UNIVERSIDAD GALILEO

MAESTRIA INTELIGENCIA DE NEGOCIOS

CIENCIA DE DATOS EN PYTHON

SECCION L

Proyecto Final Pipe line de ingeniería con Python, SQL YAWS

Alumno Willard Alfaro Osiel Gutierrez Carne 20043236 21000958



PROYECTO FINAL

Contenido

1. Alcance del Proyecto	1
1.2 Descripción de fuentes de Información	1
2. Exploración de datos	2
3. Modelo de datos	2
4. Procesamiento de datos	3
Servicio AWS (Amazon)	3
Creación de Base de datos en RDS	3
Creación de bucket en S3 Management console	3
Creación de usuario IAM Management console	4
Data Grip	4
Conexión servicio AWS:	4
Carga de Base de datos (scrips MySQL)	5
Carga de informacion en tablas (scrips MySQL)	7
Código en Python (Anaconda-JupyterLab)	9
Parámetros de conexión RDS	9
Conexión a S3	9
Exploración Base de Datos - Tablas de Pacientes y Enfermedades	10
Procesamiento Y Elaboración De Dimensiones	11
5. Analítica de Datos	13
6. Almacenamiento	15
Almacenamiento de dimensiones en S3	15
Almacenamiento de Tabla de hechos Estancia Hospitalaria en S3	16
7. Gráficas	17



1. Alcance del Proyecto

El presente proyecto tiene como fin desarrollar un pipeline de ingeniería utilizando Python, SQL y AWS como herramientas de desarrollo para una base de datos de recién nacidos de un servicio de neonatología de un hospital nacional. El objetivo es poder explorar dicha base y crear un modelo de datos a partir de la información y combinarla con los datos de Departamentos y Municipios del INE

La base de datos de neonatos será montada en RDS y se conectará con dos archivos externos del INE montados en S3, uno contiene los departamentos y otro los municipios, esto permitirá conocer a los pacientes desde la perspectiva geográfica.

El objetivo del proyecto es poder analizar los diagnósticos, días de estancia hospitalaria, medicamentos utilizados entre otros mediante un modelo elaborado en Phyton.

1.2 Descripción de fuentes de Información

La base de Datos del servicio de neonatología consiste en una tabla General que contiene información de cada recién nacido, tal como: Servicio de donde ingresa, Género, Estado Nutricional, Diagnósticos, Estancia Hospitalaria, entre otros. Esta base de datos permite conocer en la historia el estado de salud de los neonatos y guardar un registro de estos para control y estadística del Departamento de Neonatología.

- a) Se utilizo para el proyecto la base de datos de denominada "RN" con el contenido siguiente:
 - Creación de Bade de datos denominada "RN"
 - Creación de tablas:
 - Dimensión denominada Pacientes
 - Tabla de hechos denominada Enfermedades
 - b) Se utilizaron 2 archivos con extensión CSV
 - Dimensión Departamentos denominado "RNDWHDEPTO.csv"
 - Dimensión Municipios denominado "RNDWHMUNI.csv"



2. Exploración de datos

Para su realización se crearon los siguientes servicios en la nube de amazon (AWS):

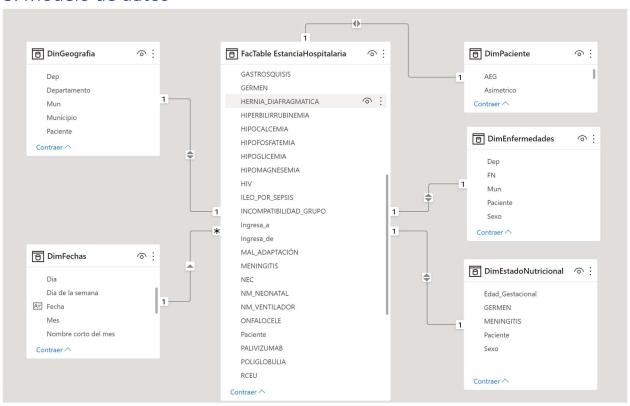
- RDS Management console
- EC2 Management console
- S3 Management console
- IAM Management console

Se efectuó la creación de base de datos, creación de tablas y carga de por medio de código MySQL utilizando el programa DataGrip efectuando la conexión a AWS.

