



PedPro

MANUAL **ANALYSIS**

Contents

- About
- Quick Start
- Pedigree File
- Error checking
- Functions
- List of all options
- CPF
- Examples
- Troubleshooting

About

[\[Back to top\]](#)

PedPro is a program to handle pedigrees. It can check for errors, detect and break loops, remove uninformative individuals for linkage analysis, find obligatory carriers, identify clusters of individuals based on relationship and affection status, identify and remove isolated individuals, merge connected families into one, and calculate individual weight to adjust for correlation between family members in an association test.

Quick start

[\[Back to top\]](#)

Just upload a pedigree file without setting any options.

Pedigree File

[\[Back to top\]](#)

Overview

A Pedigree File for PedPro is a tab- or space-delimited text file. Lines starting with # or after a blank line will be omitted. Each line corresponds to one person; each column is a variable. By default, columns are separated by a single tab.

Column header

The first line is called the header line, from which PedPro identifies the contents of each column. PedPro tries to match only the first 7 characters of a header with the first 7 characters of a variable Symbol or one of the Synonyms in a case-insensitive fashion. Therefore, it is quite flexible in recognizing pre-defined variables. For example, the header of a "father" column could be Father, Fath, Fth, Dad, Fa, Pa, PaID; the "Pedigree" column could be named PedID, Pedigree, Family, PID, FID, PED, FAM, Kindred; "Individual" could be IndID, Individual, IID, IND, ID, Subject, Person; etc.

Below are the pre-defined variables:

Description	Type	ID	Symbol	Synonyms (7 characters or less, case-insensitive)
Pedigree_ID	STR	PID	PedID	pedid pid p_id ped family fid f_id fam kindred FamilyI
Individual_ID	STR	IID	IndID	indid iid i_id ind id subject PersonI IndivID
Unique_ID	STR	UID	UniqID	uniqid uniq uid u_id
Father_ID	STR	FTH	Father	father fath fth dad fa pa PaID FathID
Mother_ID	STR	MTH	Mother	mother moth mth mom mo ma MaID MothID
Downcoded_PID	STR	DNP	dPID	dpid dp
Downcoded_IID	STR	DNI	dIID	diid di
Downcoded_UID	STR	DNU	dUID	duid du
Downcoded_FTH	STR	DNF	dFTH	dfth dfa ddad
Downcoded_MTH	STR	DNM	dMTH	dmth dmo dmom
Population	STR	POP	Pop	popu pop
Monozygosity_Twin	STR	MZT	MzTwin	mztwin mz_twin mz_t mztw mzt mz twin

Genotype	STR	GTP	Geno	geno mutn gtp
Alternative_ID	STR	ALT	AltID	altid alti alt
Comment	STR	CMT	Comment	comment comm cmt
Details	STR	DET	Details	details det
Protected_Health_Info	STR	PHI	PHI	phi
Medical_Test_Reports	STR	MTR	MedRec	medrec
Genetic_Variants	STR	GVR	GVar	gvar
Sex	DBL	SEX	Sex	gender gend sex sx sex_
Affection_Status	DBL	AFF	Aff	affe aff af
Liability	DBL	LIA	Liab	liab lia li
Proband	DBL	PRB	FPTP	fptp proband prob prb Tgt
Age	DBL	AGE	Age	age ag
Year_Of_Birth	DBL	YOB	YoB	yob byr birth
Genotype_Numer	DBL	GTN	GTN	gtn
Genotype_Error	DBL	GTE	GTE	gte
Cluster_Number	DBL	CLT	Cluster	cluster cl
Inbreeding_Coefficient	DBL	INB	Inbr	inbr
Generation_number	DBL	GEN	Gen	gen
Descendants_MaxNo.Gen	DBL	MXG	GDes	gdes
Descendants_TotalNo.	DBL	DES	NDes	ndes
Allele_1	DBL	AL1	AL1	al1 a1 allele1
Allele_2	DBL	AL2	AL2	al2 a2 allele2
Individual_Weight	DBL	IWT	IndWt	indwt weight wt
Death	TRB	DTH	Death	death dead die vital_s

You can use the option `--XXX=yyy` to customize name of a pre-defined variable. Here XXX is the variable ID; yyy is the new variable name. For example, `--AFF=BrCa` tells the program to obtain affection status from the column "BrCa", which stands for breast cancer. Be careful, this may mask an existing variable. For example, if you do `--PID=Cluster --cl-out cl,id,fa,mo,sx,af --cl-aff`, the first column in the output will not be the new Cluster_ID but Pedigree_ID.

