

# Dirección De Desarrollo Digital

Unidad De Científicos De  
Datos



**DEPARTAMENTO  
NACIONAL DE PLANEACIÓN**



**PREDICCIÓN DE CONDICIONES DE VIDA USANDO APRENDIZAJE DE MÁQUINA CON LAS FOTOS  
DE LAS FACHADAS OBTENIDAS EN EL SISBÉN IV**

**INFORME FINAL**

<b>Dependencias y entidades involucradas</b>	<ul style="list-style-type: none"><li>• Departamento Nacional de Planeación</li><li>• Dirección de Economía Naranja y Desarrollo Digital - Unidad de Científicos de Datos</li><li>• Dirección de Desarrollo Social</li></ul>
<b>Sector</b>	Ciencia, tecnología e innovación
<b>Tecnologías utilizadas</b>	Deep Learning
<b>Fuentes de datos</b>	Sisbén IV

**Contenido**

1. Presentación .....	2
2. Objetivos del proyecto .....	2
3. Metodología .....	2
4. Resultados .....	5
5. Conclusiones y recomendaciones .....	15
6. Socialización .....	15
Contacto .....	15
ANEXOS .....	16



### **3. Presentación**

La información obtenida de los hogares colombianos mediante una encuesta e información de su hogar ha permitido, la clasificación de estos hogares en 4 grandes grupos del Sisbén IV los cuales son: A (Pobreza extrema), B (Pobreza media), C (No pobres Vulnerables), y D (No Pobres). Sisbén IV contribuye con información valiosa acerca de la situación socioeconómica de muchos colombianos y de esta manera ayuda con la creación de políticas públicas que mitigan la pobreza. El presente proyecto tiene como propósito complementar los algoritmos actuales al momento de clasificar los grupos del Sisbén IV, utilizando las fotos de las fachadas de los hogares, y con los avances en el Deep Learning (Aprendizaje Profundo), una rama de la inteligencia artificial se ha demostrado que se puede entrenar una red neuronal con diferentes tipos de imágenes y clasificar una inmensa cantidad de objetos. Estas técnicas de clasificación permiten trabajar con las imágenes de las fachadas y también con los datos complementarios de las fachadas del hogar obtenidos en la encuesta. Y lograr resultados comprometedores, como los presentado en este informe.

The information obtained from Colombian households through a survey and household information has allowed the classification of these households into 4 large Sisbén IV groups, which are: A (Extreme Poverty), B (Medium Poverty), C (Vulnerable Non-Poor), and D (Non-Poor). Sisbén IV contributes valuable information about the socioeconomic situation of many Colombians and thus helps with the creation of public policies that mitigate poverty. The purpose of this project is to complement the current algorithms when classifying the Sisbén IV groups, using the photos of the facades of the homes, and with the advances in Deep Learning, a branch of artificial intelligence, it has been demonstrated that a neural network can be trained with different types of images and classify an immense amount of objects. These classification techniques make it possible to work with the images of the facades and also with the complementary data of the home facades obtained in the survey. And achieve compromising results, such as those presented in this report.

### **4. Objetivos del proyecto**

#### **4.8. General**

*Desarrollar un modelo de clasificación de imágenes utilizando metodologías de Deep Learning, con el objetivo de predecir las condiciones de vida de los hogares colombianos asociados a los cuatro niveles del Sisbén IV, a partir de fotos tomadas a las fachadas de las casas.*

#### **4.9. Específicos**

1. Analizar y preprocesar las imágenes de las fachadas para mejorar su representación, con el objetivo de tener imágenes de calidad para la etapa de entrenamiento de los modelos y depurar imágenes que no cumplan algunas condiciones de calidad o contenido que puedan entorpecer el proceso de entrenamiento de los modelos de Deep Learning.
2. Entrenar modelos de clasificación de imágenes basados en metodologías de Redes Neuronales convolucionales, con el objetivo de clasificar las fachadas de las casas en los 4 niveles del Sisbén IV.
3. Construir un pipeline de clasificación de imágenes con el mejor modelo encontrado, para evaluar nuevas imágenes.

### **5. Metodología**

La metodología se divide en 3 partes principales, que se presentan a continuación.

#### **5.8. Análisis y depuración de las imágenes**

Inicialmente se planteó el análisis de la calidad de las imágenes compartidas con el objetivo de depurar aquellas que pudieran entorpecer la etapa de entrenamiento de los modelos de clasificación de imágenes. Los problemas incluyen mala resolución, fotos con obstáculos en las fachadas, imágenes demasiado oscuras, entre otros. Cabe resaltar que se realizó un modelo de inteligencia artificial, para complementar la depuración de las imágenes de forma automática, debido a que los problemas mencionados anteriormente pueden producir errores en el entrenamiento del modelo de



clasificación, por lo tanto, se entrenó el modelo DenseNet201 con las imágenes iniciales depuradas de forma manual, evitando que no existan errores en las imágenes como los mencionados anteriormente que puedan entorpecer el entrenamiento del modelo. Adicional a esto, las imágenes depuradas, que presentan las problemáticas mencionadas se almacenaron a parte con el objetivo que el modelo pueda distinguir estos errores. Con base a lo anterior el modelo elegido fue el que mejor se adaptó al entrenamiento de estas imágenes y el que nos permitió eliminar gran parte de las imágenes finales con las problemáticas abarcadas en esta sección. Con base a esto en la Figura 1 se ilustra las gravedades de las imágenes mal tomadas en la encuesta del Sisbén IV y en la Figura 2 se ilustran, las fachadas de los hogares tomadas de forma correcta.