Se cargaron archivos con extensión "CSV" por medio del servicio S3.

Con lo anterior indicado se procedio con las conexiones AWS por medio JupyterLab para la verificación de la información a través de los scrips de Python.

3. Modelo de datos





4. Procesamiento de datos

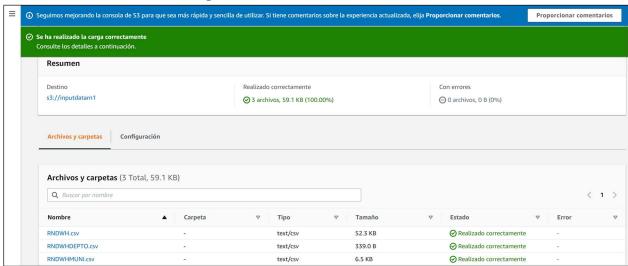
El procesamiento de datos se efectuará a través de los servicios de AWS, SQL, Python y DataGrip.

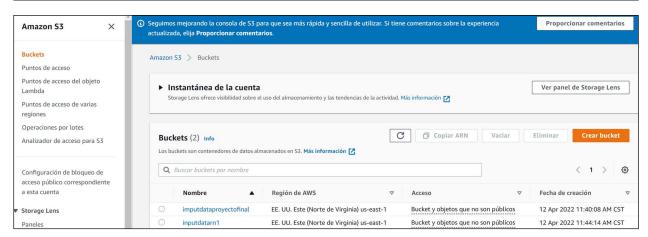
Servicio AWS (Amazon)

Creación de Base de datos en RDS

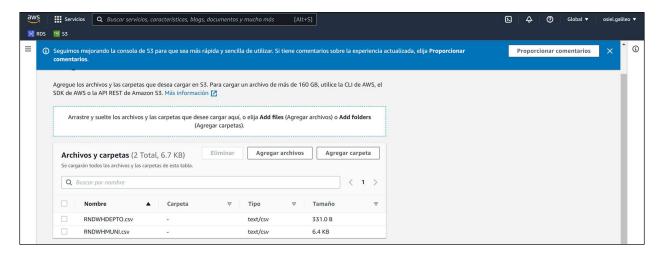


Creación de bucket en S3 Management console









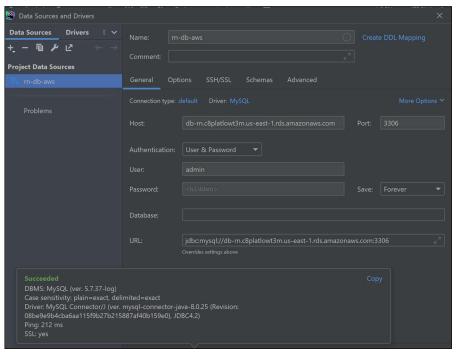
Creación de usuario IAM Management console



Data Grip

Utilización de programa para la creación de bade de datos y tablas.

Conexión servicio AWS:





Carga de Base de datos (scrips MySQL)

--Scrip de creacion de base de datos y tablas CREATE DATABASE RN;

USE RN:

CREATE TABLE PacienteLugar(

Paciente INT NOT NULL,

Dep INT NOT NULL,

Mun INT NOT NULL.

FN datetime NULL,

Sexo nvarchar(255) NULL,

Edad Gestacional float NULL,

Prematurez nvarchar(255) NULL,

PAN float NULL,

AEG nvarchar(255) NULL,

PEG nvarchar(255) NULL,

GEG nvarchar(255) NULL,

Simetrico nvarchar(255) NULL,

Asimetrico nvarchar(255) NULL.

