CommFunc.cpp:51: if (prob < 0 || prob > 1) LOGGER.e(0, "requested quantile probability is invalid");

CommFunc.cpp:138: if(!ifile) LOGGER.e(0, "cannot open the file ["+filename+"] to read.");

StatFunc.cpp:33: if (x < 0.0 || x > 1.0) LOGGER.e(0, "bad x in routine betai!");

StatFunc.cpp:101: if (x < 0.0 || a <= 0.0) LOGGER.e(0, "invalid arguments in routine gammp");

StatFunc.cpp:120: if (x < 0.0) LOGGER.e(0, "x is less than 0 in routine gser");

StatFunc.cpp:135: LOGGER.e(0, "a is too large, and ITMAX is too small in routine gser");

StatFunc.cpp:164: if (i > ITMAX) LOGGER.e(0, "a is too large, and ITMAX is too small in gcf");

StatFunc.cpp:180: if (size < 2) LOGGER.e(0, "invalid size. StatFunc::gasdev\_seq");

StatFunc.cpp:226: if (a >= b) LOGGER.e(0, "b must be larger than a. StatFunc::UniformDev");

StatFunc.cpp:271: else LOGGER.e(0, "invalid degree of freedom. StatFunc::chidev");

StatFunc.cpp:414: if (Size <= 1) LOGGER.e(0, "invalid size. StatFunc::ControlFDR");

StatFunc.cpp:559: if (h == 0.0) LOGGER.e(0, "bad xa input to routine splint");

StrFunc.cpp:99: if(Pos>=SizeA) LOGGER.e(0, "invalid Pos. StrFunc::StrVecEqual");

bivar\_reml.cpp:45: LOGGER.e(0, "bivariate REML can't handle multiple GRMs with different sample sizes. \n"

bivar\_reml.cpp:80: if (\_n < 1) LOGGER.e(0, "no individuals are in common among the input files.");

bivar\_reml.cpp:120: if (!(fabs(\_y\_Ssq) < 1e30)) LOGGER.e(0, "the phenotypic variance for trait 1 is infinite. Please check the missing data in your phenotype file. Missing values should be represented by \"NA\" or \"-9\".");

bivar\_reml.cpp:123: if (!(fabs(\_y2\_Ssq) < 1e30)) LOGGER.e(0, "the phenotypic variance for trait 2 is infinite. Please check the missing data in your phenotype file. Missing values should be represented by \"NA\" or \"-9\".");

bivar\_reml.cpp:144: //if(flag\_CC2!=\_flag\_CC) LOGGER.e(0, "for a bivariate analysis, the two traits should be both quantitative or both binary.");

bivar\_reml.cpp:361: if (!comput\_inverse\_logdet\_LU\_mkl(Vi, logdet)) LOGGER.e(0, "the variance-covariance matrix V is not invertible.");

bivar\_reml.cpp:367: if(\_reml\_have\_bend\_A) LOGGER.e(0, errmsg);

bivar\_reml.cpp:504: else LOGGER.e(0, "unable to calculate the genetic correlation because one of the genetic variance components is negative.");

bivar\_reml.cpp:518: else LOGGER.e(0, errmsg);

bivar\_reml.cpp:522: else LOGGER.e(0, errmsg);

bivar\_reml.cpp:526: else LOGGER.e(0, errmsg);

data.cpp:94: if (!Fam) LOGGER.e(0, "cannot open the file [" + famfile + "] to read.");

data.cpp:137: if (size == \_id\_map.size()) LOGGER.e(0, "Duplicate individual ID found: \"" + \_fid[i] + "\t" + \_pid[i] + "\".");

data.cpp:149: if (!Bim) LOGGER.e(0, "cannot open the file [" + bimfile + "] to read.");

data.cpp:213: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:214: if (\_keep.size() == 0) LOGGER.e(0, "no individual is retained for analysis.");

data.cpp:226: if (!BIT) LOGGER.e(0, "cannot open the file [" + bedfile + "] to read.");

data.cpp:237: if (!BIT) LOGGER.e(0, "problem with the BED file ... has the FAM/BIM file been changed?");

data.cpp:371: LOGGER.e(0, "cannot open the file [" + bfile\_list + "] to read.");

data.cpp:389: if(!Fam) LOGGER.e(0, "cannot open the file [" + famfile + "] to read.");

data.cpp:466: if (size == \_id\_map.size()) LOGGER.e(0, "duplicated individual IDs found: " + \_fid[i] + " " + \_pid[i] + ".");

data.cpp:479: if(fa\_id\_buf[i] != \_fa\_id[indx]) LOGGER.e(0, "inconsistent paternal IDs found, " + fid\_buf[i] + " " + pid\_buf[i] + ", from [" + famfile + "].");

data.cpp:480: if(mo\_id\_buf[i] != \_mo\_id[indx]) LOGGER.e(0, "inconsistent maternal IDs found, " + fid\_buf[i] + " " + pid\_buf[i] + ", from [" + famfile + "].");

data.cpp:481: if(sex\_buf[i] != \_sex[indx]) LOGGER.e(0, "inconsistent gender found, " + fid\_buf[i] + " " + pid\_buf[i] + ", from [" + famfile + "].");

data.cpp:482: if(pheno\_buf[i] != \_pheno[indx]) LOGGER.e(0, "inconsistent phenotype found, " + fid\_buf[i] + " " + pid\_buf[i] + ", from [" + famfile + "].");

data.cpp:483: if(i!=indx) LOGGER.e(0, "inconsistent order of individuals found from [" + famfile + "]. Please make sure that the order of individuals is the same across the fam files.");

data.cpp:486: LOGGER.e(0, "unexpected individual ID found, " + fid\_buf[i] + " " + pid\_buf[i] + ", from [" + famfile + "].");

data.cpp:530: if(!Bim) LOGGER.e(0, "cannot open the file [" + bimfile + "] to read.");

data.cpp:627: if(!BIT) LOGGER.e(0, "cannot open the file [" + bedfile + "] to read.");

data.cpp:642: if (!BIT) LOGGER.e(0, "problem with the BED file ... has the FAM/BIM file been changed?");

data.cpp:669: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:670: if (\_keep.size() == 0) LOGGER.e(0, "no individual is retained for analysis.");

data.cpp:712: if (!zinf.is\_open()) LOGGER.e(0, "cannot open the file [" + zinfofile + "] to read.");

data.cpp:721: if (col\_num < 7) LOGGER.e(0, errmsg);

data.cpp:722: if (vs\_buf[6] != "Rsq") LOGGER.e(0, errmsg);

data.cpp:733: if (!(ss >> c\_buf)) LOGGER.e(0, nerr);

data.cpp:735: if (!(ss >> c\_buf)) LOGGER.e(0, nerr);

data.cpp:737: for (i = 0; i < 4; i++) if (!(ss >> f\_buf)) LOGGER.e(0, nerr);

data.cpp:758: if(infofile.substr(infofile.length()-3,3)==".gz") LOGGER.e(0, "the --dosage-mach option doesn't support .gz file any more. Please check the --dosage-mach-gz option.");

data.cpp:762: if (!inf.is\_open()) LOGGER.e(0, "cannot open the file [" + infofile + "] to read.");

data.cpp:771: if (col\_num < 7) LOGGER.e(0, errmsg);

data.cpp:772: if (vs\_buf[6] != "Rsq" && vs\_buf[6] != "Rsq\_hat") LOGGER.e(0, errmsg);

data.cpp:782: if (!(ss >> c\_buf)) LOGGER.e(0, nerr);

data.cpp:784: if (!(ss >> c\_buf)) LOGGER.e(0, nerr);

data.cpp:786: for (i = 0; i < 3; i++) if (!(ss >> f\_buf)) LOGGER.e(0, nerr);

data.cpp:803: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:811: if (!zinf.is\_open()) LOGGER.e(0, "cannot open the file [" + zdosefile + "] to read.");

data.cpp:840: if (vs\_buf[0].empty()) LOGGER.e(0, "the family ID of the individual [" + str\_buf + "] is missing.");

data.cpp:904: if (\_keep.size() == 0) LOGGER.e(0, "no individual is retained for analysis.");

data.cpp:913: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:914: if(dosefile.substr(dosefile.length()-3,3)==".gz") LOGGER.e(0, "the --dosage-mach option doesn't support .gz file any more. Please check the --dosage-mach-gz option.");

data.cpp:921: if (!idose) LOGGER.e(0, "cannot open the file [" + dosefile + "] to read.");

data.cpp:949: if (vs\_buf[0].empty()) LOGGER.e(0, "the family ID of the individual [" + str\_buf + "] is missing.");

data.cpp:1010: if (\_keep.size() == 0) LOGGER.e(0, "no individual is retained for analysis.");