Besides predefined columns, PedPro can read BOADICEA-type of columns which have the following features:

- (1) Header indicates disease name; non-missing content is age of diagnosis; missing means not affected.
- (2) Disease features are in other columns.

The (`--boadicea-var`) option will setup the rules to read these columns. You may also need the (`--cvt`) option to convert content for some columns such as genotype, proband, and vital status. Please see the example below.

Column content

The option (`--cvt`) can flexibly convert column content to pre-defined codes. Argument of this option is the conversion instruction with the format of `variable1symbol:text_1/text_2=value1;text_3/text_4=value2;variable2symbol:...` For example, the default argument is

`sex:m/male/man/boy/46xy/47xyy/47xxy/48xxxy/49xxxxy=1,f/female/woman/girl/46xx/47xxx=2;aff:affected/af f/y/yes/a=2,unaffected/unaff/n/no/u=1`. This means in the SEX column, m, male, man, boy, 46xy, 47xyy, 47xxy, 48xxxy, 49xxxxy will be converted to 1, while f, female, woman, girl, 46xx, 47xxx will be converted to 2. It should be noted that `a=b,b=c` is safe, which means the program will not convert "a" to "c" by changing "a" to "b" then "b" to "c". This option will also affect output of column contents.

Codes for Monozygosity_Twins should be integers starting from 1. However, PedPro doesn't check whether the codes are successive and start from 1.

If your pedigree has a GTP column, optionally you can use the (`--gvoi`) option to populate the genotypes to the GVR column. Reversely, if your pedigree file has a GVR column, the genotypes will be populated to the GTP column. GVR can contain multiple variants. In this case, please use the (`--gvoi`) to choose which variant to population.

A Pedigree File should have at least three columns: Father_ID, Mother_ID, and either an Individual_ID (IID) or a Unique_ID (UID). IIDs are unique within each pedigree; while UIDs are unique in the whole file. If a Pedigree_ID (PID) column does not exist, the whole file is deemed a pedigree. If the IID column does not exist, IIDs will be the same as UIDs. If the UID column does not exist, a UID will be created as `PID::IID`. If a person's IID/UID is an empty string, PedPro will assign a name to the subject.

Lines

A pedigree file can have multiple lines for an individual, where subsequent input will replace the previous one.

This may cause a problem if the previous line has parental IDs while the subsequent line does not. The option (-a) will let the program keep the parental connection. This option is useful for merging connected families into one pedigree.

Error Checking

[\[Back to top\]](#)

PedPro detects problems and tries to correct them. Potential problems are listed below. PedPro reports errors for '#'s, warnings for '>'s, and none for '.'s. Solutions are listed in "[]", if there're any.

```
# Files missing Father_ID or Mother_ID or (IID or UID) [program stops]
# Lines with PID problems (PID is empty or 0 or a period) [skip the line and continue]
# Self-ancestors (A=>..=>A) [program stops]
# Same-parents (mom and dad are the same person) [founderize offsprings]
# Self-parent (one is oneself's father or mother) [founderize the persons]
# Ambiguous-gender (being a father of one person, and mother of another) [program stops]
# Impossible monozygosity twins (wrong code, different parent/sex, number of sibs !=2) [show error]
# Wrong liability class (<1) and wrong affection status (<0) [show error]
> Improbable Year-of-birth (mother <8 or >70, fathers <12 or >90 years old) [show warning]
> Wrong number of probands in a family (!=1) [show warning]
> Re-input of the same individual [update fa,mo,sex,etc.]
> Re-input of the same family [merge family members]
> Potential mistyping of IIDs in wrong letter case
> Questionable IIDs (all IIDs never show up as Father_IDs or Mother_IDs)
> Multiple clusters of individuals within a family [show warning; remove separated individuals]
> Single parent [create a dummy spouse with an IID <single-parent's IID>_01 for all offspring]
> Someone is a father but Sex is female, or is a mother but Sex is male [modify the Sex]
. Lines for a parent is missing [add a line]
```

It's always a good practice to list all pedigree members together. If the lines for different pedigrees intermingle with each other, PedPro will report a warning "Re-input family xxx". If intermingling is not expected, this may come from overlapping FIDs between recruitment institutes, and may lead to severe problems because these families are merged.

Functions

[\[Back to top\]](#)

Detect multiple clusters and remove isolated individuals

Some data may have multiple unconnected families under the same Pedigree_ID. This may be a sign of problems. PedPro can detect it and show a warning. Each cluster is assigned a unique Cluster_ID. If the --wr argument has a Cluster field, the output file will contain and is sorted by the Cluster_ID, which can be treated as a new Pedigree_ID.

