Zeppelin Notebook Job Search anonymous

Proyecto Bimestral

Head default

FINISHED

📊 Análisis Exploratorio de Datos (EDA)

Objetivo:
Realizar un Análisis Exploratorio de Datos (EDA) completo al conjunto de datos que se utilizará en el proyecto bimestral.

🔍 Integrantes:
-Juan Pablo Landi
-María Valentina Samaniego
-María Paula Guallo

Took 0 seconds. Last updated by anonymous at June 18 2025, 7:18:44 PM.

FINISHED

1 Lectura del archivo

Took 0 seconds. Last updated by anonymous at June 18 2025, 8:15:06 PM.

SPARK JOB FINISHED

```
val dfNacidos = spark
    .read
    .option("header", true)
    .option("sep", ";")
    .option("inferSchema", true)
    .csv("/workspace/progava-s10/ENV_2023-copia.csv")
```

```
dfNacidos: org.apache.spark.sql.DataFrame = [prov_insc: string, cant_insc: string ... 45 more fields]
```

Took 6 seconds. Last updated by anonymous at June 25 2025, 6:42:43 PM.

FINISHED

2 Imprimir Cantidad de Filas y Columnas

🔢 *Propósito:*
Verificar que los datos se hayan cargado correctamente, comparando las dimensiones del dataset con lo especificado en la documentación oficial de origen.

✅ *Acción:*
Imprimir el número total de *filas* (registros) y *columnas* (variables).

📌 Esta validación inicial es esencial para confirmar que no hubo errores al momento de la lectura del archivo o la conexión con la fuente de datos.

Took 0 seconds. Last updated by anonymous at June 18 2025, 8:15:27 PM.

SPARK JOB FINISHED

```
print(s"Registros (filas); ${dfNacidos.count}, Variables(columnas): ${dfNacidos.columns.length}")
```

```
Registros (filas); 241295, Variables(columnas): 47
```

Took 2 seconds. Last updated by anonymous at June 18 2025, 7:00:34 PM. (outdated)

FINISHED

3 Imprimir el Esquema del Dataset

-Propósito:
Verificar si la interfaz de lectura ha procesado correctamente los tipos de datos de cada columna.

Referencia:
Según el archivo inec_nacidosvivos_dd_2023.ods, algunas columnas esperadas como *numéricas* son:

- anio_nac (Año de nacimiento)
- dia_nac (Día de nacimiento)
- anio_insc (¿Año de inscripción?)
- mes_insc (¿Mes de inscripción?)
- dia_insc (¿Día de inscripción?)
- talla (¿Talla del recién nacido?)
- peso (¿Peso del recién nacido?)

📌 Esta validación es clave para evitar errores posteriores durante el análisis estadístico o modelado.

Took 0 seconds. Last updated by anonymous at June 18 2025, 8:15:46 PM.

FINISHED

```
dfNacidos.printSchema
```

```
root
 |-- prov_insc: string (nullable = true)
 |-- cant_insc: string (nullable = true)
 |-- parr_insc: string (nullable = true)
 |-- fecha_insc: string (nullable = true)
 |-- anio_insc: string (nullable = true)
 |-- mes_insc: string (nullable = true)
 |-- dia_insc: string (nullable = true)
 |-- sexo: string (nullable = true)
 |-- fecha_nac: string (nullable = true)
 |-- anio_nac: integer (nullable = true)
 |-- mes_nac: string (nullable = true)
 |-- dia_nac: integer (nullable = true)
 |-- talla: string (nullable = true)
 |-- peso: string (nullable = true)
 |-- sem_gest: string (nullable = true)
```

```
|-- tipo_part: string (nullable = true)
|-- lugar_ocur: string (nullable = true)
|-- apgar1: string (nullable = true)
|-- apgar5: string (nullable = true)
|-- p_emb: string (nullable = true)
|-- prov_nac: string (nullable = true)
|-- cant_nac: string (nullable = true)
|-- parr_nac: string (nullable = true)
|-- area_nac: string (nullable = true)
|-- asis_por: string (nullable = true)
|-- nac_mad: string (nullable = true)
|-- cod_pais: string (nullable = true)
|-- fecha_mad: string (nullable = true)
|-- anio_mad: string (nullable = true)
|-- mes_mad: string (nullable = true)
|-- dia_mad: string (nullable = true)
|-- edad_mad: string (nullable = true)
|-- con_pren: string (nullable = true)
|-- num_emb: string (nullable = true)
|-- num_par: string (nullable = true)
|-- hij_viv: integer (nullable = true)
|-- hij_vivm: string (nullable = true)
|-- hij_nacm: string (nullable = true)
|-- etnia: string (nullable = true)
|-- est_civil: string (nullable = true)
|-- niv_inst: string (nullable = true)
|-- sabe_leer: string (nullable = true)
|-- prov_res: string (nullable = true)
|-- cant_res: string (nullable = true)
|-- parr_res: string (nullable = true)
|-- area_res: string (nullable = true)
|-- residente: string (nullable = true)
```

