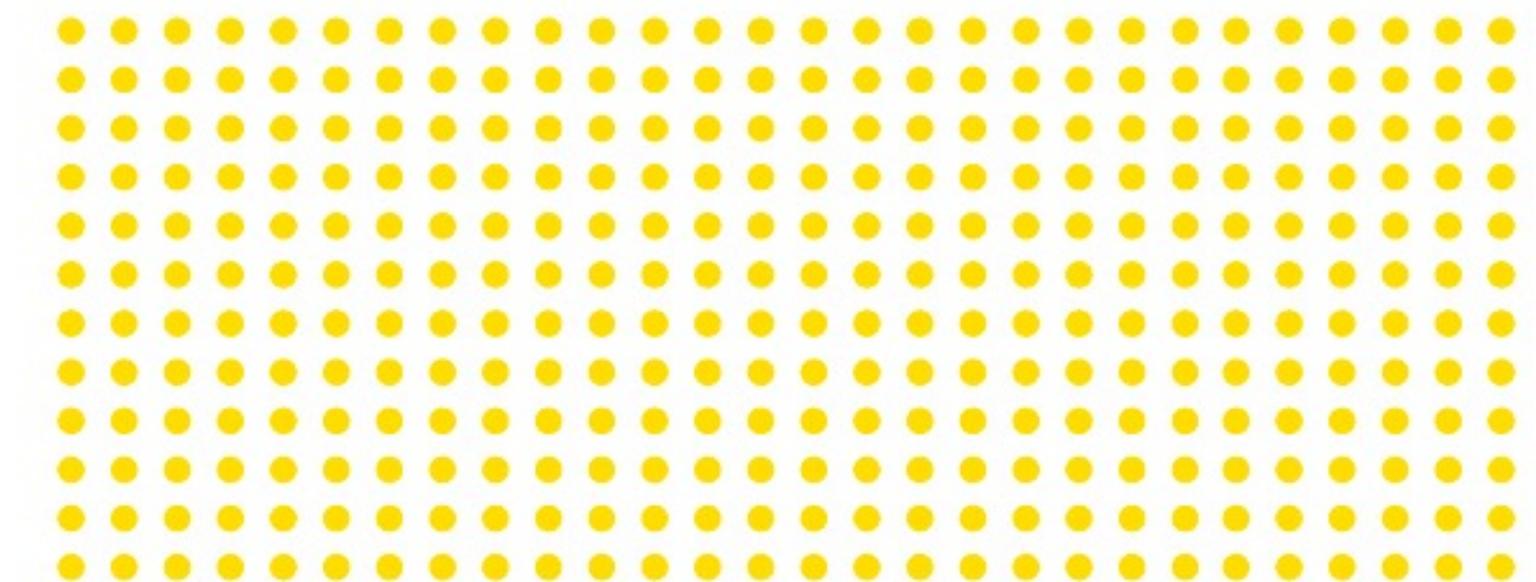




Universidad de
los Andes

**Educación
Continua**
Vicerrectoría Académica





Contenido

- Presentacion
- Introducción
- Limpieza de datos con SQL
- Limpieza de datos con Python



Docentes del curso

Wilfredy Santamaría, MsC

Docente y emprendedor

Departamento de Ing. de Sistemas y Computación

Universidad de los Andes

w.santamaria@uniandes.edu.co

Ingeniero de Sistemas de la Universidad Nacional de Colombia (2003) y Magíster en Ingeniería de Sistemas y Computación de la misma Universidad (2010) con énfasis en Inteligencia Artificial, Machine Learning y Minería de Datos.

Experto en la aplicación de lineamientos de Data Management y CRISP-DM, Diseño de Bases de Datos e Inteligencia de Negocios. De igual manera se ha desempeñado como ingeniero de desarrollo y cuenta con amplia experiencia en Análisis, Diseño y Construcción de Software. Tengo más de 15 años de experiencia en la gestión de proyectos de TI.



1. Modelo conceptual -
Modelo E/R

2. Modelo lógico -
Modelo relacional

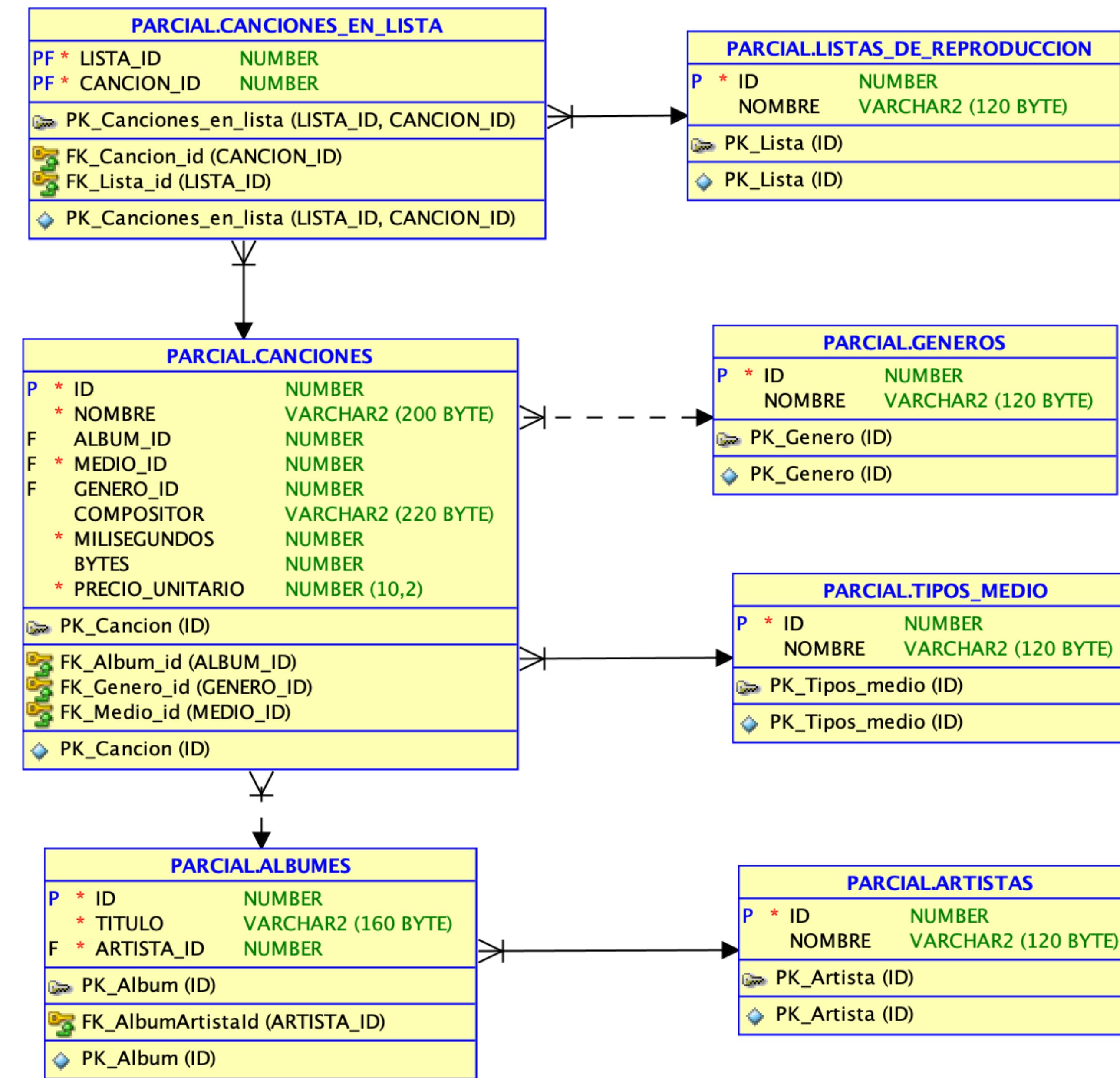
3. Calidad de los datos -
Normalización

4. Lenguaje de procesamiento de datos SQL

5. Técnicas de procesamiento y limpieza de datos

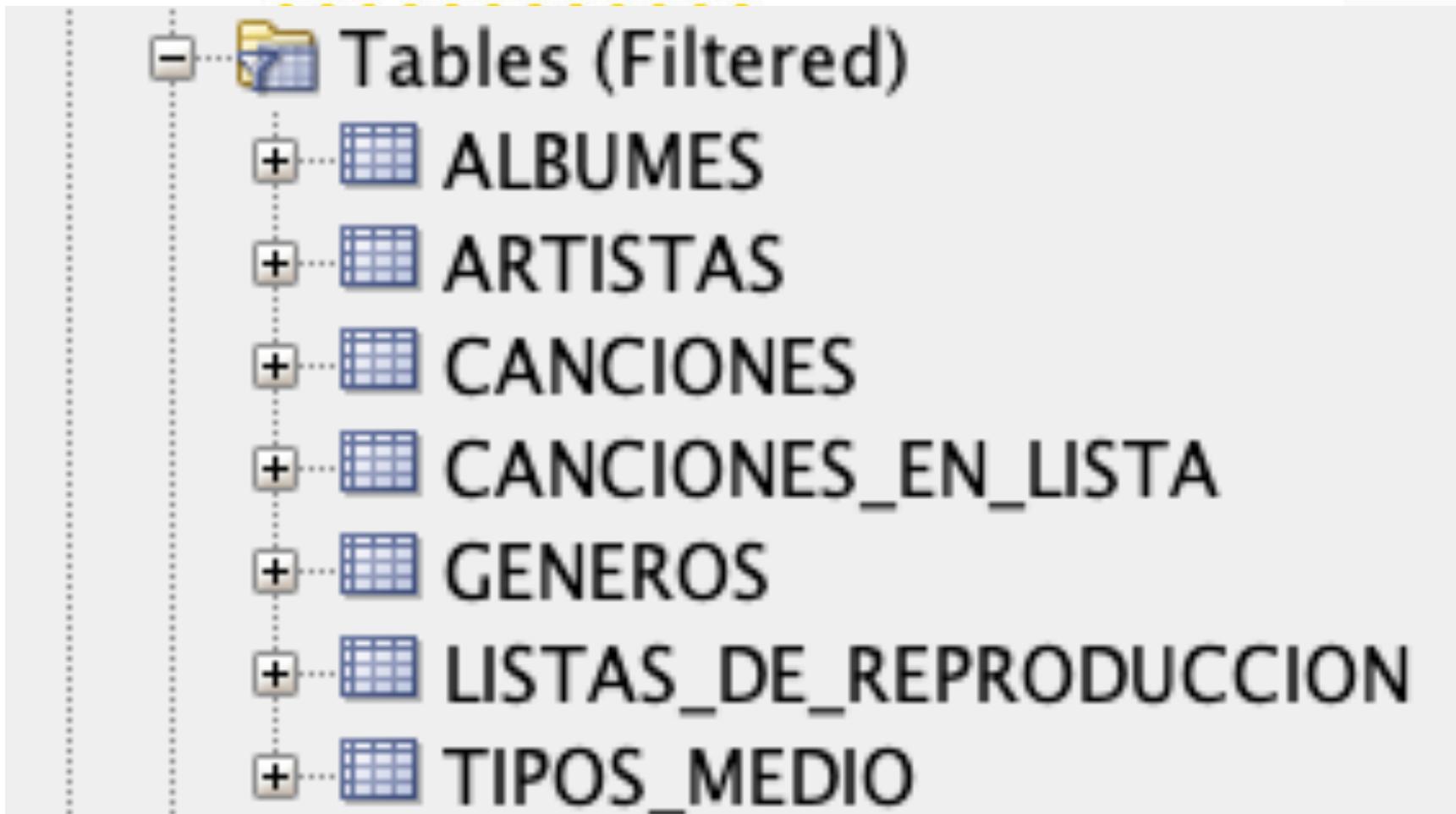
Impartirá el último módulo - 4 horas
31 de Mayo