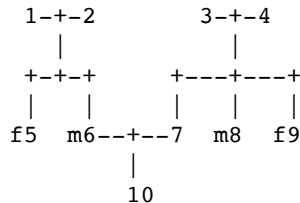
This feature can also be used to divide a big pedigree into sub-pedigrees, e.g., to identify high-risk sub-pedigrees for a disease, within which any two affected persons are connected by a sequence of affected 1-/2-/3-degree relatives. The corresponding command is "--cl-out cl,id,pa,ma,sx --cl-aff". Different from the "--wr cl,.." option, this method may assign an individual to multiple clusters, and so he/she may be printed in multiple lines. Again, the output is sorted by Cluster_IDs.

Isolated individuals who are not connected to any other persons in the pedigree, a special case of multiple clusters, will be removed in the output.

Break loops

For simplicity, in this program a loop is a consanguinity loop, while a circle is a marriage loop. For example, a circle is formed if two brothers married two sisters separately. They are legal in the real world, but could cause difficulties for some software such as MelaPRO and SLINK. PedPro can look for circles between two nuclear families within up to three generations.

For example, suppose we have a family shown below. A circle is formed if one of these pairs of individuals married: 8-5 / 3-2 / 8-2 / 6-9. PedPro can detect all of them, but not beyond 10's grandparents. The option `--br-circle` can be used to break all circles by founderizing a minimum number of individuals per circle. These individuals are selected by the following criteria sequentially: minimum number of sibs; unaffected; minimum number of affected first-degree relatives. If more than 1 individual fulfills these criteria, then PedPro randomly selects one of them. This procedure is repeated until no circle remains. The option `--br-loop` uses a similar approach to break loops.



(Parent on the left is father unless otherwise specified: f=female, m=male.)

Beware, a small loop (avuncular marriage) may also appears to be a circle, for which this program will report a warning for both a loop and a circle.

Remove uninformative individuals

The following persons are uninformative for linkage analysis: 1) unaffected ungenotyped founders with only 1 informative child; 2) unaffected ungenotyped individuals without any informative child. The `--rm-uninf` option can be used to remove them. Because this function may lead to separated individuals, `--rm-sep` is automatically activated along with `--rm-uninf`.

Calculate individual weight

The option (`--ind-wt`) will calculate individual weight for each genotyped individual to be used in an association test controlling for correlation among subjects. This option requires the Pedigree File to have the AFF and GTN column. Any non-zero integer in GTN means the person has genotype. Use "wt" as an Additional Output Field to write the weight to the output pedigree file.

Cluster affected individuals

If the pedigree is too big (such as that with thousands of individuals in more than 10 generations), it may prohibit analysis by some program. It may be useful to separate the pedigree into parts, or clusters. The option (`--cl-aff`) will create these clusters based on relationship and affection status. The goal is to find clusters of affected individuals within N degree of relatives (`--cl-dgr`).

Find obligatory carriers

The option (`--oc`) finds obligatory carriers from GTP and write `ObligatoryCarrier=yes` to Details.

List of all options

[\[Back to top\]](#)

For reading Pedigree File:

```

-d STR / -dSTR  Delimiters (multiple characters allowed; 1st character is also for output) {'\t'}
-s [B]         Treat successive delimiters as one {No}
--quoted=B     Allow quoted fields, quotation marks remain as content {No}
--comment=S    Lines starting with S are comments {#}
--skip-comment=B Skip comment lines (omit and move on) {Yes}
--keep-comment=B Keep comment lines (show and move on) {No}
--skip-blank=B Skip blank lines {No}
--read-blank=B Read blank lines if not skipped, {No}
--trim-lws=B   Skip leading whitespaces {No}
--trim-tef=B   Skip trailing empty fields {No}
--irg=B        Input is irregular: s=Y trim-lws=Y trim-tef=Y skip-blank=Y
--csv=B        Input is CSV: -d , quoted=Yes s=No trim-lws=No trim-tef=No
  
