

Predicciones de ocurrencia de precipitaciones mediante Machine Learning en la provincia de Manabí, Ecuador.

Predictions of the occurrence of rainfall using Machine Learning in the province of Manabí, Ecuador.

Rojas Ochoa Luis Enrique ¹, Rondal Viscaíno Nicolás Alejandro ¹, Yauli Lalaleo Vanesa Lorena ¹.

¹Universidad Central del Ecuador

Resumen: La región costera del Ecuador ha sido afectada por inundaciones producto de lluvias que representan un grave riesgo al conjugarse con otras condiciones del terreno. En el presente estudio se describe el proceso por el cual se realiza la predicción de la ocurrencia o ausencia de lluvias en base al análisis de imágenes satelitales por medio del aprendizaje automático; se analiza la provincia de Manabí por ser considerada una de las más afectadas por este tipo de fenómenos.

Se genera un dataset de imágenes satelitales que reflejen las características de cuando ocurren lluvias sobre los 3mm/h, utilizando técnicas de aprendizaje profundo se extraen las características principales que servirán para entrenar un modelo que finalmente permita predecir la ocurrencia o ausencia de lluvias. Con estos resultados se puede realizar un análisis más específico en zonas cuya susceptibilidad a inundaciones causadas por precipitaciones ha sido demostrada, esto permitirá generar planes de mitigación ante los fenómenos de inundación por parte de la autoridad competente. En zonas vulnerables se requerirá la atención inmediata y evaluación por parte de los técnicos y autoridades correspondientes.

Palabras clave: aprendizaje automático, redes neuronales artificiales, predicción, lluvias, riesgo

Abstract: The coastal region of Ecuador has been affected by floods caused by rains that pose a serious risk when combined with other conditions in the land. The present study describes the process by which the prediction of the occurrence or absence of rainfall is made based on the analysis of satellite images through machine learning; The province of Manabí is analyzed because it is considered one of the most affected by this type of phenomenon.

A dataset of satellite images that reflect the characteristics of when rains occur over 3mm/h is generated, using deep learning techniques are extracted the main characteristics that will serve to train a model that finally allows to predict the occurrence or absence of rains. With these results a more specific analysis can be carried out in areas where susceptibility to floods caused by precipitation has been demonstrated, this will allow generating mitigation plans for flood events by the competent authority. In vulnerable areas, immediate attention and evaluation by the relevant authorities and technicians will be required.

Keywords: machine learning, artificial neural networks, prediction, rainfall, risk

1. Introducción

Las épocas lluviosas en la costa ecuatoriana han generado graves daños al sector agrícola, ganadero e industrial, por lo que identificar la ocurrencia de precipitaciones y asociarlas con posibles zonas vulnerables a inundaciones, permitirá tomar medidas en materia de gestión de riesgo y ordenamiento territorial.

En el presente trabajo se aprovecha el Aprendizaje Automático (Machine Learning) para realizar dicha identificación. Esta técnica supone un gran avance en el desarrollo de la inteligencia artificial y es una alternativa para el tratamiento de diferentes problemáticas. (González, 2020)

Con este fin se ha elaborado un dataset a partir de la adquisición de imágenes satelitales e información de intensidad de precipitaciones, este sirve de entrada para un modelo de redes neuronales convolucionales (CNN) que permite a la máquina aprender y reconocer patrones, para luego realizar predicciones sobre determinado tema. (Bagnato, 2020)

Rull, 2016 señala que estas redes, son un tipo de arquitectura del aprendizaje profundo, que ayuda a extraer las características de una imagen para su clasificación, que en este caso será binaria, teniendo dos posibles resultados, precipitación o ausencia de precipitación.

