Package ‘iCARE’

November 20, 2016

Title A Tool for Individualized Coherent Absolute Risk Estimation (iCARE)

Version 1.2.0

Date 2015-07-20

Author Paige Maas, Nilanjan Chatterjee and William Wheeler

Description An R package to compute Individualized Coherent Absolute Risk Estimators.

Maintainer Bill Wheeler <wheelerb@imsweb.com>

Depends R (>= 3.3.0)

Suggests RUnit, BiocGenerics

License GPL-3 + file LICENSE

LazyData true

biocViews Software, StatisticalMethod, GenomeWideAssociation

NeedsCompilation yes

R topics documented:

bc_data ................................................................. 1
computeAbsoluteRisk ............................................... 2
computeAbsoluteRiskSplitInterval ................................. 5
iCARE ................................................................. 8

Index

bc_data Data for examples

Description

Data for computeAbsoluteRisk and computeAbsoluteRiskSplitInterval

Details

"mort_inc" contains age-specific incidence rates of all-cause mortality from reference (1) below
"bc_15_snps" contains published SNP information from reference (2)
"bc_inc" contains age-specific incidence rates of breast cancer from reference (3)
"ref_cov_dat" contains a subsample of data imputed using reference (4) and (5)
References


Examples

```r
temp <- data(bc_data, package="iCARE")

# Display the object names
temp
```

---

### `computeAbsoluteRisk`  Building and Applying an Absolute Risk Model

#### Description

This function is used to build absolute risk models and apply them to estimate absolute risks.

#### Usage

```r
computeAbsoluteRisk(model.formula = NULL, model.cov.info = NULL,
                     model.snp.info = NULL, model.log.RR = NULL, model.ref.dataset = NULL,
                     model.ref.dataset.weights = NULL, model.disease.incidence.rates,
                     model.competing.incidence.rates = NULL, model.bin.fh.name = NA,
                     n.imp = 5, apply.age.start, apply.age.interval.length,
                     apply.cov.profile = NULL, apply.snp.profile = NULL, use.c.code = 1,
                     return.lp = FALSE, return.refs.risk = FALSE)
```
computeAbsoluteRisk

Arguments

model.formula  an object of class formula: a symbolic description of the model to be fitted, e.g. Y~Parity+FamilyHistory.
model.cov.info  contains information about the risk factors in the model; a main list containing a list for each covariate, which must have the fields:
  • "name": a string with the covariate name, matching name in model.formula
  • "type": a string that is either "continuous" or "factor".
    If factor variable, then:
    • "levels": vector with strings of level names
    • "ref": optional field, string with name of referent level
model.snp.info  dataframe with three columns, named: ["snp.name", "snp.odds.ratio", "snp.freq"]
model.log.RR  vector with log odds ratios corresponding to the model params; no intercept; names must match design matrix arising from model.formula and model.cov.info; check names using function check_design_matrix().
model.ref.dataset  dataframe of risk factors for a sample of subjects representative of underlying population, no missing values. Variables must be in same order with same names as in model.formula.
model.ref.dataset.weights  optional vector of sampling weights for model.ref.dataset.
model.disease.incidence.rates  two column matrix [integer ages, incidence rates] or three column matrix [start age, end age, rate] with incidence rate of disease. Must fully cover age interval for estimation.
model.competing.incidence.rates  two column matrix [integer ages, incidence rates] or three column matrix [start age, end age, rate] with incidence rate of competing events. Must fully cover age interval for estimation.
model.bin.fh.name  string name of family history variable, if in model. This must refer to a variable that only takes values 0,1, NA.
n.imp  integer value for number of imputations for handling missing SNPs.
apply.age.start  single integer or vector of integer ages for the start of the interval over which to compute absolute risk.
apply.age.interval.length  single integer or vector of integer years over which absolute risk should be computed.
apply.cov.profile  dataframe containing the covariate profiles for which absolute risk will be computed. Covariates must be in same order with same names as in model.formula.
apply.snp.profile  data frame with observed SNP data (coded 0,1, 2, or NA). May have missing values.
computeAbsoluteRisk

use.c.code binary indicator of whether to run the c program for fast computation.
return.lp binary indicator of whether to return the linear predictor for each subject in apply.cov.profile.
return.refs.risk binary indicator of whether to return the absolute risk prediction for each subject in model.ref.dataset.

Details

Individualized Coherent Absolute Risk Estimators (iCARE) is a tool that allows researchers to quickly build models for absolute risk and apply them to estimate individuals’ risk based on a set of user defined input parameters. The software gives users the flexibility to change or update models rapidly based on new risk factors or tailor models to different populations based on the specification of simply three input arguments:

• (1) a model for relative risk assumed to be externally derived
• (2) an age-specific disease incidence rate and
• (3) the distribution of risk factors for the population of interest.

