

UNIVERSIDAD NACIONAL DE JAÉN

FACULTAD DE INGENIERÍA



**UNIVERSIDAD NACIONAL
DE JAÉN**

CARRERA PROFESIONAL DE INGENIERÍA CIVIL

**ESTIMACIÓN DEL NIVEL DE RIESGO POR
DESLIZAMIENTO EMPLEANDO ALGORITMOS
ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN –
CAJAMARCA, 2025**

**TESIS PARA OBTENER
EL TÍTULO PROFESIONAL DE INGENIERO CIVIL**

**Autores: Bach. Monsalve Ticliahuanca Einstein Edinson
Bach. Vílchez Silva Franklin Fernando**

Asesor: Mg. Ing. Billy Alexis Cayatopa Calderón

Coasesor: Dr. Lenin Quiñones Huatangari

Línea de investigación: LI_IC_03 Gestión y Tecnología

JAÉN – PERÚ

JUNIO 2026

Vílchez Silva Franklin F. Y Monsalve Ticliahuanca Ei...

ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA...

 Quick Submit Quick Submit Universidad Nacional de Jaén

Detalles del documento

Identificador de la entrega

trn:oid::1:3667089922

183 páginas

Fecha de entrega

2 oct 2026, 11:52 a.m. GMT-5

47.470 palabras

Fecha de descarga

2 oct 2026, 11:57 a.m. GMT-5

238.900 caracteres

Nombre del archivo

liva_Informe_Final_de_Tesis_-_FRANKLIN_FERNANDO_VLCHEZ_SILVA.pdf

Tamaño del archivo

29.6 MB



UNIVERSIDAD NACIONAL DE JAÉN



Dra. María Marleni Torres Cruz
Responsable de la Unidad de Investigación
de la Facultad de Ingeniería

7% Similitud general

El total combinado de todas las coincidencias, incluidas las fuentes superpuestas, para ca...

Filtrado desde el informe

- Bibliografía
- Texto citado
- Texto mencionado
- Coincidencias menores (menos de 15 palabras)

Fuentes principales

- 6%  Fuentes de Internet
- 1%  Publicaciones
- 5%  Trabajos entregados (trabajos del estudiante)

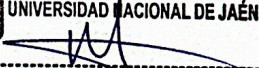
Marcas de integridad

N.º de alerta de integridad para revisión

-  **Texto oculto**
27 caracteres sospechosos en N.º de página
El texto es alterado para mezclarse con el fondo blanco del documento.

Los algoritmos de nuestro sistema analizan un documento en profundidad para buscar inconsistencias que permitirían distinguirlo de una entrega normal. Si advertimos algo extraño, lo marcamos como una alerta para que pueda revisarlo.

Una marca de alerta no es necesariamente un indicador de problemas. Sin embargo, recomendamos que preste atención y la revise.

 UNIVERSIDAD NACIONAL DE JAÉN

Dra. Marta Marient Torres Cruz
Responsable de la Unidad de Investigación
de la Facultad de Ingeniería



FORMATO 03: ACTA DE SUSTENTACIÓN

En la ciudad de Jaén, el día 30 de setiembre de 2026, siendo las 17:00 horas, se reunieron los integrantes del Jurado:

Presidente: Dr. Manuel Emilio Milla Pino

Secretaria: Dra. Zadith Nancy Garrido Campaña

Vocal: M. Sc. Marcos Antonio Gonzales Santisteban, para evaluar la Sustentación del Informe Final:

() Trabajo de Investigación

(**X**) Tesis

() Trabajo de Suficiencia Profesional

Titulado:

ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA – JAÉN – CAJAMARCA, 2025, presentado por los tesisistas: **Einstein Edinson Monsalve Ticlahuanca y Franklin Fernando Vilchez Silva**, de la Escuela Profesional de Ingeniería Civil de la Universidad Nacional de Jaén.

Después de la sustentación y defensa, el Jurado acuerda:

(**X**) Aprobar () Desaprobar (**X**) Unanimidad () Mayoría

Con la siguiente mención:

- | | | |
|----------------|------------|---------------|
| a) Excelente | 18, 19, 20 | () |
| b) Muy bueno | 16, 17 | (16) |
| c) Bueno | 14, 15 | () |
| d) Regular | 13 | () |
| e) Desaprobado | 12 ó menos | () |

Siendo las 18:00 horas del mismo día, el Jurado concluye el acto de sustentación confirmando su participación con la suscripción de la presente.

Dr. Manuel Emilio Milla Pino
Presidente

Dra. Zadith Nancy Garrido Campaña
Secretario

M. Sc. Marcos Antonio Gonzales Santisteban
Vocal

“Año de la Esperanza y el Fortalecimiento de la Democracia”

ANEXO N°06:

**DECLARACIÓN JURADA DE AUTENTICIDAD Y DE NO PLAGIO
DE LA TESIS O TRABAJO DE INVESTIGACIÓN (PREGRADO)**

Yo, EINSTEIN EDINSON MONSALVE TICLIAHUANCA, egresado de la carrera profesional de Ingeniería Civil de la Facultad de Ingeniería de la Universidad Nacional de Jaén, identificado con DNI N.º 72720089.

Declaro bajo juramento que:

1. Soy Autor del trabajo titulado:

“ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN – CAJAMARCA, 2025”.

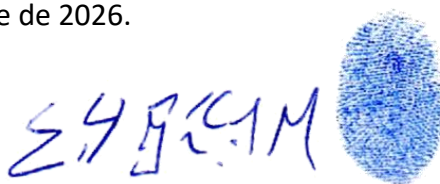
Asesorado por el Mg. Ing. Billy Alexis Cayatopa Calderón y coasesorado por el Dr. Lenin Quiñones Huatangari.

El mismo que presento bajo la modalidad de tesis para optar el Título Profesional de Ingeniero Civil.

2. El texto de mi trabajo final respeta y no vulnera los derechos de terceros, incluidos los derechos de propiedad intelectual. En el sentido, el texto de mi trabajo final no ha sido plagiado total ni parcialmente, para la cual he respetado las normas internacionales de citas y referencias de las fuentes consultadas.
3. El texto del trabajo final que presento no ha sido publicado ni presentado antes en cualquier medio electrónico o físico.
4. La investigación, los resultados, datos, conclusiones y demás información presentada que atribuyo a mi autoría son veraces.
5. Declaro que mi trabajo final cumple con todas las normas de la Universidad Nacional de Jaén.
6. Soy consciente de que el hecho de no respetar los derechos de autor y hacer plagio, es objeto de sanciones universitarias y/o legales.

El incumplimiento de lo declarado da lugar a responsabilidad del declarante, en consecuencia; a través del presente documento asumo frente a terceros, la Universidad Nacional de Jaén y/o la Administración Pública toda responsabilidad que pueda derivarse por el trabajo final presentado. Lo señalado incluye responsabilidad pecuniaria incluido el pago de multas u otros por los daños y perjuicios que se ocasionen.

Fecha: Jaén, 23 de octubre de 2026.



EINSTEIN EDINSON MONSALVE TICLIAHUANCA

Firma y huella del autor

“Año de la Esperanza y el Fortalecimiento de la Democracia”

ANEXO N°06:

**DECLARACIÓN JURADA DE AUTENTICIDAD Y DE NO PLAGIO
DE LA TESIS O TRABAJO DE INVESTIGACIÓN (PREGRADO)**

Yo, FRANKLIN FERNANDO VÍLCHEZ SILVA, egresado de la carrera profesional de Ingeniería Civil de la Facultad de Ingeniería de la Universidad Nacional de Jaén, identificado con DNI N.º 74878670.

Declaro bajo juramento que:

1. Soy Autor del trabajo titulado:

“ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN – CAJAMARCA, 2025”.

Asesorado por el Mg. Ing. Billy Alexis Cayatopa Calderón y coasesorado por el Dr. Lenin Quiñones Huatangari.

El mismo que presento bajo la modalidad de tesis para optar el Título Profesional de Ingeniero Civil.

2. El texto de mi trabajo final respeta y no vulnera los derechos de terceros, incluidos los derechos de propiedad intelectual. En el sentido, el texto de mi trabajo final no ha sido plagiado total ni parcialmente, para la cual he respetado las normas internacionales de citas y referencias de las fuentes consultadas.
3. El texto del trabajo final que presento no ha sido publicado ni presentado antes en cualquier medio electrónico o físico.
4. La investigación, los resultados, datos, conclusiones y demás información presentada que atribuyo a mi autoría son veraces.
5. Declaro que mi trabajo final cumple con todas las normas de la Universidad Nacional de Jaén.
6. Soy consciente de que el hecho de no respetar los derechos de autor y hacer plagio, es objeto de sanciones universitarias y/o legales.

El incumplimiento de lo declarado da lugar a responsabilidad del declarante, en consecuencia; a través del presente documento asumo frente a terceros, la Universidad Nacional de Jaén y/o la Administración Pública toda responsabilidad que pueda derivarse por el trabajo final presentado. Lo señalado incluye responsabilidad pecuniaria incluido el pago de multas u otros por los daños y perjuicios que se ocasionen.

Fecha: Jaén, 23 de octubre de 2026.


FRANKLIN FERNANDO VÍLCHEZ SILVA
Firma y huella del autor

ÍNDICE GENERAL

ÍNDICE GENERAL	i
ÍNDICE DE FIGURAS.....	ii
ÍNDICE DE TABLAS	iv
RESUMEN	viii
ABSTRACT.....	ix
I INTRODUCCIÓN	1
II MATERIAL Y MÉTODOS.....	12
III RESULTADOS.....	79
IV DISCUSIÓN	92
V CONCLUSIONES Y RECOMENDACIONES.....	96
VI REFERENCIAS BIBLIOGRÁFICAS.....	99
AGRADECIMIENTO	108
DEDICATORIA	109
ANEXOS	110

ÍNDICE DE FIGURAS

Figura 1 <i>Ubicación geográfica del área de estudio</i>	13
Figura 2 <i>Metodología CENEPRED</i>	17
Figura 3 <i>Metodología general para la estimación del nivel de riesgo</i>	18
Figura 4 <i>Vista de la ladera del sector La colina</i>	20
Figura 5 <i>Vista de la ladera del sector La colina</i>	21
Figura 6 <i>Pasajes con superficie de tierra en la ladera del sector La Colina</i>	22
Figura 7 <i>Condición actual de las viviendas en el sector La Colina</i>	23
Figura 8 <i>Vía nacional, Carretera Jaén – San Ignacio al margen izquierdo del sector La Colina</i>	24
Figura 9 <i>Erosión en surcos y cárcavas en corte de talud</i>	25
Figura 10 <i>Precipitaciones máximas por día entre los años 2016 – 2025</i>	26
Figura 11 <i>Mapa de geología en el sector La Colina procesado en software QGIS</i> ..	29
Figura 12 <i>Mapa de geomorfológico en el sector La Colina procesado en software QGIS</i>	31
Figura 13 <i>Portal de descargas de elevaciones digitales ALOS PALSAR</i>	33
Figura 14 <i>Mapa de pendiente en el sector La Colina procesado en software QGIS</i>	34
Figura 15 <i>Portal de descargas de Bing Maps – satellite</i>	35
Figura 16 <i>Mapa de Uso actual del suelo en el sector La Colina procesado en software QGIS</i>	36
Figura 17 <i>Procesamiento en el software QGIS de los niveles de Peligro en la zona de Estudio</i>	40
Figura 18 <i>Referenciación de lotes en Google Earth para trabajo en campo</i>	57
Figura 19 <i>Uso de AppSheet para digitalizar el cuestionario</i>	57
Figura 20 <i>Aplicación de cuestionario en campo</i>	58

