Preparación y limpieza de datos con Python

Datapath

Sobre mi

Ingeniera de telecomunicaciones y Científica de datos

Me gusta la fotografía, leer y programar

Creadora de "Al mal tiempo, buena data", blog en Medium sobre ciencia de datos

pyladies Medellín

DATA SCIENCE FEM

Agenda de hoy

1. Introducción
2. Metodología CRISP-DM
3. Limpieza de datos
4. Práctica con Python

Limpieza de datos

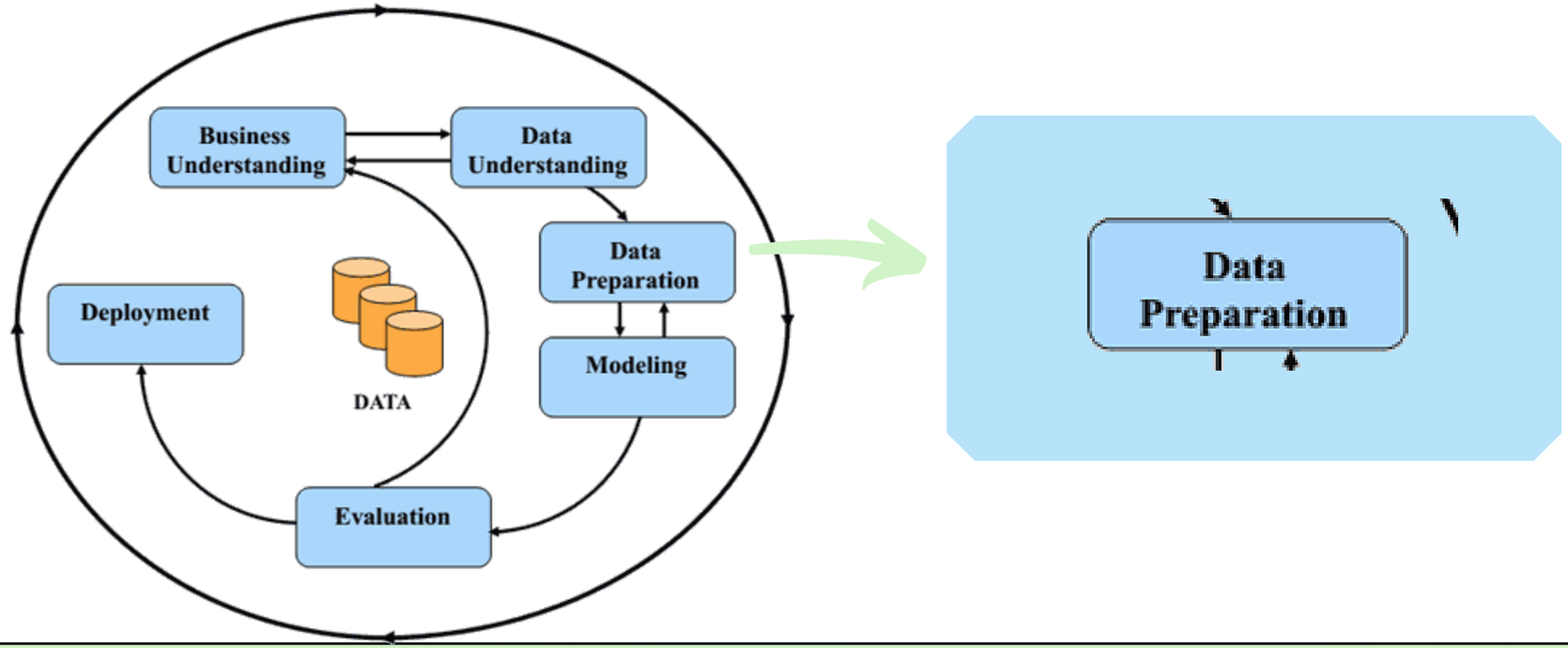
Garbage in, garbage out

1. ¿Qué es para ti limpieza de datos?
2. ¿Cuales herramientas has utilizado?
3. ¿Cuándo crees que se debe usar?