Como caso de estudio se analiza a la provincia de Manabí que se encuentra en el centro de la región costera del país, se extiende entre los 0°25' latitud norte y 1°57' latitud sur, y desde los 79°24' de longitud oeste hasta los 88°55' de longitud oeste (*Fig. 1*). La longitud de la línea costera llega a los 354 km y se extiende desde los poblados de Cojimíes a Ayampe. Limita al norte con la provincia de Esmeraldas, al Sur con Santa Elena al este con Santo Domingo de los Tsáchilas, Los Ríos y Guayas, y al oeste con el Océano

Pacífico. (CONSEJO PROVINCIAL DE MANABÍ, 2015)

Las contribuciones que genera el estudio son:

- Un dataset extenso y confiable a partir de la recopilación de imágenes satelitales de servicios meteorológicos en los que consten datos de nubes y precipitaciones.
- Un modelo basado en el Deep Learning para poder estimar la ocurrencia de lluvias y asociarlo con estudios previos que establecen zonas de riesgo ante inundaciones para el caso específico de la provincia de Manabí, Ecuador.

Ambos productos se encontrarán disponibles al público en sitios de almacenamiento en línea como GitHub y OneDrive

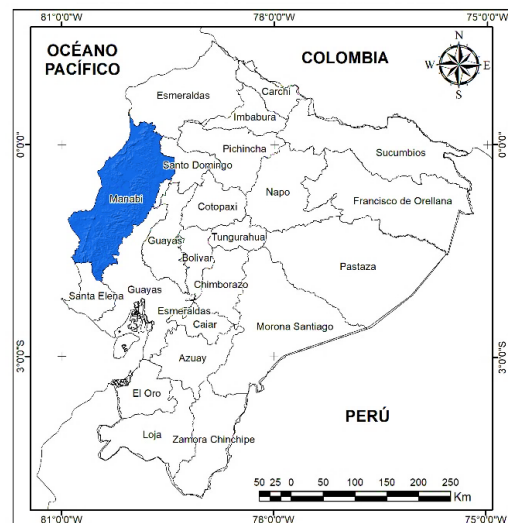


Figura 1. Mapa de ubicación del área de estudio.

2. Materiales y Métodos

2.1 Hardware y Software.

Las principales características de la plataforma computacional utilizada son: un equipo con procesador Intel Core i7 de 2.00 GHz, sistema operativo Windows 10, Memoria RAM de 12Gb y tarjeta de video AMD Radeon R7 de 4 Gb de memoria de video.

Para el software se trabaja con el Navegador Anaconda utilizando el entorno de desarrollo integrado de JupyterLab en la que utilizamos las librerías TensorFlow y Keras para la elaboración de la red neuronal, Matplotlib para la visualización de imágenes y, Numpy para operaciones y análisis de datos. Todas estas corresponden a lenguaje Python en su versión 3.16.

Las imágenes satelitales se guardan en el disco duro del computador además del uso del almacenamiento en la nube mediante Microsoft OneDrive.

2.2 Metodología

El flujograma de la *Fig. 2*, representa la metodología utilizada en este trabajo, comenzando con la elaboración del dataset, su posterior división, preprocesamiento de las imágenes, creación del modelo, el compilado y entrenamiento del modelo, su evaluación y finalmente la predicción y la clasificación de las imágenes.

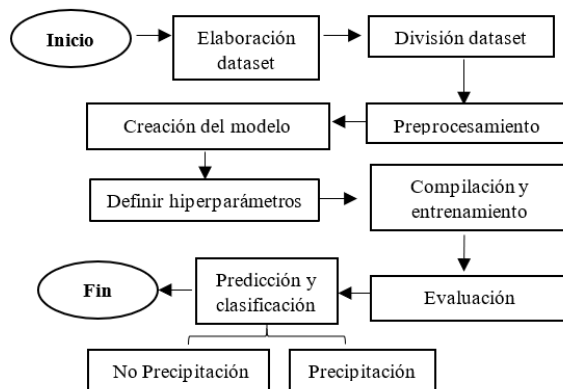


Figura 2. Flujograma de la metodología aplicada.

2.2.1. Elaboración del dataset

Las imágenes fueron obtenidas de forma gratuita de NASA WORLDVIEW (<https://worldview.earthdata.nasa.gov>), se tomaron en cuenta aquellos filtros de la página relacionados a las precipitaciones, las imágenes tienen un formato .JPG y tienen una resolución promedio de 410 x 410 píxeles, la variabilidad de tamaños se debe a la forma de capturar la imagen, sin

embargo, esta variación será corregida en etapas posteriores.