Figura 21 <i>Procesamiento de información de campo a QGIS.</i>	59
Figura 22 <i>Variables de entrada de la base de datos.</i>	63
Figura 23 <i>Importación de librerías de Python y definición de la variable objetivo...</i>	64
Figura 24 <i>Ubicación de la variable objetivo en la base de datos.</i>	66
Figura 25 <i>Detección duplicadas por formato</i>	66
Figura 26 <i>Eliminación de registros sin salidas y definición de x e y.</i>	67
Figura 27 <i>Label Encoder para codificación de la variable objetivo</i>	68
Figura 28 <i>Configuración del esquema de validación</i>	69
Figura 29 <i>Preprocesamiento con OneHotEncoder</i>	70
Figura 30 <i>Código para guardar los mejores modelos base optimizados</i>	72
Figura 31 <i>Funciones para la creación del meta-aprendiz y del clasificador Stacking</i>	74
Figura 32 <i>Definición de combinaciones de modelos ensamblados mediante Stacking</i>	74
Figura 33 <i>Validación cruzada y métricas de evaluación para seleccionar mejor ensamble</i>	75
Figura 34 <i>Distribución porcentual de niveles de peligro del área de estudio del sector La Colina</i>	80
Figura 35 <i>Distribución de frecuencias de niveles de peligro del área de estudio del sector La Colina.</i>	80
Figura 36 <i>Distribución porcentual de niveles de vulnerabilidad del área de estudio del sector La Colina.</i>	82
Figura 37 <i>Distribución de frecuencias de niveles de vulnerabilidad del sector La Colina.</i>	82
Figura 38 <i>Distribución porcentual de niveles de riesgo del sector La Colina</i>	84

Figura 39 Distribución de frecuencias de niveles de riesgo del sector La Colina	84
Figura 40 Distribución de los niveles de riesgo de la base de datos utilizada	85
Figura 41 Comparación de las métricas de evaluación de los modelos individuales	86
Figura 42 Comparación de las métricas de evaluación de los modelos ensamblados	87
Figura 43 Matrices de confusión de Gradient Boosting y el ensamble CB_1	89
Figura 44 Comparación de ensambles soft voting con Accuracy, F1 macro, y Cohen's Kappa	90
Figura 45 Matrices de confusión de los ensambles soft voting ponderado	91

ÍNDICE DE TABLAS

Tabla 1 Rangos de precipitación	27
Tabla 2 Peso ponderado del parámetro de evaluación – Periodo de retorno.....	28
Tabla 3 Peso ponderado del factor condicionante – Geología	30
Tabla 4 Peso ponderado del factor condicionante – Geomorfología	32
Tabla 5 Peso ponderado del factor condicionante – Pendiente	34
Tabla 6 Peso ponderado factor condicionante – Uso Actual del Suelo.....	36
Tabla 7 Rangos de precipitación	37
Tabla 8 Peso ponderado del factor desencadenante – Precipitación.....	38
Tabla 9 Estratificación del nivel de peligro.....	38
Tabla 10 Peso ponderado del parámetro cantidad de habitantes por vivienda	41
Tabla 11 Descriptores del parámetro grupo etario más vulnerable en la vivienda ...	41
Tabla 12 Peso ponderado del parámetro personas con dificultad permanente	42
Tabla 13 Peso ponderado del parámetro abastecimiento de agua en la vivienda	42

Tabla 14 <i>Peso ponderado del parámetro disposición sanitaria de excretas en la vivienda</i>	43
Tabla 15 <i>Peso ponderado del parámetro fuente de energía eléctrica</i>	43
Tabla 16 <i>Peso ponderado del parámetro nivel educativo más alto en la vivienda</i>	44
Tabla 17 <i>Peso ponderado del parámetro capacitación de la población en temas de GRD</i>	44
Tabla 18 <i>Peso ponderado del parámetro tipo de seguro de salud de la población ...</i>	45
Tabla 19 <i>Peso ponderado del parámetro percepción de la población acerca de los deslizamientos</i>	45
Tabla 20 <i>Peso ponderado del parámetro cercanía de la vivienda a la zona de deslizamientos</i>	46
Tabla 21 <i>Peso ponderado del parámetro antigüedad de la vivienda</i>	46
Tabla 22 <i>Peso ponderado del parámetro material de construcción de paredes de la vivienda</i>	47
Tabla 23 <i>Peso ponderado del parámetro material de construcción en techo de la vivienda</i>	47
Tabla 24 <i>Peso ponderado del parámetro material de construcción en piso de la vivienda</i>	48
Tabla 25 <i>Peso ponderado del parámetro estado de conservación de la vivienda.....</i>	48
Tabla 26 <i>Peso ponderado del parámetro altura de la vivienda</i>	49
Tabla 27 <i>Peso ponderado del parámetro protección de la vivienda</i>	49
Tabla 28 <i>Peso ponderado del parámetro ingreso familiar promedio mensual</i>	50
Tabla 29 <i>Peso ponderado del parámetro condición laboral del jefe/a de hogar</i>	50
Tabla 30 <i>Peso ponderado del parámetro localización de áreas agrícolas frente a la zona de deslizamiento</i>	51

Tabla 31 <i>Peso ponderado del parámetro malas prácticas de población que degradan el suelo</i>	51
Tabla 32 <i>Peso ponderado del parámetro cercanía de botaderos de basura</i>	52
Tabla 33 <i>Peso ponderado del parámetro conocimiento de actividades de conservación ambiental.....</i>	52
Tabla 34 <i>Estratificación de los niveles de vulnerabilidad en el área de estudio – Sector La Colina</i>	53
Tabla 35 <i>Cálculo de ICC.....</i>	55
Tabla 36 <i>Coeficiente Kappa de Cohen</i>	56
Tabla 37 <i>Matriz de riesgo</i>	60
Tabla 38 <i>Estratificación de los niveles de riesgo en el área de estudio – Sector La Colina.....</i>	60
Tabla 39 <i>Algoritmos seleccionados de Machine Learning.</i>	71
Tabla 40 <i>Combinaciones de algoritmos para el ensamblaje con stacking.....</i>	73
Tabla 41 <i>Configuración de los ensambles adicionales mediante soft voting ponderado</i>	77
Tabla 42 <i>Rangos de niveles de peligro del sector La Colina.....</i>	79
Tabla 43 <i>Peligro de deslizamientos por área de la zona de estudio - sector La Colina</i>	79
Tabla 44 <i>Rangos para la clasificación de niveles de vulnerabilidad en el sector La Colina.....</i>	81
Tabla 45 <i>Distribución de los niveles de vulnerabilidad del sector La Colina.....</i>	81
Tabla 46 <i>Rangos para la clasificación de niveles de riesgo en el sector La Colina ..</i>	83
Tabla 47 <i>Distribución de los niveles de riesgo del sector La Colina</i>	83

Tabla 48 Rendimiento de los modelos individuales mediante validación cruzada de 10 particiones.....	85
Tabla 49 Rendimiento de los modelos ensamblados mediante validación cruzada de 10 particiones.....	87
Tabla 50 Métricas de la evaluación final con <code>cross_val_predict</code>	88
Tabla 51 Métricas finales de los ensambles soft voting con la función “ <code>cross_val_predict</code> ”	90

RESUMEN

La investigación tuvo como objetivo estimar el nivel de riesgo por deslizamiento mediante algoritmos ensamblados en el sector La Colina, Jaén, Cajamarca. Fue aplicada, cuantitativa, descriptiva y no experimental. Se evaluaron 123 viviendas en 8,0786 ha mediante la metodología del Centro Nacional de Estimación, Prevención y Reducción del Riesgo de Desastres (CENEPRED), con información cartográfica y de campo. Para el modelamiento se integraron 1 092 registros, 123 de La Colina y 969 de Jangas, con 29 predictores y el nivel de riesgo como variable objetivo. Se compararon cuatro modelos individuales y siete ensambles mediante validación cruzada estratificada, considerando el F1-score macro como métrica principal. Los resultados mostraron que el 73 % del área presentó peligro alto y el 27 % peligro muy alto. De las viviendas evaluadas, el 74,8 % presentó vulnerabilidad alta y el 80,5 % riesgo alto según CENEPRED. El ensamble SV_2, conformado por Gradient Boosting, Random Forest y Extra Trees, obtuvo 86,45 % de exactitud, 85,67 % de F1-score macro y un Kappa de 0,751. Se concluyó que La Colina presenta riesgo elevado y que el soft voting ponderado constituye una herramienta complementaria para clasificar sus niveles de riesgo.

Palabras clave: Riesgo por deslizamiento, CENEPRED, aprendizaje automático, algoritmos ensamblados, soft voting.

ABSTRACT

This study estimated landslide risk levels using ensemble algorithms in the La Colina sector of Jaén, Cajamarca, Peru. The research followed an applied, quantitative, descriptive, and non-experimental design. Using cartographic and field data, 123 dwellings within an 8.0786-ha area were assessed according to the methodology developed by Peru's Centro Nacional de Estimación, Prevención y Reducción del Riesgo de Desastres (CENEPRED). The supervised learning dataset comprised 1,092 records—123 from La Colina and 969 from Jangas—29 predictors, and landslide risk level as the target variable. Four standalone models and seven ensemble configurations were compared through stratified cross-validation, with macro-averaged F1 as the primary performance metric. Results indicated that 73% of the study area was classified as high hazard and 27% as very high hazard. Among the surveyed dwellings, 74.8% had high vulnerability, while CENEPRED classified 80.5% as high risk. The SV_2 ensemble, combining Gradient Boosting, Random Forest, and Extra Trees through weighted soft voting, achieved 86.45% accuracy, 85.67% macro F1, and a Cohen's kappa of 0.751. The findings indicate that La Colina faces generally high landslide risk and support weighted soft voting as a complementary tool for risk-level classification

Keywords: Landslide risk, CENEPRED, machine learning, ensemble algorithms, soft voting.

I INTRODUCCIÓN

1.1 Planteamiento del Problema

Los deslizamientos de terreno representan una de las amenazas naturales destructivas, ocasionando pérdidas económicas significativas y un elevado número de víctimas mortales. Estos impactos se intensifican por el cambio climático y la expansión demográfica a zonas vulnerables (Grima et al., 2020). Según el Centro de Investigación de Epidemiología de Desastres (CRED, 2024) en 2023 se registraron 399 desastres naturales, que resultaron en 86,473 muertes y afectaron a 93.1 millones de personas, con un detrimento financiero aproximado de 202,700 millones de dólares. En Kivu del Sur, perteneciente a la República Democrática del Congo, las inundaciones y deslizamientos dejaron un estimado de 2,790 fallecidos. Entre 2004 y 2010, se reportaron 2,620 deslizamientos de tierra con 32,322 decesos (Petley, 2012). La norma ISO 22327:2018 destaca que los deslizamientos de tierra pueden ocurrir en diversos contextos topográficos, agravados por un uso de la tierra expansivo e incontrolado, destaca también los sistemas de alerta temprana como medida adecuada para la mitigación de riesgos; en este contexto, los algoritmos de aprendizaje automático e IA son vitales para mejorar predicciones inmediatas y a corto plazo. (ONU, 2023).