MADRE_ADOLESCENTE nvarchar(255) NULL,

VIVO nvarchar(255) NULL,

Peso Egreso float NULL,

PRIMARY KEY NONCLUSTERED (Paciente),

KEY idx_paciente (Paciente)

) ENGINE = InnoDB DEFAULT CHARSET = utf8mb4;

CREATE TABLE Enfermedades(

Paciente int NOT NULL,

Fecha ingreso datetime NULL,

Ingresa_de nvarchar(255) NULL,

Ingresa_a nvarchar(255) NULL,

SDR TIPO1 nvarchar(255) NULL,

Surfactante nvarchar(255) NULL, NM_NEONATAL nvarchar(255) NULL,

NM_VENTILADOR nvarchar(255) NULL,

SAM nvarchar(255) NULL,

TTRN nvarchar(255) NULL,

MAL_ADAPTACION nvarchar(255) NULL,

DBP nvarchar(255) NULL,

SEPSIS_NEONATAL nvarchar(255) NULL,

SEPSIS_TARDIA nvarchar(255) NULL,

SEPSIS NOSOCOMIAL nvarchar(255) NULL,

GERMEN nvarchar(255) NULL,

MENINGITIS nvarchar(255) NULL,

HIPERBILIRRUBINEMIA nvarchar(255) NULL,

INCOMPATIBILIDAD GRUPO nvarchar(255) NULL,

POLIGLOBULIA nvarchar(255) NULL,

ILEO_POR_SEPSIS nvarchar(255) NULL,

NEC nvarchar(255) NULL,

CARDIPATIA CONGENITA nvarchar(255) NULL,

GASTROSQUISIS nvarchar(255) NULL,

ONFALOCELE nvarchar(255) NULL,

HERNIA DIAFRAGMATICA nvarchar(255) NULL,

ATRESIS_ESOFAGIA nvarchar(255) NULL,

ATRESIA INTESTINAL nvarchar(255) NULL,

ANO IMPERFORADO nvarchar(255) NULL,

SINDROME_DISMORFOGENETICO nvarchar(255) NULL,

ANOMALIAS_ASOCIADAS_TORCH nvarchar(255) NULL,

SINDROME CONVULSIVO nvarchar(255) NULL,

APN nvarchar(255) NULL,

HIV nvarchar(255) NULL,

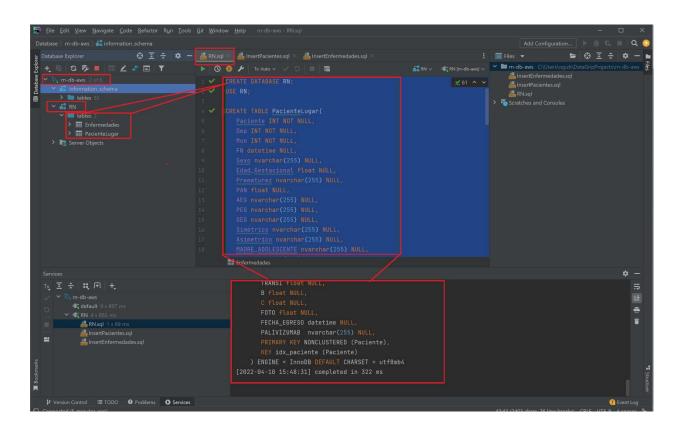
ROP nvarchar(255) NULL,

HIPOGLICEMIA nvarchar(255) NULL,

HIPOFOSFATEMIA nvarchar(255) NULL,



HIPOMAGNESEMIA nvarchar(255) NULL,
HIPOCALCEMIA nvarchar(255) NULL,
RCEU nvarchar(255) NULL,
A1 float NULL,
A2 float NULL,
AISLA float NULL,
TRANSI float NULL,
B float NULL,
C float NULL,
FOTO float NULL,
FOTO float NULL,
FECHA_EGRESO datetime NULL,
PALIVIZUMAB nvarchar(255) NULL,
PRIMARY KEY NONCLUSTERED (Paciente),
KEY idx_paciente (Paciente)
) ENGINE = InnoDB DEFAULT CHARSET = utf8mb4;





Carga de informacion en tablas (scrips MySQL)

Tabla Enfermedades (Resumen de código)

SET AUTOCOMMIT=0;