Estas imágenes se organizaron en dos carpetas: una carpeta donde se almacenan aquellas imágenes que registraron lluvias, y se las etiqueta usando el código LL-000; una segunda carpeta donde se almacenan aquellas imágenes que no registraron lluvias, estas se guardaron con el código NL-000. Esta estructura de directorios se puede ver en la *Fig. 3*.

El dataset se compone de 1280 imágenes repartidas equitativamente en cada categoría (640 imágenes cada una). Esta es una buena práctica en el aprendizaje automático para evitar sesgos o preferencias.

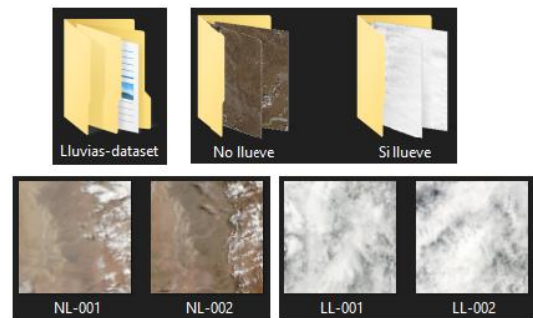


Figura 3: Estructura de directorios del dataset

2.2.2. División del dataset.

Para esta división se utilizó la librería *Sklearn* y su comando *train_test_split* que separa el conjunto de datos en entrenamiento y prueba, y cuya proporción está dada en un 80% de los datos para la parte de entrenamiento y el restante 20% corresponde a la parte de prueba. Las imágenes de cada conjunto se seleccionaron de manera aleatoria, se utilizó el parámetro *random state = 42*, para mantener esta selección a medida que se vuelve a ejecutar el código. Esta división se hace con el fin de hacer que los datos tengan una mejor precisión en su evaluación, la carpeta de entrenamiento se emplea únicamente en la creación del modelo mientras que la de prueba se usa en predicciones.

La distribución total de las imágenes del dataset se describe en la *Tabla 3*.

Tabla 3. Proporción de la división de los datos en entrenamiento y prueba.

Variable	Número de imágenes	Porcentaje
Entrenamiento	1024	80%
Prueba	256	20%
Data set	1280	100%

2.2.3. Preprocesamiento

Debido a la variedad de tamaños que tienen las imágenes, se utiliza el software en BatchPhoto en su versión de prueba gratuita para redimensionarlas a 400x400 píxeles, estas dimensiones se tomaron con el fin de no perder la resolución original y hacer una aproximación al promedio de tamaño de todas las imágenes.

El siguiente paso es normalizar las imágenes por lo que transformamos a

variables de tipo flotante para poder trabajar con un arreglo que puede ser manejado por la librería *Numpy*. Además, se requiere que los píxeles de las imágenes tengan valores únicamente entre 0 y 1 por lo que se dividen los píxeles de cada imagen entre 255.

Este proceso sirve para optimizar los tiempos de procesamiento en los siguientes pasos. Los datos se almacenan en listas por lo que utilizamos la librería *Numpy* para transformarlas en vectores.

Los parámetros de entrada para el modelo son las imágenes (X) y las etiquetas de 0 y 1, “No llueve” y “Si llueve” respectivamente (Y).

2.2.4. Creación del modelo

Para la ejecución del modelo es necesario definir una serie hiperparámetros, por ejemplo, el número y tamaño de los filtros que usará nuestra red neuronal para extraer las características para clasificar las imágenes (*Fig. 3*).