En el Perú, los deslizamientos de tierra se presentan de manera recurrente en los valles bajos, amazónicos e interandinos de los ríos Marañón, Huallaga, Apurímac y Urubamba, ocasionando daños significativos a comunidades e infraestructuras (Ordoñez, 2019). Uno de los casos ocurrió el 5 de febrero de 2023 en el distrito de Mariano Nicolás Valcárcel, en Camaná, Arequipa, donde un deslizamiento afectó a alrededor de 2,000 personas y destruyó 496 viviendas Save the Children (2024). Ante este panorama, es imprescindible incorporar herramientas basadas en inteligencia artificial para mejorar la detección, análisis y evaluación del riesgo por deslizamientos. El uso de algoritmos de machine learning, como el desarrollado

por el Instituto Geológico del Perú junto con la Université Côte d’Azur para la localización y estimación sísmica rápida, evidencia el potencial de esta tecnología para generar alertas tempranas y fortalecer las acciones de prevención y mitigación en el país (Lara et al., 2023). Esto adquiere especial relevancia para regiones como Cajamarca, donde provincias como Jaén y San Ignacio registran una de las mayores incidencias de deslizamientos a nivel nacional (Ordoñez, 2019).

En la región Cajamarca se han registrado peligros geológicos significativos debido a deslizamientos, especialmente en la provincia de Cajabamba, entre el 2022 y 2023. Estos eventos han sido provocados por efectos post sísmicos, lluvias extraordinarias y factores antrópicos. Como consecuencia, se ha destruido áreas de cultivo, postes de cableado eléctrico, viviendas, infraestructuras comunitarias y carreteras (Instituto Geológico, Minero y Metalúrgico [INGEMMET], 2024). Entre 2019 y 2023, se contabilizaron 83 deslizamientos en el departamento de Cajamarca, y hasta el año 2024 se han reportado 28 eventos adicionales ((Instituto Nacional de Defensa Civil [INDECI], 2019-2024).

En Jaén, el informe de INDECI de 2021 destaca que, entre mayo y abril de ese año, los deslizamientos de rocas afectaron a 27 personas, dañaron 10 viviendas y destruyeron 150 metros de carreteras (INDECI, 2021). Entre 2019 y 2024, el portal de INDECI registró un total de 12 informes, sobre deslizamientos ocurridos en la provincia de Jaén. Según los informes, el factor común detrás de estos deslizamientos son las intensas precipitaciones pluviales. Durante este periodo, se han registrado daños significativos, incluyendo 1,448 km de carreteras afectadas (INDECI, 2019-2024). La gestión de riesgos en la ciudad de Jaén ha sido descuidada, y la existencia de investigaciones específicas es escasa, por lo que la evaluación de riesgos se vuelve urgente, en este sentido, la investigación de Muñoz y Yamunaque de 2024 evaluó el riesgo por inundación en el sector Magllanal empleando el algoritmo random forest, abriendo la ventana a vincular el aprendizaje automático y la evaluación de riesgos en la ciudad.

El sector La Colina y la margen izquierda de la carretera Jaén - San Ignacio son considerados de alto riesgo geológico debido a la presencia de suelos cohesivos que tienden a expandirse bajo la influencia de las precipitaciones, lo que incrementa la probabilidad de daños estructurales en las edificaciones (INDECI, 2005). La topografía irregular de la zona, caracterizada por taludes cercanos a la carretera, agrava el riesgo de deslizamientos. Este peligro se ve exacerbado por la construcción de viviendas en zonas vulnerables, donde la combinación de la inestabilidad del terreno y las condiciones climáticas incrementa la susceptibilidad a desprendimientos. Frente a la realidad observada, se planteó la siguiente pregunta de investigación:

¿Cuál es el nivel de riesgo por deslizamiento estimado empleando algoritmos ensamblados en el sector La Colina - Jaén - Cajamarca - 2025?

1.2 Justificación

La investigación se justifica metodológicamente porque integró el procedimiento de evaluación de riesgos del Centro Nacional de Estimación, Prevención y Reducción del Riesgo de Desastres (CENEPRED) con técnicas de aprendizaje automático, estructuradas bajo el enfoque CRISP-DM. En este proceso, los parámetros de peligro y vulnerabilidad fueron empleados como variables de entrada, mientras que los niveles de riesgo determinados mediante el procedimiento metodológico del CENEPRED constituyeron la variable objetivo para el modelamiento supervisado. De esta manera, la investigación no reemplazó el procedimiento de evaluación del riesgo del CENEPRED, sino que lo complementó mediante el uso de algoritmos ensamblados, permitiendo evaluar y comparar el desempeño de diferentes modelos y seleccionar el que presentó un comportamiento más equilibrado en la clasificación de los niveles de riesgo por deslizamiento en el sector La Colina.

Desde el punto de vista social y práctico, la investigación se justifica porque generó información técnica sobre la distribución de los niveles de peligro, vulnerabilidad y riesgo por

deslizamiento en las viviendas evaluadas del sector La Colina, un ámbito urbano expuesto a condiciones físicas y sociales relacionadas con este fenómeno. Asimismo, respondió a la necesidad de contar con estudios específicos sobre deslizamientos en la ciudad de Jaén. Los resultados constituyen un insumo técnico que puede apoyar la toma de decisiones en gestión del riesgo de desastres, planificación urbana y priorización de medidas de prevención y reducción del riesgo, especialmente en contextos donde se requiere identificar espacialmente viviendas con mayores niveles de peligro, vulnerabilidad y riesgo.

La investigación también se justifica por su aporte a la gestión pública del riesgo de desastres, debido a que generó información organizada y clasificada que puede servir como elemento de apoyo para las autoridades locales y entidades competentes. En ese sentido, el estudio se alineó con los principios de la Ley N.º 29664, que establece la responsabilidad de las instituciones públicas en la gestión del riesgo de desastres. Además, los resultados pueden constituir información complementaria para futuros procesos de actualización de instrumentos de planificación territorial y urbana, así como para la formulación y priorización de intervenciones orientadas a la prevención y reducción del riesgo en sectores críticos.

A nivel científico, la investigación se justificó porque incorporó herramientas de ciencia de datos en la evaluación del riesgo por deslizamiento, mediante el empleo de modelos individuales y algoritmos ensamblados. La evaluación de los modelos permitió determinar el comportamiento de diferentes estrategias de clasificación y seleccionar el ensamble con mejor desempeño equilibrado entre las categorías de riesgo, aportando evidencia sobre la aplicabilidad del aprendizaje automático como herramienta complementaria en estudios de gestión del riesgo. De esta forma, la investigación contribuyó a vincular procedimientos de evaluación del riesgo, como el establecido por el CENEPRED, con técnicas de aprendizaje automático, generando una referencia metodológica susceptible de ser adaptada y evaluada en otros ámbitos con características y variables comparables

1.3 Objetivos

1.3.1 Objetivo General

Estimar el nivel de riesgo por deslizamiento empleando algoritmos ensamblados en el sector La Colina, distrito y provincia de Jaén, departamento de Cajamarca.

1.3.2 Objetivos Específicos

- O.E.1. Identificar el nivel de peligro ante deslizamientos de tierra en el sector La Colina, Jaén.
- O.E.2. Analizar el nivel de vulnerabilidad ante el deslizamiento en el sector La Colina, Jaén.
- O.E.3. Establecer el nivel del riesgo de deslizamiento en función del peligro y vulnerabilidad en el sector la Colina, Jaén, modelando su comportamiento con algoritmos ensamblados.
- O.E.4. Validar las estimaciones de riesgos del mejor ensamble de algoritmos.

1.4 Antecedentes

1.4.1 A nivel internacional.

Chen et al. (2024) tuvieron como objetivo evaluar la susceptibilidad a deslizamientos en Hong Kong, integrando conocimientos previos sobre la ocurrencia de deslizamientos y modelos de aprendizaje profundo. La metodología empleó datos históricos y temáticos de topografía, geología, hidrología, cobertura del suelo, clima y actividad humana, para entrenar un modelo que combina un Autoencoder Variacional (VAE) y un Perceptrón Multicapa (MLP). Los resultados mostraron que los modelos de aprendizaje profundo superaron a SVM y ANN en exactitud (2-5%), precisión (2-7%), recall (1-3%), F1-score (8-10%) y curva AuROC (2-4%), con un coeficiente Kappa 20% superior. Se concluyó que estos métodos mejoraron significativamente la precisión en la evaluación de la susceptibilidad a deslizamientos en la zona estudiada.

Zhang et al. (2024) llevaron a cabo una investigación con el objetivo de comparar diferentes modelos de aprendizaje automático en la evaluación de la susceptibilidad a deslizamientos de tierra, centrada en el distrito de Conghua, Guangzhou, China. La metodología incluyó la adquisición de datos geospaciales ambientales, seguida de la extracción de características y la preparación del conjunto de datos mediante ingeniería de características para el entrenamiento de modelos. Posteriormente, se aplicaron cuatro modelos de aprendizaje automático: dos algoritmos de integración, Extreme Gradient Boosting (XGBoost) y Random Forest (RF), y dos algoritmos de clasificación única, Support Vector Machine (SVM) y Regresión Logística (LR). Estos modelos fueron evaluados utilizando indicadores estadísticos y la curva ROC. Los resultados mostraron una precisión de 0.998 para XGBoost, 0.996 para RF, 0.965 para SVM y 0.752 para LR. Se concluyó que el modelo XGBoost presentó la mayor capacidad predictiva, seguido de RF, demostrando que estos algoritmos integrados superan a los algoritmos de clasificación única como SVM y LR en términos de predicción.

González et al. (2023) tuvieron por objetivo identificar las áreas susceptibles a deslizamientos en Autlán de Navarro, México. Su metodología integró datos de edafología, uso del suelo, vegetación, curvas de nivel e información climática de la base EICIII V3.2, junto con datos sísmicos y Sistemas de Información Geográfica (SIG). Se utilizó una clasificación de atributos y se aplicaron técnicas como RUSLE3D en Arcmap 10.5, considerando la erosividad de la lluvia, erodabilidad del suelo, cobertura vegetal, la pendiente y prácticas agrícolas ante factores desencadenantes como la precipitación diaria y anual máxima. Los resultados destacaron que, de las 834 manzanas estudiadas en el municipio, 507 presentaron riesgo regular, 269 riesgo alto y 85 riesgo muy alto. Se concluyó que Autlán y Ahuacapan son las localidades con mayor superficie en riesgo alto y muy alto.