Musik: caso de estudio del curso



Preparación del Ambiente- Musik

Crear una replica de la BD MusiK en cada cuenta de usuario

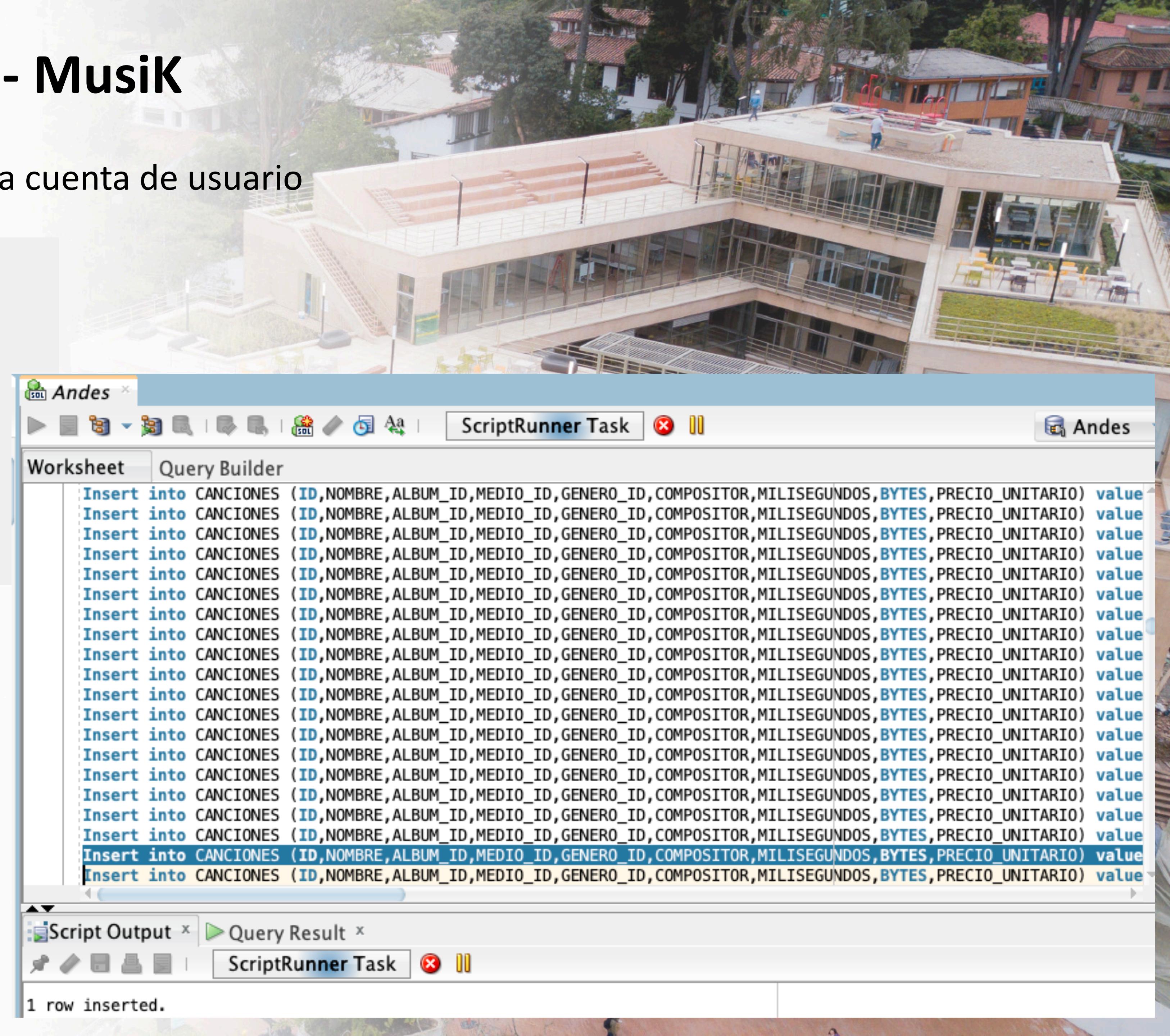


Ejecutar los siguientes archivos

1.bdalbunesLimpieza-object.sql

2.bdalbunesLimpieza data.sql

3.bdalbunesLimpieza-constraints.sql



Introducción – Limpieza de datos



La limpieza de datos es un **proceso** que implica **identificar y corregir errores, inconsistencias y valores faltantes** en los datos.

Un conjunto de datos limpio y preciso es esencial para obtener resultados confiables en sus análisis.



Introducción - Calidad de los Datos

Condición de un conjunto de valores de datos, que asegura que sean precisos, completos, confiables y relevantes para el propósito que se les pretende dar.

A nivel de base de datos se puede manejar:

Preventiva

- Diseño - Normalización(evitar redundancias)
- Restricciones de Integridad(FK) - CK(Constraint de verificación)

Correctiva

Depurar, corregir, estandarizar, consolidar



Introducción – Tipos de Atributos

Tipo Atributo	Descripcion	Ejemplo	Operaciones
Categorico (Cualitativos)	Nominal	Son simplemente nombres diferentes que proveen información suficiente para distinguir un objeto de otro.	nombre, genero, color de ojo, tipo sangre, etc
	Ordinal	Proporcionar suficiente información para ordenar un objeto. Tienen un orden o clasificación inherente entre las categorías.	Encuesta de satisfaccion cliente(Muy satisfecho,Satisfecho, Regular, Malo), grados(primer, Segundo, etc)
Numericos (Cuantitativos)	Intervalo	Para intervalos de atributos, existe una unidad de medida	Calendario de fechas, temperatura en celcius
	Radio	Valores con parte entera y decimal	Masa, valores monetarios,etc

Introducción – Técnicas de Limpieza



Deduplicación

- Eliminar registros duplicados

Normalización

- Estandarizar datos
- Re-escalar datos a un rango específico
- Técnicas: min-max, z-score, transformación logarítmica

Introducción – Técnicas de Limpieza



Detección y
tratamiento de
valores nulos

Valores Atípicos

- Se tratan mediante la eliminación, o imputación (la media, mediana, k-vecino más cercano)

- Técnicas estadísticas como: rango intercuartílico o desviación estándar

Introducción – Técnicas de Limpieza



Análisis de correlación

- Identificar relaciones entre variables
- Coeficiente de correlación

Limpieza de datos con SQL

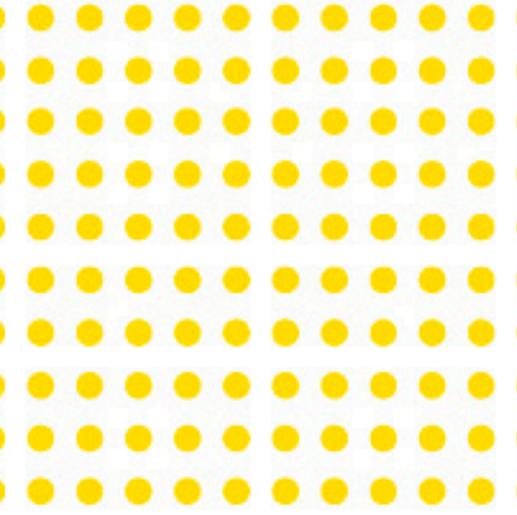


SQL ofrece una variedad de **comandos** para realizar tareas de limpieza de datos de manera eficiente. A continuación, se presentan ejemplos de cómo limpiar datos usando:

1. Eliminar registros duplicados

Los registros duplicados pueden generar resultados incorrectos en los análisis. Sentencia SQL:

```
DELETE FROM mi_tabla
WHERE rowid IN (
    SELECT rid
    FROM (
        SELECT rowid AS rid,
        ROW_NUMBER() OVER (PARTITION BY columna1, columna2, columna3 ORDER BY rowid) AS rn
        FROM mi_tabla
    )
    WHERE rn > 1
);
```



Eliminar registros duplicados

Paso 1. Buscar registros de canciones con el mismo nombre

```
SELECT nombre, count(id) as cantidad FROM canciones  
GROUP BY nombre  
HAVING COUNT(id) > 2  
ORDER BY cantidad DESC
```

NOMBRE	CANTIDAD
1 The Trooper	5
2 The Number Of The Beast	5
3 Wrathchild	5
4 2 Minutes To Midnight	5
5 Iron Maiden	5
6 Hallowed Be Thy Name	5
7 The Evil That Men Do	4
8 Fear Of The Dark	4
9 Sanctuary	4
10 Running Free	4
11 Wasting Love	3
12 Snowblind	3
13 I Can't Quit You Baby	3
14 Whole Lotta Love	3
15 Collision	3
16 Run To The Hills	3
17 The Clairvoyant	3
18 Smoke On The Water	3

Paso 2. Revisar los registros de la tabla

```
SELECT * FROM canciones  
WHERE nombre IN(  
SELECT nombre FROM canciones  
GROUP BY nombre  
HAVING COUNT(nombre) > 2  
)  
ORDER BY nombre
```