INSERT INTO Enfermedades

(A1, A2, AISLA, ANO_IMPERFORADO, ANOMALIAS_ASOCIADAS_TORCH, APN, ATRESIA_INTESTINAL, ATRESIS_ESOFAGIA, B, C, CARDIPATIA_CONGENITA, DBP, FECHA_EGRESO, Fecha_ingreso, FOTO, GASTROSQUISIS, GERMEN, HERNIA_DIAFRAGMATICA, HIPERBILIRRUBINEMIA, HIPOCALCEMIA, HIPOFOSFATEMIA, HIPOGLICEMIA, HIPOMAGNESEMIA, HIV, ILEO_POR_SEPSIS,

INCOMPATIBILIDAD_GRUPO, Ingresa_a, Ingresa_de, MAL_ADAPTACION, MENINGITIS, NEC, NM_NEONATAL, NM_VENTILADOR, ONFALOCELE, Paciente, PALIVIZUMAB, POLIGLOBULIA, RCEU, ROP, SAM, SDR_TIPO1, SEPSIS_NEONATAL, SEPSIS_NOSOCOMIAL, SEPSIS_TARDIA,

SINDROME_CONVULSIVO, SINDROME_DISMORFOGENETICO, Surfactante, TRANSI, TTRN) VALUES (0 ,15 ,0 ,'NO' ,'SI' ,'NO' ,'NO

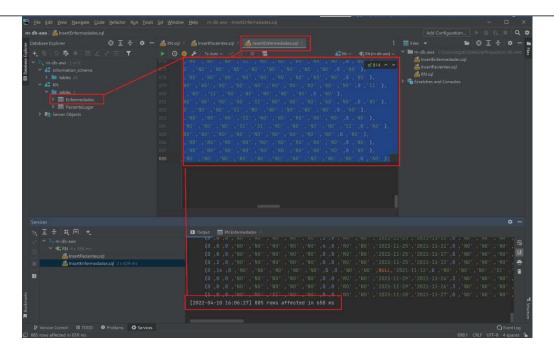
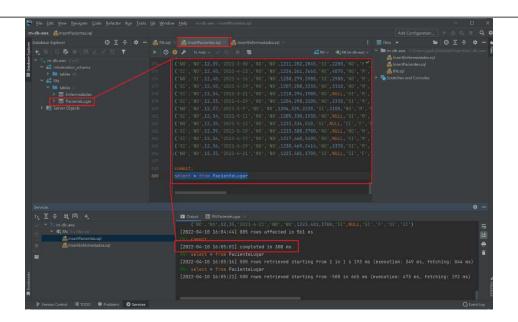




Tabla Pacientes (Resumen de código)

```
SET AUTOCOMMIT = 0;
```

```
INSERT INTO PacienteLugar (AEG, Asimetrico, Dep, Edad Gestacional, FN, GEG,
MADRE ADOLESCENTE, Mun, Paciente, PAN, PEG, Peso Egreso, Prematurez, Sexo, Simetrico, VIVO)
VALUES ('SI'.'NO'.12.39.'2021-7-7'.'NO'.'SI'.1228.497.3150.'NO'.3600.'NO'.'M'.'NO'.'SI').
('NO','NO',12,34,'2021-7-18','NO','NO',1206,508,1380,'SI',NULL,'SI','F','NO','SI'),
('SI','NO',12,37,'2021-7-18','NO','SI',1230,511,3350,'NO',3280,'NO','M','NO','SI'),
('NO','SI',12,34,'2021-7-17','NO','NO',1206,547,1230,'SI',2510,'SI','F','NO','SI'),
('SI','NO',12,39,'2021-8-2','NO','NO',1204,553,3238,'NO',NULL,'NO','F','NO','SI'),
('SI','NO',12,38,'2021-8-6','NO','NO',1201,559,3100,'NO',2890,'NO','F','NO','SI'),
('NO','SI',12,38,'2021-8-1','NO','NO',1214,566,1590,'SI',2020,'NO','M','NO','SI'),
('SI','NO',12,40,'2021-8-21','NO','NO',1217,575,3290,'NO',NULL,'NO','M','NO','NO'),
('SI','NO',12,36,'2021-9-1','NO','NO',1209,603,2230,'NO',NULL,'SI','M','NO','SI'),
('NO','NO',12,37,'2021-9-9','SI','NO',1219,610,4170,'NO',NULL,'NO','F','NO','SI'),
('NO','NO',12,37,'2021-9-14','NO','NO',1214,633,2030,'SI',NULL,'NO','M','SI','SI'),
('NO','SI',12,35,'2021-9-24','NO','NO',1201,651,1510,'SI',NULL,'SI','M','NO','SI'),
('SI','NO',12,36,'2021-10-2','NO','NO',1207,662,2960,'NO',2990,'SI','F','NO','SI'),
('SI','NO',12,36,'2021-10-8','NO','SI',1206,689,2500,'NO',2500,'SI','M','NO','NO'),
('SI','NO',12,38,'2021-10-4','NO','NO',1207,690,2500,'NO',2470,'NO','F','NO','SI'),
('SI','NO',12,37,'2021-10-7','NO','NO',1223,702,3000,'NO',NULL,'NO','F','NO','SI'),
('NO','NO',12,39,'2021-10-7','NO','NO',1202,706,2580,'SI',2640,'NO','F','SI','SI'),
('SI','NO',12,37,'2021-10-9','NO','NO',1211,716,2440,'NO',2440,'NO','M','NO','SI'),
('SI','NO',12,39,'2021-10-30','NO','NO',1204,781,3180,'NO',3150,'NO','F','NO','SI'),
('NO','NO',12,32,'2021-11-7','NO','NO',1207,797,1020,'SI',NULL,'SI','M','SI','SI'),
('NO','NO',12,40,'2021-10-30','NO','NO',1218,812,2580,'SI',2600,'NO','F','SI','SI'), .........................
```