Took 0 seconds. Last updated by anonymous at June 18 2025, 7:08:12 PM.

FINISHED

4 Estadística Descriptiva

-Objetivo:
Resumir y describir las principales características de los datos mediante medidas estadísticas básicas.

-Enfoque 1: describe()
Utilizaremos el método describe() para obtener un resumen estadístico de las columnas numéricas (y, si es posible, también de algunas categóricas).

Took 0 seconds. Last updated by anonymous at June 18 2025, 8:18:17 PM.

SPARK JOB FINISHED

```
dfNacidos
    .describe()
    .show
```

```
+-------+---------------+---------+--------------------+----------+-----------------+----------+-----------------+------+----------+-----------------+----------+-----------------+-----------------+----------
|summary|      prov_insc|cant_insc|           parr_insc|fecha_insc|        anio_insc|  mes_insc|          dia_insc|  sexo| fecha_nac|         anio_nac|   mes_nac|          dia_nac|            talla|
+-------+---------------+---------+--------------------+----------+-----------------+----------+-----------------+------+----------+-----------------+----------+-----------------+-----------------+----------
|  count|         241295|   241295|              241295|    241295|           241295|    241295|           241295|241295|    241295|           241295|    241295|           241295|           241295|
|   mean|           null|     null|                null|      null|2023.0647378454437|      null|15.856224576692474|  null|      null|2022.9372345054808|      null|15.632362875318593| 48.54522348326832|3068.71532
| stddev|           null|     null|                null|      null|0.6836599754432079|      null| 8.685787511632697|  null|      null|1.5180530471163742|      null| 8.784520746500172|2.4648283275783904|505.607865
|    min|               |         |                    |          |                 |          |                 |Hombre|1900/01/01|             1900|     Abril|                1|               38|
|    max|Zamora Chinchipe|   Zaruma|Ángel Polibio Cháves|2024/04/30|             2024|Septiembre|                9| Mujer|2023/12/31|             2023|Septiembre|               31|   Sin información|   Sin inf
+-------+---------------+---------+--------------------+----------+-----------------+----------+-----------------+------+----------+-----------------+----------+-----------------+-----------------+----------
```

Took 26 seconds. Last updated by anonymous at June 18 2025, 7:14:16 PM. (outdated)

SPARK JOB FINISHED

```
dfNacidos
    .select("anio_insc", "dia_insc", "anio_nac", "dia_nac")
    .describe()
    .show
```

```
+-------+-----------------+-----------------+-----------------+-----------------+
|summary|        anio_insc|         dia_insc|         anio_nac|         dia_nac|
+-------+-----------------+-----------------+-----------------+-----------------+
|  count|           241295|           241295|           241295|           241295|
|   mean|2023.0647378454437|15.856224576692474|2022.9372345054808|15.632362875318593|
| stddev|0.6836599754432079| 8.685787511632697|1.5180530471163742| 8.784520746500172|
|    min|                |                |             1900|                1|
|    max|             2024|                9|             2023|               31|
+-------+-----------------+-----------------+-----------------+-----------------+
```

Took 2 seconds. Last updated by anonymous at June 18 2025, 7:36:08 PM. (outdated)

SPARK JOB FINISHED

```
import org.apache.spark.sql.types._
val numericCols = dfNacidos.schema.fields.filter {
    case StructField(_, IntegerType | LongType | FloatType | DoubleType | ShortType | DecimalType (),_ ,_) => true
    case _=> false
}.map(_.name)

dfNacidos.select(numericCols.map(col):_*).describe().show
```

```
+-------+-----------------+-----------------+-----------------+
|summary|         anio_nac|          dia_nac|          hij_viv|
+-------+-----------------+-----------------+-----------------+
|  count|           241295|           241295|           241295|
|   mean|2022.9372345054808|15.632362875318593|2.0577550301498166|
| stddev|1.5180530471163742| 8.784520746500172|1.2242374055754184|
|    min|             1900|                1|                1|
|    max|             2023|               31|               16|
+-------+-----------------+-----------------+-----------------+

import org.apache.spark.sql.types._
numericCols: Array[String] = Array(anio_nac, dia_nac, hij_viv)
```

