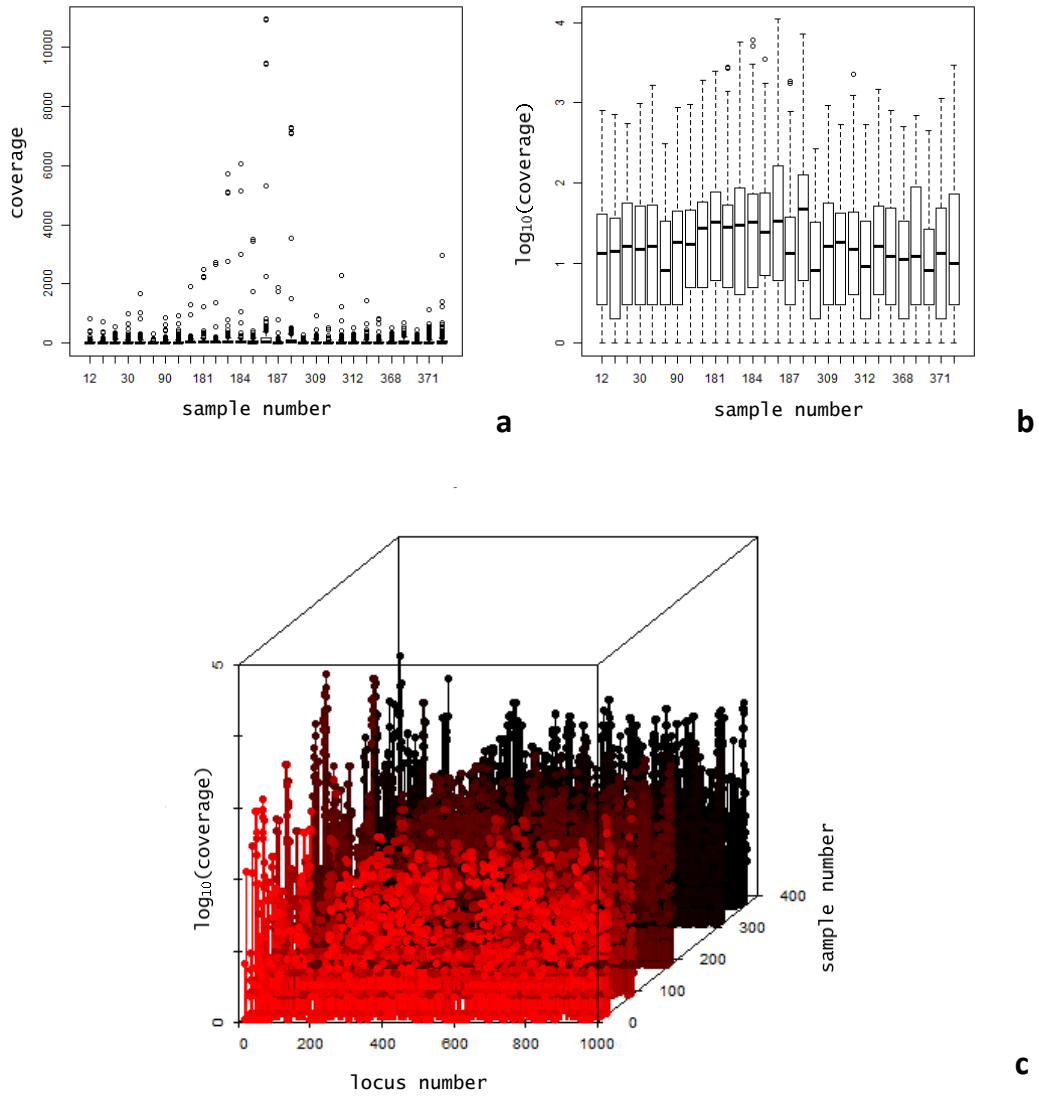

Figure 8. Coverage variability in filtered populations. The coverage is given for each sample (n=29) for the first 1000 SNPs in population badr (a). For this missing genotypes were set to NA. Furthermore, the coverage is presented log_{10}-transformed (b). Again missing genotypes were set to NA. In addition, a 3D scatterplot with the coverage per locus per sample is given for the first 1000 SNPs of population badr (c). The coverage is log_{10}-transformed. Here the coverage variability is presented for the population badr (n = 29), however the same extent of variability was observed in the other populations.

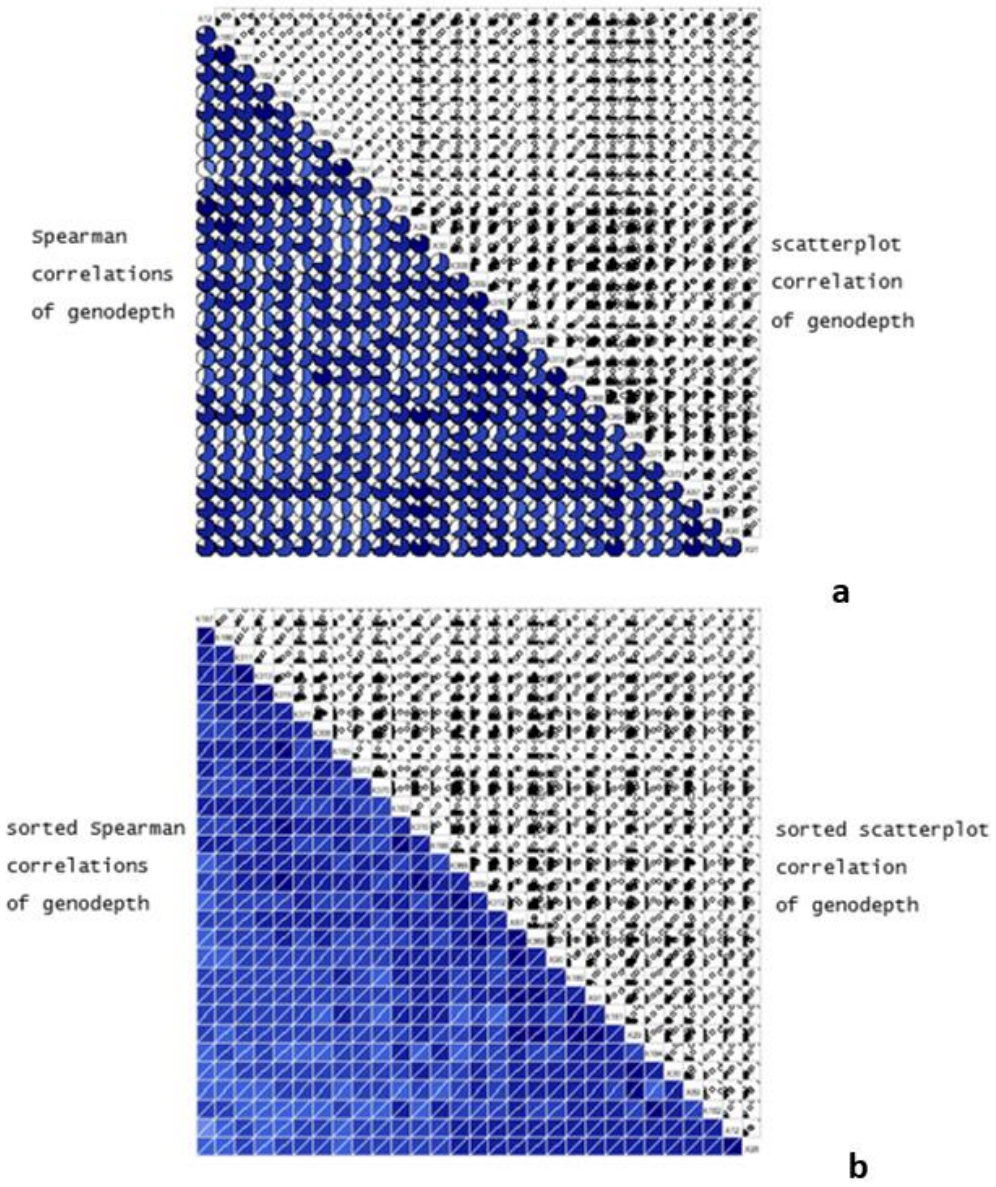

Figure 9. Correlations of coverage in filtered populations. Spearman correlations of coverage are given exemplarily in population badr (n=29) across all biallelic SNPs. Loci with missing genotypes were removed. The coverage per loci of an individual was correlated with the coverage of the same loci in a second individual (a). Scatterplots of the correlations are given in the right sphere. Pie charts, left sphere, represent the extent of correlation. The higher the correlation between to individuals, the darker is the blue colour of the pie chart. Compared individuals are listed in the diagonal between the scatterplots and the blue correlation charts. Furthermore the same correlations are given as PCA ordered Spearman correlations (b), meaning that individuals with a higher correlation are placed together (depth for each genotype was calculated using VCFtools).

3.4. Comparison of populations

To compare the populations a principal component analysis was performed on the basis of 3,154 common loci. Additionally to this PCAs were performed with common loci possessing an outlier F_{ST}-value above 0.4, 0.5 and 0.6. For all sets of loci the level of explained variance dropped after the fourth principal component and remained almost unchanged over the following components. Thus, the PCAs were performed with the first 4 principal components (Figure 10).

Five separate clusters were formed in the principal component analysis (Figure 11). The populations from the southern non-metalliferous sites of Germany, Bad Rippoldsau (badr, yellow) and Blaibach (blai, red) did not cluster with other populations.The populations from Czechia clustered together (blue and green). The three populations from the metalliferous sites Clausthal Zellerfeld (clau, purple), Lautenthal (laut, brown) and Vienenburg (vieb, turquoise) in the north-eastern of Germany formed one cluster with the population from the north-western metalliferous site in Littfeld (litt, pink). The remaining population from the north-eastern metalliferous location in Fortfun/Bestwig (fort, black) clustered separately with the population from its neighbouring metalliferous site in Wulmeringhausen (wulm, orange). Two individuals from the population in Vienenburg were found in the cluster of Fortfun/Bestwig and Wulmeringhausen, whereas one sample from the population Fortfun/Bestwig clustered with the individuals from the metalliferous sites Clausthal zellerfeld, Lautenthal, Littfeld and Vienenburg. This clustering retained in PCAs with different principal components.

Since populations from non-metalliferous sites formed distinct clusters or clustered together with individuals from heavy metal polluted soils the overall clustering pattern did not match a differentiation on the basis of heavy metal contamination. A separation due to the presence of cadmium could thus be excluded. When focussing on one location at which a population from a metalliferous and a non-metalliferous site clustered together, an effect of cadmium could be observed: individuals from the metalliferous soil separated from those growing on the non-metalliferous site. A separation due to the cadmium contamination could be found between the populations from the Czech Republic (czra, czrb) and the populations from Wulmeringhausen and Fortfun/Bestwig in the northern of Germany, which both grew in close geographical distance to each other. Due to this geographical and genetical similarity the population pairs from the Czech Republic Wulmeringhausen and Fortfun/Bestwig respectively were used to investigate the influence of heavy metal pollution on the degree of inbreeding (See 3.5.).

By comparing the clustering pattern with the distribution of the sample sites a clustering according to the population's geographic position could be revealed. Populations from Germany and the Czech Republic did not cluster together. Furthermore, within Germany the populations appeared to separate into northern and southern sample sites (cf. Figure 1). The further rejects the assumed differentiation due to heavy metal contamination and indicates a separation by distance.

With 3,154 common loci, the first two principal components only could explain 18.5% of the variance. By selecting for loci with a high F_{ST} (> 0.4, 0.5, 0.6) and thus reducing the noise for the PCA the percentage of explained variance kept increasing to above 70% (n_{loci}=36) (Figure 12-13). Even after reducing the number of loci the clustering pattern could be retained, especially when using the first principal components. This implies that the most differentiated loci carried all information for a cluster patterning according to this differentiation by distance.

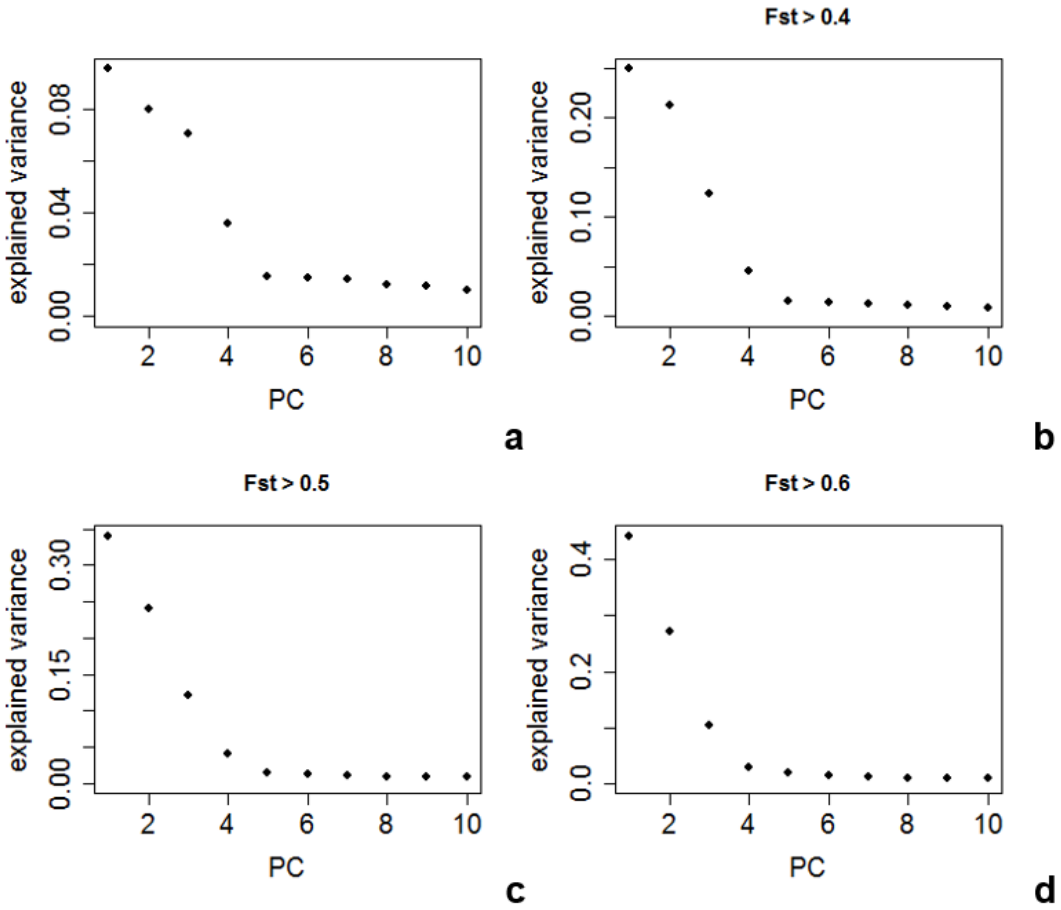

Figure 10. Explained variance. The explained variance is given for the first 10 principal components (PC) for a principal component analysis with 3,154 loci (a). Furthermore the explained variance is given for principal components of PCAs only using loci with a F_{ST} above 0.4 (n= 264) (b), 0.5 (n = 88) (b) and 0.6 (n= 36) (c).

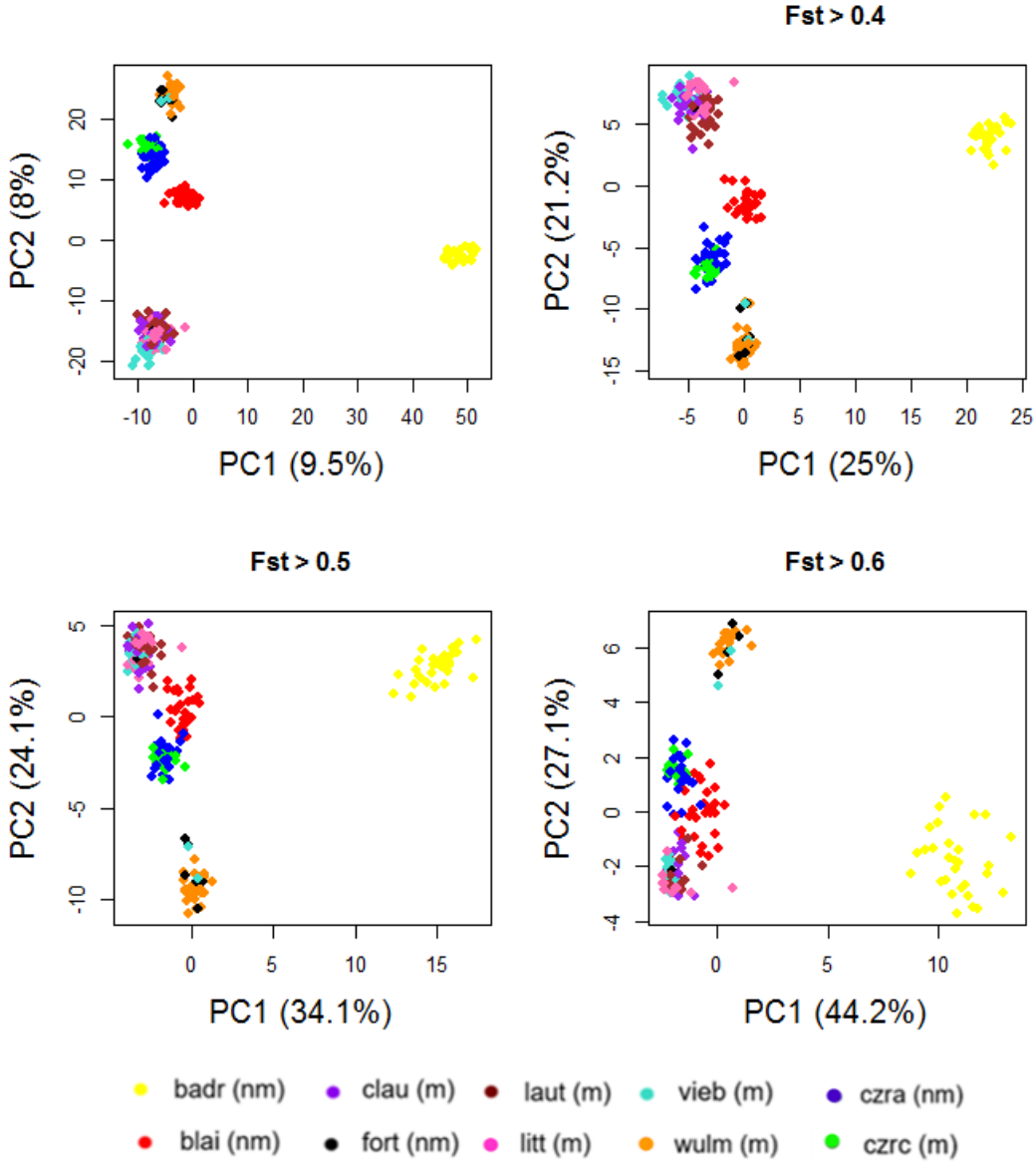

Figure 11. PCA with components explaining most of the variance. Principal component analyses on the basis of 3,154 loci (a) and for loci with F_{ST}-values above 0.4 (n = 264) (b), 0.5 (n= 88) (c) and 0.6 (n= 36) (d) are given for the first two principal components (PC). The percentage of explained variance is given in brackets for the respective principal component. In the legend populations from metalliferous sites are labelled with m, populations from non-metalliferous locations with nm.

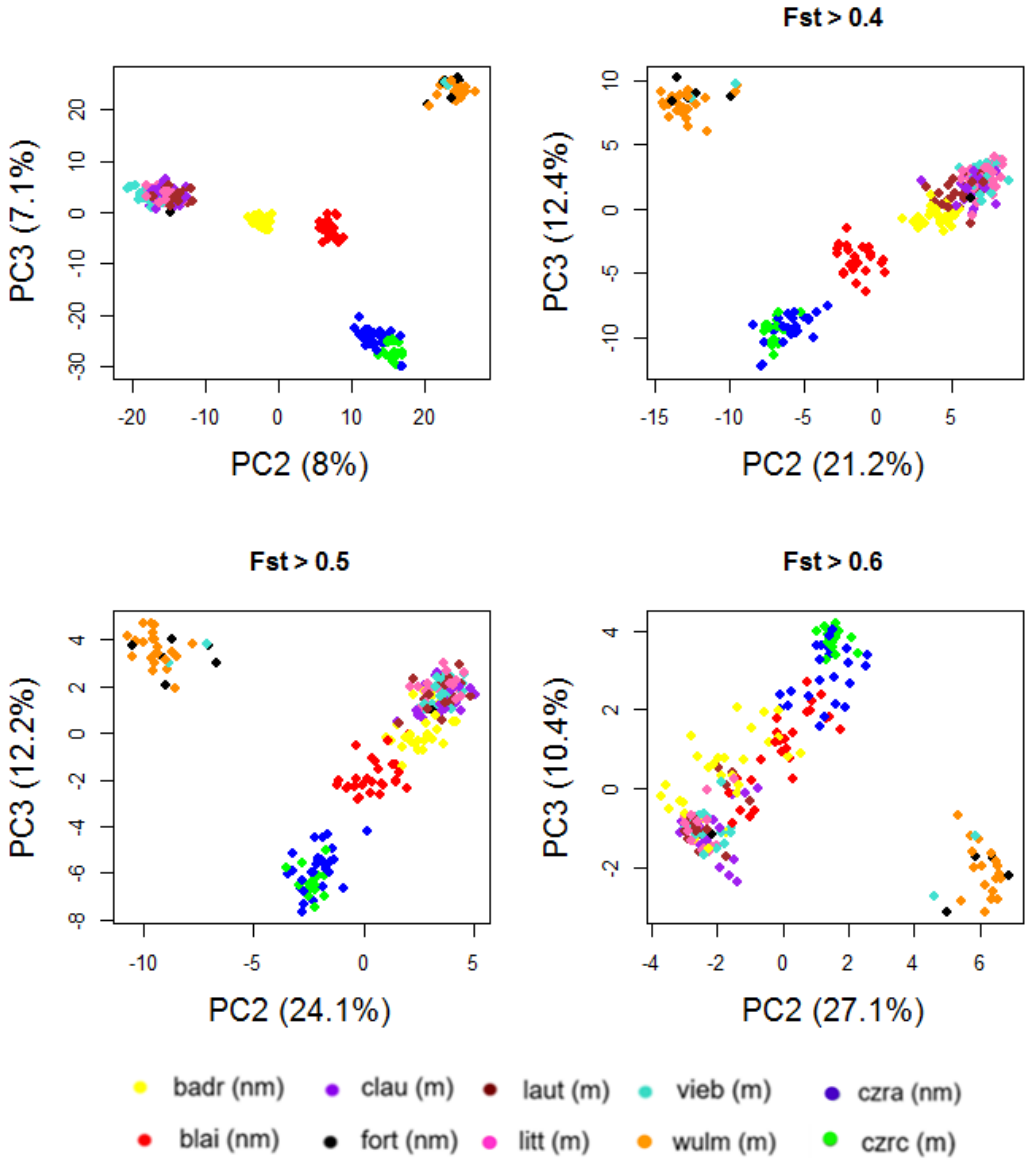

Figure 12. Second and third component of principal component analysis. A principal component analysis on the basis of 3,154 loci (a) is given for the second and third principal component (PC). Furthermore PCAs performed on the basis of loci with F_{ST}-values above 0.4 (n = 264) (b), 0.5 (n= 88) (c) and 0.6 (n= 36) (d) are shown. The amount of explained variance is given in brackets for the respective principal component. Populations from metalliferous sample sites are denoted as m, populations from non-metalliferous locations as nm.

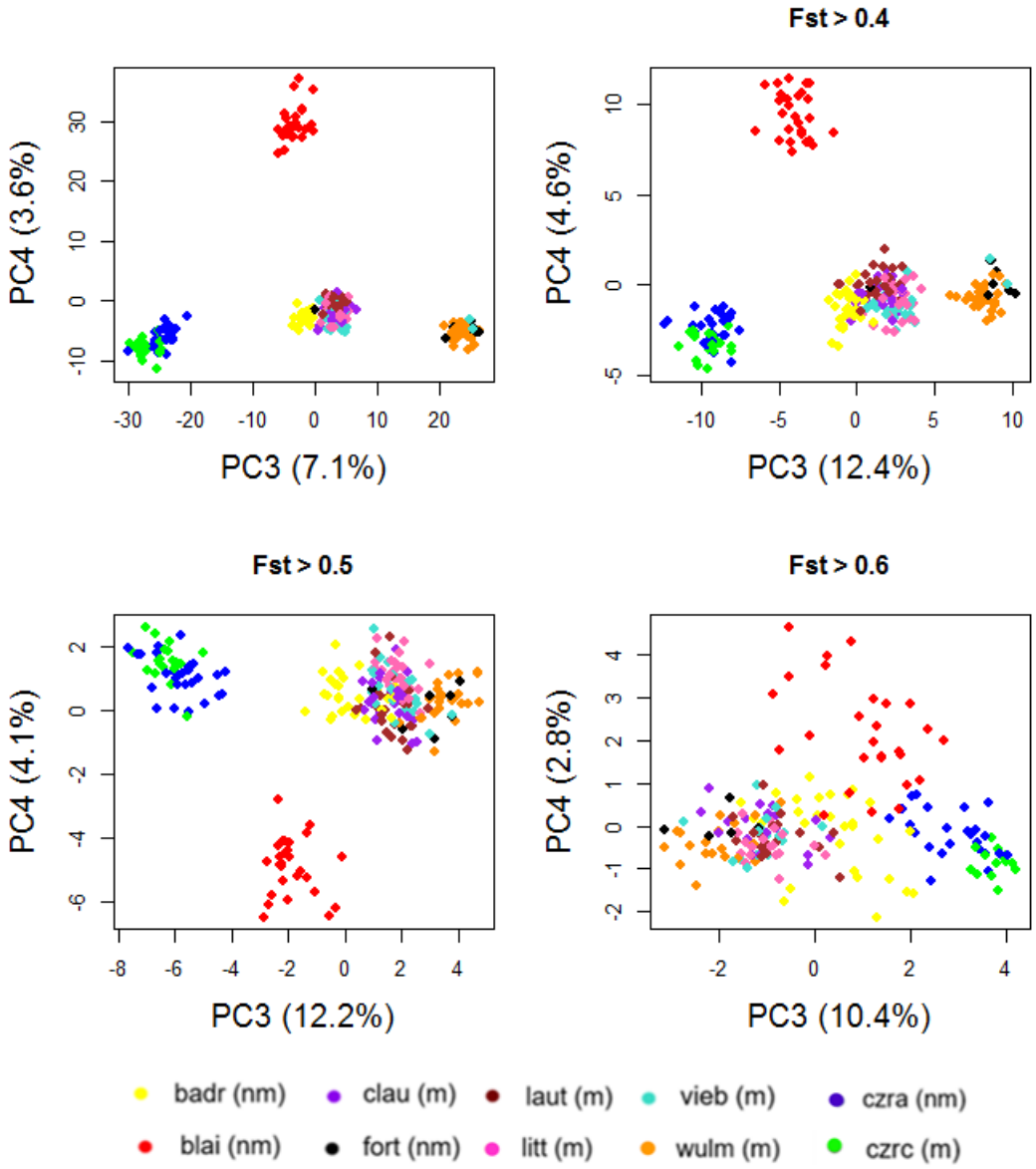

Figure 13. Third and fourth principal component of principal component analysis. A principal component analysis on the basis of 3,154 loci (a) is given for the third and fourth principal component (PC) Moreover, PCAs using loci with F_{ST}-values above 0.4 (n = 264) (b), 0.5 (n= 88) (c) and 0.6 (n= 36) (d) are given. The percental amount of explained variance is given in brackets for the respective principal component. Populations from sample sites with metalliferous soils are labelled with m, populations from non-metalliferous locations are denoted as nm.

The differences between the individual populations were further investigated by performing pairwise comparisons on the basis of the 3,154 common loci. In these pairwise comparisons the F_{ST}-values ranged from 0.0128 to 0.0403 (Table 9). No population pair with a F_{ST} value above 0.0403 was found. The highest difference was found between the populations from Blaibach (blai, nm), which was located in the south-east of Germany, and Vienenburg (vieb, m) from the north-east (cf. Figure 1). The most similar populations were from the south-western Bad Rippoldsau (badr, nm) and the south-eastern Blaibach (blai, nm).

No differences in the pairwise F_{ST}-values could be found between the populations from metalliferous and non-metalliferous sample sites. Populations from metalliferous soils were not more similar or different than populations from non-metalliferous locations. This further confirms a population differentiation independent from the contamination status of the soil.

The clustering pattern of the PCA could not be reproduced with the pairwise comparisons. Overall no difference between the populations from different clusters and

populations which clustered together could be found. In some comparisons the lowest F_{ST}-value was indeed found between populations of the same cluster. The F_{ST}- value of the pairwise comparison between the north-western populations from Fortfun/Bestiwg (fort, nm) and Wulmeringhausen (wulm, m), for example, was lower than in comparisons with non-clustering populations, but only at the third decimal place. For clusters which where formed by more than two populations the lowest F_{ST} was indeed found between two members of the same cluster, however the same population possessed pairwise F_{ST}-values with clustering populations that were higher than when comparing it with a non-clustering population (e.g. Vienenburg, vieb).