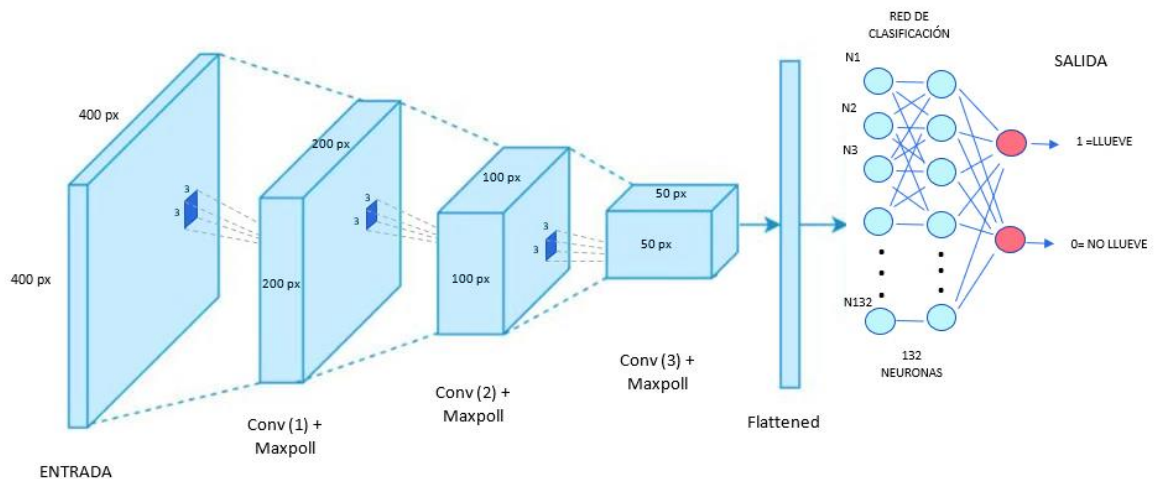


Figura 4. Esquema del modelo utilizado y sus hiperparámetros

El modelo tiene una base convolucional Conv 2D ya que estamos trabajando con imágenes en 2 dimensiones, se compone de 3 capas de convolución con 32, 64 y 128 filtros respectivamente, todos estos filtros tienen una dimensión de 3x3 píxeles, que se encargan de extraer las principales

características de la imagen y crear mapas de dichas características. En cada capa se utiliza una función de activación *Relu* que permite al modelo evitar valores negativos que pueden afectar el comportamiento del modelo.

La *Figura 4* muestra el esquema del modelo en el que la imagen de entrada se somete a filtros que extraen las características principales, se organizan en una matriz y se aplica una capa de submuestreo que reduce el tamaño de la imagen a la mitad (*Maxpoll*) para seguir extrayendo características de más alto nivel o abstracción.

Esta operación se repite en 2 capas adicionales. Le sigue una capa de *flattened* que se encarga de transformar o “aplanar” las matrices a un vector, que serán los datos de entrada para la una red neuronal clásica compuesta por 132 neuronas (N), esta cantidad de neuronas se toma gracias al ajuste que genera en el modelo al momento de realizar la predicción, aquí se aplica una función sigmoideal que arroja un resultado entre 0 y 1, los valores mayores a 0.5 se aproximan a 1 (Llueve) y si son menores o iguales a 0.5 se aproximan a 0 (No llueve), es decir, el resultado final corresponde a un valor de probabilidad de ocurrencia de precipitaciones entre 0 y 1.

Al aplicar las capas convolucionales y la red neuronal clásica se puede utilizar el comando *summary* para presentar un resumen del modelo con los parámetros obtenidos (*Fig. 5*), aquí se reflejan las dimensiones de las imágenes al entrar y salir de cada capa convolucional, al final se presenta un resumen con el total de parámetros encontrados, así como aquellos que serán entrenados en la siguiente etapa, en este caso se tiene más de 42 millones de parámetros a entrenar.

