Pontificia Universidad Javeriana

Samuel Peña García

Tomás De Aza Márquez

Juan Sebastián Córdoba Valderrama

Parcial II

Procesamiento de Datos a Gran Escala

2024-10

Bogotá, Colombia

Índice

Fase Inicial **1**
Selección de los datos 1
Preguntas propuestas 4
Exploración de los datos 4
Preparación de los datos 7
Visualización Inicial 8
Problema **11**
Descripción del problema 11
Problemas con los datos 11
Problemas de calidad de datos 11
Creación de variables derivadas 14
Analítica descriptiva de los datos 14
Implementación de técnicas de Machine Learning **15**
Entrenamiento de los modelos 17
Métricas de rendimiento 18
Solución de las preguntas propuestas 20
Conclusiones, observaciones y recomendaciones 21
Referencias **22**

Fase Inicial

Selección de los datos

Para la selección de los datos era necesario tener claro qué se iba a buscar y dónde. En este caso, el qué eran conjuntos de datos referentes a natalidad en municipios de Colombia con variables razonables, para que se puedan realizar varios análisis. En cuanto al dónde, la fuente más fiable era el DANE, Departamento Administrativo Nacional de Estadística de Colombia.

De esta manera, se realizó una búsqueda de conjuntos de datos que tuvieran como palabra clave "nacimientos", de esta forma, se encontraron los siguientes:

Nombre	Detalle	Enlace
Nacimientos	Registro de los nacimientos de personas residentes en el municipio de Medellín, sexo, edad de la madre y comuna.	Enlace
Nacimientos 2021-2022	Conjunto de datos que contiene el listado de recién nacidos en el Municipio de La Dorada en el año 2021 y mitad del año 2022.	Enlace

Estos dos conjuntos de datos cumplen con los requerimientos mencionados anteriormente, hacen referencia a nacimientos en Medellín y La Dorada, respectivamente y tienen 40 y 50 columnas respectivamente, lo que permite el análisis esperado. Asimismo, estos conjuntos de datos son de municipios importantes en Colombia. Medellín tiene una población de 2,945,034 habitantes y un aporte de más del 7% del PIB de Colombia —con corte a 2022—, por su parte, La Dorada, Caldas tiene una población de 75,319 habitantes —con corte a 2022—. Siendo así, vemos que estos datos son realmente una fuente de información importantísima para el desarrollo del proyecto.

A continuación, se presentan los diccionarios de datos de los dos conjuntos de datos:

Diccionario de datos Medellín		
Campo	**Tipo**	**Descripción**
ID	Número	Número identificador
AREANAC	Cadena	Área del nacimiento
COD_INSP	Cadena	Centro poblado del nacimiento (inspección, corregimiento o caserío)
SIT_PARTO	Cadena	Sitio del parto
OTRO_SIT	Cadena	Otro sitio, ¿cuál?
NOM_INST	Cadena	Nombre de la institución de salud
COD_INST	Cadena	Código de la institución de salud

SEXO	Cadena	Sexo del nacido vivo
PESO_NAC	Número	Peso del nacido vivo, al nacer
TALLA_NAC	Número	Talla del nacido vivo, al nacer
FECHA_NAC	Fecha	Fecha del nacimiento
ANO	Cadena	Año de la ocurrencia
MES	Cadena	Mes de la ocurrencia
ATEN_PAR	Cadena	El parto fue atendido por
OTRPARATX	Cadena	El parto fue atendido por otra persona, ¿cuál?
T_GES	Cadena	Tiempo de gestación del nacido vivo
NUMCONSUL	Cadena	Número de consultas prenatales que tuvo la madre del nacido vivo
TIPO_PARTO	Cadena	Tipo de parto de este nacimiento
MUL_PARTO	Cadena	Multiplicidad del embarazo
IDHEMOCLAS	Cadena	Hemoclasificación del nacido vivo: Grupo sanguíneo
IDFACTORRH	Cadena	Hemoclasificación del nacido vivo: Factor RH
IDPERTET	Cadena	De acuerdo con la cultura, pueblo o rasgos físicos, el fallecido era o se reconocía como
NOM_PUEB	Cadena	¿A cuál pueblo indígena pertenece?
EDAD_MADRE	Número	Edad de la madre a la fecha del parto
EST_CIVM	Cadena	Estado conyugal de la madre
NIV_EDUM	Cadena	Último año de estudio que aprobó la madre
CODPTORE	Cadena	Departamento de residencia habitual de la madre
CODMUNRE	Cadena	Municipio de residencia habitual de la madre
COD_BARRIRES	Cadena	Barrio de residencia del fallecido
N_HIJOSV	Número	Número de hijos nacidos vivos que ha tenido la madre, incluido el presente
FECHA_NACM	Fecha	Fecha de nacimiento del anterior hijo nacido vivo
N_EMB	Número	Número de embarazos, incluido el presente
SEG_SOCIAL	Cadena	Régimen de seguridad social en salud de la madre
IDCLASADMI	Cadena	Entidad administradora en salud a la que pertenece la madre
NOMCLASAD	Cadena	Nombre de la administradora en salud a la que pertenece la madre
CODCLASAD	Cadena	Código de la administradora en salud a la que pertenece la madre
EDAD_PADRE	Número	Edad del padre en años cumplidos a la fecha del nacimiento de este hijo
NIV_EDUP	Cadena	Último año de estudio que aprobó el padre
BARRIO_RES	Cadena	Barrio de residencia de la madre del nacido vivo
COMUNA_RES	Cadena	Comuna de residencia de la madre del nacido vivo

Diccionario de datos La Dorada		
Campo	**Tipo**	**Descripción**
NÚMERO CERTIFICADO	Número	Número de certificado
DEPARTAMENTO	Cadena	Departamento del nacimiento
MUNICIPIO	Cadena	Municipio del nacimiento
AREA NACIMIENTO	Cadena	Área del nacimiento
INSPECCION CORREGIMIENTO O CASERIO NACIMIENTO	Cadena	Inspección, corregimiento o caserío del nacimiento
SITIO NACIMIENTO	Cadena	Sitio del nacimiento
CÓDIGO INSTITUCIÓN	Número	Código de la institución del nacimiento
NOMBRE INSTITUCIÓN	Cadena	Nombre de la institución del nacimiento
SEXO	Cadena	Sexo del recién nacido
PESO (Gramos)	Número	Peso en gramos del recién nacido
TALLA (Centímetros)	Número	Talla en centímetros del recién nacido
FECHA NACIMIENTO	Cadena	Fecha de nacimiento recién nacido
HORA NACIMIENTO	Cadena	Hora de nacimiento recién nacido
PARTO ATENDIDO POR	Cadena	Médico y/o persona que atendió el parto
TIEMPO DE GESTACIÓN	Número	Tiempo de gestación
NÚMERO CONSULTAS PRENATALES	Número	Número de consultas prenatales
TIPO PARTO	Cadena	Tipo de parto
MULTIPLICIDAD EMBARAZO	Cadena	Multiplicidad del embarazo
GRUPO SANGUÍNEO	Cadena	Grupo sanguíneo
FACTOR RH	Cadena	Factor Rh
PERTENENCIA ÉTNICA	Cadena	Pertenencia étnica
NOMBRES MADRE	Cadena	Nombres de la madre del recién nacido
APELLIDOS MADRE	Cadena	Apellidos de la madre del recién nacido
EDAD MADRE	Cadena	Edad de la madre del recién nacido
ESTADO CONYUGAL MADRE	Cadena	Estado conyugal de la madre del recién nacido
NIVEL EDUCATIVO MADRE	Cadena	Nivel educativo de la madre del recién nacido
ULTIMO AÑO APROBADO MADRE	Cadena	Último año de estudio cursado por la madre del recién nacido