In some cases in which more than one population formed a cluster the lowest differentiation was found with a population which grew at a short distance. The geographical distance between two populations, however, had no significant impact on the population differentiation and therefore the height of the F_{ST}-values (liner mixed model, $p > 0.05$): population pairs growing in larger distance to each other were not more different than two populations from closer sites.

Table 8. Pairwise F_{ST} between all populations. F_{ST} is given between all pairs of 10 populations (n = 260). Czrb was excluded due to a high proportion of private sites. Populations growing on metalliferous soils are shaded in grey. badr = Bad Rippoldsau, blai = Blaibach, clau = Clausthal Zellerfeld, fort = Fortfun/Bestwig, laut = Lautenthal, litt = Littfeld, vieb = Vienenburg, wulm = Wulmeringhausen, czra = Bohemia Forest CZ1, czrc = Bohemia Forest CZ3.

	badr	**blai**	**clau**	**fort**	**laut**	**litt**	**vieb**	**wulm**	**czra**	**czrc**
badr	0									
blai	0.0128	0								
clau	0.0150	0.0170	0							
fort	0.0355	0.0374	0.0324	0						
laut	0.0237	0.0262	0.0161	0.0356	0					
litt	0.0305	0.0309	0.0248	0.0267	0.0240	0				
vieb	0.0370	0.0403	0.0268	0.0256	0.0205	0.0218	0			
wulm	0.0222	0.0242	0.0230	0.0220	0.0323	0.0233	0.0299	0		
czra	0.0158	0.0187	0.0172	0.0315	0.0253	0.0281	0.0311	0.0177	0	
czrc	0.0179	0.0179	0.0184	0.0241	0.0279	0.0243	0.0350	0.0203	0.0230	0

To test if loci which possessed high F_{ST}-values also strongly contributed to the population differentiation the F_{ST}-values of 3,916 loci were compared with the respective squared loadings of the first principal component (generated with R package 'adegenet'). Loci with the highest F_{ST}-values did not possess high squared loadings. (Figure 11). To confirm these observations a Spearman correlation test was performed. However, no genome wide positive correlation (Spearman, $p > 0.05$) between the F_{ST}-value and the squared loading of a locus could be found.

For the identification of sites which possessed both, a high F_{ST} and a high squared loading value, only loci with the top 10% F_{ST}-values and squared loadings respectively were selected. By just using sites with high F_{ST} and squared loadings only 11 out of 396 loci which possessed both, a F_{ST} and squared loading within the top 10% of the values, were found. Also within the top 10% loci, F_{ST} and squared loading of a locus was not significantly correlated (Spearman, $p > 0.05$). Thus, the loci with the highest F_{ST}-values did not predominantely contribute to the differentiation of the populations.

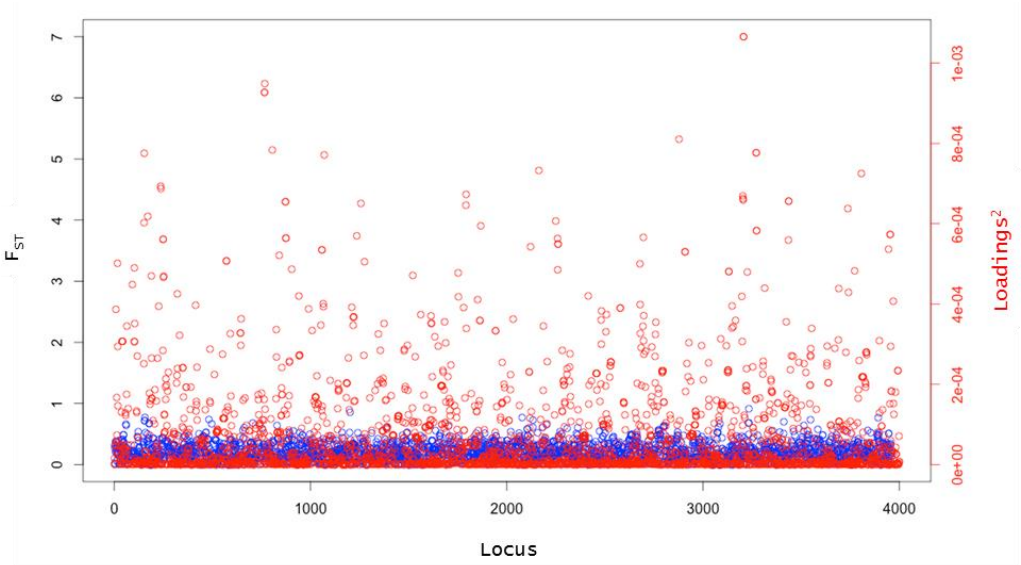

Figure 14. Squared loadings and F_{ST} for each locus. The squared PCA loadings (red) and F_{ST} (blue) for each common locus (n = 3,961) were compared to find loci which strongly contribute to the population differentiation.

The genetic variation within the populations was compared by calculating the mean level of heterozygosity. For this, only individuals with less than 50% missing loci were used. The level of heterozygosity was similar among the populations (Table 9). The lowest extent of heterozygosity, 0.2324, was found in the population from the metalliferous sample site in the Czech Republic (czrb). With 0.2557, the highest degree of heterozygosity was measured in the population from Littfeld (litt), which also represented a metalliferous location. No effect of heavy metal pollution on the degree of heterozygosity was found. Populations from metalliferous soils did not show a lower or higher extent of heterozygosity. The presence of cadmium, thus, did not affect the genetic variation of a population.

Table 9. Mean heterozygosity per population. The level of heterozygosity (calculated with VCFtools), is given for each population. Populations growing on metalliferous sample sites are shaded in grey.

Population	Abbreviation	Heterozygosity
Bad Rippoldsau	badr	0.2442
Blaibach	blai	0.2504
Clausthal Zellerfeld	clau	0.2413
Fortfun/Bestwig	fort	0.2398
Lautenthal	laut	0.2405
Littfeld	litt	0.2557
Vienenburg	vieb	0.2353
Wulmeringhausen	wulm	0.2381
Bohemia Forest CZ1	czra	0.2420
Bohemia Forest CZ3	czrc	0.2324

3.5. Inbreeding

The extent of inbreeding of each population (n=258) was determined. The populations differed significantly (linear mixed model, $p < 0.05$) in their degree of inbreeding. The median inbreeding coefficients ranged from 0.053 to 0.264. The population from the non-metalliferous site Bad Rippoldsau distinctly differed from all other populations and showed the highest median (0.264) and mean (0.268) inbreeding coefficient (Figure 15, Table 10). The lowest median (0.053) and mean (0.054) degree of inbreeding were measured in the population from the non-metalliferous sampling site in Blaibach (blai).

In general, no inbreeding coefficient above 0.4 was found. The highest measured inbreeding coefficient was calculated for one individual of the population from Fortfun/Bestwig and amounted to 0.375. Furthermore also negative inbreeding coefficients where observed. With -0.100 the lowest level of inbreeding was measured in an individual from the non-metalliferous site in Blaibach.

The effect of heavy metal pollution was investigated by comparing populations from metalliferous and metalliferous sample sites. The median and mean inbreeding coefficients of populations from contaminated sample sites were not higher than those of populations from non-polluted soils. In an overall comparison of all populations the inbreeding coefficient of populations from non-metalliferous sites was higher than in populations from heavy metal contaminated locations (Figure 16). However, this difference was not significant (linear mixed model, $p > 0.05$). The presence of cadmium in the soil had no impact on the degree of inbreeding over all populations.

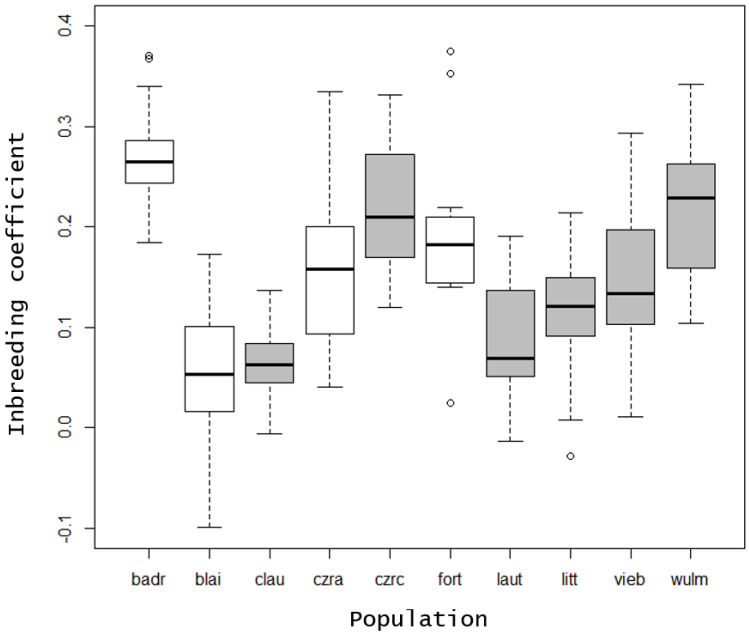

Figure 15. Inbreeding coefficient per population. The degree of inbreeding is given for each population (n = 258). Significant differences ($p < 0.05$) in the extent of inbreeding between populations are presented in Table 10. Populations from metalliferous sites are shown in grey.

Table 10. Differences in the degree of inbreeding among the populations. The inbreeding coefficients of populations are compared (n = 258). The mean inbreeding coefficient (least square mean) is given for each population. Furthermore, the upper and lower confidence limits (CL) are given. Used confidence level = 0.95. Populations from metalliferous sample sites are shaded in grey.

Population	Abbre-viation	least square means	Stand-ard error	Degrees of freedom	lower CL	upper CL
Bad Rippoldsau	badr	0.2684	0.0115	248	0.2459	0.2910
Blaibach	blai	0.0537	0.0117	248	0.0307	0.0766
Clausthal Zellerfeld	clau	0.0633	0.0112	248	0.0412	0.0855
Fortfun/Bestwig	fort	0.1920	0.0171	248	0.1583	0.2257
Lautenthal	laut	0.0868	0.0117	248	0.0639	0.1098
Littfeld	litt	0.1178	0.0117	248	0.0949	0.1408
Vienenburg	vieb	0.1446	0.0119	248	0.1212	0.1680
Wulmeringhausen	wulm	0.2220	0.0117	248	0.1990	0.2450
Bohemia Forest CZ1	czra	0.1622	0.0119	248	0.1389	0.1856
Bohemia Forest CZ2	czrc	0.2186	0.0138	248	0.1915	0.2458

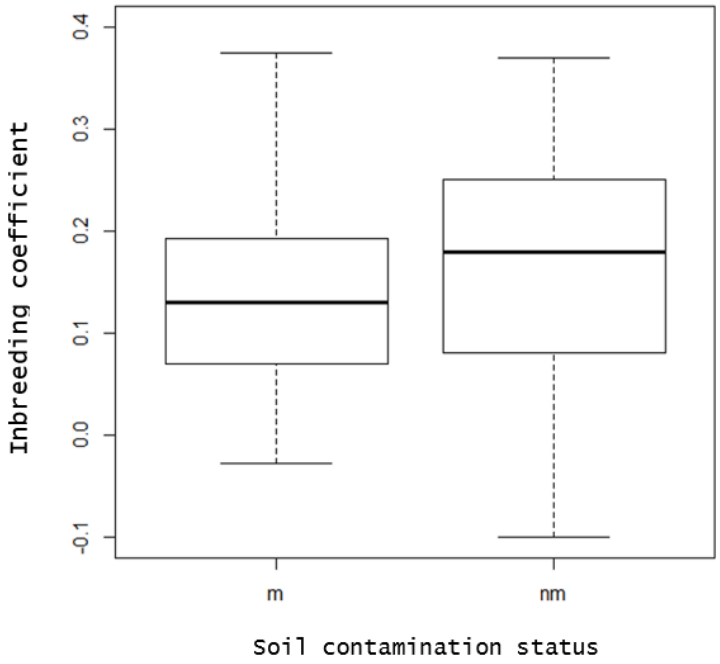

Figure 16. Inbreeding coefficients on metalliferous and non-metalliferous soils over all populations. According to the Cd-concentration of their site populations were categorized as metalliferous (m) and non-metalliferous (nm). The inbreeding coefficient dependent on the contamination status of a soil was calculated over all populations (n = 258).

Due to short distances between the population sites and the formation of a common cluster in the principal component analysis a high geographical and genetical similarity was shown for the population pairs from the Czech Republic (czra and czrb) and from Fortfun/Bestwig (fort) and Wulmeringhausen (wulm). Since in these two pairs a population from a metalliferous and one from a non-metalliferous sample site clustered together they were used to investigate the impact of cadmium on the inbreeding coefficient at lower spatial scale.

In the cluster formed by the populations from the Czech Republic individuals which grew on metal contaminated soil (czrc) were significantly (linear mixed model, $p < 0.05$) more

inbred before multiple testing than individuals from the non-metalliferous region (czra) (Figure 17). Here, the median inbreeding coefficient of populations from metalliferous soils amounted to 0.210 In contrast in populations from non-metalliferous sites a median inbreeding coefficient of 0.157 was observed.

The same effect of cadmium was found in the pairwise comparison of the populations from Fortfun/Bestwig and Wulmeringhausen (Figure 18). With a median inbreeding coefficient of 0.231 individuals from the metalliferous site in Wulmeringhausen were significantly (linear mixed model, $p < 0.05$) more inbred than the plants from the non-contaminated location in Fortfun/Bestwig (median inbreeding coefficient of 0.177).

At local scale the presence of cadmium did indeed affect the degree of inbreeding, with higher inbreeding coefficients on metalliferous soils.

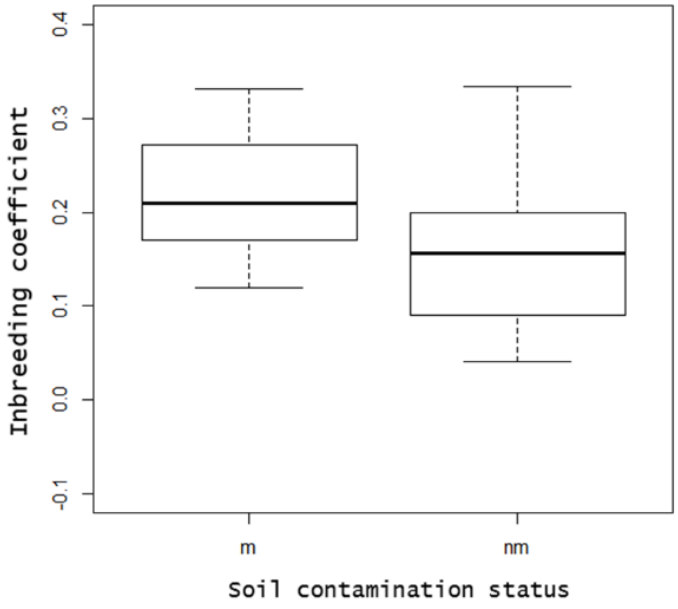

Figure 17. Influence of cadmium on the inbreeding coefficient in Czech populations. The degree of inbreeding dependent on the presence of cadmium is given for the non-metalliferous (m) sample site czra and the metalliferous location czrc (n = 48).

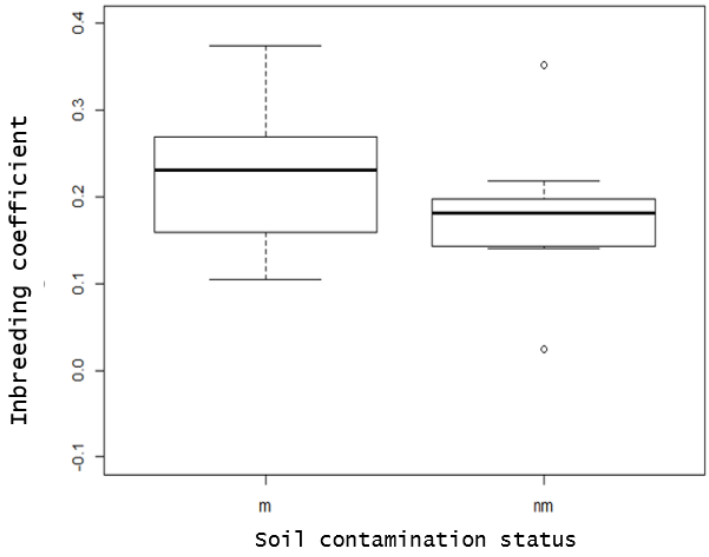

Figure 18. Influence of cadmium on the inbreeding coefficient in the populations wulm and fort. The degree of inbreeding is given for the north-western population from Wulmeringhauseb (wulm, m) and north-eastern population from Fortfun/Bestwig (fort, nm), which clustered together in the PCA (n = 41).

3.5.1. Impact of geographic location, population size and cadmium concentration

To test if the high variation in the inbreeding coefficients between the populations can be attributed to population site-specific conditions the impact of the geographic position (= geographic coordinates), the population size and the local cadmium concentration was determined with a linear mixed model (Figure 11).

Neither the longitudinal nor the latitudinal position of a population site had a significant effect (linear mixed model, $p > 0.05$) on the degree of inbreeding. For the altitudinal position also no significant influence (linear mixed model, $p > 0.05$) on the inbreeding coefficient could be observed. A geographical determination of the extent of inbreeding could thus be excluded.

Since the population size was given as a range of individuals the influence of the minimal, mean and maximal number of individuals was determined. Neither a minimal, mean nor maximal individual density at a population site affected the extent of inbreeding significantly (linear mixed model, $p > 0.05$). The population size as a cause of differences in the inbreeding content could be refuted.

The population sites differed in their soil's heavy metal content (see Table 1), however the actual cadmium concentration of a location had no significant impact (linear mixed model, $p > 0.05$) on the degree of inbreeding. This shows that the local level of cadmium could not explain differences between the inbreeding coefficients.

Table 11. Impact of geographical position, population size and cadmium concentration on the inbreeding coefficient (n=258). The effect of longitude, latitude and altitude on the extent of inbreeding are given. An effect of the number of individuals at a population site was determined for the minimal (Min), mean (Mean) and maximal (Max) population size. Furthermore, the impact of the cadmium concentration (Cadmium) is shown. For each effect also the change in the inbreeding coefficient per increasing degree, meter or µg/g respectively, is given (Estimate).

	Estimate	Standard Error	Degrees of freedom	p-value
Longitude	$-2.782e^{-02}$	$9.807e^{-03}$	2.884	0.069
Latitude	$-3.234e^{-02}$	$2.403e^{-02}$	2.696	0.272
Altitude	$9.766e^{-05}$	$9.786e^{-05}$	2.906	0.394
Min	$6.002e^{-05}$	$3.042e^{-05}$	2.881	0.147
Mean	$2.743e^{-05}$	$2.876e-^{05}$	2.878	0.413
Max	$-6.002e^{-05}$	$3.042e^{-05}$	2.880	0.147
Cadmium	$4.575e^{-03}$	$2.280e^{-03}$	2.007	0.138

The concentration of cadmium not only differed between but also within a population site (Supplementary Table S1). To investigate if these concentration heterogeneities in the soil of a location can be connected to differences in the inbreeding coefficient between populations the effect of variance in the cadmium concentration was determined using linear mixed models (residuals as well as random effects were normally distributed). The cadmium heterogeneity at a population site did not influence the degree of inbreeding significantly (linear mixed model, $p > 0.05$). A more genetically diverse population due to heterogeneous cadmium concentrations at a population site was thus rejected.

Since neither the location of a population site nor the individual number or the actual level of heavy metals could be connected to the high variation in the inbreeding coefficients between the populations an influence of the prevailing environmental conditions was excluded.

3.5.2. Model evaluation

Linear models were used to investigate differences between the populations and the impact of cadmium, geographical position and population size on the degree of inbreeding. To test the usability of linear models for the investigation of the inbreeding coefficients the distribution of the residuals and random effects was evaluated. Dependent on the population both the residuals as well as the random effects were normally distributed (Figure 19a). The usability of a linear model for the extent of inbreeding was thus confirmed. However, two outlier samples 112 and 134, which belonged to the populations in Clausthal Zellerfeld, the Bohemia Forest (czra) respectively, could be identified. These outliers were excluded from the further analysis (Figure 19b) leaving a final dataset of 258 individuals (n=258) for the investigation of the inbreeding coefficient.

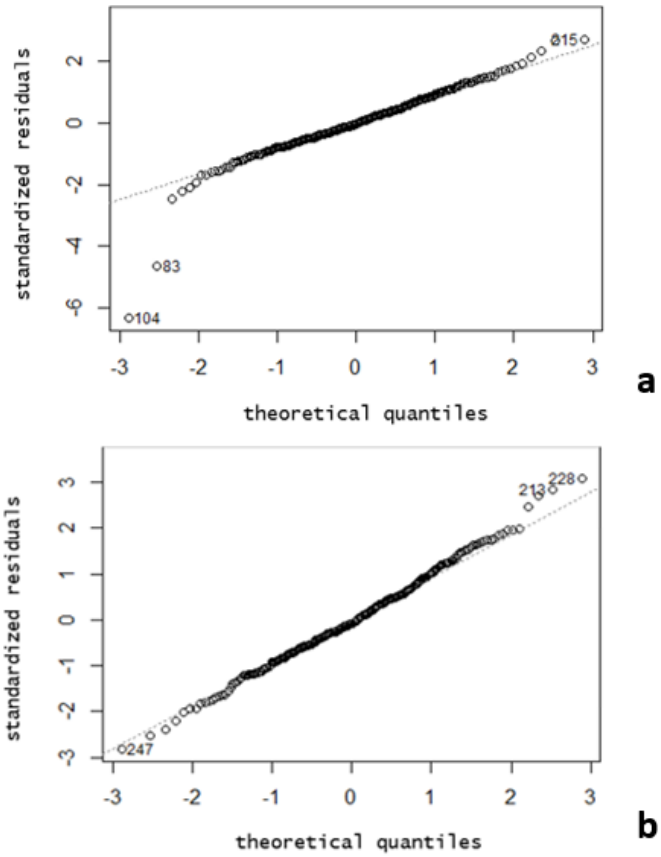

Figure 19. Residuals of inbreeding coefficient. A QQ-plots is given for the inbreeding coefficient dependent on the populations (a). The outliers 83 (=sample 112) and 104 (=sample 134) were excluded (b) leaving a final dataset of 258 individuals.

3.6. Relatedness

The degree of relatedness was estimated by calculating the kinship coefficients for all individual pairs of a population using VCFtools. The populations did not strongly differ in respect to their kinship coefficients (Figure 20). In all populations most of the individual pairs showed kinship coefficients between -0.1 and 0.2 (Figure 21). The median kinship coefficients of the populations ranged from 0.004 to 0.039 (Table 12). The population with the least related individuals grew on the metalliferous site in Vienenburg (vieb). With 0.49 also the highest kinship coefficient was measured between two individuals of this population. The highest median degree of relatedness was observed in the population form Wulmeringhausen (wulm).

In general in no population a kinship coefficient above 0.49 was measured. Not even between plants with an interindividual distance of 50cm (pairs) a kinship coefficient of 0.5 could be observed. This lack of individuals with a kinship coefficient of 0.5 indicates, that neither at metalliferous nor non-metalliferous locations, individuals were generated by clonal propagation.

To investigate if heavy metal pollution in the soil leads to an increased kinship coeffcient between the individuals of the respective site, the influence of cadmium on the distance of relatedness was tested. However, the residuals did not meet the assumptions for a linear mixed effects model. Not even after performing an arcsin, squared-root or a combined transformation of the data the usability of a liner mixed effects model could not be confirmed. As a consequence, the influence of a cadmium on the relatedness of individuals was determined only approximately. In an overall comparison the populations differed significantly (Kruskal-Wallis, $p < 0.05$) in respect of their kinship coefficients. To determine which of the populations differed significantly pairwise Wilcoxon tests were performed for all populations (Table 13).

In only 14 out of 45 pairwise comparisons a significant difference in the kinship coefficient was observed. The most significant difference was found between the populations from Wulmeringhausen (wulm) and Vienenburg, which both grew on cadmium-contaminated soils in the north of Germany. The kinship coefficients of population from Vienenburg differed significantly from almost all the other populations. In contrast, the north-western population from Fortfun/Bestwig (fort, nm) did not possess any significant difference from other populations.

From the 15 pairs in which both populations came from metalliferous soils 8 (53.3%) pairs showed a significant difference while the other 7 (46.7%) did not differ significantly. In the 6 pairs in which both pairs were collected from non-contaminated sites no significant difference was found. For 24 population pairs in which one population grew on metalliferous and the other on non-metalliferous soil only 6 (25%) pairs showed a significant difference. An effect of cadmium on the kinship coefficient could thus be rejected.

.

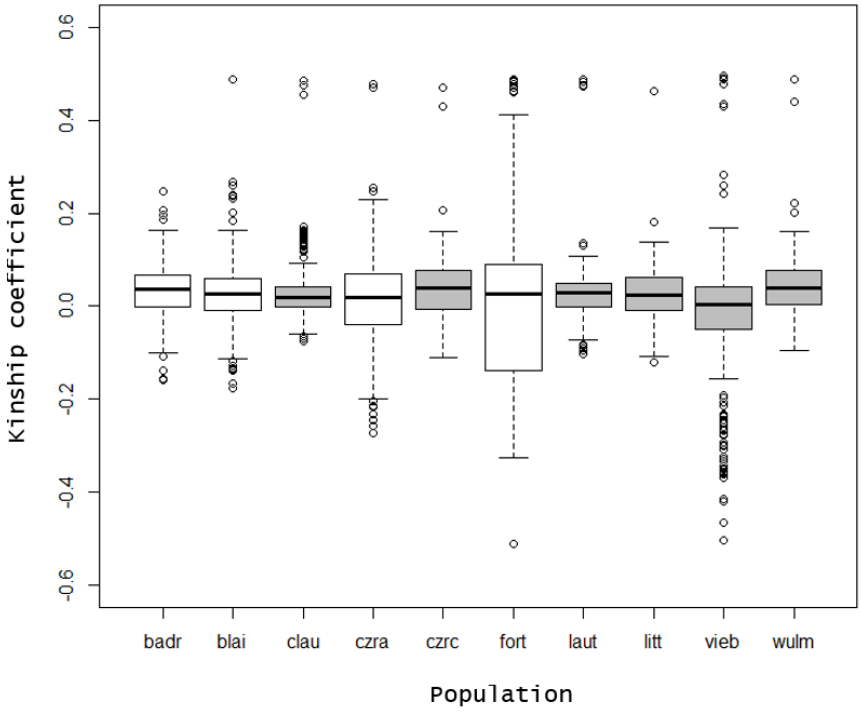

Figure 20. Comparison of kinship coefficients per population. The degree of relatedness is given for each population. Details on the distribution of kinship coefficients within a population are presented in Figure 21 and Table 12. Populations growing on metalliferous sample sites are presented in grey.

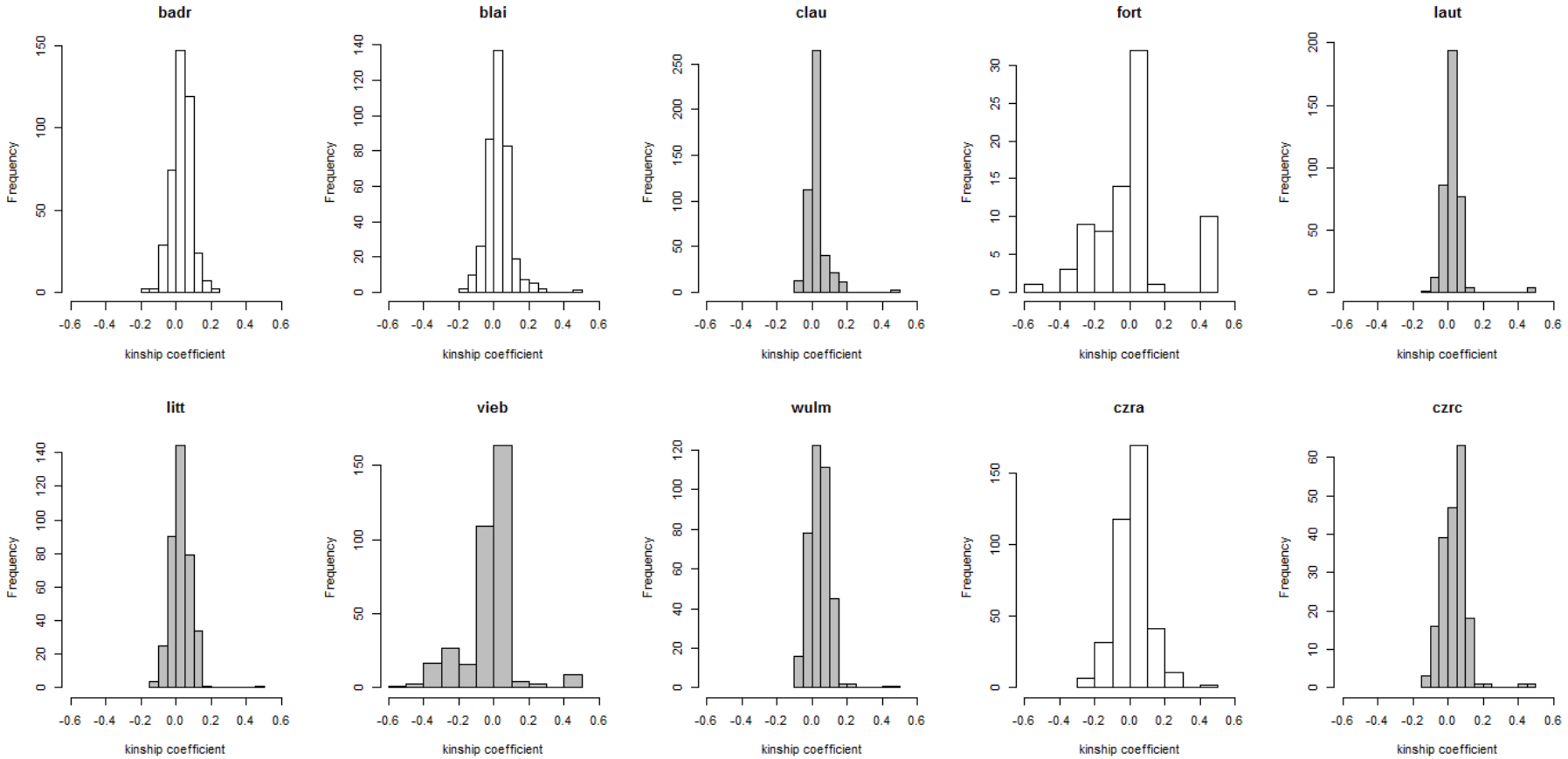

Figure 21. Frequency of kinship coefficients per population. The frequency distribution of pairwise kinship coefficients between individuals of a population is given. The minimal, median and maximal kinship coefficients of each population are summarized in Table 12. Populations from metalliferous sample sites are presented in grey.