data.cpp:1030: if (!zinf.is\_open()) LOGGER.e(0, "cannot open the file [" + zinfofile + "] to read.");

data.cpp:1038: if (!(ss >> i\_buf)) LOGGER.e(0, nerr);

data.cpp:1040: if (!(ss >> str\_buf)) LOGGER.e(0, nerr);

data.cpp:1042: if (!(ss >> i\_buf)) LOGGER.e(0, nerr);

data.cpp:1044: if (!(ss >> c\_buf)) LOGGER.e(0, nerr);

data.cpp:1046: if (!(ss >> c\_buf)) LOGGER.e(0, nerr);

data.cpp:1048: if (!(ss >> str\_buf)) LOGGER.e(0, nerr);

data.cpp:1049: if (!(ss >> str\_buf)) LOGGER.e(0, nerr);

data.cpp:1050: if (!(ss >> str\_buf)) LOGGER.e(0, nerr);

data.cpp:1051: if (!(ss >> f\_buf)) LOGGER.e(0, nerr);

data.cpp:1052: if (!(ss >> f\_buf)) LOGGER.e(0, nerr);

data.cpp:1053: if (!(ss >> f\_buf)) LOGGER.e(0, nerr);

data.cpp:1055: if (!(ss >> f\_buf)) LOGGER.e(0, nerr);

data.cpp:1056: if (!(ss >> f\_buf)) LOGGER.e(0, nerr);

data.cpp:1057: if (ss >> f\_buf) LOGGER.e(0, nerr);

data.cpp:1072: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:1083: if (!zinf.is\_open()) LOGGER.e(0, "cannot open the file [" + zdosefile + "] to read.");

data.cpp:1121: LOGGER.e(0, errmsg.str());

data.cpp:1148: if (!OutBed) LOGGER.e(0, "cannot open the file [" + OutBedFile + "] to write.");

data.cpp:1190: if (!Fam) LOGGER.e(0, "cannot open the fam file " + famfile + " to save!");

data.cpp:1204: if (!Bim) LOGGER.e(0, "cannot open the file [" + bimfile + "] to write.");

data.cpp:1268: if (!i\_snplist) LOGGER.e(0, "cannot open the file [" + snplistfile + "] to read.");

data.cpp:1290: if (\_include.empty()) LOGGER.e(0, "cannot find the SNP [" + snpname + "] in the data.");

data.cpp:1301: if(iter==\_snp\_name\_map.end()) LOGGER.e(0, "cannot find the SNP [" + snpname + "] in the data.");

data.cpp:1310: if(snplist.empty()) LOGGER.e(0, "no SNP found in this region.");

data.cpp:1324: if(snplist.empty()) LOGGER.e(0, "no SNP found in this region.");

data.cpp:1345: if(iter==\_snp\_name\_map.end()) LOGGER.e(0, "cannot find the SNP [" + snpname + "] in the data.");

data.cpp:1354: if(snplist.empty()) LOGGER.e(0, "no SNP found in this region.");

data.cpp:1368: if(snplist.empty()) LOGGER.e(0, "no SNP found in this region.");

data.cpp:1379: if (\_include.size() == include\_size) LOGGER.e(0, "cannot find the SNP [" + snpname + "] in the data.");

data.cpp:1417: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:1441: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:1462: if (\_include.size() == 0) LOGGER.e(0, "no SNP is retained for analysis.");

data.cpp:1472: if (!i\_indi\_list) LOGGER.e(0, "cannot open the file [" + indi\_list\_file + "] to read.");

data.cpp:1504: if (!isex) LOGGER.e(0, "cannot open the file [" + sex\_file + "] to read.");

data.cpp:1517: if (str\_buf != "1" && str\_buf != "2" && str\_buf != "M" && str\_buf != "F") LOGGER.e(0, "unrecognized sex code: \"" + fid + " " + pid + " " + str\_buf + "\" in [" + sex\_file + "].");

data.cpp:1530: if (confirm[\_keep[i]] != 1) LOGGER.e(0, "the sex information for all of the included individuals should be updated.");

data.cpp:1537: if (!i\_ref\_A) LOGGER.e(0, "cannot open the file [" + ref\_A\_file + "] to read.");

data.cpp:1555: } else LOGGER.e(0, "invalid reference allele for SNP \"" + \_snp\_name[iter->second] + "\".");

data.cpp:1646: if (!iRsq) LOGGER.e(0, "cannot open the file [" + zinfofile + "] to read.");

data.cpp:1662: if (fbuf > 2.0 || fbuf < 0.0) LOGGER.e(0, "invalid value of imputation Rsq for the SNP " + snp\_name\_buf + ".");

data.cpp:1676: if (!ifreq) LOGGER.e(0, "cannot open the file [" + freq + "] to read.");

data.cpp:1694: if (fbuf > 1.0 || fbuf < 0.0) LOGGER.e(0, "invalid value of allele frequency for the SNP " + snp\_name\_buf + ".");

data.cpp:1696: LOGGER.e(0, "Invalid allele type \"" + ref\_A\_buf + "\" for the SNP " + \_snp\_name[iter->second] + ".");

data.cpp:1714: if (!ofreq) LOGGER.e(0, "cannot open the file [" + save\_freq + "] to write.");

data.cpp:1729: if (!i\_indi\_blup) LOGGER.e(0, "cannot open the file [" + blup\_indi\_file + "] to read.");

data.cpp:1974: if(std && \_dosage\_flag) LOGGER.e(0, "the --recode-std is invalid for dosage data.");

data.cpp:1993: if (!zoutf.is\_open()) LOGGER.e(0, "cannot open the file [" + X\_zFile + "] to write.");

edata.cpp:19: if (!einf.is\_open()) LOGGER.e(0, "cannot open the file [" + efile + "] to read.");

edata.cpp:26: if(col\_num < 3) LOGGER.e(0, "there needs to be at least 3 columns in the file [" + efile + "].");

edata.cpp:46: if (!(ss >> id\_buf)){ errmsg<<"in line "<<i+2<<"."; LOGGER.e(0, errmsg.str()); }

edata.cpp:48: if (!(ss >> id\_buf)){ errmsg<<"in line "<<i+2<<"."; LOGGER.e(0, errmsg.str()); }

edata.cpp:51: if (!(ss >> id\_buf)){ errmsg<<"in line "<<i+2<<"."; LOGGER.e(0, errmsg.str()); }

edata.cpp:73: if (size == \_probe\_name\_map.size()) LOGGER.e(0, "duplicated probe names found: \"" + \_probe\_name[i] + "\".");

ejma.cpp:69: if (!eR\_inf.is\_open()) LOGGER.e(0, "cannot open the file [" + eR\_file + "] to read.");

ejma.cpp:80: if(!(eR\_inf >> \_ecojo\_wholeR(i,j))) LOGGER.e(0, "incorrect format of [" + eR\_file + "].");

ejma.cpp:93: if (!e\_meta) LOGGER.e(0, "cannot open the file [" + e\_metafile + "] to read.");

ejma.cpp:101: if (StrFunc::split\_string(str\_buf, vs\_buf) < 3) LOGGER.e(0, "there needs to be at least 3 columns in the file [" + e\_metafile + "].");

ejma.cpp:106: if(!(iss >> str\_buf)){ errmsg<<"in line "<<line<<"."; LOGGER.e(0, errmsg.str()); }

ejma.cpp:109: if(!(iss >> d\_buf)){ errmsg<<"in line "<<line<<"."; LOGGER.e(0, errmsg.str()); }

ejma.cpp:111: if(!(iss >> d\_buf)){ errmsg<<"in line "<<line<<"."; LOGGER.e(0, errmsg.str()); }

ejma.cpp:116: if(probe\_buf.size()<1) LOGGER.e(0, "no probe remains in the analysis.");

ejma.cpp:187: if (slct.size() >= \_keep.size()) LOGGER.e(0, "too many probes. The number of probes in a joint analysis should not be larger than the sample size.");

ejma.cpp:198: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

ejma.cpp:438: //if (!comput\_inverse\_logdet\_LU\_mkl(\_ecojo\_wholeR, logdet)) LOGGER.e(0, "\n the correlation matrix is not invertible.");

ejma.cpp:445: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

est\_hsq.cpp:65: if (!in\_phen) LOGGER.e(0, "cannot open the file [" + phen\_file + "] to read.");

est\_hsq.cpp:75: if (phen\_num <= 0) LOGGER.e(0, "no phenotype data is found.");

est\_hsq.cpp:80: LOGGER.e(0, errmsg.str());

est\_hsq.cpp:85: LOGGER.e(0, errmsg.str());

est\_hsq.cpp:104: LOGGER.e(0, errmsg.str());

est\_hsq.cpp:130: if (!in\_covar) LOGGER.e(0, "cannot open the file [" + covar\_file + "] to read.");

