**Pontificia Universidad Javeriana**



**Maestría en Ingeniería de Sistemas y Computación**   
**Inteligencia de Negocios**

**Miguel Andrés Cortés Orozco**   
**mi-cortes@javeriana.edu.co**

**Bogotá, Marzo 24 de 2020**

Contenido

[Introducción 3](#_Toc35882788)

[Inciso: Cambio de Proyecto 4](#_Toc35882789)

[Problema de Negocio 4](#_Toc35882790)

[Objetivos 5](#_Toc35882791)

[Objetivo General 5](#_Toc35882792)

[Objetivos Específicos 6](#_Toc35882793)

[Inventario de Datos 6](#_Toc35882794)

[Metodología 8](#_Toc35882795)

[Procesamiento de datos 8](#_Toc35882796)

[Codificación de las columnas 8](#_Toc35882797)

[Identificación de categorías 8](#_Toc35882798)

[Generación de vista minable 9](#_Toc35882799)

[Estimadores 9](#_Toc35882800)

[Selección de Variables 10](#_Toc35882801)

[Métricas de Evaluación 10](#_Toc35882802)

[Re-muestreo 11](#_Toc35882803)

[Optimización de hiper parámetros 11](#_Toc35882804)

[Conclusiones 13](#_Toc35882805)

[Trabajos Futuros 14](#_Toc35882806)

[Apéndice I: Transformaciones 14](#_Toc35882807)

# Introducción

En el plan de desarrollo de Bogotá para los años 2016 y 2020, se encuentra que uno de los pilares de acción es la igualdad de calidad de vida. Uno de los focos de dicho pilar, es la “Atención integral y eficiente en salud” en donde se busca el desarrollo de un esquema de promoción de la salud y prevención de la enfermedad, a partir de la identificación, clasificación e intervención del riesgo en salud.

Con base a esto, se investigaron algunos indicadores que pudieran ser la base para una estrategia basada en machine learning y minería de datos, con el fin de identificar y predecir (en cierta medida), si una persona puede llegar a sufrir de enfermedades respiratorias.

Una posible fuente de estos indicadores es la encuesta multipropósito del DANE realizada en el año 2017. Estos datos son publicados en su página web y son accesibles para todo el mundo. Esta encuesta corresponde a 109.111 hogares encuestados (77.025 en Bogotá y 32.086 en municipios de Cundinamarca) y 319.952 personas con encuesta completa (221.809 en Bogotá y 98.143 en municipios). A su vez cuenta con 14 capítulos de los cuales se seleccionaron:

* Capítulo de vivienda y entorno (B)
* Capítulo de salud (F)

A partir de estos dos conjuntos de datos unificados, se construyeron varios estimadores, a los cuales se les aplicó la métrica de evaluación de clasificación ROC-AUC con validación cruzada.

# Inciso: Cambio de Proyecto

Mi anterior proyecto consistía en generar una métrica de evaluación a partir de información espacial, y métricas de la secretaría de tránsito acerca de la accidentalidad, cantidad de comparendos y velocidad media en las vías principales de la ciudad de Bogotá.

Esta métrica la definí como la eficiencia de cobertura semafórica aledaña a las ciclo-rutas de la ciudad. A partir de esta métrica se pretendía clasificar cuadrantes de la ciudad para determinar si la cantidad de semáforos en la zona era óptima. Sin embargo, pese a realizar varias iteraciones entre las fases de entendimiento del negocio y entendimiento de los datos, no conseguí plantear la tarea de negocio como un problema de minería de datos. Esto ocurrió principalmente debido a dos razones.

* Los datos disponibles no eran los adecuados
* No se puede establecer una relación de causalidad concreta entre la cantidad de semáforos, y una reducción de la accidentalidad, número de comparendos o una velocidad de tránsito media elevada.

Por estas razones, opté por replantear el enfoque del problema de minería sujeto al plan de desarrollo de la ciudad, y fue en esta búsqueda donde identifiqué el enfoque de prevención de saludo en el pilar de Atención integral y eficiente en salud. Además, al consultar la literatura existente en cuanto a aplicaciones de minería de datos relacionados con la ciudad de Bogotá, me encontré con la encuesta multipropósito del DANE, cuyos datos tienen una gran capacidad para ser explotados con algoritmos de machine learning.