Table 12. Summary statistics of kinship coefficients per population. The minimal (Min.), median and maximal (Max.) kinship coefficient is given for each population. Populations growing on metalliferous soils are again shaded in grey.

Bad Rippoldsau (badr)		**Blaibach (blai)**	
Min.	-0.159	Min.	-0.177
Median	0.037	Median	0.027
Max.	0.248	Max.	0.489
Clausthal Zellerfeld (clau)		**Fortfun/Bestwig (fort)**	
Min.	-0.076	Min.	-0.513
Median	0.019	Median	0.025
Max.	0.486	Max.	0.489
Lautenthal (laut)		**Littfeld (litt)**	
Min.	-0.104	Min.	-0.120
Median	0.028	Median	0.024
Max.	0.489	Max.	0.463
Vienenburg (vieb)		**Wulmeringhausen (wulm)**	
Min.	-0.504	Min.	-0.096
Median	0.004	Median	0.039
Max.	0.496	Max.	0.488
Bohemia Forest CZ1 (czra)		**Bohemia Forest CZ3 (czrc)**	
Min.	-0.273	Min.	-0.111
Median	0.019	Median	0.039
Max.	0.478	Max.	0.470

Table 13. Pairwise Wilcoxon tests in relation to the kinship coefficient. The p-values (corrected for multiple testing: Holm) of the Wilcoxon tests are given for each pair of all 10 populations. The populations from metalliferous sites are shaded in grey.

	badr	**blai**	**clau**	**fort**	**laut**	**litt**	**vieb**	**wulm**	**czra**	**czrc**
badr	0									
blai	1	0								
clau	0.0013	1	0							
fort	1	1	1	0						
laut	0.0986	1	1	1	0					
litt	1	1	1	1	1	0				
vieb	$5.26e^{-11}$	$2{,}59e^{-06}$	$4.02e^{-05}$	1	$1.35e^{-06}$	$4.08e^{-07}$	0			
wulm	1	0.0133	$3.23e^{-06}$	1	$2.92e^{-04}$	0.0274	$9.90e^{-15}$	0		
czra	0.0986	1	1	1	1	1	0.0114	$2.97e^{-04}$	0	
czrc	1	1	0.125	1	0.2649	1	$1.70e^{-07}$	1	0.1785	0

4. Discussion

4.1. Methylation-sensitivity of MspI

According to the original protocol of Poland et al. (2012) the restriction enzymes MspI and PstI were used to excise DNA fragments. MspI was used as a common cutting restriction enzyme. Besides a high cutting frequency in different species, it is described to be methylation sensitive (Shirasawa et al. 2016, https://www.neb.com/products/r0106-mspi, Korch and Hagblom 1986). Methylation sensitivity is a major issue when choosing a restriction enzyme. In methylation-sensitive enzymes the methylation of one or more bases of the restriction site can prevent a cleavage (Korch and Hagblom, 1986, https://www.neb.com/products/r0106-mspi). In MspI, for instance, cutting is inhibited by a methylation at the external C in the recognition sequence (Korch and Hagblom, 1986, https://www.neb.com/products/r0106-mspi). Therefore, fragments which actually possess the restriction site but are methylated will not be excised. These fragments will not be analysed which results in a reduced amount of loci (e.g. DaCosta and Sorenson 2014, Elshire et al. 2011, Baird et al. 2008, Poland et al. 2012). Moreover, also the sequence variability between individuals cannot be properly estimated since it is not clear whether a fragment is missing because of a methylation or because of the absence of the corresponding restriction site (Pelley 2012). However, in this project methylation sensitivity of the used restriction enzymes is not of concern since only SNPs common to all populations were used for the analysis. Some SNPs might have been lost due to inhibited cleavage but an impact on the data analysis can be excluded.

4.2. Sequence quality

Quality control was done for the sequences of all 417 individuals with FastQC.
FastQC is a useful tool to evaluate the quality of sequencing data. However, it was developed to examine the data from whole genome sequencing (Cusack 2016). In this master thesis a RAD-Seq approach was used which means, that in contrast to whole genome sequencing, only a small fraction of the genome was analysed (e.g. Poland et al. 2012, Peterson et al. 2012, Baird et al. 2008). When applying FastQC to RAD-Seq data it can be expected that several quality criteria will not be met.

The sequences of all samples showed good quality scores (Phred > 20) per base and per sequences.

Within the first 13bp of a read a bias in the sequence composition was observed. This bias can be related to the use of restriction enzymes and barcodes: Since the DNA was fragmented with restriction enzymes it was cut at a defined site. Therefore, the first bases of all reads were the same. Moreover, due to the use of barcoded adapters all fragments which came from one individual possessed the same barcode sequence at the beginning of the read.

In most of the samples a high number of duplicated reads (> 70%) was observed. This was anticipated from a ddRAD-Seq approach. Duplicated reads can be generated during library preparation. When loading the PCR products onto the flow cell ideally only one copy per fragment will bind (Ebbert et al. 2016). However, also additional copies originating from the same DNA molecule can hybridize with the flow cell (Ebbert et al. 2016). As a result several identical clusters form and the same fragment will be sequenced multiple times which leads to reads that represent PCR duplicates (Ebbert et al. 2016, Bansal 2017). Especially, when sequencing only a small proportion of the genome a saturation in the sequencing process can occur, meaning that after some time no new unique fragments are left. Consequently, several fragments will be sequenced multiple times and the number of duplicated reads and also overrepresented sequences increases. A saturation was observed after around 100,000 reads. This indicates that 100,000 reads would have been sufficient for the analysis.

Overrepresented sequences were found in all samples. All of the overrepresented sequences arose from non-nuclear genomic regions (chloroplasts and mitochondria). The quality control was performed on the raw sequencing data and fragments from the chloroplast were not yet excluded and could therefore contribute to the higher representation of some sequences.

A high variability in the coverage was observed between different samples and between different loci within an individual. The coverage variability between individuals could

imply that the excision of fragments with the used restriction enzymes was not equally successful in all samples. A correlation between the locus and its coverage was observed, meaning that similar coverage was found for a particular locus in different samples and thus making them comparable for data analysis. However, no explanation for the coverage variability between loci of an individual could be found in literature.

4.3. Population comparison

Five separate population clusters were formed in a principal component analysis (PCA). Despite a different metal content of the soil populations from Czechia clustered together. The populations from Bad Rippoldsau and Blaibach, which were located on non-metalliferous sites in the south of Germany, did not cluster with any other population. Northern populations from metalliferous sites clustered together, except for north-western population from Wulmerinhausen (wulm, m). Despite being from a metalliferous site wulm clustered with the population from the north-western non-metalliferous sample site Fortfun/Bestwig (fort, nm). However, the populations of wulm and fort were very closely located to each other. The formed clusters in the PCA did not correspond to the soils contamination status and thus exclude a differentiation on the basis of heavy metal pollution. By comparing the clustering of the populations with the geographic location of the population sites, the pattern could be associated to the geographic distribution (differentiation by distance). At local scale populations separated according to the contamination status of the sample site. After excluding the strong influence of the geographic location, indeed an effect of cadmium on the separation of populations could be shown for two population pairs.

Three individual samples were found in clusters of different populations. This distribution could result from mislabelling of the samples.

The F_{ST}-values used for the principal component analysis where computed with VCFtools. VCFtools calculates a global F_{ST}-value for each locus by including the variance between populations, individuals and between the gametes of an individual (Weir and Cockerham 1984). A biased clustering of the populations due to pairwise comparisons with a highly different population could thus be excluded.

According to the low pairwise F_{ST}-values a high extent of interbreeding between the populations (Wright 1949) appears to exist independent of metal content and geographic location. A high interbreeding rate means a regular exchange of genetic material. Butterflies, syrphids and bees, which are the most frequently mentioned pollinators for *Arabidopsis halleri* (Clauss and Koch 2006) can cover long foraging distances (Schulke and Waser 2001, Townsend and Levey 2005, Escaravage and Wagner 2004, Beekman and Ratnieks 2000) and it is very likely that the high content of shared genetic diversity results from pollen transport and exchange between all populations.

In PCA the squared loading of a locus states how much variance of a variable can be explained by the components (Abdi and Williams 2010). A correlation of the squared loadings and the F_{ST}-values would indicate loci which strongly contribute to the differentiation of the populations. However, no such correlation was found, supporting the assumption of an extensive genetic exchange between the populations

Alternatively, a high genetic similarity could also result from an origination from the same genetic lineage. Šrámková-Fuxová et al. (2017) showed that individuals from *A. halleri* in Europe originated from three different genetical lineages with a well geographical separation between the groups. One group, defined as the north-western group, covered Western Europe including Germany and the Czech Republic (Šrámková-Fuxová et al. 2017). A high genetical similarity between the populations investigated in this project could therefore also be due to the same origin. A recent colonisation of the respective sample site might also have lead to low genetic differentiation. Shortly after the colonisation of new habitats and sites populations originating from the same initial population will be genetically very similar, since no founder effect or genetic drift has yet lead to site specific genetical differences (Campbell et al. 2009, Arnold 2001, Choudhuri 2014). The low genetic differentiation between the observed populations could thus also result from a recent colonisation of the sampled locations. This is also compatible with the idea of an origination from a single genetical lineage.

Furthermore, in populations consisting of a high number of individuals the effect of genetic drift and thus the genetically divergence of the populations is reduced (Campbell et al

2009, Arnold 2001, Choudhuri 2014). A large population size could have also contributed to the low differentiation.

.

4.4. Influence of cadmium on inbreeding and clonal propagation

4.4.1. Heavy metals and the extent of inbreeding

Inbreeding denotes the mating between closely related individuals (reviewed by Keller and Waller 2002). The degree of inbreeding is calculated on the basis of homozygous loci in relation to heterozygous calls (Holsinger and Weir 2009). A negative inbreeding coefficient, which was found in some of the individuals, can be explained by an excess in the number of heterozygous loci (Holsinger and Weir 2009).

Since plants differ in their tolerance level towards heavy metals a restricted number of founder individuals was expected on metalliferous soils (Bert et al. 2000, Van Rossum et al. 2004, Macnair 2002) and a high degree of relatedness and inbreeding was anticipated.

Despite differences in the extent of inbreeding in different populations the presence of cadmium did not influence the frequency of mating between relatives significantly.

By comparing the inbreeding coefficient of populations clustering together in the PCA (czra and czrc, fort and wulm), an effect of cadmium on the inbreeding content could be observed at local scale: significantly higher inbreeding coefficients were found in the populations from Wulmeringhausen (wulm, north-western Germany) and the Czech Republic (czrc), both growing on the metalliferous soils. In both cases the metalliferous and non-metalliferous sample site were located in close distance and low F_{ST}-values where found for both population pairs.

The lack of metal induced inbreeding could be related to heterogeneities in the heavy metal concentrations of a location. Individuals with different tolerance levels could colonize a location. This would also increase the number of genetically different plants and could reduce the frequency of inbreeding in a population. Heterogeneities in the heavy metal concentrations were taken into account by taking multiple soil samples of a location. Furthermore, the impact of multiple different cadmium concentrations at a site was investigated but the heterogeneities in the cadmium content had no significant effect on the inbreeding coefficient. However, the exact location at which a soil sample was taken was not specified. Especially information about the distance between a spot of a soils sample and a sampled plant individual could not be provided. Since the concentration of cadmium and other heavy metals can vary at small spatial scale (Linhart and Grant 2006, Mattner et al. 2002 as cited in Bizuox and Mahy 2007) the analysis would have been more powerful if soil samples were taken over the whole sampling area with exact documentation of the location. Ideally, soil samples should be taken next to each sampled plant.

Alternatively, the lack of inbreeding could also support that no strict levels of tolerance exist. Genotypes which are adapted to lower metal contents might also be able to grow on soils with higher Cd concentrations (Pauwels et al. 2006). Even though they will not grow as successful and fast as individuals with a higher tolerance level, some of them could develop into adults and might reproduce (Meyer et al. 2015). This in turn would also result in a higher number of genetically different individuals.

The low degree of inbreeding on metalliferous sites can also be connected to the predominant outcrossing nature of the species (Clauss and Koch 2006, Llaurens et al. 2008). To some extent the self-incompatibility mechanisms of *A. halleri* reduce the frequency of inbreeding (as already reviewed by Charlesworth and Charlesworth 1987, Furstentau and Cartwright 2017) by preventing the fertilisation with pollen which possesses the same S-locus. Nonetheless, some individuals were more inbred than others and showed inbreeding coefficients up to almost 0.4. This observation could result from a limited seed dispersal. The fruits of *Arabidopsis* species are elongated siliques which dry until maturity (Dinneny and Yanofsky 2004, Roeder and Yanofsky 2006). At maturity the siliques dehisce (Dinneny and Yanofsky 2004, Roeder and Yanofsky, 2006). The seeds are released and get distributed by wind and water (Dinneny and Yanofsky 2004). However, the seeds of *Arabidopsis* do not possess any surface increasing structures for anemochory over longer distances (Roeder and Yanofsky 2004). They will be dispersed locally which leads to short distance between closely related individuals (Furstentau and Cartwright 2017). Additionally some of the pollinators just disperse pollen over short distances (e.g. Wratten et al. 2003) and some plants might receive

pollen from their neighbouring relatives. Provided that the S-loci are different the degree of homozygosity and inbreeding can be increased.

On the contrary, the pollinators also increase the number of different mating partners. As already mentioned, some of the pollinators can cover long ranges and will transfer pollen within the whole population and also over wider distances (Schulke and Waser 2001, Townsend and Levey 2005, Escaravage and Wagner 2004, Beekman and Ratnieks 2000). Pollen from distant populations can thus pollinate local plant individuals and the number of parental plants further increases. This, in turn, leads to less related offspring and a reduced inbreeding rate on metalliferous and non-metalliferous sites. The frequent exchange between different populations could be confirmed by the low F_{ST}-values found between all populations (see population comparison).

4.4.2. Variation in the inbreeding coefficients

The presence of cadmium in the soil did not affect the inbreeding coefficient. However, a high variation in the degree of inbreeding was found between the populations. The median inbreeding coefficients of the populations ranged from 0.053 to 0.264 whereby the highest median level of inbreeding was found in the population from the non-metalliferous sample site in Bad Rippoldsau. An effect of a population's geographical position or size, however, could be excluded.

The differences in the extent of inbreeding could, as already mentioned above, be connected to a restricted seed dispersal. Alternatively, and more importantly, it could also indicate a variation in the strength of self-incompatibility. The system of self-incompatibility (SI) represents a complex process which bases on a self- and non-self discrimination between the male pollen and the female pistil of a plant (reviewed by Takayama and Isogami 2005). This discrimination is mainly controlled by the multiallelic S-locus (reviewed by Takayama and Isogami 2005). In *A. halleri*, and other Brassicaceae, selfing is prevented by sporophytic self-incompatibility (SSI). In SSI the proper development of pollen tubes is inhibited if pollen and stigma (pistil) express the same S-haplotype (reviewed by Takayama and Isogami 2005).

Specific genes which are responsible for the determination of the S-haplotype have been identified for the male and female floral components (Schopfer et al. 1999, Takayama et al. 2000, Takasaki et al. 2000, Stein et al. 1991, Silva et al. 2001, Hiscock and Tabah 2003, reviewed by Takayama and Isogami 2005): stigma cells express the S-locus receptor kinase gene (SRK) which determines the S-haplotype and is promoted by S-locus glycoproteins (SLGs) (Takasaki et al. 2000, Stein et al. 1991, Silva et al. 2001, Hiscock and Tabah 2003 reviewed by Takayama and Isogami 2005). In the male pollen the S-haplotype is specified by the *S-locus cystein-rich* (SCR) gene (Schopfer et al. 1999, Takayama et al. 2000, Hiscock and Tabah 2003, reviewed by Takayama and Isogami 2005). A mutation in one of these genes or in the associated downstream components of the rejection cascade would lead to failure in the haplotype recognition system and thus to a weakening or loss of self-incompatibility (i.a. Mable et al. 2005, Cabrillac et al. 2001, Takayama et al. 2001, Franklin-Tong 2002, Kemp and Doughty 2003).

Furthermore, also pseudo-self-incompatibility (PSC) can emerge as a breakdown of the rejection mechanism in the presence of a fully functional self-incompatibility system (Levin 1996, Baldwin and Schoen 2017). PSC usually arises from mutations downstream in the SI-cascade (i.a. in Tantikanjana et al. 1993, Stone et al. 2003, Murase et al. 2004, Baldwin and Schoenen. 2017) or by interactions of the S-locus genes that weaken the ability to recognise self-pollen (Ockendon, 1974, Stevens and Kay, 1989, Baldwin and Schoenen 2017). Besides genetic variations, also environmental conditions can lead to alternations in the self-incompatibility system (e.g. Carafa and Carratu 1997, Okazaki and Hinata 1987, Taylor 1982 as summarized in Baldwin and Schoenen 2017).

Due to this high variation in the SI-system populations of species which are actually assumed to be strictly self-incompatible can also consist of self-compatible individuals and thus selfing and inbreeding cannot be fully avoided (Mable et al. 2005, Brennan et al. 2003). This has already been shown for populations of *A. lyrata*, a close relative of *A. halleri* which was also considered to be strictly outcrossing (Mable et al. 2005). Mable and her colleagues (2005) showed differences in the extent of inbreeding depending on the proportion of self-compatible individuals in a population. Populations with a high number of self-incompatible individuals possessed inbreeding coefficients close to zero whereas higher levels of inbreeding were found

in populations with a higher amount of self-compatible individuals (Mable et al. 2005). Due to this and the observed inbreeding coefficients a high proportion of self-compatible individuals can be assumed for most of the investigated populations of *A. halleri*. In populations of Blaibach (blai), Clausthal Zellerfeld (clau) and Lautenthal (laut) which possessed median inbreeding coefficients below 0.08 the populations appears to consist of mainly self-incompatible individuals.

Beside clonal propagation this flexibility in the strength of self-incompatibility ensures reproductive success even though the number of potential mating partners or the abundance of pollinators is low (Mable et al. 2005). Regarding this, *A. halleri* could rather overcome self-incompatibility to establish and maintain a population under harsh conditions than ensuring it via clonal propagation.

In conclusion, the observed population specific inbreeding levels could support a variation in the strength of self-incompatibility. Due to this, it could be confirmed that a dichotomous categorization (self-compatible or self-incompatible) is not sufficient to properly describe a specie's mating system (Baldwin et al. 2017). Furthermore, the classification of *A. halleri* as a self-incompatible and strictly outcrossing species should also be reconsidered.

4.4.3. Impact of cadmium on the kinship coefficient

The kinship coefficient was calculated for each individual pair of a population. In no population kinship coefficients of 0.5 could be found. Some individual pairs even showed negative kinship coefficients, which indicated a strong non-relatedness (Chen, 2017).

However, since the data did not fit a linear model the influence of cadmium on the degree of relatedness was only approximately determined and interpreted via the results of the inbreeding coefficient.

The kinship coefficients of the populations were compared. For this a Kruskal-Wallis test for overall population comparison and pairwise Wilcoxon tests were performed. It needs to be mentioned that by applying a Wilcoxon test the compared populations will erroneously be treated as dependent (Wilcoxon 1945) which results in a wrong number of significant differences (false negative error).

4.4.4. Cadmium and the frequency of clonal propagation

In *Arabidopsis halleri* clonal propagation is achieved by the formation of stolons (Al-Shehbaz and O'Kane 2002, Clauss and Koch 2006). In stoloniferous propagation the stem of an individual plant forms advantageous routs from which new plants can bud and develop (Campbell et al. 2009). Since this mode of reproduction does not include a recombination stage the originating individuals are genetically identical (Campbell et al. 2009). Therefore, a kinship coefficient of 0.5 should be observed between the clones. It was assumed, that metal contamination increases the frequency of clonal propagation and high kinship coefficients were expected on metalliferous soils. However, not even between individuals growing at a distance of 50cm (P) a kinship coefficient of 0.5 was measured. Neither on metalliferous nor non-metalliferous sites clones could be found. A predominance of clonal propagation on metal-polluted sites could thus not be confirmed and the presumed small number of founder individuals was rejected. Similar was already shown for the metalliferous pansy *Viola calaminaria* (Bizoux and Mahy 2007).

This could, as mentioned above, indicate a genetic diversity due to fine scaled heterogeneities in the soils concentration (Linhart and Grant 2006, Mattner et al. 2002 as cited in Bizuox and Mahy 2007). Furthermore, it can also imply that a low level of tolerance is not compulsorily associated with an absolute exclusion from metalliferous sites (i.a. Meyer 2015).

The lack of clones could also be connected to the sampling method. In greenhouse experiments performed by the collaborators in Tübingen, clonal growth could be observed (Tielbörger and Gruntman 2016). According to the proposal, plants were kept in small flowerpots and clonal individuals grew in very close distance to each other (approximately 5 cm) (Tielbörger and Gruntman 2016). When collecting leaves from the field, however, material was collected from individuals with a distance of 50cm, 3m respectively (in correspondence with the collaborators). To investigate clonal propagation and integration in natural populations a sampling from individuals with a shorter distance to each other should be considered.

4.4.5. Cadmium and the role of clonal integration

In clonal plants resources can be shared via horizontal interconnections like rhizomes and stolons between individuals of one genet (Tielbörger and Gruntman 2016, Hutchings and Wijesinghe 1997). It was assumed that this exchange could represent a major selective advantage especially on soils with a heterogeneous metal content since it reduces the effect of cadmium on a single individual. By this clonal integration can contribute to heavy metal tolerance and hyperaccumulation (Hutchings and Wijesinghe 1997, Alpert 1999, Tielbörger and Gruntman 2016). Since no clones could be found, neither on metalliferous nor non-metalliferous soils, clonal integration could not be investigated.

4.5. Effect of cadmium concentration

Metalliferous sample sites differed in their cadmium concentration. Here, not just the presence of Cd itself, but also the height of the concentration did not influence the inbreeding coefficient. If the tolerance level would be the same in all populations higher cadmium concentrations would represent a stressor in those with a lower tolerance level (Bert et al. 2000). This could lead to a reduced number of successfully developing seeds and reproducing plants. Consequently, a lower number of potential mating partners would be present.

A general high tolerance level, in contrast, would be disadvantageous on soils with a low cadmium content, since the mechanisms involved in metal tolerance and accumulation could constitute an expense factor (reviewed by Ernst 2006). Metallicolous populations of *Thlaspi caerulescens*, for instance, showed reduced fitness on non-metalliferous soils (Dechamps et al. 2008). The missing effect of the actual cadmium concentration confirms the presence of varying tolerance levels with a higher tolerance in populations growing on metalliferous sites (shown by Bert et al. 2000).

5. Summary and Conclusion

A high similarity between all populations could be observed. The number of variant loci, which are responsible for the differentiation of the 10 populations, could be narrowed to 36.

Cadmium did not affect the population structure and has no impact on the reproduction mode. Due to the lack of clones and closely related individuals a low number of founder individuals on metalliferous sites could be rejected. This implies that tolerance mechanisms in *Arabidopsis halleri* work very efficiently and that the tolerance levels might not be as restricting as assumed.

Population specific levels of inbreeding were shown which could not be associated with differences in the population size or geographic location. However, the differences in the inbreeding coefficient indicate a variation in the extent of self-incompatibility and implies a higher flexibility in a specie’s mating system.

6. References

Abdi H, Williams LJ (2010) Principal component analysis. WIREs Computational Statistics, 2, 433-459

Adler D, Murdoch D, Nenadig O, Urbanek S, Chen M, Gebhardt A, Bolker B, Csarfi G, Strzelecki A, Senger A, The R core Team (2017) rgl: 3D Visualization Using OpenGL. R package version 0.98.1, https://CRAN.R-project.org/package=rgl

Al-Shehbaz IA, O'Kane Jr. SL (2002) Taxonomy and phylogeny of Arabidopsis (Brassicaceae). In: Somerville CR, Meyerowitz EM (eds). The Arabidopsis Book. American Society of Plant Biologist, Rockville, e0001

Alpert P (1999) Clonal integration in *Fragaria chiloensis* differs between populations: ramets from grassland are selfish. Oecologia, 12.1, 69.76

Altschul SF, Gish W, Miller W, Myers EW, Lipman DJ (1990) Basic local alignment search tool. Journal of Molecular Biology, 215.3, 403-410

Andrews KR, Good JM, Miller MR, Luikart G, Hohenlohe PA (2016) Harnessing the power of RADseq for ecological and evolutionary genomics. Nature ReviewsGenetics, 17.2, 81–92.