The tool can handle missing information on risk factors for risk estimation using an approach where all estimates are derived from a single model through appropriate model averaging.

Value

This function returns a list of results objects, including:

• risk: absolute risk estimates over the specified interval for subjects given by apply.cov.profile
• details: dataframe with the start of the interval, the end of the interval, the covariate profile, and the risk estimates for each individual
• beta.used: the log odds ratios used in the model
• lps: linear predictors for subjects in model.cov.profile, if requested by return.lp
• refs.risk: absolute risk estimates for subjects in model.ref.dataset, if requested by return.refs.risk; computes for first age interval provided

Examples

data(bc_data, package="iCARE")
results = computeAbsoluteRisk(model.formula = caco ~ famhist + as.factor(parity),
model.cov.info = bc_model_cov_info,
model.snp.info = bc_15_snps,
model.log.RR = bc_model_log_or,
model.ref.dataset = ref_cov_dat,
model.disease.incidence.rates = bc_inc,
model.competing.incidence.rates = mort_inc,
model.bin.fh.name = "famhist",
apply.age.start = 50,
apply.age.interval.length = 30,
apply.cov.profile = new_cov_prof,
computeAbsoluteRiskSplitInterval

apply.snp.profile = new_snprof,
return.refs.risk = TRUE)
summary(results$risk)
plot(density(results$risk, na.rm=TRUE))
boxplot(results$risk ~ new_cov_prof$famhist, na.rm=TRUE)

--

computeAbsoluteRiskSplitInterval

Building and Applying an Absolute Risk Model: Compute Risk over Interval Split in Two Parts

Description

This function is used to build an absolute risk model that incorporates different input parameters before and after a given time point. The model is then applied to estimate absolute risks.

Usage

computeAbsoluteRiskSplitInterval(apply.age.start, apply.age.interval.length,
apply.cov.profile, model.formula, model.disease.incidence.rates, model.log.RR,
model.ref.dataset, model.ref.dataset.weights=NULL, model.cov.info, use.c.code=1,
model.competing.incidence.rates=NULL, return.lp=FALSE, apply.snp.profile=NULL,
model.snp.info=NULL, model.bin.fh.name=NULL, cut.time=NULL,
apply.cov.profile.2=NULL, model.formula.2=NULL, model.log.RR.2=NULL,
model.ref.dataset.2=NULL, model.ref.dataset.weights.2=NULL, model.cov.info.2=NULL,
model.bin.fh.name.2=NULL, n.imp=5, return.refs.risk=FALSE)

Arguments

apply.age.start
  single integer or vector of integer ages for the start of the interval over which to compute absolute risk.
apply.age.interval.length
  single integer or vector of integer years over which absolute risk should be computed.
apply.cov.profile
  dataframe containing the covariate profiles for which absolute risk will be computed. Covariates must be in same order with same names as in model.formula.
model.formula
  an object of class formula: a symbolic description of the model to be fitted, e.g. Y~Parity+FamilyHistory.
model.disease.incidence.rates
  two column matrix [integer ages, incidence rates] or three column matrix [start age, end age, rate] with incidence rate of disease. Must fully cover age interval for estimation.
model.log.RR
  vector with log odds ratios corresponding to the model params; no intercept; names must match design matrix arising from model.formula and model.cov.info; check names using function check_design_matrix().
model.ref.dataset
  dataframe of risk factors for a sample of subjects representative of underlying population, no missing values. Variables must be in same order with same names as in model.formula.
computeAbsoluteRiskSplitInterval

model.ref.dataset.weights
optional vector of sampling weights for model.ref.dataset.

model.cov.info
contains information about the risk factors in the model; a main list containing
a list for each covariate, which must have the fields:

- "name": a string with the covariate name, matching name in model.formula
- "type": a string that is either "continuous" or "factor".

If factor variable, then:

- "levels": vector with strings of level names
- "ref": optional field, string with name of referent level

use.c.code
binary indicator of whether to run the c program for fast computation.

model.competing.incidence.rates
two column matrix [integer ages, incidence rates] or three column matrix [start
age, end age, rate] with incidence rate of competing events. Must fully cover
age interval for estimation.

return.lp
binary indicator of whether to return the linear predictor for each subject in
apply.cov.profile.

apply.snp.profile
data frame with observed SNP data (coded 0,1, 2, or NA). May have missing
values.

model.snp.info
data frame with three columns [rs number, odds ratio, allele frequency]

model.bin.fh.name
string name of family history variable, if in model. This must refer to a variable
that only takes values 0,1, NA.