Hussain et al. (2022) en su investigación, tuvieron como objetivo identificar y mapear la susceptibilidad a deslizamientos de tierra en la autopista Karakorum en el norte de Pakistán, utilizando algoritmos de aprendizaje automático validados por la técnica InSAR de dispersión persistente. Emplearon la metodología PS-InSAR del radar de apertura sintética interferométrica para estudiar el desplazamiento de modelos recuperados y recopilaron datos de 332 áreas de deslizamiento de tierra a lo largo de la autopista para generar un mapa de inventario, categorizando las áreas en dos tramos del 30% y 70% para la validación y entrenamiento de los modelos. Los algoritmos utilizados fueron el modelo de RF, XGBoost, K Nearest Neighbors (KNN) y Bayes Naive (NB), comparados mediante la técnica de curva ROC para evaluar su precisión. Los resultados mostraron una precisión de 83.08% para RF, 82.15% para XGBoost, 80.31% para KNN y 72.92% para NB. Se concluye que la técnica RF ofrece los mejores resultados, integrándose con los datos de PS-InSAR para crear un nuevo mapa de susceptibilidad a deslizamientos.

Goyes-Peñafiel y Hernandez-Rojas (2021) desarrollaron un estudio, cuyo objetivo fue evaluar la susceptibilidad a deslizamientos de tierra en Popayán, Colombia. La metodología incluyó un inventario de deslizamientos y once factores condicionantes: ocho derivados de un Modelo de Elevación Digital (DEM) y tres variables temáticas (geología superficial, geomorfología y uso del suelo). Para evitar la multicolinealidad, se aplicó Análisis de Componentes Principales (PCA). Se desarrollaron tres propuestas para evaluar la susceptibilidad usando variables continuas y discretas. El desempeño de cada propuesta se evaluó mediante el Área Bajo la Curva (AUC), calculada a partir de las curvas ROC (Característica Operativa del Receptor). Los resultados mostraron que la Propuesta 1, tuvo un AUC de 0.8578, la Propuesta 2, con un AUC de 0.8459, tuvo la mejor evaluación del Índice de Susceptibilidad a Deslizamientos (LSI), mientras que la Propuesta 3, con un AUC de 0.8054,

mostró un desempeño inferior. Se concluyó que, en comparación con el método WoE, las propuestas incrementaron la susceptibilidad alta y muy alta en áreas con topografía compleja, consistente con los deslizamientos reportados.

1.4.2 A nivel Nacional

López et al. (2021) en su investigación tuvo como objetivo determinar un modelo predictivo basado en machine learning con la finalidad de estimar la vulnerabilidad de riesgo de inundación y deslizamiento en las instituciones educativas del Perú. La metodología que se utilizó está basada en Machine learning entrenando el modelo mediante datos de dataset de más de 65 000 registros publicados por el Ministerio de Educación del Perú. Se usaron diferentes algoritmos para su comparación y selección del mejor modelo de estimación de vulnerabilidad, entre ellos se tiene a DT, KNN, perceptrón multicapa y RF. Los resultados determinaron que la tasa de aciertos (con una división de datos K-Folds Cross validation, de 10 iteraciones de datos) es del 92.407 para el riesgo de deslizamiento y de 97.612 para el riesgo de inundación con el algoritmo RF, siendo estos los mejores resultados comparados con otros modelos de machine learning. Se concluye que el Random Forest describe con mayor ajuste los datos analizados, siendo una herramienta de apoyo para elegir acciones ante la vulnerabilidad de riesgo de inundaciones y deslizamiento, como cambiar de sede, o acciones que mitiguen la zona en la que se va a construir la institución.

Condorhuacho y Suni (2023) tuvieron por objetivo determinar los niveles de riesgo por deslizamiento en diferentes áreas ubicadas en la Quebrada Saqramayo, en el Distrito, Provincia y Región de Cusco. La metodología empleada es la de evaluación de riesgos del CENEPRED (2014) a través de la función del riesgo en torno a vulnerabilidad y peligro, se estudió una muestra que consistió de 241 lotes, tuvo 3 etapas, pre campo (información previa y antecedentes), campo (caracterización de la zona de estudio, parámetros de peligro y

vulnerabilidad) y post campo (procesamiento de la información). Los resultados mostraron que, en cuanto al peligro de deslizamientos, 115 lotes presentaron un peligro muy alto, 57 un peligro alto, 30 un peligro medio y 39 un peligro bajo. Respecto a la vulnerabilidad, se encontraron 38 lotes con vulnerabilidad muy alta, 125 en nivel alto, 45 en nivel medio y 33 en nivel bajo. En términos del nivel de riesgo, 109 lotes resultaron con un riesgo muy alto, 61 con un riesgo alto, 36 con un riesgo medio y 35 con un riesgo bajo. Se concluyó finalmente que el riesgo por deslizamiento en la zona de estudio es muy alto.

Muñoz y Solís (2021), en su investigación, tuvieron como objetivo estimar los niveles de riesgo por deslizamiento e inundaciones fluviales en la localidad de Izchaca, distrito y provincia de Anta. La metodología que se empleó es el proceso de análisis jerárquico CENEPRED, estimando el peligro, en el que se consideró factores condicionantes y uno desencadenante de anomalías de precipitación con periodo de retorno de 20 años, por otro lado, se tiene a la vulnerabilidad, con un análisis de población mediante encuestas. Los resultados determinaron que, ante la ocurrencia de deslizamiento, el 32% de la zona estudiada tiene un peligro muy alto, el 55% de esta zona cuenta con una vulnerabilidad muy alta, por lo que se concluye que el 63% del predio tiene un riesgo muy alto. En el caso de inundaciones, el 70% del área tiene un peligro muy alto, y el 61% de esta muestra una vulnerabilidad alta, por lo que se concluye que el 90% de los predios tiene un nivel de riesgo alto.

En un estudio realizado por la Municipalidad provincial del Cusco, Subgerencia de estudios y proyectos (2020), en San Pedro, Cusco que tuvo como objetivo general determinar los niveles de riesgos de deslizamientos en el área de influencia del proyecto de mejoramiento y ampliación del centro de salud de San Pedro AVP se aplicó la metodología CENEPRED, donde como principal resultado se obtuvo que el riesgo ante un deslizamiento en el centro de salud mencionado está en nivel medio, asimismo se podría generar una pérdida de 4339.49 soles por cada metro cuadrado afectado en cuanto a estructuras se refiere, por otra parte, la

tolerancia del riesgo se encuentra en un nivel inaceptable. Este antecedente servirá como guía porque utiliza la misma metodología que se empleará para llevar a cabo esta tesis.

Anchelía y Mori (2020) tuvieron por objetivo determinar el más adecuado método para definir zonas susceptibles a inundaciones en la cuenca baja del río Chancay de la región Lambayeque. La metodología comparó el proceso de análisis jerárquico (AHP), utilizado por el CENEPRED, con el método estadístico Random Forest. Se analizó la cuenca baja del río Chancay Lambayeque, con un área de 1337.57 km². Los resultados mostraron que, utilizando el método AHP, el 7.3% del área estudiada estaba en peligro muy alto, el 19.7% en peligro alto, el 21.2% en peligro medio, el 12.3% en peligro bajo y el 39.5% en nivel muy bajo. En contraste, con Random Forest, el 21.2% del área estaba en riesgo muy alto, el 10.6% en riesgo alto, el 11.6% en riesgo medio, el 5% en riesgo bajo y el 51% en riesgo muy bajo. La comparación, de la curva ROC, arrojó un área bajo la curva de 0.9941 para RF, y 0.9774 para AHP. El estudio concluyó que la evaluación de peligro ante inundaciones es más preciso con el algoritmo Random Forest que con el método AHP.

1.4.3 A nivel regional – local

Palomino et al. (2024) tuvieron como objetivo estimar la vulnerabilidad ante sismos en viviendas construidas con albañilería confinada en el sector Pueblo Libre-Jaén, ciudad de Jaén, Perú. La metodología consistió en algoritmos de ensamblaje, utilizando una base de datos de 3760 viviendas. El 80% de la muestra se usó para entrenar el algoritmo y el 20% para validarlo. Los modelos base entrenados incluyeron RF, Decision Tree (DT), Extra-Trees (ET) y Gradient Boosting (GB). Los resultados indicaron que los modelos lograron precisiones entre 94.95% y 95.90%. En cuanto a la vulnerabilidad sísmica, el método aplicado en 67 viviendas mostró que el 1.48% se encuentran en vulnerabilidad alta, el 32.85% en nivel moderado y el 65.67% en nivel bajo. El estudio concluyó que el modelo de mayor precisión fue el compuesto por DT y

GB, además, aunque la mayoría de las viviendas presentan baja vulnerabilidad, el porcentaje de vulnerabilidad moderada es alarmante.

Muñoz y Yamunaque (2024) en su investigación tuvieron como objetivo evaluar el nivel de riesgo por inundación fluvial en el Sector Magllanal, Jaén, Cajamarca. Utilizaron la metodología de CENEPRED (2014) para estudiar la probabilidad de desborde de lotes a causa de la precipitación, en la que se necesitó la recopilación de información, análisis de elemento expuestos y análisis vulnerabilidad. Dentro de los 68 lotes analizados, los resultados indicaron que 1 lote registra un riesgo muy alto, 47 lotes registran un riesgo alto, y 20 lotes un riesgo medio, dando, así como resultado general un nivel de riesgo alto. Se concluye con la necesidad de implementar medidas estructurales y no estructurales de mitigación, entre estas, la construcción de drenaje fluvial, muros de contención, puentes peatonales, alcantarillas y limpieza de cauce en la zona de intervención.

Idrogo (2023) en su investigación tuvo como objetivo identificar los niveles de riesgo por deslizamientos de tierra en el sector 9 de Chota, Perú, utilizando la metodología del CENEPRED (2014). Su metodología combinó el análisis de datos históricos con observaciones de campo sobre factores como tipo de suelo, inclinación del terreno, capacidad portante y proximidad al río, además de evaluar la vulnerabilidad de las construcciones. Con un diseño descriptivo causal simple, se cuantificó el riesgo en la zona de estudio. Los resultados indicaron que el 32.08% del área presenta un riesgo alto, el 48.13% un riesgo medio, y el 19.80% un riesgo bajo. El análisis de estabilidad de laderas, arrojó que los taludes son estables, con factores de seguridad que varían de 1.874 a 2.136 en análisis estático y de 1.233 a 1.566 en análisis dinámico. El estudio concluyó que el riesgo es predominantemente medio y factores externos como sismos o lluvias podrían incrementar el riesgo, lo que sugiere la necesidad de estudios adicionales.

Tirado (2020) tuvo como objetivo evaluar el riesgo asociado a la vulnerabilidad física en laderas inestables en un tramo de 10 km de la carretera Cajamarca-Gavilán. La metodología de la investigación se planteó en torno a manuales del CENEPRED y el INDECI para estimar riesgos de movimientos en masa, identificando la precipitación como el principal factor desencadenante. Los resultados mostraron que el 76% de las viviendas evaluadas presenta un nivel de riesgo muy alto, mientras que el 24% tiene un riesgo alto. En términos de peligro, el 68% del área estudiada (186.82 Ha) presenta un peligro muy alto, y el 32% (88.99 Ha) un peligro alto. En cuanto a la vulnerabilidad, el 78% de las viviendas presenta un nivel muy alto y el 22% un nivel alto. El estudio concluyó que 76 de las 100 viviendas evaluadas están en riesgo muy alto de sufrir daños socioeconómicos a causa de un deslizamiento, mientras que 24 viviendas están en riesgo alto.