# Problema de Negocio

Predecir si una persona tiene problemas respiratorios a partir de datos acerca de su

* Vivienda: como el material de paredes, el tipo de estrato, el tipo de casa, etc.
* Condiciones del sector: Cercanía a fábricas, botaderos, caños de aguas, mal olor, mal manejo de basuras, etc.
* Prevención y Consulta: Consultas medicina general, medicina especializada, participación en actividades de prevención, etc.

El conjunto de datos utilizado cuenta con un total de 9672 observaciones que reportan problemas respiratorios. Estas predicciones podrán ser utilizadas para enfocar campañas de salud y prevención en estas poblaciones con mayor riesgo, dadas las condiciones ambientales en las que habitan. Esto podrá verse traducido en una reducción de costos en materia de hospitalización y medicinas ocasionados por dichas enfermedades.

# Objetivos

## Objetivo General

Generar un modelo predictivo de clasificación que permita identificar si un ciudadano de Bogotá o municipios aledaños tiene o puede tener problemas respiratorios a partir de las condiciones de su hábitat y participación en consultas médicas y campañas de prevención.

## Objetivos Específicos

* Evaluar la calidad del conjunto de datos disponible y proponer una estrategia de limpieza.
* Identificar las diez variables o características más importantes del conjunto de datos con las cuales se puede construir un modelo predictivo.
* Evaluar el desempeño de diferentes estimadores de clasificación a partir de una métrica de calidad
* Evaluar el efecto de realizar un re-muestreo sobre los datos para balancear las instancias.
* Evaluar el desempeño de técnicas de optimización de parámetros sobre los mejores estimadores encontrados.

# Inventario de Datos

Los datos utilizados para la minería se tomaron de la encuesta multipropósito del DANE para el año 2017. En particular, se utilizaron los siguientes datasets:

|  |  |  |  |
| --- | --- | --- | --- |
| **Dataset** | **Descripción** | **Columnas** | **Filas** |
| Capítulo\_B | Datos de la vivienda y su entorno | 51 | 107218 |
| Capítulo\_F | Datos de salud | 113 | 319952 |

Tabla 1: Inventario de datos

De los cuales se seleccionaron las siguientes columnas:

Capítulo B:

* NVCBP8A: La vivienda presenta humedad
* NVCBP8B: La vivienda presenta goteras
* NVCBP8D: La vivienda presenta fallos en tuberías
* NVCBP8G: La vivienda cuenta con poca ventilación
* NVCBP8H: La vivienda presenta inundación cuando llueve o crece el río
* NVCBP10: Tipo de vivienda (casa, apartamento, cuarto, otro)
* NVCBP11AA: Estrato para tarifa de la vivienda
* NVCBP11B: Vivienda cuenta con servicio de acueducto
* NVCBP11C: Vivienda cuenta con servicio de alcantarillado
* NVCBP11D: Vivienda cuenta con servicio de recolección de basuras
* NVCBP12: Material predominante de las paredes exteriores
* NVCBP13: Material predominante de los pisos
* NVCBP14A: Vivienda se encuentra cerca de fábricas.
* NVCBP14B: Vivienda se encuentra cerca de botaderos de basuras.
* NVCBP14I: Vivienda se encuentra cerca de caños de aguas residuales.
* NVCBP15D: El sector presenta contaminación del aire
* NVCBP15E: El sector presenta malos olores
* NVCBP15F: En el sector se genera y se maneja inadecuadamente las basuras
* DIRECTORIO: Identificador de la vivienda

Capítulo C:

* NPCFP1: Está afiliado a alguna entidad de seguridad social en salud.
* NPCFP2: Régimen de seguridad social (contributivo, especial, subsidiado, no sabe)
* NPCFP7: Ha participado en acciones de promoción de salud y prevención
* NPCFP13A: Sin estar enfermo por lo menos una vez al año consulta medicina general
* NPCFP13B: Sin estar enfermo por lo menos una vez al año consulta medicina especializada.
* NPCFP13D: Sin estar enfermo por lo menos una vez al año consulta medicina alternativa.
* NPCFP13E: Sin estar enfermo por lo menos una vez al año consulta médica.
* NPCFP14B: Le han diagnosticado alguna enfermedad respiratoria, bronquitis o enfisema (target).
* DIRECTORIO: Identificador de la vivienda