Layer (type)	Output Shape	Param #
conv2d_4 (Conv2D)	(None, 400, 400, 32)	896
max_pooling2d_4 (MaxPooling2)	(None, 200, 200, 32)	0
dropout_4 (Dropout)	(None, 200, 200, 32)	0
conv2d_5 (Conv2D)	(None, 200, 200, 64)	18496
max_pooling2d_5 (MaxPooling2)	(None, 100, 100, 64)	0
dropout_5 (Dropout)	(None, 100, 100, 64)	0
conv2d_6 (Conv2D)	(None, 100, 100, 128)	73856
max_pooling2d_6 (MaxPooling2)	(None, 50, 50, 128)	0
flatten_1 (Flatten)	(None, 320000)	0
dense_2 (Dense)	(None, 132)	42240132
leaky_re_lu (LeakyReLU)	(None, 132)	0
dropout_6 (Dropout)	(None, 132)	0
dense_3 (Dense)	(None, 2)	266
=====		
Total params: 42,333,646		
Trainable params: 42,333,646		
Non-trainable params: 0		

Figura 5. Resumen del modelo producto del comando summary

2.2.5. Compilación y entrenamiento.

Mediante el comando *compile* se configura el proceso de entrenamiento, este comando requiere aclarar la función de pérdida, que en este caso corresponde a *categorical_crossentropy* además de un optimizador (*adam*) que establece la tasa de aprendizaje (*learning rate*) del modelo, para este caso se emplea una tasa de 0.001.

Se utiliza la instrucción *fit()* para entrenar el modelo con lotes de 64 imágenes (*batch_size*) en un total de 10 épocas o iteraciones. El tiempo que tarda el entrenamiento depende plenamente de las características del equipo utilizado y de la extensión del dataset, además del número de épocas que se realicen, en este caso se realizó en un tiempo aproximado de 40 minutos.

Con el fin de comprobar que el modelo que se está tomando en este estudio es el que mejores resultados presenta se realizaron varios modelos más, uno de estos trató de aplicar una función de activación diferente (linear), sin embargo, en este caso, al analizar las gráficas de precisión y pérdida se puede ver que se genera un “underfitting” que corresponde a un desajuste de la capacidad de aprendizaje de la máquina por debajo de la media (*Fig. 6*).

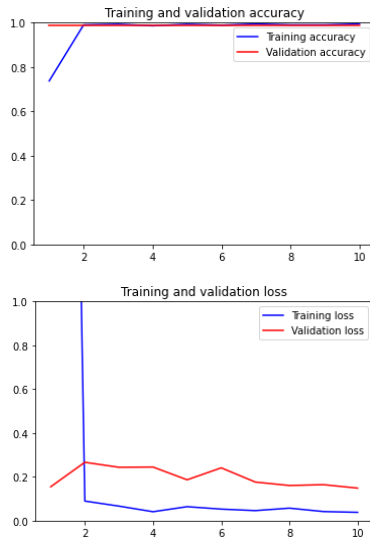


Figura 6. Gráficas de precisión y pérdida con *underfitting*

Teniendo en cuenta el modelo previo se realizó un tercer modelo que además de aplicar la función de activación Relu, reduce el número de neuronas de la capa densa de 132 a 64, además se redujo el número de filtros que se aplican a cada capa de convolución, en este caso se aplicaron 32 filtros de tamaño similar al modelo ideal (3x3) en cada capa, permitiendo así entrenar menos parámetros.

El resultado de esta variación de modelo evidencia un “overfitting” o sobreentrenamiento por parte de la máquina, es decir, la máquina aprende características específicas de memoria, siendo incapaz de encontrar características diferentes en nuevas imágenes, esto se refleja en la gráfica de precisión y pérdida de la Fig. 7, en la que se puede ver que la precisión del modelo varía irregularmente a lo largo de las iteraciones realizadas.

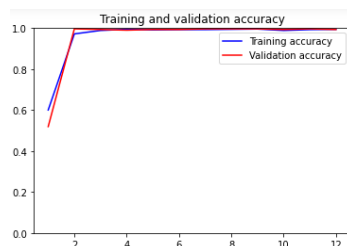


Figura 7. Gráficas de precisión y pérdida con *overfitting*

2.2.6. Evaluación

En esta etapa se utilizan los datos de prueba que se separaron en etapas previas.

La librería *matplotlib* permite representar las gráficas de pérdida y precisión (Fig. 8.) para los conjuntos de entrenamiento y validación del modelo ideal descrito en la Fig. 3.