Took 3 seconds. Last updated by anonymous at June 18 2025, 7:56:40 PM. (outdated)

SPARK JOB FINISHED

```
dfNacidos.select(numericCols.map(col): _*).summary().show
```

```
+-------+------------------+------------------+------------------+
|summary|          anio_nac|           dia_nac|           hij_viv|
+-------+------------------+------------------+------------------+
|  count|            241295|            241295|            241295|
|   mean|2022.9372345054808|15.632362875318593|2.0577550301498166|
| stddev|1.5180530471163742| 8.784520746500172|1.2242374055754184|
|    min|              1900|                 1|                 1|
|    25%|              2023|                 8|                 1|
|    50%|              2023|                16|                 2|
|    75%|              2023|                23|                 3|
|    max|              2023|                31|                16|
+-------+------------------+------------------+------------------+
```

Took 3 seconds. Last updated by anonymous at June 18 2025, 8:01:44 PM.

SPARK JOB FINISHED

```
dfNacidos.select(numericCols.map(col): _*).summary("stddev", "25%").show
```

```
+-------+------------------+-----------------+------------------+
|summary|          anio_nac|          dia_nac|           hij_viv|
+-------+------------------+-----------------+------------------+
| stddev|1.5180530471163742|8.784520746500172|1.2242374055754184|
|    25%|              2023|                8|                 1|
+-------+------------------+-----------------+------------------+
```

Took 1 second. Last updated by anonymous at June 18 2025, 8:07:55 PM.

SPARK JOB FINISHED

```
dfNacidos.select(numericCols.map(col): _*).summary("stddev", "91%").show
```

```
+-------+------------------+-----------------+------------------+
|summary|          anio_nac|          dia_nac|           hij_viv|
+-------+------------------+-----------------+------------------+
| stddev|1.5180530471163742|8.784520746500172|1.2242374055754184|
|    91%|              2023|               28|                 4|
+-------+------------------+-----------------+------------------+
```

Took 3 seconds. Last updated by anonymous at June 18 2025, 8:09:36 PM.

SPARK JOB FINISHED

```
dfNacidos.select(numericCols.map(col): _*).summary("count", "count_distinct").show()
```

```
+--------------+--------+-------+-------+
|       summary|anio_nac|dia_nac|hij_viv|
+--------------+--------+-------+-------+
|         count|  241295| 241295| 241295|
|count_distinct|      41|     31|     15|
+--------------+--------+-------+-------+
```

Took 2 seconds. Last updated by anonymous at June 18 2025, 8:11:41 PM.

FINISHED

5. Transformaciones

Took 3 seconds. Last updated by anonymous at June 18 2025, 8:13:31 PM.

FINISHED

```
val dfNacidosClean = dfNacidos.withColumn("fecha_insc_date", to_date(col("fecha_insc"), ""))
```

```
dfNacidosClean: org.apache.spark.sql.DataFrame = [prov_insc: string, cant_insc: string ... 46 more fields]
```

Took 0 seconds. Last updated by anonymous at June 18 2025, 8:17:17 PM. (outdated)

