Revista Brasileira de Zootecnia

pedSimulate – An R package for simulating pedigree, genetic merit, phenotype, and genotype data

Mohammad Ali Nilforooshan[1]* (iD)

[1] Livestock Improvement Corporation, Newstead, New Zealand.

How to cite: Nilforooshan, M. A. 2022.
pedSimulate – An R package for simulating
pedigree, genetic merit, phenotype, and
genotype data. Revista Brasileira de Zootecnia
51:e20210131.
https://doi.org/10.37496/rbz5120210131

ABSTRACT - This study aimed to introduce R package pedSimulate, which was built to simulate pedigree, genetic merit, phenotype, and genotype data. These are amongst the most important data types that animal breeders and quantitative geneticists deal with. Twenty pedigrees with ten generations were simulated applying different combinations of three parameters: genetic variance (10 vs. 20), proportion of males selected (10 vs. 20%), and the pattern for selecting females (random, positively, or negatively based on own phenotype or parent average). Males were selected positively based on parent average. Consequently, assortative mating was applied to the pedigrees in which females were positively selected based on their own phenotype or parent average. Disassortative mating was applied to the pedigrees in which females were selected negatively based on phenotype or parent averages. Genetic gain and response to selection over generations were positive for all the pedigrees due to high selection intensity on males, mating each male with multiple females, and moderate to high heritability (0.25 and 0.40 for genetic variances 10 and 20, and the residual variance of 30). Genetic variance showed a slightly increasing trend over generations by assortative mating and lower selection intensity on males. Selection intensity on females was the same in all the pedigrees. This study provided examples of how R package pedSimulate can be adopted for pedigree, genetic merit, phenotype, and genotype data simulation in animal breeding studies. By using different functions and combining different parameters for their arguments, many scenarios can be simulated by R package pedSimulate.

Keywords: assortative, disassortative, random, selection, simulation

1. Introduction

When dealing with real-world problems, using real data is preferred. However, due to data ownership and regulations, researchers do not always have access to these data. On the other hand, testing specific hypotheses requires specific data, usually simulated. The reason for this is that, unlike real data, which is influenced by many effects, many of them uncontrolled and unknown, simulated data provide a controlled environment suitable for testing the hypothesis.

Pedigree and phenotype data are critical information in animal breeding and genetics. For decades, this information has been used to estimate genetic merit of animals using the BLUP methodology (Henderson et al., 1959). In the last decade, the livestock breeding industry has been revolutionized by the use of genomic information. New methods have incorporated pedigree, phenotype, and genotype information in genetic evaluations (e.g., Aguilar et al., 2010; Fernando et al., 2014).

There is a limited number of free and open-source software that simulate pedigree, phenotype, and genotype data. The existing softwares are designed and developed for different purposes and are not very adaptable for general users' needs. Consequently, each software has its specific functionalities

and, based on what the user needs to achieve, one is preferred over the other. Some of the software that are published or available in an official public repository are: GenoSim (Jorjani, 2009), written in Fortran90/95; PedigreeSim (Voorrips and Maliepaard, 2012), written in Java; and R packages SimRVPedigree (Nieuwoudt and Graham, 2020) and AlphaSimR (Gaynor et al., 2021).

R package pedSimulate (Nilforooshan, 2022a) is a convenient tool for simulating pedigree, genetic merit, phenotype, and genotype data. It provides various parameters (function arguments) for tuning the simulation, including the population size of the founder animals, additive genetic and residual variances, litter size, mortality rate, generation overlap for males and females, selection intensity for males and females, selection criteria for males and females, and allele frequency and mutation rate at each genetic marker.

The objective of this study was to introduce R package pedSimulate and its functionalities. Examples of random, different assortative, and disassortative matings, different selection patterns, and various selection intensities for males and females were presented, and the results of the different scenarios were compared. Functions for genotype simulation and tracing close matings (full-sibs, half-sibs, and parent-progeny) in the population were presented with small examples.