ID	NOMBRE	ALBUM_ID	MEDIO_ID	GENERO_ID	COMPOSITOR
1221	2 Minutes To Midnight	95	1	3	Adrian Smith/Bruce Dickinson
1319	2 Minutes To Midnight	104	1	1	Adrian Smith/Bruce Dickinson
1345	2 Minutes To Midnight	107	1	3	Smith/Dickinson
1289	2 Minutes To Midnight	102	1	3	Smith/Dickinson
1357	2 Minutes To Midnight	108	1	3	Adrian Smith/Bruce Dickinson
1230	Afraid To Shoot Strangers	96	1	3	Steve Harris
1313	Afraid To Shoot Strangers	103	1	1	(null)
1258	Afraid To Shoot Strangers	99	1	1	Steve Harris

Eliminar registros duplicados

Paso 3. Enumerar los registros duplicados

```
SELECT nombre, rowid AS rid,  
       ROW_NUMBER() OVER (PARTITION BY nombre ORDER BY rowid) AS rn  
FROM canciones  
WHERE nombre like '2 Mi%'
```

NOMBRE	RID	RN
2 Minutes To Midnight	AACZmPAAQAAAAn39AAC	1
2 Minutes To Midnight	AACZmPAAQAAAAn39AAc	2
2 Minutes To Midnight	AACZmPAAQAAAAn39AAo	3
2 Minutes To Midnight	AACZmPAAQAAAAn3/ACH	4
2 Minutes To Midnight	AACZmPAAQAAAAn3/ADL	5

Paso 4 Eliminar registros

```
DELETE FROM canciones  
WHERE rowid IN (  
    SELECT rid  
    FROM (  
        SELECT nombre , rowid AS rid,  
               ROW_NUMBER() OVER (PARTITION BY nombre ORDER BY rowid) rn  
        FROM canciones  
        ORDER BY nombre  
    )  
    WHERE rn > 1  
);
```

Después Borrado

ID	NOMBRE	ALBUM_ID	MEDIO_ID	GENERO_ID	COMPOSITOR
1319	2 Minutes To Midnight	104	1	1	Adrian Smith/Bruce Dickinson

2. Manejar valores NULL

Se puede reemplazar un NULL con un valor adecuado o eliminar las filas que los contienen

Paso 1. Reemplazar con un valor

UPDATE tabla

SET columna = valor_por_defecto

WHERE columna **IS NULL**;

a) Consultar campos con registros nulos

```
SELECT * FROM canciones  
WHERE compositor IS NULL
```

ID	NOMBRE	ALBUM_ID	MEDIO_ID	GENERO_ID	COMPOSITOR
456	Heart of the Night	38	1	2 (null)	
457	De La Luz	38	1	2 (null)	
458	Westwood Moon	38	1	2 (null)	
459	Midnight	38	1	2 (null)	
460	Playtime	38	1	2 (null)	
461	Surrender	38	1	2 (null)	
462	Valentino's	38	1	2 (null)	
463	Believe	38	1	2 (null)	

b) Dependiendo del tipo de atributo, reemplazar el valor. Para el caso : "Unknow"

```
UPDATE canciones  
SET compositor='Unknow'  
WHERE compositor IS NULL;
```

ID	NOMBRE	COMPOSITOR
459	Midnight	Unknow
460	Playtime	Unknow
461	Surrender	Unknow
462	Valentino's	Unknow
463	Believe	Unknow
464	As We Sleep	Unknow
465	When Evening Falls	Unknow
466	J Squared	Unknow
467	Best Thing	Unknow
468	Maria	Billie Joe Armstrong -Words Green Day -Music
469	Poprocks And Coke	Billie Joe Armstrong -Words Green Day -Music

2. Manejar valores NULL

Se puede reemplazar un NULL con un valor adecuado o eliminar las filas que los contienen

Paso 2. Eliminar filas con valor NULL

DELETE FROM tabla

WHERE columna **IS NULL;**

a) Consultar campos con registros nulos

```
SELECT * FROM canciones  
WHERE compositor IS NULL
```

b) Borrar Registros

```
DELETE FROM canciones  
WHERE compositor IS NULL;
```

ID	NOMBRE	ALBUM_ID	MEDIO_ID	GENERO_ID	COMPOSITOR
456	Heart of the Night	38	1	2	(null)
457	De La Luz	38	1	2	(null)
458	Westwood Moon	38	1	2	(null)
459	Midnight	38	1	2	(null)
460	Playtime	38	1	2	(null)
461	Surrender	38	1	2	(null)
462	Valentino's	38	1	2	(null)
463	Believe	38	1	2	(null)

3. Eliminar caracteres especiales

Corrección datos tipográficos

Paso 1. Campos de texto

UPDATE tabla

SET columna_numerica = **REGEXP_REPLACE**(columna , patron , valorreemplazo);

a) Expresión regular: sólo permite números , letras , espacios y dieresis

SELECT REGEXP_REPLACE ('God Gaves Rock n Roll To You \$5#@@@', '[^a-zA-Z0-9áéíóúÁÉÍÓÚüÜñÑ]', '') as salida **FROM DUAL**;

SALIDA
God Gaves Rock n Roll To You 5

b) Actualización

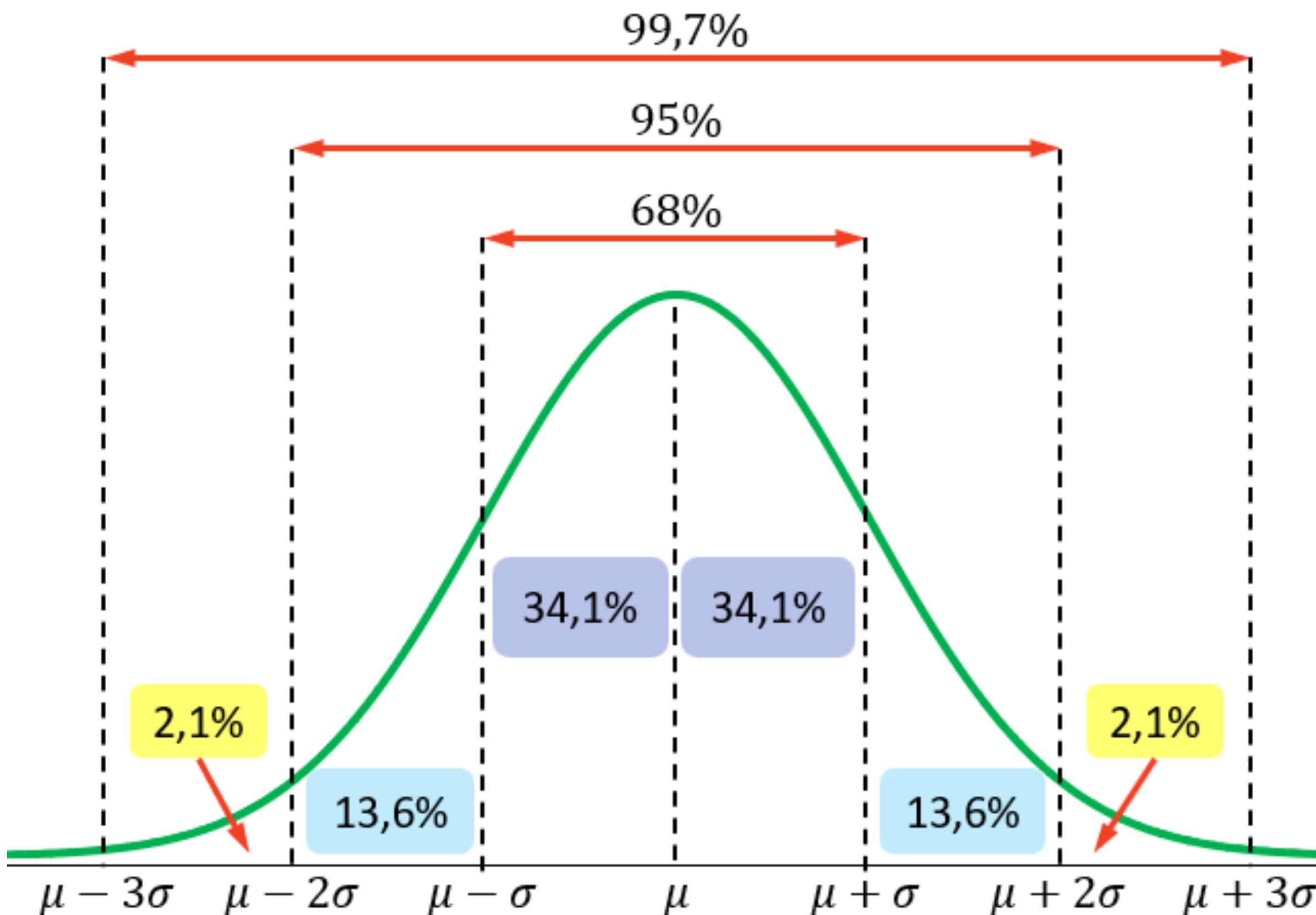
UPDATE canciones
SET nombre = **REGEXP_REPLACE**(nombre, '[^a-zA-Z0-9áéíóúÁÉÍÓÚüÜñÑ]', '')
WHERE id **IN**
(3401,3402,3403,3405,3406,3407,3408,3409,3410,3411,3412,3413,3414,3415)

SINREEMPLAZO	CONREEMPLAZADO
Battlestar Galactica, Pt. 2	Battlestar Galactica Pt 2
Battlestar Galactica, Pt. 3	Battlestar Galactica Pt 3
Lost Planet of the Gods, Pt. 1	Lost Planet of the Gods Pt 1
Lost Planet of the Gods, Pt. 2	Lost Planet of the Gods Pt 2

4. Tratar valores atípicos:(outlier)

Los valores atípicos pueden distorsionar los resultados del análisis. Puede identificarlos y eliminarlos o reemplazarlos con valores más razonables.

Regla empírica



Paso 1. Identifica valores atípicos

```
SELECT *  
FROM tabla  
WHERE columna_numerica > (  
    SELECT AVG(columna_numerica) + 3 * STDDEV(columna_numerica)  
    FROM tabla  
)
```

4. Tratar valores atípicos:(outlier)

Los valores atípicos pueden distorsionar los resultados del análisis. Puede identificarlos y eliminarlos o reemplazarlos con valores más razonables.