FINISHED

```
dfNacidosClean.printSchema
```

```
root
 |-- prov_insc: string (nullable = true)
 |-- cant_insc: string (nullable = true)
 |-- parr_insc: string (nullable = true)
 |-- fecha_insc: string (nullable = true)
 |-- anio_insc: string (nullable = true)
 |-- mes_insc: string (nullable = true)
 |-- dia_insc: string (nullable = true)
 |-- sexo: string (nullable = true)
 |-- fecha_nac: string (nullable = true)
 |-- anio_nac: integer (nullable = true)
 |-- mes_nac: string (nullable = true)
 |-- dia_nac: integer (nullable = true)
 |-- talla: string (nullable = true)
 |-- peso: string (nullable = true)
 |-- sem_gest: string (nullable = true)
 |-- tipo_part: string (nullable = true)
 |-- lugar_ocur: string (nullable = true)
 |-- apgar1: string (nullable = true)
 |-- apgar5: string (nullable = true)
 |-- p_emb: string (nullable = true)
 |-- prov_nac: string (nullable = true)
 |-- cant_nac: string (nullable = true)
 |-- parr_nac: string (nullable = true)
 |-- area_nac: string (nullable = true)
 |-- asis_por: string (nullable = true)
 |-- nac_mad: string (nullable = true)
 |-- cod_pais: string (nullable = true)
 |-- fecha_mad: string (nullable = true)
```

```
|-- anio_mad: string (nullable = true)
|-- mes_mad: string (nullable = true)
|-- dia_mad: string (nullable = true)
|-- edad_mad: string (nullable = true)
|-- con_pren: string (nullable = true)
|-- num_emb: string (nullable = true)
|-- num_par: string (nullable = true)
|-- hij_viv: integer (nullable = true)
|-- hij_vivm: string (nullable = true)
|-- hij_nacm: string (nullable = true)
|-- etnia: string (nullable = true)
|-- est_civil: string (nullable = true)
|-- niv_inst: string (nullable = true)
|-- sabe_leer: string (nullable = true)
|-- prov_res: string (nullable = true)
|-- cant_res: string (nullable = true)
|-- parr_res: string (nullable = true)
|-- area_res: string (nullable = true)
|-- residente: string (nullable = true)
|-- fecha_insc_date: date (nullable = true)
```

Took 0 seconds. Last updated by anonymous at June 18 2025, 8:19:19 PM.

FINISHED

```
val dfNacidosClean = dfNacidos
    .withColumn("fecha_insc_date",to_date(col("fecha_insc"),"yyyy/MM/dd"))
    .withColumn("fecha_nac_date",to_date(col("fecha_nac"),"yyyy/MM/dd"))
    .withColumn("fecha_mad_date",to_date(col("fecha_mad"),"yyyy/MM/dd"))
```

```
dfNacidosClean: org.apache.spark.sql.DataFrame = [prov_insc: string, cant_insc: string ... 48 more fields]
```

Took 0 seconds. Last updated by anonymous at June 25 2025, 6:43:00 PM.

FINISHED

```
dfNacidosClean.printSchema
```

```
root
 |-- prov_insc: string (nullable = true)
 |-- cant_insc: string (nullable = true)
 |-- parr_insc: string (nullable = true)
 |-- fecha_insc: string (nullable = true)
 |-- anio_insc: string (nullable = true)
 |-- mes_insc: string (nullable = true)
 |-- dia_insc: string (nullable = true)
 |-- sexo: string (nullable = true)
 |-- fecha_nac: string (nullable = true)
 |-- anio_nac: integer (nullable = true)
 |-- mes_nac: string (nullable = true)
 |-- dia_nac: integer (nullable = true)
 |-- talla: string (nullable = true)
 |-- peso: string (nullable = true)
 |-- sem_gest: string (nullable = true)
 |-- tipo_part: string (nullable = true)
 |-- lugar_ocur: string (nullable = true)
 |-- apgar1: string (nullable = true)
 |-- apgar5: string (nullable = true)
 |-- p_emb: string (nullable = true)
 |-- prov_nac: string (nullable = true)
 |-- cant_nac: string (nullable = true)
 |-- parr_nac: string (nullable = true)
 |-- area_nac: string (nullable = true)
 |-- asis_por: string (nullable = true)
 |-- nac_mad: string (nullable = true)
 |-- cod_pais: string (nullable = true)
 |-- fecha_mad: string (nullable = true)
 |-- anio_mad: string (nullable = true)
 |-- mes_mad: string (nullable = true)
 |-- dia_mad: string (nullable = true)
 |-- edad_mad: string (nullable = true)
 |-- con_pren: string (nullable = true)
 |-- num_emb: string (nullable = true)
 |-- num_par: string (nullable = true)
 |-- hij_viv: integer (nullable = true)
 |-- hij_vivm: string (nullable = true)
 |-- hij_nacm: string (nullable = true)
 |-- etnia: string (nullable = true)
 |-- est_civil: string (nullable = true)
 |-- niv_inst: string (nullable = true)
 |-- sabe_leer: string (nullable = true)
 |-- prov_res: string (nullable = true)
 |-- cant_res: string (nullable = true)
 |-- parr_res: string (nullable = true)
 |-- area_res: string (nullable = true)
 |-- residente: string (nullable = true)
 |-- fecha_insc_date: date (nullable = true)
 |-- fecha_nac_date: date (nullable = true)
 |-- fecha_mad_date: date (nullable = true)
```

Took 1 second. Last updated by anonymous at June 18 2025, 8:29:15 PM.