II MATERIAL Y MÉTODOS

2.1 Objeto de estudio

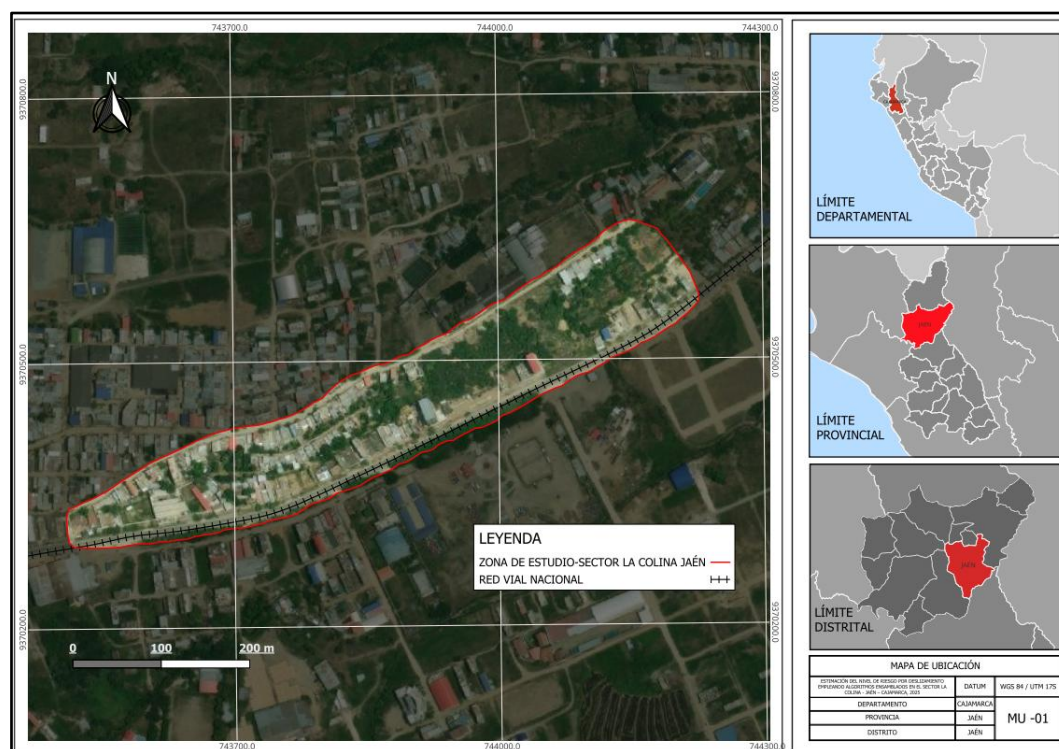
El objeto de estudio fue el riesgo por deslizamiento en las viviendas del sector La Colina, ubicado en el distrito y provincia de Jaén, departamento de Cajamarca. Para su evaluación se consideraron los componentes de peligro y vulnerabilidad presentes en el área de estudio. A partir de los parámetros analizados en ambos componentes, se determinaron los niveles de riesgo y posteriormente fueron empleadas en el modelamiento mediante algoritmos ensamblados de aprendizaje automático.

2.2 Ubicación geográfica

El área de estudio se encuentra ubicada en la provincia de Jaén, departamento de Cajamarca (ver figura 1).

Figura 1

Ubicación geográfica del área de estudio



Nota: La figura muestra la delimitación geográfica del área de estudio en el sector La Colina, distrito y provincia de Jaén, departamento de Cajamarca, así como su ubicación respecto a la carretera Jaén–San Ignacio.

2.3 Población Muestra y Muestreo

2.3.1 Población

La población es la totalidad de elementos u observaciones que son motivo de interés para la investigación (Montgomery & Runger, 2013). Esta debe estar claramente delimitada por sus características de contenido, ubicación y tiempo (Hernández et al., 2014).

En el estudio, la población está constituida por los elementos de registro de información correspondientes a un área de 23,99 hectáreas del sector La Colina, ubicado en el distrito y provincia de Jaén, departamento de Cajamarca, durante el año 2025. Estos elementos comprenden información física, geográfica, estructural, social, económica y ambiental, utilizada para identificar el nivel de peligro y analizar el nivel de vulnerabilidad y,

posteriormente, establecer el nivel de riesgo por deslizamiento mediante el uso de algoritmos ensamblados de aprendizaje automático.

2.3.2 Muestra

La muestra se determina como un subgrupo de elementos que pertenecen a un grupo definido en sus características al que llamamos población (Hernández et al., 2014).

La muestra estuvo constituida por los elementos de registro de información correspondientes a un área de 8.0786 hectáreas, ubicada en la margen izquierda del sector La Colina, entre la calle La Molina y la carretera Jaén–San Ignacio. Esta superficie representa aproximadamente el 33.67 % de las 23.99 hectáreas consideradas como población territorial del estudio.

Dentro del área delimitada se evaluaron 123 viviendas. La selección de este ámbito respondió a que concentra viviendas asentadas sobre la ladera y condiciones físicas asociadas al fenómeno estudiado, tales como pendientes, cortes de talud, procesos erosivos y exposición de las edificaciones. Por ello, el tamaño de la muestra se definió en función de la delimitación territorial del área de interés y de las unidades de análisis existentes dentro de ella, y no mediante un cálculo muestral probabilístico.

2.3.3 Muestreo

Se empleó un muestreo no probabilístico por conveniencia. El área de estudio se seleccionó por su accesibilidad y, sobre todo, por su relación directa con el problema investigado, ya que comprende viviendas ubicadas en ladera y presenta condiciones físicas vinculadas al peligro por deslizamiento, identificadas durante el reconocimiento preliminar y la revisión de antecedentes técnicos.

Al tratarse de un muestreo no probabilístico, no se buscó que la muestra fuera estadísticamente representativa de todo el sector La Colina. En cambio, se consideró que tenía una representatividad técnico-territorial, debido a que abarcó el ámbito donde se concentran

las condiciones de exposición y los elementos necesarios para identificar el peligro, analizar la vulnerabilidad y establecer el riesgo por deslizamiento.

2.4 Variables de estudio

2.4.1 Variables independientes

- Peligro
- Vulnerabilidad

2.4.2 Variable dependiente

- Riesgo

2.5 Tipo, nivel y enfoque de investigación

La investigación fue de tipo aplicada, este tipo de investigaciones se orientan a generar conocimiento con un objetivo práctico específico OECD (2015). En este caso, se empleó la metodología del CENEPRED y algoritmos ensamblados de clasificación supervisada para estimar el nivel de riesgo por deslizamiento en el sector La Colina, con la finalidad de clasificar las viviendas según su nivel de riesgo y aportar información técnica para la toma de decisiones en prevención y reducción del riesgo.

El nivel de la investigación fue descriptivo, ya que este tipo de estudios analizan y muestran cómo se presenta un fenómeno dentro de un contexto específico (Hernández, 2014). En ese sentido, la investigación describió las variables de peligro y vulnerabilidad que influyen en el riesgo por deslizamiento, permitiendo representar el nivel de riesgo de las viviendas del sector analizado.

Asimismo, el enfoque fue cuantitativo, porque las variables fueron ponderadas, codificadas y procesadas mediante procedimientos estadísticos y computacionales (Creswell, 2014). Aunque los parámetros de peligro y vulnerabilidad fueron en su mayoría categóricos,

estos se organizaron en una base de datos para su análisis mediante CENEPRED y algoritmos de clasificación supervisada.

2.6 Diseño de investigación

El diseño de la investigación fue no experimental y transeccional. Fue no experimental porque no se manipularon deliberadamente las variables, sino que se analizaron tal como se presentaron en el sector La Colina. Además, fue transeccional porque la recolección y análisis de datos se realizó en un periodo específico, sin evaluar la evolución temporal del fenómeno (Hernández, 2014).

La investigación no consideró grupos de control ni condiciones experimentales. El proceso general consistió en recolectar datos de peligro y vulnerabilidad, estimar el riesgo mediante la metodología del CENEPRED, organizar y codificar la base de datos, aplicar algoritmos ensamblados de clasificación supervisada, evaluar su desempeño y clasificar finalmente el nivel de riesgo por deslizamiento en las viviendas del sector La Colina.

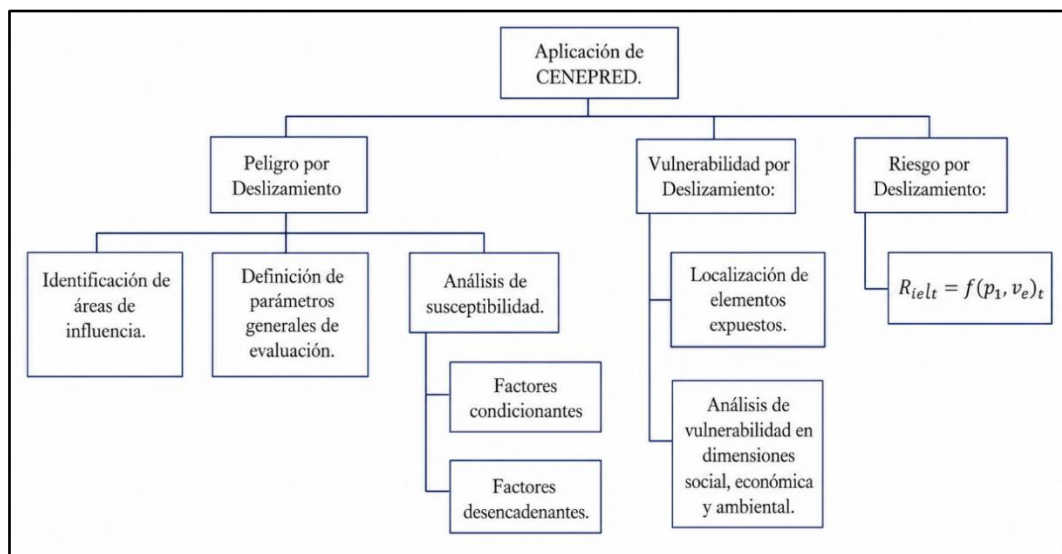
2.7 Métodos

Metodología CENEPRED

Se tomó como base el proceso metodológico establecido en el Manual para la evaluación de riesgos originados por fenómenos naturales del CENEPRED (2014). Para ello, se identificó el nivel de peligro por deslizamiento mediante la identificación de parámetros de evaluación, factores condicionantes y factores desencadenantes. Asimismo, se analizó el nivel de vulnerabilidad de la población expuesta considerando las dimensiones social, económica y ambiental, con la finalidad de integrar ambos componentes y estimar el nivel de riesgo en el sector de estudio.