"This is not what I meant when I said 'we need better data cleansing!'"

Tomado de: https://towardsdatascience.com/5-steps-of-a-data-science-project-lifecycle-26c50372b492

CRISP-DM


Business Understanding
Data Understanding
Data Preparation
Modeling
Evaluation
Deployment
DATA
Data Preparation

Preparación de datos

Análisis de datos

Limpieza de datos

- Unión de varias fuentes de datos
- Tipos de datos, rangos y duplicados
- Uniformidad y completitud
- Problemas de datos numéricos y categóricos

Manos a la Obra

Recordemos

Tipos de datos:

FLOAT

OBJECT

CATEGORY

INTEGER

CATEGORY

DATETIME

```
#Validar tipo de dato
df.dtypes

#Cambiar tipo de dato
df['<nombre_columna>'].astype('<tipo_dato>')
```

Recordemos

Fechas

```
#Convertir a Datetime
datetime.strptime(df['<columna_fechas>'], <formato>)

#Obtener fecha actual
fecha_actual = dt.date.today()

#Si la columna es tipo dato datetime
df['año'] = df['fecha'].dt.year #Obtener año
df['mes'] = df['fecha'].dt.month #Obtener mes
```

Duplicados

```
#Visualizar valores duplicados
df[df.duplicated()] #Opción 1
df[df.duplicated(subset = [<nombres_columnas>], keep = False)] #Opción 2

#Eliminar duplicados
df.drop_duplicates(inplace = True)
```

Variables categoricas

```
#Visualizar valores no definidos
df['<col_categorica>'] = pd.cut(df['<col_num>'], bins= <rango_div>,labels = <nombres_div>)

#Cambiar valores
df['<columna>'] = df['<columna>'].replace({<valor_anterior>:<valor_nuevo>})

#Eliminar espacios
df['<columna>'].str.strip()

#Pasar a minúsculas
df['<columna>'] = df['<columna>'].str.lower()

#Pasar a mayúsculas
df['<columna>'] = df['<columna>'].str.upper()
```

Valores no definidos

```
#Visualizar valores no definidos
inconsistentes = set(df1['<nombre_columna>']).difference(df2['<nombre_columna>'])

#Eliminar valores no definidos
df = df[~df['<columna>'].isin(inconsistentes)]
```

Recordemos

¿Qué podemos hacer con los datos nulos?
Eliminarlos o reemplazarlos

Datos nulos

Estos valores pueden afectar el rendimiento y la capacidad de predicción de nuestros modelos. ¿Cómo podemos identificarlos?