PAÍS RESIDENCIA	Cadena	País de residencia
DEPARTAMENTO RESIDENCIA	Cadena	Departamento de residencia
MUNICIPIO RESIDENCIA	Cadena	Municipio de residencia
AREA RESIDENCIA	Cadena	Área de residencia
BARRIO	Cadena	Barrio de residencia
DIRECCIÓN	Cadena	Dirección de residencia
CENTRO POBLADO	Cadena	Centro poblado de residencia
RURAL DISPERSO	Cadena	Rural disperso
NÚMERO HIJOS NACIDOS VIVOS	Número	Número de hijos nacidos vivos
FECHA ANTERIOR HIJO NACIDO VIVO	Cadena	Fecha de nacimiento del hijo anterior nacido vivo
NÚMERO EMBARAZOS	Número	Número de embarazos
RÉGIMEN SEGURIDAD	Cadena	Régimen de seguridad
TIPO ADMINISTRADORA	Cadena	Tipo de administradora

Preguntas propuestas

Hay que tener en cuenta la información que presentan los conjuntos de datos y el contexto de la sociedad colombiana, donde hay un alto grado de embarazos a temprana edad. De acuerdo con Profamilia, en 2022 se presentaron 4,169 nacimientos con madres entre los 10 y 14 años y 93,096 en mujeres adolescentes entre 15 y 19 años. Es evidente que esto representa un problema para la sociedad colombiana, y en tanto esto, es necesario encontrar soluciones que puedan visibilizar este problema para poder atacarlo de manera correcta.

Teniendo en cuenta esto, se plantearon las siguientes preguntas:

- ¿Es posible hacer un buen sistema de detección de embarazos a temprana edad a través de modelos de Machine Learning?
- ¿Es posible encontrar a través de gráficos cuál de los dos municipios requiere más atención frente al problema de los embarazos a temprana edad?

Exploración de los datos

Para la exploración de los datos se optó por revisar de forma preliminar los datos, para posteriormente revisar las dimensiones y realizar un análisis estadístico.

Revisión preliminar Medellín:

```
mde_df.display()
```

(2) Spark Jobs

Table

	ID	AREANAC	COD_INSP	SIT_PARTO	OTRO_SIT	NOM_INST	COD_INST	SEXO	PESC
1	1	1	null	1	null	PROMOTORA MEDICA LAS AMERICAS S.A	050010212601	2	1070
2	2	1	null	1	null	CLINICA DEL PRADO S.A.	050010464801	2	2900
3	3	1	null	1	null	CLINICA UNIVERSITARIA BOLIVARIANA	050010344803	2	2510
4	4	1	null	1	null	UNIDAD HOSPITALARIA DE MANRRIQUE HERMENEGILDO DE FEX	050010217804	1	3400
5	5	1	null	1	null	UNIDAD HOSPITALARIA DE MANRRIQUE HERMENEGILDO DE FEX	050010217804	1	3760
6	6	1	null	1	null	E.S.E. HOSPITAL GENERAL DE MEDELL?N LUZ CASTRO DE GUTIERREZ	050010214401	1	2943

```
mde_df.display()
```

(2) Spark Jobs

Table

	PESO_NAC	TALLA_NAC	FECHA_NAC	ANO	MES	ATEN_PAR	OTRPARATX	T_GES	NUMCONSUL	TIPO_PARTO	MUL_PARTO	IDHEMOCLAS	IDFACTORRH	IDPERTET	NOM_PL
1	1070	38	23/04/2012	2012	4	1	null	30	8	2	1	2	1	null	null
2	2900	47	30/05/2012	2012	5	1	null	40	6	2	1	3	1	null	null
3	2510	48	04/06/2012	2012	6	1	null	36	7	1	1	3	1	null	null
4	3400	50	25/09/2012	2012	9	1	null	41	8	2	1	2	1	null	null
5	3760	53	17/04/2012	2012	4	1	null	39	8	1	1	1	1	null	null
6	2943	48	10/08/2012	2012	8	1	null	38	7	1	1	4	1	null	null

```
mde_df.display()
```

(2) Spark Jobs

Table

	NOM_PUEB	EDAD_MADRE	EST_CIVM	NIV_EDUM	CODPTORE	CODMUNRE	COD_BARRIRES	N_HIJOSV	FECHA_NACM	N_EMB	SEG_SOCIAL	IDCLASADMI	NOMCLASAD
2	null	21	1	3	5	1	310	1	01/01/1900	1	1	1	SALUD TOTAL S.A. ENTIDAD PROMOTORA DE S
3	null	37	6	8	5	1	1209	2	25/09/2011	2	5	null	null
4	null	22	1	5	5	1	808	1	01/01/1900	1	5	null	null
5	null	40	1	3	5	1	1020	3	06/08/1996	3	2	2	COMFAMA ANTIOQUIA- CAJA DE COMPENSAC
6	null	36	1	9	5	1	1020	1	01/01/1900	1	2	2	COMFAMA ANTIOQUIA- CAJA DE COMPENSAC

```
mde_df.display()
```

(2) Spark Jobs

Table

	NOMCLASAD	CODCLASAD	EDAD_PADRE	NIV_EDUP	BARRIO_RES	COMUNA_RES
2	SALUD TOTAL S.A. ENTIDAD PROMOTORA DE SALUD	EPS002	25	99	Versalles N.2	03 Manrique
3	null	null	34	8	Santa Mónica	12 La America
4	null	null	24	99	Enciso	08 Villa Hermosa
5	COMFAMA ANTIOQUIA- CAJA DE COMPENSACION FAMILIAR DE ANTIOQUIA	CCF002	41	2	San Diego	10 La Candelaria
6	COMFAMA ANTIOQUIA- CAJA DE COMPENSACION FAMILIAR DE ANTIOQUIA	CCF002	37	2	San Diego	10 La Candelaria

Revisión preliminar La Dorada:

```
ldr_df.display()
```

Table

	NÚMERO CERTIFICADO	DEPARTAMENTO	MUNICIPIO	AREA NACIMIENTO	INSPECCION CORREGIMIENTO O CASERIO NACIMIENTO	SITIO NACIMIENTO	CÓDIGO INSTITUCIÓN	NOMBRE I
1	162060770	CALDAS	LA DORADA	CABECERA MUNICIPAL	null	INSTITUCIÓN DE SALUD	173800051901	173800051
2	162060788	CALDAS	LA DORADA	CABECERA MUNICIPAL	null	INSTITUCIÓN DE SALUD	173800051901	173800051
3	162060795	CALDAS	LA DORADA	CABECERA MUNICIPAL	null	INSTITUCIÓN DE SALUD	173800051901	173800051
4	162060806	CALDAS	LA DORADA	CABECERA MUNICIPAL	null	INSTITUCIÓN DE SALUD	173800051901	173800051
5	162060813	CALDAS	LA DORADA	CABECERA MUNICIPAL	null	INSTITUCIÓN DE SALUD	173800051901	173800051
6	162060820	CALDAS	LA DORADA	CABECERA MUNICIPAL	null	INSTITUCIÓN DE SALUD	173800051901	173800051
7	162060838	CALDAS	LA DORADA	CABECERA MUNICIPAL	null	INSTITUCIÓN DE SALUD	173800051901	173800051

```
ldr_df.display()
```

Table

	NOMBRE INSTITUCIÓN	SEXO	PESO (Gramos)	TALLA (Centimetros)	FECHA NACIMIENTO	HORA NACIMIENTO	PARTO ATENDIDO POR	TIEMPO DE GESTACIÓN	NÚMERO CONSULTAS PRENATALES	TI
1	173800051901 ESE HOSPITAL SAN FELIX	MASCULINO	3410	51	2021-01-01T05:00:00Z	1899-12-31T06:20:00Z	MÉDICO	40	8	ES
2	173800051901 ESE HOSPITAL SAN FELIX	MASCULINO	3002	49	2021-01-01T05:00:00Z	1899-12-31T14:54:00Z	MÉDICO	40	7	ES
3	173800051901 ESE HOSPITAL SAN FELIX	FEMENINO	2215	45	2021-01-01T05:00:00Z	1899-12-31T17:14:00Z	MÉDICO	38	2	CE
4	173800051901 ESE HOSPITAL SAN FELIX	FEMENINO	2875	46	2021-01-01T05:00:00Z	1900-01-01T03:08:16Z	MÉDICO	37	4	ES
5	173800051901 ESE HOSPITAL SAN FELIX	FEMENINO	3265	50	2021-01-02T05:00:00Z	1899-12-31T09:00:00Z	MÉDICO	39	5	ES
6	173800051901 ESE HOSPITAL SAN FELIX	MASCULINO	2575	49	2021-01-03T05:00:00Z	1899-12-31T12:04:00Z	MÉDICO	39	3	ES
7	173800051901 ESE HOSPITAL SAN FELIX	FEMENINO	3520	50	2021-01-03T05:00:00Z	1899-12-31T17:05:00Z	MÉDICO	39	5	ES

```
ldr_df.display()
```