2. Material and Methods

2.1. Simulated data

R package pedSimulate (Nilforooshan, 2022a) was used in this study. The primary purpose of R package pedSimulate is simulating pedigree, (true) genetic merit, phenotype, and genotype data. It is equipped with seven functions. The function "simulatePed" is used for non-genotype data simulation. It starts with a base population (F0) with an even user-defined (UD) size. The F0 is equally divided into males and females. Users can define the number of generations to be simulated after F0. No premature mortality, selection, and non-random mating are imposed on this generation. Other UD parameters are the genetic and residual variances (V_A, V_E) and litter size. If no litter size is defined, it is by default equal to 1. The F0 population reproduces the F1 generation. Other UD/default parameters are imposed on F1 onwards (default values are used unless changed by the user, as described in the package user manual (Nilforooshan, 2022a)). Mortality is imposed in two stages, pre- and post-maturity. Obviously, in pre-maturity, the individual has no chance of reproduction. The post-mature mortality is imposed via generation overlap, which can be set to different numbers for different sexes. After imposing mortality (premature mortality and generation overlaps for males and females), available males and females (selection candidates) undergo different patterns (random and non-random) and rates of selection. Then, the selected males and females are mated either in the same order that they have been selected or in a different order (e.g., random). Different combinations of selection and mating patterns shape various forms of random, assortative, and disassortative mating.

The simulated pedigree has nine columns: Animal ID, Sire ID, Dam ID, Sex, Generation, Parent Average (PA), Mendelian Sampling (MS), Environmental plus residual effects (E), and Phenotype (P). The PA of an individual is the average of its parents' true genetic merit, and the true genetic merit (TBV) of an individual equals PA + MS. As F0 individuals have no parent information, their PA is considered zero. For F0, the MS term is drawn from an $N(0, IV_A)$ distribution rather than an $N(0, \frac{1}{2}IV_A)$ distribution due to $V_{PA} = 0$. The MS variance is half the additive genetic variance in the base population ($V_{MS} = \frac{1}{2}V_A$; Bijma and Rutten, 2002). The E term is drawn from an $N(0, IV_E)$ distribution, in which V_E is assumed to be constant across generations. The phenotype (P) of an individual is calculated in two steps: P = PA + MS + E and P = P + μ, in which μ = −2Min(P). Users can rebase and rescale phenotypes to the desired mean ($\mu_{\acute{p}}$) and standard deviation ($\sigma_{\acute{p}}$) using the conversion formula $\acute{P} = ((P - \mu_P)\sigma_{\acute{p}}/\sigma_P) + \mu_{\acute{p}}$.

As the simulatePed function does not include genotype information, the MS term is simply sampled from a normal distribution. A better practice of determining MS, rather than simulating it, is calculating it via simulated QTL and linkage disequilibrium (e.g., Jorjani et al., 1997). This way, the MS term is a direct function of the sampled alleles in the gametic phase and their effect.

Five selection (ordering) and mating patterns [random, positively or negatively based on P (P, –P), positively or negatively based on PA (PA, –PA)] are considered, which can differ or be the same between males and females and between selection and mating. The PA and –PA options are considered for simulating selection in favor or against a trait early in life, before own or progeny phenotypes are observed.

Twenty pedigrees were simulated (Nilforooshan, 2022b) with different combinations of V_A (10 and 20), proportion of males selected (10% and 20%), and selection patterns of females (random, P, –P, PA, and –PA) (Table 1). Other parameters were the same across the pedigrees. No inputs were provided to the arguments defining the mating order of females and males ("f.order" and "m.order"). Consequently, by default, mating order was considered the same as the selection order. For each pedigree, nine generations were simulated, followed by a simulated F0 with a size of 100 individuals, litter size = 2, V_E = 30, (premature) mortality rate = 0 (default), generation overlap of 1, 80% of females were selected, and males were selected (10% or 20%) and ordered for mating based on PA.

Table 1 - Differences in the parameters used to simulate the pedigrees (PED1-20)

fsel	V_A = 10		V_A = 20	
	m% = 10	*m%* = 20	*m%* = 10	*m%* = 20
R	PED1	PED2	PED3	PED4
P	PED5	PED6	PED7	PED8
–P	PED9	PED10	PED11	PED12
PA	PED13	PED14	PED15	PED16
–PA	PED17	PED18	PED19	PED20

fsel - selection pattern for females; V_A - additive genetic variance; *m%* - proportion of males selected; R - random; P - positively based on phenotype; –P - negatively based on phenotype; PA - positively based on parent average; –PA - negatively based on parent average.