Paso 2. Buscar mas de 3 Desviaciones. Para el ejemplo vamos a trabajar sobre el campo de milisegundos en la tabla canciones

```
SELECT id, nombre , milisegundos  
,round((SELECT AVG(milisegundos) FROM canciones),0) media  
,round((SELECT 3 * STDDEV(milisegundos) FROM canciones),0) desviacion  
FROM canciones  
WHERE milisegundos > (  
    SELECT AVG(milisegundos) + 3 * STDDEV(milisegundos)  
    FROM canciones  
)  
ORDER BY milisegundos DESC
```

ID	NOMBRE	MILISEGUNDOS	MEDIA_MILISEGUNDOS	DESVIACION
2820	Occupation / Precipice	5286953	393599	1605016
3224	Through a Looking Glass	5088838	393599	1605016
3244	Greetings from Earth, Pt. 1	2960293	393599	1605016
3242	The Man With Nine Lives	2956998	393599	1605016
3227	Battlestar Galactica, Pt. 2	2956081	393599	1605016
3226	Battlestar Galactica, Pt. 1	2952702	393599	1605016
3243	Murder On the Rising Star	2935894	393599	1605016
3228	Battlestar Galactica, Pt. 3	2927802	393599	1605016
3248	Take the Celestra	2927677	393599	1605016

4. Tratar valores atípicos:(outlier)

Los valores atípicos pueden distorsionar los resultados del análisis. Puede identificarlos y eliminarlos o reemplazarlos con valores más razonables.

Paso 3. Eliminar valores atípicos

```
DELETE FROM canciones  
WHERE milisegundos > (  
    SELECT AVG(milisegundos) + 3 * STDDEV(milisegundos)  
    FROM canciones  
);
```

5. Validación de datos

Se puede usar consultas SQL para verificar la integridad de los datos y detectar posibles problemas

a. Verificar integridad de referencias externas

```
SELECT *
FROM tabla_hija
WHERE id_padre NOT IN (
    SELECT id
    FROM tabla_padre
);
```

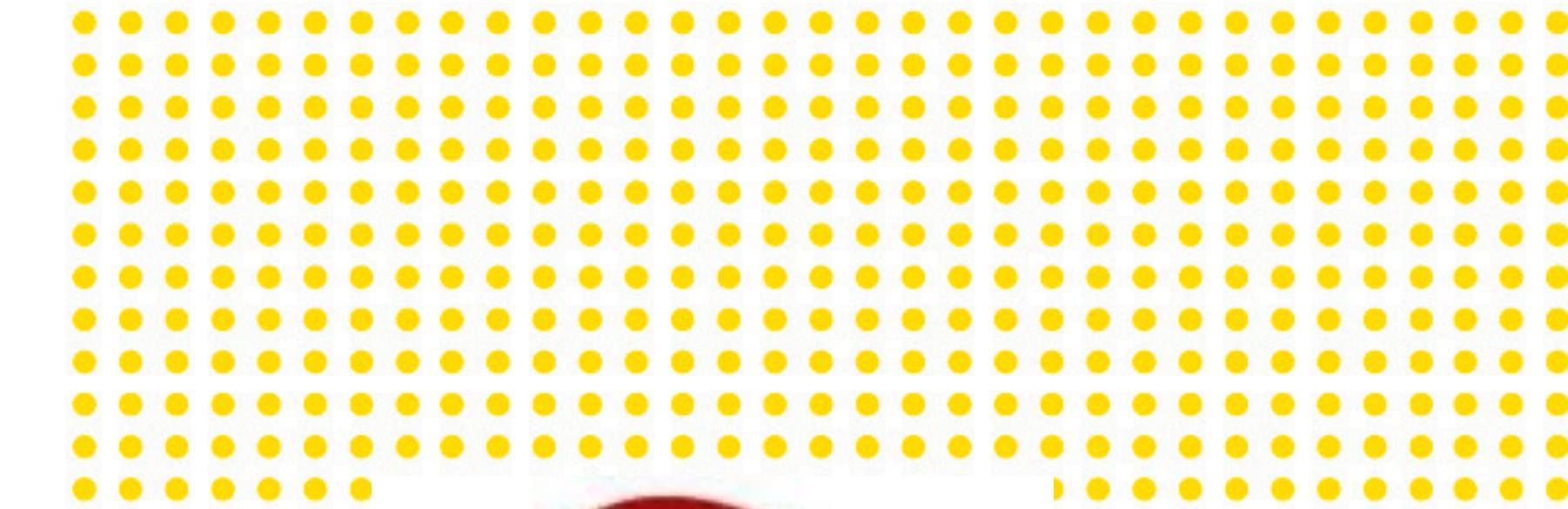
Tabla padre: canciones

Tabla hija: canciones_en_lista

```
SELECT * FROM canciones_en_lista
WHERE CANCION_ID NOT IN(
    SELECT id FROM canciones
)
```

canciones_en_lista	
LISTA_ID	CANCION_ID
1	3336
1	3389
3	2840
3	2842
3	2855
3	2876
3	2879
3	2901
3	3171
3	3428
5	77
5	78
5	79
5	80
5	81
5	82
5	83
5	84
5	121
5	208
5	212

canciones			
ID	NOMBRE	ALBUM...	MEDIO...



5. Validación de datos

Se puede usar consultas SQL para verificar la integridad de los datos y detectar posibles problemas

b. Contar registros por categoría

```
SELECT columna_categorica, COUNT(*) AS conteo  
FROM tabla  
GROUP BY columna_categorica;
```

Caso1. Contar las canciones por “Genero”

```
SELECT g.nombre, count(c.id) as total  
FROM canciones c  
INNER JOIN generos g ON (c.genero_id=g.id)  
GROUP BY g.nombre  
ORDER BY total desc;
```

NOMBRE	TOTAL
Rock	1297
Latin	579
Metal	374
Alternative & Punk	332
Jazz	130
TV Shows	93
Blues	81
Classical	74
Drama	64
R&B/Soul	61

Caso2. Contar las canciones por “Genero” y “Medio”

```
SELECT g.nombre as genero, t.nombre as medio,  
count(c.id) as total  
FROM canciones c  
INNER JOIN generos g ON (c.genero_id=g.id)  
inner JOIN tipos_medio t ON (c.medio_id=t.id)  
GROUP BY g.nombre, t.nombre  
ORDER BY g.nombre;
```

GENERO	MEDIO	TOTAL
Alternative	Protected AAC audio file	38
Alternative	Protected MPEG-4 video file	1
Alternative	Purchased AAC audio file	1
Alternative & Punk	MPEG audio file	332
Blues	MPEG audio file	81
Bossa Nova	MPEG audio file	15
Classical	AAC audio file	1
Classical	Protected AAC audio file	67
Classical	Purchased AAC audio file	6

6. Normalización

```

SELECT id, nombre, milisegundos, round((SELECT AVG(milisegundos)AS mean FROM canciones),0) media,
round((SELECT STDDEV(milisegundos) FROM canciones),0) desviacion,
round(((milisegundos - (SELECT AVG(milisegundos)AS mean FROM canciones)) /
(SELECT STDDEV(milisegundos) FROM canciones)),2)as zscore
FROM canciones
ORDER BY milisegundos desc
    
```

a. z-score

$$z = \frac{X - \bar{X}}{S}$$

Sujeto → Población
Desviación Típica ←

ID	NOMBRE	MILISEGUNDOS	MEDIA	DESVIACION	ZSCORE
2820	Occupation / Precipice	5286953	393599	535005	9,15
3224	Through a Looking Glass	5088838	393599	535005	8,78
3244	Greetings from Earth, Pt. 1	2960293	393599	535005	4,8
3242	The Man With Nine Lives	2956998	393599	535005	4,79
3227	Battlestar Galactica, Pt. 2	2956081	393599	535005	4,79
3226	Battlestar Galactica, Pt. 1	2952702	393599	535005	4,78

b. Min- Max

$$z = \frac{x - \min(x)}{\max(x) - \min(x)}$$

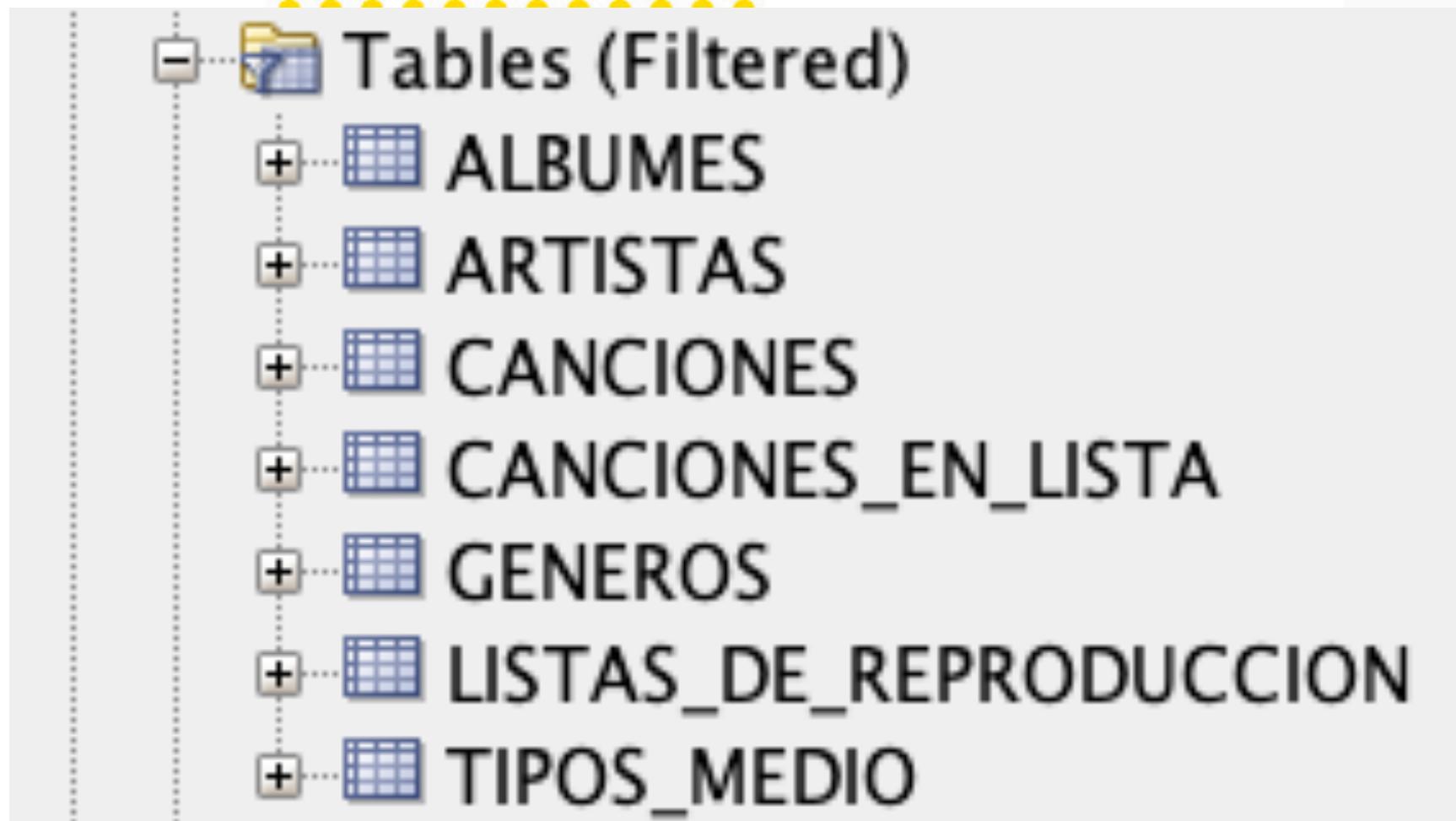
```

SELECT id,nombre, milisegundos, round(((milisegundos -(SELECT MIN(milisegundos) FROM canciones)) /
((SELECT MAX(milisegundos) FROM canciones) - (SELECT MIN(milisegundos)FROM canciones))),2)
AS normalized_value
FROM canciones;
    
```