Table

	TIPO PARTO	MULTIPLICIDAD EMBARAZO	GRUPO SANGUINEO	FACTOR RH	PERTENENCIA ÉTNICA	NOMBRES MADRE	APELLIDOS MADRE	EDAD MADRE	ESTAD
1	ESPONTÁNEO	SIMPLE	O	POSITIVO	NINGUNO DE LOS ANTERIORES	LEYDI TATIANA	ORTIZ GARZON	21(4)	NO
2	ESPONTÁNEO	SIMPLE	A	POSITIVO	NINGUNO DE LOS ANTERIORES	DAYANNA ANDREA	GUERRA CIFUENTES	20(4)	NO
3	CESÁREA	SIMPLE	O	POSITIVO	NINGUNO DE LOS ANTERIORES	CAROLAY	VALENZUELA REINA	21(4)	EST
4	ESPONTÁNEO	SIMPLE	O	POSITIVO	NINGUNO DE LOS ANTERIORES	GLORI YINETH	VARGAS BUSTOS	22(4)	NO
5	ESPONTÁNEO	SIMPLE	O	POSITIVO	NINGUNO DE LOS ANTERIORES	TATIANA	SANCHEZ AGUIRRE	19(4)	NO
6	ESPONTÁNEO	SIMPLE	O	POSITIVO	NINGUNO DE LOS ANTERIORES	NATALIA	LLANOS OCAMPO	24(4)	NO
7									

```
ldr_df.display()
```

Table

	ESTADO CONYUGAL MADRE	NIVEL EDUCATIVO MADRE	ULTIMO AÑO APROBADO MADRE	PAÍS RESIDENCIA	DEPARTAMENTO RESIDENCIA	MUNICIPIO RESIDENCIA	AREA RESIDEN
1	NO ESTÁ CASADA Y LLEVA DOS AÑOS O MÁS VIVIENDO CON SU PAREJA	MEDIA ACADÉMICA O CLÁSICA	11	COLOMBIA	CALDAS	LA DORADA	CABECERA
2	NO ESTÁ CASADA Y LLEVA MENOS DE DOS AÑOS VIVIENDO CON SU PAREJA	MEDIA ACADÉMICA O CLÁSICA	10	COLOMBIA	CUNDINAMARCA	PUERTO SALGAR	CABECERA
3	ESTÁ SOLTERA	BÁSICA SECUNDARIA	9	COLOMBIA	CALDAS	LA DORADA	CABECERA
4	NO ESTÁ CASADA Y LLEVA DOS AÑOS O MÁS VIVIENDO CON SU PAREJA	MEDIA ACADÉMICA O CLÁSICA	11	COLOMBIA	CALDAS	LA DORADA	CABECERA
5	NO ESTÁ CASADA Y LLEVA DOS AÑOS O MÁS VIVIENDO CON SU PAREJA	MEDIA ACADÉMICA O CLÁSICA	11	COLOMBIA	CALDAS	NORCASIA	RURAL DISP
6	NO ESTÁ CASADA Y LLEVA DOS AÑOS O MÁS VIVIENDO CON SU PAREJA	BÁSICA PRIMARIA	5	COLOMBIA	CALDAS	LA DORADA	CABECERA
7							

```
ldr_df.display()
```

Table

	AREA RESIDENCIA	BARRIO	DIRECCIÓN	CENTRO POBLADO	RURAL DISPERSO	NÚMERO
1	CABECERA MUNICIPAL	EL CABRERO	KR 13 17-45	null	null	2
2	CABECERA MUNICIPAL	ALTO BUENOS AIRES	CL 12 5 - 06	null	null	1
3	CABECERA MUNICIPAL	LAS FERIAS	AC TA B2 APT 502	null	null	1
4	CABECERA MUNICIPAL	LA CIUDADELA	CL TERRAZA F BLOQUE 2 APTO 402	null	null	1
5	RURAL DISPERSO	null	null	null	ISAZA	2
6	CABECERA MUNICIPAL	LIBERO VARIANTE	CL 16A 16-20	null	null	2
7						

```
ldr_df.display()
```

Table

	NÚMERO HIJOS NACIDOS VIVOS	FECHA ANTERIOR HIJO NACIDO VIVO	NÚMERO EMBARAZOS	RÉGIMEN SEGURIDAD	TIPO ADMINISTRADORA	NOMBRE ADMINISTRADORA
1	2	'27/05/2018	2	SUBSIDIADO	ENTIDAD PROMOTORA DE SALUD SUBSIDIADO	CONVIDA - ARS CONVIDA
2	1	'1/01/1900	1	SUBSIDIADO	ENTIDAD PROMOTORA DE SALUD SUBSIDIADO	E.P.S. FAMISANAR LTDA.-CM
3	1	'1/01/1900	1	SUBSIDIADO	ENTIDAD PROMOTORA DE SALUD SUBSIDIADO	ASMET SALUD ESS - ASOCIACION MUTUAL LA ESPERANZA
4	1	'1/01/1900	1	SUBSIDIADO	ENTIDAD PROMOTORA DE SALUD SUBSIDIADO	E.P.S. FAMISANAR LTDA.-CM
5	2	'19/03/2019	2	SUBSIDIADO	ENTIDAD PROMOTORA DE SALUD SUBSIDIADO	LA NUEVA EPS S.A.-CM
6	2	'15/07/2015	2	CONTRIBUTIVO	ENTIDAD PROMOTORA DE SALUD	NUEVA EPS SA
7	1	'1/01/1900	1	SUBSIDIADO	ENTIDAD PROMOTORA DE SALUD SUBSIDIADO	CONVIDA - ARS CONVIDA

```
ldr_df.display()
```

Table

	EDAD PADRE	NIVEL EDUCATIVO PADRE	ULTIMO AÑO APROBADO PADRE	NOMBRES Y APELLIDOS CERTIFICADOR	PROFESIÓN CERTIFICADOR	DEPARTAMENTO EXPEDICIÓN	MUNICIPIO EXPEDICIÓN	FECHA EXPEDICIÓN
1	26(4)	TECNOLÓGICA	3	ANDREA DUQUE LOPEZ	MÉDICO	CALDAS	LA DORADA	2021-01-01T05:00:00Z
2	24(4)	MEDIA ACADÉMICA O CLÁSICA	11	ALDEMAR PIEDRAHITA VILLAMIL	MÉDICO	CALDAS	LA DORADA	2021-01-01T05:00:00Z
3	25(4)	BÁSICA SECUNDARIA	9	LAURA CAMILA ORTIZ HERNANDEZ	MÉDICO	CALDAS	LA DORADA	2021-01-01T05:00:00Z
4	20(4)	BÁSICA SECUNDARIA	9	ANDREA DUQUE LOPEZ	MÉDICO	CALDAS	LA DORADA	2021-01-02T05:00:00Z
5	22(4)	BÁSICA SECUNDARIA	9	ANDREA DUQUE LOPEZ	MÉDICO	CALDAS	LA DORADA	2021-01-02T05:00:00Z
6	26(4)	MEDIA ACADÉMICA O CLÁSICA	11	JESSICA MARYAM CARDONA MUÑOZ	MÉDICO	CALDAS	LA DORADA	2021-01-03T05:00:00Z
7	34(4)	SIN INFORMACIÓN	null	LAURA CAMILA ORTIZ HERNANDEZ	MÉDICO	CALDAS	LA DORADA	2021-01-03T05:00:00Z