*Figura 1: Imágenes con errores a eliminar.*



Fuente: elaboración propia.

*Figura 2: Imágenes tomadas de forma correcta.*



Fuente: elaboración propia.

### **5.9. Entrenamiento y pruebas de modelos**



En esta etapa se entrenaron distintos modelos de clasificación de imágenes basados en redes neuronales convolucionales pre entrenadas, entre las cuales podemos mencionar: Xception, vgg16, vgg19, ResNet50, ResNet152V2, InceptionV3, InceptionresNetV2, DenseNet201, EfficientNetB0, EfficientNetB1, entre otras. Estas redes en general utilizan transfer learning (El transfer learning permite tener la capacidad de utilizar una red ya pre entrenada y tuneada, alcanzando resultados superiores y rápidos, solo cargando los pesos de las redes diseñadas por diferentes investigadores). En contraste, diseñar una red desde cero conllevaría meses de trabajo adicional (si no más). Cabe resaltar que en el 1 entregable los mejores resultados alcanzados fueron con los siguientes modelos: DenseNet201, para clasificación de los 4 niveles socioeconómicos, y DenseNet201, EfficientNetB1, InceptionV3 y Xception para la clasificación de un grupo del sisbén IV contra los demás.

Con base a lo anterior se realizaron los siguientes experimentos:

1. Entrenamiento de los 4 grupos del Sisbén IV.
2. Entrenamiento de un grupo del Sisbén IV contra los demás.
3. Entrenamiento de la unión de los grupos del Sisbén 1 y 2, luego el 3 y 4, para una clasificación binaria debido a que estas agrupaciones presentan similitud, por lo que unir los grupos más pobres y no pobres, corroborará a que podamos clasificar más eficientemente estas imágenes.
4. Entrenamiento del grupo 1 del Sisbén contra el 2.
5. Entrenamiento del grupo 3 del Sisbén contra el 4.
6. Por último, se utilizó 3 datos obtenidos de las encuestas del Sisbén IV, los cuales eran material de los pisos, paredes y tipo de vivienda. Se realizó una unión entre estos datos de la encuesta con los datos extraídos de las convoluciones de los modelos (las convoluciones son matrices de NxN, que le van a permitir al modelo extraer los datos característicos de las imágenes ya sean bordes, formas, curvas entre otros datos específicos de la imagen).

Para llevar a cabo estos experimentos se utilizó la API de Keras (Keras implementa la librería de Tensorflow facilitando el diseño de los modelos de inteligencia artificial, sin la necesidad de crear códigos desde cero), que contiene librerías propias de Tensorflow (Tensorflow es una librería de Python que permite diseñar modelos de inteligencia artificial).

Luego de las pruebas iniciales se optimizaron los hiperparámetros de cada red neuronal, como son el número de capas densas, las cuales permitieron al modelo aprender los datos característicos de una manera óptima y hacer el modelo un poco más robusto, ya que entre más capas densas se le da al modelo, se torna más profundo y su tiempo de entrenamiento aumentaría. El *learning rate* es otro hiperparámetro, este le va a permitir al modelo aprender más rápido o más lento, lo que implicaría que se puede sobre entrenar o no aprender lo suficiente, por lo que una tasa grande puede afectar a que el modelo no converja fácilmente y no encuentre un mínimo local, mientras que una tasa pequeña puede demorar el entrenamiento de los modelos, por lo que elegir la óptima es una tarea necesaria. Por último, el optimizador le brinda al modelo la capacidad de mejorar el entrenamiento a medida que pasa las épocas. Un ejemplo claro es con el optimizador Adam, el cual es un método de descenso de gradiente estocástico que se basa en la estimación adaptativa, lo que brindará un entrenamiento un poco más robusto.

Para el experimento 2 se guardaron los pesos de las épocas de los modelos, con el objetivo de elegir el mejor modelo y al final realizar un ensamble, ósea una unión entre los 4 modelos de dicho experimento ya que se debió tomar un grupo de Sisbén y realizar la clasificación entre este, y los 3 restantes y así sucesivamente y al finalizar el ensamble unir sus salidas, con el objetivo de elegir el grupo con mayor probabilidad. Aclarando que la unión de los modelos permitió brindar mayor resultado, además los ajustes de los parámetros del modelo ayudaron a que el modelo no se sobre entrene, y clasificar de forma eficiente los grupos, algo que no sucedería entrenando los 4 grupos juntos, por lo que el ensamble busca captar las características principales de un grupo de Sisbén contra los demás, para luego unir estos datos aprendidos y poder predecir a que grupo pertenece la imagen ingresada.

El último experimento consistió en clasificar a que grupos puede pertenecer el encuesto, a 1 y 2 o 3 y 4. Y luego su siguiente paso dependerá del grupo al que pertenece, si al 1 y 2 o 3 y 4, lo que conllevaría a otra clasificación binaria de 1 contra 2 o 3 contra 4. Aumentando la probabilidad de obtener el grupo adecuado.