ID	NOMBRE	MILISEGUNDOS	NORMALIZED_VALUE
444	Do You Love Me	214987	0,04
445	She	248346	0,05
446	I Was Made For Loving You	271360	0,05
447	Shout It Out Loud	219742	0,04
448	God Of Thunder	255791	0,05
449	Calling Dr. Love	225332	0,04

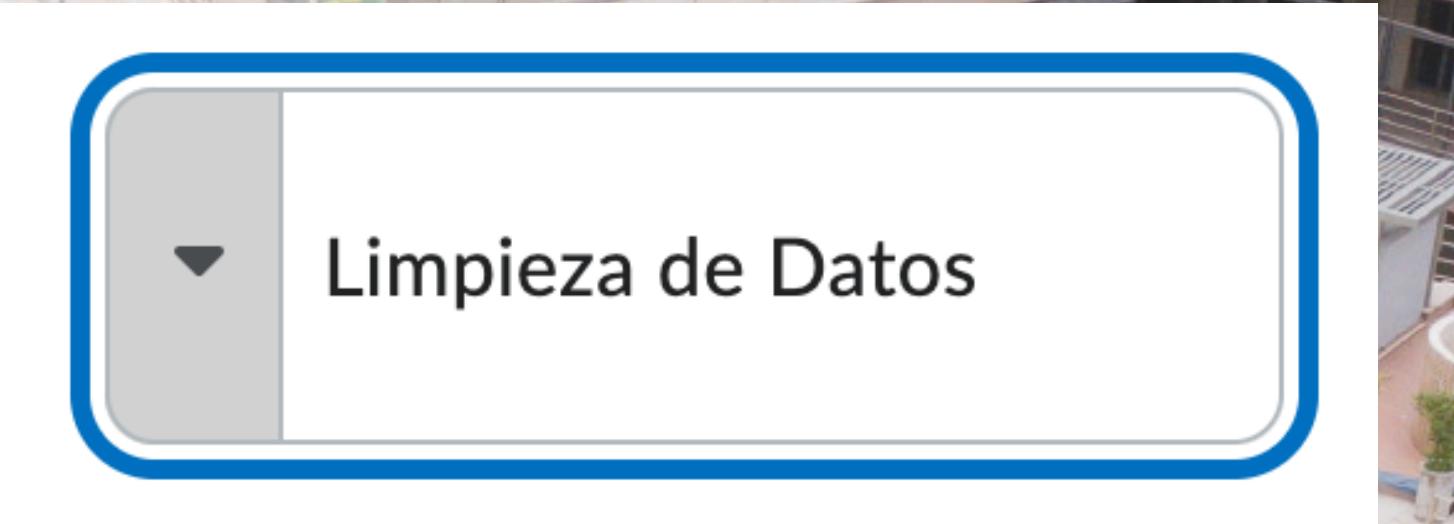
Apropiación

Ejecutar las sentencias SQL para limpieza de datos
dado los casos anteriores sobre las tablas MusiK en
cada cuenta de usuario



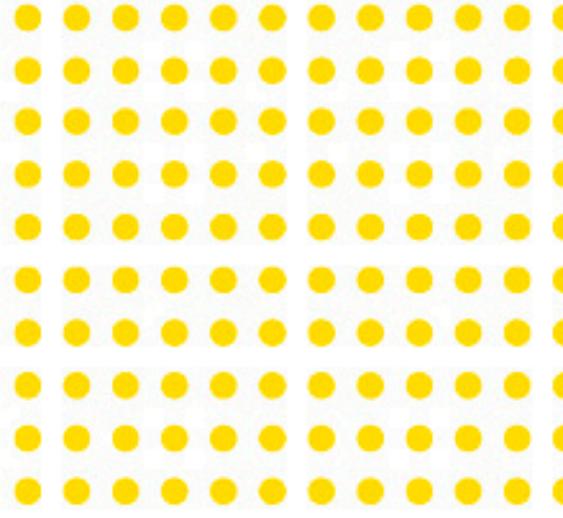
Ejecutar El siguiente archivo paso a paso

a) Script-SQL-Tecnicas de Limpieza.sql



CON SQL: Script de Limpieza de datos - MusiK

1. [Script-SQL-Tecnicas de Limpieza.sql](#)



Ejercicio de Apropiación



Sobre la tabla de CANCIONES queremos eliminar registros duplicados , cuyo **nombre de canción** y **nombre de género** , tengan más de un registro (tip. La duplicidad de campos es teniendo en cuenta dos columnas(nombre cancio y nombre género), el nombre del género está en la tabla GÉNERO debe hacer un JOIN para poder visualizar el nombre)- Elabore las sentencias SQL

	NOMBRE_GENERO	ID	NOMBRE	ALBUM_ID	MEDIO_ID	GENERO_ID	COMPOSITOR
1	Rock	1365	Fear Of The Dark	109	1	1	Steve Harris
2	Rock	1267	Fear Of The Dark	99	1	1	Steve Harris
3	Rock	1314	Fear Of The Dark	103	1	1	(null)
4	Metal	1390	Hallowed Be Thy Name	112	1	3	Steve Harris
5	Metal	1223	Hallowed Be Thy Name	95	1	3	Steve Harris
6	Metal	1296	Hallowed Be Thy Name	102	1	3	Harris
7	Rock	338	I Can't Quit You Baby	30	1	1	Willie Dixon
8	Rock	1589	I Can't Quit You Baby	128	1	1	Willie Dixon
9	Rock	1625	I Can't Quit You Baby	132	1	1	Willie Dixon
10	Rock	1320	Iron Maiden	104	1	1	(null)

Limpieza de datos con Python



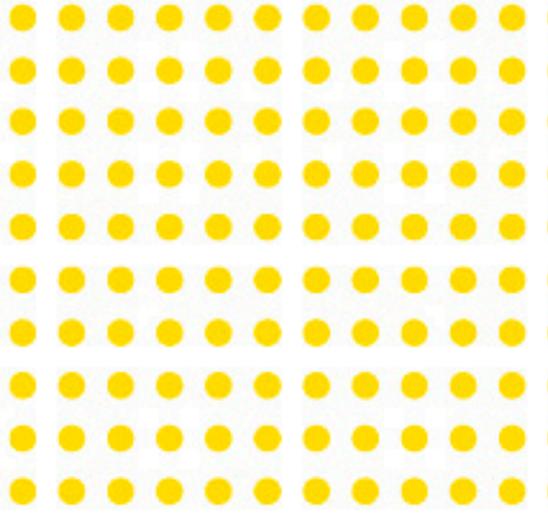
Data-Model Cervical Cancer

Dataset specification: <https://archive.ics.uci.edu/dataset/383/cervical+cancer+risk+factors>

Library to visualize missing data <https://github.com/ResidentMario/missingno>

Variables Table

Variable Name	Role	Type	Demographic	Description	Units	Missing Values
Age	Feature	Integer	Age			no
Number of sexual partners	Feature	Continuous	Other			yes
First sexual intercourse	Feature	Continuous				yes
Num of pregnancies	Feature	Continuous				yes
Smokes	Feature	Continuous				yes
Smokes (years)	Feature	Continuous				yes
Smokes (packs/year)	Feature	Continuous				yes
Hormonal Contraceptives	Feature	Continuous				yes
Hormonal Contraceptives (years)	Feature	Continuous				yes
IUD	Feature	Continuous				yes



Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google

1. Importar librerias

```
import pandas as pd  
import matplotlib.pyplot as plt  
import seaborn as sns  
import missingno as miss
```