The following R code was used to simulate pedigrees. The simulated pedigrees and the code to analyze the data are available in a public data repository (Nilforooshan, 2022b).

```
library(pedSimulate)
l = 1
for(i in c("R","P","-P","PA","-PA"))
{
   for(j in c(10,20))
   {
      for(k in c(0.1,0.2))
      {
         ped = simulatePed(F0size=100, Va0=j, Ve=30, littersize=2, ngen=9,
                           overlap.s=1, overlap.d=1, f.rate=0.8, m.rate=k,
                           fsel=i, msel="PA")
         write.csv(ped, file=paste0("PED",l,".csv"), row.names=F, quote=F)
         l = l+1
      }
   }
}
```

For a more realistic situation, the user may consider setting some pedigree and phenotype data to missing.

2.2. Analyses

The simulated pedigrees were analyzed for the trends of genetic gain, response to selection, and genetic variance. The variance of the true genetic merits was considered as an indicator of the genetic variance. It was calculated for different generations and pedigrees. The average (phenotypic) response

to selection, average genetic gain, and slope of the trend in genetic variance over generations were calculated starting from F1 (i.e., response to selection from F1 to F2) because in all the pedigrees, the F1 generation resulted from random mating and no selection in F0.

2.3. Package functions

2.3.1. simulatePed

This function was described in subsection 2.1. Should reproducible simulations be needed, this function contains the optional argument "seed", which receives a numeric value as input.

2.3.2. appendPed

Researchers might be interested in simulating future generations of an existing pedigree. Function "appendPed" is similar to function "simulatePed" with the difference that it starts from an existing pedigree. Simulation strategies are limitless, and no single software can accommodate all the possibilities. Suppose a researcher is interested in simulating selection and mating patterns not covered by the package, the simulation can be done generation by generation using function "appendPed" and applying the selection and mating patterns of interest. For example, the following R code reads the first pedigree (out of the 20 simulated pedigrees) and appends a generation to it with all the default options (litter size of 1, no premature mortality, no generation overlap, no selection, and random mating).

```
ped1 = appendPed(read.csv("PED1.csv", header=TRUE), Va0=10, Ve=30, ngen=1)
# ped has 5592 rows.
# Va0 = 10
# Ve = 30
# littersize = 1
# ngen = 1
# mort.rate = 0
# overlap.s = 0
# overlap.d = 0
# f.rate = 1
# m.rate = 1
# fsel = R
# msel = R
# f.order = R
# m.order = R
# Started with a pedigree of 5592 individuals and maximum generation number 9
# Simulated generation 10 with 826 individuals.
```

To simulate a pedigree with selection and mating patterns not provided by the package, such as selection and mating based on the estimated breeding value, the function "appendPed" should be used instead of function "simulatePed" by following these steps:

1. Start with an existing pedigree in the same format as a pedigree object created by the function "simulatePed" or "appendPed".

2. Calculate TBV = PA + MS.

3. Estimate EBV using BLUP (no repeated records).

4. Replace values in the PA column with the EBV values estimated in the previous step.

5. Replace values in the MS column with TBV – EBV.

6. Simulate a generation with selection based on PA.

7. Continue the steps above for as many generations as needed.

Should reproducible simulations be needed, this function contains the optional argument "seed", which receives a numeric value as input.

2.3.3. simulateGen

This function simulates diploid and bi-allelic marker genotypes for a given pedigree. Allele frequencies and mutation rates at different marker loci are considered (default "mut.rate = 0" for no mutation). The function assumes that parents appear before progeny in the pedigree. For animals with both parents unknown, a genotype is simulated based on the UD allele frequencies. For animals with only one parent known, the following steps are taken: first, a gamete is simulated for the unknown parent; second, a gamete is produced from the genotype of the known parent; lastly, the gametes are combined into the genotype of the progeny. Mutation rate can be different from one marker to another, and it works by randomly changing an allele to its alternative. No linkage disequilibrium is modelled between markers. Should reproducible simulations be needed, this function contains the optional argument "seed", which receives a numeric value as input.