Código en Python (Anaconda-JupyterLab)

Parámetros de conexión RDS

Conexión a S3

```
[233]: import boto3
       import os
       import sys
       import s3fs
       import pandas as pd
[234]: secrete = 'OP1+P/CX3RZ3bVKK7fv8EnsarINatWDVHIRwi7P1'
       access = 'AKIA3ATVHBSGFBVDXKQM'
[236]: s3list = boto3.resource(
          service_name = 's3',
          region_name = 'us-east-2',
           aws_access_key_id = access,
          aws_secret_access_key = secrete
       for bucket in s3list.buckets.all():
           print(bucket.name)
       inputdatarn
[238]: S3_BUCKET_NAME = 'inputdatarn'
       for obj in s3list.Bucket(S3_BUCKET_NAME).objects.all():
           print(obj.key)
       RNDWHDEPTO.csv
       RNDWHMUNI.csv
```



```
import os
import pandas as pd
from smart_open import smart_open

secrete = 'OP1+P/CX3RZ3bVKK7fv8EnsarINatWDVHIRwi7P1'
access = 'AKIA3ATVHBSGFBVDXKQM'

bucket_name = 'inputdatarn'
object_key_depto = 'RNDWHDEPTO.csv'
object_key_muni = 'RNDWHMUNI.csv'

path_depto = 's3://{}:{}@{}/{}'.format(access, secrete, bucket_name, object_key_depto)
path_muni = 's3://{}:{}@{}/{}'.format(access, secrete, bucket_name, object_key_muni)
path_pac = 's3://{}:{}@{}/{}'.format(access, secrete, bucket_name, object_key_muni)
path_pac = 's3://{}:{}@{}/{}'.format(access, secrete, bucket_name, object_key_pac)

dfDepto = pd.read_csv(smart_open(path_depto))
dfMuni = pd.read_csv(smart_open(path_muni))
```

Exploración Base de Datos - Tablas de Pacientes y Enfermedades

Importar Datos de RDS a Pandas

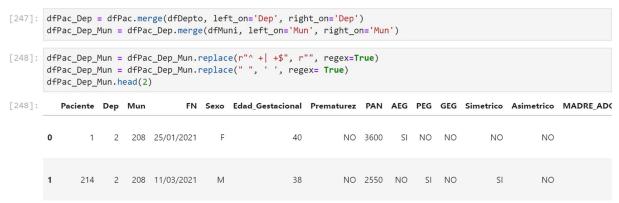


2 rows × 49 columns



Importar Datos de S3 a Pandas

limpieza de Datos de Depto y Muni debido a que se encontró espacios en blanco al principio y final