# Metodología

Para el procesamiento, entrenamiento y validación, se utilizó Python debido a la gran cantidad de paquetes y librerías enfocadas a machine learning. Entre ellas: numpy, scikit-learn, pandas y matplotlib. Todo el análisis se realizó en notebooks de Jupiter, y el código se publicó en el siguiente repositorio de Github:

## Procesamiento de datos

### Codificación de las columnas

Para facilitar el entendimiento de los datos, se realizaron los siguientes renombramientos:

|  |  |
| --- | --- |
| **Columna Original** | **Columna Renombrada** |
| NPCFP1 | SEGURIDAD\_SOCIAL\_SI |
| NPCFP2 | SEGURIDAD\_SOCIAL\_REGIMEN |
| NPCFP7 | PARTICIPA\_PROMO\_PREVE |
| NPCFP13A | CONSULTA\_MEDICINA\_GENERAL |
| NPCFP13B | CONSULTA\_MEDICINA\_ESPECIALIZADA |
| NPCFP13D | CONSULTA\_MEDICINA\_ALTERNATIVA |
| NPCFP13E | CONSULTA |
| NPCFP14B | TARGET\_ENFERMEDAD\_RESPIRATORIA |
| NVCBP8A | VIV\_HUMEDAD |
| NVCBP8B | VIV\_GOTERAS |
| NVCBP8D | VIV\_FALLO\_TUBERIAS |
| NVCBP8G | VIV\_POCA\_VENTILACION |
| NVCBP8H | VIV\_INUNDACION |
| NVCBP10 | VIV\_TIPO |
| NVCBP11AA | VIV\_ESTRATO |
| NVCBP11B | VIV\_ACUEDUCTO |
| NVCBP11C | VIV\_ALCANTARILLADO |
| NVCBP11D | VIV\_RECOLECCION\_BASURA |
| NVCBP12 | VIV\_MATERIAL\_PARED |
| NVCBP13 | VIV\_MATERIAL\_PISO |
| NVCBP14A | VIV\_CERCANIA\_FABRICAS |
| NVCBP14B | VIV\_CERCANIA\_BOTADEROS |
| NVCBP14I | VIV\_CERCANIA\_CANO\_AGUAS |
| NVCBP15D | SECTOR\_CONTAMINACION\_AIRE |
| NVCBP15E | SECTOR\_MAL\_OLOR |
| NVCBP15F | SECTOR\_MAL\_MANEJO\_BASURAS |

Tabla 2: Renombramiento de columnas

Los datos de salud y de vivienda se unieron a partir de la columna DIRECTORIO. Esta columna es el código único de la vivienda. De esta manera se obtuvo un conjunto de 26 columnas y 319951 observaciones.

### Identificación de categorías

Según los metadados de la encuesta, los valores van desde 0 hasta 9. Sin embargo, al cuantificar todas las ocurrencias se obtuvo la siguiente lista de posibles valores:

[1, 2, 3, 4, 5, 6, 7, 8, 9, ' ', '0', '1', '2', '3', '4', '5', '6', '8', '9']

Donde se observa que algunas columnas están codificadas como texto y otras como números. Además, existe un valor de cadena de texto vacía ‘‘.

La transformación realizada sobre el conjunto de datos se anexa en el Apéndice I.

### Generación de vista minable

El conjunto de datos transformado cuenta con 1 columna numérica (VIV\_ESTRATO). El resto de las columnas son categóricas nominales cuyo valor es SI o NO. Para estas columnas se aplicó One-Hot Encoding. Esta transformación generó un conjunto de datos con un total de 62 dimensiones (sin incluir la variable objetivo) de columnas con un tipo de dato entero cuyo valor es cero o uno.

Debido a que varios modelos funcionan mejor con datos escalados (cuya media es cero y varianza 1), se aplicó una transformación de escalamiento sobre cada columna. Por último, la columna objetivo se codificó con 1 para SI (ha sido diagnosticado con enfermedades respiratorias) y 0 para NO, obteniendo así la vista minable.