Figura 2

Metodología CENEPRED



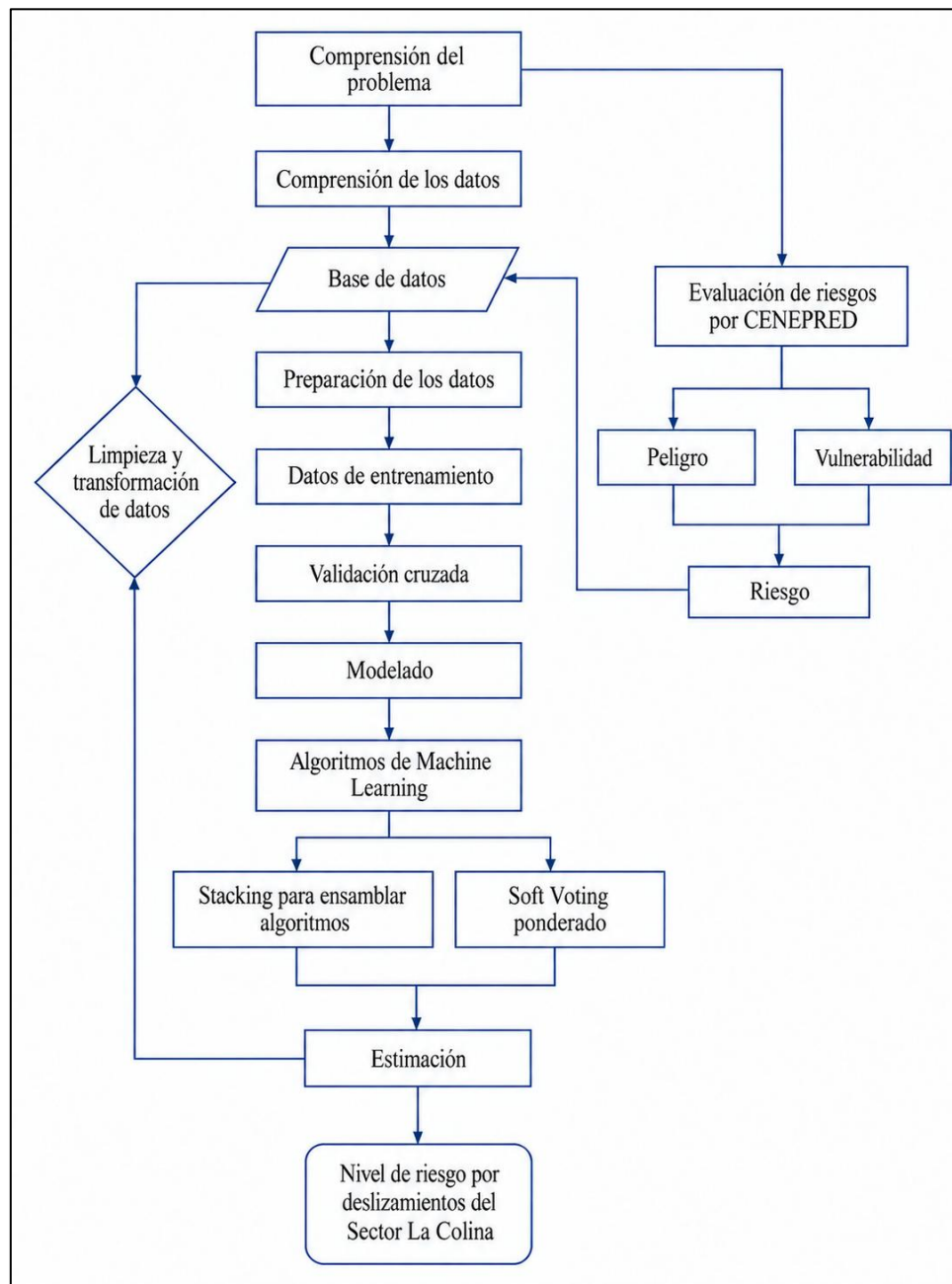
Nota: La figura presenta el procedimiento metodológico del CENEPRED 2014 para la evaluación del riesgo por deslizamiento.

Metodología CRISP-DM

Se seleccionó la metodología de Proceso estándar de Minería de Datos para cualquier industria (CRISP-DM), ampliamente reconocida a nivel mundial por su capacidad y adaptabilidad para estructurar y diseñar proyectos de análisis de datos con eficiencia y efectividad. Esta metodología abarca seis fases principales que son iterativas y flexibles: Comprensión del Problema, Comprensión de los Datos, Preparación de los Datos, Modelado, Evaluación y Despliegue. Es relevante aclarar que los alcances de esta investigación llegan hasta la evaluación.

Figura 3

Metodología general para la estimación del nivel de riesgo



Nota: La figura presenta la integración de la metodología CENEPRED 2014 con algoritmos de machine learning para estimar el nivel de riesgo por deslizamiento.

2.8 Materiales, técnicas e instrumentos

2.8.1 Materiales

- Plano catastral de Jaén 2025

- Mapa geomorfológico de la región Cajamarca
- Mapa geológico del cuadrángulo de Jaén
- Registros y capas topográficas, geológicas y geomorfológicas de GEOCATMIN e INGEMMET
- Registros y publicaciones de SENAMHI
- Elevaciones satelitales de radar de apertura sintética (SAR), correspondientes a la misión Sentinel 1, descargadas desde el Alaska Satellite Facility Distributed Active Archive Center (ASF DAAC) de la NASA.
- Imágenes satelitales ópticas del software SAS.Planet, servicio Bing Maps – Satellite.

2.8.2 *Técnicas*

- Revisión documental y cartográfica
- Encuesta
- Procesamiento geoespacial
- Observación directa

2.8.3 *Instrumentos*

Cuestionario de vulnerabilidad

Ficha de recolección de datos

2.9 *Proceso metodológico*

2.9.1 *Reconocimiento del área de estudio*

Se realizó el recorrido del entorno del sector estudiado, abarcando por la parte externa de la ladera se encuentra la calle La Molina en la parte superior y la vía nacional en el tramo de la carretera Jaén – San Ignacio en la parte inferior, así como también, en la parte interior de la ladera, se realizó un recorrido en el pasaje Ciruela, 18 de abril, Los Cocos y El Limón. Asimismo, se realizaron charlas con la comunidad del sector La Colina para sensibilizarla sobre el potencial de deslizamientos en la zona. Se obtuvo el apoyo de la comunidad para la

investigación, facilitando el acceso a la información y a los diferentes puntos dentro del área de estudio. Estas actividades también permitieron recabar las perspectivas de los residentes sobre las condiciones del entorno, lo que contribuyó a un análisis más completo de las variables de riesgo y vulnerabilidad consideradas en el estudio.

2.9.2 *Identificación del nivel de peligro*

2.9.2.1 Análisis de elementos expuestos

a) Situación actual de la ladera en el sector

La ladera muestra una pendiente moderada a fuerte con suelos aparentemente del tipo granular suelto, lo que puede hacer susceptible a erosión. También se observó cortes verticales en la base a lo largo de la ladera sin obras de contención formales, además, se encontraron señales de desgaste en el talud y pequeños desprendimientos, lo que puede generar una pérdida de estabilidad local.

Figura 4

Vista de la ladera del sector La colina



Nota: Registro fotográfico de la ladera del sector La Colina, realizado el 15 de diciembre de 2025 a las 12:00 horas.

b) Situación actual de taludes

Se observaron cortes de talud con geometría empinada en tramos largos. Se encontró zonas degradadas aparentemente por causa de precipitación fluvial, así mismo, se observó material suelto en el base debido a caídas previas y no se observaron estructuras formales de prevención. En conclusión, los cortes de talud en la zona se encuentran en condición vulnerable por inestabilidad y lluvias.

Figura 5

Vista de la ladera del sector La colina



Nota: Registro fotográfico de un talud del sector La Colina, realizado el 15 de diciembre de 2025 a las 12:30 horas

c) Situación actual de pasajes

Los pasajes presentan una superficie de tierra sin pavimentar con presencia de material suelto que genera polvo y erosión. Así mismo, se observó pasajes reducidos y sin delimitación adecuada de hasta 2.5 metros de sección y con escaleras empinadas en mal estado dificultando el libre tránsito peatonal. En los bordes existen muros de contención contruidos de una manera informal (piedra y sacos) lo que evidencia intervenciones precarias por parte la comunidad vecinal. Asimismo, no se identificó infraestructura de drenaje pluvial adecuada en la zona; además, la presencia de escombros y maleza en los pasajes sugiere que han sido arrastrados por la lluvia, evidenciando la existencia de un drenaje informal.

Figura 6

Pasajes con superficie de tierra en la ladera del sector La Colina



Nota: Registro fotográfico del pasaje Ciruela, realizado el 15 de diciembre de 2025 a las 13:00 horas.

d) Situación actual de las viviendas

Las viviendas presentan condiciones de construcción básicas, en la que predomina materiales como el ladrillo sin revestimiento y adobe. Se observó también que algunas estructuras contienen adaptaciones informales de estabilidad o contención. El entorno de la vivienda muestra la falta repavimentación y drenaje de desagüe expuestas en taludes.

Figura 7

Condición actual de las viviendas en el sector La Colina



Nota: Registro fotográfico de viviendas del sector La Colina, realizado el 16 de diciembre de 2025 a las 11:00 horas.

e) Situación actual de la carretera Jaén – San Ignacio

La vía nacional, carretera Jaén San – San Ignacio presenta una superficie de asfalto en condiciones aceptables. Se observó también material suelto en los drenajes superficiales (cunetas) debido a la falta de mantenimiento en los bordes de la ladera del sector La Colina lo que puede afectar la evacuación adecuada de aguas pluviales.

Figura 8

Vía nacional, Carretera Jaén – San Ignacio al margen izquierdo del sector La Colina



Nota: Registro fotográfico del entorno de la carretera Jaén–San Ignacio, realizado el 16 de diciembre de 2025 a las 12:00 horas.

2.9.2.2 Parámetro de evaluación del fenómeno

Para identificar el parámetro de evaluación del peligro, se llevó a cabo una observación detallada del área de estudio, donde se identificó erosión en surcos y cárcavas en cortes de talud, material granular y detritos a lo largo de los pasajes en pendiente, todo generado por escorrentía superficial.