Procesamiento Y Elaboración De Dimensiones

Dimensión Geográfica

```
[249]: DimGeografica = dfPac_Dep_Mun.loc[: , ['Paciente', 'Dep', 'Departamento', 'Mun', 'Municipio']]

[249]: Paciente Dep Departamento Mun Municipio

0 1 2 El Progreso 208 San Antonio la Paz

1 214 2 El Progreso 208 San Antonio la Paz
```

Dimensión Paciente



Dimensión Enfermedad

[251]:		Paciente	NM_NEONATAL	SEPSIS_NEONATAL	SEPSIS_TARDIA	HIPERBILIRRUBINEMIA	HIV	HIPOFOSFATEMIA
	0	1	SI	SI	NO	NO	NO	NO
	1	2	NO	NO	NO	NO	NO	NO

Dimensión Tiempo

```
[252]: DimTiempo = dfPaciente.loc[: , ['FN']]
DimTiempo['year'] = pd.DatetimeIndex(DimTiempo['FN']).year
DimTiempo['month'] = pd.DatetimeIndex(DimTiempo['FN']).month
DimTiempo['quarter'] = pd.DatetimeIndex(DimTiempo['FN']).quarter
DimTiempo['day'] = pd.DatetimeIndex(DimTiempo['FN']).day
DimTiempo['dayofweek'] = pd.DatetimeIndex(DimTiempo['FN']).dayofweek
DimTiempo.head()
```

[252]:		FN	year	month	quarter	day	dayofweek
	0	2021-01-25	2021	1	1	25	0
	1	2021-01-22	2021	1	1	22	4
	2	2021-01-20	2021	1	1	20	2
	3	2021-01-25	2021	1	1	25	0
	4	2021-01-09	2021	1	1	9	5

Dimensión Estado Nutricional

[254]: DimEstadoNutricional = dfPaciente.loc[: , ['Paciente', 'PAN', 'Edad_Gestacional', 'AEG', 'PEG', 'GEG', 'Peso_Egr DimEstadoNutricional.head(2)

[254]:		Paciente	PAN	${\bf Edad_Gestacional}$	AEG	PEG	GEG	Peso_Egreso	Simetrico	Asimetrico
	0	1	3600.0	40.0	SI	NO	NO	3600.0	NO	NO
	1	2	3000.0	39.0	SI	NO	NO	3100.0	NO	NO



Tabla de Hechos Estancia Hospitalaria

```
[255]: Fact_Est_Hospi = dfEnfermedad.merge( DimGeografica, left_on='Paciente', right_on='Paciente')
                           Fact_Est_Hospi = Fact_Est_Hospi.merge(DimPaciente, left_on='Paciente', right_on='Paciente')
                           Fact_Est_Hospi = Fact_Est_Hospi.loc[: , [ 'Paciente','Dep','Mun','FN','A1','A2', 'AISLA', 'TRANSI','B','C','FOTO
                           Fact_Est_Hospi['Estancia'] = Fact_Est_Hospi['A1'] + Fact_Est_Hospi['A2'] + Fact_Est_Hospi['AISLA']+ Fact_Est_Est_Hospi['AISLA']+ Fact_Est_Est_Hospi['AISLA']+ Fact_Est_Hospi['AISLA']+ Fact_Est_Hospi['AISLA']+ Fact_Est_Hospi['AISLA']+ Fact_Est_Hospi['AISLA']+ Fact_Est_Hospi['AISLA']+ Fact_Est_Hospi['AISLA']+ Fact_Est_Hospi['AISLA']+ Fact_Est_Hosp
                           Fact_Est_Hospi.head(5)
                                                                                                                                FN A1 A2 AISLA TRANSI
                                                                                                                                                                                                                                        B C FOTO Estancia
                                    Paciente Dep Mun
                           0
                                                                           2
                                                                                      208 2021-01-25 0.0 15.0
                                                                                                                                                                                           0.0
                                                                                                                                                                                                                      0.0 0.0 0.0
                                                                                                                                                                                                                                                                         0.0
                                                                                                                                                                                                                                                                                                    15.0
                                                                           1
                                                                                        109 2021-01-22 0.0
                                                                                                                                                                 0.0
                                                                                                                                                                                                                      0.0 0.0 0.0
                                                                                                                                                                                                                                                                                                      3.0
                           2
                                                                      11 1103 2021-01-20 3.0
                                                                                                                                                             0.0
                                                                                                                                                                                           0.0
                                                                                                                                                                                                                      0.0 0.0 0.0
                                                                                                                                                                                                                                                                         3.0
                                                                                                                                                                                                                                                                                                      6.0
                                                                        13 1308 2021-01-25 1.0
                                                                                                                                                             0.0
                                                                                                                                                                                           0.0
                                                                                                                                                                                                                      0.0 1.0 0.0
                                                                                                                                                                                                                                                                         0.0
                                                                                                                                                                                                                                                                                                      2.0
                                                                        17 1703 2021-01-09 12.0 0.0
                                                                                                                                                                                                                      0.0 6.0 0.0
                                                                                                                                                                                                                                                                                                   18.0
```