SPARK JOB FINISHED

```
import org.apache.spark.sql.types._
import org.apache.spark.sql.functions._

val dateColumnNames = dfNacidosClean
    .schema
    .fields
    .filter {
        case StructField (_, TimestampType | DateType, _, _ ) => true
        case _=> false
    }
    .map(_.name)

// Ejemplo de calculo de estadisticas comunes
val statsExprs = dateColumnNames
    .flatMap { colName =>
        Seq(
            count(col(colName)).alias(s"${colName}_count"),
            min(col(colName)).alias(s"${colName}_min"),
            max(col(colName)).alias(s"${colName}_max"),
            countDistinct(col(colName)).alias(s"${colName}_countDistinct")
        )
    }
```

```
val dfStatsDateCols = dfNacidosClean
    .select(statsExprs:_*)

dfStatsDateCols.show()
```

```
+--------------------+-------------------+-------------------+----------------------------+-------------------+-----------------+-----------------+----------------------------+--------------------+-----------
|fecha_insc_date_count|fecha_insc_date_min|fecha_insc_date_max|fecha_insc_date_countDistinct|fecha_nac_date_count|fecha_nac_date_min|fecha_nac_date_max|fecha_nac_date_countDistinct|fecha_mad_date_count|fecha_mad_c
+--------------------+-------------------+-------------------+----------------------------+-------------------+-----------------+-----------------+----------------------------+--------------------+-----------
|              231333|         2001-01-01|         2024-04-30|                         452|             241295|       1900-01-01|       2023-12-31|                        2028|              239787|        192
+--------------------+-------------------+-------------------+----------------------------+-------------------+-----------------+-----------------+----------------------------+--------------------+-----------

import org.apache.spark.sql.types._
import org.apache.spark.sql.functions._
dateColumnNames: Array[String] = Array(fecha_insc_date, fecha_nac_date, fecha_mad_date)
statsExprs: Array[org.apache.spark.sql.Column] = Array(count(fecha_insc_date) AS fecha_insc_date_count, min(fecha_insc_date) AS fecha_insc_date_min, max(fecha_insc_date) AS fecha_insc_date_max, count(fecha_insc_da
```

Took 4 seconds. Last updated by anonymous at June 25 2025, 6:43:09 PM.

SPARK JOB FINISHED

```
dfNacidosClean
    .select(min($"fecha_insc_date"), max($"fecha_insc_date"), count($"fecha_insc_date"), countDistinct($"fecha_insc_date")).show
```

```
+-------------------+-------------------+--------------------+------------------------------+
|min(fecha_insc_date)|max(fecha_insc_date)|count(fecha_insc_date)|count(DISTINCT fecha_insc_date)|
+-------------------+-------------------+--------------------+------------------------------+
|         2001-01-01|         2024-04-30|              231333|                           452|
+-------------------+-------------------+--------------------+------------------------------+
```

Took 3 seconds. Last updated by anonymous at June 25 2025, 6:49:24 PM.

SPARK JOB FINISHED

```
dfNacidos.select(
    mode($"anio_nac"),
    median($"anio_nac"),
    stddev($"anio_nac"),
    stddev_pop($"anio_nac"), // divide N
    stddev_samp($"anio_nac")) // divide N - 1
    .show
```

```
+-------------+---------------+--------------------+-------------------+--------------------+
|mode(anio_nac)|median(anio_nac)|stddev_samp(anio_nac)|stddev_pop(anio_nac)|stddev_samp(anio_nac)|
+-------------+---------------+--------------------+-------------------+--------------------+
|         2023|         2023.0|   1.5180530471163742|  1.5180499014759343|   1.5180530471163742|
+-------------+---------------+--------------------+-------------------+--------------------+
```

Took 2 seconds. Last updated by anonymous at June 25 2025, 6:56:02 PM. (outdated)

READY

```
%md
1. clasificar las columnas enteras, seleccionar o ver
2. Ver columnas que se supone que deben ser enteras pero estan como String

QUE VAMOS A HACER
Obtener los valores distintos de una columna
LUEGO
agrupar y contar cuantos de esos valores distintos existen
```