Figura 9

Erosión en surcos y cárcavas en corte de talud

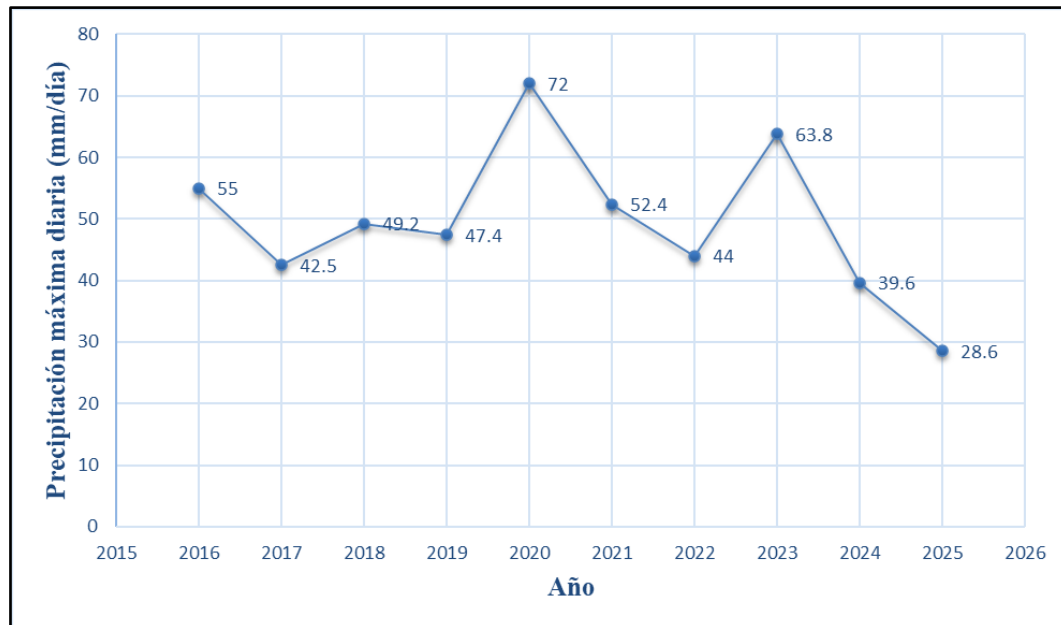


Nota: La figura, registrada el 19 de diciembre de 2025 a las 11:00 horas, se muestra erosión en surcos y cárcavas en un corte de talud del sector La Colina, asociada a la escorrentía superficial.

Basándose en este análisis, se consideró el periodo de retorno de lluvias intensas como el principal parámetro de evaluación. Para la ponderación se consideró la información hidrometeorológica de la estación más cercana de la ciudad de Jaén donde se descargaron datos históricos desde el año 2016, hasta el año 2025.

Figura 10

Precipitaciones máximas por día entre los años 2016 – 2025



Nota: La figura muestra la variación de las precipitaciones máximas diarias registradas en la estación Jaén entre 2016 y 2025, utilizadas para el análisis del periodo de retorno de lluvias intensas.

A partir de esta serie se determinaron los valores asociados a diferentes periodos de retorno mediante la distribución probabilística de Gumbel. La selección de esta distribución se realizó debido a que el análisis estuvo conformado por una serie de máximos anuales de precipitación diaria, por lo que resulta adecuada para el análisis probabilístico de valores extremos hidrológicos y permite relacionar la magnitud de una precipitación extrema con su probabilidad de ocurrencia y periodo de retorno Gumbel (1958). De esta manera, se estimaron las precipitaciones máximas diarias correspondientes a los periodos de retorno considerados en la evaluación del peligro por deslizamiento.

Primero se calculó el promedio de los datos, obteniéndose un valor de $\bar{x} = 49.45$ mm/día. Luego se calculó la desviación estándar, obteniéndose $s = 12.33$ mm/día. De acuerdo

con Gumbel (1958) con estos valores se determinaron los parámetros de escala α y el parámetro de posición u , mediante las Ecuaciones 1 y 2, respectivamente.

Parámetros de Gumbel:

$$\alpha = \frac{s\sqrt{6}}{\pi} \quad (1)$$

$$u = \bar{x} - 0.5772\alpha \quad (2)$$

Reemplazando:

$$\alpha = 9.61$$

$$u = 43.90$$

Una vez aplicada la fórmula de Gumbel, se obtuvieron precipitaciones máximas diarias para los periodos de retorno de 5, 15, 25 y 35 años mediante la Ecuación 3 (Gumbel, 1958).

$$P_T = u - \alpha \ln \left[-\ln \left(1 - \frac{1}{T} \right) \right] \quad (3)$$

Se obtuvieron los siguientes resultados:

Tabla 1

Rangos de precipitación

Periodo de retorno	Precipitación máxima diaria
5 años	58.31 mm/día
15 años	69.59 mm/día
25 años	74.64 mm/día
35 años	77.93 mm/día

Fuente: Elaboración propia

Se debe tener en cuenta que para el análisis se trabajó con una serie de 10 años de registros, comprendida entre 2016 y 2025. Los periodos de retorno de 5, 15, 25 y 35 años fueron considerados debido a que corresponden a los límites de los intervalos establecidos para el parámetro de evaluación empleado en la metodología CENEPRED: menor de 5 años, de 5 a 15 años, de 15 a 25 años, de 25 a 35 años y mayor de 35 años. En ese sentido, estos periodos no fueron seleccionados arbitrariamente, sino que permitieron relacionar las precipitaciones estimadas mediante Gumbel con las categorías utilizadas posteriormente en la evaluación del

peligro. Debido a que algunos periodos de retorno superan la longitud de la serie disponible, principalmente los correspondientes a 15, 25 y 35 años, sus valores representan extrapolaciones probabilísticas y deben interpretarse como aproximaciones para fines de evaluación del peligro por deslizamiento.

El evento máximo identificado en la serie fue de 72 mm/día, valor que supera el umbral de 41.5 mm/día establecido por SENAMHI para la categoría de evento extremadamente lluvioso en la estación Jaén. A partir de la distribución de Gumbel ajustada a la serie analizada, este evento presentó un periodo de retorno aproximado de 19.10 años, por lo que se ubicó dentro del intervalo de 15 a 25 años considerado para el parámetro de evaluación. La ponderación correspondiente al periodo de retorno se determinó posteriormente de acuerdo con la metodología CENEPRED mediante el Proceso de Análisis Jerárquico.

Tabla 2

Peso ponderado del parámetro de evaluación – Periodo de retorno

Parámetro			Peso Ponderado	
Descriptores	PP-01	muy alta (<5 años)	PS-1	0.499
	PP-02	alta (5 - 15 años)	PS-2	0.277
	PP-03	media (15 - 25 años)	PS-3	0.118
	PP-04	baja (25 - 35 años)	PS-4	0.071
	PP-05	muy baja (>35 años)	PS-5	0.035

Fuente: Adaptado de CENEPRED (2014).

2.9.2.3 Análisis de la susceptibilidad

2.9.2.3.1 Factores condicionantes

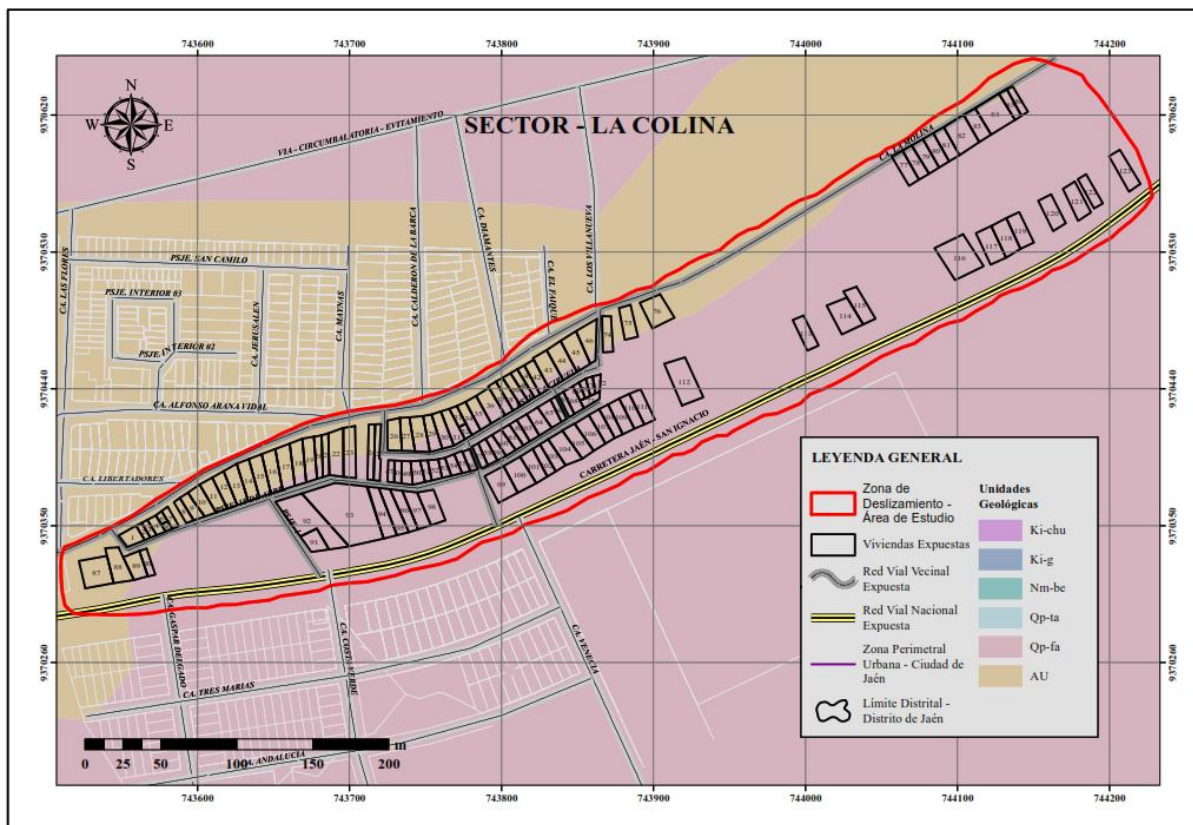
- Geología

La información de la geología en la zona de estudio fue obtenida mediante el mapa geológico de cuadrángulo de Jaén proveniente del Instituto Geológico, Minero y Metalúrgico (INGEMET) del departamento de Cajamarca en escala de 1:100 000 en su última actualización en 1996. Dicha información fue obtenida del repositorio institucional del INGEMET y las

capas en formato shapes del geoservidor Geocatmin con el fin de obtener una mayor cantidad de datos de las unidades estratigráficas en el software QGIS.

Figura 11

Mapa de geología en el sector La Colina procesado en software QGIS



Nota: La figura muestra la distribución de las unidades geológicas en el sector La Colina, obtenidas del mapa geológico del cuadrángulo de Jaén y procesadas en QGIS.

La ponderación de las unidades geológicas se determinó de acuerdo a la metodología CENEPRED basado en el Proceso de Análisis Jerárquico.

Tabla 3*Peso ponderado del factor condicionante – Geología*

Parámetro			Peso Ponderado	
Descriptores	GE-01	Qp-fa (Depósitos Fluvioaluviales)	PS-1	0.426
	GE-02	Qp-ta (Fm. Tamborapa)	PS-2	0.301
	GE-03	Nm-be (Fm. Bellavista)	PS-3	0.146
	GE-04	Ki-chu (Fm. Chulec)	PS-4	0.081
	GE-05	Ki-g (Gpo. Goyllarisquizga)	PS-5	0.046

Fuente: Adaptado de CENEPRED (2014).