## Estimadores

Los siguientes son los estimadores utilizados junto con sus hiperparámetros más importantes

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Regresión Logística** | |  | **Árbol de decisión** | |
| Penalidad | L2 |  | Griterio | Gini |
| Tolerancia | 0,0001 |  | Splitter | Mejor |
| Iteraciones Máximas | 100 |  | Profundidad máxima | ninguna |
| Parámetro de Regularización | 1 |  | Muestras mínimas por split | 2 |
|  |  |  | Muestras mínimas por hoja | 1 |
|  |  |  |  |  |
| **Bagging Extra Random Tree** | |  | **Bósque aleatorio** | |
| Número de árboles | 100 |  | Número de estimadores | 50 |
| Mínimo de features | 1 |  | Criterio | Gini |
| Criterio de los árboles | Gini |  | Profundidad máxima | ninguna |
| Splitter de los árboles | Random |  | Bootstrap | VERDADERO |
| Muestras mínimas split árbol | 2 |  |  |  |
| Muestras mínimas por hoja árbol | 1 |  |  |  |
|  |  |  |  |  |
| **Gradient Boosting** | |  | **Support Vector Classifier Lineal** | |
| Estimadores | 411 |  | Penalidad | l2 |
| Tasa de aprendizaje | 0,1 |  | Regularización | 1 |
| Criterio | Friedman |  | Función a optimizar | (Pérdida Hinge)^2 |
| Tolerancia | 0,0001 |  |  |  |
| Función a optimizar | Reg. Logís |  |  |  |

## Selección de Variables

Se aplicó un algoritmo de eliminación recursiva de características (RFE por sus siglas en inglés) para reducir la dimensionalidad del dataset. Dicho algoritmo funciona con un estimador y realizando un proceso iterativo de entrenamiento, asignando un ranking de importancia a cada característica y descartando aquella con la menor importancia para la siguiente iteración. El estimador utilizado fue un bosque aleatorio con 100 estimadores. El número de características a mantener se fijó a 10.

Después de su ejecución, se encontraron las siguientes características:

* VIV\_ESTRATO
* SEGURIDAD\_SOCIAL\_REGIMEN\_contributivo
* CONSULTA\_MEDICINA\_ESPECIALIZADA\_SI
* VIV\_TIPO\_casa
* VIV\_CERCANIA\_FABRICAS\_NO
* VIV\_CERCANIA\_BOTADEROS\_SI
* VIV\_CERCANIA\_CANO\_AGUAS\_NO
* SECTOR\_CONTAMINACION\_AIRE\_SI
* SECTOR\_MAL\_OLOR\_NO
* SECTOR\_MAL\_MANEJO\_BASURAS\_SI

Todas con el mismo ranking de importancia (100%).

## Métricas de Evaluación

A continuación, se muestra una matriz con los resultados de cada estimador por métrica de evaluación.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Métrica | Bagging | RF | GB | LR | Tree | SVC |
| T. entrenamiento | 26,873279 | 11,096387 | 44,425915 | 1,38973 | 0,708337 | 95,164172 |
| T. score | 4,346398 | 2,708615 | 0,221649 | 0,039066 | 0,056255 | 0,037678 |
| ROC-AUC | 0,612217 | 0,611962 | 0,645912 | 0,645184 | 0,610997 | 0,645072 |
| Accuracy | 0,966605 | 0,966591 | 0,966855 | 0,966855 | 0,966478 | 0,966855 |
| Precisión | 0,06 | 0,066667 | 0 | 0 | 0,077692 | 0 |
| Recall | 0,00031 | 0,000207 | 0 | 0 | 0,00031 | 0 |
| F1 | 0,000616 | 0,000412 | 0 | 0 | 0,000616 | 0 |

Se observa que el mejor resultado para ROC-AUC lo tiene el Gradient Boosting, aunque también es uno de los estimadores que tarda más en entrenar.

## Re-muestreo

Debido a que el dataset está altamente desbalanceado, se intentó realizar un re-muestreo para balancear las clases. En particular, se hizo un down-sampling a la clase mayoritaria y se volvieron a aplicar todas las métricas para los 3 mejores estimadores obtenidos en el punto anterior utilizando validación cruzada. Estos fueron los resultados:

|  |  |  |  |
| --- | --- | --- | --- |
|  | GB | LR | SVC |
| T. entrenamiento | 1,006115 | 0,058327 | 2,930241 |
| T. score | 0,031353 | 0,014118 | 0,014036 |
| ROC-AUC | 0,641826 | 0,642829 | 0,642775 |
| Accuracy | 0,612335 | 0,61523 | 0,614713 |
| Precisión | 0,627652 | 0,631992 | 0,633398 |
| Recall | 0,519234 | 0,551491 | 0,544254 |
| F1 | 0,587198 | 0,58887 | 0,585259 |

Se observa que en esta oportunidad los estimadores predicen valores para ambas clases. De los tres, el estimador con mejores métricas es regresor logístico con un ROC-AUC de 0.642829.