est\_hsq.cpp:158: if (line > 0 && fac\_num != prev\_fac\_num) LOGGER.e(0, "each row should have the same number of columns.\n" + id\_buf + "\t" + str\_buf);

est\_hsq.cpp:176: if (!in\_GE) LOGGER.e(0, "cannot open the file [" + GE\_file + "] to read.");

est\_hsq.cpp:187: if (GE\_num == 0) LOGGER.e(0, "no " + env + " factor is specified. Please check the format of the file: " + GE\_file + ".");

est\_hsq.cpp:196: if (!in\_phen) LOGGER.e(0, "cannot open the weight [" + phen\_file + "] to read.");

est\_hsq.cpp:206: if (phen\_num <= 0) LOGGER.e(0, "no weight data is found.");

est\_hsq.cpp:219: LOGGER.e(0, errmsg.str());

est\_hsq.cpp:314: if (\_n < 1) LOGGER.e(0, "no individual is in common among the input files.");

est\_hsq.cpp:329: //if(mlmassoc) LOGGER.e(0, "the option --mlm-assoc is valid for quantitative traits only.");

est\_hsq.cpp:531: LOGGER.e(0, errmsg.str());

est\_hsq.cpp:535: if (\_r\_indx.size() == \_r\_indx\_drop.size()) LOGGER.e(0, "no component has been dropped from the model. Please check the --reml-lrt option.");

est\_hsq.cpp:553: if (qcovar\_num == 0) LOGGER.e(0, "no quantitative covariate is found.");

est\_hsq.cpp:612: if (value.size() > 0.5 \* vec.size()) LOGGER.e(0, errmsg1); // LOGGER.e(0, "too many classes for the environmental factor. \nPlease make sure you input a discrete variable as the environmental factor.");

est\_hsq.cpp:613: if (value.size() == 1) LOGGER.e(0, errmsg2); //LOGGER.e(0, "the environmental factor should have at least two classes.");

est\_hsq.cpp:648: LOGGER.e(0, "invalid phenotype. Please check the phenotype file.");

est\_hsq.cpp:731: if (!(fabs(\_y\_Ssq) < 1e30)) LOGGER.e(0, "the phenotypic variance is infinite. Please check the missing data in your phenotype file. Missing values should be represented by \"NA\" or \"-9\".");

est\_hsq.cpp:736: LOGGER.e(0, errmsg.str());

est\_hsq.cpp:740: LOGGER.e(0, errmsg.str());

est\_hsq.cpp:744: if (\_n < 10) LOGGER.e(0, "sample size is too small.");

est\_hsq.cpp:851: if (!o\_reml) LOGGER.e(0, "cannot open the file [" + reml\_rst\_file + "] to write.");

est\_hsq.cpp:997: if (d\_buf > 1.0) LOGGER.e(0, "\n --reml-priors. The sum of all prior values for trait 1 should not exceed 1.0.");

est\_hsq.cpp:1004: if (d\_buf > 1.0) LOGGER.e(0, "\n --reml-priors. The sum of all prior values for trait 2 should not exceed 1.0.");

est\_hsq.cpp:1027: if (d\_buf > 1.0) LOGGER.e(0, "\n --reml-priors. The sum of all prior values should not exceed 1.0.");

est\_hsq.cpp:1096: if(!calcu\_Vi(\_Vi, varcmp, logdet, iter)) LOGGER.e(0, "V matrix is not positive-definite.");

est\_hsq.cpp:1154: LOGGER.e(0, "analysis stopped because more than half of the variance components are constrained. The result would be unreliable.\n You may have a try of the option --reml-no-constrain.");

est\_hsq.cpp:1158: //if (constrain\_num == \_r\_indx.size()) LOGGER.e(0, "analysis stopped because all variance components are constrained. You may have a try of the option --reml-no-constrain.");

est\_hsq.cpp:1190: if (\_reml\_max\_iter > 1) LOGGER.e(0, errmsg.str());

est\_hsq.cpp:1322: LOGGER.e(0, "the variance-covariance matrix V is not invertible.\nWe may try --reml-alg-inv 1 to add a small constant value to the diagonals or --reml-alg-inv 2 to bend the matrix V.");

est\_hsq.cpp:1369: else LOGGER.e(0, "the variance-covariance matrix V is not positive definite.");

est\_hsq.cpp:1385: else LOGGER.e(0, "the variance-covariance matrix V is not invertible using LU decomposition.");

est\_hsq.cpp:1520: if(!SquareMatrixInverse(Xt\_Vi\_X\_i, logdet\_Xt\_Vi\_X, rank, method)) LOGGER.e(0, "\n the X^t \* V^-1 \* X matrix is not invertible. Please check the covariate(s) and/or the environmental factor(s).");

est\_hsq.cpp:1563: else LOGGER.e(0, "the information matrix is not invertible.");

est\_hsq.cpp:1630: else LOGGER.e(0, "the information matrix is not invertible.");

est\_hsq.cpp:1785: if (!o\_b\_snp) LOGGER.e(0, "cannot open the file " + o\_b\_snp\_file + " to write.");

est\_hsq.cpp:1828: LOGGER.e(0, "file [" + grm\_files[i] + "] contains a different number of individuals from other GRM files.");

est\_hsq.cpp:1833: if ((\*A\_bin[i]).bad()) LOGGER.e(0, "cannot open the file [" + grm\_binfile + "] to read.");

est\_hsq.cpp:1854: if (\_n < 1) LOGGER.e(0, "no individual is in common among the input files.");

est\_hsq.cpp:2093: if (!os) LOGGER.e(0, "cannot open the file [" + ofile + "] to write.");

est\_hsq.cpp:2129: LOGGER.e(0, "file [" + grm\_files[i] + "] contains a different number of individuals from other GRM files.");

est\_hsq.cpp:2134: if ((\*A\_bin[i]).bad()) LOGGER.e(0, "cannot open the file [" + grm\_binfile + "] to read.");

est\_hsq.cpp:2159: if (\_n < 1) LOGGER.e(0, "no individual is in common among the input files.");

est\_hsq.cpp:2195: if (y1.size()==0) LOGGER.e(0, "no non-missing phenotypes for trait 1.");

est\_hsq.cpp:2196: if (y2.size()==0) LOGGER.e(0, "no non-missing phenotypes for trait 2.");

est\_hsq.cpp:2611: if (!os) LOGGER.e(0, "cannot open the file [" + ofile + "] to write.");

est\_hsq.cpp:2638: if (\_n < 1) LOGGER.e(0, "no individual is in common among the input files.");

est\_hsq.cpp:2691: if (!os) LOGGER.e(0, "cannot open the file [" + ofile + "] to write.");

gbat.cpp:18: if (!in\_snpAssoc) LOGGER.e(0, "cannot open the file [" + snpAssoc\_file + "] to read.");

gbat.cpp:25: if (StrFunc::split\_string(str\_buf, vs\_buf) != 2) LOGGER.e(0, "in line \"" + str\_buf + "\".");

gbat.cpp:57: if (\_include.size() < 1) LOGGER.e(0, "no SNP is included in the analysis.");

gbat.cpp:58: else if (\_chr[\_include[0]] < 1) LOGGER.e(0, "chromosome information is missing.");

gbat.cpp:59: else if (\_bp[\_include[0]] < 1) LOGGER.e(0, "bp information is missing.");

gbat.cpp:64: if (!in\_gAnno) LOGGER.e(0, "cannot open the file [" + gAnno\_file + "] to read.");

gbat.cpp:69: if (StrFunc::split\_string(str\_buf, vs\_buf) != 4) LOGGER.e(0, "in line \"" + str\_buf + "\".");

gbat.cpp:174: if (mapped < 1) LOGGER.e(0, "no gene can be mapped to the SNP data. Please check the input data regarding chromosome and bp.");

gbat.cpp:228: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

grm.cpp:23: if (\_chr[\_include[i]] > \_autosome\_num) LOGGER.e(0, "this option is for the autosomal SNPs only. Please check the option --autosome.");

grm.cpp:29: if (\_chr[\_include[i]] != (\_autosome\_num + 1)) LOGGER.e(0, "this option is for SNPs on the X chromosome only.");

grm.cpp:35: if (\_sex[\_keep[i]] != 1 && \_sex[\_keep[i]] != 2) LOGGER.e(0, "sex information of the individual \"" + \_fid[\_keep[i]] + " " + \_pid[\_keep[i]] + "\" is missing.\nUse --update-sex option to update the sex information of the individuals.");

grm.cpp:196: if (!A\_Bin) LOGGER.e(0, "cannot open the file [" + grm\_file + "] to write.");

grm.cpp:210: if (!N\_Bin) LOGGER.e(0, "cannot open the file [" + grm\_N\_file + "] to write.");