### 5.10. Métricas de resultados

Si bien se utilizan distintos modelos de clasificación, para el análisis de resultados se utilizaron los mismos criterios. Se utilizaron matrices de confusión y las métricas *F1-Score*, *recall*, *precisión*, *accuracy* y *AUC* para la interpretación de los resultados y comparación de rendimiento de los diferentes modelos:

- *Accuracy*: mide el porcentaje global de muestras que el modelo ha clasificado correctamente, lo que nos permite medir la calidad del modelo y el número de aciertos.
- *Recall*: calcula qué parte de los verdaderos positivos ha clasificado correctamente el modelo como positivo. Esta métrica se utiliza para seleccionar el mejor modelo cuando hay un alto coste asociado a los falsos negativos.
- *Precision*: es la fracción de todas las instancias relevantes dividida por las instancias obtenidas. Se utiliza para medir la calidad del modelo, identificando las predicciones positivas que fueron realmente correctas.
- *F1-Score*: se utiliza para combinar las medidas de *precision* y *recall* en un único valor. Esto es práctico porque facilita la comparación del rendimiento combinado de la precisión y la exhaustividad (*recall*) entre varias soluciones, independientemente de si el conjunto de pruebas está equilibrado o no. También cabe destacar que es la métrica más utilizada para medir la capacidad de clasificación de un modelo en bases de datos no equilibradas.
- *AUC*: La curva ROC nos dice qué tan bien puede distinguir el modelo entre los grupos del Sisbén IV. Los mejores modelos pueden distinguir con precisión entre los grupos, mientras que un modelo pobre tendrá dificultades para distinguir entre los grupos del Sisbén.

Cabe aclarar que *F1-Score* fue la métrica que mayor peso va a tener a la hora de comparar los resultados, debido al desbalance que existe entre los 4 grupos del Sisbén, lo que nos va a proporcionar una mejor confiabilidad del modelo.

## 6. Resultados

A través del desarrollo metodológico descrito en la sección 5, se obtuvieron los resultados que se presentan a continuación. Toda retroalimentación desde un punto de vista experto o de usuario por parte de la DDS y la DG es bienvenida. Este insumo será de gran ayuda para mejorar la calidad y utilidad de los resultados obtenidos, de manera que agreguen mayor valor.

### 4.1 Análisis y depuración de las imágenes

Se obtuvieron 855.815 imágenes para el entrenamiento de los modelos, de estas muchas contenían errores tales como imágenes borrosas, poca iluminación, obstáculos frente de la fachada, como árboles, plantas e incluso personas, entre otros errores, por lo que se diseñó un modelo por medio de la red neuronal convolucional **DenseNet201**, donde sus datos de entrenamientos fueron las imágenes iniciales depuradas y tomadas de forma correcta, obteniendo 6142 imágenes para el entrenamiento de este modelo, logrando resultados de *accuracy* de 92% y *f1-score* de 91% tal como se ilustra en la Tabla 1. Este modelo fue de gran utilidad para la clasificación automática de imágenes, evitando la depuración manual de estas y obteniendo una cantidad total de imágenes depuradas de 350.988, que se entrenaron en modelos con divisiones de entrenamiento y prueba.

Tabla 1: Resultados obtenidos en entrenamiento de la DenseNet201 para depuración de los datos

Modelo	Accuracy	F1-Score	Recall	Precision
DenseNet201 Con los pesos de imagenet y optimizador Adam	0.92	0.91	0.91	0.91



Fuente: elaboración propia.

#### 4.2 Resultado de la clasificación de los 4 niveles socioeconómicos.

Con base a la metodología propuesta en la sección de Entrenamiento y pruebas de modelos, la clasificación se hizo con las imágenes sin realizar ningún tipo de filtro, ni preprocesamiento, clasificando los 4 grupos del Sisbén IV, cuyos resultados se encuentran en la *Tabla 2*

*Tabla 2: Resultados con todas las imágenes y sin preprocesamiento*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.37	0.36	0.37	0.36
<b>DenseNet201 sin los pesos de Imagenet y optimizador Adam</b>	0.36	0.36	0.36	0.36
<b>DenseNet201 con los pesos de imagenet y optimizador Adadelta</b>	0.38	0.37	0.38	0.37

Fuente: elaboración propia.

Respecto a la *Tabla 2* se evidencian en la primera columna los modelos utilizados y en las siguientes las métricas utilizadas para evaluar los modelos de *deep learning*. El mejor resultado se obtiene con la DenseNet201 y el optimizador Adadelta, siendo superior a los otros por muy poco y con la métrica F1-score, siendo de 37%, el cual es bajo debido al ser tan parecidos los primeros y los siguientes grupos de Sisbén IV, provocando bajo porcentaje precisión en su entrenamiento.

#### 4.3 Resultados con unión de niveles socioeconómicos

Como se explicó en la sección de metodología, se realizó el agrupamiento del grupo 1 del Sisbén IV contra la unión de los demás grupos, donde el mejor resultado fue de 60% con la DenseNet201, y se evidencia en la *Tabla 3*.

*Tabla 3: Resultados el entrenamiento del grupo 1 de Sisbén contra los demás*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.82	0.60	0.60	0.60
<b>VGG16 Con los pesos de imagenet y optimizador Adam</b>	0.87	0.46	0.50	0.43

Fuente: elaboración propia

El siguiente experimento fue el agrupamiento del grupo 2 contra la unión de los demás grupos del Sisbén IV, el mejor resultado fue de 54% con la DenseNet201 aproximadamente, y se evidencia en la *Tabla 4*

*Tabla 4: Resultados el entrenamiento del grupo 2 de Sisbén contra los demás*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.62	0.54	0.53	0.54
<b>Efficient B1 Con los pesos de imagenet y optimizador Adam</b>	0.71	0.41	0.5	0.57

Fuente: elaboración propia

El siguiente experimento fue el agrupamiento del grupo 3 contra la unión de los demás grupos del Sisbén IV, el mejor resultado fue de 58% aproximadamente con la DenseNet201, y se evidencia en la *Tabla 5*.