Conjunto de datos	Filas	Columnas
Medellín	272526	40
La Dorada	1558	50

A continuación, se presentan los resúmenes estadísticos de los conjuntos de datos:

Medellín:

```
#Se hace un resumen estadístico de la información de la ciudad de Medellín
display(mde_df.describe())
```

(2) Spark Jobs

Table

	summary	ID	AREANAC	COD_INSP	SIT_PARTO	OTRO_SIT	NOM_INST	COD_INST	SEXO
1	count	272526	272526	21	272526	206	272480	272479	272526
2	mean	136263.5	1.001115489898211 5	17.142857142857142	1.0035629627998797	null	null	5.0423110901951 19E10	1.4856857694311
3	stddev	78671.62406802087	0.058154376732514425	16.888711136080463	0.08062548715771939	null	null	1.2019072871550 64E10	0.49992811220321
4	min	1	1	1.0	1	A 2 CUADRAS DE LA CLINICA EL R	050010209202 CLINICA EL ROSARIO SEDE EL TESORO	050010115001	1
5	max	272526	9	57.0	9	vereda chever	VIRREY SOLIS I.P.S S.A SAN DIEGO	850010000101	3

La Dorada:

```
#Se hace un resumen estadístico de la información de la ciudad de La Dorada
display(ldr_df.describe())
```

(2) Spark Jobs

Table

	summary	NÚMERO CERTIFICADO	DEPARTAMENTO	MUNICIPIO	AREA NACIMIENTO	INSPECCION CORREGIMIENTO O CASERIO NACIMIENTO	SITIO NACIMIENTO	CÓDIGO INSTITUCIÓN	NOMBRE INSTITUCIÓN
1	count	1558	1558	1558	1558	1	1558	1558	1558
2	mean	1.6615013914955078 8	null	null	null	null	null	1.73800051901E11	null
3	stddev	3666492.03766905	null	null	null	null	null	0.0	null
4	min	162060770	CALDAS	LA DORADA	CABECERA MUNICIPAL	GUARINOCITO	EL DOMICILIO	173800051901	173800051901 ESE HOSPITAL SAN FE
5	max	172536374	CALDAS	LA DORADA	RURAL DISPERSO	GUARINOCITO	INSTITUCIÓN DE SALUD	173800051901	173800051901 ESE HOSPITAL SAN FE

En este caso, podríamos decir que no es necesario ver todo el resumen estadístico de los dos conjuntos, teniendo en cuenta que nos podemos hacer una idea de los resultados, teniendo en cuenta la alta cantidad de variables categóricas y la existencia de valores no cuantificables estadísticamente —Como el ID, Número de certificado, Situación de parto o Código de institución—.

Por ende, hay que revisar qué variables se pueden eliminar para evitar un conjunto de datos innecesariamente grande, lo cual será visto más adelante.

Preparación de los datos

Medellín

En el caso de Medellín, el primer paso fue identificar columnas que no fueran útiles para realizar el análisis deseado, en este caso, las columnas eliminadas fueron las siguientes:

Id, Sexo, Peso_Nac, Talla_Nac, Otro_Sit, Nom_Inst, Cod_Inst, Idhemoclas, Idfactorrh, Idpertet, Nom_Pueb, Idclasadmi, Nomclasad, Codclasad, Edad_Padre, Niv_Edup, Cod_Insp, Sit_Parto, Otro_Sit, Nom_Inst, Cod_Inst, Sexo, Peso_Nac, Talla_Nac, Fecha_Nac, Ano, Mes, Aten_Par, Otrparatx, T_Ges, Numconsul, Tipo_Parto, Mul_Parto, Idhemoclas, Idfactorrh, Idpertet, N_Emb, N_Hijosv, Fecha_Nacm, Codptore, Codmunre

Por otro lado, estas fueron las columnas que quedaron en el conjunto de datos:

Areanac, Edad_Madre, Est_Civm, Niv_Edum, Cod_Barrires, Seg_Social, Barrio_Res, Comuna_Res

Esto permite realizar el análisis deseado sin necesidad de tener un conjunto de datos sumamente grande.

La Dorada:

En este caso, se transformaron las columnas Edad Madre, Último Año Aprobado Madre y Barrio. A continuación, el detalle

Columna	Transformación
Edad Madre	Eliminación de un "(4)" al lado de todos los valores de edad y casteo a tipo entero.
Último Año Aprobado Madre	Imputación de valores para sustitución de nulos.
Barrio	Imputación de valores para sustitución de nulos.

Se agregó

Por otro lado, se eliminaron las siguientes columnas:

Número Certificado, Sexo, Peso (Gramos), Talla (Centímetros), Hora Nacimiento, Parto Atendido Por, Grupo Sanguineo, Factor Rh, Pertenencia Étnica, Nombres Madre, Apellidos Madre, Edad Padre, Nivel Educativo Padre, Ultimo Año Aprobado Padre, Nombres Y Apellidos Certificador, Profesión Certificador, Departamento Expedición, Municipio Expedición, Fecha Expedición, Estado Certificado, Inspeccion Corregimiento O Caserio Nacimiento, Centro Poblado, Dirección, Rural Disperso, Código Institución, Nombre Institución, Tipo Parto, Multiplicidad Embarazo, Número Hijos Nacidos Vivos, Fecha Anterior Hijo Nacido Vivo, Régimen Seguridad, Tipo Administradora, Número Consultas Prenatales, Tiempo De Gestación, Nombre Administradora, Número Embarazos, Area Nacimiento, Sitio Nacimiento, Fecha Nacimiento.

Quedando así un conjunto de datos con las siguientes columnas:

```
ldr_df.display()
```

(7) Spark Jobs

Table +

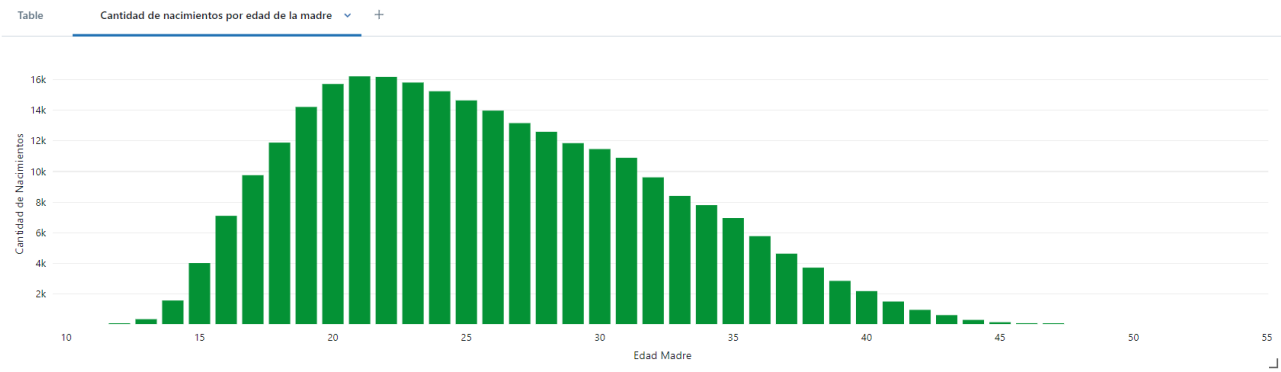
	EDAD MADRE	DEPARTAMENTO	MUNICIPIO	ESTADO CONYUGAL MADRE	NIVEL EDUCATIVO MADRE	ULTIMO AÑO APROBADO MADRE	PAÍS RESIDENCIA
1	13	CALDAS	LA DORADA	ESTÁ SOLTERA	BÁSICA PRIMARIA	5	COLOMBIA
2	13	CALDAS	LA DORADA	ESTÁ SOLTERA	BÁSICA SECUNDARIA	7	COLOMBIA
3	14	CALDAS	LA DORADA	ESTÁ SOLTERA	BÁSICA SECUNDARIA	8	COLOMBIA
4	14	CALDAS	LA DORADA	NO ESTÁ CASADA Y LLEVA MENOS DE DOS AÑOS VIVIENDO CON SU PAREJA	BÁSICA SECUNDARIA	6	COLOMBIA
5	14	CALDAS	LA DORADA	ESTÁ SOLTERA	MEDIA ACADÉMICA O CLÁSICA	10	COLOMBIA
6	14	CALDAS	LA DORADA	ESTÁ SOLTERA	BÁSICA PRIMARIA	5	COLOMBIA
7	15	CALDAS	LA DORADA	ESTÁ SOLTERA	BÁSICA SECUNDARIA	7	COLOMBIA