The following example simulates six marker genotypes with allele frequencies from 0.1 to 0.6 by 0.1 and no mutation for a pedigree of eight individuals. The first three columns of the pedigree (animal, sire, and dam, with no restriction on the column names) are used, and missing parents are coded 0.

```
ped = data.frame(animal=1:8, sire=c(0,0,1,0,3,3,6,6), dam=c(0,0,2,2,4,4,5,7))
(gen = simulateGen(ped, AF=seq(0.1,0.6,0.1)))
# No mutation was simulated.
#       [,1] [,2] [,3] [,4] [,5] [,6]
# [1,]    0    0    1    1    1    1
# [2,]    0    1    2    1    1    2
# [3,]    0    0    1    0    1    2
# [4,]    0    0    1    1    2    2
# [5,]    0    0    1    0    2    2
# [6,]    0    0    1    0    1    2
# [7,]    0    0    0    0    2    2
# [8,]    0    0    1    0    2    2
```

2.3.4. appendGen

This function simulates genotypes for an appended pedigree to an existing pedigree with genotypes. For example, consider the sample pedigree with eight individuals and their genotypes from subsection 2.3.3. For an appended pedigree of individuals 9 and 10, in which 9 has both parents unknown, and 10 has 9 and 7 as sire and dam, genotypes are simulated for individuals 9-10 and appended to the genotypes of individuals 1-8.

```
ped2 = data.frame(animal=9:10, sire=c(0,9), dam=c(0,7))
appendGen(rbind(ped, ped2), M=gen, AF=seq(0.1,0.6,0.1))
# No mutation was simulated.
#        [,1] [,2] [,3] [,4] [,5] [,6]
#  [1,]    0    0    1    1    1    1
#  [2,]    0    1    2    1    1    2
#  [3,]    0    0    1    0    1    2
#  [4,]    0    0    1    1    2    2
#  [5,]    0    0    1    0    2    2
#  [6,]    0    0    1    0    1    2
#  [7,]    0    0    0    0    2    2
#  [8,]    0    0    1    0    2    2
#  [9,]    0    0    0    2    2    0
# [10,]    0    0    0    1    2    1
```

Should reproducible simulations be needed, this function contains the optional argument "seed", which receives a numeric value as input.

Depending on the trait and the assumptions, users might need to simulate marker effects differently regarding the number of markers with major effects, the proportion of the phenotypic variance explained by the markers, and the distribution of marker effects. The package does not simulate marker effects. Where genomic selection is simulated, the user needs to assign (allele substitution) effects to the simulated markers. For genomic selection, generations undergoing genomic selection need to be simulated one at a time followed by selecting candidates based on their marker breeding value. Here, a genomic selection scenario is briefly explained.

1. Simulate the first generation(s), in which genomic selection is not practiced, using "simulatedPed" function. Any selection (if any) in these generations would be translated to genomic selection after simulating their genotypes and marker effects (likely to be a random selection).

2. Simulate genotypes for the pedigree simulated in the previous step, using "simulatedGen" function.

3. Simulate marker effects.

4. Calculate marker breeding values (G) given genotypes and marker effects.

5. If markers do not completely explain the genetic variance, simulate residual polygenic effect (Δ). Draw Δ values from a $N(0, A(V_A – V_G))$ distribution, in which A is the numerator relationship matrix, and $V_P = V_{PA} + V_{MS} + V_E = V_A + V_E = V_G + V_\Delta + V_E$.

6. Replace values in PA and MS columns with G and Δ values, respectively (the definitions of PA and MS columns are changed).

7. Re-calculate column P (P = PA + MS + E).

8. Simulate a new generation using "appendPed" function followed by genomic selection on the selection candidates (selection on PA (containing G)).