*Tabla 5: Resultados el entrenamiento del grupo 3 de Sisbén contra los demás*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.84	0.58	0.56	0.59
<b>InceptionV3 Con los pesos de imagenet y optimizador Adam</b>	0.38	0.27	0.5	0.19

Fuente: elaboración propia

El siguiente experimento fue el agrupamiento del grupo 4 contra la unión de los demás grupos del Sisbén IV, el mejor resultado fue de 58% aproximadamente con la DenseNet201, y se evidencia en la Tabla 6.

*Tabla 6: Resultados el entrenamiento del grupo 4 de Sisbén contra los demás*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.84	0.57	0.56	0.57
<b>Xception Con los pesos de imagenet y optimizador Adam</b>	0.61	0.56	0.64	0.59

Fuente: elaboración propia

#### 4.4 Resultados Ensamble inicial

En la tabla Tabla 7 se evidencia los resultados de realizar un Ensamble de los 4 modelos anteriores, que corresponderían a un grupo de Sisbén contra los demás, esta unión nos permitirá tomar los porcentajes más altos de cada modelo y agruparlos para elegir el porcentaje más alto de precisión del grupo correspondiente. Alcanzando un resultado de 38%.

*Tabla 7: Resultado Ensamble de los experimentos anteriores.*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.38	0.38	0.37	0.37

Fuente: elaboración propia

#### 4.5 Resultados con unión de niveles socioeconómicos

En la Tabla 8 se evidencian los resultados de la unión de los grupos del Sisbén (1, 2) y (3,4), donde su resultado fue de 60%. Siendo el primer paso, para la clasificación de estos grupos correspondientes.

*Tabla 8: Resultados el entrenamiento la unión de los grupos del Sisbén (1, 2) y (3,4)*

Modelo	Accuracy	F1-Score	Recall	Precision	AUC
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.60	0.60	0.60	0.61	0.59

Fuente: elaboración propia

En la Tabla 9 se evidencia los resultados obtenidos de la clasificación únicamente del grupo 1 del Sisbén contra el 2. Alcanzando resultados de 62%.

*Tabla 9: Resultados el entrenamiento del grupo 1 contra el 2*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.65	0.62	0.61	0.60



Fuente: elaboración propia

En la Tabla 10 se evidencia los resultados obtenidos de la clasificación únicamente del grupo 3 del Sisbén contra el 4. Alcanzando resultados de 52%.

*Tabla 10: Resultados el entrenamiento del grupo 3 contra el 4*

Modelo	Accuracy	F1-Score	Recall	Precision	AUC
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.55	0.52	0.53	0.52	0.51