Andrews S (2011) FastQC: a quality control tool for high throughput sequence data. Bioinformatics Babraham, Cambridge, UK Babraham Institute

Andrews S (2016) Analysis Models/Duplicated Sequences on https://www.bioinformatics.babraham.ac.uk/projects/fastqc/Help/3%20Analysis%20Modules/8%20Duplicate%20Sequences.html accessed: 20.9.2017

Andrews S, Krüger F, Wingett S, Ewels P (2016) https://sequencing.qcfail.com/articles/position-specific-failures-of-flowcells/. accessed: 12.8.2017

Arnold J (2001) Genetic Drift. In: Brenner S, Miller JH (Eds) Encyclopedia of Genetics, New York: Academic Press, 832-834

Baird AN, Etter PD, Atwood TS, Currey MC, Shiver AL, Lewis ZA, Selker EU, Cresko WA, Johnson EA (2008) Rapid SNP Discovery and Genetic Mapping Using Sequenced RAD Markers. PLoS ONE 3.10, e3376

Baker AJM (1981) Accumulators and excluders-strategies in the response of plants to heavy metals. Journal of plant nutrition, 3.1-4, 643-654

Baldwin SJ, Schoen DJ (2017) Genetic variation for pseudo-self-compatibility in self-incompatible populations of *Leavenworthia alabamica* (Brassicaceae). New Phytologist, 213, 430-439

Bansal V (2017) A computational method for estimating the PCR duplication rate in DNA and RNA-seq experiments. BMC Bioinformatics, 18.3, 43

Bates D, Maechler M, Bolker B, Walker S (2015) Fitting Linear Mixed-Effects Models Using lme4. Journal of Statistical Software, 67.1, 1-48

Becher M, Talke IN, Krall L, Krämer U (2004) Cross-species microarray transcript profiling reveals high constitutive expression of metal homeostasis genes in shoots of the zinc hyperaccumulator *Arabidopsis halleri*. the plant journal 37.2, 251-268

Beekman M, Ratnieks FLW (2000) Long-range foraging by the honey-bee, Apis mellifera L. . Functional Ecology, 14, 490-496

Bengtsson H (2017) matrixStats: Functions that Apply to Rows and Columns of Matrices (and to Vectors). R package version 0.52.2, https://CRAN.R-project.org/package=matrixStats

Bert V, Macnair MR, De Laguerie P, Saumitou-Laprade P, Petit D (2000) Zinc tolerance and accumulation in metallicolous and nonmetallicolous populations of Arabidopsis halleri (Brassicaceae) New Phytologist, 225-233

Bert V, Meerts P, Saumitou-Laprade P, Salis P, Gruber W, Verbruggen N (2003) Genetic basis of Cd tolerance and hyperaccumulation in *Arabidopsis halleri*. Plant and Soil, 249, 9-18

Bizoux J-P, Mahy G (2007) Within-Population Genetic Structure and Clonal Diversity of a threatened Endemic Metallophyte, *Viola calaminaria* (Violaceae). American Journal of Botany 94.5, 887-895

Brennan AC, Harris SA, Hiscock SJ (2005) Modes and rates of selfing and associated inbreedin depression in the self-incompatible plant *Senecio squalidus* (Asteraceae): a successful colonizing species in the British Isles. New Phytologist, 168, 475-486

Briskine RV, Paape T, Shimizu-Inatsugi R, Nishiyama T, Akama S, Sese J, Shimizu KK (2016a) Genome assembly and annotation of *Arabidopsis halleri*, a model for heavy metal hyperaccumulation and evolutionary ecology. Molecular Ecology Resources

Briskine RV, Paape T, Shimizu-Inatsugi R, Nishiyama T, Akama S, Sese J, Shimizu KK (2016b) Data from Genome assembly and annotation of *Arabidopsis halleri*, a model for heavy metal hyperaccumulation and evolutionary ecology. Dryad Digital Repository

Butkus V, Petrauskiené L, Manelienė Z, Klimasauskas S, Laucys V, Janulaitis A (1987) Cleavage of methylated CCCGGG sequences containing either N4-methylcytosine or 5-mehtylcytosine with MspI, HpaII, SmaI, XmaI and Cfr-9I restriction endonucleases. Nucleic Acids Research, 15, 7091-7102

Cabrillac D, Cock JM, Dumas C, Gaude T (2000) The S-locus receptor kinase is inhibited by thioredoxins and actibated by pollen coat proteins. Nature, 410, 220-223

Cain ML (2009) Die Evolution von Populationen. In: Campbell NA, Reece JB, Urry LA, Cain ML, Wasserman SA, Minorsky PV, Jackson RB: Biologie, Boston: Pearson, 628-652

Cao CC, Xiao S (2016) Combinatorial pooled sequencing: experiment design and decoding. Quantitative Biology, 4.1, 36-46

Carafa AM, Carratu G (1997) Stigma treatment with saline solutions: a new method to overcome self-incompatibility in *Brassica oleracea* L. Journal of Horticultural Science, 72, 531-535

Castric V, Vekemans X (2004) Plant self-incompatibility in natural populations: a critical assessment of recent theoretical and empirical advances. Molecular Ecology, 13, 2873-2889

Charlesworth D, Charlesworth B (1987) Inbreeding Depression and its Evolutionary Consequences. Annual Review of Ecology and Systematics, 18, 237-268

Charlesworth D, Willis JH (2009) The genetics of inbreeding depression. Nature Reviews Genetics, 10.11, 783-796

Chatterjee J, Chatterjee C (2000) Phytotoxicity of cobalt, chromium and copper in cauliflower. Environmental Pollution, 109.1, 69-74

Chen WM (2017) http://people.virginia.edu/~wc9c/KING/manual.html accessed: 4.6.2017

Chiang HC, Lo JC, Yeh KC (2006) Genes Associated with Heavy Metal Tolerance and Accumulation in Zn/Cd Hyperaccumulator *Arabidopsis halleri*: A Genomic Survey with cDNA Microarray. Environmental science & technology, 6792-6798

Chibuike GU, Obiora SC (2014) Heavy Metal Polluted Soils: Effect on Plants and Bioremediation Methods. Applied and Evironmental Soils Science, Article ID: 752708

Choudhuri S (2014) Fundamentals of Molecular Evolution. In: Bioinformatics for Beginners, Oxford: Academic Press, 27-53

Clauss MJ, Koch MA (2006) Poorly known relatives of *Arabidopsis thaliana*. Trends in plant science 11.9, 449-459

Courbot M, Willems G, Motte P, Arvidsson S, Roosens N, Saumitou-Laprade P, Verbruggen N (2007) A Major Quatitiative Trait Locus for cadmium Tolerance in *Arabidopsis halleri* Colocalizes with *HMA4*, a Gene Encoding a Heavy Metal ATPase. Plant Physiology, 144, 1052-1065

Cusack S (2016) https://github.com/edamame-course/FastQC/blob/master/final/2016-06-22_FastQC_tutorial.md accessed: 29.10.2017

DaCosta JM, Sorenson MD (2014) Amplification Biases and Consistent Recovery of Loci in a Double-Digest Rad-seq Protocol. PLoS One, 9.9, e106713

Danecek P, Auton A, Abecasis G, Alber CA, Banks E, DePristo MA, Handsaker RE, Lutner G, Marth GT, McVean STSG, Durbin R (2011) The variant call format and VCFtools. Bioinformatics, 27.15, 2156-2158

Davey JW, Blaxter ML (2011) RADSeq: next-generation population genetics. Briefing in functional genomics, 9.5, 416-423

Davey WJ, Hohenlohe PA, Etter PD, Boone JQ, Catchen JM, Blaxter ML (2011) Genome-wide genetic marker discovery and genotyping using next-generation sequencing. Nature Reviews Genetics, 12, 499-510

Dechamp C, Noret N, Mozek R, Escarré J, Lefebvre C, Gruber W, Meerts P (2008) Cost of adaptation to a metalliferous environment for Thlaspi caerulescens: a field of reciprocal transplantation approach. New Phytologist, 177.1, 167-177

Dinneny JR, Yanofsky M (2004) Drawing lines and borders: how the dehiscent fruit of Arabidopsis is patterned. Bioessays, 27.1, 42-49

Dong M (1996a) Clonal growth in plants in relation to resource heterogeneity: foraging behavior. Acta Botanica Sinica, 38, 828–835.

Dong M (1996b) Plant clonal growth in heterogeneous habitats: risk-spreading. Acta Phytoecologica Sinica 20, 543–548.

Dong M (2011) Ecology of Clonal Plants. Beijing: Science Press.

Ebbert MTW, Wadsworth ME, Staley LA, Hoyt KL, Pickett B, Miller J, Duce J, Kauwe JSK, Ridge PG (2016) Evaluating the necessity of PCR duplicate removal from next-generation sequencing data and a comparison of approaches. BMC Bioinformatics, 17.7, 239

Elshire RJ, Glaubitz JC, Sun, Q, Poland JA, Kawamoto K, Buckler ES, Mitchell SE (2011) A Robust, Simple Genotyping-by-Sequencing (GBS) Approach for High Diversity Species. PLoS One, 6.5,e19379

Ernst WHO (2006) Evolution of metal in higher plants. Forest Snow Landscape Research, 80.3, 251-274.

Escaravage N., Wagner J. (2004) Pollination effectiveness and pollen dispersal in a Rhododendron ferrugineum (Ericaceae) population. Plant Biology, 6, 606–615.

Etter PD, Bassham S, Hohenlohe PA, Johnson EA, Cresko WA (2011) SNP discovery and genotyping for evolutionary genetics using RAD sequencing. In: Etter PD, Bassham S, Hohenlohe PA, Johnson EA, Cresko WA, editors.Molecular methods for evolutionary genetics. Berlin: Springer, 157–78

Fox J, Weisberg S (2011) An {R} Companion to Applied Regression. Thousand Oaks, Ca: Sage, 2nd edition

Franklin-Tong N (2002) Receptor-ligand interaction demonstrated in *Brassica* self-incompatibility. Trends in Genetics, 18.3, 113-115

Furstentau TN, Carthwright RA (2017) The impact of self-incompatibility systems on the prevention of biparental inbreeding. PeerJ, 5, e4085

Garrison E, Marth G (2012) Haplotype-based variant detection from short-read sequencings. arXiv preprint arXic:1207.3907

Gaudeul M, Stenøien K, Ågren J (2007) Landscape Structure, Clonal Propagation and Genetic Diversity in Scandinavian Populations of Arabidopsis lyrata (Brassicaceae). American Journal of Botany 94.7, 1146-1155

Gautier M, Foucaud J, Gharbi K, Cézard T, Galan M, Loiseau A, Thomson M, Puldo P, Kerdelhué C, Estoup A (2013) Estimation of population allele frequencies from next-generation sequencing data: pool-versus individual-based genotyping. Molecular Ecology, 22.14, 3766-3779

Goodwin S, McPherson JD, McCombie WR (2016) Coming of age: ten years of next-generation sequencing technologies. Nature Reviews Genetics, 17.6, 333-351

Goudet J, Jombart T (2015) hierfstat: Estimation and Tests of Hierarchical F-Statistics. R package version 0.04-22. https://CRAN.R-project.org/package=hierfstat

Gross J, Ligges U (2015) nortest: Tests for Normality. R package version 1.0-4, https://CRAN.R-project.org/package=nortest

Hahsler M, Buchta C, Hornik (2017) seriation: Infrastructure for Ordering Objects Using Seriation. R package version 1.2-2, https://CRAN.R-project.org/package=seriation

Hahsler M, Hornik K, Buchta C (2008) Getting things in order: An introduction to the R package seriation. Journal of Statistical software, 25.3, 1-34

Halim M, Conte P, Piccolo A (2003) Potential availability of heavy metals to phytoextracation from contaminated soils induced by exogenous humic substances. Chemosphere, 52.1, 265-275

Hanikenne M, Talke IN, Haydon MJ, Lanz C, Nolte A, Motte P, Kroymann J, Weigel D, Krämer U (2008) Evolution of metal hyperaccumulation required *cis*-regulatory changes and triplication of HMA4. Nature, 453, 391-395

Harper J (1977) Population Biology of Plants. London: Academic Press

Takayama S, Isogami A (2005) Self-Incompatibility in Plants. Annual Review of Plant Biology, 56, 467-489

Holsinger KE, Weir BS (2009) Genetics in geographically structured populations: defining, estimating and interpreting Fst. Nature Reviews Genetics, 10.9, 639-650

https://blast.ncbi.nlm.nih.gov/Blast.cgi?PROGRAM=blastn&PAGE_TYPE=BlastSearch&LINK_LOC=blasthome accessed: 23.2.2018

http://broadinstitute.github.io/picard accessed: 12.9.2017

https://gist.github.com/danielecook/8e9afb2d2df7752efd8a accessed: 2.8.2017

https://map-maker.education.nationalgeographic.com accessed: 14.3.2017

http://rebase.neb.com/cgi-bin/msget?MspI accessed: 2.8.2017

http://samtools.github.io/hts-specs/VCFv4.3.pdf accessed: 12.9.2017

https://software.broadinstitute.org/gatk/documentation/article.php?id=4260 accessed: 11.9.2017

http://vcftools.sourceforge.net/man_latest.htm accessed: 24.4.2018

https://www.arabidopsis.org/tools/bulk/go/index.jsp accessed: 13.2.2017

https://www.neb.com/products/r0106-mspi accessed: 27.7.2017

Hutchings MJ, Wijesinghe DK (1997) Patchy Habitats, Division of Labour and Growth Dividends in Clonal Plants. Trends in Ecology & Evolution, 12, 390-394

Jentsch S, Günthert U, Trautner TA (1981) DNA methyltransferases affecting the sequence 5'CCGG. Nucleic Acids Research, 9.21, 2753-2759

Johnston JS, Pepper AE, Hall, AE, Chen ZJ, Hodnett G, Drabek J, Lopez R, Price HJ (2005) Evolution of Genome Size in Brassicaceae. Annals of Botany, 95, 229-235

Jombart T (2008) adegenet: a R package for the multivariate analysis of genetic markers. Bioinformatics, 24, 1403-1405

Jombart T, Ahmed I (2011) adegenet 1.3-1: new tools for the analysis of genome-wide SNP data. Bioinformatics, 27.21, 3070-3071

Keller LF, Waller DM (2002) Inbreeding effects in wild populations. Trends in Ecology & Evolution, 17.5, 230-241

Kemp BP, Doughty J (2002) Just how comples is the *Brassica* S-receptor complex? Journal of Experimental Botany, 54.380, 157-168

Keunen E, Remands T, Bohler S, Vangronsveld J, Cuypers A (2011) Metal-Induced Oxidative Stress and Plant Mitochondria. International Journal of Molecular Sciences, 12.10, 6894-6918

Kircher M, Hayn P, Kelso J (2011) Adressing challenges in the production and analysis of illumine sequencing data. BMC Genomics, 12.382

Knaus BJ, Grünwald NJ (2016) VcfR: an R package to manipulate and visualize VCF format data. BioRxiv

Knaus BJ, Grünwald NJ (2017) VCFR: a package to manipulate and visualize variant call format data in R. Molecular Ecology Ressources, 17.1, 44-53

Koch MA, Haubold B, Mitchell-Olds T (2000) Comparative evolutionary analysis of chalcone synthase and alcohol dehydrogenase loci in *Arabidopsis*, *Arabis* and related genera (Brassicaceae). Molecular Biology and evolution, 17, 1483-1498

Kolník M, Marhold K (2006) Distribution, chromosome numbers and nomenclature conspect of *Arabidopsis halleri* (Brassicaceae) in the Carpathians. Biologia, 61.1, 41-50

Korch C, Hagblom P (1986) In-vivo-modified gonococcal plasmid pJD1: A model system for analysis of restriction enzyme sensitivity to DNA modifications. European Journal of Biochemistry, 161, 519-524

Kozarewa I, Ning Z, Quail MA, Sanders MJ, Berriman M, Turner DJ (2009) Amplification-free Illumina sequencing-library preparation facilitates improved mapping and assembly of (G+C)-biased genomes.Nature Methods, 6.4,291-5.

Kubota H, Takenaka C (2003) *Arabis gemmifera* is a Hyperaccumulator of Cd and Zn. International Journal of Phytoremediation, 5.3, 197-201

Kusaba M, Tung CQ, Nasrallah ME, Nasrallah JB (2002) Monoallelic Expression and Dominance Interactions in Anthers of self-incompatible *Arabidopsis lyrata*. Plant Pysiology, 128.1, 17-20

Kuznetsova A, Brockhoff PB, Chhristensen RHB (2017) lmerTest Package: Tests in Linear Mixed Effects Models. Journal of Statistical Software, 82.13, 1-26

Lenth RV (2016) Least-Squares Means: The R Package lsmeans. Journal of Statistical Software, 69-1, 1-33

Levin DA (1995) The Evolutionary Significance of Pseudo-self-fertility. The American Nautralist, 148.2, 321-332

Li H (2013) Aligning sequence reads, clone sequences and assembly contigs with BWA-MEM. arXiv:1303.3997v1 [q-bio.GN]

Li H, Handsaker B, Wysoker A, Fennel T, Ruan J, Homer N, Marth G, Abecasis G, Durbin R, 1000Genome Project Data Processing Subgroup (2009) The Sequence alignment/map (SAM) format and SAMtools.Bioinformatics, 25, 2078-2079

Ligges U, Maechler M (2003) Scatterplot3d - an R Package for Visualizing Multivariate Data. Journal of Statistical Software, 11.8, 1-20

Llaurens V, Castri V, Austerlitz F, Vekemans X (2008) High paternal diversity in the self-incompatible herb *Arabidopsis halleri* despite clonal reproduction and spatially restricted pollen dispersal, Molecular Ecology, 17, 1577-1588

Lolkema PC, Vooijs R (1985) Copper tolerance in *Silene cucubalus*. Planta, 162.2, 174-179

Mable BK, Robertson AV, Dart S, Berardo CD, Witham L (2005) Breakdown of self-incompatibility in the perennial *Arabidopsis lyrata* (Brassicaceae) and its genetic consequences. Evolution, 59.7, 1427-1448

Manichaikul A, Mychaleckyi JC, Rich SS, Daly K, Sale , Chen WM (2010) Robust relationship inference in genome-wide association studies. Bioinformatics, 26.22, 2867-2873

McCormack JE, Hird SM, Zellmer AJ, Carstens BC, Brumfield RT (2012) Applications of next-generation sequencing to phylogeography and phylogenetics. Molecular Phylogenetics and Evolution, 66.2, 526-538

McKenna A, Hanna M, Banks E, Sivachenko A, Cibulskis K, Kernytsky A, Garimella K, Altshuler D, Gabriel S, Daly M, DePristo MA (2010) The Genome Analysis Toolkit: A MapReduce framework for analysing next-generation DNA sequencing data. Genome research, 20.9, 1297-1303

Meyer CL, Juraniec M, Huguet S, Chaves-Rodriguez E, Salis P, Isaure MP, Goormaghtigh E, Verbruggen N (2015) Intraspecific variability of cadmium tolerance and accumulation, and cadmium-induced cell wall modifications in the metal hyperaccumulator *Arabidopsis halleri*. Journal of Experimental Botany, 66.1, 3215-3227

Miller SA, Dykes DD, Polesky HFRN (1988) A simple salting out procedure for extracting DNA from human nucleated cells. Nucleic acids research, 16.3, 1215

Minorsky PV (2009) Blütenpflanzen: Strucktur, Wachstum, Entwicklung. In: Campbell NA, Reece JB, Urry LA, Cain ML, Wasserman SA, Minorsky PV, Jackson RB: Biologie, Boston: Pearson, 1002-1003

Mitchell-Olds T (2001) *Arabidopsis thaliana* and its wild relatives: a model system for ecology and evolution. Trends in Ecology and Evolution 16.12, 693-700

Mullen MP, Creevey CJ, Berry DP, McCabe MS, Magee DA, Howard DJ, Killeen AP, Park SD McGettigan PA, Lucy MC, MacHugh DE, Waters SM (2012) Polymorphism discovery and allele frequency estimation using high-throughput DNA sequencing of target-enriched pooled DNA samples. BMC genomics, 13.1, 16

Murase K, Shiba H, Iwano M, Che FS, Watanabe M, Isogai A, Takayama S (2004) A membrane-anchored protein kinase involved in Brassica self-incompaitbility signaling. Science, 303, 1516-1519

Naz, A, Khan S, Muhammad S, Khalid S, Alam S, Siddique S, Toqeer A, Scholz M (2015) Toxicity and Bioaccumulation of Heavy Metals in Spinach (*Spinacia oleracea*) Grown in a Controlled Environment. International Journal of Environmental Research and Public Health, 12, 7400-7416

Nelson M, Christ C, Schildkraut I (1984) Alternation of apparent restriction endonuclease recognition specificities by DNA methylases. Nucleic Acids Research, 12.13, 5165-5173

Novikova PY, Hohmann N, Nizhynska V, Tsuchimatsu T, Ali J, Muir G, Guggisberg A, Paape T, Schmid K, Fedorenko OM, Holm S, Säll T, Schlötterer C, Marhold K, Widmer A, Sese J, Shimizu KK, Weigel D, Krämer U, Koch MA, Nordborg M (2016) Sequencing of the genus *Arabidopsis* identifies a complex history of nonbifurcating speciation and abundant trans-specific polymorphism. Nature Genetics, 48.9, 1077-1082

Ockendon DJ (1974) Distribution of self-incompatibility alleles and breeding structure of open-pollinated cltivars of Brussels sprouts. Heredity, 33, 159-171

O'Kane Jr. SL, Al-Shehbaz IA (1997) A Synopsis of Arabidopsis (Brassicaceae). Novon, 7.3, 323-327

Okazaki K, Hinata K (1987) Repressing the expression of self-incompatibility in crucifers by short-term high temperature treatment. Theoretical and Applied Genetics, 73, 496.500

Oancea S, Foca N, Airinei A (2005) Effects of heavy metals on plant growth and Photosynthetic activity. Analele Stiintifice ale Universitatii "AL. I. CUZA" IASI, Tomul I, s. Biofizica, Fizica medicala si Fizica mediului, 107-110

Pauwels M, Frérot H, Bonnin I, Saumitou-Laprade P (2006) A broad-scale analysis of population differentiation for Zn tolerance in an emerging model species for tolerance study: *Arabidopsis halleri* (Brassicaceae). Journal of Evolutionary Biology, 19.6, 1838-50

Pelley JW (2012) Recombinant DNA and Biotechnology. In Elsevier's Integrated Review Biochemistry. Philadelphia: Saunders, 161-169

Peterson BK, Weber JN, Kay Eh, Fisher HS, Hoekstra HE (2012) Double Digest RADseq: An Inexpensive Method for De Novo SNP Discovery and Genotyping in Model and Non-Model Species. PLoS ONE 7.5, e37135

Poland JA, Brown PJ, Sorrells ME, Jannik JL (2012) Development of High-Density Genetic Maps for Parley and Wheat using a Novel Two-Enzyme Genotyping-by-Sequencing Approach. PloS one, 7.2, e32253

Quinlan AR, Hall IM (2010) BEDTools: a flexible suite of utilities for comparing genomic features. Bioinformatics, 26.6, 841-842

R Core Team (2017) R: A language and environment for statistical computing. R Foundation for Statistical Computing, Vienna, Austria. URL https://www.R-project.org/.

Roberts RJ, Vincze T, Posfai J, Macelis D (2015) REBASE- a database for DNA restriction and modification: enzymes, genes, and genomes. Nucleic Acids Research, 43, D298-D299

Roeder A HK, Yanofsky MF (2006) Fruit Development in Arabidopsis. The Arabidopsis Book, e0075

Roosens NHCJ, Willems G, Saumitou-Laprade P (2008) UsingArabidopsis to explore zinc tolerance and hyperaccumulation. Trends in Plant Science 13, 208–215.

Roux C, Castric V, Pauwels M, Wright SI, Saumitou-Laprade P, Vekemans X (2011) Does Speciation between *Arabidopsis halleri* and *Arabidopsis lyrata* Coincide with Major Changes in a Molecular Target of Adaptation? PLoS ONE, 6.11, e26872

Russel PJ (2010) Population Genetics. In: iGenetics: a molecular approach. Upper Saddle River, Harlow: Pearson Education, 639

Salemaa M, Canha-Majamaa I, Gardner PJ (1999) Compensatory growth of two clonal dwarf shrubs, *Arctostaphylos uva-ursi* and *Vaccinium uliginosum* in heavy metal polluted environment. Plant Ecology, 141, 79-91

Salemma M, Sievanen R (2002) The effect of apical dominance on the branching architecture of *Arctostaphylos uva-ursi* in four contrasting environments. Flora, 197, 429-442

Sandalino LM, Dalurzo HC, Gómez M, Romero-Puertas MC, del Río LA (2001) cadmium-induced changes in the growth and oxidative metabolism of pea plants. Journal of Experimental Botany, 52.364, 2115-2126

Schield DR, Walsh MR, Card DC, Andrew AL, Adams RH, Castoe TA (2016) EpiRADseq: scalable analysis of genomewide patterns of methylation using next-generation sequencing. Methods in Ecology and Evolution,7, 60-69

Schlötterer C, Tobler R, Kofler R, Nolte V (2014) Sequencing pools of individuals- mining genome-wide polymorphism data without big funding. Nature Reviews Genetics, 15.11, 749-763

Schopfer CR, Nasrallah ME, Nasrallah JB (1999) The Male Determinant of Self-Incompatibility in *Brassica*. Science, 286, 1697-1700

Schulke B., Waser N.M. (2001) Long-distance pollinator flights and pollen dispersal between populations of Delphinium nuttallianum. Oecologia, 127, 239–245.

Shimizu KK, Tsuchimatsu T (2015) Evolution of selfing: recurrent patterns in molecular adaptation. Annual Review of Ecology, Evolution and Systematics, 46: 593-622

Shirasawa K, Hirakawa H, Isobe S (2016) Analytical workflow of double-digest restriction site-assciated DNA sequencing based on empirical and in *silico* optimization in tomato. DNA Research, 23.2, 145-153

Sievert C, Parmer C, Hocking T, Chamberlain S, Ram K, Corvellec M, Despouy P (2017) plotly: Create Interactive Web Graphics via 'plotly.js'. R package version 4.7.1, https://CRAN.R-project.org/package=plotly

Silva NF, Stone SL, Christie LN, Sulaman W, Nazarian KA, Burnett LA, Arnoldo MA, Rothstein SJ, Goring DR (2001) Expression of the S receptor kinase in self-incompatible Brassica napus cv. Westar leads to the allele-specific rejection of self-incompatible Brassica napus pollen. Molecular Genetics and Genomics, 265, 552-559

Singh S, Parihar P, Singh R, Singh VP, Prasad SM (2016) Heavy Metal Tlerance in Plants: Role of Transcriptomics, Proteomics, Metabolomics, and Ionomics. Frontiers in Plant Science, 6.1143

Šrámková-Fuxová G, Záveská E, Kolár F, Lucanová M, Španiel S, Marhold K (2017) Range-wide genetic structure of *Arabidopsis halleri* (Brassicaceae): glacial persistence in multiple refugia and origin of Northern Hemisphere disjunction. Botanical Journal of Linnean Society, 185, 321-342

Stein RJ, Höreth S, de Melo RF, Syllwasschy L, Lee G, Garbin ML, Clemens S, Krämer U (2017) Relationships between soil and leaf mineral composition are element-specific, environment-dependent and geographically structured in the emerging model *Arabidopsis halleri*. New Phytologist, 213.3, 1274-1286

Stein JC, Howlett B, Boxes DC, Nasrallah ME, Nasrallah JB (1991) Molecular cloning of a putative receptor protein kinase gene encoded at the self-incompatibility locus of *Brassica oleracea*. Proceedings of the National Academy of Sciences of the United States of America, 88, 8816-8820

Stevens JP, Kay N (1989) The number, dominance relationships and frequncies of self-incompatibility alleles in a natural population of *Sinapis arvensis* L. in South Wales. Heredity, 62, 199-205

Stolpe C, Krämer U, Müller C (2017) Heavy metal (hyper)accumulation in leaves of *Arabidopsis halleri* is accompanied by a reduced performance of herbivores and shifts in leaf glucosinolate and element concentrations. Environmental and Experimental Botany, 133, 78-86

Stone SL, Anderson EM, Mullen RT, Goring DR (2003) ARC1 is an E3 ubiquitin ligase and promotes the ubiquitination of proteins during the rejection of self-incompatible Brassica pollen. Plant Cell, 15.4, 885-898

Sun S, Gao X, Gai Y (2001) Variations ins sexual and asexual reproduction of *Scirpus mariqueter* alon an elevational gradient. Ecological research, 16, 263-274

Takasaki T, Hatakeyama K, Suzuki G, Watanabe M, Isogai A, Hinata K (2000) The S receptor kinase determines self-incompatibility in *Brassica* stigma Nature, 403, 913-916

Takayama S, Isogami A (2005) Self-Incompatibility in Plants. Annual Review of Plant Biology, 56, 467-489

Takayama S, Shiba H, Iwano M, Shimisato H, Che FS, Kai N, Watanabe M, Suzuki G, Hinata K, Isogai A (2000) The pollen determinant of self-incompatibility in Brassica campestris. Proceedings of the national Academy of Sciences of the United States of America, 97.4, 1920-1925

Takayama S, Shimosato H, Shiba H, Funato M, Che FS, Watanabe M, Iwano M, Isogai A (2001) Direct ligand-receptor complex interaction controls *Brassica* self-incompatibility. Nature, 413, 534-538

Tantikanjana T, Nasrallah ME; Stein JC, Chen CH, Nasrallah JB (1993) An alternative transcript of the S locus glycoprotein gene in a class II pollen-recessive self-incompatibility haplotype of *Brassica oleracea* encodes a membrane-anchored protein. Plant Cell, 5, 657-666

Tardy-Planenchaud S, Fujimoto J, Lin SS, Sowers LC (1997) Solid phase synthesis and restriction endonuclease cleavage of oligodeoxynucleotides containing 5-(hydroxymethyl)-cytosine. Nucleic Acids Research, 25.3, 553-558

Taylor JP (1982) Carbon dioxide treatment as an effecive aid to the production of selfed seeds in kale and Brussels sprouts. Euphytica, 31, 957-964

The Arabidopsis Information Resource (TAIR), www.arabidopsis.org/portals/education/aboutarabidopsis.jsp, on www.arabidopsis.org accessed: 21.1. 2017

Tielbörger K, Gruntman, M (2016) Project description: The role of biotic interactions in determining phenotypic and genotypic variation in metal hyperaccumulation and hypertolerance in to model Brassicaceae species. University Tübingen

Townsend P.A., Levey D.J. (2005) An experimental test of whether habitat corridors affect pollen transfer. Ecology, 86,466–475.

Unseld M, Marienfeld JR, Brandt P, Brennicke A (1997) The mitochondrial genome of Arabidopsis thaliana contains 57 genes in 366,924 nucleotides. Nature Genetics, 15.1, 57-61 (NCBI accession number: Y08501.2)

Urry LA (2009) Meiose und geschlechtliche Fortpflanzung. In: Campbell NA, Reece JB, Urry LA, Cain ML, Wasserman SA, Minorsky PV, Jackson RB: Biologie, Boston: Pearson, 332-335

van Dijk EL, Auger H, Jaszczyszyn Y, Thermes C (2014) Ten years of next-generation sequencing technology. Trends in Genetics, 30.9, 418-426

Van Rossum F, Bonnin I, Fénart S, Pauwels M, Petit D, Saumitou-Laprade P (2004) Spatial genetic structure within a metallicolous population of *Arabidopsis halleri*, a clonal, self-incompatible and heavy-metal-tolerant species. Molecular Ecology 13.10, 2959-2967

Vert G, Grotz N, Dedaldechamp F, Gaymard F, Guerinot ML, Briat JF, Curie C (2000) IRT1, an *Arabidopsis* transporter essential for iron uptake from the soil and for plant growth. Plant Cell, 14, 1223-1233

Weber M, Harada E, Vess C, v. Roepenack-Lahaye E, Clemens S (2004) Comparative microarray analysis of *Arabidopsis thaliana* and *Arabidopsis halleri* roots identifies nicotinamine synthase, a ZIP transporter and other genes as potential metal hyperaccumulation factors. The Plant Journal, 37.2, 269-281

Wei T, Simko V (2016) corrplot: Visualization of a Correlation Matrix. R package version 0.77, https://CRAN.R-project.org/package=corrplot

Weir BS, Cockerham CC (1984) Estimating F-Statistics for the Analysis of Population Structure. Evolution, 38.6, 1358-1370

Wilcoxon F (1945) Individual comparisons by ranking methods. Biometrics Bulletin, 1, 80–83.