## Optimización de hiper parámetros

Se aplicó un algoritmo de Grid Search para optimización de parámetros en los siguientes estimadores:

* Árbol de decisión con selección de variables.
* Regresión logística con selección de variables y re-muestreo.

A continuación, se muestran los hiper parámetros de cada estimador:

Árbol de decisión

|  |  |
| --- | --- |
| **Parámetro** | **Posibles valores** |
| Peso de clases | Balanceada, ninguna |
| Criterio | Gini, Entropia |
| Profundidad máxima | Entero, ninguna, opcional |
| Numero de ejemplares mínimo por split | Entero, opcional |

Regresión Logística

|  |  |
| --- | --- |
| **Parámetro** | **Posibles valores** |
| Penalidad | L2, L1, elasticnet, opcional, ninguna |
| Formulación dual o primal | Dual solo está disponible cuando se utiliza L2 con el solver “liblinear” |
| Peso de clases | Balanceada, opcional, ninguna |
| Solver | Newton-cg, lbfgs, liblinear, sag, saga.  Para datasets pequeños, liblinear es una buena elección, mientras que sag y saga son más rápidos para datasets grandes |
| Iteraciones máximas | Entero |

El algoritmo de optimización retornó los siguientes estimadores, con una validación cruzada de 10:

Árbol

|  |  |
| --- | --- |
| Peso de clases | Ninguno |
| Criterio | Gini |
| Profundidad máxima | Ninguna |
| Ejemplares mínimos por split | 2 |
| ROC-AUC | 0.6123885657 |

Regresor logístico

|  |  |
| --- | --- |
| Peso de clases | Balanceado |
| Dual | False |
| Iteraciones máximas | 100 |
| Penalidad | L1 |
| Solver | liblinear |
| ROC-AUC | 6.42859809 |

Se observa que el regresor logístico obtenido es ligeramente mejor según la métrica ROC-AUC, para estos parámetros.

# Conclusiones

Se propuso una estrategia de limpieza de datos que consiste en One-hot encoding, para transformar las características categóricas en características numéricas, y un escalamiento que convierte la media de cada columna en 0 con varianza unitaria.

De igual forma se investigó el funcionamiento del método de selección de variables envolvente RFE, utilizando como meta estimador un bosque aleatorio con un total de 100 árboles. Se escogió un total de 10 características para la reducción, las cuales fueron:

* VIV\_ESTRATO
* SEGURIDAD\_SOCIAL\_REGIMEN\_contributivo
* CONSULTA\_MEDICINA\_ESPECIALIZADA\_SI
* VIV\_TIPO\_casa
* VIV\_CERCANIA\_FABRICAS\_NO
* VIV\_CERCANIA\_BOTADEROS\_SI
* VIV\_CERCANIA\_CANO\_AGUAS\_NO
* SECTOR\_CONTAMINACION\_AIRE\_SI
* SECTOR\_MAL\_OLOR\_NO
* SECTOR\_MAL\_MANEJO\_BASURAS\_SI

A partir de los datos procesados se utilizaron varios estimadores, entre ellos: Árbol de decisión, regresor logístico, clasificador tipo vector soporte lineal, Gradient Boost, bosque aleatorio, y bagging con 50 árboles extra aleatorios. De los resultados obtenidos, se concluye que el mejor estimador obtenido a partir del conjunto de datos re-muestreado, balanceando la clase mayoritaria a la misma cantidad de la clase minoritaria, y utilizando selección de variables con el algoritmo envolvente RFE, es el regresor logístico con penalidad L1, peso de clases balanceado y algoritmo de optimización liblinear. Dicho estimador obtuvo un score ROC-AUC de 6.42859809 con validación cruzada de 10 splits.