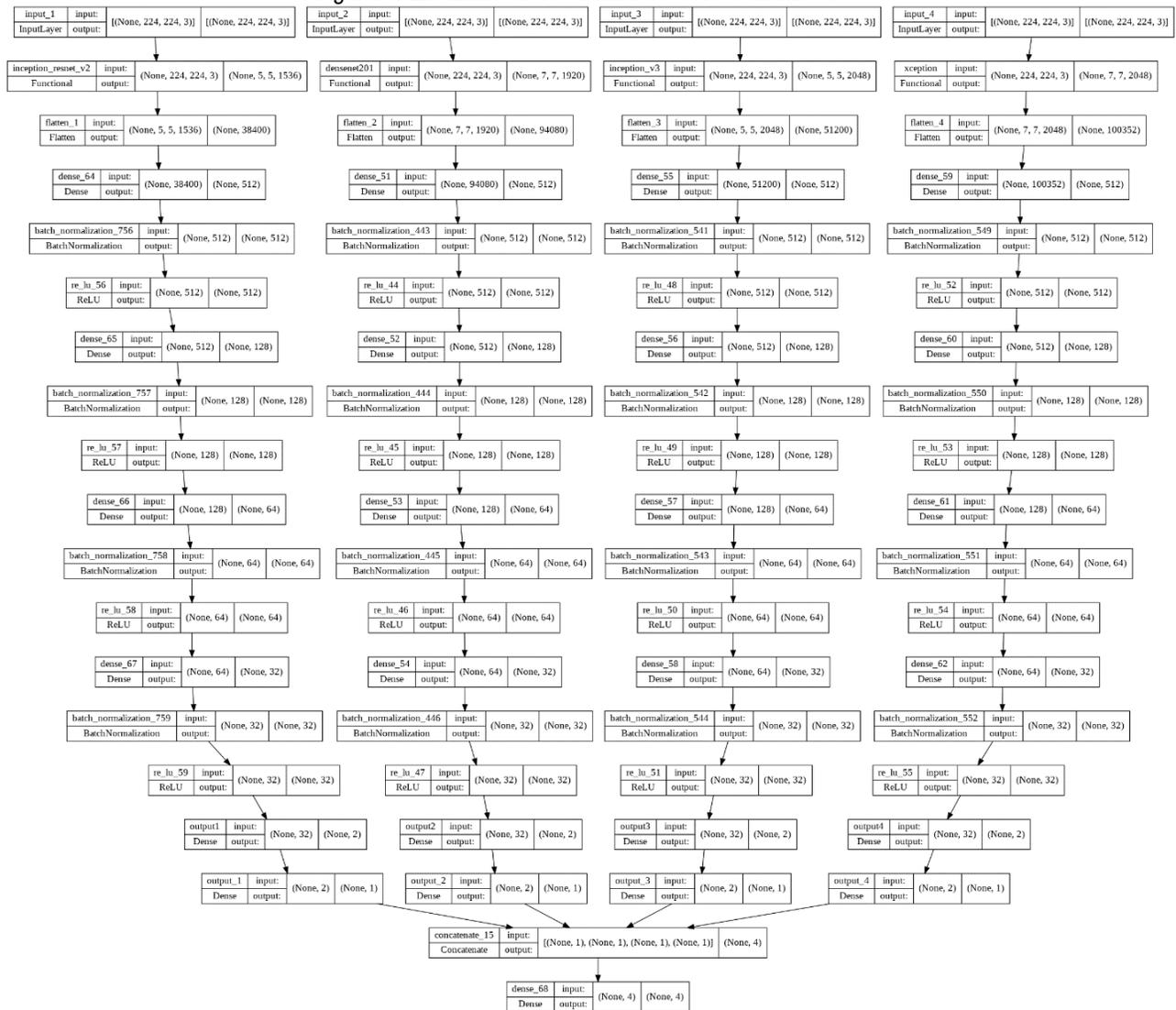
Fuente: elaboración propia

#### **4.6 Resultados Ensamble Final**

Con base a la metodología propuesta en la sección de Entrenamiento y pruebas de modelos en la Figura 3 se observa la estructura del ensamble al realizar la unión de los 4 grupos del Sisbén IV, donde se observan las 4 entradas a los 4 modelos y al final la unión de las salidas, con el propósito de unir sus datos aprendidos y mejorar los porcentajes de clasificación del modelo.



**Figura 3: Ensamble de los 4 niveles socioeconómicos**



Fuente: elaboración propia

Los resultados obtenidos nos permiten obtener la clasificación de los 4 grupos del Sisbén de una manera óptima y poder abarcar todas las metodologías posibles al momento de clasificar los 4 niveles socioeconómicos. En la Tabla 11 se evidencia los resultados obtenidos, alcanzando un 33% de precisión.

*Tabla 11: Ensamble de los experimentos anteriores*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.34	0.33	0.35	0.33

Fuente: elaboración propia

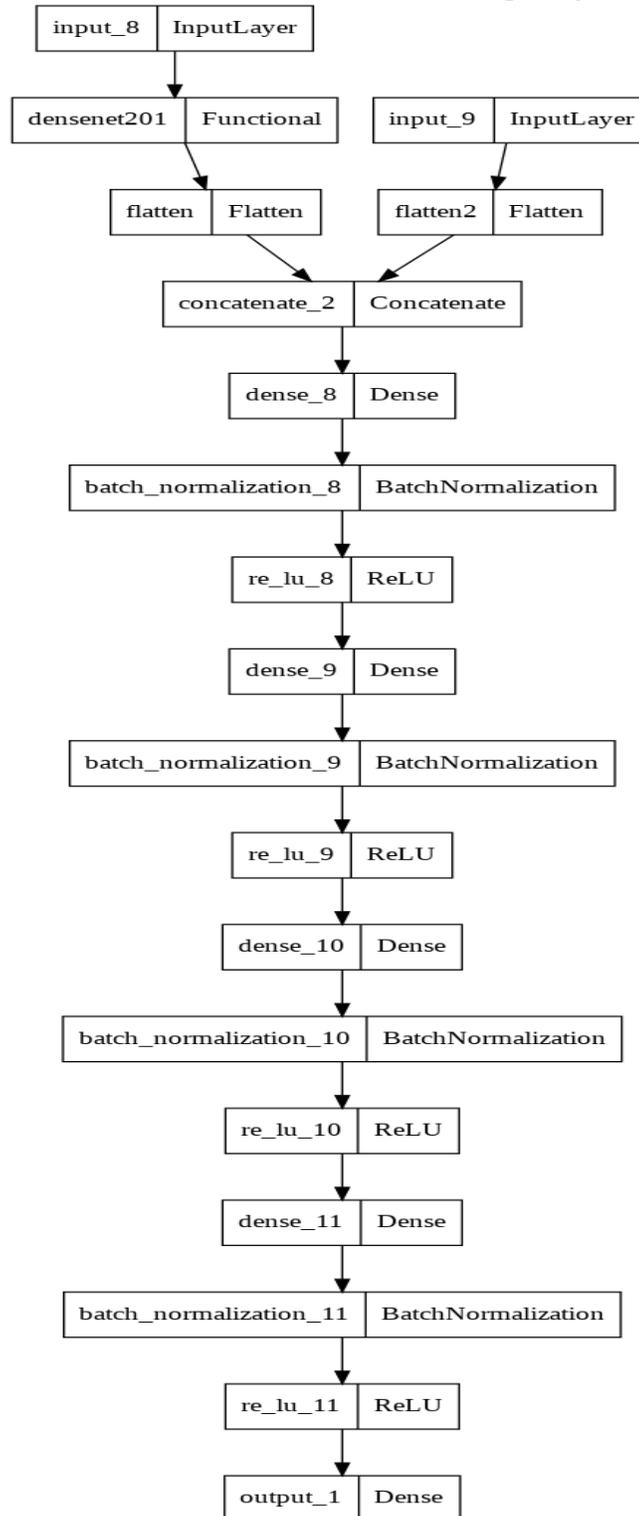
**4.7 Resultados con metadatos.**

En la Figura 4 se observa la estructura de la DenseNet201, en donde las imágenes ingresadas al modelo se hacen por el “input\_8”, y las entradas de los metadatos se realizan por el “input\_9”, luego de concatenan sus datos por



medio de la función de aplanamiento llamada “flatten” la cual permitió aplanar los datos, ósea convertir todo a un vector, para luego conllevar al aprendizaje por medio de las capas densas y clasificar los niveles socioeconómicos.

Figura 4: Estructura del modelo al ser entrenado con imágenes y metadatos.