9. Simulate new genotypes for the new generation, using "appendGen" function.

10. Do steps 4-7, only for the new generation.

11. Do steps 8-10 for as many generations as desired.

12. Add μ (e.g., –2Min(P)) to column P.

2.3.5. Functions for finding close matings

Given a pedigree data frame with three columns for animal, sire, and dam IDs, functions "fs_mate_finder", "hs_mate_finder", and "pp_mate_finder" report full-sib, half-sib, and parent-progeny matings in the pedigree. Considering the sample pedigree with eight individuals from subsection 2.3.3, the examples are:

```
fs_mate_finder(ped)
#   SIRE DAM
# 2    6    5
hs_mate_finder(ped)
#   SIRE DAM
# 1    3    4
pp_mate_finder(ped)
#   SIRE DAM
# 4    6    7
```

3. Results

The simulated pedigrees had an average size of 5,124.4 individuals. Pedigrees 11 and 2 had the largest (5,718) and the smallest (4,456) sizes, respectively. Except for the parents of F0, no information was missing.

Pedigrees 15 and 18 showed the highest and the lowest slopes of genetic trend, respectively (Figure 1). Those pedigrees differed in negatively and positively selecting and ordering (for mating) females based on PA, genetic variance, and proportion of males selected. Assortative mating, higher genetic variance, and higher selection intensity on males (same for females across all the pedigrees) increased the slope of the genetic trend. Selection based on PA was more effective than selection based on P. All the pedigrees showed a positive genetic trend due to the high-intensity selection of males, positively based on PA.

The trends of genetic (true genetic merit) variance over generations are shown in Figure 2, in which the slope of the trend from F1 to F9 is presented within each pertaining plot (matings were random in F0 for all pedigrees). Genetic variances for the pedigrees simulated with V_A = 20 were almost twice as those for the pedigrees simulated with V_A = 10. No clear trend was observed in response to the proportion of males selected and selection patterns in females. However, a lower rate of male selection and assortative matings showed tendencies toward increasing the genetic variance.
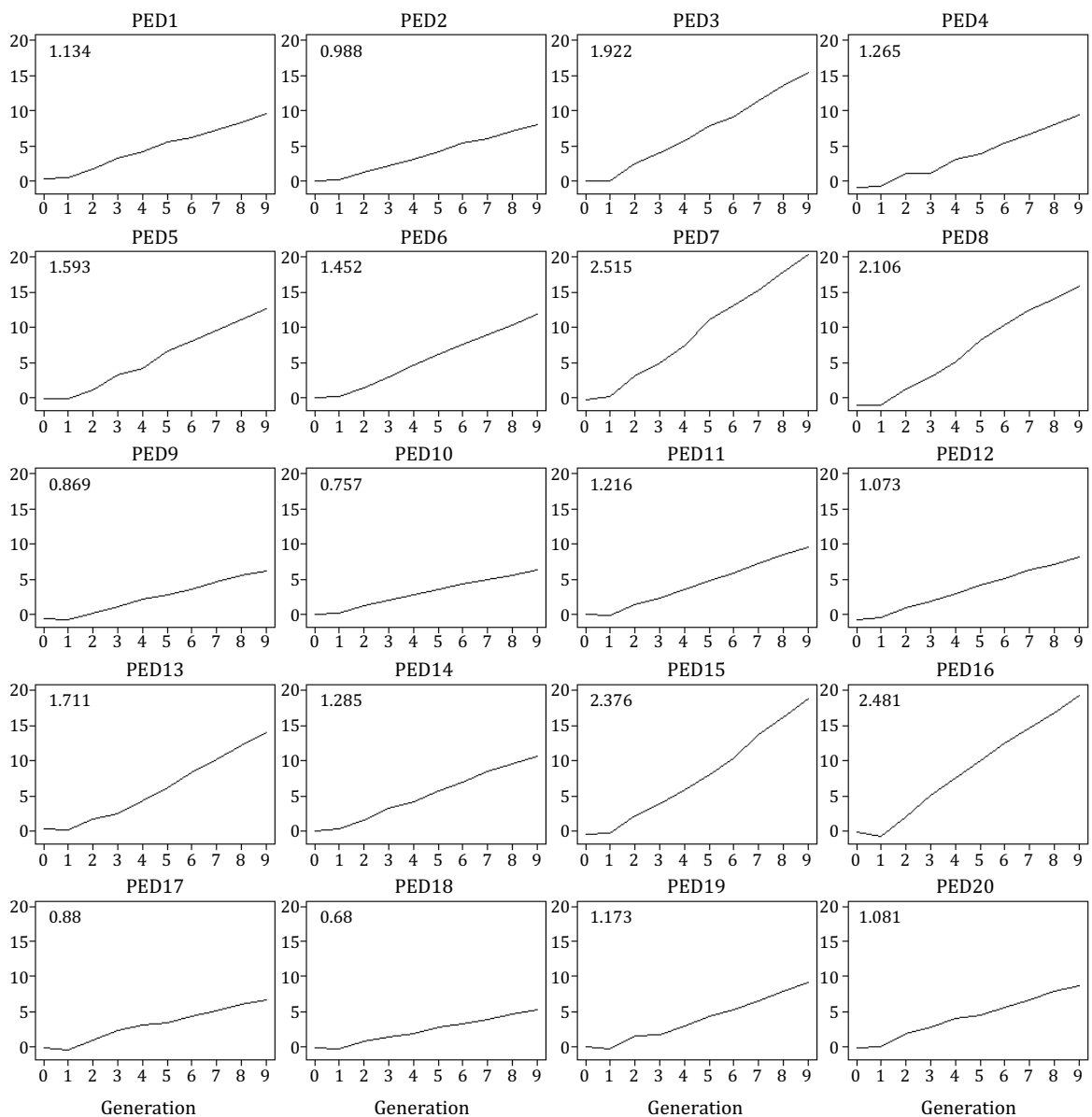
Average genetic gain from generation 1 to 9 is given at the top-left corner of each plot.