grm.cpp:227: if (!zoutf.is\_open()) LOGGER.e(0, "cannot open the file [" + grm\_file + "] to write.");

grm.cpp:247: if (!Fam) LOGGER.e(0, "cannot open the file [" + famfile + "] to write.");

grm.cpp:259: if (!i\_grm\_id) LOGGER.e(0, "cannot open the file [" + grm\_id\_file + "] to read.");

grm.cpp:305: if (!zinf.is\_open()) LOGGER.e(0, "cannot open the file [" + grm\_gzfile + "] to read.");

grm.cpp:317: if (!(ss >> indx1)) LOGGER.e(0, errmsg + buf);

grm.cpp:318: if (!(ss >> indx2)) LOGGER.e(0, errmsg + buf);

grm.cpp:319: if (!(ss >> grm\_N\_buf)) LOGGER.e(0, errmsg + buf);

grm.cpp:320: if (!(ss >> grm\_buf)) LOGGER.e(0, errmsg + buf);

grm.cpp:321: if (indx1 < indx2 || indx1 > n || indx2 > n) LOGGER.e(0, errmsg + buf);

grm.cpp:326: if (ss >> str\_buf) LOGGER.e(0, errmsg + buf);

grm.cpp:332: LOGGER.e(0, errmsg\_tmp.str());

grm.cpp:333: //LOGGER.e(0, "incorrect number of lines in the grm file. Are \*.grm.gz file and \*.grm.id file mismatched?");

grm.cpp:346: if (!A\_bin.is\_open()) LOGGER.e(0, "cannot open the file [" + grm\_binfile + "] to read.");

grm.cpp:353: if (!(A\_bin.read((char\*) &f\_buf, size))) LOGGER.e(0, "Is the size of the [" + grm\_binfile + "] file incorrect?");

grm.cpp:362: if (!N\_bin.is\_open()) LOGGER.e(0, "cannot open the file [" + grm\_Nfile + "] to read.");

grm.cpp:369: if (!(N\_bin.read((char\*) &f\_buf, size))) LOGGER.e(0, "Is the size of the [" + grm\_Nfile + "] file incorrect?");

grm.cpp:531: if (\_n == 0) LOGGER.e(0, "no individual is in common among the GRM files.");

grm.cpp:577: if (\_n == 0) LOGGER.e(0, "no individual is in common among the GRM files.");

grm.cpp:624: if (!merge\_grm) LOGGER.e(0, "cannot open the file [" + merge\_grm\_file + "] to read.");

grm.cpp:634: if (grm\_files.size() > 1000) LOGGER.e(0, "too many GRM file names specified in [" + merge\_grm\_file + "]. The maximum number is 1000.");

grm.cpp:635: if (grm\_files.size() < 1) LOGGER.e(0, "no GRM file name is found in [" + merge\_grm\_file + "].");

grm.cpp:682: if (!o\_eval) LOGGER.e(0, "cannot open the file [" + eval\_file + "] to read.");

grm.cpp:688: if (!o\_evec) LOGGER.e(0, "cannot open the file [" + evec\_file + "] to read.");

grm.cpp:704: if (!in\_eigenval) LOGGER.e(0, "cannot open the file [" + eigenval\_file + "] to read.");

grm.cpp:707: if (!in\_eigenvec) LOGGER.e(0, "cannot open the file [" + eigenvec\_file + "] to read.");

grm.cpp:722: if(d\_buf > 1e10 || d\_buf < 1e-10) LOGGER.e(0, "invalid eigenvalue in the file [" + eigenval\_file + "].");

grm.cpp:726: if(eigenvec\_num != eigenval\_num) LOGGER.e(0, "inconsistent numbers of eigenvalues and eigenvectors in the files [" + eigenval\_file + "] and [" + eigenvec\_file + "]");

grm.cpp:739: if(\_n < 1) LOGGER.e(0, "no individual is in common among the input files.");

grm.cpp:766: if(!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

grm.cpp:789: if (!h\_pc\_load) LOGGER.e(0, "cannot open the loading file [" + f\_pc\_load + "] to read.");

grm.cpp:795: if(!ofile) LOGGER.e(0, "failed to open the file [" + out\_filename + "] to write.");

grm.cpp:815: LOGGER.e(0, "invalid number of columns in the PC loading file. Please check.");

grm.cpp:818: LOGGER.e(0, "only " + to\_string(N\_loading\_file) + " vectors of PC loadings available, thus not able to project onto " + to\_string(N) + " PCs.");

gsmr.cpp:13: LOGGER.e(0, "cannot open file [" + input\_file + "] to read.");

gsmr.cpp:25: LOGGER.e(0, "the GWAS summary data should be in GCTA-COJO format. Please check.");

gsmr.cpp:51: LOGGER.e(0, "cannot open the file [" + gsmr\_file\_list + "] to read.");

gsmr.cpp:63: LOGGER.e(0, "the format of file [" + gsmr\_file\_list + "] is incorrect, line " + to\_string(line\_number) + ".");

gsmr.cpp:76: LOGGER.e(0, "invalid sample prevalence for [" + pheno\_name[line\_number] + "].");

gsmr.cpp:81: LOGGER.e(0, "invalid population prevalence for [" + pheno\_name[line\_number] + "].");

gsmr.cpp:157: if(nsnp<nsnp\_gsmr) LOGGER.e(0, "not enough SNPs to perform the GSMR analysis.");

gsmr.cpp:173: if(\_meta\_vp\_trait[i] < 0) LOGGER.e(0, "negative phenotypic variance of trait " + \_gwas\_trait\_name[i] + ".");

gsmr.cpp:196: if(nsnp<1) LOGGER.e(0, "no SNP is retained for the GSMR analysis.");

gsmr.cpp:512: if(!o\_pleio\_snp) LOGGER.e(0, "cannot open file [" + pleio\_snpfile + "] to write pleiotropic SNPs.");

gsmr.cpp:530: if (!zofile) LOGGER.e(0, "cannot open the file [" + output\_filename + "] to write.");

gsmr.cpp:544: if (!ofile) LOGGER.e(0, "cannot open the file [" + output\_filename + "] to write.");

gwas\_simu.cpp:23: if (!i\_qtl) LOGGER.e(0, "cannot open the file [" + qtl\_file + "] to read.");

gwas\_simu.cpp:38: if (fabs(qtl\_eff\_buf) > 1e5) LOGGER.e(0, "invalid effect size specified for the causal variant [" + str\_buf + "].");

gwas\_simu.cpp:54: if(qtl\_name\_buf.size() < qtl\_name.size()) LOGGER.e(0, "there are duplicated SNP IDs.");

gwas\_simu.cpp:64: if (!out\_par) LOGGER.e(0, "cannot open par file [" + out\_parfile + "] to write!");

gwas\_simu.cpp:75: if (!phen) LOGGER.e(0, "cannot open the file [" + phenfile + "] to write.");

gwas\_simu.cpp:205: if (!out\_emBayesB) LOGGER.e(0, "cannot open the file [" + out\_rstfile + "] to write.");

gwas\_simu.cpp:229: if(gnrt<1.0 || gnrt>1e5) LOGGER.e(0, "--simu-gener should be within the range from 1 to 100000.");

gwas\_simu.cpp:230: if(hsq>1.0 || hsq<0.0) LOGGER.e(0, "--simu-h2 should be within the range from 0 to 1.");

gwas\_simu.cpp:231: if(K>0.5 || K<0.0001) LOGGER.e(0, "--simu-K should be within the range from 0.0001 to 0.5.");

gwas\_simu.cpp:232: if(!curr\_popu && (case\_num>1e5 || case\_num<1)) LOGGER.e(0, "--simu-cc, Invalid number of cases.");

gwas\_simu.cpp:233: if(!curr\_popu && (control\_num>1e6 || control\_num<1)) LOGGER.e(0, "--simu-cc, Invalid number of controls.");

gwas\_simu.cpp:299: if(!out\_par) LOGGER.e(0, "cannot open par file ["+out\_parfile+"] to write!");

gwas\_simu.cpp:370: if(!out\_fam) LOGGER.e(0, "cannot open fam file ["+out\_famfile+"] to write!");

gwas\_simu.cpp:410: if(!OutBed) LOGGER.e(0, "cannot open the file ["+OutBedFile+"] to write.");

gwas\_simu.cpp:475: if(!Fam) LOGGER.e(0, "cannot open the fam file "+famfile+" to save!");

gwas\_simu.cpp:490: if(!Bim) LOGGER.e(0, "cannot open the file ["+bimfile+"] to write.");

gwas\_simu.cpp:516: if (!i\_genet\_map) LOGGER.e(0, "cannot open the HAPMAP genetic map file " + genet\_mapfile + ".");