DEPARTAMENTO RESIDENCIA	MUNICIPIO RESIDENCIA	AREA RESIDENCIA	BARRIO	JOVEN MADRE
CUNDINAMARCA	PUERTO SALGAR	CABECERA MUNICIPAL	GAITAN	1
CALDAS	VICTORIA	CENTRO POBLADO (INSPECCIÓN, CORREGIMIENTO O CASERÍO)	null	1
CALDAS	LA DORADA	CABECERA MUNICIPAL	LAS FERIAS	1
CALDAS	VICTORIA	RURAL DISPERSO	null	1
CALDAS	LA DORADA	CABECERA MUNICIPAL	EL CABRERO	1
CALDAS	LA DORADA	CABECERA MUNICIPAL	LAS FERIAS	1

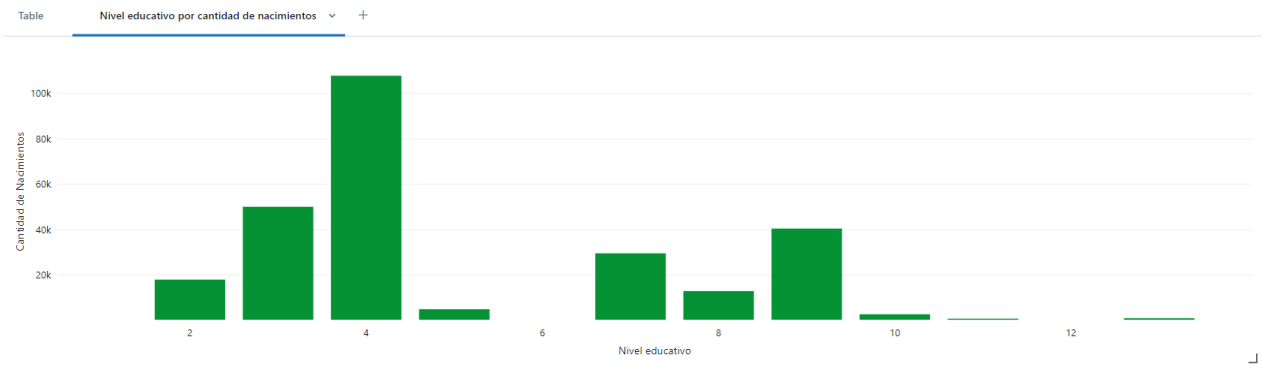
Visualización Inicial

Medellín

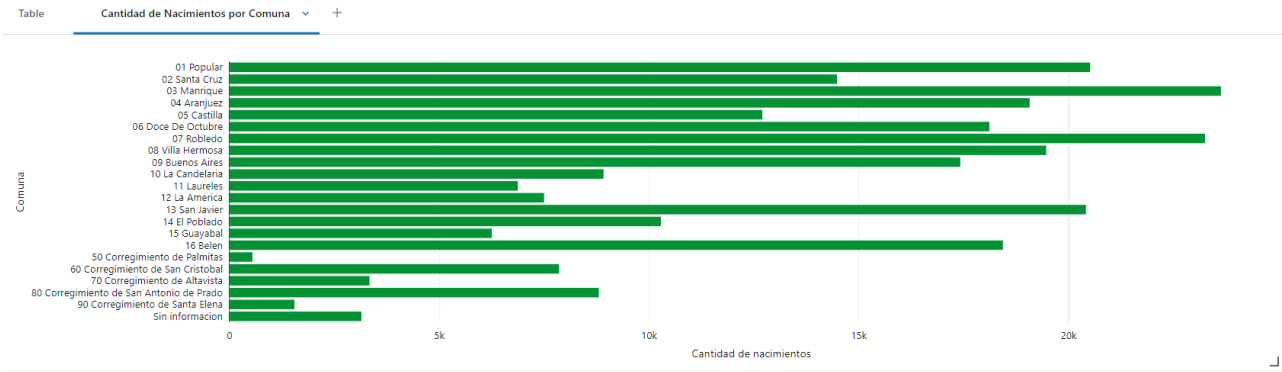
Cantidad de nacimientos por edad de la madre



Cantidad de nacimientos por nivel educativo de la madre

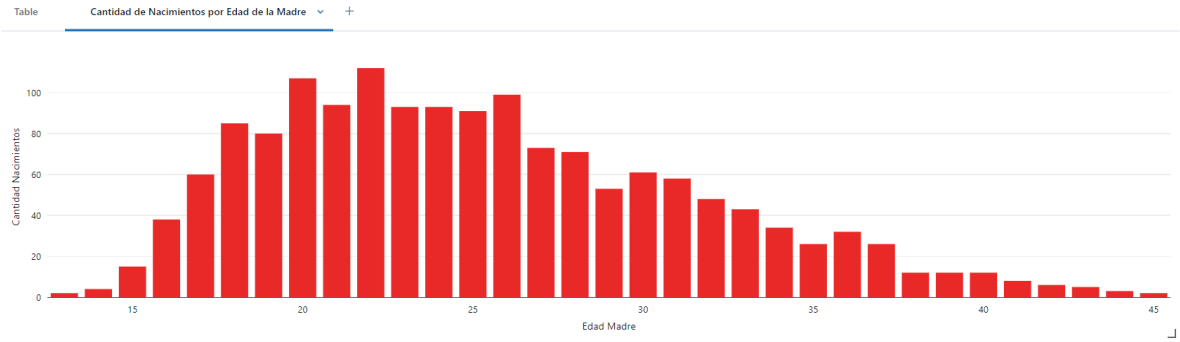


Cantidad de nacimientos por comuna

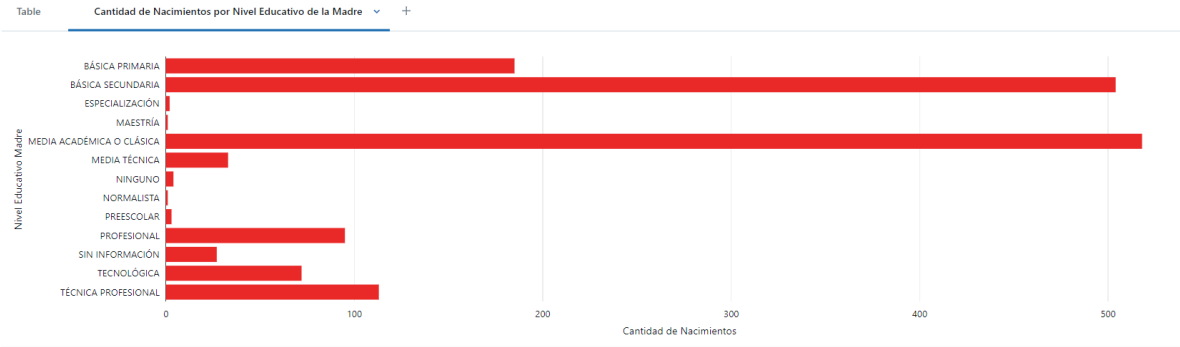


La Dorada

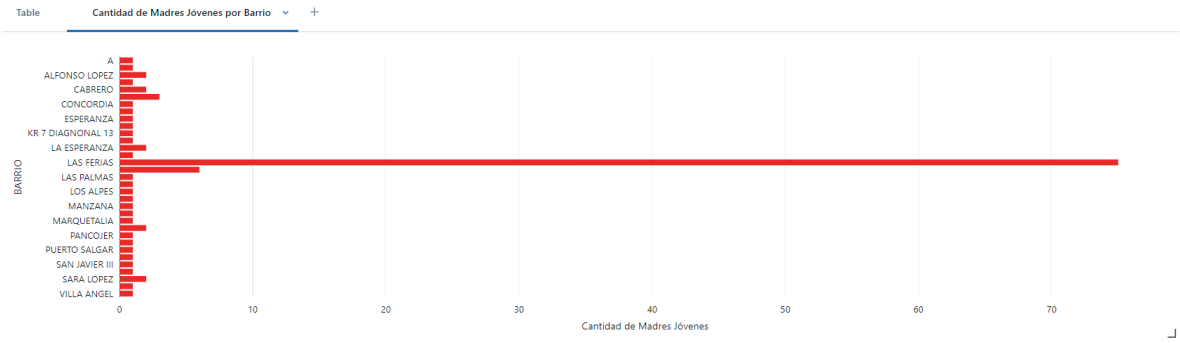
Cantidad de nacimientos por edad de la madre