Figure 1 - Genetic trend over generations for the simulated pedigrees (PED1-20).
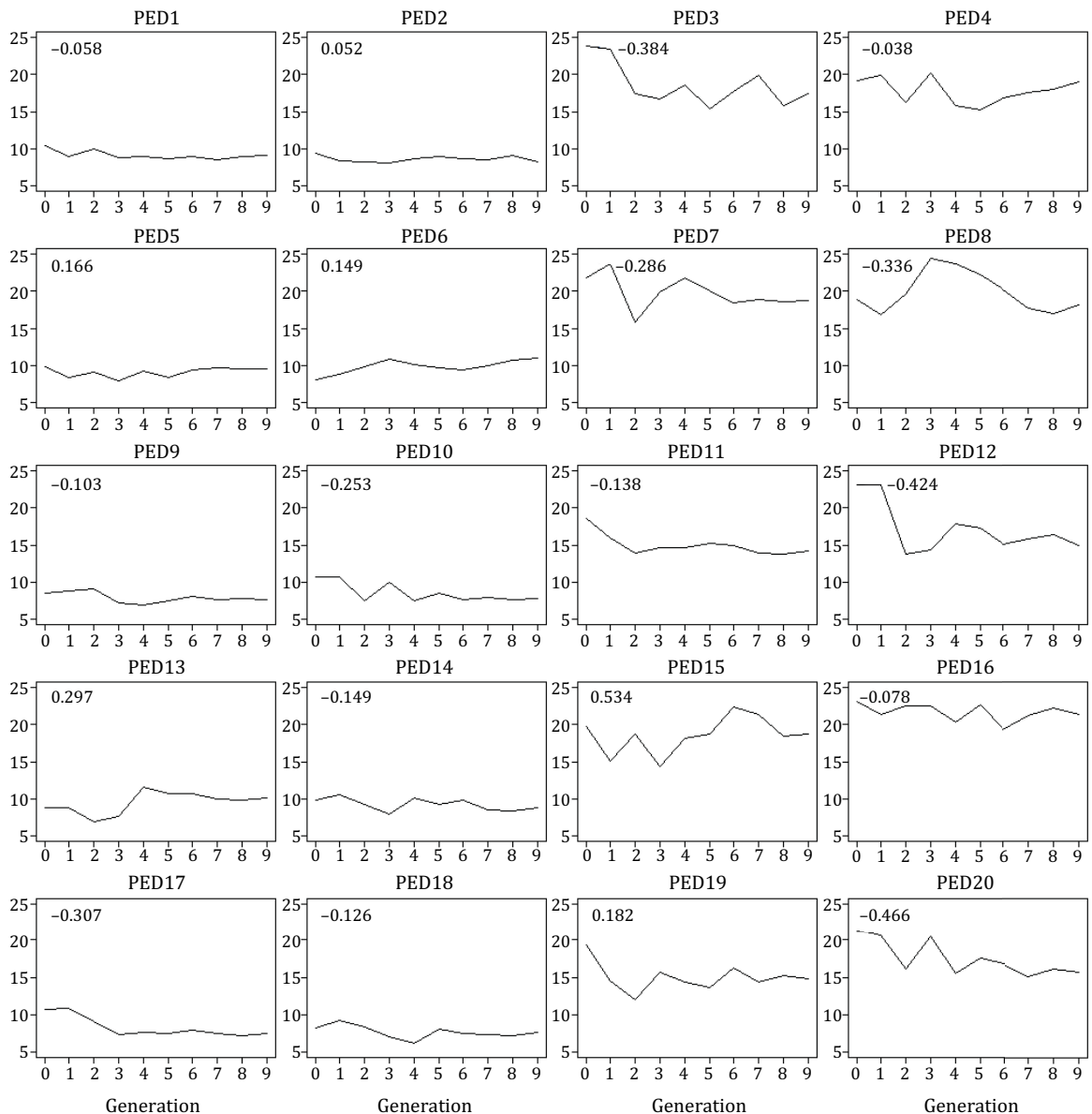
The slope of each trend from generation 1 to 9 is given at the top-left corner of each plot.