Fuente: elaboración propia



Una vez preprocesado los metadatos correspondientes a las imágenes finales, se conllevó un análisis más estratégico mejorando los porcentajes, utilizando estos datos obtenidos de las encuestas del Sisbén IV en el entrenamiento de los modelos. Logrando los siguientes resultados.

#### 4.7.1 Resultados con unión de niveles socioeconómicos

En Tabla 12 se evidencia los resultados de la unión de los grupos del Sisbén (1, 2) y (3,4), adicionando los metadatos en su entrenamiento brindando una mejora en sus resultados siendo de 68%.

*Tabla 12: Resultados el entrenamiento la unión de los grupos del Sisbén (1, 2) y (3,4) y metadatos*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.70	0.68	0.69	0.69

Fuente: elaboración propia

En la Tabla 13 también se observan mejoras significativas al utilizar los metadatos en la clasificación únicamente del grupo 1 del Sisbén contra el 2, alcanzando resultados de 65%.

*Tabla 13: Resultados el entrenamiento del grupo 1 contra el 2 con metadatos*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.704	0.65	0.64	0.65

Fuente: elaboración propia

En la Tabla 14 También se observan mejoras significativas al momento de clasificar únicamente del grupo 3 del Sisbén contra el 4, alcanzando resultados de 61%.

*Tabla 14: Resultados el entrenamiento del grupo 3 contra el 4 con metadatos*

Modelo	Accuracy	F1-Score	Recall	Precision
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.65	0.61	0.61	0.60

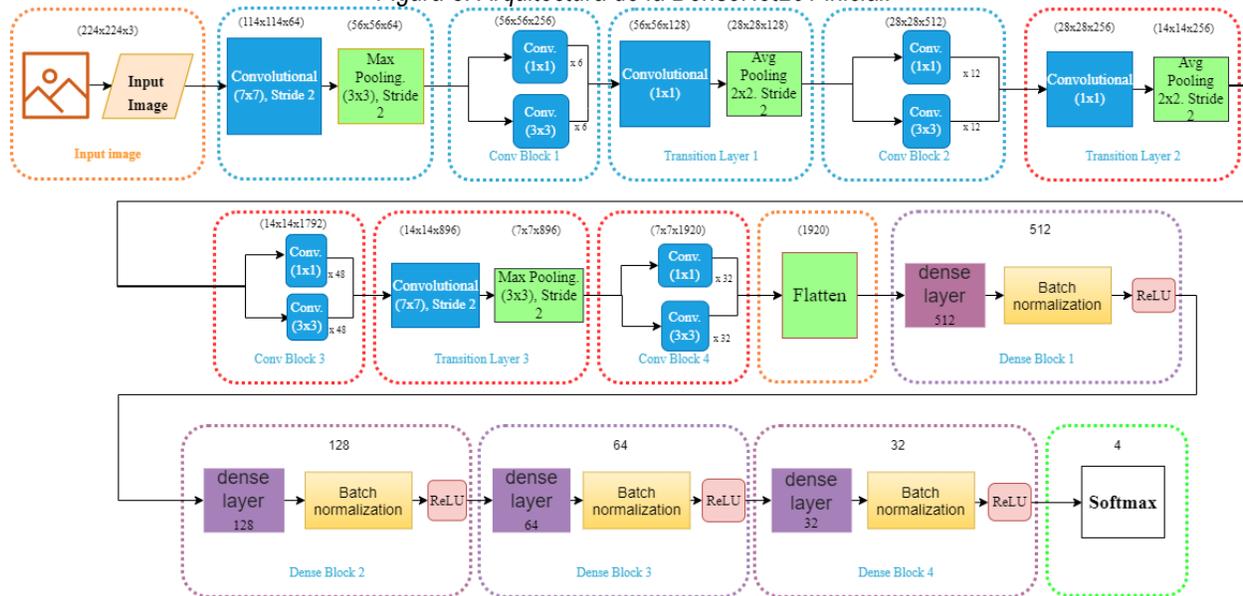
Fuente: elaboración propia

#### 4.7.2 Optimización de hiper parámetros

Dados los experimentos anteriores, se observan mejoras al entrenar con los metadatos, por lo que se decidió realizar un tuneo de nuevo a los hiper parámetros con el objetivo de subir aún más los resultados, donde hay que aclarar por medio de la **¡Error! No se encuentra el origen de la referencia.**, las redes neuronales, o capas densas modificadas. En esta imagen se observa en las primeras capas de color azul las convoluciones para la extracción de características, luego después del Flatten en los recuadros morados se observan las capas densas que inicialmente fueron de 512, 128, 64,32 y 2. Donde estas capas le permiten al modelo aprender de las características adquiridas de las convoluciones y así poder predecir en qué nivel socioeconómico pertenece el hogar encuestado.



Figura 5: Arquitectura de la DenseNet201 inicial.



Fuente: elaboración propia

En la Tabla 15 se evidencia todas las capas densas utilizadas para el entrenamiento la unión de los grupos del Sisbén (1, 2) y (3,4) con metadatos, de donde la mejor combinación de capas densas fue de 1024, 512 y 2. Mejorando los porcentajes actuales obtenidos, siendo de 72% su precisión.