# Trabajos Futuros

Se propone utilizar métodos de aprendizaje no supervisado como clústering para hallar características intrínsecas en el conjunto de datos que permitan determinar las clases naturales que puedan ocurrir. También es posible que se logre un mejor score de clasificación a partir de otros métodos para reducción de dimensionalidad como PCA. Por último, se propone que, a partir de un conocimiento de negocio, como lo es el coste que pueden tener los Falsos positivos de clasificación, así como los Falsos negativos, se determine una mejor selección de clasificador.

# Apéndice: Transformaciones

|  |  |  |  |
| --- | --- | --- | --- |
| **Columna** | **Valor** | **Nuevo valor** | **Otro Procesamiento** |
| CONSULTA | Vacío | NO |  |
| 1 | SI |  |
| CONSULTA\_MEDICINA\_GENERAL | Vacío | NO |  |
| 1 | SI |  |
| CONSULTA\_MEDICINA\_ESPECIALIZADA | Vacío | NO |  |
| 1 | SI |  |
| CONSULTA\_MEDICINA\_ALTERNATIVA | Vacío | NO |  |
| 1 | SI |  |
| PARTICIPA\_PROMO\_PREVE | 2 | NO | Se eliminaron los registros con observaciones Vacío |
| 1 | SI |
| SECTOR\_CONTAMINACION\_AIRE | 2 | NO |  |
| 1 | SI |  |
| SECTOR\_MAL\_MANEJO\_BASURAS | 2 | NO |  |
| 1 | SI |  |
| SECTOR\_MAL\_OLOR | 2 | NO |  |
| 1 | SI |  |
| SEGURIDAD\_SOCIAL\_SI |  |  | Se eliminó esta columna puesto que para todos es SI |
| VIV\_ACUEDUCTO | 2 | NO |  |
| 1 | SI |  |
| VIV\_ALCANTARILLADO | 2 | NO |  |
| 1 | SI |  |
| VIV\_RECOLECCION\_BASURA | 2 | NO |  |
| 1 | SI |  |
| VIV\_CERCANIA\_FABRICAS | 2 | NO |  |
| 1 | SI |  |
| VIV\_CERCANIA\_BOTADEROS | 2 | NO |  |
| 1 | SI |  |
| VIV\_CERCANIA\_CANO\_AGUAS | 2 | NO |  |
| 1 | SI |  |
| VIV\_FALLO\_TUBERIAS | 2 | NO | Se eliminaron los registros con observaciones Vacío |
| 1 | SI |
| VIV\_GOTERAS | 2 | NO | Se eliminaron los registros con observaciones Vacío |
| 1 | SI |
| VIV\_HUMEDAD | 2 | NO | Se eliminaron los registros con observaciones Vacío |
| 1 | SI |
| VIV\_INUNDACION | 2 | NO | Se eliminaron los registros con observaciones Vacío |
| 1 | SI |
| VIV\_POCA\_VENTILACION | 2 | NO | Se eliminaron los registros con observaciones Vacío |
| 1 | SI |
| SEGURIDAD\_SOCIAL\_REGIMEN | 1 | contributivo |  |
| 2 | especial |  |
| 3 | subsidiado |  |
| TARGET\_ENFERMEDAD\_RESPIRATORIA | 1 | SI |  |
| 2 | NO |  |
| VIV\_ESTRATO | 0 | 0 | Se eliminaron las observaciones con 9 |
| 1 | 1 |
| 2 | 2 |
| 3 | 3 |
| 4 | 4 |
| 5 | 5 |
| 6 | 6 |
| 8 | 7 |
| VIV\_MATERIAL\_PARED | 1 | rustico |  |
| 2 | tapia |  |
| 3 | bahareque\_revocado |  |
| 4 | bahareque\_sin\_revocar |  |
| 5 | madera |  |
| 6 | prefabricado |  |
| 7 | natural |  |
| 8 | industrial |  |
| 9 | sin\_paredes |  |
| VIV\_MATERIAL\_PISO | 1 | tapete |  |
| 2 | madera\_pulida\_laca |  |
| 3 | marmol |  |
| 4 | baldosa |  |
| 5 | madera\_burda |  |
| 6 | cemento |  |
| 7 | tierra |  |
| VIV\_TIPO | 1 | casa |  |
| 2 | apartamento |  |
| 3 | cuarto |  |
| 4 | otro |  |