gwas\_simu.cpp:557: if (!out\_bim) LOGGER.e(0, "cannot open file " + out\_bimfile + " to write.");

joint\_meta.cpp:22: if (!Meta) LOGGER.e(0, "cannot open the file [" + metafile + "] to read.");

joint\_meta.cpp:34: if (StrFunc::split\_string(str\_buf, vs\_buf) < 7) LOGGER.e(0, "format error in the input file [" + metafile + "].");

joint\_meta.cpp:57: if (N\_buf < 10) LOGGER.e(0, "invalid sample size in line:\n\"" + str\_buf0 + "\"");

joint\_meta.cpp:62: if (Vp\_buf < 0.0) LOGGER.e(0, "in line:\n\"" + str\_buf0 + "\"");

joint\_meta.cpp:66: if (GC\_buf < 0) LOGGER.e(0, "in line:\n\"" + str\_buf0 + "\"");

joint\_meta.cpp:157: if (\_include.empty()) LOGGER.e(0, "none of the SNPs in the GWAS summary data can be found in the genotype data.");

joint\_meta.cpp:211: if (givenSNPs.empty()) LOGGER.e(0, "failed to read any SNP from the file [" + snplistfile + "].");

joint\_meta.cpp:221: else LOGGER.e(0, "none of the given SNPs can be matched to the genotype and summary data.");

joint\_meta.cpp:264: if (slct.size() >= \_keep.size()) LOGGER.e(0, "too many SNPs. The number of SNPs in a joint analysis should not be larger than the sample size.");

joint\_meta.cpp:294: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

joint\_meta.cpp:313: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

joint\_meta.cpp:330: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

joint\_meta.cpp:353: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

joint\_meta.cpp:433: if (!init\_B(slct)) LOGGER.e(0, "there is a collinearity problem of the given list of SNPs.\nYou can try the option --cojo-slct to remove one of each pair of highly correlated SNPs.");

joint\_meta.cpp:463: if (d\_buf - n < 1) LOGGER.e(0, "no degree of freedom is left for the residues. The model is over-fitted. Please specify a more stringent p-value cut-off.");

joint\_meta.cpp:465: if (Ve <= 0.0) LOGGER.e(0, "residual variance is out of boundary. The model is over-fitted. Please specify a more stringent p-value cut-off.");

joint\_meta.cpp:471: if (!init\_B(indx)) LOGGER.e(0, "there is a collinearity problem of the given list of SNPs.\nYou can try the option --cojo-slct to remove one of each pair of highly correlated SNPs.");

joint\_meta.cpp:501: if (!init\_B(slct)) LOGGER.e(0, "there is a collinearity problem of the given list of SNPs.\nYou can try the option --cojo-slct to remove one of each pair of highly correlated SNPs.");

joint\_meta.cpp:801: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

joint\_meta.cpp:807: } else LOGGER.e(0, "Jacobi iteration cannot converge. You can increase the maximum number of iterations by the option --massoc-sblup-maxit.");

joint\_meta.cpp:826: LOGGER.e(0, "there are " + to\_string(num\_err\_snp) + " SNPs with MAF=0.");

joint\_meta.cpp:886: LOGGER.e(0, "decomposition failed. The SNP correlation matrix is not positive definite.");

joint\_meta.cpp:890: LOGGER.e(0, "solving failed. Unable to solve the BLUP equation.");

ld.cpp:37: if (\_ld\_target\_snp.size() == 0) LOGGER.e(0, "no target SNP is retained to estimate the LD structure.");

ld.cpp:69: if (!SavFile) LOGGER.e(0, "cannot open the file [" + SavFileName + "] to save result!");

ld.cpp:75: if (!SavFile) LOGGER.e(0, "cannot open the file [" + SavFileName + "] to save result.");

ld.cpp:83: if (!SavFile) LOGGER.e(0, "cannot open the file [" + SavFileName + "] to save result.");

ld.cpp:174: if (N != y.size() || N < 1) LOGGER.e(0, "The lengths of x and y do not match.");

ld.cpp:507: if (!in\_snpset\_filenames) LOGGER.e(0, "cannot open the file [" + snpset\_filenames\_file + "] to read.");

ld.cpp:519: if (set\_num < 1) LOGGER.e(0, "no filename found in [" + snpset\_filenames\_file + "].");

ld.cpp:524: if (!i\_snplist) LOGGER.e(0, "cannot open the file [" + snpset\_filenaems[i] + "] to read.");

ld.cpp:752: if (!ild) LOGGER.e(0, "cannot open the file [" + i\_ld\_file + "] to read.");

ld.cpp:777: if(!(ild >> \_chr[i])) LOGGER.e(0, "in the file [" + i\_ld\_file + "].");

ld.cpp:778: if(!(ild >> \_bp[i])) LOGGER.e(0, "in the file [" + i\_ld\_file + "].");

ld.cpp:779: if(!(ild >> \_mu[i])) LOGGER.e(0, "in the file [" + i\_ld\_file + "].");

ld.cpp:781: if(!(ild >> mrsq[i])) LOGGER.e(0, "in the file [" + i\_ld\_file + "].");

ld.cpp:782: if(!(ild >> snp\_num[i])) LOGGER.e(0, "in the file [" + i\_ld\_file + "].");

ld.cpp:783: if(!(ild >> max\_rsq[i])) LOGGER.e(0, "in the file [" + i\_ld\_file + "].");

ld.cpp:784: if(!(ild >> ldscore[i])) LOGGER.e(0, "in the file [" + i\_ld\_file + "].");

mkl.cpp:303: if (!A\_Bin) LOGGER.e(0, "cannot open the file [" + grm\_file + "] to write.");

mkl.cpp:313: if (!N\_Bin) LOGGER.e(0, "cannot open the file [" + grm\_N\_file + "] to write.");

mkl.cpp:325: if (!zoutf.is\_open()) LOGGER.e(0, "cannot open the file [" + grm\_file + "] to write.");

mkl.cpp:338: if (!Fam) LOGGER.e(0, "cannot open the file [" + famfile + "] to write.");

mkl.cpp:373: LOGGER.e(0, "Cholesky decomposition failed. Invalid values found in the matrix.\n");

mkl.cpp:394: LOGGER.e(0, "invalid values found in the variance-covariance (V) matrix.\n");

mkl.cpp:436: if (INFO < 0) LOGGER.e(0, "LU decomposition failed. Invalid values found in the matrix.\n");

mkl.cpp:456: LOGGER.e(0, "invalid values found in the variance-covariance (V) matrix.\n");

mkl.cpp:498: if (INFO < 0) LOGGER.e(0, "LU decomposition failed. Invalid values found in the matrix.\n");

mkl.cpp:519: if (INFO < 0) LOGGER.e(0, "invalid values found in the variance-covariance (V) matrix.\n");

mlm\_assoc.cpp:26: if (subtract\_grm\_flag && m\_grm\_flag) LOGGER.e(0, "the --mlma-subtract-grm option cannot be used in combination with the --mgrm option.");

mlm\_assoc.cpp:35: LOGGER.e(0, "no file name in --pheno.");

mlm\_assoc.cpp:49: LOGGER.e(0, "no individual is in common among the input files.");

mlm\_assoc.cpp:88: if(\_n<1) LOGGER.e(0, "no individual is in common in the input files.");

mlm\_assoc.cpp:232: if(!ofile) LOGGER.e(0, "cannot open the file ["+filename+"] to write.");

mlm\_assoc.cpp:335: if(!comput\_inverse\_logdet\_LU\_mkl\_array(col\_num, Xt\_Vi\_X, d\_buf)) LOGGER.e(0, "Xt\_Vi\_X is not invertible.");

mlm\_assoc.cpp:368: LOGGER.e(0, "no file name in --pheno.");

mlm\_assoc.cpp:383: if(\_n<1) LOGGER.e(0, "no individual is in common among the input files.");

mlm\_assoc.cpp:389: if(vi\_buf.size()<2) LOGGER.e(0, "There is only one chromosome. The MLM leave-on-chromosome-out (LOCO) analysis requires at least two chromosomes.");

mlm\_assoc.cpp:509: if(!ofile) LOGGER.e(0, "cannot open the file ["+filename+"] to write.");

mtcojo.cpp:141: LOGGER.e(0, "cannot open the file [" + mtcojolist\_file + "] to read.");

mtcojo.cpp:153: LOGGER.e(0, "the format of file [" + mtcojolist\_file + "] is incorrect, line " + to\_string(line\_number) + ".");

mtcojo.cpp:165: LOGGER.e(0, "invalid sample prevalence for trait [" + target\_pheno + "].");

mtcojo.cpp:170: LOGGER.e(0, "invalid population prevalence for trait [" + target\_pheno + "].");

