Package ‘OOI’

December 18, 2020

Type Package
Title Outside Option Index
Version 0.1.0
Description Calculates the Outside Option Index proposed by Caldwell and Danieli (2018) <https://drive.google.com/file/d/1j-uwD19S4qggXIXeYch9jGBCaDhWZlRQ/view>. This index uses the cross-sectional concentration of similar workers across job types to quantify the availability of outside options as a function of workers’ characteristics (e.g. commuting costs, preferences, and skills.)
License GPL-3
Encoding UTF-8
LazyData true
RoxygenNote 7.1.0
Suggests testthat
Imports modi, stats, utils
NeedsCompilation no
Author Elad Guttman [aut, cre], Oren Danieli [aut]
Maintainer Elad Guttman <eladguttman@mail.tau.ac.il>
Repository CRAN
Date/Publication 2020-12-18 09:20:05 UTC

R topics documented:

add_prefix ................................. 2
geo_dist .................................. 2
OOI ......................................... 3
predict.ooi ................................ 5

Index 7
**Add prefix**

Description

Adds a prefix to the column names of a matrix / data.frame.

Usage

```
add_prefix(df, prefix)
```

Arguments

- `df`: a data.frame or a matrix.
- `prefix`: a prefix to be added.

Value

A matrix / data.frame with new column names.

---

**geo_dist**

**Geographical distance**

Description

Calculates geo distance between *two* points.

Usage

```
geo_dist(x.loc, z.loc)
```

Arguments

- `x.loc`: a 2-length vector. The first value is for longitude, the second for latitude.
- `z.loc`: a 2-length vector. The first value is for longitude, the second for latitude.

Value

Distance in miles.
Description

calculates the 'outside option index' (defined as \[- \sum P(Z|X) \times \log(P(Z|X)/P(Z))\]) for workers, using employer-employee data.

Usage

OOI(
  formula = NULL,
  X,
  Z = NULL,
  X.location = NULL,
  Z.location = NULL,
  wgt = rep(1, nrow(X)),
  pred = TRUE,
  method = "logit",
  sim.factor = 1,
  dist.fun = geo_dist,
  dist.order = NULL,
  seed = runif(1, 0, .Machine$integer.max)
)

Arguments

formula  a formula describing the model to be fitted in order to estimate P(Z|X) / P(Z). This formula uses a syntax similar to STATA, and so "x_" refers to all variables with the prefix "x", while "z_" refers to all variables with the prefix "z". Similarly, "d" refers to the distance polynomial (see the example below).
X       matrix or data frame with workers characteristics. Note that all column names should start with "x" (necessary for the inner function 'coef_reshape').
Z       an optional matrix or data frame with jobs characteristics. Note that all column names should start with "z" (necessary for the inner function 'coef_reshape').
X.location an optional matrix or data frame with location for workers. Could be geographical location (i.e., geo-coordinates) or any other feature that can be used in order to measure distance between worker and job using 'dist.fun'. Currently the package supports only numeric inputs.
Z.location same as 'X.location' but for jobs.
wgt      an optional numeric vector of weights.
pred     logical. If TRUE (default), predicts the ooi for the provided data.
sim.factor a variable that determines how much fake data to simulate (relative to real data).
dist.fun  a distance function to calculate the distance between X.location and Z.location. Users interested in using more than one distance metric should provide a function that returns for each row of X.location and Z.location a vector with all the necessary metrics. Also - the function should use columns by their index and not by their names. The default function is `geo_dist`, which is suitable for data with geo-coordinates.

dist.order  a numeric vector specifying for each distance metric an order of the distance polynomial.

seed  the seed of the random number generator.

Value

An "ooi" object. This object is a list containing the following components:

coeffs  coefficients from the estimated logit.

coeffs_sd  coefficients SE.

pseudo_r2  McFadden’s pseudo-R squared for the estimated logit.

standardized_coeffs  standardized coefficients.

ooi  the Outside Option Index.

hhi  the Herfindahl-Hirschman Index, an alternative measure for outside options.

job_worker_prob  the log probability of each worker to work at his *specific* job (rather than to work at a job with his specific z)

orig_arg  a list containing the original arguments (necessary for `predict.ooi`).