Figure 2 - Genetic (true breeding value) variance trends over generations for the simulated pedigrees (PED1-20).

Generally, the patterns for the average response to selection over generations (starting from F1) (Table 2) were similar to those observed in Figure 1. Higher genetic variance, higher selection intensity on males (same for females across all the pedigrees), and assortative mating increased the response to selection. Compared with assortative and random matings, response to selection was reduced by disassortative matings.

4. Discussion

In all the simulated pedigrees, males were selected and ordered (for mating) based on their PA. Selection intensities on males were high, and each male was mated to several (8 or 4, for 10% or 20% of males selected) females. These guaranteed genetic gain (Figure 1) and positive response to selection (Table 2) even with disassortative mating, in which females were selected and ordered for mating based on –P or –PA. Compared to selection on PA, selection on P is equivalent to selection on PA + MS + E. The MS term increases the accuracy of selection (higher genetic gain), and the E

Table 2 - Average response to selection over nine generations for the simulated pedigrees[1]

fsel	$V_A = 10$		$V_A = 20$	
	m% = 10	m% = 20	m% = 10	m% = 20
R	0.981	0.925	1.885	1.351
P	1.734	1.520	2.490	1.992
−P	0.750	0.753	1.149	1.001
PA	1.654	1.339	2.521	2.466
−PA	0.839	0.715	1.076	0.985

[1] The values correspond to the pedigree numbers in Table 1.
fsel - selection pattern for females; V_A - additive genetic variance; *m%* - proportion of males selected; R - random; P - positively based on phenotype; −P - negatively based on phenotype; PA - positively based on parent average; −PA - negatively based on parent average.

term reduces the accuracy of selection. Theoretically, as long as $V_{MS} < V_E$ (i.e., $V_A < 2V_E$), selection on PA results in more genetic gain than selection on P. Assortative mating increased the genetic gain and response to selection (Figure 1 and Table 2), especially with higher genetic variance and higher selection intensity on males (same for females across all the pedigrees).

In all the pedigrees, the differences between F0 and F1 were random and not due to any selection and mating strategy, because the F0 individuals randomly and equally contributed to F1. There were high fluctuations in the trends of genetic (true genetic merit) variance in the first few generations (Figure 2) due to the small number of parents in those generations.

Compared with the disassortative mating scenarios, the assortative mating scenarios showed an increasing trend in the genetic variance (Figure 2). This was in agreement with previous studies (Wright, 1921; Jorjani et al., 1997; Hayashi, 1998), in which assortative mating was suggested to increase heritability and additive genetic variance by increasing linkage disequilibrium.