mtcojo.cpp:184: LOGGER.e(0, "the format of file [" + mtcojolist\_file + "] is incorrect, line " + to\_string(line\_number) + ".");

mtcojo.cpp:197: LOGGER.e(0, "invalid sample prevalence for trait [" + line\_elements[0] + "].");

mtcojo.cpp:202: LOGGER.e(0, "invalid population prevalence for trait [" + line\_elements[0] + "].");

mtcojo.cpp:217: if(size==snp\_name\_map.size()) LOGGER.e(0, "Duplicated SNP ID found: " + snplist[i] + ".");

mtcojo.cpp:259: if(size==snp\_add\_map.size()) LOGGER.e(0, "Duplicated SNP ID found: " + snplist[i] + ".");

mtcojo.cpp:324: LOGGER.e(0, "cannot open the file [" + metafile + "] to read.");

mtcojo.cpp:337: LOGGER.e(0, "the GWAS summary data file [" + metafile + "] should be in GCTA-COJO format, line " + to\_string(line\_number) + ".");

mtcojo.cpp:363: LOGGER.e(0, "cannot open the file [" + metafile + "] to read.");

mtcojo.cpp:374: if (!(ss >> snpbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:379: if (!(ss >> strbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:401: LOGGER.e(0, "cannot open the file [" + metafile + "] to read.");

mtcojo.cpp:417: LOGGER.e(0, "the GWAS summary data file [" + metafile + "] should be in GCTA-COJO format, line " + to\_string(line\_number) + ".");

mtcojo.cpp:467: LOGGER.e(0, "cannot open the file [" + metafile + "] to read.");

mtcojo.cpp:487: if (!(ss >> snpbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:492: if (!(ss >> strbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:496: if (!(ss >> strbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:500: if (!(ss >> valbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:504: if (!(ss >> valbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:508: if (!(ss >> valbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:512: if (!(ss >> valbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:516: if (!(ss >> valbuf)) LOGGER.e(0, err\_msg + to\_string(line\_number) + "," + buf);

mtcojo.cpp:613: if(!obadsnp) LOGGER.e(0, "cannot open file [" + badsnpfile + "] to write bad SNPs.");

mtcojo.cpp:666: if(!oafsnp) LOGGER.e(0, "cannot open file [" + afsnpfile + "] to write bad SNPs.");

mtcojo.cpp:672: LOGGER.e(0, "there are too many SNPs that have large difference in allele frequency. Please check the GWAS summary data.");

mtcojo.cpp:694: if(!oafsnp) LOGGER.e(0, "cannot open file [" + afsnpfile + "] to write bad SNPs.");

mtcojo.cpp:785: if(\_meta\_vp\_trait(0) < 0) LOGGER.e(0, "negative phenotypic variance of the target trait, " + \_covar\_pheno\_name[0] + ".");

mtcojo.cpp:798: if(\_meta\_vp\_trait(i+1) < 0) LOGGER.e(0, "negative phenotypic variance of the covariate #" + to\_string(i+1) + ", " + \_covar\_pheno\_name[i+1] + ".");

mtcojo.cpp:822: if(nsnp<1) LOGGER.e(0, "no SNP is retained after filtering.");

mtcojo.cpp:848: if (!extern\_bxy) LOGGER.e(0, "cannot open the file [" + filestr + "] to read.");

mtcojo.cpp:856: if(line\_elements.size() != 2) LOGGER.e(0, "the format of file [" + filestr + "] is not correct.");

mtcojo.cpp:1137: LOGGER.e(0, "the variance-covariance matrix of bxy is not invertible.");

mtcojo.cpp:1706: if (!ref\_marker) LOGGER.e(0, "cannot open the file [" + filestr + "] to read.");

mtcojo.cpp:1711: if(line\_elements.size() != 1) LOGGER.e(0, "the format of file [" + filestr + "] is incorrect.");

mtcojo.cpp:1738: LOGGER.e(0, "the format of file [" + filestr + "] is incorrect, line " + to\_string(line\_number) + ".");

mtcojo.cpp:1779: LOGGER.e(0, "the format of file [" + filestr + "] is not correct, line " + to\_string(line\_number) + ".");

mtcojo.cpp:1816: else LOGGER.e(0, "cannot open the file [" + filestr\_t1 + "] or [" + filestr\_t2 + "] to read.");

mtcojo.cpp:2128: LOGGER.e(0, "negative SNP heritability estimate for " + trait\_name[i] + ". Exiting ...");

mtcojo.cpp:2199: LOGGER.e(0, "\nno SNP in common between the summary data and the LD score files. Please check.");

mtcojo.cpp:2319: if (!ofile) LOGGER.e(0, "cannot open the file [" + output\_file + "] to write.");

mtcojo.cpp:2377: LOGGER.e(0, "not enough SNPs to perform the univariate LD score regression analysis. The mtCOJO analysis will be conducted assuming no sample overlap between the GWAS data for the target and covariate traits.");

mtcojo.cpp:2379: LOGGER.e(0, "not enough SNPs to perform the univariate and bivariate LD score regression analyses. The mtCOJO analysis requires SNP-based heritability from univariate LD score regression analysis and genetic correlation from bivariate LD score regression analysis.");

mtcojo.cpp:2413: if(!o\_pleio\_snp) LOGGER.e(0, "cannot open file [" + pleio\_snpfile + "] to write pleiotropic SNPs.");

option.cpp:183: if (thread\_num < 1 || thread\_num > 1000) LOGGER.e(0, "\n --thread-num should be from 1 to 1000.\n");

option.cpp:188: if (thread\_num < 1 || thread\_num > 1000) LOGGER.e(0, "\n --threads should be from 1 to 1000.\n");

option.cpp:199: if (GC\_cutoff < 0.0 || GC\_cutoff > 1.0) LOGGER.e(0, "\n --gencall should be within the range from 0 to 1.\n");

option.cpp:238: if (dose\_Rsq\_cutoff < 0.0 || dose\_Rsq\_cutoff > 1.0) LOGGER.e(0, "\n --imput-rsq should be within the range from 0 to 1.\n");

option.cpp:266: if (extract\_chr\_start < 1 || extract\_chr\_start > 100) LOGGER.e(0, "\n --chr should be within the range from 1 to 100.\n");

option.cpp:270: if (autosome\_num < 1 || autosome\_num > 100) LOGGER.e(0, "\n invalid number specified after the option --autosome-num.\n");

option.cpp:290: if(extract\_region\_wind < 1000 || extract\_region\_wind > 1e8) LOGGER.e(0, "\n the second parameter of --extract-region is distance in Kb unit. It should take value between 1 and 1e5.");

option.cpp:297: if(extract\_region\_wind < 1000 || extract\_region\_wind > 1e8) LOGGER.e(0, "\n the second parameter of --extract-region is distance in Kb unit. It should take value between 1 and 1e5.");

option.cpp:306: if(exclude\_region\_wind < 1000 || exclude\_region\_wind > 1e8) LOGGER.e(0, "\n the second parameter of --exclude-region is distance in Kb unit. It should take value between 1 and 1e5.");

option.cpp:313: if(exclude\_region\_wind < 1000 || exclude\_region\_wind > 1e8) LOGGER.e(0, "\n the second parameter of --exclude-region is distance in Kb unit. It should take value between 1 and 1e5.");

option.cpp:317: if (maf < 0 || maf > 0.5) LOGGER.e(0, "\n --maf should be within the range from 0 to 0.5.\n");

option.cpp:321: if (max\_maf <= 0) LOGGER.e(0, "\n --max-maf should be > 0.\n");

option.cpp:385: if (rm\_high\_ld\_cutoff <= 0 || rm\_high\_ld\_cutoff >= 1) LOGGER.e(0, "\n the value to be specified after --rm-high-ld should be within the range from 0 to 1.\n");

option.cpp:400: if (make\_grm\_mtd < 0 || make\_grm\_mtd > 1) LOGGER.e(0, "\n --make-grm-alg should be 0 or 1.\n");

option.cpp:447: if (grm\_adj\_fac < 0 || grm\_adj\_fac > 1) LOGGER.e(0, "\n the value to be specified after --grm-adj should be within the range from 0 to 1.\n");

option.cpp:451: if (dosage\_compen != 0 && dosage\_compen != 1) LOGGER.e(0, "\n the value to be specified after --dc should be 0 or 1.\n");

option.cpp:461: if (bK\_threshold < 0 || bK\_threshold > 1) LOGGER.e(0, "\n --make-bK threshold should be range from 0 to 1.\n");

option.cpp:472: if (out\_pc\_num < 1) LOGGER.e(0, "\n the value to be specified after --pca should be positive.\n");

option.cpp:479: //if(pcl\_grm\_N < 1 || pcl\_grm\_N > 1e20) LOGGER.e(0, "\n invalid number of SNPs used to calculate PCs.");

