Package 'STE'

July 25, 2022

Type Package

Title Estimate and Study the Strategic Treatment Effects

Version 0.1.0

Authors Vansh Gupta & Jorge Guzman

Maintainer Jorge Guzman <jorge.guzman@columbia.edu>

Description The package allows a user to estimate and study the Strategic Treatment Effect of a strategic choice, as outlined by the paper: Guzman, Jorge, Treatment Effects in Strategic Management (September 1, 2021). Available at SSRN: https://ssrn.com/abstract=3915606 or http://dx.doi.org/10.2139/ssrn.3915606. The five functions in the package allow one to systematically study the strategic treatment effects of a strategic choice by using a Random Forest model to estimate the propensity score (as mentioned in the paper), which is then used to estimate the treatment effects as well as the strategic treatment effects of the strategic choice. This information is then used to estimate the value of coherence for firms.

License MIT + file LICENSE.md

Encoding UTF-8

LazyData true

RoxygenNote 7.2.0

Imports dplyr,
fixest,
glmnet,
randomForest,
stats,
stringr,
fANCOVA

Suggests knitr,
rmarkdown

VignetteBuilder knitr

R topics documented:

`estimate_coherence` *Coherence*

Description

This function calculates the value of coherence for a model.

Usage

```
estimate_coherence(y, x, x.no_inter)
```

Arguments

`y`	Treatment Effect to be studied.
`x`	Variables with interactions.
`x.no_inter`	Variables without interactions.

Value

The value of coherence.

Dependencies

- glmnet

Examples

```
estimate_coherence(
  y = df$teffect,
  x = df[, ml_variables],
  x.no_inter = df[[ml_variables[grep("^[^X]", ml_variables)]]]
  )
```

`estimate_main_effect` *Estimate Main Effect*

Description

This function estimates the main effect of the treatment.

Usage

```
estimate_main_effect(y_var, treatment_var, X, data_df)
```

Arguments

`y_var`	String containing the name of the outcome variable to be studied.
`treatment_var`	String containing the name of treatment variable to be studied.
`X`	X-variables.
`data_df`	The complete dataframe.

Value

The feols regression object from the fixed-effects OLS regression.

Dependencies

- glmnet
- stringr
- fixest
- dplyr

Examples

```
estimate_main_effect(
  y_var = "outcome_var_name",
  treatment_var = "treatment_var_name",
  X = df[, ml_variables],
  data_df = df
  )
```

estimate_propensity *Estimate Propensity Score*

Description

This function estimates the propensity score using a RandomForest model.

Usage

```
estimate_propensity(treatment, X)
```

Arguments

treatment	Treatment to be studied.
X	Complete sample of the x-variables to estimate the propensity score for all the observations in the dataframe.

Value

The propensity scores for each row in our dataset.

Dependencies

- randomForest
- dplyr

Examples

```
estimate_propensity(
  treatment = df$treatment,
  X = df[[ml_variables]]
  )
```

estimate_ste *Estimate the Strategic Treatment Effect.*

Description

This function estimates the strategic treatment effect.

Usage

```
estimate_ste(y, treatment, propensity, df)
```

Arguments

y	Outcome variable to be studied.
treatment	Treatment variable.
propensity	Propensity score generated earlier.
df	The complete dataset.

Value

df The same as input dataframe with the added column of the strategic treatment effect.

Dependencies

- dplyr
- fANCOVA

Examples

```
estimate_ste(
   y = df$equity_growth,
   treatment = df$treatment,
   propensity = p_scores,
   df = df
)
```

---

get_top_ste_determinants

Study Determinants of the STE.

Description

This function studies the determinants of the strategic treatment effect.

Usage

```
get_top_ste_determinants(X, teffect)
```

Arguments

X	x-variables for the LASSO.
teffect	Treatment effect of the model.

Value

ste_features A dataframe containing the STE features.

Dependencies

- glmnet
- dplyr

Examples

```
get_top_ste_determinants(
   X = df[, ml_variables],
   teffect = df$teffect
)
```

Index