Willems G, Dräger DB, Courbot M, Godé C , Verbruggen N, Saumitou-Laprade P (2007) The genetic basis of zinc tolerance in the metallophyte Arabidopsis halleri ssp. halleri (Brassicaceae): an analysis of quantitative trait loci. Geneitcs, 176, 659-674

Wratten S.D., Bowie M.H., Hickman J.M., Evans A.M., SedcoleJ.R., Tylianakis J.M. (2003) Field boundaries as barriers to movement of hover flies (Diptera: Syrphidae) in cultivatedland. Oecologia, 134, 605–611.

Wright K (2017) corrgram: Plot a Correlogram. R package version 1.12, https://CRAN.R-project.org/package=corrgram

Wright S (1949) The genetical structure of populations. Annals of Human Genetics, 15.1, 323-354

Wuana RA, Okieimen FE (2011) Heavy Metals in Contaminated Soils: A Review of Sources Chemistry, Risks and Best Available Strategies for Remediation. International Scholarly Research Network, Ecology, 402647

Yang XE, Jin XF, Feng Y, Islam E (2005) Molecular Mechanisms and Genetic Basis of Heavy Metal Tolerance and Hyperaccumulation in Plants. Journal of Integrative Plant Biology, 47.9, 1025-1035

Zhang MK, Liu ZY, Wang H (2010) Use of Single Extraction Methods to Predict Bioavailability of Heavy Metals in Polluted Soils to Rice. Communications in Soils Science and Plant Analysis, 41.7, 820-831

7. List of figures and tables

8. Supplementary

S1. Cadmium concentration of individual soil samples per site. To consider heterogeneities in the cadmium concentration of a sample site 9-16 soil samples were taken per location. The individual concentration (cadmium) measurements are given for each population in µg/g. Populations from metalliferous sample sites are shaded in grey.

Sample	cadmium [µg/g]	Sample	cadmium [µg/g]	Sample	Cadmium [µg/g]
badr_1	0.585	fort_7	1.653	wulm_3	18.310
badr_2	0.812	fort_8	4.101	wulm_4	38.690
badr_3	0.841	fort_9	2.415	wulm_5	69.470
badr_4	0.801	fort_10	1.698	wulm_6	17.300
badr_5	0.595	fort_11	2.709	wulm_7	34.320
badr_6	1.025	fort_12	2.573	wulm_8	6.380
badr_7	0.700	laut_1	21.460	wulm_9	8.573
badr_8	0.610	laut_2	57.120	wulm_10	14.340
badr_9	0.318	laut_3	58.550	wulm_11	12.610
badr_10	0.311	laut_4	33.090	wulm_12	13.390
badr_11	0.332	laut_5	47.400	czra_1	0.617
badr_12	0.537	laut_6	10.710	czra_2	0.770
badr_13	0.691	laut_7	49.180	czra_3	1.080.
badr_14	0.731	laut_8	23.820	czra_4	1.150.
badr_15	0.724	laut_9	29.350	czra_5	0.986
badr_16	0.636	laut_10	27.660	czra_6	1.309
blai_1	1.014	laut_11	35.710	czra_7	1.183
blai_2	1.144	laut_12	25.430	czra_8	1.034
blai_4	1.235	litt_1	23.360	czra_9	0.701
blai_6	1.909	litt_2	26.670	czrb_1	0.617
blai_7	1.586	litt_3	25.000	czrb_2	0.770
blai_8	1.667	litt_4	15.690	czrb_3	1.080
blai_9	1.437	litt_5	22.600.	czrb_4	1.150
blai_10	1.817	litt_6	30.570	czrb_5	0.986
blai_11	1.525	litt_7	18.600.	czrb_6	1.309
blai_12	0.961	litt_8	33.120	czrb_7	1.183
clau_1	9.217	litt_9	8.913	czrb_8	1.034
clau_2	43.570	litt_10	5.315	czrb_9	0.701
clau_3	61.940	litt_11	8.308	czrc_1	23.980
clau_4	29.720	litt_12	7.251	czrc_2	31.820
clau_5	33.080	vieb_1	50.010	czrc_3	25.050
clau_6	78.560	vieb_2	45.570	czrc_4	21.970
clau_7	7.941	vieb_3	55.640	czrc_5	16.190
clau_8	15.630	vieb_4	19.980	czrc_6	18.790
clau_9	25.870	vieb_5	11.370	czrc_7	30.690
clau_10	69.840	vieb_6	10.040	czrc_8	14.310
clau_11	40.760	vieb_7	8.898	czrc_9	19.450
clau_12	26.850	vieb_8	11.720.	czrc_10	19.190
fort_1	1.474	vieb_9	16.540	czrc_11	19.180
fort_2	2.104	vieb_10	7.521	czrc_12	14.850
fort_3	1.449	vieb_11	13.000		
fort_4	2.477	vieb_12	9.782		
fort_5	1.919	wulm_1	13.250		
fort_6	2.037	wulm_2	21.030		

S2. Used tools and reagents. A list of all used tool and reagents are given. The tools and reagents are listed according to the order they are mentioned in the method-section.

MiniG™ 1600	Spex® SamplePrep USA
PHMT Grant-bio Thermomixer	Grant Instruments Ltd. United Kingdom
RNAse A (100mg/ml)	QIAGEN Hilden, Germany
eppendorf Centrifuge 5424R	Eppendorf AG Germany
IKA® Vortex Genius 3	IKA®-Werke GmbH & Co. KG Germany
eppendorf Centrifuge 5424	Eppendorf AG Hamburg, Germany
Quant-iT™ dsDNA HS Assay Kit	invitrogen™ by Thermo Fisher Scientific USA
Qubit™ fluormeter	invitrogen™ by Thermo Fisher Scientific USA
PstI-HF (CTGCAG) MspI (CCGG) restriction enzyme	New England Biolabs ® Inc. Ipswich, USA
ProFlex PCR system	Thermo Fisher scientific USA
Elution Buffer (10mM Tris-Cl, pH 8.5)	QIAGEN Hilden, Germany
CutSmart® Buffer	New England Biolabs ® Inc. Ipswich, USA
T4 DNA ligase	New England Biolabs ® Inc. Ipswich, USA
QIAquick® PCR Purification Kit	QIAGEN Hilden, Germany
primer forward (5' AATGATACGGCGACCACGAGATCTACACTCTTTCC CTACACGACGCTC TTCCGATCT - 3')	invitrogen™ by Thermo Fisher Scientific USA
primer reverse (5' - CAAGCAGAAGACGGCATACGAGATCGGTCTCGGC ATTCCTGCTGAA - 3')	invitrogen™ by Thermo Fisher Scientific USA
Phusion polymerase (2x)	New England Biolabs ® Inc. Ipswich, USA
Molecular Imager® Gel Doc™ XR+ system	Bio-Rad Laboratories Inc. California, USA
ChromaLight blue Converstion Screen Emiss. 460nm, 21x26 cm light area	distributed by Biozym Scientific GmbH Germany
MinElute® Gel Extraction Kit	QIAGEN Hilden, Germany

S3. Barcode and adapter sequences. The 220 used barcodes and their respective length are given. Moreover, the full sequence of the forward adapters are listed (adapter_top, adapter_bot). All adapters were ordered from Sigma® Life Science (USA).

	Barcode	adapter_top	adapter_bot	length
1	TATTCGCAT	cacgacgctcttccgatctTATTCGCATtgca	ATGCGAATAagatcggaagagcgtcgtg	9
2	ATAGAT	cacgacgctcttccgatctATAGATtgca	ATCTATagatcggaagagcgtcgtg	6
3	CCGAACA	cacgacgctcttccgatctCCGAACAtgca	TGTTCGGagatcggaagagcgtcgtg	7
4	GGAAGACAT	cacgacgctcttccgatctGGAAGACATtgca	ATGTCTTCCagatcggaagagcgtcgtg	9
5	GGCTTA	cacgacgctcttccgatctGGCTTAtgca	TAAGCCagatcggaagagcgtcgtg	6
6	AACGCACATT	cacgacgctcttccgatctAACGCACATTtgca	AATGTGCGTTagatcggaagagcgtcgtg	10
7	GAGCGACAT	cacgacgctcttccgatctGAGCGACATtgca	ATGTCGCTCagatcggaagagcgtcgtg	9
8	CCTTGCCATT	cacgacgctcttccgatctCCTTGCCATTtgca	AATGGCAAGGagatcggaagagcgtcgtg	10
9	GGTATA	cacgacgctcttccgatctGGTATAtgca	TATACCagatcggaagagcgtcgtg	6
10	TCTTGG	cacgacgctcttccgatctTCTTGGtgca	CCAAGAagatcggaagagcgtcgtg	6
11	GGTGT	cacgacgctcttccgatctGGTGTtgca	ACACCagatcggaagagcgtcgtg	5
12	GGATA	cacgacgctcttccgatctGGATAtgca	TATCCagatcggaagagcgtcgtg	5
13	CTAAGCA	cacgacgctcttccgatctCTAAGCAtgca	TGCTTAGagatcggaagagcgtcgtg	7
14	ATTAT	cacgacgctcttccgatctATTATtgca	ATAATagatcggaagagcgtcgtg	5
15	GCGCTCA	cacgacgctcttccgatctGCGCTCAtgca	TGAGCGCagatcggaagagcgtcgtg	7
16	ACTGCGAT	cacgacgctcttccgatctACTGCGATtgca	ATCGCAGTagatcggaagagcgtcgtg	8
17	TTCGTT	cacgacgctcttccgatctTTCGTTtgca	AACGAAagatcggaagagcgtcgtg	6
18	ATATAA	cacgacgctcttccgatctATATAAtgca	TTATATagatcggaagagcgtcgtg	6
19	TGGCAACAGA	cacgacgctcttccgatctTGGCAACAGAtgca	TCTGTTGCCAagatcggaagagcgtcgtg	10
20	CTCGTCG	cacgacgctcttccgatctCTCGTCGtgca	CGACGAGagatcggaagagcgtcgtg	7
21	GCCTACCT	cacgacgctcttccgatctGCCTACCTtgca	AGGTAGGCagatcggaagagcgtcgtg	8
22	CACCA	cacgacgctcttccgatctCACCAtgca	TGGTGagatcggaagagcgtcgtg	5
23	AATTAG	cacgacgctcttccgatctAATTAGtgca	CTAATTagatcggaagagcgtcgtg	6
24	GGAACGA	cacgacgctcttccgatctGGAACGAtgca	TCGTTCCagatcggaagagcgtcgtg	7
25	ACAACT	cacgacgctcttccgatctACAACTtgca	AGTTGTagatcggaagagcgtcgtg	6
26	ACTGCT	cacgacgctcttccgatctACTGCTtgca	AGCAGTagatcggaagagcgtcgtg	6
27	CGTGGACAGT	cacgacgctcttccgatctCGTGGACAGTtgca	ACTGTCCACGagatcggaagagcgtcgtg	10

28	TGGCACAGA	cacgacgctcttccgatctTGGCACAGAtgca	TCTGTGCCAagatcggaagagcgtcgtg	9
29	TGCTT	cacgacgctcttccgatctTGCTTtgca	AAGCAagatcggaagagcgtcgtg	5
30	GCAAGCCAT	cacgacgctcttccgatctGCAAGCCATtgca	ATGGCTTGCagatcggaagagcgtcgtg	9
31	CGCACCAATT	cacgacgctcttccgatctCGCACCAATTtgca	AATTGGTGCGagatcggaagagcgtcgtg	10
32	CTCGCGG	cacgacgctcttccgatctCTCGCGGtgca	CCGCGAGagatcggaagagcgtcgtg	7
33	AACTGG	cacgacgctcttccgatctAACTGGtgca	CCAGTTagatcggaagagcgtcgtg	6
34	ATGAGCAA	cacgacgctcttccgatctATGAGCAAtgca	TTGCTCATagatcggaagagcgtcgtg	8
35	CTTGA	cacgacgctcttccgatctCTTGAtgca	TCAAGagatcggaagagcgtcgtg	5
36	GCGTCCT	cacgacgctcttccgatctGCGTCCTtgca	AGGACGCagatcggaagagcgtcgtg	7
37	ACCAGGA	cacgacgctcttccgatctACCAGGAtgca	TCCTGGTagatcggaagagcgtcgtg	7
38	CCACTCA	cacgacgctcttccgatctCCACTCAtgca	TGAGTGGagatcggaagagcgtcgtg	7
39	TCACGGAAG	cacgacgctcttccgatctTCACGGAAGtgca	CTTCCGTGAagatcggaagagcgtcgtg	9
40	TATCA	cacgacgctcttccgatctTATCAtgca	TGATAagatcggaagagcgtcgtg	5
41	TAGCCAA	cacgacgctcttccgatctTAGCCAAtgca	TTGGCTAagatcggaagagcgtcgtg	7
42	ATATCGCCA	cacgacgctcttccgatctATATCGCCAtgca	TGGCGATATagatcggaagagcgtcgtg	9
43	CTCTA	cacgacgctcttccgatctCTCTAtgca	TAGAGagatcggaagagcgtcgtg	5
44	GGTGCACATT	cacgacgctcttccgatctGGTGCACATTtgca	AATGTGCACCagatcggaagagcgtcgtg	10
45	CTCTCGCAT	cacgacgctcttccgatctCTCTCGCATtgca	ATGCGAGAGagatcggaagagcgtcgtg	9
46	CAGAGGT	cacgacgctcttccgatctCAGAGGTtgca	ACCTCTGagatcggaagagcgtcgtg	7
47	GCGTACAAT	cacgacgctcttccgatctGCGTACAATtgca	ATTGTACGCagatcggaagagcgtcgtg	9
48	ACGCGCG	cacgacgctcttccgatctACGCGCGtgca	CGCGCGTagatcggaagagcgtcgtg	7
49	GTCGCCT	cacgacgctcttccgatctGTCGCCTtgca	AGGCGACagatcggaagagcgtcgtg	7
50	AATAACCAA	cacgacgctcttccgatctAATAACCAAtgca	TTGGTTATTagatcggaagagcgtcgtg	9
51	AATGAACGA	cacgacgctcttccgatctAATGAACGAtgca	TCGTTCATTagatcggaagagcgtcgtg	9
52	CGTCGCCACT	cacgacgctcttccgatctCGTCGCCACTtgca	AGTGGCGACGagatcggaagagcgtcgtg	10
53	ATGGCAA	cacgacgctcttccgatctATGGCAAtgca	TTGCCATagatcggaagagcgtcgtg	7
54	GAAGCA	cacgacgctcttccgatctGAAGCAtgca	TGCTTCagatcggaagagcgtcgtg	6
55	AACGTGCCT	cacgacgctcttccgatctAACGTGCCTtgca	AGGCACGTTagatcggaagagcgtcgtg	9
56	CCTCG	cacgacgctcttccgatctCCTCGtgca	CGAGGagatcggaagagcgtcgtg	5
57	CTCAT	cacgacgctcttccgatctCTCATtgca	ATGAGagatcggaagagcgtcgtg	5

58	ACGGTACT	cacgacgctcttccgatctACGGTACTtgca	AGTACCGTagatcggaagagcgtcgtg	8
59	GCGCCG	cacgacgctcttccgatctGCGCCGtgca	CGGCGCagatcggaagagcgtcgtg	6
60	CAAGT	cacgacgctcttccgatctCAAGTtgca	ACTTGagatcggaagagcgtcgtg	5
61	TCCGAG	cacgacgctcttccgatctTCCGAGtgca	CTCGGAagatcggaagagcgtcgtg	6
62	TAGATGA	cacgacgctcttccgatctTAGATGAtgca	TCATCTAagatcggaagagcgtcgtg	7
63	TGGCCAG	cacgacgctcttccgatctTGGCCAGtgca	CTGGCCAagatcggaagagcgtcgtg	7
64	GCACGAT	cacgacgctcttccgatctGCACGATtgca	ATCGTGCagatcggaagagcgtcgtg	7
65	TTGCTG	cacgacgctcttccgatctTTGCTGtgca	CAGCAAagatcggaagagcgtcgtg	6
66	CGCAACCAGT	cacgacgctcttccgatctCGCAACCAGTtgca	ACTGGTTGCGagatcggaagagcgtcgtg	10
67	TCACTG	cacgacgctcttccgatctTCACTGtgca	CAGTGAagatcggaagagcgtcgtg	6
68	ACAGT	cacgacgctcttccgatctACAGTtgca	ACTGTagatcggaagagcgtcgtg	5
69	GGAGTCAAG	cacgacgctcttccgatctGGAGTCAAGtgca	CTTGACTCCagatcggaagagcgtcgtg	9
70	TGAAT	cacgacgctcttccgatctTGAATtgca	ATTCAagatcggaagagcgtcgtg	5
71	CATAT	cacgacgctcttccgatctCATATtgca	ATATGagatcggaagagcgtcgtg	5
72	GTGACACAT	cacgacgctcttccgatctGTGACACATtgca	ATGTGTCACagatcggaagagcgtcgtg	9
73	TATGT	cacgacgctcttccgatctTATGTtgca	ACATAagatcggaagagcgtcgtg	5
74	CAGTGCCATT	cacgacgctcttccgatctCAGTGCCATTtgca	AATGGCACTGagatcggaagagcgtcgtg	10
75	ACAACCAACT	cacgacgctcttccgatctACAACCAACTtgca	AGTTGGTTGTagatcggaagagcgtcgtg	10
76	TGCAGA	cacgacgctcttccgatctTGCAGAtgca	TCTGCAagatcggaagagcgtcgtg	6
77	CATCTGCCG	cacgacgctcttccgatctCATCTGCCGtgca	CGGCAGATGagatcggaagagcgtcgtg	9
78	GGACAG	cacgacgctcttccgatctGGACAGtgca	CTGTCCagatcggaagagcgtcgtg	6
79	ATCTGT	cacgacgctcttccgatctATCTGTtgca	ACAGATagatcggaagagcgtcgtg	6
80	AAGACGCT	cacgacgctcttccgatctAAGACGCTtgca	AGCGTCTTagatcggaagagcgtcgtg	8
81	GAATGCAATA	cacgacgctcttccgatctGAATGCAATAtgca	TATTGCATTCagatcggaagagcgtcgtg	10
82	TAGCAG	cacgacgctcttccgatctTAGCAGtgca	CTGCTAagatcggaagagcgtcgtg	6
83	ATCCG	cacgacgctcttccgatctATCCGtgca	CGGATagatcggaagagcgtcgtg	5
84	CTTAG	cacgacgctcttccgatctCTTAGtgca	CTAAGagatcggaagagcgtcgtg	5
85	TTATTACAT	cacgacgctcttccgatctTTATTACATtgca	ATGTAATAAagatcggaagagcgtcgtg	9
86	GCCAACAAGA	cacgacgctcttccgatctGCCAACAAGAtgca	TCTTGTTGGCagatcggaagagcgtcgtg	10
87	TGCCGCAT	cacgacgctcttccgatctTGCCGCATtgca	ATGCGGCAagatcggaagagcgtcgtg	8

88	CGTGTCA	cacgacgctcttccgatctCGTGTCAtgca	TGACACGagatcggaagagcgtcgtg	7
89	CAACCACACA	cacgacgctcttccgatctCAACCACACAtgca	TGTGTGGTTGagatcggaagagcgtcgtg	10
90	GCTCCGA	cacgacgctcttccgatctGCTCCGAtgca	TCGGAGCagatcggaagagcgtcgtg	7
91	TCAGAGAT	cacgacgctcttccgatctTCAGAGATtgca	ATCTCTGAagatcggaagagcgtcgtg	8
92	CGTTCA	cacgacgctcttccgatctCGTTCAtgca	TGAACGagatcggaagagcgtcgtg	6
93	CATCACAAG	cacgacgctcttccgatctCATCACAAGtgca	CTTGTGATGagatcggaagagcgtcgtg	9
94	TCCAG	cacgacgctcttccgatctTCCAGtgca	CTGGAagatcggaagagcgtcgtg	5
95	AACTGAAG	cacgacgctcttccgatctAACTGAAGtgca	CTTCAGTTagatcggaagagcgtcgtg	8
96	GATTCA	cacgacgctcttccgatctGATTCAtgca	TGAATCagatcggaagagcgtcgtg	6
97	CAAGCCAATT	cacgacgctcttccgatctCAAGCCAATTtgca	AATTGGCTTGagatcggaagagcgtcgtg	10
98	CAATCAT	cacgacgctcttccgatctCAATCATtgca	ATGATTGagatcggaagagcgtcgtg	7
99	ACATCACCG	cacgacgctcttccgatctACATCACCGtgca	CGGTGATGTagatcggaagagcgtcgtg	9
100	TTGCGCT	cacgacgctcttccgatctTTGCGCTtgca	AGCGCAAagatcggaagagcgtcgtg	7
101	CGCAGACACT	cacgacgctcttccgatctCGCAGACACTtgca	AGTGTCTGCGagatcggaagagcgtcgtg	10
102	TGTGGA	cacgacgctcttccgatctTGTGGAtgca	TCCACAagatcggaagagcgtcgtg	6
103	TGGATA	cacgacgctcttccgatctTGGATAtgca	TATCCAagatcggaagagcgtcgtg	6
104	ATAGCGT	cacgacgctcttccgatctATAGCGTtgca	ACGCTATagatcggaagagcgtcgtg	7
105	CCATAGA	cacgacgctcttccgatctCCATAGAtgca	TCTATGGagatcggaagagcgtcgtg	7
106	GGCACGCAT	cacgacgctcttccgatctGGCACGCATtgca	ATGCGTGCCagatcggaagagcgtcgtg	9
107	GTGTT	cacgacgctcttccgatctGTGTTtgca	AACACagatcggaagagcgtcgtg	5
108	ATTAACAATT	cacgacgctcttccgatctATTAACAATTtgca	AATTGTTAATagatcggaagagcgtcgtg	10
109	CAATA	cacgacgctcttccgatctCAATAtgca	TATTGagatcggaagagcgtcgtg	5
110	TAGTCCAT	cacgacgctcttccgatctTAGTCCATtgca	ATGGACTAagatcggaagagcgtcgtg	8
111	CGTGACCT	cacgacgctcttccgatctCGTGACCTtgca	AGGTCACGagatcggaagagcgtcgtg	8
112	CTTCAGA	cacgacgctcttccgatctCTTCAGAtgca	TCTGAAGagatcggaagagcgtcgtg	7
113	ATCTGCAACA	cacgacgctcttccgatctATCTGCAACAtgca	TGTTGCAGATagatcggaagagcgtcgtg	10
114	AAGGA	cacgacgctcttccgatctAAGGAtgca	TCCTTagatcggaagagcgtcgtg	5
115	TTACT	cacgacgctcttccgatctTTACTtgca	AGTAAagatcggaagagcgtcgtg	5
116	CCTTCG	cacgacgctcttccgatctCCTTCGtgca	CGAAGGagatcggaagagcgtcgtg	6
117	TTATCCAT	cacgacgctcttccgatctTTATCCATtgca	ATGGATAAagatcggaagagcgtcgtg	8

118	GGATTG	cacgacgctcttccgatctGGATTGtgca	CAATCCagatcggaagagcgtcgtg	6
119	GACGTGA	cacgacgctcttccgatctGACGTGAtgca	TCACGTCagatcggaagagcgtcgtg	7
120	GACGGCA	cacgacgctcttccgatctGACGGCAtgca	TGCCGTCagatcggaagagcgtcgtg	7
121	CGTCTG	cacgacgctcttccgatctCGTCTGtgca	CAGACGagatcggaagagcgtcgtg	6
122	TCTGA	cacgacgctcttccgatctTCTGAtgca	TCAGAagatcggaagagcgtcgtg	5
123	ATCTTA	cacgacgctcttccgatctATCTTAtgca	TAAGATagatcggaagagcgtcgtg	6
124	TGTATACAG	cacgacgctcttccgatctTGTATACAGtgca	CTGTATACAagatcggaagagcgtcgtg	9
125	AACTT	cacgacgctcttccgatctAACTTtgca	AAGTTagatcggaagagcgtcgtg	5
126	GAGTCACAAT	cacgacgctcttccgatctGAGTCACAATtgca	ATTGTGACTCagatcggaagagcgtcgtg	10
127	CGGTTGCAT	cacgacgctcttccgatctCGGTTGCATtgca	ATGCAACCGagatcggaagagcgtcgtg	9
128	TGTGACAAGA	cacgacgctcttccgatctTGTGACAAGAtgca	TCTTGTCACAagatcggaagagcgtcgtg	10
129	GTCCTGCCA	cacgacgctcttccgatctGTCCTGCCAtgca	TGGCAGGACagatcggaagagcgtcgtg	9
130	GTTACA	cacgacgctcttccgatctGTTACAtgca	TGTAACagatcggaagagcgtcgtg	6
131	GCGGA	cacgacgctcttccgatctGCGGAtgca	TCCGCagatcggaagagcgtcgtg	5
132	ATGATACG	cacgacgctcttccgatctATGATACGtgca	CGTATCATagatcggaagagcgtcgtg	8
133	CTGTTG	cacgacgctcttccgatctCTGTTGtgca	CAACAGagatcggaagagcgtcgtg	6
134	TCAGTAAT	cacgacgctcttccgatctTCAGTAATtgca	ATTACTGAagatcggaagagcgtcgtg	8
135	TCACA	cacgacgctcttccgatctTCACAtgca	TGTGAagatcggaagagcgtcgtg	5
136	GTCGT	cacgacgctcttccgatctGTCGTtgca	ACGACagatcggaagagcgtcgtg	5
137	ACGCTAA	cacgacgctcttccgatctACGCTAAtgca	TTAGCGTagatcggaagagcgtcgtg	7
138	ATAGG	cacgacgctcttccgatctATAGGtgca	CCTATagatcggaagagcgtcgtg	5
139	CCTGCCA	cacgacgctcttccgatctCCTGCCAtgca	TGGCAGGagatcggaagagcgtcgtg	7
140	GGTATGAAT	cacgacgctcttccgatctGGTATGAATtgca	ATTCATACCagatcggaagagcgtcgtg	9
141	TAAGACA	cacgacgctcttccgatctTAAGACAtgca	TGTCTTAagatcggaagagcgtcgtg	7
142	TGAGA	cacgacgctcttccgatctTGAGAtgca	TCTCAagatcggaagagcgtcgtg	5
143	AATGCAG	cacgacgctcttccgatctAATGCAGtgca	CTGCATTagatcggaagagcgtcgtg	7
144	CCGTGA	cacgacgctcttccgatctCCGTGAtgca	TCACGGagatcggaagagcgtcgtg	6
145	GCCAGACATT	cacgacgctcttccgatctGCCAGACATTtgca	AATGTCTGGCagatcggaagagcgtcgtg	10
146	GTGCG	cacgacgctcttccgatctGTGCGtgca	CGCACagatcggaagagcgtcgtg	5
147	TTACACA	cacgacgctcttccgatctTTACACAtgca	TGTGTAAagatcggaagagcgtcgtg	7