cut.time
integer age for which to split computation into before and after

apply.cov.profile.2
see apply.cov.profile, to be used for estimation in ages after the cutpoint

model.formula.2
see model.formula, to be used for estimation in ages after the cutpoint

model.log.RR.2
see model.log.RR, to be used for estimation in ages after the cutpoint

model.ref.dataset.2
see model.ref.dataset, to be used for estimation in ages after the cutpoint

model.ref.dataset.weights.2
see model.ref.dataset.weights, to be used for estimation in ages after the cutpoint

model.cov.info.2
see model.cov.info, to be used for estimation in ages after the cutpoint

model.bin.fh.name.2
see model.bin.fh.name, to be used for estimation in ages after the cutpoint

n.imp
integer value for number of imputations for handling missing SNPs.

return.refs.risk
binary indicator of whether to return the absolute risk prediction for each subject
in model.ref.dataset.
**Details**

Individualized Coherent Absolute Risk Estimators (iCARE) is a tool that allows researchers to quickly build models for absolute risk and apply them to estimate individuals’ risk based on a set of user defined input parameters. The software gives users the flexibility to change or update models rapidly based on new risk factors or tailor models to different populations based on the specification of simply three input arguments:

- (1) a model for relative risk assumed to be externally derived
- (2) an age-specific disease incidence rate and
- (3) the distribution of risk factors for the population of interest.

The tool can handle missing information on risk factors for risk estimation using an approach where all estimates are derived from a single model through appropriate model averaging.

**Value**

This function returns a list of results objects, including:

- `risk`: absolute risk estimates over the specified interval for subjects given by `apply.cov.profile`
- `details`: dataframe with the start of the interval, the end of the interval, the covariate profile, and the risk estimates for each individual
- `beta.used`: the log odds ratios used in the model
- `lps.1`: linear predictors based on first set of parameters for subjects in `model.cov.profile`, if requested by `return.lp`
- `lps.2`: linear predictors based on second set of parameters for subjects in `model.cov.profile`, if requested by `return.lp`
- `refs.risk`: absolute risk estimates for subjects in `model.ref.dataset`, if requested by `return.refs.risk`; computes for first age interval provided

**Examples**

data(bc_data, package="iCARE")

form <- caco ~ famhist + as.factor(parity)
results <- computeAbsoluteRiskSplitInterval(model.formula=form,
cut.time = 50,
model.cov.info = bc_model_cov_info,
model.snp.info = bc_15_snps,
model.log.RR = bc_model_log_or,
model.log.RR.2 = bc_model_log_or_post_50,
model.ref.dataset = ref_cov_dat,
model.ref.dataset.2 = ref_cov_dat_post_50,
model.disease.incidence.rates = bc_inc,
model.competing.incidence.rates = mort_inc,
model.bin.fh.name = "famhist",
apply.age.start = 30,
apply.age.interval.length = 40, 

apply.cov.profile
apply.cov.profile = new_cov_prof,
apply.snp.profile = new_snp_prof,
return.refs.risk = TRUE)

summary(results$risk)
plot(density(results$risk, na.rm=TRUE))
boxplot(results$risk ~ new_cov_prof$famhist, na.rm=TRUE)

---

**iCARE**

_A Tool for Individualized Coherent Absolute Risk Estimation (iCARE)_

**Description**

Individualized Coherent Absolute Risk Estimators (iCARE) is a tool that allows researchers to quickly build models for absolute risk and apply them to estimate individuals’ risk based on a set of user defined input parameters. The software gives users the flexibility to change or update models rapidly based on new risk factors or tailor models to different populations based on the specification of simply three input arguments: (1) a model for relative risk assumed to be externally derived (2) an age-specific disease incidence rate and (3) the distribution of risk factors for the population of interest. The tool can handle missing information on risk factors for risk estimation using an approach where all estimates are derived from a single model through appropriate model averaging.

**Details**

The main functions for building and applying an absolute risk model are `computeAbsoluteRisk` and `computeAbsoluteRiskSplitInterval`. The first of these computes absolute risks over the specified time interval using a single set of parameters. The second provides more advanced functionality and computes absolute risk over the interval in two parts. `computeAbsoluteRiskSplitInterval` allows the user to compute absolute risk over the interval in two parts, incorporating two different sets of parameters before and after a specified cutpoint. This function allows a different cutpoint for each covariate profile if desired.

**Author(s)**

Paige Maas, Nilanjan Chatterjee and William Wheeler <wheelerb@imsweb.com>
Index

*Topic data
  bc_data, 1

*Topic package
  iCARE, 8

bc_15_snps (bc_data), 1
bc_data, 1
bc_inc (bc_data), 1
bc_model_cov_info (bc_data), 1
bc_model_log_or (bc_data), 1
bc_model_log_or_post_50 (bc_data), 1

computeAbsoluteRisk, 1, 2, 8
computeAbsoluteRiskSplitInterval, 1, 5, 8

iCARE, 8

mort_inc (bc_data), 1

new_cov_prof (bc_data), 1
new_snp_prof (bc_data), 1

ref_cov_dat (bc_data), 1
ref_cov_dat_post_50 (bc_data), 1