```

```

-Wno-single      Suppress warnings of single-parents (still show how to fix)
-Wno-gender      Suppress warnings of wrong-genders that are corrigible
-Wno-re-inp      Suppress warnings of re-input pedigrees or individuals
-Wno-subped      Suppress warnings of 2+ disconnected pedigrees per PID
-Wno-dupUID      Suppress warnings of Identical UIDs
-w              Suppress all warnings
--ma-age=I-I     Maternal age range for error detection, default is 8-70
--pa-age=I-I     Paternal age range for error detection, default is 12-90
--id-del=S       Set the delimiter between PID and IID for making UIDs. {:;}
--var-known Vs   Keep only Vs as pre-defined variables.
-a              Read pedigree data aggregately.
--XXX=yyyy       Convert variable names.
--cvt S          Convert variable contents.
--gvoi S         Genetic variant of interest is S (you don't need this option if there is only one variant in GVar)
--boadicea-var S BOADICEA-type event columns

```

For operations:

```

--keep-unaff     Keep unaffected sibs in doing --rm-uninf (below)
--rm-uninf       Remove uninformative individuals (requires AFF,GTN)
--br-circle      Break circles by founderizing 1+ persons per circle
--br-loop        Break loops by founderizing 1+ persons per loop
--ind-wt         Generate individual weights for association test (requires AFF,GTN)
--cl-dgr INT     Set degree of relatives for --cl-aff. Should be set before --cl-aff. {3}
--cl-aff         Cluster individuals who are affected
--oc             Find obligatory carriers and write ObligatoryCarrier=yes to Details

```

For writing outputs:

```

--wr-alt         Output alternative IDs
--prepend-pid    Prepend PedID_ to each IndID, FatherID, MotherID
--append-prb     Append a proband flag "[P]" at the end of an Individual ID

```

Comprehensive Pedigree Format

[\[Back to top\]](#)

Cosegregation analysis, risk prediction, and penetrance estimation are potential usages of a pedigree file. Joint analysis of pedigrees from multiple institutes is important for increasing analysis power. To facilitate sharing and re-using pedigree files, I have created a format called [Comprehensive Pedigree Format \(CPF\)](#). PedPro may help to convert your existing pedigree file to CPF.

Cosegregation analysis requires pedigree structure, sex, affection status for relevant disease(s), age, population, birth year, genotype and the first person within each pedigree tested positive for the variant of interest, and monozygotic twin status. Here, age is the diagnosis age of the associated disease(s), diagnosis age of other diseases that compete with or increase the risk of the associated disease(s), age of risk-reducing treatments, or age of the last follow-up, whichever comes first. Pedigree structure refers to biological (blood) relationships only. Make sure your existing pedigree file contains the above information. It should contain all relevant diseases and all age of diagnoses data instead of an affection status and a combined age. This will significantly improve the re-usability of the file (i.e., you can directly analyze an old pedigree file without modification even when you change the disease risk model or the gene of interest). Favorably, the file should also contain environment risk factors, polygenic risk scores, genotypes of other high-penetrance genes, the type of genetic testing, disease subtype and disease characteristics. These information maybe useful for future analyses.

Examples

[\[Back to top\]](#)

Example 1: reading a BOADICEA-type Pedigree File

Below is an example file (do not copy and paste; the original BOADICEA file uses a tab to separate columns and a space to represent missing value; I have replaced the tab with spaces to make it look good on the web):

Name	Tgt	IndivID	FathID	MothID	Sex	Twin	Status	Age	Yob	1BrCa	2BrCa	OvCa	ProCa	PanCa	Gtest	Mutn	Ashkn	Er	Pr	Her2	Ck14	Ck56
1 Amanda	T	1	2	3	F		dead	65	1920	30		45				srch	brca1&2		+ve	-ve	-ve	
2 mom		3			F		dead	unsp														
3 dad		2			M		alive	65	1892													

Below is the option for reading this file:

```

--DTH=Status --boadicea-var
event=BrCa:head=1BrCa:main=Dx:other=Er,Pr,Her2,Ck14,Ck56/event=OvCa:main=Dx/event=ProCa:main=Dx/
event=PanCa:main=Dx --cvt Geno:brca1=Het,brca1&2=Het;Death:dead=yes,alive=no;fptp:t=1 --gvoi BRCA1

```

Troubleshooting

[\[Back to top\]](#)

PedPro log is at the end of the result page. You can search for “warning” or “error” to find potential problems.

In addition, you may want to check whether the program recognizes all columns you want the program to handle. This program allows unknown columns. Therefore, the program does not know whether it missed some columns you want it to read (hence no warning or error). To check whether this is the case, you can search for “recognized” in the result page. That line shows what columns have been recognized by the program.

Second, check whether some lines are skipped, by searching “Skip lines” in the result page. Hopefully, no lines are skipped. A common mistake is, you put an empty string in a field, leading to two consecutive tabs. And you use the option “-s” which cause the program to read multiple consecutive tabs as one delimiter, hence skipping a