option.cpp:486: if(project\_N < 1 || project\_N > 1e3) LOGGER.e(0, "\n invalid number of PCs to output");

option.cpp:498: if (LD\_step < 1 || LD\_step > 20) LOGGER.e(0, "\n --ld-step should be within the range from 1 to 20.\n");

option.cpp:506: LOGGER.e(0, err\_msg.str());

option.cpp:511: if (LD\_sig <= 0) LOGGER.e(0, "\n --ld-sig should be > 0.\n");

option.cpp:519: if (LD\_prune\_rsq < 0.0001 || LD\_prune\_rsq > 0.9999) LOGGER.e(0, "\n --ld-pruning should be within the range from 0.0001 to 0.9999.\n");

option.cpp:539: LOGGER.e(0, err\_msg.str());

option.cpp:554: if (LD\_seg < 10) LOGGER.e(0, "\n the input value for --ld-score-region needs to be > 10.\n");

option.cpp:569: if (simu\_case\_num < 10) LOGGER.e(0, "--simu-cc, Invalid number of cases. Minimum number 10.");

option.cpp:570: if (simu\_control\_num < 10) LOGGER.e(0, "--simu-cc, Invalid number of controls. Minimum number 10.");

option.cpp:574: if (simu\_rep < 1 || simu\_rep > 10000) LOGGER.e(0, "--simu-rep should be within the range from 1 to 10000.");

option.cpp:578: if (simu\_h2 > 1.0 || simu\_h2 < 0.0) LOGGER.e(0, "--simu-h2 should be within the range from 0 to 1.");

option.cpp:582: if (simu\_K > 0.5 || simu\_K < 0.0001) LOGGER.e(0, "--simu-K should be within the range from 0.0001 to 0.5.");

option.cpp:596: if (simu\_seed <= 100) LOGGER.e(0, "--simu-seed should be >100.");

option.cpp:600: if (simu\_eff\_mod != 0 && simu\_eff\_mod !=1) LOGGER.e(0, "--simu-eff-mod should be 0 or 1.");

option.cpp:621: if (mphen\_buf.size() < 2 && mphen\_buf.size() > 0) LOGGER.e(0, "\n --HEreg-bivar. Please specify two traits for the bivariate HE regression analysis.");

option.cpp:629: if (mphen < 1 || mphen2 < 1 || mphen == mphen2) LOGGER.e(0, "\n --HEreg-bivar. Invalid input parameters.");

option.cpp:640: if (prevalence <= 0 || prevalence >= 1) LOGGER.e(0, "\n --prevalence should be between 0 to 1.\n");

option.cpp:653: if (reml\_mtd < 0 || reml\_mtd > 2) LOGGER.e(0, "\n --reml-alg should be 0, 1 or 2.\n");

option.cpp:672: if (err\_flag || reml\_priors.empty()) LOGGER.e(0, "\n --reml-priors. Prior values of variance explained should be between 0 and 1.\n");

option.cpp:689: if (reml\_priors\_var.empty()) LOGGER.e(0, "\n " + s\_buf + ". Prior values of variance components are required.\n");

option.cpp:710: if (err\_flag || reml\_drop.empty()) LOGGER.e(0, "\n invalid values specified after --reml-lrt.\n");

option.cpp:714: if (MaxIter < 1 || MaxIter > 10000) LOGGER.e(0, "\n --reml-maxit should be within the range from 1 to 10000.\n");

option.cpp:749: if (mphen < 1) LOGGER.e(0, "--mpheno should be > 0.");

option.cpp:789: if (mphen\_buf.size() < 2 && mphen\_buf.size() > 0) LOGGER.e(0, "\n --reml-bivar. Please specify two traits for the bivariate REML analysis.");

option.cpp:797: if (mphen < 1 || mphen2 < 1 || mphen == mphen2) LOGGER.e(0, "\n --reml-bivar. Invalid input parameters.");

option.cpp:807: if (K\_buf.size() < 1 || K\_buf.size() > 2) LOGGER.e(0, "\n --reml-bivar-prevalence. Please specify the prevalence of the two diseases.");

option.cpp:809: if (K\_buf[0] < 0.0 || K\_buf[0] > 1.0 || K\_buf[1] < 0.0 || K\_buf[1] > 1.0) LOGGER.e(0, "\n --reml-bivar-prevalence. Disease prevalence should be between 0 and 1.");

option.cpp:814: if (K\_buf[0] < 0.0 || K\_buf[0] > 1.0) LOGGER.e(0, "\n --reml-bivar-prevalence. Disease prevalence should be between 0 and 1.");

option.cpp:835: if (err\_flag || fixed\_rg\_val.empty()) LOGGER.e(0, "\n --reml-bivar-lrt-rg. Any input parameter should be within the range from -1 to 1.\n");

option.cpp:839: if ((CommFunc::FloatNotEqual(fixed\_rg\_val[0], 0.0) && haveZero) || (CommFunc::FloatEqual(fixed\_rg\_val[0], 0.0) && !haveZero)) LOGGER.e(0, "\n --reml-bivar-lrt-rg. Input parameters should be all zero or all non-zero values.\n");

option.cpp:868: if (massoc\_top\_SNPs < 1 || massoc\_top\_SNPs > 10000) LOGGER.e(0, "\n --cojo-top-SNPs should be within the range from 1 to 10000.\n");

option.cpp:875: if (massoc\_p > 0.05 || massoc\_p <= 0) LOGGER.e(0, "\n --cojo-p should be within the range from 0 to 0.05.\n");

option.cpp:881: if (massoc\_collinear > 0.99 || massoc\_collinear < 0.01) LOGGER.e(0, "\n --cojo-collinear should be within the ragne from 0.01 to 0.99.\n");

option.cpp:887: if (massoc\_wind > 100000) LOGGER.e(0, "\n invalid value for --cojo-wind. Valid range: 100 ~ 100000\n");

option.cpp:889: //if (massoc\_wind < 100 || massoc\_wind > 100000) LOGGER.e(0, "\n invalid value for --cojo-wind. Valid range: 100 ~ 100000\n");

option.cpp:905: if (massoc\_gc\_val < 1 || massoc\_gc\_val > 10) LOGGER.e(0, "\n invalid value specified after --cojo-gc.\n");

option.cpp:912: if (massoc\_sblup\_fac < 0) LOGGER.e(0, "\n invalid value for --cojo-sblup.\n");

option.cpp:942: if (sbat\_ld\_cutoff <= 0.1) LOGGER.e(0, "\n --fastBAT\_ld\_cutoff should be > 0.1\n");

option.cpp:961: if (sbat\_wind < 0 || sbat\_wind > 1000) LOGGER.e(0, "\n invalid value for --fastBAT-wind. Valid range: 0 ~ 1000\n");

option.cpp:972: if (sbat\_seg\_size < 10 || sbat\_seg\_size > 10000) LOGGER.e(0, "\n invalid value for --fastBAT-seg. Valid range: 10 ~ 10000\n");

option.cpp:999: if (ecojo\_p > 0.05 || ecojo\_p <= 0) LOGGER.e(0, "\n --ecojo-p should be within the range from 0 to 0.05.\n");

option.cpp:1004: if (ecojo\_collinear > 1 || ecojo\_collinear < 0.01) LOGGER.e(0, "\n --ecojo-collinear should be within the range from 0.01 to 0.99.\n");

option.cpp:1010: if (ecojo\_lambda < 0.01 || ecojo\_lambda > 0.99) LOGGER.e(0, "\n --ecojo-blup should be within the range from 0.01 to 0.99.\n");

option.cpp:1028: if (make\_erm\_mtd < 1 || make\_erm\_mtd > 3) LOGGER.e(0, "\n --make-erm-alg should be 1, 2 or 3.\n");

option.cpp:1040: LOGGER.e(0, "--gsmr-file, please specify the GWAS summary data for the exposure(s) and the outcome(s).");

option.cpp:1053: LOGGER.e(0, "--gsmr-direction should be 0 (forward-GSMR), 1 (reverse-GSMR) or 2 (bi-GSMR).");

option.cpp:1056: LOGGER.e(0, "--gsmr-alg has been superseded by --gsmr-direction.");

option.cpp:1061: LOGGER.e(0, "--gsmr-so should be 0 (LD score regression) or 1 (correlation of SNP effects).");

option.cpp:1104: LOGGER.e(0, "--diff-freq, Invalid threshold used to check allele frequency difference.");

option.cpp:1109: LOGGER.e(0, "--gwas-thresh, Invalid GWAS p-value threshold.");

option.cpp:1120: LOGGER.e(0, "--heidi-thresh, please specify p-value threshold(s) for the HEIDI-outlier analysis.");