148	CCGTCACAGT	cacgacgctcttccgatctCCGTCACAGTtgca	ACTGTGACGGagatcggaagagcgtcgtg	10
149	CTGTGT	cacgacgctcttccgatctCTGTGTtgca	ACACAGagatcggaagagcgtcgtg	6
150	CGCGCCG	cacgacgctcttccgatctCGCGCCGtgca	CGGCGCGagatcggaagagcgtcgtg	7
151	CTAACA	cacgacgctcttccgatctCTAACAtgca	TGTTAGagatcggaagagcgtcgtg	6
152	GGCCTG	cacgacgctcttccgatctGGCCTGtgca	CAGGCCagatcggaagagcgtcgtg	6
153	TGACGT	cacgacgctcttccgatctTGACGTtgca	ACGTCAagatcggaagagcgtcgtg	6
154	ACTGAG	cacgacgctcttccgatctACTGAGtgca	CTCAGTagatcggaagagcgtcgtg	6
155	GCGCACT	cacgacgctcttccgatctGCGCACTtgca	AGTGCGCagatcggaagagcgtcgtg	7
156	GGTAAGCA	cacgacgctcttccgatctGGTAAGCAtgca	TGCTTACCagatcggaagagcgtcgtg	8
157	AATCGGAGG	cacgacgctcttccgatctAATCGGAGGtgca	CCTCCGATTagatcggaagagcgtcgtg	9
158	TGGAGCCT	cacgacgctcttccgatctTGGAGCCTtgca	AGGCTCCAagatcggaagagcgtcgtg	8
159	GATGGCCAT	cacgacgctcttccgatctGATGGCCATtgca	ATGGCCATCagatcggaagagcgtcgtg	9
160	TGCAA	cacgacgctcttccgatctTGCAAtgca	TTGCAagatcggaagagcgtcgtg	5
161	AACGG	cacgacgctcttccgatctAACGGtgca	CCGTTagatcggaagagcgtcgtg	5
162	GAGACG	cacgacgctcttccgatctGAGACGtgca	CGTCTCagatcggaagagcgtcgtg	6
163	CTTATCA	cacgacgctcttccgatctCTTATCAtgca	TGATAAGagatcggaagagcgtcgtg	7
164	CCGTACCACT	cacgacgctcttccgatctCCGTACCACTtgca	AGTGGTACGGagatcggaagagcgtcgtg	10
165	GTAACG	cacgacgctcttccgatctGTAACGtgca	CGTTACagatcggaagagcgtcgtg	6
166	TCCTCACAT	cacgacgctcttccgatctTCCTCACATtgca	ATGTGAGGAagatcggaagagcgtcgtg	9
167	TCGTA	cacgacgctcttccgatctTCGTAtgca	TACGAagatcggaagagcgtcgtg	5
168	GTATTGACT	cacgacgctcttccgatctGTATTGACTtgca	AGTCAATACagatcggaagagcgtcgtg	9
169	GCGCGAG	cacgacgctcttccgatctGCGCGAGtgca	CTCGCGCagatcggaagagcgtcgtg	7
170	GCTCA	cacgacgctcttccgatctGCTCAtgca	TGAGCagatcggaagagcgtcgtg	5
171	ACGATA	cacgacgctcttccgatctACGATAtgca	TATCGTagatcggaagagcgtcgtg	6
172	CAGTAA	cacgacgctcttccgatctCAGTAAtgca	TTACTGagatcggaagagcgtcgtg	6
173	GGAGAGCAT	cacgacgctcttccgatctGGAGAGCATtgca	ATGCTCTCCagatcggaagagcgtcgtg	9
174	CCATG	cacgacgctcttccgatctCCATGtgca	CATGGagatcggaagagcgtcgtg	5
175	CGCTCACACA	cacgacgctcttccgatctCGCTCACACAtgca	TGTGTGAGCGagatcggaagagcgtcgtg	10
176	TGTTACG	cacgacgctcttccgatctTGTTACGtgca	CGTAACAagatcggaagagcgtcgtg	7
177	GATTGGAAGA	cacgacgctcttccgatctGATTGGAAGAtgca	TCTTCCAATCagatcggaagagcgtcgtg	10

178	AATAAGAGT	cacgacgctcttccgatctAATAAGAGTtgca	ACTCTTATTagatcggaagagcgtcgtg	9
179	GAGCAA	cacgacgctcttccgatctGAGCAAtgca	TTGCTCagatcggaagagcgtcgtg	6
180	CTTCCGCAA	cacgacgctcttccgatctCTTCCGCAAtgca	TTGCGGAAGagatcggaagagcgtcgtg	9
181	TACAAG	cacgacgctcttccgatctTACAAGtgca	CTTGTAagatcggaagagcgtcgtg	6
182	TTCAGCCAGT	cacgacgctcttccgatctTTCAGCCAGTtgca	ACTGGCTGAAagatcggaagagcgtcgtg	10
183	TGAAGCAACT	cacgacgctcttccgatctTGAAGCAACTtgca	AGTTGCTTCAagatcggaagagcgtcgtg	10
184	ACAACGCAT	cacgacgctcttccgatctACAACGCATtgca	ATGCGTTGTagatcggaagagcgtcgtg	9
185	GGCGGACGA	cacgacgctcttccgatctGGCGGACGAtgca	TCGTCCGCCagatcggaagagcgtcgtg	9
186	ATCGTACGT	cacgacgctcttccgatctATCGTACGTtgca	ACGTACGATagatcggaagagcgtcgtg	9
187	AATGTA	cacgacgctcttccgatctAATGTAtgca	TACATTagatcggaagagcgtcgtg	6
188	GTACGGACG	cacgacgctcttccgatctGTACGGACGtgca	CGTCCGTACagatcggaagagcgtcgtg	9
189	CTCTCCAG	cacgacgctcttccgatctCTCTCCAGtgca	CTGGAGAGagatcggaagagcgtcgtg	8
190	TAATTG	cacgacgctcttccgatctTAATTGtgca	CAATTAagatcggaagagcgtcgtg	6
191	ATCTCGT	cacgacgctcttccgatctATCTCGTtgca	ACGAGATagatcggaagagcgtcgtg	7
192	GACAACT	cacgacgctcttccgatctGACAACTtgca	AGTTGTCagatcggaagagcgtcgtg	7
193	CTCGCAA	cacgacgctcttccgatctCTCGCAAtgca	TTGCGAGagatcggaagagcgtcgtg	7
194	TGGACACT	cacgacgctcttccgatctTGGACACTtgca	AGTGTCCAagatcggaagagcgtcgtg	8
195	TGTCAAT	cacgacgctcttccgatctTGTCAATtgca	ATTGACAagatcggaagagcgtcgtg	7
196	TCCTGCT	cacgacgctcttccgatctTCCTGCTtgca	AGCAGGAagatcggaagagcgtcgtg	7
197	GAACTT	cacgacgctcttccgatctGAACTTtgca	AAGTTCagatcggaagagcgtcgtg	6
198	ATGCT	cacgacgctcttccgatctATGCTtgca	AGCATagatcggaagagcgtcgtg	5
199	ATTCCAA	cacgacgctcttccgatctATTCCAAtgca	TTGGAATagatcggaagagcgtcgtg	7
200	GACACACT	cacgacgctcttccgatctGACACACTtgca	AGTGTGTCagatcggaagagcgtcgtg	8
201	CGCGT	cacgacgctcttccgatctCGCGTtgca	ACGCGagatcggaagagcgtcgtg	5
202	CATACGCG	cacgacgctcttccgatctCATACGCGtgca	CGCGTATGagatcggaagagcgtcgtg	8
203	CTATCACT	cacgacgctcttccgatctCTATCACTtgca	AGTGATAGagatcggaagagcgtcgtg	8
204	CTGAACCA	cacgacgctcttccgatctCTGAACCAtgca	TGGTTCAGagatcggaagagcgtcgtg	8
205	TCTCCGT	cacgacgctcttccgatctTCTCCGTtgca	ACGGAGAagatcggaagagcgtcgtg	7
206	TGTACA	cacgacgctcttccgatctTGTACAtgca	TGTACAagatcggaagagcgtcgtg	6
207	AAGCAACT	cacgacgctcttccgatctAAGCAACTtgca	AGTTGCTTagatcggaagagcgtcgtg	8

208	ACCGA	cacgacgctcttccgatctACCGAtgca	TCGGTagatcggaagagcgtcgtg	5
209	GTAAG	cacgacgctcttccgatctGTAAGtgca	CTTACagatcggaagagcgtcgtg	5
210	TGATCGCT	cacgacgctcttccgatctTGATCGCTtgca	AGCGATCAagatcggaagagcgtcgtg	8
211	TGCGG	cacgacgctcttccgatctTGCGGtgca	CCGCAagatcggaagagcgtcgtg	5
212	ACTAA	cacgacgctcttccgatctACTAAtgca	TTAGTagatcggaagagcgtcgtg	5
213	GAGGTCCT	cacgacgctcttccgatctGAGGTCCTtgca	AGGACCTCagatcggaagagcgtcgtg	8
214	TAGCTAT	cacgacgctcttccgatctTAGCTATtgca	ATAGCTAagatcggaagagcgtcgtg	7
215	CAGCGCAAGA	cacgacgctcttccgatctCAGCGCAAGAtgca	TCTTGCGCTGagatcggaagagcgtcgtg	10
216	GCTCGCCAT	cacgacgctcttccgatctGCTCGCCATtgca	ATGGCGAGCagatcggaagagcgtcgtg	9
217	TGTACCAG	cacgacgctcttccgatctTGTACCAGtgca	CTGGTACAagatcggaagagcgtcgtg	8
218	TGTACGCA	cacgacgctcttccgatctTGTACGCAtgca	TGCGTACAagatcggaagagcgtcgtg	8
219	TTGGCGCT	cacgacgctcttccgatctTTGGCGCTtgca	AGCGCCAAagatcggaagagcgtcgtg	8
220	GTTCACA	cacgacgctcttccgatctGTTCACAtgca	TGTGAACagatcggaagagcgtcgtg	7

Adapter dilution

According to Poland and colleagues (2012) the annealed adapters needed to be diluted 3:10 (~ 50ng/µl). Subsequently the adapters were quantified and normalized. However, the dilution depended on the length of the barcode and consequently the size of the forward adapter. The amount of adapter in ng/µl which corresponded to 0.1pmol/µl was calculated for each barcode length (Table S2.). On the basis of this 100µl of 0.1µM adapter solution were prepared. For this the equitation $V_1 = (c_2 * V_2)/c_1$ was used. Example calculations with an initial concentration of 50ng/µl are shown in Table S4. For the actual calculations the concentrations determined with the Qubit™ fluorometer were used.

S4. Adapter dilution. The amount of adapter which corresponded to 0.1pmol/µl was calculated using the equitation: ng=adapter length*0.1pmol*660 pg/pmol/1000. Given are the barcode and adapter lengths and the corresponding amount of adapter in ng/µl (Poland et al. 2012).

barcode length	adapter length	ng/µl adapter corresponding to 0.1 pmol/µl
5	24	1.584
6	25	1.650
7	26	1.716
8	27	1.782
9	28	1.848
10	29	1.914

S5. Example final adapter solution. The amount of adapter (µl of 3:10 dilution) for the final solution was calculated with $V_1 = (c_2 * V_2)/c_1$. For this example an initial concentration (c_1) of 50ng/µl was used.

barcode length	adapter length	µl of the 3:10 dilution	ng/µl corresponding to 0.1pmol/µl	final volume [0.1 pmol/µl]
5	24	3.168	1.584	100
6	25	3.300	1.650	100
7	26	3.432	1.716	100
8	27	3.564	1.782	100
9	28	3.696	1.848	100
10	29	3.828	1.914	100

S6. Used shell-scripts

```
#!/bin/bash
#----------
# author:  Dolezal Marlies May 2017
# RAD seq workflow

Dir=/Volumes/Temp/Schneider
#cd $Dir
#multiplexed pooled BAMs are in:
rawDataDir=/Volumes/Temp/Schneider/rawdata
#rawDataDir=/Volumes/Temp/
scriptsDir=/Volumes/Temp/Schneider/scripts/
#scriptsDir=/Volumes/Temp/Lukas/Tools/Scripts/
fastqDir=/Volumes/Temp/Schneider/fastq
annotationDir=/Volumes/Temp/Schneider/annotations
RefGenome=GCA_900078215.1_Ahal2.2_genomic.fna
sortedBAMsDir=/Volumes/Temp/Schneider/sortedBAMs
SNPdir=/Volumes/Temp/Schneider/SNPcalls
freebayesDir=/Volumes/Temp/Schneider/freebayes/bin


function SplitMultiplexedBAMfilterRestrictionEnzyme  () {
      echo "-----------------------------------------------------"
      echo "FUNCTION: $FUNCNAME"
      echo "-----------------------------------------------------"

      # author: Lukas Endler
      # date: 24.11.2016
      # to run: python split_bam_by_barcodes_radseq.py --in $1 --
tags $2 \>\> $LOGFILE 2\>\> $ERRORLOG >>  $LOGFILE

      # from the source code of "split_bam_by_barcodes_radseq.py"
      # parser.add_argument("--in", dest="infile", help="bam file
with tagged reads", required=True)
      # parser.add_argument("--tags", dest="tags", help="comma
separated list of tags (eg:
\"TGACCAAT,ACAGTGAT,GCCAATAT,CTTGTAAT\") or pairs of ids and tags
(eg:
\"light_RI:TGACCAAT,light_RII:ACAGTGAT,dark_RI:GCCAATAT,dark_RII:
CTTGTAAT\") or name of tag file", required=True)
      # parser.add_argument("--subs", dest="subs", type=int,
help="maximal number of substitutions in tag sequence
(default=1)", default=1)
      # parser.add_argument("--enz1", dest="enz1",  help="read 1
restr. enzyme overlap (default=TGCAG)", default="TGCAG")
      # parser.add_argument("--enz2", dest="enz2", help="read 2
restr. enzyme overlap (default=CGG)", default="CGG")
      # parser.add_argument("--single", dest="single",
help="single end reads (default=False)", default=False,
action="store_true")
      # parser.add_argument("--clip", dest="clip", help="clip off
barcodes (default=True)", default=True, action="store_false")

      # --in multiplexed BAM file
      # --tags barcode file
      #$scriptsDir/split_bam_by_barcodes_radseq.py --in $1 --tags
$2 \>\> $LOGFILE 2\>\> $ERRORLOG >>  $LOGFILE
```

```
	#echo "multiplexed BAM file:" $1
	#echo "barcodes file:" $2


	# FN=`basename $1 .bam`
	# echo "FN:" $FN
	# # just to replace the whole bloody name tag
	# #ATTENTION _XX needs to be adapted dependending on the
name of multiplexed BAM file!!!
	FN=${FN%_CB*}
	echo "FN:" $FN

	# LOGFILE=${FN}.log
	# echo $LOGFILE
	# ERRORLOG=${FN}.err.log
	# echo $ERRORLOG

	# echo at `date` >> $LOGFILE
	# # echo python ${SCRIPTS}/split_bam_by_barcodes_radseq.py -
-in $1 --tags $2 \>\> $LOGFILE 2\>\> $ERRORLOG >>  $LOGFILE
	# #python split_bam_by_barcodes_radseq.py --in $1 --tags $2
>> $LOGFILE 2>> $ERRORLOG


	echo python /split_bam_by_barcodes_radseq.py --in $1 --tags
$2
	python $scriptsDir/split_bam_by_barcodes_radseq.py --in $1 -
-tags $2
	# # #splits multiplexed RADseq BAM files by multiplex tags
and filters for reads containing the restricition enzyme
	# ES=$?
	# echo finished splitting bam file at `date` with exit state
$ES >> $LOGFILE

	# for i in ${FN}_[ACTG]*.bam;do
		# #SP=`basename $i .bam` --> replace the long sample
name to Pool_barcode_samplenumber.bam
		# #bamToFastq -i $i -fq ${SP}_1.fq -fq2 ${SP}_2.fq  >>
$LOGFILE 2>> $ERRORLOG
		# #generates fastq files out of the bam files, so that
they run in bwa
		# #mkdir $FN
	  # # bash /Volumes/Temp/Kathi/run_bwa.sh $i $REFGENOME '.'
>> $LOGFILE 2>> $ERRORLOG
	# done
}
#SplitMultiplexedBAMfilterRestrictionEnzyme
Pool_343a_CB14CANXX_6_20170518B_20170523.bam
barcodes_pool343a_sample221_417.txt


function BamToFastQandFastQC () {
	echo "-----------------------------------------------------"
	echo "FUNCTION: $FUNCNAME"
	echo "-----------------------------------------------------"
	cd /Volumes/Temp/Schneider/splitBAMs
	fastqDir=/Volumes/Temp/Schneider/fastq
```

```
      #bedtools Version: v2.25.0
      # # FastQC
      for i in *; do
            echo $i          #eg 221.bam
            echo ${i%.*}     #eg 221

            bedtools bamtofastq -i $i -fq $fastqDir/${i%.*}_1.fq -
fq2 $fastqDir/${i%.*}_2.fq

            echo "fastqc"
            fastqc -t 15 $fastqDir/${i%.*}_1.fq
            fastqc -t 15 $fastqDir/${i%.*}_2.fq
            #echo "gzip"
            gzip $fastqDir/${i%.*}_1.fq $fastqDir/${i%.*}_2.fq
            let counter=counter+1
            echo "----------------------"
      done
}
#BamToFastQandFastQC


function ParseFASTQC () {
      echo "-----------------------------------------------------"
      echo "FUNCTION: $FUNCNAME"
      echo "-----------------------------------------------------"
      #
https://gist.github.com/danielecook/8e9afb2d2df7752efd8a#file-
fastqc_aggregate-sh FastQC_aggregate.sh

      fastqDir=/Volumes/Temp/Schneider/fastq
      cd $fastqDir

      zips=`ls *.zip`
      for i in $zips; do
            unzip -o $i &>/dev/null;
      done
      fastq_folders=${zips/.zip/}


      if [ ! -d FASTQCall ]; then
            mkdir FASTQCall
      fi

      if [ -e FASTQCall/FASTQC.all.summmary.out ]; then
            rm FASTQCall/FASTQC.all.summmary.out
      fi

      # Concatenate Statistics
      for folder in $fastq_folders; do
            folder=${folder%.*}

            sampleID=`grep 'Filename' ${folder}/fastqc_data.txt`
            Encoding=`grep 'Encoding' ${folder}/fastqc_data.txt`
            TotalSequences=`grep 'Total Sequences'
${folder}/fastqc_data.txt`
            PoorQualSequences=`grep 'Sequences flagged as poor
quality' ${folder}/fastqc_data.txt`
```

```
            SequenceLength=`grep 'Sequence length'
${folder}/fastqc_data.txt`
            PercentageGC=`grep '%GC' ${folder}/fastqc_data.txt`
            BasicStatistics=`grep 'Basic Statistics'
${folder}/fastqc_data.txt`
            PerBaseSequenceQual=`grep 'Per base sequence quality'
${folder}/fastqc_data.txt`
            PerTileSequenceQual=`grep 'Per tile sequence quality'
${folder}/fastqc_data.txt`
            PerSequenceQualScores=`grep 'Per sequence quality
scores' ${folder}/fastqc_data.txt`
            PerBaseSequenceContent=`grep 'Per base sequence
content' ${folder}/fastqc_data.txt`
            PerSequenceGCcontent=`grep 'Per sequence GC content'
${folder}/fastqc_data.txt`
            PerBaseNcontent=`grep 'Per base N content'
${folder}/fastqc_data.txt`
            SequenceLengthDistribution=`grep 'Sequence Length
Distribution' ${folder}/fastqc_data.txt`
            SequenceDuplicationLevels=`grep 'Sequence Duplication
Levels' ${folder}/fastqc_data.txt`
            TotalDeduplicatedPercentage=`grep 'Total Deduplicated
Percentage' ${folder}/fastqc_data.txt`
            OverrepresentedSequences=`grep 'Overrepresented
sequences' ${folder}/fastqc_data.txt`
            AdapterContent=`grep 'Adapter Content'
${folder}/fastqc_data.txt`
            KmerContent=`grep 'Kmer Content'
${folder}/fastqc_data.txt`


            cat >>  FASTQCall/FASTQC.all.summmary.out << EOF
$sampleID  $Encoding  $TotalSequences  $PoorQualSequences
      $SequenceLength  $PercentageGC    \
$BasicStatistics $PerBaseSequenceQual   $PerTileSequenceQual
      $PerSequenceQualScores $PerBaseSequenceContent
      $PerSequenceGCcontent  \
$PerBaseNcontent $SequenceLengthDistribution
      $SequenceDuplicationLevels  $TotalDeduplicatedPercentage
      $OverrepresentedSequences   \
$AdapterContent  $KmerContent
EOF
            echo FASTQC.all.summmary.out
            # Filename
            # Encoding
            # Total Sequences
            # Sequences flagged as poor quality
            # Sequence length
            # %GC
            # >>Basic Statistics
            # >>Per base sequence quality
            # >>Per tile sequence quality
            # >>Per sequence quality scores
            # >>Per base sequence content
            # >>Per sequence GC content
            # >>Per base N content
            # >>Sequence Length Distribution
            # >>Sequence Duplication Levels
```

```
            # Total Deduplicated Percentage
            # >>Overrepresented sequences
            # >>Adapter Content
            # >>Kmer Content
            rm -rf ${folder}
      done
      sed 's/\#//' FASTQC.all.summmary.out >
FASTQC.all.summmary.final.out
      # this line is needed bc one of the columns is # Total
Deduplicated Percentage and R ignores everything after #
}
#ParseFASTQC


function StatsSummary () {
      echo "------------------------------------------------------"
      echo "FUNCTION: $FUNCNAME"
      echo "------------------------------------------------------"
      cd $sortedBAMsDir

      for IDs in {221..417}
      do
            # echo "samtools index started  at" `date`
            # samtools index $sortedBAMsDir/${IDs}.sorted.bam
            # echo "samtools flagstat started  at" `date`
            # samtools flagstat $sortedBAMsDir/${IDs}.sorted.bam
>> $sortedBAMsDir/${IDs}.flagstats
            # echo "samtools idxstats started  at" `date`
            # samtools idxstats $sortedBAMsDir/${IDs}.sorted.bam
>> $sortedBAMsDir/${IDs}.idxstats
            # echo "samtools stats started  at" `date`
            # samtools stats $sortedBAMsDir/${IDs}.sorted.bam
>>$sortedBAMsDir/${IDs}.stats


            QCpassedTotalReads=$(awk 'NR==1{print $1}'
${IDs}.flagstats)
            QCfailedTotalReads=$(awk 'NR==1{print $3}'
${IDs}.flagstats)
            secondaryReads=$(awk 'NR==2{print $1}'
${IDs}.flagstats)
            mappedReads=$(awk 'NR==5{print $1}' ${IDs}.flagstats)
            pairedReads=$(awk 'NR==6{print $1}' ${IDs}.flagstats)
            properlyPairedReads=$(awk 'NR==9{print $1}'
${IDs}.flagstats)
            withItselfAndMmateMapped=$(awk 'NR==10{print $1}'
${IDs}.flagstats)
            singletons=$(awk 'NR==11{print $1}' ${IDs}.flagstats)
            withMateMappedToDiffChromosome=$(awk 'NR==12{print
$1}' ${IDs}.flagstats)
            withMateMappedToDiffChromosomeMapQ5=$(awk
'NR==132{print $1}' ${IDs}.flagstats)

            echo "ID:" ${IDs} $QCpassedTotalReads
$QCfailedTotalReads $secondaryReads $mappedReads $pairedReads
$properlyPairedReads $withItselfAndMmateMapped $singletons
$withMateMappedToDiffChromosome
$withMateMappedToDiffChromosomeMapQ5
```

```
            echo ${IDs} $QCpassedTotalReads $QCfailedTotalReads
$secondaryReads $mappedReads $pairedReads $properlyPairedReads
$withItselfAndMmateMapped $singletons
$withMateMappedToDiffChromosome
$withMateMappedToDiffChromosomeMapQ5 >> Pool343.summary.flagstats

cat >>  Pool343.summary.flagstats << EOF
${IDs} $QCpassedTotalReads $QCfailedTotalReads $secondaryReads
$mappedReads $pairedReads $properlyPairedReads
$withItselfAndMmateMapped $singletons
$withMateMappedToDiffChromosome
$withMateMappedToDiffChromosomeMapQ5
EOF

      done

}
#StatsSummary


function PrepareReference () {
      echo "------------------------------------------------------"
      echo "FUNCTION: $FUNCNAME"
      echo "------------------------------------------------------"
      cd $annotationDir

      # echo "samtools faidx $REFGENOME "
      # samtools faidx $RefGenome
      # echo "bwa index $RefGenome  "
      # bwa index $RefGenome
      # echo "CreateSequenceDictionary"`date`
      java -jar  /Volumes/Temp/Mueller/picard.jar
CreateSequenceDictionary R= $RefGenome O= $RefGenome.dict
}
#PrepareReference


function Mapping () {
      echo "------------------------------------------------------"
      echo "FUNCTION: $FUNCNAME"
      echo "------------------------------------------------------"

      for IDs in {1..417}
      do
      LOGFILE=$Dir/${IDs}.log
      ERRORLOG=$Dir/${IDs}.err.log
            echo "IDs:" $IDs
            echo "${IDs}:" ${IDs}
            echo "start bwa mem  at" `date`
            /Volumes/Temp/Mueller/tools/bwa-0.7.15/bwa mem -R
"@RG  ID:${IDs}  SM:${IDs}" -V -t 40 $annotationDir/$RefGenome
$fastqDir/${IDs}_1.fq.gz $fastqDir/${IDs}_2.fq.gz 2>>
$ERRORLOG | samtools view -Shb - | samtools sort -m
10000000000 - > $sortedBAMsDir/${IDs}.sorted.bam
            echo "finished with bwa at"`date` with exit state
$ES
      done
```

```
}
#Mapping

function freebayes () {
echo "-----------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "-----------------------------------------------------------"

#https://github.com/ekg/freebayes
#Garrison E, Marth G. Haplotype-based variant detection from
short-read sequencing. arXiv preprint arXiv:1207.3907 [q-bio.GN]
2012
#git clone --recursive git://github.com/ekg/freebayes.git

#vetgrid10 : version:  v1.1.0-dirty
#vetlinux:

#make
#sudo make install

cd $sortedBAMsDir/all
echo "start freebayes  at" `date`
      $freebayesDir/freebayes \
      --use-best-n-alleles 4 \
      --genotype-qualities \
      --fasta-reference $annotationDir/$RefGenome \
      --bam-list $annotationDir/populations/all.bamlist \
      > $SNPdir/all.vcf
      echo "finished with freebayes at"`date`


      pops=( badr blai clau czra czrb czrc fort laut litt vieb
wulm )
      counter=0

      for pop in "${pops[@]}"
      do
            echo ${pops[$counter]}
            if [ ! -e $annotationDir/$pop ]