Cantidad de nacimientos por nivel educativo de la madre



Cantidad de madres jóvenes por barrio



Problema

Descripción del problema

Tipo de Problema: Clasificación

El objetivo es predecir si una madre tendrá un hijo a temprana edad basado en características demográficas, educativas y residenciales.

Usando los datos de nacimientos en La Dorada y Medellín, se desarrollarán modelos de clasificación que puedan predecir efectivamente si una madre es menor de 18 años en el momento del nacimiento de su hijo. Esto es importante para políticas de salud pública, intervenciones educativas y programas de apoyo a jóvenes madres.

El enfoque es utilizar dos técnicas de clasificación para evaluar cuál maneja mejor las peculiaridades de los conjuntos de datos de La Dorada y Medellín:

Regresión Logística:

Utilidad: Proporciona un modelo de probabilidad que es fácil de interpretar, lo que es vital para comprender los factores que influyen en el embarazo a temprana edad.

Árbol de Decisión:

Utilidad: Ofrece un modelo que puede capturar relaciones no lineales entre características sin necesidad de preprocesamiento complejo.

Problemas con los datos

Problemas de calidad de datos

En el caso de Medellín se presentó el siguiente problema con los datos:

Celda	Problema	Solución
NIV_EDUM	Se presentaba el valor atípico 99 en aproximadamente 4000 registros.	Se eliminaron estos registros teniendo en cuenta la poca representatividad sobre el total de los datos.

A continuación, se presenta la solución en código:

```
###Se evidencia la existencia del valor atípico 99 en la columna correspondiente al nivel educativo de la madre
print("La cantidad de registros con NIV_EDUM igual a 99 es", mde_df.filter(mde_df["NIV_EDUM"] == 99).count())
```

(2) Spark Jobs

```
La cantidad de registros con NIV_EDUM igual a 99 es 3870
```

```
mde_df= mde_df.filter(mde_df["NIV_EDUM"] != 99)
```

mde_df: pyspark.sql.dataframe.DataFrame = [AREANAC: long, EDAD_MADRE: long ... 6 more fields]

```
print("La cantidad de registros con NIV_EDUM igual a 99 tras la transformación es", mde_df.filter(mde_df["NIV_EDUM"] == 99).count())
```

(1) Spark Jobs

```
La cantidad de registros con NIV_EDUM igual a 99 tras la transformación es 0
```

Por otro lado, en el conjunto de La Dorada se presentaron los siguientes problemas:

Celda	Problema	Solución
EDAD MADRE	Se presentaba un "(4)" al lado de cada valor de edad, imposibilitando tomar el valor como entero.	Se eliminó este "(4)" y se realizó un casteo a tipo entero.
ULTIMO AÑO APROBADO MADRE	Se presentaban 30 valores nulos.	Se realizó una imputación con la media del último año aprobado por edad.
BARRIO	Se presentaban 474 valores nulos.	Se realizó una imputación con la moda.

A continuación, se presenta la solución en código:

Transformación Edad Madre:

```
ldr_df.select("EDAD MADRE").distinct().show()
```

(2) Spark Jobs

```
+----------+
|EDAD MADRE|
+----------+
|     44(4)|
|     26(4)|
|     29(4)|
|     30(4)|
|     24(4)|
|     17(4)|
```

```
ldr_df = ldr_df.withColumn("EDAD MADRE", regexp_extract(col("EDAD MADRE"), r'(\d+)', 1).cast('int'))
```

ldr_df: pyspark.sql.dataframe.DataFrame = [NÚMERO CERTIFICADO: long, DEPARTAMENTO: string ... 48 more fields]

Imputación de valores Nulos en Ultimo año aprobado madre:

```
ldr_df.select("ULTIMO AÑO APROBADO MADRE").distinct().show()
```

(2) Spark Jobs

```
+-------------------------+
|ULTIMO AÑO APROBADO MADRE|
+-------------------------+
|                      8.0|
|                      7.0|
|                     NULL|
|                      1.0|
|                      4.0|
```

```
def imputar_ultimo_ano(df, age_col, education_col):

    avg_df = df.groupBy(age_col).agg(avg(education_col).alias("PROMEDIO ULTIMO AÑO"))

    df = df.join(avg_df, on=age_col, how="left")

    df = df.withColumn(
        education_col,
        when(col(education_col).isNull(), col("PROMEDIO ULTIMO AÑO")).otherwise(col(education_col))
    ).drop("PROMEDIO ULTIMO AÑO")

    return df

ldr_df = imputar_ultimo_ano(ldr_df, "EDAD MADRE", "ULTIMO AÑO APROBADO MADRE")

ldr_df.display()
```

```
ldr_df.select([count(when(isnan(c) | col(c).isNull() , c)).alias(c) for c in ldr_df.columns]).show()
```

(4) Spark Jobs

EDAD MADRE	DEPARTAMENTO	MUNICIPIO	ESTADO CONYUGAL MADRE	NIVEL EDUCATIVO MADRE	ULTIMO AÑO APROBADO MADRE	PAÍS RESIDENCIA	DEPARTAMENTO RESIDENCIA	MUNICIPIO RESIDENCIA	AREA RESIDENCIA	BARRIO	JOVEN MADRE
0	0	0	0	0	0	0	0	0	0	474	0

Imputación de valores Nulos en Barrio:

```
moda = ldr_df.groupBy("BARRIO").agg(count("BARRIO")).orderBy(count("BARRIO"), ascending=False).first()

print("Moda de barrio: ", moda["BARRIO"])

ldr_df = ldr_df.fillna({"BARRIO": moda["BARRIO"]})
```

(2) Spark Jobs

ldr_df: pyspark.sql.dataframe.DataFrame = [EDAD MADRE: integer, DEPARTAMENTO: string ... 10 more fields]

```
Moda de barrio:  LAS FERIAS
```

```
ldr_df.select([count(when(isnan(c) | col(c).isNull() , c)).alias(c) for c in ldr_df.columns]).show()
```

(4) Spark Jobs

```
+----------+------------+---------+---------------------+---------------------+-------------------------+---------------+-----------------------+---------------------+---------------+------+-----------+
|EDAD MADRE|DEPARTAMENTO|MUNICIPIO|ESTADO CONYUGAL MADRE|NIVEL EDUCATIVO MADRE|ULTIMO AÑO APROBADO MADRE|PAÍS RESIDENCIA|DEPARTAMENTO RESIDENCIA|MUNICIPIO RESIDENCIA|AREA RESIDENCIA|BARRIO|JOVEN MADRE|
+----------+------------+---------+---------------------+---------------------+-------------------------+---------------+-----------------------+---------------------+---------------+------+-----------+
|         0|           0|        0|                    0|                    0|                        0|              0|                      0|                    0|              0|     0|          0|
+----------+------------+---------+---------------------+---------------------+-------------------------+---------------+-----------------------+---------------------+---------------+------+-----------+
```

