

# Documentación del Transformador de Privacidad Diferencial

Luisa María De La Hortúa\*, David Moreno†, Allan Ramírez‡, and David Romero§

Master's student MIIA, Industrial Engineering Department, Universidad de Los Andes, Bogotá,  
Colombia

## Contents

|          |  |          |
|----------|--|----------|
| <b>1</b> | <b>Introducción</b>                                | <b>2</b> |
| <b>2</b> | <b>Descripción General del Paquete</b>             | <b>2</b> |
| <b>3</b> | <b>Estructura del Paquete</b>                      | <b>2</b> |
| <b>4</b> | <b>Clase DifferentialPrivacyTransformer</b>        | <b>2</b> |
| 4.1      | Inicialización . . . . .                           | 2        |
| 4.2      | Métodos Principales . . . . .                      | 3        |
| 4.2.1    | fit_quantitative . . . . .                         | 3        |
| 4.2.2    | fit_categorical . . . . .                          | 3        |
| 4.2.3    | calculate_utility_metrics . . . . .                | 4        |
| 4.2.4    | Métodos de Graficación . . . . .                   | 4        |
| <b>5</b> | <b>Funciones de Graficación</b>                    | <b>4</b> |
| 5.1      | plot_mean_vs_privatized . . . . .                  | 4        |
| 5.2      | plot_histograms . . . . .                          | 4        |
| <b>6</b> | <b>Mecanismos de Privacidad Diferencial</b>        | <b>4</b> |
| 6.1      | Para Datos Cuantitativos . . . . .                 | 4        |
| 6.2      | Para Datos Categóricos . . . . .                   | 5        |
| <b>7</b> | <b>Ejemplos de Uso</b>                             | <b>5</b> |
| 7.1      | Privatización de Variables Cuantitativas . . . . . | 5        |
| 7.2      | Privatización de Variables Categóricas . . . . .   | 5        |
| <b>8</b> | <b>Notas Finales</b>                               | <b>6</b> |

---

\*Email: lm.delahortua@uniandes.edu.co

†Email: ds.morenom1@uniandes.edu.co

‡Email: as.ramirez2@uniandes.edu.co

§Email: ds.romerog1@uniandes.edu.co

# 1 Introducción

La privacidad diferencial es un marco matemático que permite el análisis de datos sensibles garantizando la privacidad individual. Este documento presenta un transformador de privacidad diferencial que facilita la aplicación de diferentes mecanismos de privacidad a conjuntos de datos, tanto para variables cuantitativas como categóricas. El objetivo es proporcionar una herramienta flexible y fácil de usar para investigadores y profesionales que deseen proteger la privacidad en sus análisis de datos.

## 2 Descripción General del Paquete

El paquete proporciona una clase principal, `DifferentialPrivacyTransformer`, que actúa como interfaz para aplicar mecanismos de privacidad diferencial a un conjunto de datos. Además, incluye funciones para graficar y analizar el impacto de la privatización en los datos, así como implementaciones de varios mecanismos de privacidad diferencial para datos cuantitativos y categóricos.

## 3 Estructura del Paquete

El paquete está organizado de la siguiente manera:

- **dp\_transformer.py**: Contiene la clase `DifferentialPrivacyTransformer`.
- **dp\_plotting.py**: Incluye funciones para la visualización y análisis de datos privatizados.
- **dp\_mechanisms**: Directorio que alberga los mecanismos de privacidad diferencial.
  - **quantitative.py**: Implementaciones de mecanismos para datos cuantitativos.
  - **categorical.py**: Implementaciones de mecanismos para datos categóricos.

## 4 Clase `DifferentialPrivacyTransformer`

La clase `DifferentialPrivacyTransformer` es la interfaz principal para aplicar mecanismos de privacidad diferencial a un conjunto de datos de pandas. Permite transformar variables cuantitativas y categóricas utilizando diferentes mecanismos y ajustar el nivel de privacidad mediante el parámetro  $\epsilon$ .

### 4.1 Inicialización

```
DifferentialPrivacyTransformer(  
    df,  
    epsilon_quantitative=1.0,  
    epsilon_categorical=1.0,  
    quantitative_vars=None,  
    categorical_vars=None  
)
```

#### Parámetros:

- **df** (*pd.DataFrame*): `DataFrame` que contiene los datos a transformar.
- **epsilon\_quantitative** (*float*): Presupuesto de privacidad para variables cuantitativas (valor por defecto: 1.0).
- **epsilon\_categorical** (*float*): Presupuesto de privacidad para variables categóricas (valor por defecto: 1.0).
- **quantitative\_vars** (*list de str*): Lista de nombres de variables cuantitativas. Si es `None`, se detectan automáticamente.

- `categorical_vars` (*list de str*): Lista de nombres de variables categóricas. Si es `None`, se detectan automáticamente.

## 4.2 Métodos Principales

### 4.2.1 `fit_quantitative`

Aplica mecanismos de privacidad diferencial a las variables cuantitativas seleccionadas.

```
fit_quantitative(
    quantitative_vars=None,
    method='duchi',
    epsilon=None
)
```

#### Parámetros:

- `quantitative_vars` (*list de str*): Variables cuantitativas a transformar. Si es `None`, se utilizan las variables definidas en la instancia.
- `method` (*str*): Método de privatización a utilizar. Opciones disponibles:
  - 'duchi'
  - 'laplace'
  - 'piecewise'
  - 'multidimensional\_duchi'
  - 'multidimensional'
- `epsilon` (*float*): Presupuesto de privacidad. Si es `None`, se utiliza el valor definido en la instancia.