            then mkdir $annotationDir/$pop
            fi
            cd $sortedBAMsDir

            freebayes   \
            --use-best-n-alleles 4  \
            --fasta-reference $annotationDir/$RefGenome  \
            --bam-list $anotationDir/populations/$pop  \
            > $SNPdir/raw.$pop.vcf

            let counter=counter+1


      done

}
#freebayes
```

```
function RenameBAMs () {
      echo "-----------------------------------------------------"
      echo "FUNCTION: $FUNCNAME"
      echo "-----------------------------------------------------"
      #cd /Volumes/Temp/Schneider/splitBAMs
      cd /Volumes/Temp/Schneider/sortedBAMs/Pool337

      for i in *; do
      echo $i
      echo ${i##*_}
      mv $i ${i##*_}
      ls -lh
      done
}
#RenameBAMs



#!/bin/bash
#----------
# author:  Dolezal Marlies and Schneider Katharina
# workflow data analysis

Dir=/Volumes/Temp/Schneider
SNPdir=/Volumes/Temp/Schneider/SNPcalls
annotationDir=/Volumes/Temp/Schneider/annotations
RefGenome=GCA_900078215.1_Ahal2.2_genomic.fna

#populations:
#badr blai clau czrb crza czrc fort laut litt vieb wulm
#population1=badr
#population2=blai
#population3=clau
#population4=fort
#population5=laut
#population6=litt
#population7=vieb
#poopulation8=wulm
#population9=czra
#population10=czrc

function RepeatMasked () {
echo "-----------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "-----------------------------------------------------------"

summaryfile=SNPfiltering.RepeatMasked.summary

if [ -e $SNPdir/$summaryfile ]; then
      echo "-----------------------------------------------------"
      echo "deleting $SNPdir/$summaryfile"
      echo "-----------------------------------------------------"
            rm $SNPdir/$summaryfile
      fi

      cat >> $SNPdir/$summaryfile << EOF
population numberloci numberlociRepeatMasked
```

```
EOF

      populations=( badr blai clau czra czrb czrc fort laut litt
vieb wulm )
            counter=0
            for pop in "${populations[@]}"
            do
            echo "counter:$counter"
            echo "population:  ${populations[$counter]}"
            numberloci=`grep -v '#'
$SNPdir/raw.${populations[$counter]}.vcf | wc -l`
            echo "number loci": $numberloci

            #extract polymorphisms in RepeatMasker regions

            vcftools --vcf
$SNPdir/raw.${populations[$counter]}.vcf \
            --positions $annotationDir/RepeatMasker.bed \
            --recode \
            --out $SNPdir/${populations[$counter]}.RepeatMasked

            numberlociRepeatMasked=`grep -v '#'
$SNPdir/${populations[$counter]}.RepeatMasked.recode.vcf | wc -l`
            echo "number loci on RepeatMasked":
$numberlociRepeatMasked

            cat >> $SNPdir/$summaryfile << EOF
${populations[$counter]} $numberloci $numberlociRepeatMasked
EOF
            let counter=counter+1
            echo "----------------------"
      done
}
#RepeatMasked

function Chloroplasts () {
echo "------------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "------------------------------------------------------------"

summaryfile=SNPfiltering.chloroplasts.summary

      if [ -e $SNPdir/$summaryfile ]; then
            echo "-----------------------------------------------"
            echo "deleting $SNPdir/$summaryfile"
            echo "-----------------------------------------------"
            rm $SNPdir/$summaryfile
      fi

      cat >> $SNPdir/$summaryfile << EOF
population numberloci numberlocichloroplasts
EOF


      populations=( badr blai clau czra czrb czrc fort laut litt
vieb wulm )
            counter=0
            for pop in "${populations[@]}"
```

```
            do
            echo "counter:$counter"
            echo "population:  ${populations[$counter]}"
            numberloci=`grep -v '#'
$SNPdir/raw.${populations[$counter]}.vcf | wc -l`
            echo "number loci": $numberloci

            #extract polymorphisms on chloroplasts

            vcftools --vcf
$SNPdir/raw.${populations[$counter]}.vcf \
            --positions $annotationDir/CPcontig.bed \
            --recode \
            --out $SNPdir/${populations[$counter]}.chloroplasts

            numberlocichloroplasts=`grep -v '#'
$SNPdir/${populations[$counter]}.chloroplasts.recode.vcf | wc -l`
            echo "number loci on chloroplasts":
$numberlocichloroplasts

            cat >> $SNPdir/$summaryfile << EOF
${populations[$counter]} $numberloci $numberlocichloroplasts
EOF


            let counter=counter+1
            echo "----------------------"
      done
}
#Chloroplasts



function FilterVCFs () {
echo "-----------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "-----------------------------------------------------------"

summaryfile=SNPfiltering.summary
      if [ -e $SNPdir/$summaryfile ]; then
            echo "-----------------------------------------------"
            echo "deleting $SNPdir/$summaryfile"
            echo "-----------------------------------------------"
            rm $SNPdir/$summaryfile
      fi

      cat >> $SNPdir/$summaryfile << EOF
population numberloci numberlocifiltered numberSNPs
EOF

      populations=( badr blai clau czra czrb czrc fort laut litt
vieb wulm )
            counter=0
            for pop in "${populations[@]}"
            do
            echo "counter:$counter"
            echo "population:  ${populations[$counter]}"
```

```
            numberloci=`grep -v '#'
$SNPdir/raw.${populations[$counter]}.vcf | wc -l`
            echo "number loci": $numberloci

            #remove indels
            vcftools --vcf
$SNPdir/raw.${populations[$counter]}.vcf \
            --recode \
            --max-alleles 2 \
            --remove-indels \
            --out $SNPdir/${populations[counter]}.vcf

            #filter regions that are repeat masked plus in the
chloroplasts
            vcftools --vcf $SNPdir/${populations[$counter]}.vcf \
            --exclude-positions
$annotationDir/RepeatMaskedChloroplasts.bed \
            --recode \
            --out $SNPdir/${populations[$counter]}.recode.vcf

            numberlocifiltered=`grep -v '#'
$SNPdir/${populations[$counter]}.recode.vcf | wc -l`
            echo "number loci filtered": $numberlocifiltered

            #filter for biallelic SNPs and quality
            echo "start GATK for ${populations[$counter]}"
            java -Xmx20g -jar
/Volumes/Temp/Mueller/tools/GenomeAnalysisTK.jar \
            -T  SelectVariants \
            -R $annotationDir/$RefGenome \
            --variant $SNPdir/${populations[$counter]}.recode.vcf
\
            --restrictAllelesTo BIALLELIC \
            --selectTypeToInclude SNP \
            -select "QUAL > 100.0" \
            -o $SNPdir/${populations[$counter]}.final.vcf

            numberSNPs=`grep -v '#'
$SNPdir/${populations[$counter]}.final.vcf | wc -l`
            echo "numberSNPs": $numberSNPs

            #filter for loci with less than 50% missing data
            vcftools --vcf
$SNPdir/${populations[$counter]}.final.vcf \
            -- keep $annotationDir/Popmap_keep.txt\
            --recode \
            --out
$SNPdir/${populations[$counter]}.indfilter.recode.vcf


            cat >> $SNPdir/$summaryfile << EOF
${populations[$counter]} $numberloci $numberlocifiltered
$numberSNPs
EOF
            let counter=counter+1
            echo "----------------------"
      done
}
```

```
#FilterVCFs

function CompareVCFs () {
echo "------------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "------------------------------------------------------------"

java -Xmx20g -jar
/Volumes/Temp/Mueller/tools/GenomeAnalysisTK.jar \
      -T CombineVariants \
      -R $annotationDir/$RefGenome \
      --variant:$population1 $SNPdir/raw.$population1.vcf \
      --variant:$population2 $SNPdir/raw.$population2.vcf \
      -o $SNPdir/$population1.$population2.union

java -Xmx20g -jar
/Volumes/Temp/Mueller/tools/GenomeAnalysisTK.jar \
      -T SelectVariants \
      -R $annotationDir/$RefGenome \
      --variant $SNPdir/$population.$population2.union \
      -select set == "Intersection" \
      -o $SNPdir/$population.$population2.concordance

java -Xmx20g -jar
/Volumes/Temp/Mueller/tools/GenomeAnalysisTK.jar \
      -T SelectVariants \
      -R $annotationDir/$RefGenome \
      --variant $SNPdir/raw.$population1.vcf \
      --concordance $SNPdir/raw.$population2.vcf \
      -o $SNPdir/$population.$population2.concordance

java -Xmx20g -jar
/Volumes/Temp/Mueller/tools/GenomeAnalysisTK.jar \
      -T SelectVariants \
      -R $annotationDir/$RefGenome \
      --variant $SNPdir/raw.$population1.vcf \
      --discordance $SNPdir/raw.$population2.vcf \
      -o $SNPdir/$population.$population2.discordance

java -Xmx20g -jar
/Volumes/Temp/Mueller/tools/GenomeAnalysisTK.jar \
      -T SelectVariants \
      -R $annotationDir/$RefGenome \
      --variant $SNPdir/raw.$population2.vcf \
      --discordance $SNPdir/raw.$population1.vcf \
      -o $SNPdir/$population2.$population1.discordance


      numberloci_population2=`grep -v '#'
$SNPdir/raw.$population2.vcf | wc -l`
      union_population1_population2=`grep -v '#'
$SNPdir/$population1.$population2.union | wc -l`
      concordance_population1_population2=`grep -v '#'
$SNPdir/$population1.$population2.concordance | wc -l`
      discordance_population1_population2=`grep -v '#'
$SNPdir/$population1.$population2.discordance | wc -l`
      discordance_population2_population1=`grep -v '#'
$SNPdir/$population2.$population1.discordance | wc -l`
```

```
      echo "population1: "$population1
      echo "population2: "$population2

      echo "numberloci population1:" $numberloci_population1
      echo "numberloci population2:" $numberloci_population2

      echo "union_population1_population2"
$union_population1_population2
      echo "concordance_population1_population2"
$concordance_population1_population2
      echo "discordance_population1_population2"
$discordance_population1_population2
      echo "discordance_population2_population1"
$discordance_population2_population1

#private sites per population (exemplary for population wulm)

java -Xmx20g -jar
/Volumes/Temp/Mueller/tools/GenomeAnalysisTK.jar \
           -T SelectVariants \
           -R $annotationDir/$RefGenome \
           --variant $SNPdir/raw.$population1.vcf \
           --variant $SNPdir/raw.$population2.vcf \
           --variant $SNPdir/raw.$population3.vcf \
           --variant $SNPdir/raw.$population4.vcf \
           --variant $SNPdir/raw.$population5.vcf \
           --variant $SNPdir/raw.$population6.vcf \
           --variant $SNPdir/raw.$population7.vcf \
           --variant $SNPdir/raw.$population8.vcf \
           --variant $SNPdir/raw.$population9.vcf \
           -o $SNPdir/disc.wulm.unique

} CompareVCFs

function RunMultisamplecalling () {
echo "-----------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "-----------------------------------------------------------"

Pic=/Volumes/Temp/Kathi/picard-tools-1.136
Dir=/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsresults/
out=/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsresults/privateS
ites
Ref=/Volumes/Temp/Kathi/Genomes/
Jar=/Volumes/Temp/Kathi/GenomeAnalysisTK-3.7-0

cd$Dir

java -jar $Jar/GenomeAnalysisTK.jar \
   -T CombineVariants \
   -R $Ref/GCA_900078215.1_Ahal2.2_genomic.fna\
   --variant $Dir/badr.indfilter.recode.vcf \
   --variant $Dir/blai.indfilter.recode.vcf \
   --variant $Dir/clau.indfilter.recode.vcf \
   --variant $Dir/fort.indfilter.recode.vcf \
   --variant $Dir/laut.indfilter.recode.vcf \
   --variant $Dir/litt.indfilter.recode.vcf \
```

```
   --variant $Dir/vieb.indfilter.recode.vcf \
   --variant $Dir/wulm.indfilter.recode.vcf \
   --variant $Dir/czra.indfilter.recode.vcf \
   --variant $Dir/czrb.indfilter.recode.vcf \
   -o $out/no_czrc.vcf \
   -genotypeMergeOptions UNIQUIFY

java -jar $Jar/GenomeAnalysisTK.jar \
   -T SelectVariants \
   -R $Ref/GCA_900078215.1_Ahal2.2_genomic.fna\
   --variant $Dir/czrb.indfilter.recode.vcf \
   --discordance $out/no_czrb.vcf \
   -o $out/czrb_private\


pops=( badr blai clau czra czrb czrc fort laut litt vieb wulm )
      counter=0

      for pop in "${pops[@]}"
      do

            echo ${pops[$counter]}

            #if [ ! -e $Dir/$pop ]
            #then mkdir $Dir/$pop
            #fi

            cd $Dir

            echo "start singletons  at" `date`
            vcftools   \
            —vcf $fileDir/$pop_private  \
            ——singletons \
            ——out $pop
            echo "finished with singletons at"`date`
            let counter=counter+1


      done


      cd$Dir

java -jar $Jar/GenomeAnalysisTK.jar \
   -T CombineVariants \
   -R $Ref/GCA_900078215.1_Ahal2.2_genomic.fna\
   --variant $Dir/badr_private \
   --variant $Dir/blai_private \
   --variant $Dir/clau_private \
   --variant $Dir/fort_private \
   --variant $Dir/laut_private \
   --variant $Dir/litt_private \
   --variant $Dir/vieb_private \
   --variant $Dir/wulm_private \
   --variant $Dir/czra_private \
   --variant $Dir/czrb_private \
   --variant $Dir/czrc_private \
   -o $out/privateSet.vcf \
```

```
   -genotypeMergeOptions UNIQUIFY


java -jar $Jar/GenomeAnalysisTK.jar \
   -T SelectVariants \
   -R $Ref/GCA_900078215.1_Ahal2.2_genomic.fna\
   --variant $Dir/privateSet.vcf\
   --concordance $Multi/Multisample.final.vcf \
   -o  $Dir/Concordance

vcftools  --vcf Concordance --geno-depth --out concordance


}
RunMultisamplecalling


function RunCreateCluster () {
echo "------------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "------------------------------------------------------------"


ClusterDir=/Volumes/Temp/Kathi/populationmaps

           grep -w "\wulm\" $ClusterDir/Metainformation >>
$ClusterDirCluster1
        grep -w "\fort\" $ClusterDirMetainformation >>
$ClusterDirCluster1

        grep -w "\czra\" $ClusterDir/Metainformation >>
$ClusterDirCluster2
        grep -w "\czrb\" $ClusterDir/Metainformation >>
$ClusterDir/Cluster2

}
RunCreateCluster

function Runvcftools () {
echo "------------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "------------------------------------------------------------"

      #http://vcftools.sourceforge.net/man_latest.html

      SNPdir=/Volumes/Temp/Schneider/SNPcalls/individualfiltered
      outDir=/Volumes/Temp/Schneider/SNPcalls/individualfiltered
      cd $SNPdir

      summaryfile=VCFtools.individualfiltered.summary

      if [ -e $outDir/$summaryfile ]; then
            echo "------------------------------------------------"
            echo "deleting $SNPdir/$summaryfile"
            echo "------------------------------------------------"
            rm $outDir/$summaryfile
      fi
```

```
# cat >> $outDir/$summaryfile << EOF
# population numberSNPs nrows_frq nrows_counts nrows_depth
nrows_imiss nrows_lmiss nrows_singletons nrows_TajimaD
nrows_sitepi nrows_het
# EOF

      pop=( badr blai clau czra czrb czrc fort laut litt vieb wulm
)
            counter=0
            for pop in "${pop[@]}"
            do
            echo "counter:$counter"
            echo "population:  ${pop[$counter]}"
            file=${populations[$counter]}
            $population.indfilter.recode.vcf


            vcftools --vcf
$SNPdir/${populations[$counter]}.indfilter.recode.vcf --freq --
out $outDir/${populations[$counter]}
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
counts --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
depth --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
site-depth --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
site-mean-depth --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
geno-depth --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
site-quality --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
missing-indv --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
missing-site --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
singletons --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
site-pi --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
het --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
relatedness2 --out $outDir/$pop
            vcftools --vcf $SNPdir/$file.indfilter.recode.vcf  --
weir-fst-pop $annotationDir/$pop.map --out $outDir/$pop

            # echo "number of variable sites in $file"
            # grep -v '#' $file.indfilter.recode.vcf | wc -l
>>$outDir/$summaryfile

            # # numberSNPs=`grep -v '#'
$SNPdir/$file.indfilter.recode.vcf | wc -l`
            # # nrows_frq=`grep -v '#'
$SNPdir/$file.indfilter.recode.vcf | wc -l`
            # # nrows_counts=
            # # nrows_depth=
            # # nrows_imiss=
```

```
            # # nrows_lmiss=
            # # nrows_singletons=
            # # nrows_sitepi=
            # # nrows_het=


                # # cat >> $outDir/$summaryfile << EOF
# # $pop numberSNPs nrows_frq nrows_counts nrows_depth
nrows_imiss nrows_lmiss nrows_singletons nrows_TajimaD
nrows_sitepi nrows_het
# # EOF


            # echo "--------------------------------------"
>>$outDir/$summaryfile
             echo $pop  >>$outDir/$summaryfile
             grep -v '#' $file.indfilter.recode.vcf | wc -l
>>$outDir/$summaryfile

            # wc -l $outDir/$file.frq  >>$outDir/$summaryfile
            # wc -l $outDir/$file.frq.count >>$outDir/$summaryfile
            # wc -l $outDir/$file.idepth >>$outDir/$summaryfile
            # wc -l $outDir/$file.ldepth >>$outDir/$summaryfile
            # wc -l $outDir/$file.ldepth.mean
>>$outDir/$summaryfile
            # wc -l $outDir/$file.gdepth >>$outDir/$summaryfile
            # wc -l $outDir/$file.imiss >>$outDir/$summaryfile
            # wc -l $outDir/$file.lmiss >>$outDir/$summaryfile
            # wc -l $outDir/$file.lqual >>$outDir/$summaryfile
            # wc -l $outDir/$file.sites.pi >>$outDir/$summaryfile
            # wc -l $outDir/$file.het >>$outDir/$summaryfile
            # wc -l $outDir/$file.relatedness
>>$outDir/$summaryfile
            # wc -l $outDir/$file.relatedness
>>$outDir/$summaryfile
            # wc -l $outDir/$file.Tajima.D  >>$outDir/$summaryfile
            # wc -l $outDir/$file.singletons
>>$outDir/$summaryfile

            egrep -v '#' $outDir/$file.lmiss |  awk
'{OFS=FS="\t"}; {if ($6 ==0) {print $0}}' >
$outDir/$file.nonmiss
            wc -l  $outDir/$file.nonmiss  >>$outDir/$summaryfile

            let counter=counter+1
            echo "----------------------"
      done
}
Runvcftools

#relatedness
relDir=/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsresults/cutof
f

function RunFindSelfPairs () {
echo "-----------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "-----------------------------------------------------------"
```

```
      cd $relDir/

      echo "start SelfPairs  at" `date`
            grep -w '0.5' rel2_paired.txt > Self_paired
      echo "finished with SelfPairs at" `date`

}
RunFindSelfPairs



function RunRemoveSelfPairs () {
echo "------------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "------------------------------------------------------------"

      cd $relDir/

      echo "start Remove SelfPairs  at" `date`
            awk 'NR==FNR{a[$0];next} !($0 in a)' Self_paired
rel2_paired.txt > UnPaired_final
      echo "finished with Remove SelfPairs at" `date`

}
RunRemoveSelfPairs

function RunUniqueSamples() {
echo "------------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "------------------------------------------------------------"

      cd $relDir/

      echo "start Unique  at" `date`
            sort -u -k3,3 -k4,4 -k7,7 -k8,8 Paired_samples >
Unique_Paired
            sort -u -k3,3 -k4,4 -k7,7 -k8,8 unpaired_samples >
Unique_Unpaired
      echo "finished with Unique at" `date`

}
RunUniqueSamples

function RunUniqueSet() {
echo "------------------------------------------------------------"
echo "FUNCTION: $FUNCNAME"
echo "------------------------------------------------------------"

      cd $relDir/

      echo "start Unique  at" `date`
            sort -u -k3,3 -k4,4 -k7,7 -k8,8 Unpaired_final >
Unique_Set_final
      echo "finished with Unique at" `date`

}
RunUniqueSet
```