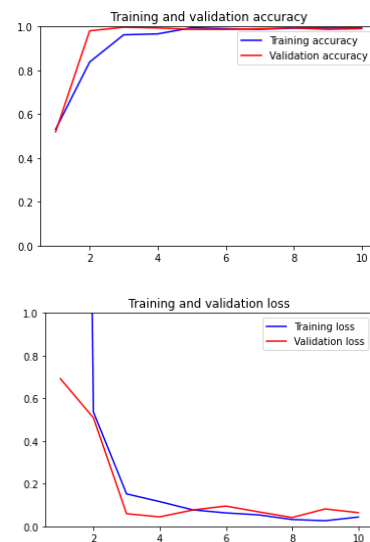


Figura 8: Gráficas de precisión (a) y pérdida (b) para la carpeta de entrenamiento

En las gráficas de la Fig. 8 se puede ver un comportamiento adecuado del aprendizaje del modelo, evidenciando un aumento constante de la precisión y un descenso constante del error del modelo hasta un valor aceptable, que en este caso se evidencia en la época 5, luego de la cual se vuelve constante. Al tener un comportamiento similar de las curvas de entrenamiento y validación se puede considerar como adecuado el

entrenamiento con los hiperparámetros establecidos.

La matriz de confusión se genera mediante el comando `confusión_matrix` y corresponde a una comparación numérica entre aciertos y fallos a partir de los datos de prueba del modelo que en este caso corresponde a 256, es decir, permite analizar los verdaderos positivos, verdaderos negativos (diagonal principal), falsos positivos y falsos negativos (diagonal secundaria) generados en el modelo (Recuerdo, 2018) (Fig. 9). Esta matriz evalúa los parámetros que se muestran en la Fig. 10. Se analiza la precisión, la sensibilidad y especificidad (recall) y una relación entre ambas (f1-score).

```
array([[141,  0],
       [ 3, 112]], dtype=int64)
```

Figura 9. Matriz de confusión del modelo

	precision	recall	f1-score	support
Class 0	0.97	1.00	0.98	126
Class 1	1.00	0.97	0.98	130
accuracy			0.98	256
macro avg	0.98	0.98	0.98	256
weighted avg	0.98	0.98	0.98	256

Figura 10. Parámetros evaluados en una matriz de confusión aplicados al modelo

La precisión en este modelo es mayor en el caso de la clase 1, es decir, en la categoría de lluvias, entendiendo que se predecirá de mejor manera cuando exista probabilidad de ocurrencia de precipitaciones, mientras que la sensibilidad y especificidad es mayor para la clase 0 (no llueve). Sin embargo, al hacer una relación entre estos dos factores es de 0.98 (F1-score) con lo que podemos calificar el modelo como adecuado para el fin requerido.

2.2.7. Predicción y clasificación

La predicción de imágenes comienza identificando aquellas que fueron etiquetadas de forma correcta (Fig. 11.) y

errónea, (Fig. 12.) durante el entrenamiento. Dentro del código se imprimen las imágenes junto a su respectiva etiqueta, sea esta correcta o incorrecta.

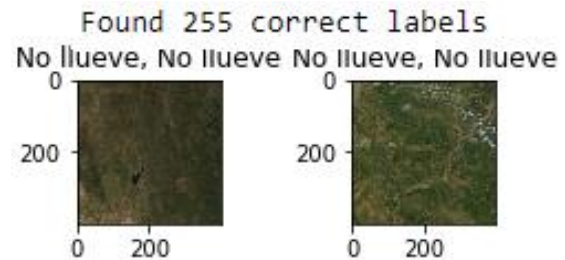


Figura 11. Imágenes etiquetadas correctamente

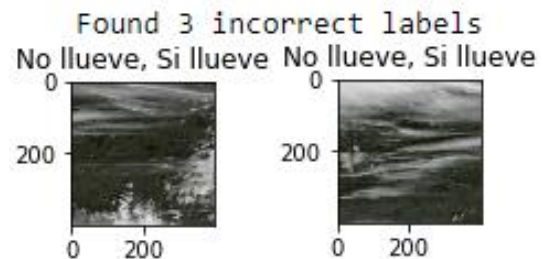


Figura 12. Imágenes etiquetadas de forma errónea.