- NaN
- None
- 0
- Null
- Na
- Not Available

Media, moda, mediana

Imputar valores con ML

valor anterior o valor siguiente

Recordemos

¿Qué podemos hacer con los datos nulos?
Eliminarlos o reemplazarlos

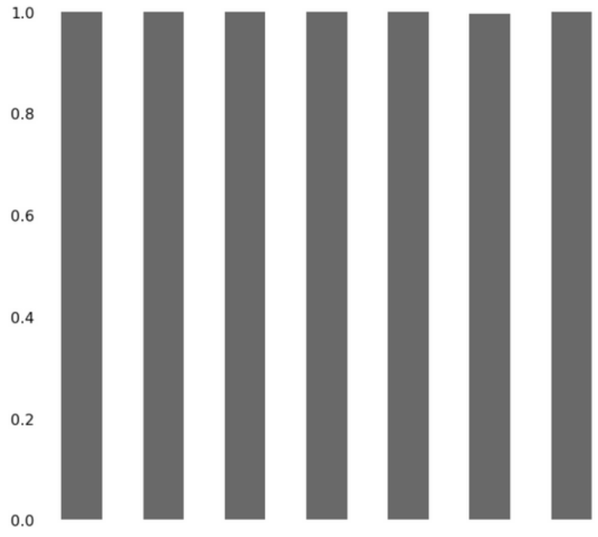
Datos nulos

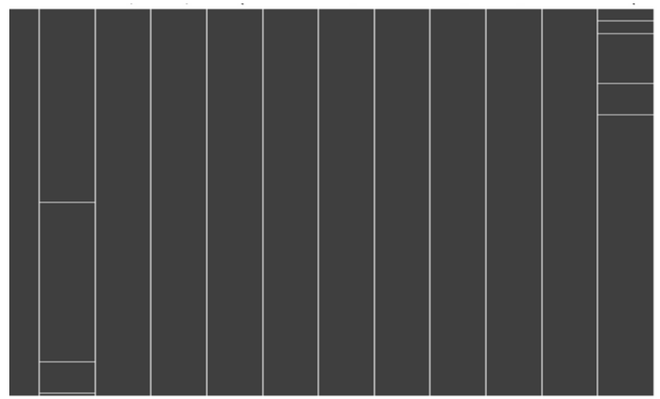
```python
#Visualizar cantidad de valores nulos
df.isna().sum()

#Graficar valores nulos
import missingno as msno
msno.bar(df) #Opción 1
msno.matrix(android_games) #Opción 2

#Eliminar valores nulos
df.dropna()

#Reemplazar valores nulos
df.fillna(value=values, inplace=True)
```

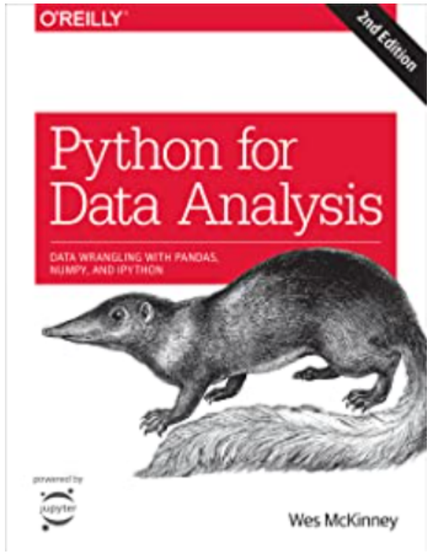




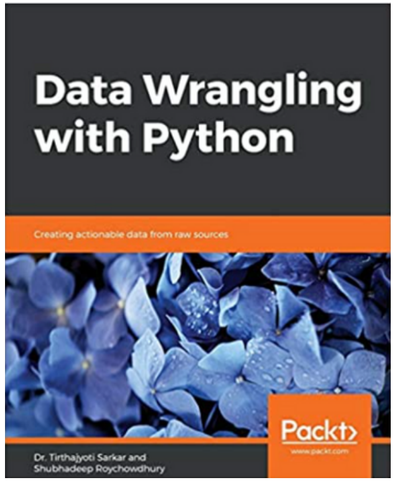
Recursos

1. Documentación oficial de las librerias
2. Curso Cleaning Data in Python en DataCamp

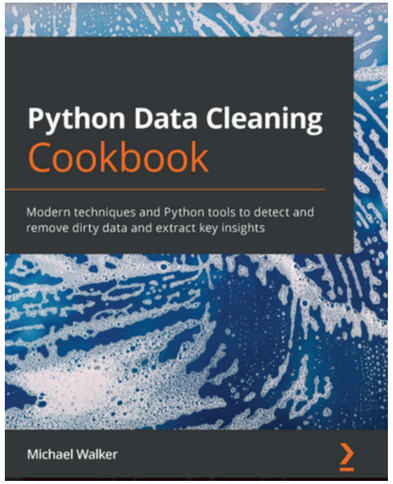
Libros



Libros



Libros



Medium

Limpieza de datos con Python

Al mal tiempo, buena data Jun 8 · 8 min read

"Exploratory data analysis can never be the whole story, but nothing else can serve as the foundation stone"

– John Tukey –

¡Gracias!

@lauralpezb