Tabla 15: Ajuste de capas densas del entrenamiento la unión de los grupos del Sisbén (1, 2) y (3,4)

Capas Densas	Accuracy	F1-Score	Recall	Precision
1024,512,2	0.74	0.72	0.72	0.73
2048,1024,2	0.73	0.72	0.73	0.72
1024,512,128,2	0.74	0.72	0.72	0.73
128,64,2	0.74	0.72	0.72	0.73
32,16,4,2	0.74	0.72	0.72	0.73
128,64,16,2	0.74	0.72	0.72	0.74
256,2	0.74	0.72	0.71	0.72
256,128,32,2	0.74	0.72	0.7	0.73
512,256,64,2	0.73	0.72	0.72	0.72
512,256,64,32,2	0.74	0.72	0.72	0.73
512,256,2	0.73	0.71	0.70	0.72
32,16,2	0.74	0.71	0.70	0.74
64,32,8,4,2	0.74	0.71	0.70	0.74
1024,512,128,64,2	0.72	0.71	0.709	0.71
64,32,2	0.74	0.71	0.70	0.74
512,2	0.73	0.70	0.69	0.74
128,64,16,8,2	0.73	0.70	0.70	0.73
256,128,32,16,2	0.73	0.70	0.69	0.73
2048,1024,256,2	0.69	0.68	0.69	0.68
256,128,2	0.68	0.64	0.64	0.68



<b>32,16,4,2,2</b>	0.59	0.37	0.50	0.29
<b>128,2</b>	0.59	0.37	0.50	0.29
<b>1024,2</b>	0.59	0.37	0.50	0.29
<b>32, 2</b>	0.59	0.37	0.50	0.29
<b>64,2</b>	0.59	0.37	0.50	0.29
<b>2048,2</b>	0.59	0.37	0.50	0.29
<b>64,32,8,2</b>	0.59	0.37	0.50	0.29

Fuente: elaboración propia

Por otro lado, la Tabla 16 evidencia los optimizadores probados, para llevar a cabo el tuneo de los hiper parámetros de las capas densas anteriores, obteniendo los mayores porcentajes los optimizadores RMSprop y Adam, siendo de 72% para ambos el porcentaje de precisión.

*Tabla 16: Ajuste de optimizadores del entrenamiento la unión de los grupos del Sisbén (1, 2) y (3,4)*

Optimizadores	Accuracy	F1-Score	Recall	Precision	AUC
Adam	0.73	0.72	0.72	0.72	0.72
RMSprop	0.74	0.72	0.72	0.73	0.72
Adamax	0.73	0.69	0.69	0.73	0.69
Nadam	0.72	0.68	0.68	0.74	0.68
SGD	0.69	0.67	0.66	0.67	0.66
Adagrad	0.68	0.66	0.66	0.67	0.66
Adadelta	0.67	0.65	0.65	0.65	0.65
Ftrl	0.59	0.37	0.50	0.29	0.50

Fuente: elaboración propia

La Tabla 17 por su parte evidencia todas las capas densas utilizadas para el entrenamiento del grupo 1 contra el 2 con metadatos, siendo la mejora combinación de capas densas de 1024, 512 y 2. Con resultados de 66% respectivamente.

*Tabla 17: Ajuste de capas densas del entrenamiento del grupo 1 contra el 2*

Capas Densas	Accuracy	F1-Score	Recall	Precision
<b>32,16,2</b>	0.73	0.67	0.68	0.68
<b>1024,512,2</b>	0.74	0.66	0.65	0.68
<b>256,128,32,16,2</b>	0.74	0.65	0.64	0.69
<b>64,32,2</b>	0.74	0.66	0.65	0.68
<b>1024,512,128,64,2</b>	0.74	0.65	0.64	0.68
<b>256,2</b>	0.74	0.64	0.63	0.69
<b>256,128,32,2</b>	0.74	0.64	0.63	0.70
<b>512,256,64,32,2</b>	0.74	0.63	0.62	0.70
<b>128,64,16,2</b>	0.74	0.63	0.62	0.70
<b>1024,512,128,2</b>	0.74	0.62	0.61	0.70
<b>1024,2</b>	0.69	0.62	0.61	0.62
<b>256,128,2</b>	0.74	0.61	0.61	0.69
<b>64,32,8,4,2</b>	0.74	0.61	0.61	0.72
<b>2048,2</b>	0.73	0.61	0.60	0.69
<b>128,64,2</b>	0.74	0.61	0.60	0.71
<b>128,64,16,8,2</b>	0.73	0.61	0.60	0.68
<b>512,256,2</b>	0.60	0.60	0.60	0.71



64,32,8,2	0.74	0.59	0.59	0.72
2048,1024,256,2	0.71	0.55	0.56	0.64
512,256,64,2	0.70	0.54	0.56	0.66
32, 2	0.70	0.41	0.50	0.35
512,2	0.70	0.41	0.50	0.35
32,16,4,2,2	0.70	0.41	0.50	0.35
64,2	0.70	0.41	0.50	0.35
128,2	0.70	0.41	0.50	0.35
32,16,4,2	0.70	0.41	0.50	0.35
2048,1024,2	0.36	0.33	0.53	0.54

Fuente: elaboración propia

La Tabla 18 evidencia el mejor optimizador para las capas densas anteriores con mayor porcentaje, evidenciando que el mejor optimizador fue el Adam con porcentaje de 64%.

*Tabla 18: Ajuste de optimizadores del entrenamiento del grupo 1 contra el 2*

Optimizadores	Accuracy	F1-Score	Recall	Precision	AUC
Adam	0.75	0.64	0.63	0.70	0.63
Nadam	0.64	0.63	0.67	0.64	0.67
SGD	0.71	0.61	0.60	0.62	0.60
Adamax	0.71	0.60	0.60	0.64	0.60
RMSprop	0.73	0.58	0.59	0.71	0.59
Adagrad	0.69	0.57	0.57	0.60	0.58
Adadelta	0.67	0.56	0.56	0.58	0.56
Ftrl	0.70	0.41	0.50	0.35	0.50

Fuente: elaboración propia

Por último, la Tabla 19 se evidencia los mejores optimizadores para el entrenamiento del grupo 3 contra el 4, en donde no hubo necesidad de tunear las capas densas debido a que para los 2 modelos anteriores las mejores capas fueron 1024, 512 y 2, las cuales fueron tomadas para este modelo también. Siendo el mejor optimizador el Adam con 58% aproximadamente.