Creación de variables derivadas

En el caso de Medellín, se creó la variable derivada Joven Madre, que toma como valor 1 en caso de que la madre sea menor de 18 años y 0 en caso contrario.

```
mde_df = mde_df.withColumn("EDAD_MADRE", col("EDAD_MADRE").cast("int"))

# Se crea una nueva columna llamada Joven Madre, que tiene valor 1 cuando la edad de la madre es menor a 18.
mde_df = mde_df.withColumn("JOVEN_MADRE", when(col("EDAD_MADRE") < 18, 1).otherwise(0))
```

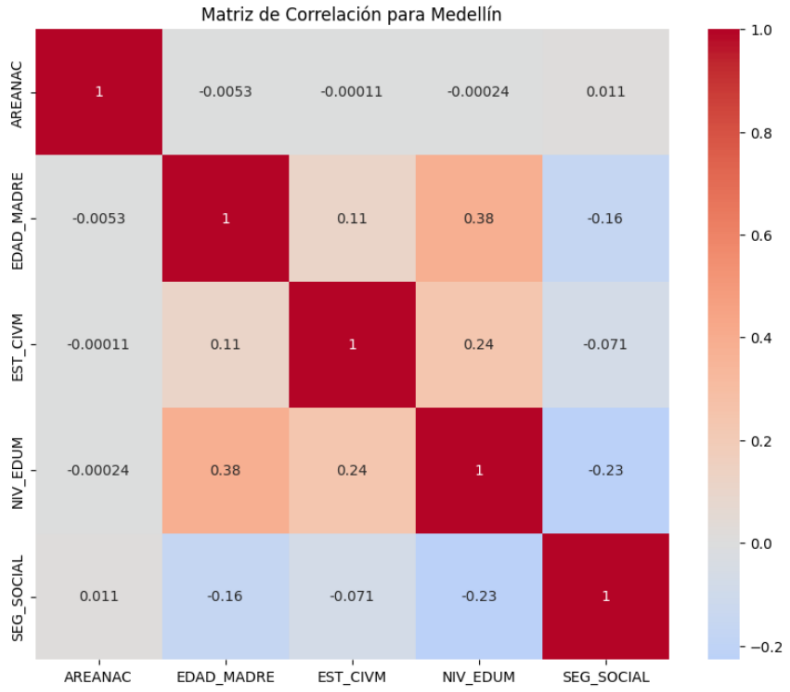
En el caso de La Dorada, se creó la misma variable.

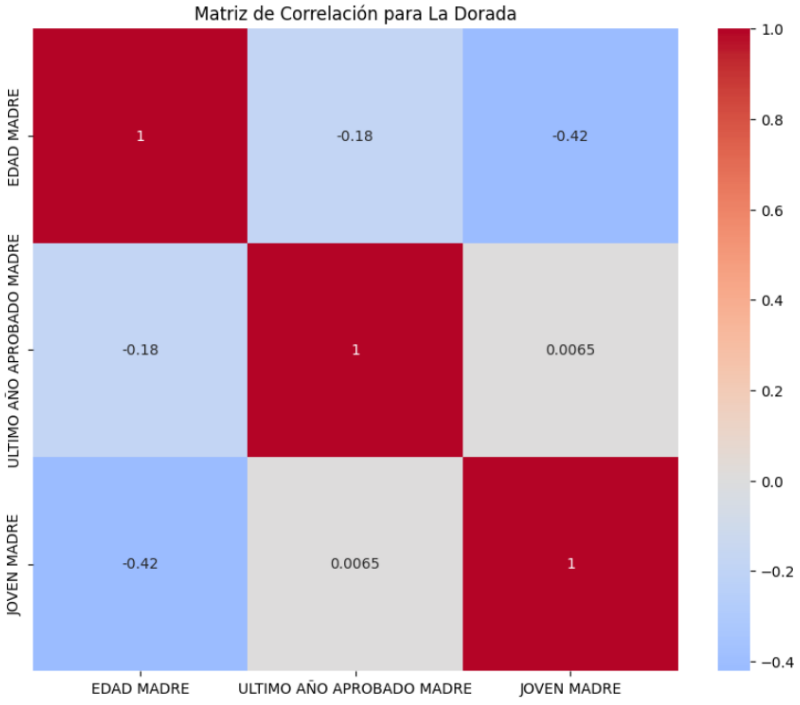
```
ldr_df = ldr_df.withColumn("EDAD MADRE", col("EDAD MADRE").cast("int"))

# Se crea una nueva columna llamada Joven Madre, que tiene valor 1 cuando la edad de la madre es menor a 18.
ldr_df = ldr_df.withColumn("JOVEN MADRE", when(col("EDAD MADRE") < 18, 1).otherwise(0))
```

```
ldr_df: pyspark.sql.dataframe.DataFrame = [NÚMERO CERTIFICADO: long, DEPARTAMENTO: string ... 49 more fields]
```

Analítica descriptiva de los datos

Matrices de correlación





Implementación de técnicas de Machine Learning

Para ambos conjuntos de datos se optó por escoger 2 modelos, un modelo de regresión logística y un árbol de clasificación, los cuales se usaron para evaluar la posibilidad de que

una mujer hubiese sido madre joven usando los demás datos sociodemográficos que existían por cada individuo dentro de datos. Primero se deben importar las librerías que nos ayudarán de manera general con el modelo.

```
from pyspark.ml.classification import DecisionTreeClassifier
```

```
from pyspark.ml.classification import LogisticRegression
```

Posteriormente se deben preparar los conjuntos de datos para que cada dato se vuelva una variable interpretable por la máquina. Es decir, se debe indexar cada uno de los datos dentro de las variables categóricas de la siguiente manera:

```
noCategoricos = ['EDAD MADRE', 'ULTIMO AÑO APROBADO MADRE', 'JOVEN MADRE', 'DEPARTAMENTO', 'PAÍS RESIDENCIA', 'AREA RESIDENCIA', 'classWeight']

# StringIndexer for categorical column
indices = [StringIndexer(inputCol= column, outputCol=column+"_IND").fit(weighted_df) for column in list(set(weighted_df.columns)-set(noCategoricos))]
```

```
noCategoricos = ['NIV_EDUM', 'JOVEN_MADRE','SEG_SOCIAL','EST_CIVM','EDAD_MADRE','AREANAC']

# StringIndexer for categorical column
indices = [StringIndexer(inputCol= column, outputCol=column+"_IND").fit(mde_df) for column in list(set(mde_df.columns)-set(noCategoricos))]


pipeML = Pipeline(stages=indices)
mde_df_ml = pipeML.fit(mde_df).transform(mde_df)

mde_df_ml.display()
```

Esto se hace de manera análoga para ambos conjuntos de datos.

Posteriormente se borran las variables categóricas dentro del conjunto de datos para generar un vector denso, en el cual se almacenará la información de cada uno de los registros dentro del dataframe.

```
feature = VectorAssembler(inputCols = ldr_df_ml.drop("JOVEN MADRE").columns, outputCol='features')

feature_vector = feature.transform(ldr_df_ml)
feature_vector.toPandas()
```

```
feature = VectorAssembler(inputCols = mde_df_ml.drop("JOVEN_MADRE").columns, outputCol='features')

feature_vector = feature.transform(mde_df_ml)
feature_vector.toPandas()
```

Excepción: Para el caso del conjunto de datos de La Dorada, se tuvo que implementar un método de pesos ya que el conjunto de datos estaba demasiado desbalanceado, y en los diferentes intentos de modelo, se presentaba un sobreajuste que no es confiable en un modelo de aprendizaje.