Examples

```r
#generate data
#worker and job characteristics:
n <- 100
men <- rbinom(n, 1, 0.5)
size <- 1 + rgeom(n, 0.1)
size[men == 0] <- size[men == 0] + 2
worker_resid <- data.frame(r = round(runif(n, 0, 20), 1))
job_location <- data.frame(l = round(runif(n, 20, 40), 1))

#prepare data
#define distance function:
dist_metric <- function(x, y){abs(y - x)}
X <- data.frame(men = men)
Z <- data.frame(size = size)

#add "x" / "z" to column names:
X <- add_prefix(X, "x.")
Z <- add_prefix(Z, "z.")

#estimate P(Z|X) / P(Z) and calculate the ooi:
ooi_object <- OOI(formula = ~ x_*z_ + x_*d + z_*d, X = X, Z = Z,
                  X.location = worker_resid, Z.location = job_location,
                  sim.factor = 3, dist.fun = dist_metric, dist.order = 3)
```


# we can extract the ooi using predict():
ooi <- predict(ooi_object)
summary(ooi)

## S3 method for class 'ooi'
predict(
  object,
  new.coef = NULL,
  new.X = NULL,
  new.Z = NULL,
  new.X.location = NULL,
  new.Z.location = NULL,
  new.wgt = NULL,
  hhi = FALSE,
  both = FALSE,
  ...
)

Arguments

- **object**: an ooi object.
- **new.coef**: a new *named* vector of coefficients. Check the coefficients produced by the main function to see the right format for this vector.
- **new.X**: a new X matrix / data frame.
- **new.Z**: a new Z matrix / data frame.
- **new.X.location**: a new X.location matrix / data frame.
- **new.Z.location**: a new Z.location matrix / data frame.
- **new.wgt**: a new vector of weights
- **hhi**: whether to predict the HHI (Herfindahl-Hirschman Index, an alternative measure for outside options) instead of the OOI. default is FALSE.
- **both**: whether to return a list with both HHI and OOI when suppling new inputs (default is FALSE). Necessary especially when predicting takes a lot of time.
- **...**: further arguments passed to or from other methods.
Value

If there are no new arguments, returns the original results (ooi/hhi). Otherwise, returns a vector of ooi/hhi (or a list of both) calculated using the new arguments.

Examples

```r
#generate data
#worker and job characteristics:
n <- 100
men <- rbinom(n, 1, 0.5)
size <- 1 + rgeom(n, 0.1)
size[men == 0] <- size[men == 0] + 2
worker_resid <- data.frame(r = round(runif(n, 0, 20), 1))
job_location <- data.frame(l = round(runif(n, 20, 40), 1))

#prepare data
#define distance function:
dist_metric <- function(x, y){abs(y - x)}
X <- data.frame(men = men)
Z <- data.frame(size = size)
#add "x" / "z" to column names:
X <- add_prefix(X, "x.")
Z <- add_prefix(Z, "z.")

#estimate P(Z|X) / P(Z) and calculate the ooi:
ooi_object <- OOI(formula = ~ x_*z_ + x_*d + z_*d, X = X, Z = Z,
                  X.location = worker_resid, Z.location = job_location,
                  sim.factor = 3, dist.fun = dist_metric, dist.order = 3)

#we can extract the ooi using predict():
ooi <- predict(ooi_object)

#or the hhi:
ooi <- predict(ooi_object, hhi = TRUE)

#we can also estimate the ooi with different coefficients:
coeffs <- ooi_object$coeffs
coeffs[indices(coefficients) == "x.men"] <- 0
new_ooi <- predict(ooi_object, new.coef = coeffs)

#or new data:
Z2 <- data.frame(z.size = 1 + rgeom(n, 0.1))
new_ooi <- predict(ooi_object, new.Z = Z2)
```
Index

add_prefix, 2
geo_dist, 2, 4
OOI, 3
predict.ooi, 4, 5