

Package ‘Foodpricev2’

June 30, 2024

Type Package

Title Paquete en R para el análisis de asequibilidad de dietas de costo mínimo.

Version 1.0.0

Author Daniel González, Sergio A. Barona, Juan Camilo Ordoñez

Maintainer Juan Camilo Ordoñez <kmilinm19@gmail.com>

Depends R (>= 4.3.2)

Description Este paquete de R proporciona herramientas y funciones para analizar la asequibilidad de dietas de costo mínimo, permitiendo a los usuarios evaluar los gastos dietéticos en relación con el ingreso del hogar.

License MIT

Encoding UTF-8

LazyData true

NeedsCompilation No

URL <https://github.com/Foodpricev2>

Date 2024-02-26

Contents

Afford	1
HCost	3
IncomeCol	5
Index	7

Afford	<i>Cálculo de Indicadores de Asequibilidad de Dietas de Costo Mínimo</i>
--------	--

Description

La función Afford calcula indicadores de asequibilidad para tres tipos de dietas de costo mínimo, utilizando los resultados generados por la función HCost y los datos de gasto de hogares proporcionados en el data frame Hexpanse. Hexpanse debe contener las siguientes columnas: ‘id_hogar’ (identificador único del hogar), ‘ung’ (número de miembros del hogar), ‘income’ (ingreso mensual total del hogar) y ‘food_exp’ (gasto mensual en alimentos y bebidas no alcohólicas). Por otro lado, HCost es el resultado de la función HCost y debe contener información detallada sobre los modelos CoCA, CoNA y CoRD estimados para cada hogar representativo, incluyendo la composición por sexo y grupos demográficos de los miembros del hogar.

Usage

```
Afford(
  Hexpense,
  Model_CoCA = NULL,
  Model_CoNA = NULL,
  Model_CoRD = NULL
)
```

Arguments

Hexpense: Data frame con los datos de gasto de hogares, incluyendo 'id_hogar', 'ung', 'income' y 'food_exp'.

Model_CoCA: Data frame opcional con los resultados del modelo CoCA generado por la función HCost.

Model_CoNA: Data frame opcional con los resultados del modelo CoNA generado por la función HCost.

Model_CoRD: Data frame opcional con los resultados del modelo CoRD generado por la función HCost.

Details

La función Afford combina los datos de gasto de hogares (Hexpense) con los resultados de los modelos CoCA, CoNA y CoRD (si están disponibles) para calcular indicadores de asequibilidad. Estos indicadores ayudan a evaluar la capacidad de los hogares para alcanzar dietas de costo mínimo recomendadas, considerando variables como el ingreso total, el gasto en alimentos y el tamaño del hogar.

Value

Devuelve una lista con dos elementos:

- 'Poverty_outcome': Resultados de los indicadores de asequibilidad calculados para los modelos CoCA, CoNA y CoRD, incluyendo tasas, brechas y severidades por decil.
- 'Mean_income_food': Promedios y ratios de ingreso y gasto en alimentos por decil, junto con los umbrales calculados para cada tipo de dieta.

Note

Es esencial asegurarse de que los datos proporcionados a la función Afford estén correctamente estructurados según lo especificado para Hexpense y HCost, tal como se describe en la documentación de Foodprice v.1.0.0.

Examples

```
# Ejemplo de uso:
datos <- Foodpricev2::IncomeCol(Mes=1, Año=2022, Ciudad="Cali")

# Cargar datos previamente calculados
datos_hcost <- HCost(Year = 2022, City = "Cali", Month = 6,
Household = Household, ERR = EER, EER_LL = ER_LL, UL = UL,
Serv = serv, Diverse = diverse)

# Calcular indicadores de asequibilidad
```

```
resultados <- Afford(Hexpense = datos_h, Model_CoCA = datos_hcost$Model_CoCA,
  Model_CoNA = datos_hcost$Model_CoNA, Model_CoRD = datos_hcost$Model_CoRD)
```

HCost	<i>Estimación de Métricas de Costo Mínimo para Dietas en Hogares</i>
-------	--

Description

Función diseñada para estimar métricas basadas en tres tipos de dietas de costo mínimo para un hogar representativo, utilizando datos proporcionados por el paquete Foodprice. Esta función integra las funciones CoCA(), CoNA() y CoRD() del paquete Foodprice v.1.0.0 (González et al., 2024), unificándolas en un solo proceso mediante los parámetros especificados.

Usage

```
HCost (
  Month = NULL,
  Year = NULL,
  City = NULL,
  Household,
  Data = NULL,
  ERR = NULL,
  EER_LL = NULL,
  UL = NULL,
  Serv = NULL,
  Diverse = NULL
)
```

Arguments

Month:	Valor numérico que indica el mes para el cual se realizará la estimación. Debe ser un valor entre 1 y 12.
Year:	Valor numérico que indica el año para el cual se realizará la estimación. Debe estar comprendido entre 2022 y 2023.
City:	Cadena de caracteres que especifica la ciudad para la cual se realizará la estimación. Actualmente, el paquete funciona para las 13 principales ciudades y áreas metropolitanas del país: Bogotá D.C., Medellín A.M., Cali A.M., Barranquilla A.M., Bucaramanga A.M., Manizales A.M., Pereira A.M., Cúcuta A.M., Pasto, Ibagué, Montería, Cartagena y Villavicencio.
Household:	Data frame con tres columnas que definen la composición del hogar representativo según sexo y edad de sus miembros. La columna ‘Person’ indica el identificador de la persona, ‘Sex’ especifica el sexo de la persona (0=Hombre, 1=Mujer), y ‘Demo_Group’ define el grupo demográfico al que pertenece, con los mismos niveles utilizados en las salidas de los modelos CoCA, CoNA y CoRD para una implementación correcta (consulte la documentación del paquete Foodprice para más información).
Data:	Data frame que contiene información detallada sobre alimentos locales disponibles, incluyendo nombre, grupo/subgrupo, tamaño de porción comestible, precios, contenido nutricional, etc.