Como un ejemplo de la predicción del modelo usamos la Fig. 13, correspondiente al lunes 22 de marzo del 2021 en la provincia de Manabí que arrojó como resultado la etiqueta de no llueve.



D:/Prueba Software/Pred/pred3.jpg No llueve

Figura 13. Ejemplo de predicción del modelo

3. Resultados y Discusión

Al analizar las gráficas de precisión y pérdida (Fig 8.) se evidencia que se alcanzó una precisión del 98%. En estas graficas se

observa también que no existe un sobreajuste lo que quiere decir que el margen de error del modelo en nuevas predicciones será muy bajo basado en las características aprendidas, y tampoco tendrá un desajuste ya que se ha trabajado con un base de datos extensa. (Bagnato, Aprende Machine Learning, 2017).

Por estas razones se puede considerar este modelo como válido.

Por otra parte, según la Secretaria de Gestión de Riesgos (2014), en la provincia de Manabí los escenarios que representan factores de riesgo respecto a inundaciones son las lluvias intensas y localizadas, lluvias de baja intensidad, pero de duración prolongada, lluvias en las cabeceras de cuencas hidrográficas y lluvias que coinciden con estados de marea alta. También, según INFOPLAN IGM (2021) se establece que la región costera presenta una constante amenaza morfoclimática ante inundaciones causadas, entre otros factores, por las lluvias torrenciales, quedando en detrás únicamente de la provincia del Guayas.

Este modelo se puede acoplar a estudios en escalas más pequeñas por lo que puede considerarse un complemento a algunos estudios como el caso de Lucas (2017), que está centrado en la susceptibilidad del cantón Tosagua ante la ocurrencia de inundaciones por lluvias intensas, generando el desbordamiento de ríos, como es el caso también el caso del Cantón Flavio Alfaro donde Cartaya & Mantuano (2016) identifican manchas de inundación con la simulación de las crecidas del río Pescadillo, brindando información sobre posibles zonas agrícolas en riesgo.

Existen algunos casos donde las inundaciones no están ligadas solamente al factor de las precipitaciones sino, a otros factores como la ubicación geográfica, factores económicos y sociales mencionado en el estudio de Burgos, Cartaya & Valle (2019) donde analiza la vulnerabilidad de la parroquia Santa Ana y

del cantón Rocafuerte realizado por Pacheco (2018).

En el caso de Pacheco, Rodríguez, Loor & Arteaga (2019) en el que se estudia específicamente la Parroquia La Unión, analizando los impactos económicos que tienen las zonas más vulnerables a inundaciones, se puede hacer una correlación adecuada en base a la estimación de periodos de lluvia en las zonas de alta vulnerabilidad.

Es por esto que el análisis de la ocurrencia de lluvias se relaciona directamente con la amenaza ante inundaciones. En base a la información arrojada como resultado se podrían establecer ciertos periodos de tiempo en los cuales la amenaza de inundación se ve aún más clara en el área debido a la presencia del Fenómeno del Niño ya que, según INFOPLAN (2021) la mayor parte de las inundaciones en la costa se restringen a estos periodos de tiempo.

4. Conclusiones.

Se ha visto a lo largo de la historia que este tipo de fenómenos (precipitación e inundaciones) puede generar graves afectaciones tanto a infraestructuras como a los habitantes de la provincia de Manabí.

Los datos obtenidos en el presente estudio pueden ser correlacionados con estudios que determinan zonas con mayor exposición a fenómenos desastrosos como inundaciones, la correcta difusión de los resultados entre las autoridades competentes puede representar la diferencia entre una correcta gestión del riesgo y la ocurrencia del desastre, principalmente en la época del año en la que el Fenómeno del Niño afecta gravemente a las costas ecuatorianas.

En este caso las técnicas de aprendizaje profundo han servido realizar una predicción de la ocurrencia de las precipitaciones, implementando una red convolucional que extrajo las características de cada imagen, para luego utilizar una red neuronal que permita

clasificarlas y etiquetarlas en caso de que llueva o no llueva.