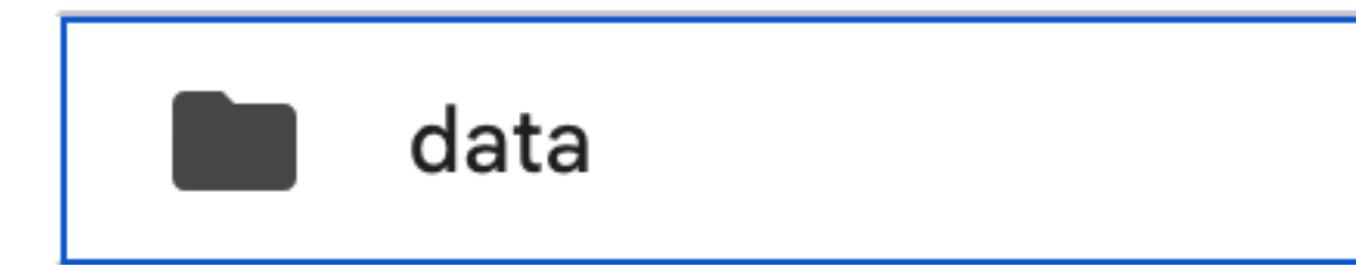
2. Cargar dataset

a) Montar el Google Drive

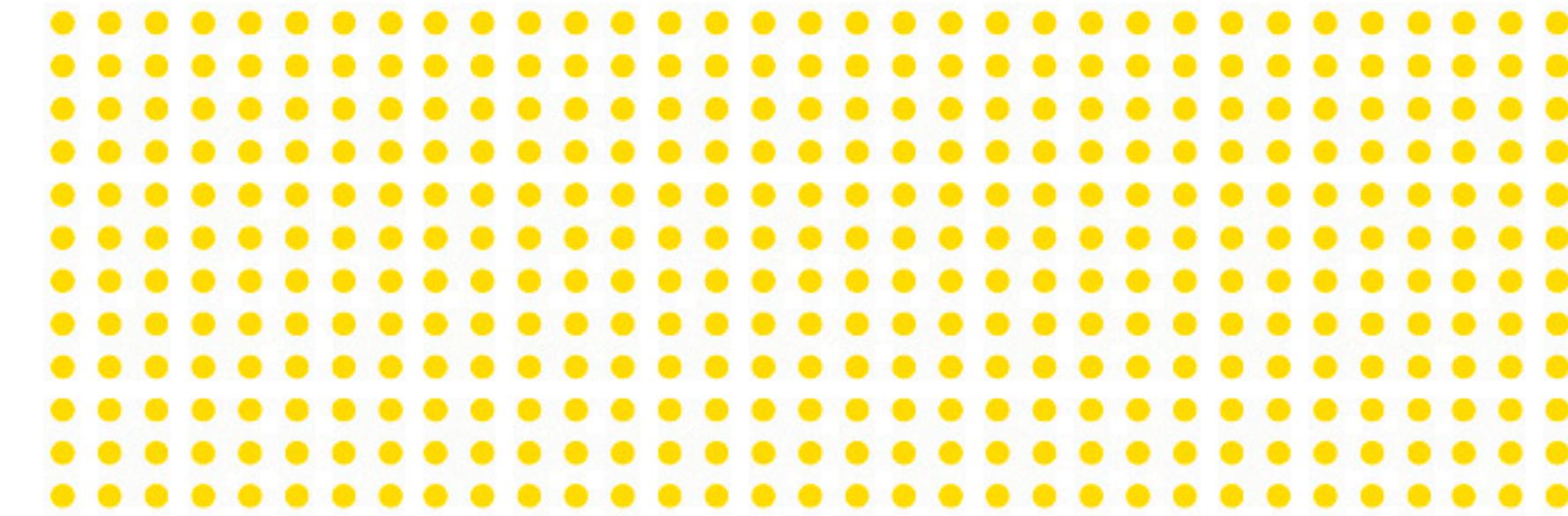
```
from google.colab import drive  
drive.mount('/content/drive')
```

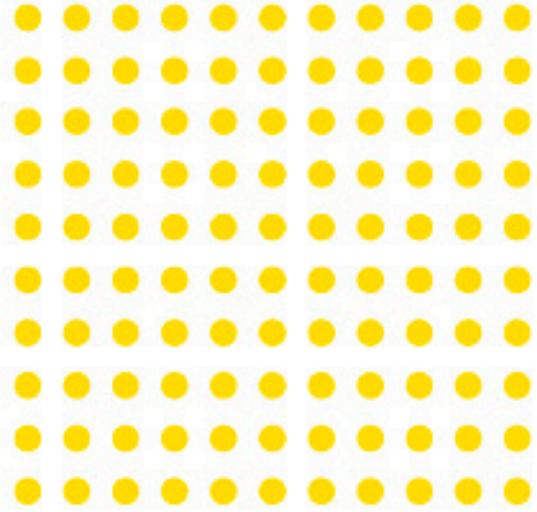
b) Leer datos

```
data = pd.read_csv('/content/drive/MyDrive/data/risk_factors_cervical_cancer.csv',na_values = na_values)
```



co ModelCancer.ipynb





Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google



data



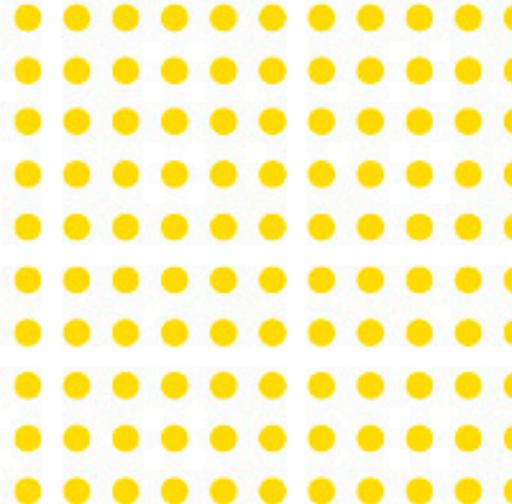
ModelCancer.ipynb

3. Visualizar data

```
data.head()
```

	Age	Number of sexual partners	First sexual intercourse	Num of pregnancies	Smokes	Smokes (years)	Smokes (packs/year)	Hormonal Contraceptives	Hormonal Contraceptives (years)	IUD	
0	18	4.0	15.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	15	1.0	14.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	34	1.0	Nan	1.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	52	5.0	16.0	4.0	1.0	37.0	37.0	1.0	3.0	0.0	
4	46	3.0	21.0	4.0	0.0	0.0	0.0	1.0	15.0	0.0	

5 rows × 36 columns



Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google



data



ModelCancer.ipynb

4. Resumen estadistico de los datos del data frame por columna

```
data_filled.describe()
```

	Age	Number of sexual partners	First sexual intercourse	Num of pregnancies	Smokes	Smokes (years)	Smokes (packs/year)	Hormonal Contraceptives	Hormonal Contraceptives (years)	IUD	
count	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	858.000000	
mean	26.820513	2.527644	16.995300	2.275561	0.145562	1.219721	0.453144	0.641333	2.256419	0.112011	
std	8.497948	1.642267	2.791883	1.399325	0.350189	4.057885	2.209657	0.448671	3.519082	0.293260	
min	13.000000	1.000000	10.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
25%	20.000000	2.000000	15.000000	1.000000	0.000000	0.000000	0.000000	0.000000	0.000000	0.000000	
50%	25.000000	2.000000	17.000000	2.000000	0.000000	0.000000	0.000000	1.000000	1.000000	0.000000	
75%	32.000000	3.000000	18.000000	3.000000	0.000000	0.000000	0.000000	1.000000	2.256419	0.000000	
max	84.000000	28.000000	32.000000	11.000000	1.000000	37.000000	37.000000	1.000000	30.000000	1.000000	