ERR:	Data frame con requerimientos de energía estimados para cada grupo demográfico del hogar representativo.
EER_LL:	Data frame que incluye niveles mínimos de ingesta requeridos para nutrientes específicos según grupo demográfico.
UL:	Data frame que especifica niveles máximos de ingesta recomendados para nutrientes según grupo demográfico.
Serv:	Data frame con recomendaciones de porciones diferenciadas por grupos de alimentos y demográficos dentro del hogar representativo.
Diverse:	Data frame que indica el número de alimentos seleccionados por grupo de alimentos según la estructura definida en Data.

Details

La función requiere únicamente el parámetro obligatorio 'Household'. Si dispone de datos con estimaciones de alimentos, grupos y composición nutricional, puede utilizarlos mediante el parámetro 'Data'. Para usar estimaciones específicas de Colombia utilizando la función 'DataCol()' del paquete Foodprice, simplemente ingrese los parámetros 'Month', 'Year' y 'City'. Además, la función es flexible y permite calcular solo los modelos pertinentes:

- Si omite el parámetro 'ERR', la función no calculará el costo de la dieta CoCA.
- Si omite alguno de los parámetros 'EER_LL' o 'UL', la función no calculará el costo de la dieta CoNA.
- Si omite alguno de los parámetros 'Serv' o 'Diverse', la función no calculará el costo de la dieta CoRD.

Value

Devuelve una lista con data frames (dependiendo del número de modelos que requiera ejecutar): Model_CoCA, Model_CoNA, Model_CoRD. En general, los anteriores data frames tienen las siguientes columnas:

- cost_day: Costo diario estimado para la dieta.
- total_household: Costo total estimado para el hogar.
- per_capita: Costo per cápita estimado para la dieta.
- per_capita_year: Costo anual per cápita estimado para la dieta.
- per_capita_month: Costo mensual per cápita estimado para la dieta.

Note

Es crucial asegurarse de proporcionar correctamente todos los parámetros necesarios para la función 'HCoSt', según la estructura y formatos detallados en la documentación de Foodprice v.1.0.0.

Examples

```
# Ejemplo de uso :

# cargando datos de Foodprice
library(Foodprice)

datos_h=HCoSt <- Foodpricev2::HCoSt(Year = 2022, City = "Cali", Month=6,
Household=Household, ERR = EER, EER_LL = EER_LL, UL = UL,
Serv = serv,Diverse=diverse)
```

Description

Función diseñada para calcular los ingresos actuales de individuos basándose en datos de la Encuesta Integrada de Hogares (GEIH). Además, utiliza información de la Encuesta de Calidad de Vida (ECV) para calcular las proporciones del ingreso destinadas al gasto en alimentos y bebidas no alcohólicas, diferenciadas por niveles de ingreso.

Usage

```
IncomeCol (
    Month,
    Year,
    City
)
```

Arguments

Mes :	Valor numérico que indica el período mensual para el cual se desean calcular los ingresos. Debe ser un valor entre 1 y 12.
Año :	Valor numérico que indica el año de los datos a cargar. Debe estar comprendido entre 2022 y 2023.
Ciudad :	Cadena de caracteres que especifica la ciudad para la cual se calcularán los ingresos. Actualmente, el paquete funciona para las 13 principales ciudades y áreas metropolitanas del país, a saber: Bogotá D.C., Medellín A.M., Cali A.M., Barranquilla A.M., Bucaramanga A.M., Manizales A.M., Pereira A.M., Cúcuta A.M., Pasto, Ibagué, Montería, Cartagena y Villavicencio.

Details

La función carga en total seis módulos. El módulo 0 se encarga de descargar y cargar las librerías necesarias. El módulo 1 se encarga de descargar y cargar los datos de la Encuesta Integrada de Hogares (GEIH) para el mes y año asignados. El módulo 2 aplica un algoritmo para el cálculo de los ingresos corrientes mensuales de las unidades de gasto. El módulo 3 aplica factores de expansión a los hogares. El módulo 4 descarga y carga los datos de la Encuesta de Calidad de Vida (ECV), con la cual se calcularán las proporciones del gasto en alimentos y bebidas no alcohólicas. Finalmente, el módulo 5 concluye con los cálculos de los ingresos.

Value

Devuelve un conjunto de datos con las siguientes columnas:

- deciles: Factor que indica el decil al que pertenece cada fila o registro.
- household_id: Identificador único del hogar.
- ung: Tamaño del hogar.
- income: Ingresos totales del hogar.
- per_capita_income: Ingresos per cápita calculados como $\text{income} / \text{ung}$.

- share: Proporción de ingresos destinada a la alimentación.
- food_exp: Gasto en alimentación calculado como `'income * share'`.
- food_exp_per_capita: Gasto per cápita en alimentación calculado como `'food_exp / ung'`.
- food_exp_per_capita_year: Gasto anual per cápita en alimentación calculado como `'food_exp_per_capita * 12'`.

Note

Es pertinente asegurarse de especificar correctamente los parámetros de entrada (Mes, Año, Ciudad) para cargar los datos relevantes de GEIH.

Examples

```
# Ejemplo de uso :  
  
datos <- Foodpricev2::IncomeCol(Month=1, Year=2022, City="Cali")  
head(datos_ingresos)
```

Index

Afford, [1](#)

HCost, [3](#)

IncomeCol, [5](#)