Package ‘summclust’

October 14, 2022

Title  Module to Compute Influence and Leverage Statistics for Regression Models with Clustered Errors

Version  0.5

Description  Module to compute cluster specific information for regression models with clustered errors, including leverage and influence statistics. Models of type 'lm' and 'fixest'(from the 'stats' and 'fixest' packages) are supported. summclust implements similar features as the user-written 'summclust.ado' Stata module (MacKinnon, Nielsen & Webb, 2022; <arXiv:2205.03288v1>).

License  MIT + file LICENSE

Encoding  UTF-8

RoxygenNote  7.2.0

Imports  utils, dreamerr, MASS, collapse, generics

Suggests  ggrepplot2, latex2exp, fabricatr, fixest, haven, sandwich, lmtest, testthat (>= 3.0.0), knitr, rmarkdown, covr

Config/testthat/edition  3

URL  https://s3alfisc.github.io/summclust/

BugReports  https://github.com/s3alfisc/summclust/issues

VignetteBuilder  knitr

NeedsCompilation  no

Author  Alexander Fischer [aut, cre]

Maintainer  Alexander Fischer <alexander-fischer1801@t-online.de>

Repository  CRAN

Date/Publication  2022-10-04 05:30:03 UTC

R topics documented:

plot.summclust ........................................................................... 2
summary.summclust ..................................................................... 3
summclust .................................................................................... 4
Description

Plots residual leverage, partial leverage and the leave-one-cluster-out regression coefficients

Usage

```r
## S3 method for class 'summclust'
plot(x, ...)
```

Arguments

- `x` : An object of type `summclust`
- `...` : other optional function arguments

Details

Note that the function requires `ggplot2` to be installed.

Value

A list containing

- `residual_leverage` : A `ggplot` of the residual leverages
- `coef_leverage` : A `ggplot` of the coefficient leverages
- `coef_beta` : A `ggplot` of the leave-one-out cluster jackknife regression coefficients

References

Examples

```r
if(requireNamespace("summclust") && requireNamespace("haven")){
  library(summclust)
  library(haven)

  nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
  # drop NAs at the moment
  nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
  nlswork <- na.omit(nlswork)

  lm_fit <- lm(
    ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
    data = nlswork)

  res <- summclust(
    obj = lm_fit,
    params = c("msp", "union"),
    cluster = ~ind_code,
  )

  plot(res)
}
```

Description

A summary() method for objects of type summclust

Usage

```r
## S3 method for class 'summclust'
summary(object, ...)
```

Arguments

- `object`: An object of type summclust
- `...`: misc arguments

Value

The function summarysummclust returns a range of cluster leverage statistics based on an object of type summclust.
References


Examples

```r
if(requireNamespace("summclust") && requireNamespace("haven")) {
  library(summclust)
  library(haven)

  nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
  # drop NAs at the moment
  nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
  nlswork <- na.omit(nlswork)

  lm_fit <- lm(
    ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
    data = nlswork)

  res <- summclust(
    obj = lm_fit,
    params = c("msp", "union"),
    cluster = ~ind_code,
  )

  summary(res)
}
```

summclust

---

**Compute Influence and Leverage Metrics**

**Description**


**Usage**

`summclust(obj, ...)`

**Arguments**

- `obj` An object of class `lm` or `fixest`
- `...` Other arguments
Value

An object of type `summclust`, including a CRV3 variance-covariance estimate as described in MacKinnon, Nielsen & Webb (2022)

References


See Also

`summclust.lm`, `summclust.fixest`

Examples

```r
if(requireNamespace("summclust") && requireNamespace("haven")){

library(summclust)
library(haven)

nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
# drop NAs at the moment
nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
nlswork <- na.omit(nlswork)

lm_fit <- lm(
    ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
    data = nlswork)

res <- summclust(
    obj = lm_fit,
    params = c("msp", "union"),
    cluster = ~ind_code,
)

summary(res)
tidy(res)
plot(res)
}
```
**Description**

Compute influence and leverage metrics for clustered inference based on the Cluster Jackknife as described in MacKinnon, Nielsen & Webb (2022) for objects of type `fixest`.