5. Conclusions

There is a shortage of free and open-source software for simulating pedigree, phenotype, and genotype data. Most of the available simulation software are too specialized and not flexible or user-friendly. Although advanced, they may not deliver what most users might need, a general-purpose, user-friendly, and flexible software. The R package pedSimulate was developed to fill the shortage of free, open-source, and user-friendly software for simulating pedigree, genetic merit, phenotype, and genotype data. The package is produced in the R ecosystem, a popular programming language among students, scientists, and researchers. It provides several functions equipped with a range of arguments, thus providing the possibility to simulate many different scenarios through the combination of parameters. As an open-source package, researchers can modify the source code should their simulation have specific needs not covered by the package. R package pedSimulate is a user-friendly tool for general-purpose data simulations in animal breeding and genetics.

Conflict of Interest

The author declares no conflict of interest.

Author Contributions

Conceptualization: M.A. Nilforooshan. Data curation: M.A. Nilforooshan. Formal analysis: M.A. Nilforooshan. Investigation: M.A. Nilforooshan. Methodology: M.A. Nilforooshan. Resources: M.A. Nilforooshan. Software: M.A. Nilforooshan. Writing-original draft: M.A. Nilforooshan. Writing-review & editing: M.A. Nilforooshan.

Acknowledgments

References

Aguilar, I.; Misztal, I.; Johnson, D. L.; Legarra, A.; Tsuruta, S. and Lawlor, T. J. 2010. Hot topic: A unified approach to utilize phenotypic, full pedigree, and genomic information for genetic evaluation of Holstein final score. Journal of Dairy Science 93:743-752. https://doi.org/10.3168/jds.2009-2730

Bijma, P. and Rutten, M. 2002. Lecture notes for the SelAction workshop. Available at: <https://www.wur.nl/en/Research-Results/Chair-groups/Animal-Sciences/Animal-Breeding-and-Genomics-Group/Research/Software.htm>. Accessed on: June 30, 2021.

Fernando, R. L.; Dekkers, J. C. M. and Garrick, D. J. 2014. A class of Bayesian methods to combine large numbers of genotyped and non-genotyped animals for whole-genome analyses. Genetics Selection Evolution 46:50. https://doi.org/10.1186/1297-9686-46-50

Gaynor, R. C.; Gorjanc, G. and Hickey, J. M. 2021. AlphaSimR: an R package for breeding program simulations. G3 Genes|Genomes|Genetics 11:jkaa017. https://doi.org/10.1093/g3journal/jkaa017

Hayashi, T. 1998. Genetic variance under assortative mating in the infinitesimal model. Genes & Genetic Systems 73:397-405. https://doi.org/10.1266/ggs.73.397

Henderson, C. R.; Kempthorne, O.; Searle, S. R. and von Krosigk, C. M. 1959. The estimation of environmental and genetic trends from records subject to culling. Biometrics 15:192-218. https://doi.org/10.2307/2527669

Jorjani, H. 2009. A general genomics simulation program. Interbull Bulletin 40:202-206.

Jorjani, H.; Engström, G.; Strandberg, E. and Liljedahl, L.-E. 1997. Genetic studies of assortative mating—a simulation study. II. Assortative mating in unselected populations. Acta Agriculturae Scandinavica, Section A — Animal Science 47:74-81. https://doi.org/10.1080/09064709709362373

Nieuwoudt, C. and Graham, J. 2020. SimRVPedigree: Simulate Pedigrees Ascertained for a Rare Disease. version 0.4.4. Available at: <https://cran.r-project.org/package=SimRVPedigree>. Accessed on: May 1, 2022.

Nilforooshan, M. A. 2022a. pedSimulate: Pedigree, Genetic Merit, Phenotype, and Genotype Simulation. version 1.3.2. Available at: <https://cran.r-project.org/package=pedSimulate>. Accessed on: May 1, 2022.

Nilforooshan, M. A. 2022b. Twenty simulated pedigrees with different combinations of three parameters using R package pedSimulate. Mendeley Data, v2. https://doi.org/10.17632/c4pv8w8pmp.2

Voorrips, R. E. and Maliepaard, C. A. 2012. The simulation of meiosis in diploid and tetraploid organisms using various genetic models. BMC Bioinformatics 13:248. https://doi.org/10.1186/1471-2105-13-248

Wright, S. 1921. Systems of mating. III. Assortative mating based on somatic resemblance. Genetics 6:144-161.