El código usado en el presente estudio se puede encontrar en la plataforma GitHub (https://github.com/luis3106179/Prediccion_lluvias.git), el acceso al mismo es libre, al igual que a la base de datos empleada que se encuentra almacenada en la plataforma de OneDrive (<https://rb.gy/ilyllj>)

Referencias

- Bagnato, J. (12 de Diciembre de 2017). *Aprende Machine Learning*. Obtenido de Qué es overfitting y underfitting y cómo solucionarlo: <https://www.aprendemachinelarning.com/que-es-overfitting-y-underfitting-y-como-solucionarlo/>
- Bagnato, J. (31 de Mayo de 2020). Obtenido de Redes Neuronales Convolucionales: https://github.com/jbagnato/machine-learning/blob/master/Ejercicio_CN_N.ipynb
- Burgos, B., Cartaya, S., & Valle, D. M. (2019). Análisis de la vulnerabilidad a inundaciones de la parroquia Santa Ana de Vuelta Larga, provincia de Manabí, Ecuador. *Investigaciones Geográficas*(98), 1-14.
- Cartaya, S., & Mantuano, R. (2016). Identificación de zonas en riesgo de inundación mediante la simulación hidráulica en un segmento del Río Pescadillo, Manabí, Ecuador. *SciELO*, 40(89), 158-170.
- CONSEJO PROVINCIAL DE MANABÍ. (2015). Obtenido de Plan de desarrollo y ordenamiento Territorial Manabí 2015-2024: https://www.manabi.gob.ec/doc_transparencia/doc_internos/21_s/PDyOT%20Manabi%20actualizado.pdf
- Demoraes, F., & D´ercole, R. (2001). *SAVGIS*. Obtenido de CARTOGRAFÍA DE RIESGOS Y CAPACIDADES EN EL ECUADOR: http://www.savgis.org/SavGIS/Etudes_realisees/DEMORAES_DERCOLE_Cartografia_riesgos_2001.pdf
- González, A. (2020). *Cleverdata*. Obtenido de ¿Qué es Machine Learning?: <https://cleverdata.io/que-es-machine-learning-big-data/>
- INFOPLAN. (2021). *Gestión de Riesgos*. Obtenido de Atlas, Espacios geográficos expuesto a Amenazas naturales y antrópicas: <https://nube.gestionderiesgos.gob.ec/index.php/s/posFfNmWXfHPna7#pdfviewer>
- Lucas, C. (2017). *SUSCEPTIBILIDAD A INUNDACIONES EN LA COMUNIDAD EL JUNCAL, CANTÓN TOSAGUA, PROVINCIA DE MANABÍ*.
- NASA . (2021). Obtenido de NASA WorldView: <https://worldview.earthdata.nasa.gov/>
- PACHECO, Á. (2018). *SUSCEPTIBILIDAD A INUNDACIONES POR INTENSAS*.
- Pacheco, T., Rodríguez, E., Loor, L., & Arteaga, I. (2019). Valoración económica del impacto de la inundación de febrero 2019 en la Unión, Santa Ana, Manabí,

Ecuador. *Revista San Gregorio*(43), 21-35.

- Recuero, P. (23 de Enero de 2018).
Obtenido de Machine Learning a
tu alcance: La matriz de confusión:
[https://empresas.blogthinkbig.com/
ml-a-tu-alcance-matriz-confusion/](https://empresas.blogthinkbig.com/ml-a-tu-alcance-matriz-confusion/)
- Rull, M. V. (2016). *Reconocimiento de
Objetos usando Deep Learning*.

Secretaria de gestión de riesgos. (Enero de
2014). *Gestión de riesgos*.
Obtenido de Amenaza
Inundaciones:
[https://www.gestionderiesgos.gob.
ec/wp-
content/uploads/downloads/2014/0
4/Escenario_ENERO_Inundaci%C
3%B3n_Mensual_1.pdf](https://www.gestionderiesgos.gob.ec/wp-content/uploads/downloads/2014/04/Escenario_ENERO_Inundaci%C3%B3n_Mensual_1.pdf)