8 rows x 34 columns

Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google



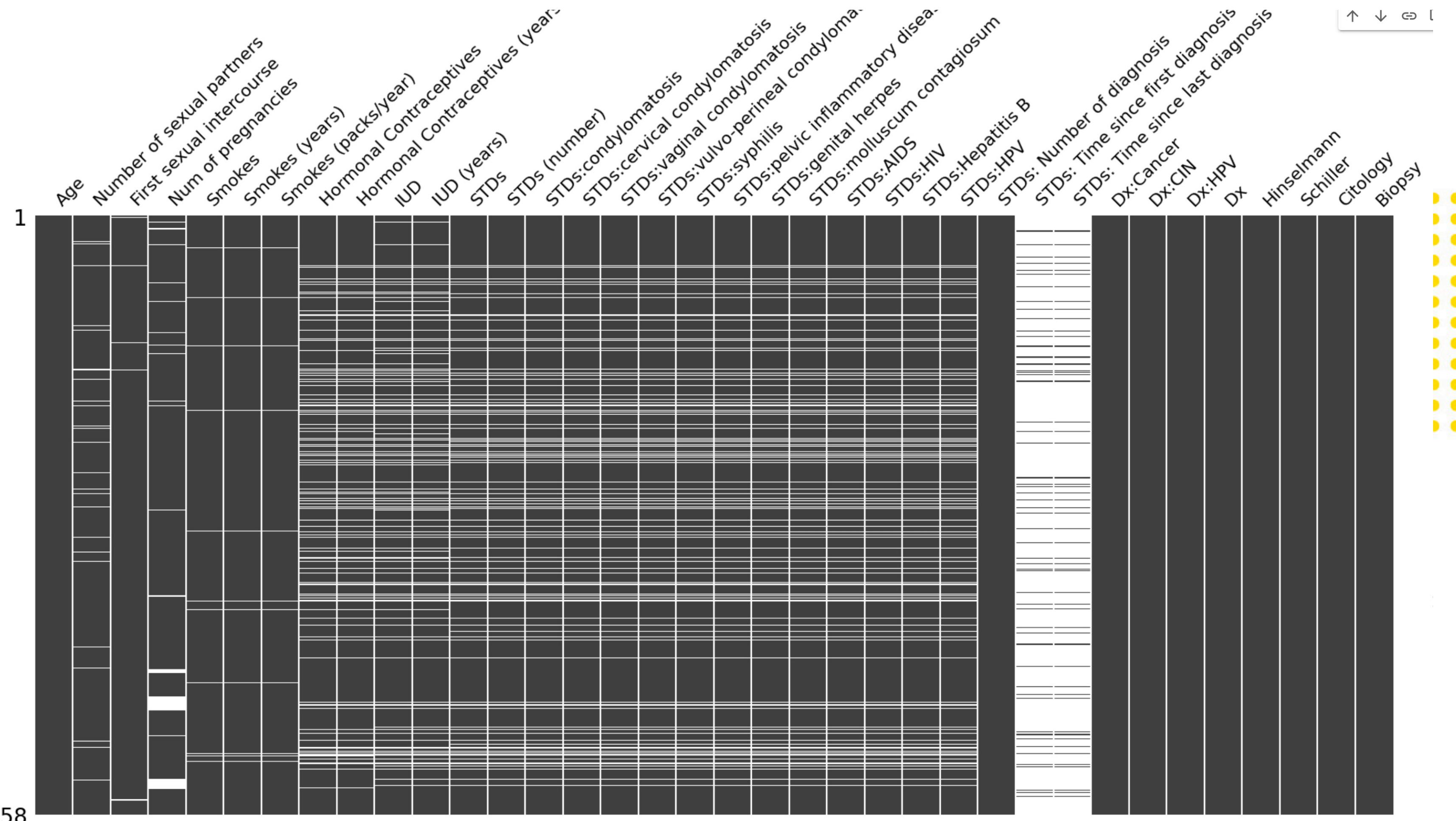
data

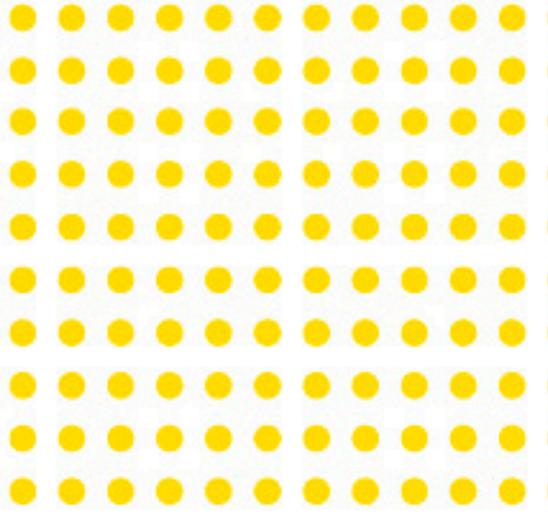


ModelCancer.ipynb

5. Visualizar datos perdidos y Outlier

```
miss.matrix(data);
```





Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google



data

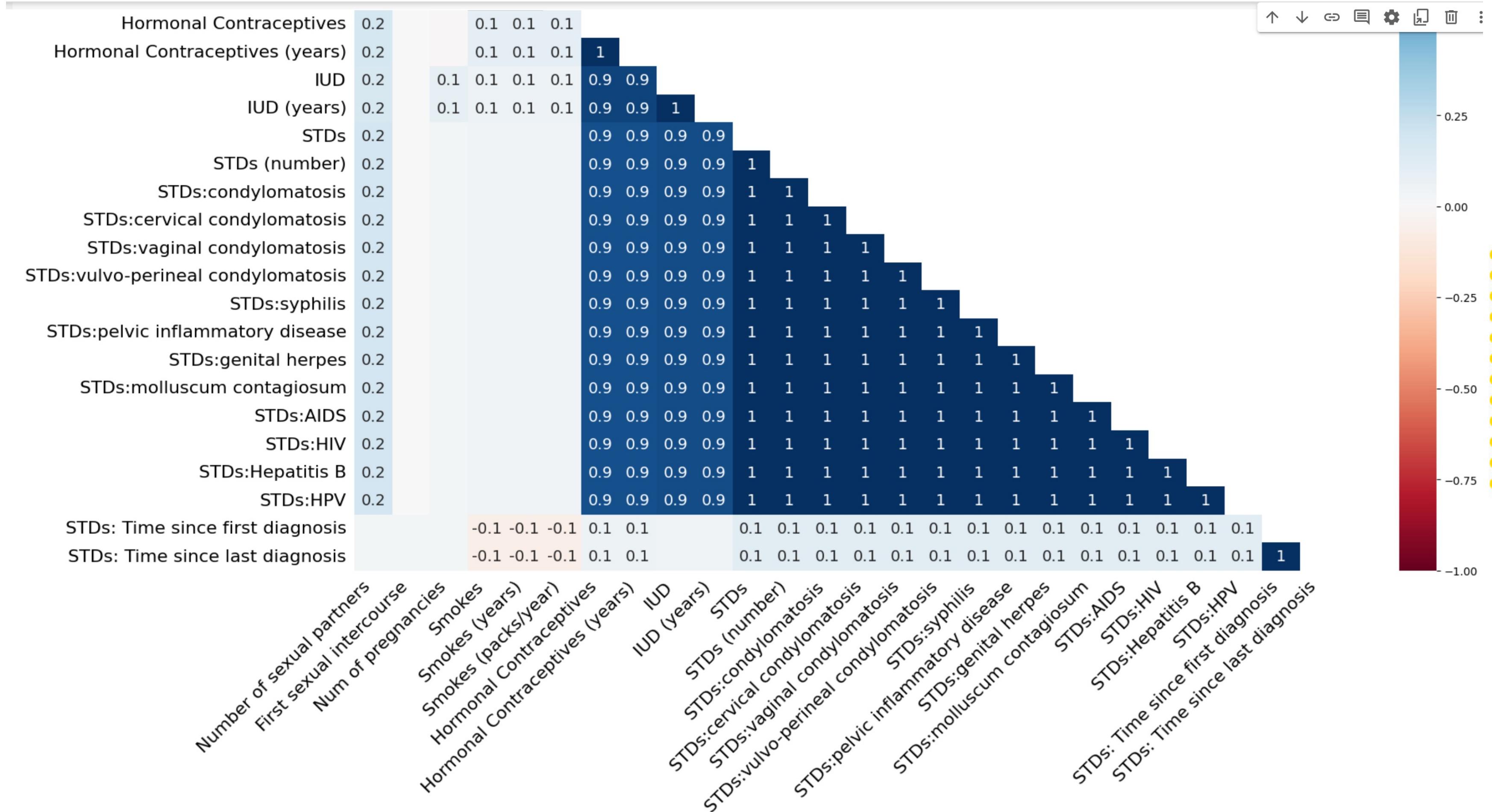


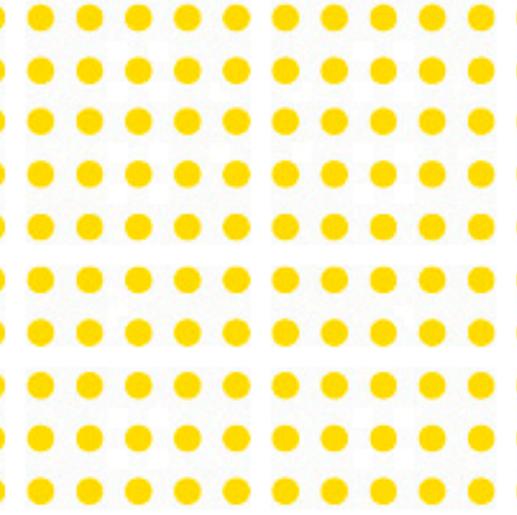
ModelCancer.ipynb

6. Mapa de calor(correlation heatmap)

miss.heatmap(data)

- a) -1 (si una variable aparece, la otra definitivamente no)
- b) 0 (las variables que aparecen o no no tienen ningún efecto entre sí)
- c) 1 (si aparece una variable, definitivamente también aparece la otra).





Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google



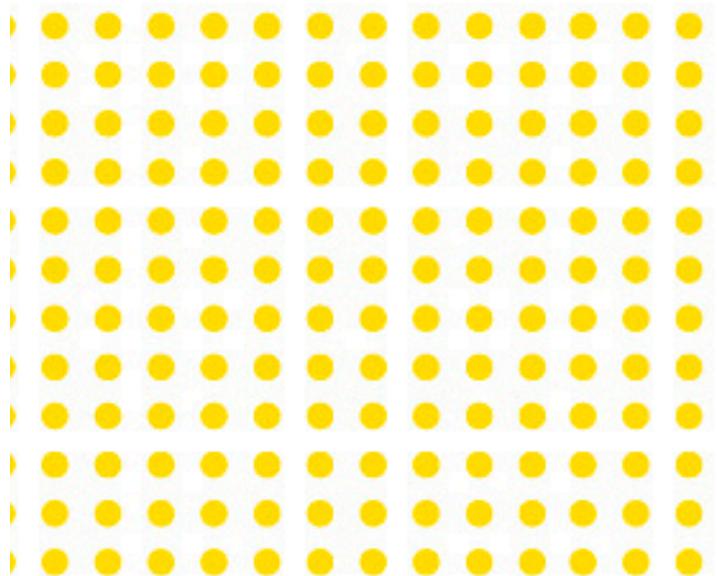
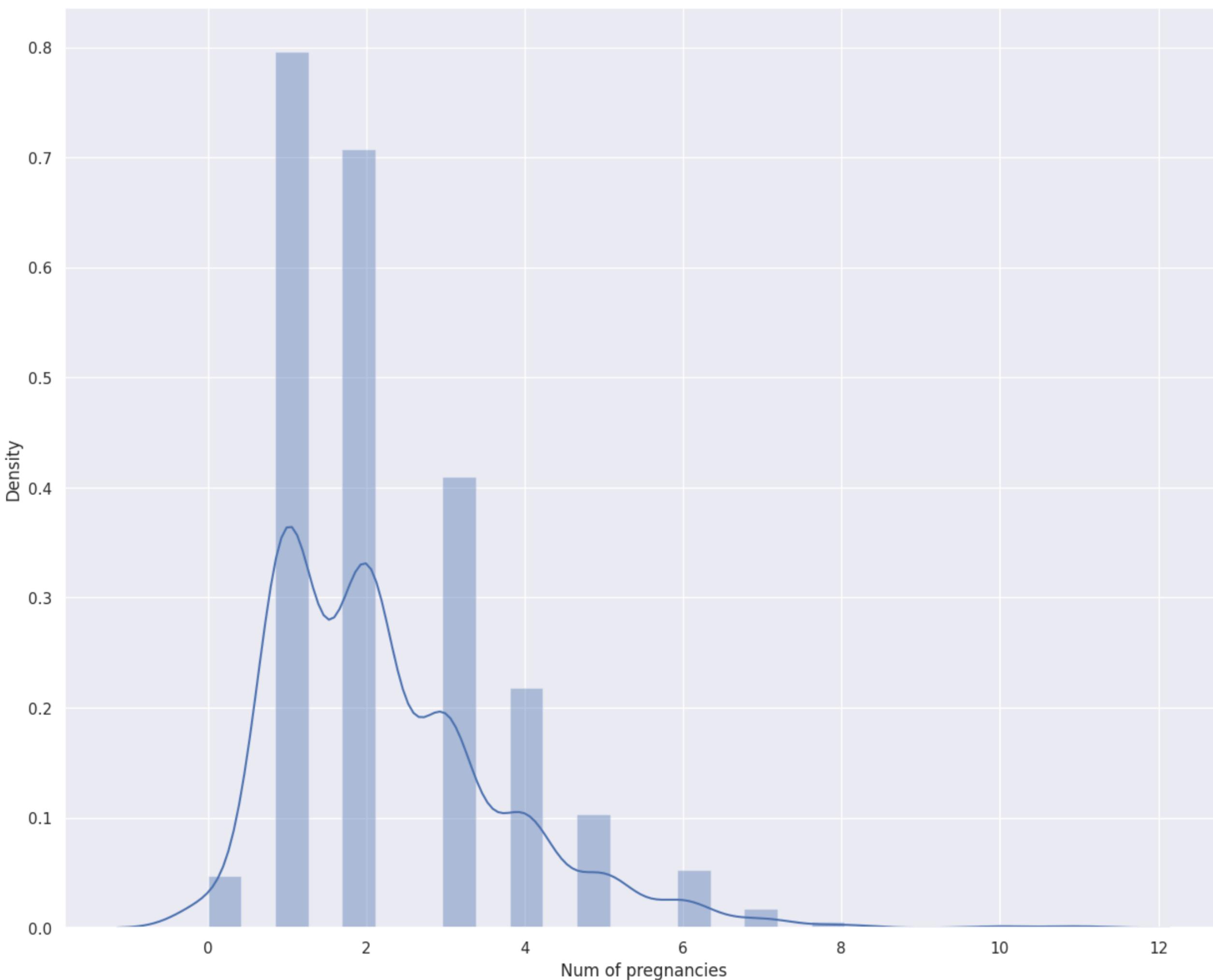
data