**Usage**

```r
## S3 method for class 'fixest'
summclust(obj, cluster, params, absorb_cluster_fixef = TRUE, type, ...)
```

**Arguments**

- `obj`: An object of type `fixest`
- `cluster`: A clustering vector
- `params`: A character vector of variables for which leverage statistics should be computed. If NULL, leverage statistics will be computed for all k model covariates
- `absorb_cluster_fixef`: TRUE by default. Should the cluster fixed effects be projected out? This increases numerical stability and decreases computational costs
- `type`: "CRV3" or "CRV3J" following MacKinnon, Nielsen & Webb
- `...`: other function arguments passed to `vcov`

**Value**

An object of type `summclust`, including a CRV3 variance-covariance estimate as described in MacKinnon, Nielsen & Webb (2022)

- `coef_estimates`: The coefficient estimates of the linear model.
- `vcov`: A CRV3 or CRV3J variance-covariance matrix estimate as described in MacKinnon, Nielsen & Webb (2022)
- `leverage_g`: A vector of leverages.
- `leverage_avg`: The cluster leverage.
- `partial_leverage`: The partial leverages.
- `coef_var_leverage_avg`: Coefficient of Variation for the leverage statistic
- `coef_var_leverage_g`: Coefficient of Variation for the Partial Leverage Statistics
- `coef_var_N_G`: Coefficient of Variation for the Cluster Sizes.
- `beta_jack`: The jackknifed' leave-on-cluster-out regression coefficients.
- `params`: The input parameter vector 'params'.
- `N_G`: The number of clusters-
- `call`: The `summclust()` function call.
- `cluster`: The names of the clusters.
References


Examples

```r
if(requireNamespace("summclust")
  && requireNamespace("haven")
  && requireNamespace("fixest")){

  library(summclust)
  library(haven)
  library(fixest)

  nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
  # drop NAs at the moment
  nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
  nlswork <- na.omit(nlswork)

  feols_fit <- lm(
    ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
    data = nlswork)

  res <- summclust(
    obj = feols_fit,
    params = c("msp", "union"),
    cluster = ~ind_code,
  )

  summary(res)
  tidy(res)
  plot(res)
}
```

---

**summclust.lm**

*Compute Influence and Leverage Metrics for objects of type `lm`*

**Description**

Compute influence and leverage metrics for clustered inference based on the Cluster Jackknife as described in MacKinnon, Nielsen & Webb (2022) for objects of type `lm`.

**Usage**

```
## S3 method for class 'lm'
summclust(obj, cluster, params, type = "CRV3", ...)
```
Arguments

- **obj**
  - An object of type `lm`
- **cluster**
  - A clustering vector
- **params**
  - A character vector of variables for which leverage statistics should be computed.
- **type**
  - "CRV3" or "CRV3J" following MacKinnon, Nielsen & Webb. CRV3 by default
  - other function arguments passed to 'vcov'

Value

An object of type `summclust`, including a CRV3 variance-covariance estimate as described in MacKinnon, Nielsen & Webb (2022)

- **coef_estimates**
  - The coefficient estimates of the linear model.
- **vcov**
  - A CRV3 or CRV3J variance-covariance matrix estimate as described in MacKinnon, Nielsen & Webb (2022)
- **leverage_g**
  - A vector of leverages.
- **leverage_avg**
  - The cluster leverage.
- **partial_leverage**
  - The partial leverages.
- **beta_jack**
  - The jackknifed' leave-on-cluster-out regression coefficients.
- **params**
  - The input parameter vector 'params'.
- **N_G**
  - The number of clusters-
- **call**
  - The `summclust()` function call.
- **cluster**
  - The names of the clusters.

References


Examples

```r
if(requireNamespace("summclust") & requireNamespace("haven")){

library(summclust)
library(haven)

nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
# drop NAs at the moment
nlswork[,-c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
nlswork <- na.omit(nlswork)

lm_fit <- lm(
  ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
  data = nlswork),
  cluster = nlswork[, "clust"]
)

summary(lm_fit)
}
```

"R guide"
tidy.summclust

```r
data = nlswork

res <- summclust(
  obj = lm_fit,
  cluster = ~ind_code,
  params = c("msp", "union")
)

summary(res)
tidy(res)
plot(res)
```

---

**tidy.summclust**

*S3 method to summarize objects of class boottest into tidy data.frame*

**Description**

Obtain results from a `summclust` object in a tidy data frame.

**Usage**

```r
## S3 method for class 'summclust'
tidy(x, ...)
```

**Arguments**

- `x`  
  An object of class 'summclust'
- `...`  
  Other arguments

**Value**

A data.frame containing coefficient estimates, t-statistics, standard errors, p-value, and confidence intervals based on CRV3 variance-covariance matrix and t(G-1) distribution

**References**


**Examples**

```r
if(requireNamespace("summclust") && requireNamespace("haven")){
  library(summclust)
}
```r
library(haven)

nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
# drop NAs at the moment
nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
nlswork <- na.omit(nlswork)

lm_fit <- lm(
  ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
  data = nlswork)

res <- summclust(
  obj = lm_fit,
  params = c("msp", "union"),
  cluster = ~ind_code,
)

tidy(res)
}
```

---

**vcov_CR3J**

*Compute CRV3 covariance matrices via a cluster jackknife as described in MacKinnon, Nielsen & Webb (2022)*

**Description**

Compute CRV3 covariance matrices via a cluster jackknife as described in MacKinnon, Nielsen & Webb (2022)

**Usage**