5. Analítica de Datos

En el presente apartado se dará solución a las interrogantes planteadas en el presente proyecto siendo las siguientes:

1. ¿Cuál es el Promedio de la Edad Gestacional por Género?

2. ¿Cuál es el Promedio de la Estancia Hospitalaria?



3. ¿Qué servicio es el que más estancia hospitalaria genera?

```
[260]: Prom = Fact_Est_Hospi.mean() ['A1'], Fact_Est_Hospi.mean() ['A2'], Fact_Est_Hospi.mean() ['AISLA'], Fact_Est_Ho
                   4
                  C:\Users\Willard\AppData\Local\Temp/ipykernel_4012/2661206570.py:1: FutureWarning: DataFrame.mean and DataFrame.
                   median with numeric_only=None will include datetime64 and datetime64tz columns in a future version.
                      Prom = Fact_Est_Hospi.mean() ['A1'], Fact_Est_Hospi.mean() ['A2'], Fact_Est_Hospi.mean() ['AISLA'] , Fact_Est_
                  Hospi.mean() ['TRANSI'], Fact_Est_Hospi.mean() ['B'], Fact_Est_Hospi.mean() ['C'], Fact_Est_Hospi.mean() ['FOT
                  0']
[260]: (2.7864406779661017.
                     1.8937853107344633,
                     0.32090395480225986,
                     0.04858757062146893,
                     3.8192090395480225,
                     0.6045197740112994.
                     1.0045197740112994)
[261]: PromMax = max(Fact_Est_Hospi.mean() ['AI'], Fact_Est_Hospi.mean() ['AISLA'], Fact_Est_Hospi.
                   4
                  C:\Users\Willard\AppData\Local\Temp/ipykernel_4012/2343260451.py:1: FutureWarning: DataFrame.mean and DataFrame.
                  median with numeric_only=None will include datetime64 and datetime64tz columns in a future version.
                       PromMax = max(Fact_Est_Hospi.mean() ['A1'], Fact_Est_Hospi.mean() ['A2'], Fact_Est_Hospi.mean() ['AISLA'] , Fa
                   ct_Est_Hospi.mean() ['TRANSI'], Fact_Est_Hospi.mean() ['B'], Fact_Est_Hospi.mean() ['C'], Fact_Est_Hospi.mean()
                  ['FOTO'])
[261]: 3.8192090395480225
```

4. ¿Qué Edad Gestacional es la que mayor estancia hospitalaria tiene?

```
[262]: dfEstNut = Fact_Est_Hospi.merge(DimEstadoNutricional, left_on='Paciente', right_on='Paciente')
       MaxEdadGesta = dfEstNut.groupby('Edad_Gestacional').mean()['Estancia'].sort_values(ascending=True)
       MaxEdadGesta.head(200)
[262]: Edad_Gestacional
       23.0
                1.000000
       25.0
                1.000000
       27.0
                2.000000
                4.500000
       26.0
                4.571429
       28.0
       39.0
                6.200000
                6.666667
       41.0
       42.0
                7.500000
       40.0
                7.774510
       38.0
                8.528662
       37.0
                9.020833
       30.0
                9.588235
       32.0
               11.755556
       36.0
               12.210938
       31.0
               17.000000
               17.402985
       34.0
       35.0
               17.531646
       29.0
               18.666667
               19.933333
       33.0
       Name: Estancia, dtype: float64
```