*Tabla 19: Ajuste de optimizadores del entrenamiento del grupo 3 contra el 4*

Optimizadores	Accuracy	F1-Score	Recall	Precision	AUC
Adam	0.61	0.57	0.57	0.57	0.57
SGD	0.59	0.53	0.53	0.54	0.53
Adamax	0.62	0.55	0.55	0.56	0.55
Adadelta	0.60	0.53	0.53	0.54	0.53
Adagrad	0.60	0.53	0.538	0.54	0.538
Nadam	0.65	0.50	0.54	0.60	0.54
RMSprop	0.65	0.39	0.50	0.61	0.50
Ftrl	0.65	0.39	0.50	0.32	0.50

Fuente: elaboración propia

### **6.8. Ensamble final, después de la búsqueda de hiper parámetros de mejor rendimiento.**

Finalmente, en la Tabla 20, se evidencia la unión de los 3 modelos anteriores, obteniendo un resultado de 43%.



*Tabla 20: Resultado del Ensamble final después del tuneo de hiper parámetros.*

<b>Modelo</b>	<b>Accuracy</b>	<b>F1-Score</b>	<b>Recall</b>	<b>Precision</b>
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.45	0.43	0.42	0.46

Fuente: elaboración propia

## **7. Conclusiones y recomendaciones**

1. Se hallaron imágenes en condiciones no deseadas para el modelo de estimación del nivel socioeconómico mediante las fachadas de los hogares encuestados, debido a que se presentaban imágenes con obstáculos, que no permitían captar completamente la fachada, al igual que imágenes que no pertenecían a las fachadas, entre otras imágenes con sombras y muy mala resolución.
2. Una vez recibida la totalidad de imágenes (cerca de 1 millón), el modelo de depuración de imágenes se tuvo que volver a entrenar debido a que la naturaleza de las nuevas imágenes era diferente con las que se entrenó inicialmente. Adicional, se recomienda que este modelo de depuración se ajuste dada algunas características regionales que pueden presentar las fachadas, es decir una fachada en la costa es diferente a una en Bogotá. Lo que brindaría más seguridad y precisión al momento de depurar las imágenes nuevas de una encuesta.
3. Adicional a esto, se observó un aumento en cuanto al rendimiento del modelo al momento de utilizar los metadatos, brindando una mejora 0.7% a la hora de clasificar los 4 niveles socioeconómicos y esto se observa en las *Tabla 2* y *Tabla 21*, siendo de gran utilidad a la hora de clasificar de manera correcta los grupos socioeconómicos y obtener más datos de la fachada, brindando mayor precisión y rendimiento en el entrenamiento.
4. Las características socioeconómicas del grupo A (Pobreza extrema) y B (Pobreza media), C (No pobres Vulnerables), y D (No Pobres) presentan similitudes que hacen difícil la discriminación entre ellas. Al combinar estos grupos se consigue maximizar la separabilidad entre clases partiendo que las características de A y B son menos similares que las de C y D, lo que nos da de entrada predecir los grupos pobres y no pobres, obteniendo resultados superiores al 70%.
5. Se evidenció que alrededor del 60% de las imágenes no presentan las condiciones mínimas para una estimación del nivel socioeconómico, a partir de ellas, debido a que presentan obstáculos, que no permitían captar completamente la fachada, al igual que imágenes que no pertenecían a las fachadas, entre otras imágenes con sombras y muy mala resolución. Se recomienda fuertemente que para aumentar la precisión de la metodología propuesta, se estandarice la captura de imágenes evitando al máximo que se presenten las condiciones anteriormente mencionadas.
6. La inclusión de metadatos al modelo de imágenes significó una mejora en la precisión del 0.7%. La información que contienen los metadatos ayuda a complementar la información que no se puede obtener de las fachadas tales como material de los pisos, paredes y tipo de vivienda, siendo estos los metadatos usados, concluyendo que aumentando la cantidad de datos obtenidos de la encuesta como son tipos de baños, personas en el hogar, ingresos, ayudaran significativamente al modelo a predecir una nueva encuesta incluyendo las fotos de las fachadas.
7. En general estos modelos conllevan un alto costo computacional y para mejorar los resultados es necesario contar con buenas prestaciones computacionales para realizar el entrenamiento y mayor número de imágenes con las condiciones mínimas deseadas.

## **8. Socialización**

*El proyecto ha sido socializado únicamente en la universidad autónoma de Manizales y la UCD.*

### **Contacto**

Si tiene alguna duda, comentario o sugerencia sobre este proyecto, o si le gustaría conversar con la Unidad de Científicos de Datos sobre la posibilidad de una nueva fase para el mismo, puede comunicarse con nosotros a través del correo electrónico [ucd@dnpp.gov.co](mailto:ucd@dnpp.gov.co).



**ANEXOS**

*Tabla 21: Resultados clasificación de los 4 grupos de Sisbén con metadatos.*

<b>Modelo</b>	<b>Accuracy</b>	<b>F1-Score</b>	<b>Recall</b>	<b>Precision</b>
<b>DenseNet201 Con los pesos de imagenet y optimizador Adam</b>	0.47	0.44	0.43	0.43

Fuente: elaboración propia