```r
vcov_CR3J(obj, ...)
```

**Arguments**

- `obj` An object of class `lm` or `fixest` computed?
- `...` misc function argument

**Value**

An object of type `vcov_CR3J`

**References**

vcov_CR3J.fixest

See Also

vcov_CR3J.lm, vcov_CR3J.fixest

Examples

```r
if(requireNamespace("sumclust") & requireNamespace("haven")){

library(sumclust)
library(haven)

nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
# drop NAs at the moment
nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
nlswork <- na.omit(nlswork)

lm_fit <- lm(
  ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
  data = nlswork)

# CRV3 standard errors
vcov <- vcov_CR3J(
  obj = lm_fit,
  cluster = ~ind_code,
  type = "CRV3"
)

# CRV3 standard errors
vcovJN <- vcov_CR3J(
  obj = lm_fit,
  cluster = ~ind_code,
  type = "CRV3J",
)
}
```

vcov_CR3J.fixest

Compute CRV3 covariance matrices via a cluster jackknife as described in MacKinnon, Nielsen & Webb (2022) for objects of type fixest

Description

Compute CRV3 covariance matrices via a cluster jackknife as described in MacKinnon, Nielsen & Webb (2022) for objects of type fixest
Usage

```r
## S3 method for class 'fixest'
vcov_CR3J(
  obj,
  cluster,
  type = "CRV3",
  return_all = FALSE,
  absorb_cluster_fixef = TRUE,
  ...
)
```

Arguments

- `obj`: An object of type `fixest`
- `cluster`: A clustering vector
- `type`: "CRV3" or "CRV3J" following MacKinnon, Nielsen & Webb. CRV3 by default
- `return_all`: Logical scalar, FALSE by default. Should only the vcov be returned (FALSE) or additional results (TRUE)
- `absorb_cluster_fixef`: TRUE by default. Should the cluster fixed effects be projected out? This increases numerical stability.
- `...`: other function arguments passed to `vcov`

Value

An object of class `vcov_CR3J`

References


Examples

```r
if(requireNamespace("summclust")
  && requireNamespace("haven")
  && requireNamespace("fixest")){
    library(summclust)
    library(haven)
    library(fixest)

    nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
    # drop NAs at the moment
    nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
    nlswork <- na.omit(nlswork)
```
feols_fit <- feols(
  ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
  data = nlswork)

# CRV3 standard errors
vcov <- vcov_CR3J(
  obj = feols_fit,
  cluster = ~ind_code,
  type = "CRV3"
)

# CRV3 standard errors
vcovJN <- vcov_CR3J(
  obj = feols_fit,
  cluster =~ind_code,
  type = "CRV3J",
)

---

**vcov_CR3J.lm**  
Compute CRV3 covariance matrices via a cluster jackknife as described in MacKinnon, Nielsen & Webb (2022) for objects of type `lm`

---

**Description**

Compute CRV3 covariance matrices via a cluster jackknife as described in MacKinnon, Nielsen & Webb (2022) for objects of type `lm`

**Usage**

```r
## S3 method for class 'lm'
vcov_CR3J(obj, cluster, type = "CRV3", return_all = FALSE, ...)
```

**Arguments**

- `obj`  
  An object of type `lm`  
- `cluster`  
  A clustering vector  
- `type`  
  "CRV3" or "CRV3J" following MacKinnon, Nielsen & Webb. CRV3 by default  
- `return_all`  
  Logical scalar, FALSE by default. Should only the vcov be returned (FALSE) or additional results (TRUE)  
- `...`  
  other function arguments passed to `vcov`  

**Value**

An object of class `vcov_CR3J`
References


Examples

```r
if(requireNamespace("summclust") && requireNamespace("haven")){

library(summclust)
library(haven)

nlswork <- read_dta("http://www.stata-press.com/data/r9/nlswork.dta")
# drop NAs at the moment
nlswork <- nlswork[, c("ln_wage", "grade", "age", "birth_yr", "union", "race", "msp", "ind_code")]
nlswork <- na.omit(nlswork)

lm_fit <- lm(
  ln_wage ~ union + race + msp + as.factor(birth_yr) + as.factor(age) + as.factor(grade),
  data = nlswork)

# CRV3 standard errors
vcov <- vcov_CR3J(
  obj = lm_fit,
  cluster = ~ind_code,
  type = "CRV3"
)

# CRV3 standard errors
vcovJN <- vcov_CR3J(
  obj = lm_fit,
  cluster = ~ind_code,
  type = "CRV3J",
)
}
```
Index

plot.summclust, 2
summary.summclust, 3
summclust, 4
summclust.fixest, 5, 5
summclust.lm, 5, 7
tidy.summclust, 9
vcov_CR3J, 10
vcov_CR3J.fixest, 11, 11
vcov_CR3J.lm, 11, 13