option.cpp:1127: LOGGER.e(0, "--heidi-thresh, Invalid p-value threshold for single-SNP-based HEIDI-outlier test.");

option.cpp:1129: LOGGER.e(0, "--heidi-thresh, Invalid p-value threshold for multi-SNP-based HEIDI-outlier test.");

option.cpp:1135: LOGGER.e(0, "--heidi-snp is discontinued. Please use --gsmr-snp-min to specify minimum number of SNP instruments for the HEIDI-outlier analysis.");

option.cpp:1140: LOGGER.e(0, "--gsmr-snp-min, Invalid SNP number threshold for the GSMR analysis.");

option.cpp:1145: LOGGER.e(0, "--gsmr-ld-fdr, Invalid FDR threshold for LD correlation matrix.");

option.cpp:1148: LOGGER.e(0, "--clump-p1 is discontinued. Please use --gwas-thresh to specify a p-value threshold for index SNPs.");

option.cpp:1152: LOGGER.e(0, "--clump-kb, Invalid window size for the clumping analysis.");

option.cpp:1157: LOGGER.e(0, "--clump-r2, Invalid LD r2 threshold for the clumping analysis.");

option.cpp:1172: LOGGER.e(0, errmsg.str());

option.cpp:1177: if (bfile2\_flag && !bfile\_flag) LOGGER.e(0, "the option --bfile2 should always go with the option --bfile.");

option.cpp:1178: if(bfile\_flag && grm\_cutoff>-1.0) LOGGER.e(0, "the --grm-cutoff option is invalid when used in combination with the --bfile option.");

option.cpp:1215: if (dosage\_compen>-1 && update\_sex\_file.empty()) LOGGER.e(0, "you need to specify the sex information for the individuals by the option --update-sex because of the use of the –dc option.");

option.cpp:1216: if (bfile2\_flag && update\_freq\_file.empty()) LOGGER.e(0, "you need to update the allele frequency by the option --update-freq because there are two datasets.");

option.cpp:1217: if ((dose\_beagle\_flag || dose\_mach\_flag || dose\_mach\_gz\_flag) && dominance\_flag) LOGGER.e(0, "unable to calculate the GRM for dominance effect using imputed dosage data.");

option.cpp:1218: if (make\_grm\_xchar\_flag && dominance\_flag) LOGGER.e(0, "unable to calculate the GRM for dominance effect for the X chromosome.");

option.cpp:1229: if(bivar\_reml\_flag && prevalence\_flag) LOGGER.e(0, "--prevalence option is not compatible with --reml-bivar option. Please check the --reml-bivar-prevalence option!");

option.cpp:1231: if(ref\_ld\_flag && !w\_ld\_flag) LOGGER.e(0, "--ref-ld-chr, please specify the directory of the LD score files.");

option.cpp:1232: if(!ref\_ld\_flag && w\_ld\_flag) LOGGER.e(0, "--w-ld-chr, please specify the directory of the LD scores for the regression weights.");

option.cpp:1237: // if(gsmr\_so\_alg == 0 && !ref\_ld\_flag && !w\_ld\_flag) LOGGER.e(0, "Please specify the directory of LD score files to perform the LD score regression analysis.");

option.cpp:1283: if (RG\_summary\_file.empty()) LOGGER.e(0, "please input the summary information for the raw data files by the option --raw-summary.");

option.cpp:1301: if (update\_freq\_file.empty()) LOGGER.e(0, "since there are two datasets, you should update the allele frequencies calculated from the combined dataset.");

option.cpp:1379: if (massoc\_slct\_flag | massoc\_joint\_flag | !massoc\_cond\_snplist.empty()) LOGGER.e(0, "the --dosage option can't be used in combination with the --cojo options.");

option.cpp:1416: else if ((reml\_flag || bivar\_reml\_flag) && phen\_file.empty()) LOGGER.e(0, "\n a phenotype file is required for reml analysis.\n");

option.cpp:1430: else LOGGER.e(0, "no analysis has been launched by the option(s).\n");

pc\_adjust.cpp:18: if (!in\_lambda) LOGGER.e(0, "cannot open the file [" + eigenvalue\_file + "] to read.");

pc\_adjust.cpp:35: if (!meta\_list) LOGGER.e(0, "cannot open the file [" + pcadjust\_list\_file + "] to read.");

pc\_adjust.cpp:53: LOGGER.e(0, "At least 1 PC loading vector is required.");

pc\_adjust.cpp:55: LOGGER.e(0, "there are summary data for " + to\_string(ncovar) + " covariates, but " + to\_string(\_eigen\_value.size()) + " eigenvalues are provided.");

pc\_adjust.cpp:102: if(i==0) LOGGER.e(0, "negative phenotypic variance of the target trait.");

pc\_adjust.cpp:103: else LOGGER.e(0, "negative phenotypic variance of the covariate #" + to\_string(i+1) + ".");

pc\_adjust.cpp:127: if(nsnp<1) LOGGER.e(0, "no SNP is retained after filtering.");

pc\_adjust.cpp:176: if(!oafsnp) LOGGER.e(0, "cannot open file [" + afsnpfile + "] to write bad SNPs.");

pc\_adjust.cpp:182: LOGGER.e(0, "there are too many SNPs without allele frequencies. Please check the GWAS summary data.");

pc\_adjust.cpp:428: if (!ofile) LOGGER.e(0, "cannot open the file [" + output\_file + "] to write.");

popu\_genet.cpp:27: if(!i\_aa) LOGGER.e(0, "cannot open the file ["+aa\_file+"] to read.");

popu\_genet.cpp:101: if(!o\_paa) LOGGER.e(0, "cannot open the file ["+paa\_file+"] to write.");

popu\_genet.cpp:202: if(!o\_ibc) LOGGER.e(0, "cannot open the file ["+ibc\_file+"] to write.");

popu\_genet.cpp:235: if(!ofile) LOGGER.e(0, "cannot open the file ["+outfile+"] to write.");

popu\_genet.cpp:274: if(!ifstream\_subpopu) LOGGER.e(0, "cannot open the file ["+filename+"] to read.");

raw\_geno.cpp:20: if(!i\_snp\_info) LOGGER.e(0, "cannot open the file ["+snp\_info\_file+"] to read.");

raw\_geno.cpp:40: if(!omap) LOGGER.e(0, "cannot open the file ["+map\_file+"] to write.");

raw\_geno.cpp:47: if(!i\_fnames) LOGGER.e(0, "cannot open the file ["+fname\_file+"] to read.");

raw\_geno.cpp:67: if(!oped) LOGGER.e(0, "cannot open the file ["+ped\_file+"] to read.");

raw\_geno.cpp:88: if(!i\_IRG) LOGGER.e(0, "cannot open the file ["+IRG\_fname+"] to read.");

raw\_geno.cpp:101: if(snp\_num!=\_snp\_num) LOGGER.e(0, "the number of SNPs specified in the summary data file does not match that in the raw genotype file ["+IRG\_fname+"].");

raw\_geno.cpp:106: if(vs\_buf[0]!=\_snp\_name[i]) LOGGER.e(0, "the SNP ["+vs\_buf[0]+"] specified in the summary data file does not match that in the raw genotype file ["+IRG\_fname+"]. Has the order of the SNPs been changed?");

reml\_within\_family.cpp:60: LOGGER.e(0, "can't invert the within-family relatedness matrix");

sbat.cpp:18: if (!in\_snpAssoc) LOGGER.e(0, "cannot open the file [" + snpAssoc\_file + "] to read.");

sbat.cpp:26: if (StrFunc::split\_string(str\_buf, vs\_buf, " \t") != 2) LOGGER.e(0, "in line \"" + str\_buf + "\".");

sbat.cpp:62: if (\_include.size() < 1) LOGGER.e(0, "no SNP is included in the analysis.");

sbat.cpp:63: else if (\_chr[\_include[0]] < 1) LOGGER.e(0, "chromosome information is missing.");

sbat.cpp:64: else if (\_bp[\_include[0]] < 1) LOGGER.e(0, "bp information is missing.");

sbat.cpp:69: if (!in\_gAnno) LOGGER.e(0, "cannot open the file [" + gAnno\_file + "] to read.");

sbat.cpp:74: if (StrFunc::split\_string(str\_buf, vs\_buf) != 4) LOGGER.e(0, "in line \"" + str\_buf + "\".");

sbat.cpp:156: if (mapped < 1) LOGGER.e(0, "no gene can be mapped to the SNP data. Please check the input data regarding chromosome and bp.");

sbat.cpp:228: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

sbat.cpp:247: if (!in\_snpset) LOGGER.e(0, "cannot open the file [" + snpset\_file + "] to read.");

sbat.cpp:366: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");

sbat.cpp:462: if (!ofile) LOGGER.e(0, "cannot open the file [" + filename + "] to write.");