5. ¿Qué peso al nacer genera mayor estancia hospitalaria?

```
[263]: dfEstNut = Fact_Est_Hospi.merge(DimEstadoNutricional, left_on='Paciente', right_on='Paciente')
       MaxPesoEstancia = dfEstNut.groupby('PAN').mean()['Estancia'].sort_values(ascending=True)
       MaxPesoEstancia.head(200)
[263]: PAN
       5340.0
                  0.000000
       1477.0
                  0.000000
                  0.000000
       1931.0
       2260.0
                  0.000000
                 0.000000
       1164.0
       2130.0
                 9.000000
       1610.0
                  9.000000
       2410.0
                 9.600000
       3430.0
                  9.833333
       3610.0 10.000000
       Name: Estancia, Length: 200, dtype: float64
```

6. Almacenamiento

Almacenamiento de dimensiones en S3

```
[264]: s3 = boto3.client('s3',
                        aws_access_key_id= access,
                        aws_secret_access_key= secrete)
       #Crear Csv's para cada Dimensión
       Csv_DimPaciente = DimPaciente
       Csv_DimPaciente.to_csv('DimPaciente.csv')
       Csv_DimEnfermedad = DimEnfermedad
       Csv_DimEnfermedad.to_csv('DimEnfermedad.csv')
       Csv_DimGeografica = DimGeografica
       Csv_DimGeografica.to_csv('DimGeografica.csv')
       Csv_DimTiempo = DimTiempo
       Csv_DimTiempo.to_csv('DimTiempo.csv')
       Csv_DimEstadoNutricional = DimEstadoNutricional
       Csv_DimEstadoNutricional.to_csv('DimEstadoNutricional.csv')
       # Cargar csv a s3
       s3.upload_file(
            'DimPaciente.csv',
            'inputdatarn',
           'DimPaciente.csv')
       s3.upload_file(
            'DimEnfermedad.csv',
            'inputdatarn',
           'DimEnfermedad.csv')
       s3.upload_file(
           'DimGeografica.csv',
            'inputdatarn',
            'DimGeografica.csv')
       s3.upload_file(
            'DimTiempo.csv',
            'inputdatarn',
            'DimTiempo.csv')
       s3.upload_file(
            'DimEstadoNutricional.csv',
            'inputdatarn',
            'DimEstadoNutricional.csv')
```



Almacenamiento de Tabla de hechos Estancia Hospitalaria en S3

```
[265]: s3 = boto3.client('s3',
                        aws_access_key_id= access,
                        aws_secret_access_key= secrete)
       Csv_Fact_Est_Hospi = Fact_Est_Hospi
       Csv_Fact_Est_Hospi.to_csv('Fact_Est_Hospi.csv')
       s3.upload_file(
           'Fact_Est_Hospi.csv',
           'inputdatarn',
           'Fact_Est_Hospi.csv')
[266]: s3list = boto3.resource(
          service_name = 's3',
region_name = 'us-east-2',
          aws_access_key_id = access,
           aws_secret_access_key = secrete
       for obj in s3list.Bucket('inputdatarn').objects.all():
           print(obj.key)
       DimEnfermedad.csv
       DimEstadoNutricional.csv
       DimGeografica.csv
       DimPaciente.csv
       DimTiempo.csv
       Fact_Est_Hospi.csv
       RNDWHDEPTO.csv
       RNDWHMUNI.csv
```



7. Gráficas

```
[270]: reciennacidos_x_mes = DimTiempo.groupby('month').count()['FN'].sort_values(ascending=True)
       reciennacidos_x_mes
[270]: month
       12
       1
              36
       8
              40
       7
              58
       6
              72
       2
              74
              90
       11
              92
       5
              93
             102
             104
            120
       10
       Name: FN, dtype: int64
[217]: reciennacidos_x_mes.plot.bar()
[217]: <AxesSubplot:xlabel='month'>
        600
```