### 4.2.2 `fit_categorical`

Aplica mecanismos de privacidad diferencial a las variables categóricas seleccionadas.

```
fit_categorical(
    categorical_vars=None,
    method='direct_encoding',
    epsilon=None
)
```

#### Parámetros:

- `categorical_vars` (*list de str*): Variables categóricas a transformar. Si es `None`, se utilizan las variables definidas en la instancia.
- `method` (*str*): Método de privatización a utilizar. Opciones disponibles:
  - 'direct\_encoding'
  - 'oue' (Optimized Unary Encoding)
  - 'rappor'
- `epsilon` (*float*): Presupuesto de privacidad. Si es `None`, se utiliza el valor definido en la instancia.

### 4.2.3 calculate\_utility\_metrics

Calcula métricas de utilidad para evaluar el impacto de la privatización en las variables cuantitativas.

```
calculate_utility_metrics(  
    variables=None  
)
```

**Parámetros:**

- **variables** (*list de str*): Variables cuantitativas para calcular las métricas. Si es `None`, se utilizan las variables definidas en la instancia.

### 4.2.4 Métodos de Graficación

**plot\_mean\_vs\_privatized** Genera gráficos que comparan la media de las variables originales con las medias de las variables privatizadas para diferentes valores de  $\epsilon$ .

```
plot_mean_vs_privatized(  
    variables=None,  
    method='duchi',  
    epsilons=None,  
    **kwargs  
)
```

**plot\_histograms** Genera histogramas comparativos de las variables originales y privatizadas.

```
plot_histograms(  
    variables=None,  
    method='duchi',  
    epsilon=1.0,  
    bins=30,  
    **kwargs  
)
```

## 5 Funciones de Graficación

### 5.1 plot\_mean\_vs\_privatized

Genera gráficos que muestran cómo varía la media de las variables privatizadas con respecto a diferentes valores de  $\epsilon$ .

### 5.2 plot\_histograms

Genera histogramas que comparan la distribución de las variables originales con las privatizadas.

## 6 Mecanismos de Privacidad Diferencial

### 6.1 Para Datos Cuantitativos

Los mecanismos para datos cuantitativos se implementan en `dp_mechanisms/quantitative.py`.

- **Mecanismo de Duchi et al.** ('duchi'): Adecuado para datos en el rango  $[-1, 1]$ , utiliza respuesta aleatorizada para garantizar la privacidad.
- **Mecanismo de Laplace** ('laplace'): Agrega ruido Laplaciano proporcional a la sensibilidad del dato.

- **Mecanismo por Tramos** ('piecewise'): Divide el dominio en tramos y aplica ruido de acuerdo con la posición del dato.
- **Mecanismo Multidimensional de Duchi** ('multidimensional\_duchi'): Extensión del mecanismo de Duchi para datos multidimensionales.
- **Mecanismo Multidimensional Personalizado** ('multidimensional'): Aplica mecanismos unidimensionales de forma selectiva en dimensiones aleatorias.

## 6.2 Para Datos Categóricos

Los mecanismos para datos categóricos se implementan en `dp_mechanisms/categorical.py`.

- **Codificación Directa** ('direct\_encoding'): Asigna probabilidades a cada categoría y aplica respuesta aleatorizada.
- **Codificación Unaria Optimizada (OUE)** ('oue'): Representa categorías mediante codificación unaria y aplica perturbación optimizada.
- **RAPPOR** ('rappor'): Utiliza codificación unaria y múltiples rondas de perturbación para mejorar la privacidad.

## 7 Ejemplos de Uso

### 7.1 Privatización de Variables Cuantitativas

```
import pandas as pd
from dp_transformer import DifferentialPrivacyTransformer

# Cargar el conjunto de datos
df = pd.read_csv('data.csv')

# Definir variables cuantitativas
quantitative_vars = ['edad', 'ingreso']

# Crear instancia del transformador
dp_transformer = DifferentialPrivacyTransformer(
    df,
    epsilon_quantitative=1.0,
    quantitative_vars=quantitative_vars
)

# Aplicar privatización usando el mecanismo de Duchi
df_priv = dp_transformer.fit_quantitative(method='duchi')

# Mostrar las primeras filas del DataFrame transformado
print(df_priv.head())
```

### 7.2 Privatización de Variables Categóricas

```
# Definir variables categóricas
categorical_vars = ['sexo', 'ocupacion']

# Crear instancia del transformador
```

```

dp_transformer = DifferentialPrivacyTransformer(
    df,
    epsilon_categorical=1.0,
    categorical_vars=categorical_vars
)

# Aplicar privatización usando codificación directa
df_priv = dp_transformer.fit_categorical(method='direct_encoding')

# Mostrar las primeras filas del DataFrame transformado
print(df_priv.head())

```

## 8 Notas Finales

- **Flexibilidad:** El paquete permite aplicar diferentes mecanismos y ajustar los niveles de privacidad según las necesidades del usuario.
- **Facilidad de Uso:** La interfaz de alto nivel simplifica el proceso de privatización, haciendo que sea accesible incluso para usuarios sin experiencia en privacidad diferencial.
- **Análisis Integrado:** Las funciones de graficación y cálculo de métricas facilitan la evaluación del impacto de la privatización en la utilidad de los datos.
- **Extensibilidad:** El diseño modular permite agregar nuevos mecanismos y funcionalidades en el futuro.