S7. Used R-scripts

```
#------------------------------------------------------------
#Author Dolezal Marlies
#Date May / June 2017
#Descriptive Statistics for FASTQC summary files
#------------------------------------------------------------
setwd("R://WORKDATA//BioinformatikPlattform//Schneider")
getwd()
library(car)
#readingin flagstats summary data:
#reading in FASTQC summary data:
input_<-
read.table(file="R://WORKDATA//BioinformatikPlattform//Schneider/
/FASTQC.all.summmary.final.out", sep="\t", header=F, skip = 0, na
= ".")

#merge with barcodes
#correlation btw barcode length and total sequence!!!!!

input <- input_[-c(1, 3:5,
7,9,11,13,15,17,19,21,23,25,27,29,31,33,35,37,39,40)]
colnames(input)<-c("sampleID","TotalSequence",
                   "poorqual",
                   "length",
                   "GCpercent",
                   "BasicStats",
                   "PerBaseSeqQual",
                   "PerTileSeqQual",
                   "PerSeqQualScores",
                   "PerBaseSeqContent",
                   "PerSquenceGC",
                   "PerBaseNcontent",
                   "SequenceLengthDistribution",
                   "SequenceDuplicationLevels",
                   "DeduplicatedPercentage",
                   "OverrepresentedSequences",
                   "AdapterContent",
                   "KmerContent")

summary(input$DeduplicatedPercentage)

hist(input$length, breaks=20)


input$DuplicatedPercentage<-100-input$DeduplicatedPercentage

plot(input$DeduplicatedPercentage,input$DuplicatedPercentage)

png(file="R://WORKDATA//BioinformatikPlattform//Schneider//Histog
ram_DuplicationPercentag.png")
hist(input$DuplicatedPercentage, breaks=20,main="FastQC 417
samples duplication %")
dev.off()
```

```
png(file="R://WORKDATA//BioinformatikPlattform//Schneider//Plot_D
uplicated_TotalNumberReads.png")
plot(input$DuplicatedPercentage,log10(input$TotalSequence))
dev.off()
plot(input$DeduplicatedPercentage,input$TotalSequence)


cor.test(input$TotalSequence,input$DuplicatedPercentage,method="s
pearman")
plot(log10(input$TotalSequence),input$DuplicatedPercentage)


png(file="R://WORKDATA//BioinformatikPlattform//Schneider//Histog
ram_TotalNumberReads.png")
hist(input$TotalSequence,breaks=20, main="FastQC 417 samples
total number of reads")
dev.off()


png(file="R://WORKDATA//BioinformatikPlattform//Schneider//Histog
ram_Log10TotalNumberReads.png")
hist(log10(input$TotalSequence),breaks=20, main="FastQC 417
samples total number of reads")
dev.off()

hist(log10(input$TotalSequence),breaks=20)
summary(input$TotalSequence)
hist(input$GCpercent,breaks=20)
plot(input$GCpercent,input$DuplicatedPercentage)

hist(input$DeduplicatedPercentage, breaks=20)
plot(input$TotalSequence,input$DuplicatedPercentage)
plot(log10(input$TotalSequence),input$DuplicatedPercentage)



#-------------------------------------------------------------
#Descriptive Statistics for vcftools results
#-------------------------------------------------------------

#install.packages("plotly")
library(plotly)
#install.packages("car")
library(car)
library(nortest)
#setwd("R://WORKDATA//BioinformatikPlattform//Schneider//vcftools
")

setwd("R://WORKDATA//BioinformatikPlattform//Schneider//individua
lfiltered")
#setwd("D://BioinformatikPlattform//Schneider//")
getwd()
#outDir<-("D://BioinformatikPlattform//Schneider//vcftools")
outDir<-
("R://WORKDATA//BioinformatikPlattform//Schneider//individualfilt
ered")
```

```
#install.packages("scatterplot3d")
library(scatterplot3d)
library(rgl)
library(corrplot)
library(corrgram)


#VCFTOOLS STATS
#-----------------------------------------------------

populations<-c("badr","blai","clau","czra", "czrb", "czrc",
"fort", "laut" ,"litt" ,"vieb" ,"wulm")
populations
#enter here max % missing loci per sample
cutoff=0.35

print ("number of samples before & after filtering")
for (i in populations) {
  print(i)
  imiss<-read.table(file=paste(i,".imiss", sep=""), sep="\t",
header=TRUE, skip = 0, na = "." )

  filtered<-subset(imiss,F_MISS<=cutoff)
  print(paste(nrow(imiss),nrow(filtered),sep=" "))
}

#lmiss<-read.table(file=paste(population,".lmiss", sep=""),
sep="\t", header=TRUE, skip = 0, na = "." )


populationlongname<-"pop name" #fill in meaningful name for pop
here
populationlongname

gdepth<-read.table(file=paste(population,".gdepth", sep=""),
sep="\t", header=TRUE, skip = 0, na = "-1", nrows=1000)

gdepth<-read.table(file=paste(population,".gdepth", sep=""),
sep="\t", header=TRUE, skip = 0, na = "-1")


gdepthminus1<-read.table(file=paste(population,".gdepth",
sep=""), sep="\t", header=TRUE, skip = 0, na = ".", nrows=1000)
#head(gdepthminus1)
ncol(gdepth)

?corrgram

gdepth_depth<-gdepth[,3:ncol(gdepth)]
numbersamples<-ncol(gdepth_depth)


sampleIDs<-colnames(gdepth_depth)


pseudoSampleIDs<-seq(from= 1,to=ncol(gdepth_depth)*100,by=100)
```

```
gdepth_long<-reshape(gdepth,
                     varying = sampleIDs,
                     idvar = "id", direction="long", sep = "")
colnames(gdepth_long)<-
c("chrom","pos","sample","coverage","consecutivenumber")


png(file=paste("Boxplot_CoveragePerSNPperSample",population,".png
", sep=""))
boxplot(gdepth_long$coverage~gdepth_long$sample,main="coverage
per biallelic SNP per sample missing GTs set to
NA",ylab="coverage")
dev.off()

png(file=paste("Boxplot_Log10CoveragePerSNPperSample",population,
".png", sep=""))
boxplot(log10(gdepth_long$coverage)~gdepth_long$sample,main="cove
rage per biallelic SNP per sample missing set to NA",ylab="log10
coverage")
dev.off()


gdepthminus1_long<-as.data.frame(reshape(gdepth,
                                         varying = sampleIDs,
                                         idvar = "id",
direction="long", sep = ""))

colnames(gdepthminus1_long)<-
c("chrom","pos","sample","coverage","consecutivenumber")


head(gdepthminus1_long)
boxplot(gdepthminus1_long$coverage~gdepthminus1_long$sample)
boxplot(log10(gdepthminus1_long$coverage)~gdepthminus1_long$sampl
e,main="coverage per biallelic SNPs per Sample missing -1")



gdepthminus1_depth<-gdepthminus1[,3:ncol(gdepthminus1)]
head(gdepthminus1_depth)

png(file=paste("CorrplotPCAclusteredGdepthwithoutminus1.",populat
ion,".png", sep=""))

corrgram(gdepth_depth, order=TRUE, lower.panel=panel.shade,
         upper.panel=panel.pts,
text.panel=panel.txt,cor.method="spearman",
         main="PCA ordered Spearman correlations of gdepth
without -1")
dev.off()

png(file=paste("CorrplotGdepthwithoutminus1.",population,".png",
sep=""))
corrgram(gdepth_depth, order=FALSE, lower.panel=panel.pie,
         upper.panel=panel.pts,
text.panel=panel.txt,cor.method="spearman",
         main="Spearman correlations of gdepth without -1
order=False")
```

```
dev.off()

cor.test(gdepth_depth,method="spearman",use="pairwise.complete.ob
s")
cor(gdepth_depth,method="spearman",use="pairwise.complete.obs")


diff<-apply(combn(ncol(gdepth_depth), 2), 2, function(x)
gdepth_depth[,x[1]] - gdepth_depth[,x[2]])

diff<-as.data.frame(apply(combn(ncol(gdepth_depth), 2), 2,
function(x) gdepth_depth[,x[1]] - gdepth_depth[,x[2]]))
diffminus1<-apply(combn(ncol(gdepthminus1_depth), 2), 2,
function(x) gdepth_depth[,x[1]] - gdepth_depth[,x[2]])
head(diff)

boxplot(diff)

hist
hist(diff, breaks=1000)
hist(log10(diff), breaks=1000)
hist(diff, breaks=1000)


hist(diff, breaks=20,xlim=c(-50,50))
hist(diff, breaks=20,xlim=c(-100,100))
hist(diff, breaks=1000,xlim=c(-100,100))
summary(diff)


apply(combn(ncol(gdepth_depth), 2), 2, print)

?rle


apply(combn(ncol(d), 2), 2, function(x) d[,x[1]] - d[,x[2]])

table(gdepth_long$sample)

png(file=paste("3DScatterplot_CoveragePerSNPperSample",population
,".png", sep=""))

scatterplot3d(gdepth_long$consecutivenumber,gdepth_long$sample,lo
g10(gdepth_long$coverage), type="h",
              pch=16, highlight.3d=TRUE,main=paste("3D
Scatterplot ",population," 1000 SNPs"))
dev.off()

scatterplot3d(gdepth_long$consecutivenumber,gdepth_long$sample,gd
epth_long$coverage, type="h",
              pch=16, highlight.3d=TRUE,main="3D Scatterplot")


scatterplot3d(gdepth_long$consecutivenumber,gdepth_long$sample,gd
epth_long$coverage, zlim=c(0,50),type="h",
              pch=16, highlight.3d=TRUE,main="3D Scatterplot")

#?scatterplot3d
```

```
scatter3d(gdepth_long$consecutivenumber,gdepth_long$sample,gdepth
_long$coverage)


plot3d(gdepth_long$consecutivenumber,gdepth_long$sample,log10(gde
pth_long$coverage),type="h")
plot3d(gdepth_long$consecutivenumber,gdepth_long$sample,gdepth_lo
ng$coverage,type="h")



idepth<-read.table(file=paste(population,".idepth", sep=""),
sep="\t", header=TRUE, skip = 0, na = "." )
head(idepth)

summary(idepth)
hist(idepth$N_SITES, breaks=20,main=(paste(populationlongname,"#
loci with GT per individual", sep=" ")))
boxplot(idepth$N_SITES,main=(paste(populationlongname,"# loci
with GT per individual", sep=" ")))

plot(idepth$N_SITES,idepth$MEAN_DEPTH)

hist(idepth$MEAN_DEPTH,
breaks=20,main=(paste(populationlongname,"mean coverage per
genotyped locus per individual", sep=" ")))


ldepth<-read.table(file=paste(population,".ldepth", sep=""),
sep="\t", header=TRUE, skip = 0, na = "." )


ldepth.mean<-read.table(file=paste(population,".ldepth.mean",
sep=""), sep="\t", header=TRUE, skip = 0, na = "." )

t_gdpeth<-as.data.frame(head(t(gdepth)))


imiss<-read.table(file=paste(population,".imiss", sep=""),
sep="\t", header=TRUE, skip = 0, na = "." )

lmiss<-read.table(file=paste(population,".lmiss", sep=""),
sep="\t", header=TRUE, skip = 0, na = "." )


for (chr in 1:3) {
  assign(paste("data",chr,sep="."),read.table(paste(resultsDir,
"/genoCN_chr",chr,".CNVs.bed", sep="")))
  #ATTENTION to the "," before read.delim instead of <-
}


#GENOME STATS
#-----------------------------------------------------
```

```
genome<-
read.table(file="R://WORKDATA//BioinformatikPlattform//Schneider/
/GenomeforR.txt",sep="\t", header=TRUE)
head(genome)
summary(genome)
png(file="R://WORKDATA//BioinformatikPlattform//Schneider//Histog
ram_ContigLength.png")
hist(genome$length, main="contig size", breaks=100)
dev.off()

png(file="R://WORKDATA//BioinformatikPlattform//Schneider//Histog
ram_Log10ContigLength.png")
hist(log10(genome$length), main="contig size", breaks=100)
dev.off()



contigs<-read.table(file=("Contigs.csv"), sep=";", header=TRUE,
skip = 0, na = "." )

head(contigs)
str(contigs)

hist(contigs$length)
hist(log10(contigs$length),breaks=100)
summary(contigs$length)
summary(log10(contigs$length))
longcontigs<-contigs[ which(contigs$length >100000),]



#FLAGSTATS
#-----------------------------------------------------------

pool="343"
pool="337"
pool="all"

Pool337new_343new.summary.flagstats

flagstats<-read.table(file="Pool337new_343new.summary.flagstats",
header=TRUE, skip = 0, na = "." )

flagstats<-
read.table(file=paste("Pool",pool,".summary.flagstats", sep=""),
header=TRUE, skip = 0, na = "." )
head(flagstats)

#colnames(flagstats)
# [1]"IDs"                                  "QCpassedTotalReads"
"QCfailedTotalReads"                "secondaryReads"
# [5] "mappedReads"                         "pairedReads"
"properlyPairedReads"               "withItselfAndMmateMapped"
# [9] "singletons"
"withMateMappedToDiffChromosome"
"withMateMappedToDiffChromosomeMapQ5"
```

```
flagstats$percMapped<-
(flagstats$mappedReads/flagstats$QCpassedTotalReads)*100
flagstats$percProperlyPaired<-
(flagstats$properlyPairedReads/flagstats$QCpassedTotalReads)*100

png(file=paste("Hist_numberMappedReadsPool", pool,".png",
sep=''))
hist(flagstats$mappedReads,breaks=100, main=paste("# mapped reads
in pool", pool, sep=' '))
dev.off()

png(file=paste("Hist_LogNumberMappedReadsPool", pool,".png",
sep=''))
hist(log10(flagstats$mappedReads),breaks=100, main=paste("#
mapped reads in pool", pool, sep=' '))
dev.off()

png(file=paste("Hist_percentMappedReadsPool", pool,".png",
sep=''))
hist(flagstats$percMapped,breaks=20,
xlim=c(0,100),ylim=c(0,100),main=paste("% mapped reads in pool",
pool, sep=' '))
dev.off()
png(file=paste("Hist_percentProperlyPairedReadsPool",
pool,".png", sep=''))
hist(flagstats$percProperlyPaired, breaks=20,
xlim=c(0,100),ylim=c(0,100), main=paste("% properly paired reads
in pool", pool, sep=' '))
dev.off()


#============================================================
#Author Dolezal Marlies and Schneider Katharina
#Date 2017
#============================================================

setwd("/Users/Kathi/Desktop/26.9-/")
getwd()

#---------------------------
#find technical artefacts
#---------------------------

Concordance=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcft
oolsresults/unique/Disc/concordance.gdepth", header=TRUE)

Concordance$Contig=paste(Concordance$CHROM, Concordance$POS,
sep="_")

Concordance$no_calls=rowSums(Concordance == "-1")
Concordance$calls=rowSums(Concordance !="-1")

summaryfile=data.frame(Concordance$Contig, Concordance$no_calls,
Concordance$calls)
summaryfile

art=subset(summaryfile, Concordance.calls =="6")
art1=subset(Concordance,calls ==5)
```

```
art2=subset(Concordance, calls ==35)

#-------------------
#pairwise Fst
#-------------------

Fst_all=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftools
results/cutoff/common_set/fst_all.weir.fst", header=TRUE)
vcf=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsresu
lts/cutoff/common_set/keep_indfilter.recode.vcf",
skip=2301,sep="\t")
genind=vcfR2genind(vcf)

samples=scan(text="104 105   106   107   108   109   110   111   112
      113   114   115   116   117   119   12    120   121   122   123
      124   125   126   127   128   129   13    130   131   132   133
      134   135   136   137   138   139   14    140   141   15    152
      153   154   155   156   157   158   159   16    160   161   163
      164   166   167   168   169   17    170   171   172   173   175
      176   177   178   179   18    180   181   182   183   184   185
      186   187   188   189   19    190   191   192   193   194   195
      196   199   2     20    200   21    214   215   216   218   22
      220   225   226   229   23    230   231   232   233   235   237
      238   24    240   241   248   249   25    250   251   256   257
      258   259   26    260   262   263   264   265   268   269   27
      274   275   276   279   28    281   285   289   29    294   296
      297   299   30    300   301   302   304   305   308   309   31
      310   311   312   313   315   317   318   32    320   33    337
      34    343   35    355   356   359   36    362   363   364   365
      367   368   369   37    370   371   373   375   378   379   38
      385   387   388   39    392   393   394   395   396   397   398
      399   40    400   401   402   403   404   405   406   407   408
      409   41    410   411   412   413   414   415   416   417   42
      43    44    45    46    47    48    49    5     50    51    52
      53    54    55    57    58    59    60    61    62    63    64
      73    74    75    76    77    78    79    80    81    82    83
      86    87    89    9     90    91    94    95    96")

sample_IDs=as.data.frame(samples)
Pop=merge(popmap, sample_IDs, by.x="Ind", by.y="samples")
individuals=as.factor(Pop$Pop)

Fstpop<-pairwise.fst(genind, individuals,res.type=NULL)
print(Fstpop)


#correlation Fst and distance between populations
#-------------------------------------------------

Dis=read.table("/Volumes/Temp/Kathi/populationmaps/Distance_corr.
txt", header=TRUE)

Dis$km=(Dis$Dis/1000)
test=lm(Fst~km, data=Dis)
summary(test)

hist((Dis$Fst))
hist(log10(Dis$Fst),breaks=20)
```

```
plot(Dis$Dis, Dis$Fst)

#--------------
#PCA
#--------------

#install.packages("vcfR")
library(vcfR)
#install.packages("hierfstat")
library(hierfstat)
library(vcfR)
#install.packages("adegenet")
library(adegenet)
#install.packages("matrixStats")
library(matrixStats)

vcf=read.vcfR("/Users/Kathi/Desktop/26.9-
/cutoff/common_set/keep_indfilter.recode.vcf")
genlight_pca=vcfR2genlight(vcf)
Meta=read.table("/Volumes/Temp/Kathi/populationmaps/Metainformati
on", header=TRUE)

genlight_mat=data.matrix(genlight_pca)

Fst=read.table("/Users/Kathi/Desktop/26.9-
/cutoff/fst_all.weir.fst", header=TRUE)
hist(Fst$WEIR_AND_COCKERHAM_FST, breaks=100)
Fst$Contig = paste(Fst$CHROM, Fst$POS, sep="_")

#Loci with Fst > 0.4, 0.5, 0.6)
#------------------------------

Fst_04=subset(Fst,Fst$WEIR_AND_COCKERHAM_FST > 0.4)
Fst_05=subset(Fst,Fst$WEIR_AND_COCKERHAM_FST > 0.5)
Fst_06=subset(Fst,Fst$WEIR_AND_COCKERHAM_FST > 0.6)

write.table(Fst_04, file = "Set_loci04", sep = " ")
write.table(Fst_05, file = "Set_loci05", sep = " ")
write.table(Fst_06, file = "Set_loci06", sep = " ")

genlight_t=as.data.frame(t(genlight_mat))
genlight_t$Contig=rownames(genlight_t)
SNP_04=merge(Fst_04, genlight_t, by="Contig")
SNP_05=merge(Fst_05, genlight_t, by="Contig")
SNP_06=merge(Fst_06, genlight_t, by="Contig")

exclude=c("CHROM", "POS", "WEIR_AND_COCKERHAM_FST")
contig_04=SNP_04[ , !(names(SNP_04) %in% exclude)]
contig_05=SNP_05[ , !(names(SNP_05) %in% exclude)]
contig_06=SNP_06[ , !(names(SNP_06) %in% exclude)]

contig_04_t=t(contig_04)
colnames(contig_04_t) <- as.character(unlist(contig_04_t[1,]))
Fst_04_t=contig_04_t[-1, ]

contig_05_t=t(contig_05)
colnames(contig_05_t) <- as.character(unlist(contig_05_t[1,]))
Fst_05_t=contig_05_t[-1, ]
```

```
contig_06_t=t(contig_06)
colnames(contig_06_t) <- as.character(unlist(contig_06_t[1,]))
Fst_06_t=contig_06_t[-1, ]

Fst_04_df=as.data.frame(Fst_04_t)
Fst_05_df=as.data.frame(Fst_05_t)
Fst_06_df=as.data.frame(Fst_06_t)

Fst_04_mat=data.matrix(Fst_04_df)
Fst_05_mat=data.matrix(Fst_05_df)
Fst_06_mat=data.matrix(Fst_06_df)

#############################################################
#if no na.omit in PCA
# get NA-count for each locus
naCnt.locus <- colSums(is.na(genlight_mat))
naCnt.locus_04=colSums(is.na(Fst_04_mat))
# get NA-count for each individuum
naCnt.ind <- rowSums(is.na(genlight_mat))
naCnt.ind_04=rowSums(is.na(Fst_04_mat))
# keep only loci without missing data
genlight_mat <- genlight_mat[,naCnt.locus == 0]
Fst_04_mat=Fst_04_mat[,naCnt.locus_04 == 0]
#############################################################

# remove columns with 0 variance
#-----------------------------------
genlight_mat <- genlight_mat[,colVars(genlight_mat) != 0]
Fst_04_mat=Fst_04_mat[,colVars(Fst_04_mat) !=0]


pca <- prcomp(genlight_mat, retx=TRUE, center=TRUE, scale=TRUE)
PCA_04 = prcomp((na.omit(Fst_04_mat)), retx=TRUE, center=T,
scale.=T)
PCA_05=prcomp((na.omit(Fst_05_mat)), retx=TRUE, center=TRUE,
scale.=TRUE)
PCA_06=prcomp((na.omit(Fst_06_mat)), retx=TRUE, center=TRUE,
scale.=TRUE)

plot(pca$sdev^2/sum(pca$sdev^2),xlim=c(1,10), pch=19, xlab="PC",
ylab="explained variance", cex.lab=1.5, cex.axis=1.5)

plot(pca$x[,"PC1"], pca$x[,"PC2"],
     xlab=paste0("PC1 (",
round(pca$sdev[1]^2/sum(pca$sdev^2)*100, 1), "%)"),
     ylab=paste0("PC2 (",
round(pca$sdev[2]^2/sum(pca$sdev^2)*100, 1),
"%)"),col=c("turquoise","turquoise",
"turquoise","turquoise","turquoise","turquoise","purple","purple"
,

"purple","purple", "purple","purple", "purple","purple","purple",
"yellow","purple","purple", "purple","purple",

"brown","brown","brown","brown","brown","brown","red","brown","br
own","blue","blue",
```

```
"blue","blue","blue","blue","blue","blue","brown",

"blue","blue","purple","green","green","green","green","green","g
reen","green",

"hotpink","purple","hotpink","hotpink","hotpink","hotpink","hotpi
nk","red","red","red","brown","red","red","red",

"red","red","red","red","red","red","purple","yellow","yellow","y
ellow","yellow","yellow","yellow","yellow","yellow","yellow",

"darkorange","darkorange","darkorange","darkorange","darkorange",
"darkorange","darkorange","darkorange","darkorange","darkorange",

"black","green","darkorange","green","turquoise","purple",
"purple","purple","blue",

"brown","brown","brown","brown","blue","brown","brown","brown","b
lue","blue","blue","blue",

"turquoise","blue","blue","green","green","hotpink","green","gree
n","green","green","green","green",

"red","turquoise","hotpink","hotpink","hotpink","hotpink","hotpin
k","red","red","darkorange","darkorange","darkorange",

"turquoise","yellow","turquoise","purple","blue","yellow","green"
,"green","hotpink","hotpink",

"yellow",
"hotpink","hotpink","hotpink","hotpink","red","yellow","yellow","
darkorange","yellow","yellow","yellow","yellow","yellow",

"darkorange","darkorange","darkorange","darkorange","darkorange",
"turquoise","black","blue","turquoise","hotpink","hotpink","hotpi
nk",

"turquoise","red","red","red","red","red","yellow","yellow","turq
uoise","yellow","yellow","yellow","darkorange","black","turquoise
","turquoise",

"red","turquoise","brown","turquoise","hotpink","darkorange","dar
korange","darkorange","darkorange","black","black","black","purpl
e","black","black","black","black","black","black","black",

"turquoise","turquoise","turquoise","purple","turquoise","turquoi
se","turquoise","purple","purple",
"purple","turquoise","turquoise","purple","purple",
"purple","purple","purple", "purple","brown","brown",

"blue",
"brown","brown","brown","brown","brown","brown","brown","brown","
blue","blue","blue","blue","blue","blue","green","hotpink","hotpi
nk","hotpink","hotpink","hotpink","hotpink",

"red","red","red","red","red","yellow","yellow","blue","yellow","
yellow","darkorange","darkorange","darkorange"), pch=19)
```

```
#correlation Fst and squared loadings
#-------------------------------------

cf=read.vcfR("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsresult
s/cutoff/common_set/keep_indfilter.recode.vcf")

genlight_pca=vcfR2genlight(vcf)
genind=vcfR2genind(vcf)

Meta=read.table("/Volumes/Temp/Kathi/populationmaps/Metainformati
on", header=TRUE)

PCA_scale=glPca(genlight_pca, nf=10, scale=TRUE)
PCAplot_scale_=as.data.frame(PCA_scale$scores)
PCAplot_scale_$Ind=rownames(scores_scale)
PCAplot_scale=merge(PCAplot_scale_, Meta, by="Ind")


vcf_tab=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftools
results/cutoff/common_set/keep_indfilter.recode.vcf",
skip=2301,sep="\t")
vcf=as.data.frame(vcf_tab)
head(vcf)
View(vcf)
VCF=c(1,2)
vcf_final=vcf[,VCF]
head(vcf_final)

Fst=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsresu
lts/cutoff/common_set/fst_all.weir.fst",header=TRUE)
#Fst_log=-log10(Fst$WEIR_AND_COCKERHAM_FST)
Fst_df=as.data.frame(Fst)
#Fst_log_df=as.data.frame(Fst_log)
Fst_df$Pos=rownames(Fst_df)
#Fst_log_df$Pos=rownames(Fst_log_df)

loadings=PCA_scale$loadings
loadings_df=as.data.frame(loadings)
loadings_sq=(loadings)^2
loadings_sq_df=as.data.frame(loadings_sq)
loadings_sq_df$Pos=rownames(loadings_sq_df)
load_fst=merge(loadings_sq_df, Fst_df, by="Pos")
parOld <- par(mar=c(5, 4, 1, 5))
plot(load_fst$WEIR_AND_COCKERHAM_FST~load_fst$Pos, xlab="Locus",
ylab="Fst", col=c("blue"), xlim=c(1,4000), ylim=c(0,7))
par(new=TRUE)
plot(load_fst$Pos, load_fst$Axis1, axes=FALSE, col="red",
xlab="Locus", ylab="")
axis(side=4, col="red", col.axis="red")
mtext("Loadings^2", side=4, line=3, col="red")


#loci with top 10% Fst and squared loadings
#-------------------------------------------

head(Fst_all)
head(loadings_sq)
```

```
loadings_sq_df=as.data.frame(loadings_sq)
head(vcf_final)

vcf_final$Pos=rownames(vcf_final)
Fst_all$Pos=rownames(Fst_all)
loadings_sq_df$Pos=rownames(loadings_sq_df)

Fst_final=merge(Fst_all, vcf_final, by ="Pos")
loadings_final=merge(loadings_sq_df, vcf_final, by="Pos")

Fst_order=Fst_final[order(-Fst_final$WEIR_AND_COCKERHAM_FST),]
loadings_order=loadings_final[order(-
loadings_final$loadings_sq),]

Fst_rank=Fst_final[rank(Fst_final$WEIR_AND_COCKERHAM_FST),]
loadings_rank=loadings_final[rank(loadings_final$loadings_sq),]

Fst_sel=c(1:396)
load_sel=c(1:396)

Fst_top=Fst_order[Fst_sel,]
load_top=loadings_order[load_sel,]

Fst_load_top=merge(Fst_top, load_top, by ="Pos")
cor.test(Fst_load_top$loadings_sq,
Fst_load_top$WEIR_AND_COCKERHAM_FST, method = "spearman")

Fst_rank_top=Fst_rank[Fst_sel,]
loadings_rank_top=loadings_rank[load_sel,]
Rank=merge(Fst_rank_top, loadings_rank_top, by="Pos")
cor.test(Rank$loadings_sq, Rank$WEIR_AND_COCKERHAM_FST, method =
"spearman")

Rank_all=merge(Fst_rank, loadings_rank, by ="Pos")
cor.test(Rank_all$WEIR_AND_COCKERHAM_FST, Rank_all$loadings_sq,
method="spearman")

loadingplot(PCA)
loadingplot(PCA_scale)
load_log=-log10(loadings)
log_load$Loci=rownames(loadings)

Fst_log=-log10(Fst_all)

vcf_final$Loci=rownames(vcf)
Pos=merge(vcf_final, loadings, by="Loci")
Fst_all$Loci=rownames(Fst_all)
Fst=merge(Pos,Fst_all, by="Loci")
head(Fst)
View(Fst)
Rel=c(1,2,3,4,16)
Fst_Axis1=Fst[,Rel]
View(Fst_Axis1)

plot(Fst_Axis1$Axis1~Fst_Axis1$WEIR_AND_COCKERHAM_FST)
hist(Fst_Axis1$Axis1)
hist(Fst_Axis1$WEIR_AND_COCKERHAM_FST)
```

```
cor.test(Fst_Axis1$Axis1, Fst_Axis1$WEIR_AND_COCKERHAM_FST,
method = "spearman")

loadings_sq=(Fst_Axis1$Axis1)^2
head(Fst_Axis1$Axis1)
head(loadings_sq)
cor.test(loadings_sq, Fst_Axis1$WEIR_AND_COCKERHAM_FST,
method="spearman")
plot(loadings_sq~Fst_Axis1$WEIR_AND_COCKERHAM_FST)
loadingplot(PCA_scale)
loadingplot(PCA)

#-------------------------
#genetic variation
#-------------------------

badr=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/badr.sites.pi", header=TRUE)
blai=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/blai.sites.pi", header=TRUE)
clau=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/clau.sites.pi", header=TRUE)
fort=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/fort.sites.pi", header=TRUE)
laut=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/laut.sites.pi", header=TRUE)
litt=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/litt.sites.pi", header=TRUE)
vieb=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/vieb.sites.pi", header=TRUE)
wulm=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/wulm.sites.pi", header=TRUE)
czra=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/czra.sites.pi", header=TRUE)
czrb=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/czrb.sites.pi", header=TRUE)
czrc=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/Site_pi/czrc.sites.pi", header=TRUE)

summary(badr$PI)
summary(blai)
summary(clau)
summary(fort)
summary(laut)
summary(litt)
summary(vieb)
summary(wulm)
summary(czra)
summary(czrb)
summary(czrc)

#----------------------------
# Inbreeding coefficient
#-------------------------

#install.packages(lme4)
library(lme4)
install.packages("lmerTest")
```

```
library(lmerTest)
#install.packages("lsmeans")
library(lsmeans)

het=read.table("/Volumes/Tem/Kathi/marlies/SNPcalls/vcftools/cuto
ff/co50.het", header=TRUE)

# inbreeding per populations
-----------------------------
  inb_Pop =lm(F~Pop, data=het)
summary(inb_Pop)
plot(inb_Pop, which=1)
plot(inb_Pop, which=2)

Meta_res=read.table("/Volumes/Temp/Kathi/populationmaps/Metainfor
mation_residual", header=TRUE)
inbreeding=merge(het, Meta_res, by="Ind")

inb_Pop_res= lm(F~Pop, data=inbreeding)
plot(inb_Pop_res, which=1)
plot(inb_Pop_res, which=2)
anova(inb_Pop_res)
summary(inb_Pop_res)
(ls_inb_pop_res<-lsmeans(inb_Pop_res,"Pop"))
cld(ls_inb_pop_res, Letters=letters)
lsmip(inb_Pop_res,~"Pop")

boxplot(inbreeding$F~inbreeding$Pop, ylab ="Inbreeding
Coefficient", xlab="Population", ylim=c(-0.1,
0.4),col=c("white","white", "grey", "white", "grey","white",
"grey", "grey","grey","grey"))

install.packages("lsmeans")
library(lsmeans)

# correlation inbreeding and contamination status
#-------------------------------------------------

linearModel_metal<-lm(F~Cont, data=inbreeding)
plot(linearModel_metal, which=1)
plot(linearModel_metal, which=2)

names(inbreeding)
inb_cont<-lmer(F~Cont+(1|Pop), data=inbreeding)
summary(inb_cont)
(ls_F_cont<-lsmeans(inb_cont,"Cont"))
Fred<-lmer(F~(1|Pop), data=inbreeding)
summary(Fred)
anova(inb_cont, Fred)
boxplot(inbreeding$F~inbreeding$Cont, xlab="Soil contamination
status", ylab="Inbreeding coefficent", ylim=c(-0.1, 0.4),
col=c("grey","white"))


# comparison inbreeding coefficients of populations clustering in
PCA
#-----------------------------------------------------------------
```

```
cluster1=read.table("/Volumes/Temp/Kathi/populationmaps/Cluster1"
,header=TRUE)
cluster2=read.table("/Volumes/Temp/Kathi/populationmaps/Cluster2"
, header=TRUE)

wf=merge(rel2, cluster1, by.x="INDV1", by.y="Ind")
CZ=merge(rel2,cluster2, by.x="INDV1", by.y="Ind")

hetwf=merge(het, cluster1, by="Ind")
hetCZ=merge(het,cluster2, by="Ind")


inb_full=lmer(F~Cont+(1|Pop), data=hetCZ)
inb_red=lmer(F~(1|Pop), data=hetCZ)
anova(inb_full, inb_red)
boxplot(hetCZ$F~hetCZ$Cont)

wfcont=lmer(F~Cont+(1|Pop), data=hetwf)
wfred=lmer(F~(1|Pop), data=hetwf)
anova(wfcont, wfred)
boxplot(hetwf$F~hetwf$Cont)

#influence location and Cd-concentration
#----------------------------------------

FCoordfull=lmer(F~N+E+A+Total_Cd+(1|Pop), data=inbreeding)
summary(FCoordfull)


#Test Heterogeneity of Cd Concentration.
#---------------------------------------

Het=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsresu
lts/cutoff/co50.het", header=TRUE)
Meta=read.table("/Volumes/Temp/Kathi/populationmaps/Metainformati
on", header=TRUE)

het=merge(Het, Meta, by="Ind")


badr=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_badr",
header=TRUE)
blai=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_blai",
header=TRUE)
clau=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_clau",
header=TRUE)
fort=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_fort",
header=TRUE)
laut=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_laut",
header=TRUE)
litt=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_litt",
header=TRUE)
vieb=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_vieb",
header=TRUE)
wulm=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_wulm",
header=TRUE)
czra=read.table("//Volumes/Temp/Kathi/populationmaps/Cd_czra",
header=TRUE)
```

```
czrc=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_czrc",
header=TRUE)
total=read.table("/Volumes/Temp/Kathi/populationmaps/Cd_total",
header=TRUE)

hist(badr$Total_Cd)
hist(blai$Total_Cd)
hist(clau$Total_Cd)
hist(fort$Total_Cd)
hist(laut$Total_Cd)
hist(litt$Total_Cd)
hist(vieb$Total_Cd)
hist(wulm$Total_Cd)
hist(czra$Total_Cd)
hist(czrc$Total_Cd)
hist(total$Total_Cd, breaks = 20)

boxplot(badr$Sample~badr$Total_Cd)
boxplot(total$Total_Cd~total$Population)

badr$Cd_sd=sd(badr$Total_Cd)
blai$Cd_sd=sd(blai$Total_Cd)
clau$Cd_sd=sd(clau$Total_Cd)
fort$Cd_sd=sd(fort$Total_Cd)
laut$Cd_sd=sd(laut$Total_Cd)
litt$Cd_sd=sd(litt$Total_Cd)
vieb$Cd_sd=sd(vieb$Total_Cd)
wulm$Cd_sd=sd(wulm$Total_Cd)
czra$Cd_sd=sd(czra$Total_Cd)
czrc$Cd_sd=sd(czrc$Total_Cd)

total_sd=rbind(badr, blai,clau, fort, laut, litt, vieb, wulm,
czra, czrc)

Inb=het[,c("Ind","Pop", "F")]
var=total_sd[c("Population", "Cd_sd")]
Fvar=merge(Inb, var, by.x = "Pop", by.y="Population" )
Fvar_uniqe=Fvar[!duplicated(Fvar), ]

library(lme4)
library(lmerTest)
infl=lmer(F~Cd_sd + (1|Pop), data=Fvar_uniqe)
summary(infl)
plot(infl, which=1)

#residuals
lm_Fvar=lm(F~Cd_sd, data= Fvar_uniqe)
plot(lm_Fvar, which=1)
plot(lm_Fvar, which=2)

Meta_res=read.table("/Volumes/Temp/Kathi/populationmaps/Metainfor
mation_residual", header =TRUE)
res=merge(Meta_res, Fvar_uniqe, by="Ind")
lm_res=lm(F~Cd_sd, data=res)
plot(lm_res, which=1)
plot(lm_res, which=2)

infl_res=lmer(F~Cd_sd + (1|Pop.x), data=res)
```

```
summary(infl_res)
plot(infl_res, which=1)

# check normal distribution of residuals
qqnorm(residuals(infl_res),main="QQ Plot for residuals")
qqline(residuals(infl_res),col="red")

# check normal distribution random effects
qqnorm(ranef(infl_res)[[1]][,1],main="QQ Plot for random
intercepts")
qqline(ranef(infl_res)[[1]][,1],col="red")

boxplot(Fvar_uniqe$Cd_sd)
hist(Fvar_uniqe$Cd_sd)

#popwise comparison
badr_blai=var.test(badr$Total_Cd, blai$Total_Cd)
badr_clau=var.test(badr$Total_Cd, clau$Total_Cd)
badr_fort=var.test(badr$Total_Cd,fort$Total_Cd)
badr_laut=var.test(badr$Total_Cd,laut$Total_Cd)
badr_litt=var.test(badr$Total_Cd,litt$Total_Cd)
badr_vieb=var.test(badr$Total_Cd,vieb$Total_Cd)
badr_wulm=var.test(badr$Total_Cd,wulm$Total_Cd)
badr_czra=var.test(badr$Total_Cd,czra$Total_Cd)
badr_czrc=var.test(badr$Total_Cd,czrc$Total_Cd)


#----------------------------------------
#kinship coefficients
#----------------------------------------

Meta=read.table("/Volumes/Temp/Kathi/populationmaps/Metainformati
on", header=TRUE)
relatedness2=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcf
toolsresults/cutoff/Unique_Set_final", header=TRUE)
badr=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.badr", header=TRUE)
blai=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.blai", header=TRUE)
clau=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.clau", header=TRUE)
czra=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.czra", header=TRUE)
czrc=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.czrc", header=TRUE)
fort=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.fort", header=TRUE)
laut=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.laut", header=TRUE)
litt=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.litt", header=TRUE)
vieb=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.vieb", header=TRUE)
wulm=read.table("/Volumes/Temp/Kathi/marlies/SNPcalls/vcftoolsres
ults/cutoff/Unique.wulm", header=TRUE)


relMeta=merge(relatedness2, Meta, by.x="INDV1", by.y="Ind")
```

```
install.packages("nortest")
library("nortest")
ad.test(relatedness2$RELATEDNESS_PHI)
kruskal.test(relatedness2$RELATEDNESS_PHI~relatedness2$Pop)

Rel2=lmer(RELATEDNESS_PHI~(1|Pop), data=relatedness2)
plot(Rel2, which=1)
qqnorm(residuals(Rel2))
qqline(residuals(Rel2))

arcrel2=lm(arc~Pop, data=relarc)
plot(arcrel2, which=1)
plot(arcrel2, which=2)

relsq=lm(sqrt~Pop, data=relsq)
plot(relsq, which=1)
plot(relsq, which=2)

relarcsq=lm(arc_sqrt~Pop, data=relarcsq)
plot(relarcsq, which=1)
plot(relarcsq, which=2)

badr_rel=badr$RELATEDNESS_PHI
blai_rel= i$RELATEDNESS_PHI
clau_rel=clau$RELATEDNESS_PHI
fort_rel=fort$RELATEDNESS_PHI
laut_rel=laut$RELATEDNESS_PHI
litt_rel=litt$RELATEDNESS_PHI
vieb_rel=vieb$RELATEDNESS_PHI
wulm_rel=wulm$RELATEDNESS_PHI
czra_rel=czra$RELATEDNESS_PHI
czrc_rel=czrc$RELATEDNESS_PHI

hist(badr_rel)
hist(blai_rel)
hist(clau_rel)
hist(fort_rel)
hist(laut_rel)
hist(litt_rel)
hist(vieb_rel)
hist(wulm_rel)
hist(czra_rel)
hist(czrc_rel)

#pop=c(badr_rel, blai_rel, clau_rel, fort_rel, laut_rel,
litt_rel, vieb_rel, wulm_rel, czra_rel, czrc_rel)
pop=assign(c("blai"))
get(pop)

for (i in pop)
{
  print(i)
  print(str(i))
  print(wilcox.test(badr$RELATEDNESS_PHI,get(i)$RELATEDNESS_PHI))
}

pvals<-c(0.05,0.0001,0.000056)
```

```
pvalues<-c( 0.05029, 0.00003811, 0.2271, 0.003255, 0.09753,
0.000000000001196, 0.06705, 0.003182, 0.5572, 0.1515, 0.4421,
0.7224, 0.8084, 0.00000006472, 0.0004022, 0.1239, 0.06431,
0.7189, 0.04242, 0.1317, 0.000001059, 0.00000008276, 0.4045,
0.004294, 0.8, 0.3146, 0.1113, 0.04477,0.677,0.1199, 0.5802,
0.00000003506, 0.000007903, 0.217, 0.009812, 0.00000000972,
0.0008566, 0.06478, 0.1251, 2.2e-16, 0.0003341, 0.000000003963,
0.000008249, 0.336, 0.006376 )
p.adjust(pvalues,method="holm")

p.adjust(pvals, method="holm")
```