Entrenamiento de los modelos

Por lo tanto, en el caso de la regresión logística en el conjunto de La Dorada, para este modelo se deben asignar junto con las variables características, y las variables objetivo, la variable cual se asigna el peso.

```
df_3 = feature_vector.select(['features','JOVEN MADRE','classWeight'])

train , test = df_3.randomSplit([0.7,0.3])
```

Usando la función "randomSplit" podemos hacer la separación de los datos de manera a que queden 70% de datos de entrenamiento y 30% de datos de prueba

```
# Configurar la regresión logística con regularización
LogReg = LogisticRegression(
    featuresCol='features',
    labelCol='JOVEN MADRE',
    weightCol="classWeight",
    regParam=0.1,           # Aumentar para más fuerza de regularización
    elasticNetParam=0.8,    # Más cercano a 1 para favorecer L1, que puede inducir sparse features
    maxIter=100,            # Aumentar si el algoritmo no converge
    fitIntercept=True       # Generalmente es una buena idea ajustar el intercepto
)
```

También buscando evitar el sobreajuste, se hizo una validación de los datos en cruz para que se tengan en cuenta, de la manera más equitativa los datos a la hora ser entrenado.

```
pipeline = Pipeline(stages=[LogReg])

# Configurar la grilla de parámetros
paramGrid = ParamGridBuilder() \
    .addGrid(LogReg.regParam, [0.01, 0.1]) \
    .addGrid(LogReg.elasticNetParam, [0.0, 1.0]) \
    .addGrid(LogReg.maxIter, [10, 50]) \
    .build()


# Configurar el evaluador
evaluator = BinaryClassificationEvaluator(labelCol="JOVEN MADRE", metricName="areaUnderROC")

# Configurar la validación cruzada
crossval = CrossValidator(estimator=pipeline,
                          estimatorParamMaps=paramGrid,
                          evaluator=evaluator,
                          numFolds=3)  #
```

Posterior a esto se escoge el mejor modelo:

```
# Ejecutar la validación cruzada
cvModel = crossval.fit(train)

# Obtener el mejor modelo
bestModel = cvModel.bestModel
```

En el caso de Medellín se decidió hacer este proceso de manera más simple con el fin de ver si se veían afectados los modelos tanto por los conjuntos de datos, como por los parámetros que se estaban personalizando.

```
LogReg = LogisticRegression(featuresCol='features', labelCol='JOVEN_MADRE')
model = LogReg.fit(train)
```

Para el caso del árbol de decisión dentro del conjunto de datos de La Dorada se instancio la información usando nuevamente parámetros personalizados buscando evitar el sobreajuste.

```
dt = DecisionTreeClassifier(labelCol="JOVEN MADRE", featuresCol="features")

# Configurar la grilla de parámetros para la validación cruzada
paramGrid = ParamGridBuilder() \
    .addGrid(dt.maxDepth, [5, 20]) \
    .addGrid(dt.maxBins, [20, 32]) \
    .build()

# Configurar el evaluador
evaluator = MulticlassClassificationEvaluator(labelCol="JOVEN MADRE", predictionCol="prediction", metricName="accuracy")

# Configurar la validación cruzada
crossval = CrossValidator(estimator=pipeline,
                          estimatorParamMaps=paramGrid,
                          evaluator=evaluator,
                          numFolds=5)
```

Modificando los parámetros de profundidad y agrupaciones por árbol.

```
# Entrenar el modelo usando el conjunto de entrenamiento
cvModel = crossval.fit(train_data)

# Obtener el mejor modelo
bestModel = cvModel.bestModel
```

Para nuevamente escoger el mejor modelo de los posibles.

Métricas de rendimiento

Regresión logística (La Dorada)

```
predictions = bestModel.transform(train)
auc = evaluator.evaluate(predictions)

print("Mejor AUC: ", auc)
```

(6) Spark Jobs

predictions: pyspark.sql.dataframe.DataFrame

```
Mejor AUC:  1.0
```

Regresión logística ("Medellín")

```
# Evaluar el modelo
from pyspark.ml.evaluation import BinaryClassificationEvaluator
evaluator = BinaryClassificationEvaluator(labelCol="JOVEN_MADRE", metricName="areaUnderROC")
predictions = model.transform(test)
auc = evaluator.evaluate(predictions)
print(f"AUC: {auc}")
```

(4) Spark Jobs

predictions: pyspark.sql.dataframe.DataFrame

```
AUC: 0.9999995357561424
```

Árbol de decisión (La Dorada)

```
# Usar el mejor modelo para hacer predicciones en el conjunto de prueba
predictions = bestModel.transform(test_data)

# Evaluar el modelo
accuracy = evaluator.evaluate(predictions)
print("Accuracy: ", accuracy)

```

(3) Spark Jobs

predictions: pyspark.sql.dataframe.DataFrame

```
Accuracy:  1.0
```

Árbol de decisión (“Medellín”)

```
# Usar el mejor modelo para hacer predicciones en el conjunto de prueba
predictions = bestModel.transform(test_data)

# Evaluar el modelo
accuracy = evaluator.evaluate(predictions)
print("Accuracy: ", accuracy)
```

```
▸ (6) Spark Jobs
▸ predictions: pyspark.sql.dataframe.DataFrame
Accuracy:  1.0
```

Englobando de manera general todas las métricas de evaluación de los cuatro modelos, se puede sostener que hubo un problema serio a la hora de seleccionar los conjuntos de datos, ya que estos se encontraban muy desbalanceados en cuanto a sus datos, y por lo tanto no pudimos inferir nada de nuestro análisis predictivo.

Solución de las preguntas propuestas

- ¿Es posible hacer un buen sistema de detección de embarazos a temprana edad a través de modelos de Machine Learning?

  Depende, en este caso la calidad de los datos no permite dar un veredicto completamente contundente teniendo en cuenta que los modelos no dan la suficiente claridad y no se puede confiar en ellos del todo. No obstante, con unos datos de muy alta calidad sí sería posible realizar un sistema de detección de embarazos a temprana edad mediante modelos de Machine Learning.

- ¿Es posible encontrar a través de gráficos cuál de los dos municipios requiere más atención frente al problema de los embarazos a temprana edad?

  Sí. Los datos brindan suficiente información un análisis descriptivo de cuál de los 2 municipios tienen más presente la problemática de embarazos a tempana edad.

En este caso, la ciudad de Medellín puede requerir mayor atención que La Dorada, teniendo en cuenta que Medellín cuenta con una proporción de 8.37% de embarazos jóvenes, mientras que La Dorada tiene una proporción de 7.64% embarazos jóvenes.

Conclusiones, observaciones y recomendaciones

Conclusiones

Influencia del Desbalance de Datos: Los modelos entrenados revelaron la significativa influencia del desbalance en los datos sobre el rendimiento del modelo. En particular, el desbalance presente en el conjunto de datos de La Dorada condujo a problemas de sobreajuste, lo que afectó negativamente la capacidad del modelo para generalizar a nuevos datos.

Efectividad de las Técnicas de Regularización y Ponderación: La implementación de técnicas como la asignación de pesos y la validación cruzada ayudó a mitigar algunos de los problemas causados por el desbalance de datos. Sin embargo, la efectividad fue limitada por la calidad y naturaleza de los datos disponibles.

Comparación entre Modelos y Localidades: Los modelos aplicados en ambos municipios mostraron diferencias en su rendimiento, lo que indica que las características sociodemográficas y la distribución de las clases varían significativamente entre La Dorada y Medellín. Esto sugiere que las estrategias de intervención pueden necesitar ser adaptadas a las condiciones locales específicas.

Observaciones

Calidad de los Datos: La calidad y relevancia de los datos son críticas para el éxito de los modelos de machine learning. En este proyecto, la calidad y el desbalance de los datos limitaron la capacidad de los modelos para realizar predicciones precisas y confiables.

Recomendaciones

Mejora en la Recolección de Datos: Para futuros trabajos, es fundamental mejorar los procesos de recolección de datos para asegurar un mayor equilibrio y representatividad. Esto incluye la adquisición de más datos balanceados o el uso de técnicas avanzadas de sampling o generación sintética de datos.

Optimización de Modelos: Continuar con la experimentación y ajuste de modelos, utilizando un rango más amplio de técnicas de preprocesamiento y modelado, incluyendo ensambles de modelos y técnicas de machine learning más avanzadas conduce a una constante mejora de resultados, lo que es siempre deseable.

Referencias

Profamilia. (2023). Nota de política: Profamilia. https://profamilia.org.co/wp-content/uploads/2023/03/NOTA-POLITICA_PROFAMILIA.pdf