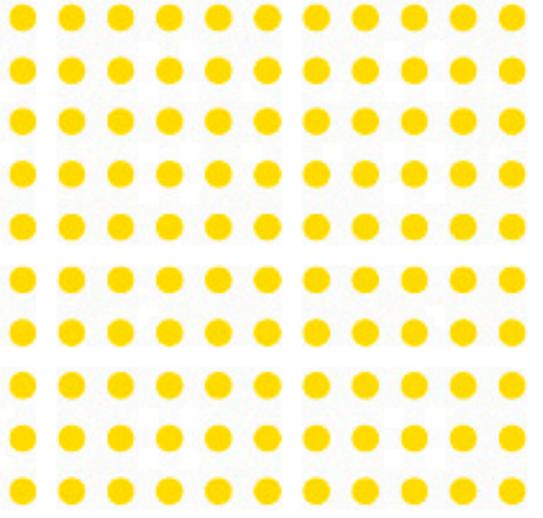


ModelCancer.ipynb

7. Visualizar la distribucion de una observacion

```
sns.distplot(data['Num of pregnancies']);
```



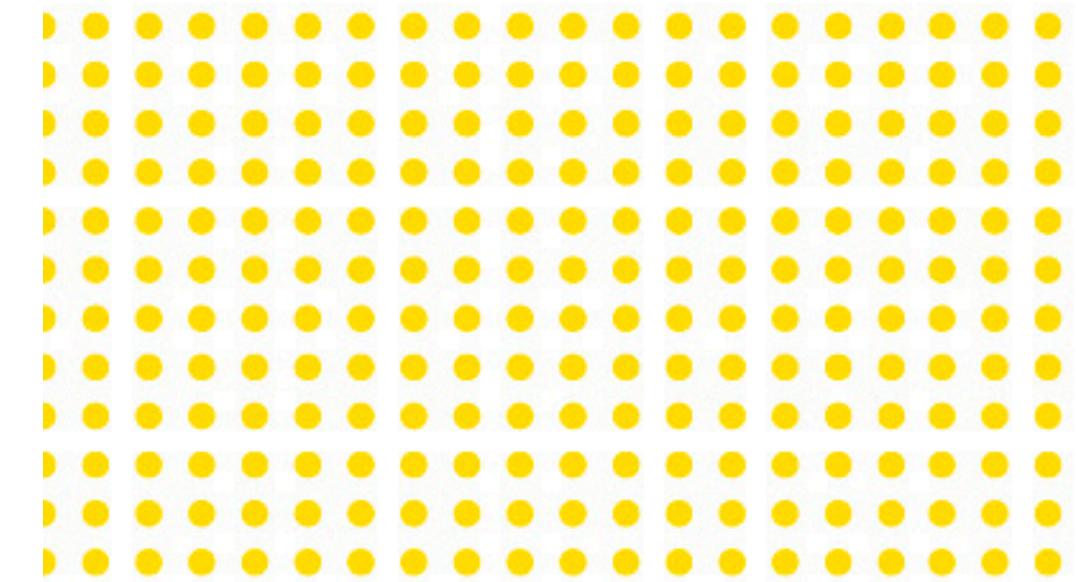
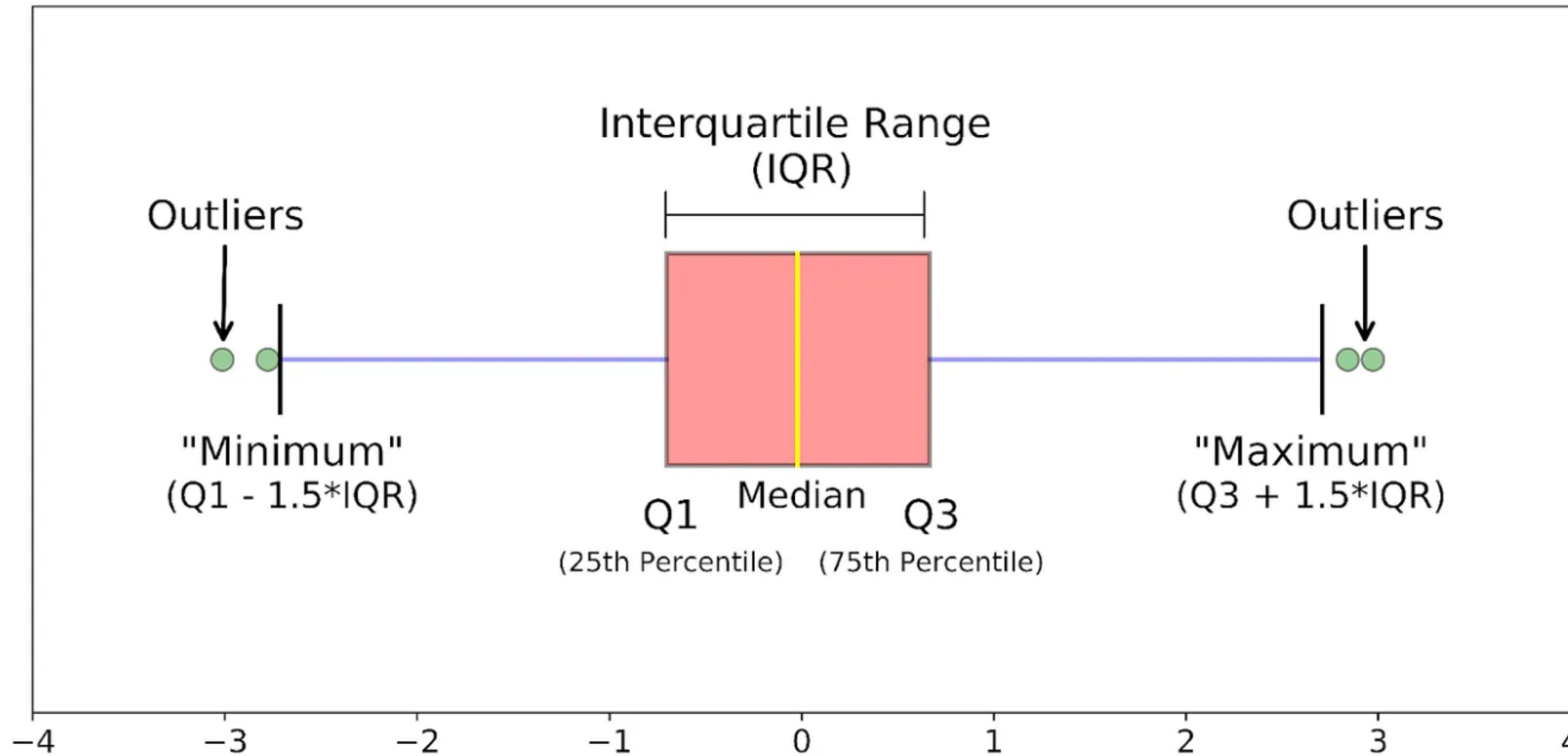


Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google



8. Detección de Outlier – Rango Intercuartilico



Google Collaborate

ModelCancer.ipynb copiar archivo al Drive de Google



data



ModelCancer.ipynb

8. Detección de Outlier

```
sns.set(rc={'figure.figsize':(15,12)})  
sns.boxplot(data=data_filled,palette='rainbow',orient='h');
```



DESAFÍOS EN LA LIMPIEZA DE DATOS

La limpieza de datos presenta desafíos como:

- La escalabilidad
- La calidad de los datos de origen y
- La automatización.

Es crucial abordar estos desafíos para garantizar la eficiencia y confiabilidad de los procesos de limpieza.

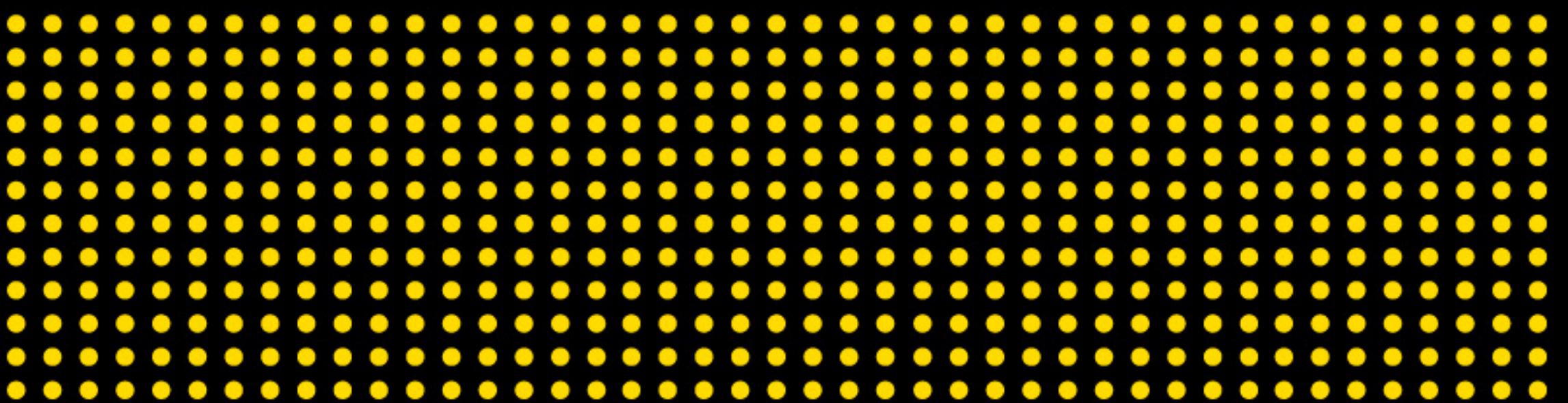


¡Gracias!

Aprendiendo juntos a lo largo de la vida

educacioncontinua.uniandes.edu.co

Síguenos: **EdcoUniandes**     



**Educación
Continua**
Vicerrectoría Académica

Universidad de los Andes | Vigilada Mineducación. Reconocimiento como Universidad: Decreto 1297 del 30 de mayo de 1964. Reconocimiento personería jurídica: Resolución 28 del 23 de febrero de 1949 Minjusticia.



¡Gracias!

Aprendiendo juntos a lo largo de la Vida

educacioncontinua.uniandes.edu.co

Siguenos en **EdcoUniandes**      