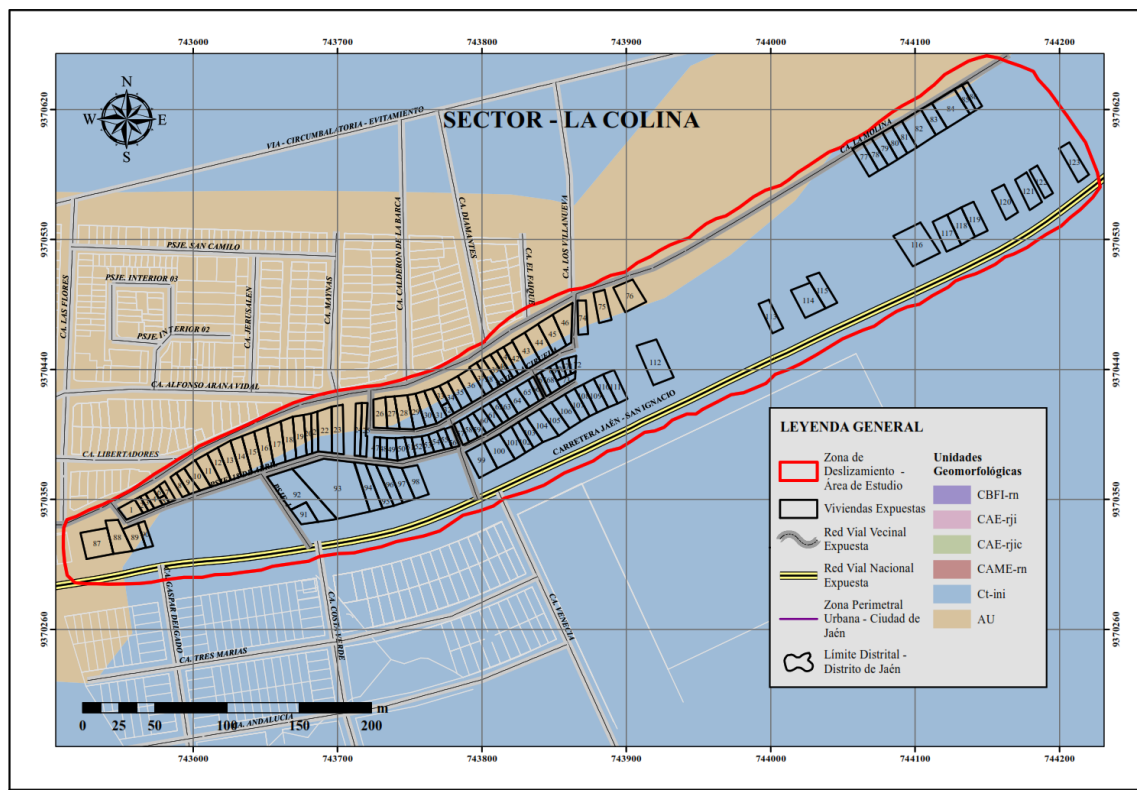
Nota: el código AU representa al casco urbano la cual está dentro de la unidad geológicas de Qp-fa (Depósitos Fluvioaluviales).

- **Unidades geomorfológicas**

La información de la geomorfología en la zona de estudio fue obtenida mediante el mapa geomorfológico de la zonificación económica y ecológica del proyecto Ordenamiento Territorial de la Región Cajamarca en escala de 1:250 000 en su última actualización en 2012. Dicha información y las capas en formato shapes fueron obtenidas en Mapoteca Virtual del Ordenamiento Territorial del Gobierno de Cajamarca con el fin de obtener una mayor cantidad de datos de las unidades estratigráficas en el software QGIS.

Figura 12

Mapa de geomorfológico en el sector La Colina procesado en software QGIS



Nota: La figura muestra la distribución de las unidades geomorfológicas del sector La Colina, obtenidas de la zonificación ecológica y económica de la Región Cajamarca y procesadas en QGIS.

La ponderación de las unidades geomorfológicas se determinó de acuerdo a la metodología CENEPRED basado en el Proceso de Análisis Jerárquico.

Tabla 4*Peso ponderado del factor condicionante – Geomorfología*

		Parámetro	Peso Ponderado	
Descriptores	GM-01	CAE-rjic (Colina alta empinada en rocas del jurasico inferior y del cuaternario)	PS-1	0.438
	GM-02	CAE-rji (Colina alta empinada en rocas del jurasico inferior)	PS-2	0.280
	GM-03	CBFI-rn (Colina baja fuertemente inclinada en rocas el neógeno)	PS-3	0.123
	GM-04	CAME-rn (Colina alta moderadamente empinada en rocas del neógeno)	PS-4	0.111
	GM-05	Ct-ini (Complejo de terrazas inundable y no inundable)	PS-5	0.048

Fuente: Adaptado de CENEPRED (2014).

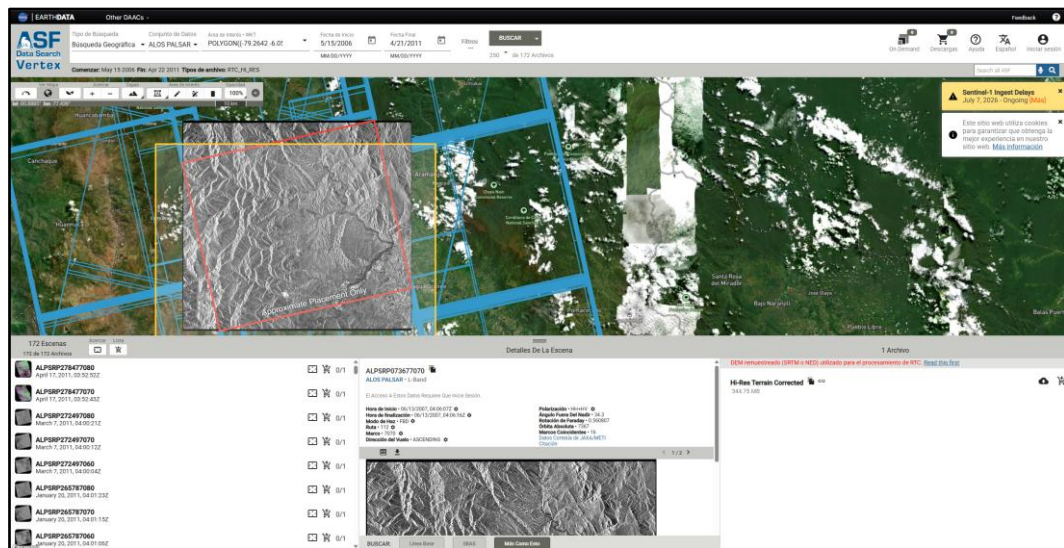
Nota: el código AU representa al casco urbano la cual está dentro de la unidad geomorfológica de Ct-ini (Complejo de terrazas inundable y no inundable).

- **Pendiente**

Para la elaboración de las pendientes del área de estudio, se descargó información altimétrica asociada al producto ALOS PALSAR, obtenido desde la plataforma ASF Data Search Vertex. La escena seleccionada corresponde al producto Hi-Res Terrain Corrected, con identificador ALPSRP073677070, adquirido el 13 de junio de 2007, en modo de órbita ascendente y polarización HH+HV.

Figura 13

Portal de descargas de elevaciones digitales ALOS PALSAR

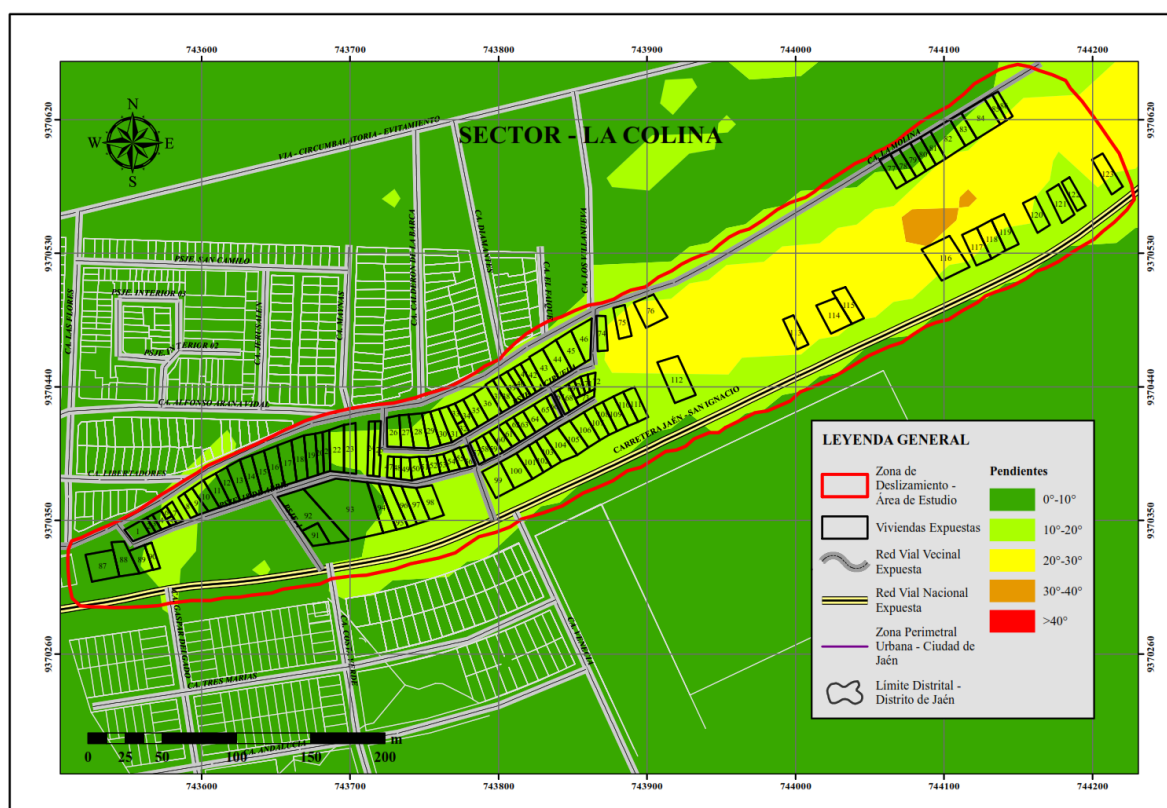


Nota: La figura muestra el portal ASF Data Search Vertex utilizado para descargar la información altimétrica del producto ALOS PALSAR empleada en la elaboración del mapa de pendientes.

Posteriormente, la información fue procesada en el software QGIS para generar el mapa de pendientes del área de estudio. La unidad cartográfica de este factor se determinó en rangos porcentuales, conforme a lo indicado en el Reglamento de Clasificación de Tierras por su Capacidad de Uso Mayor aprobado mediante D.S. N.º 017-2009-AG, así como en los criterios utilizados en la Zonificación Ecológica y Económica departamental.

Figura 14

Mapa de pendiente en el sector La Colina procesado en software QGIS



Nota: La figura muestra la distribución de los rangos de pendiente en el sector La Colina, obtenidos a partir de información altimétrica y procesados en QGIS.

La ponderación del factor pendiente se determinó de acuerdo a la metodología CENEPRED basado en el Proceso de Análisis Jerárquico.

Tabla 5

Peso ponderado del factor condicionante – Pendiente

Parámetro			Peso Ponderado	
Descriptores	PE-01	Muy empinada (>40°)	PS-1	0.503
	PE-02	Empinada (30°-40°)	PS-2	0.260
	PE-03	Moderadamente empinada (20°-30°)	PS-3	0.134
	PE-04	Ligeramente inclinada (10°-20°)	PS-4	0.068
	PE-05	Plano o llano (0°-10°)	PS-5	0.035

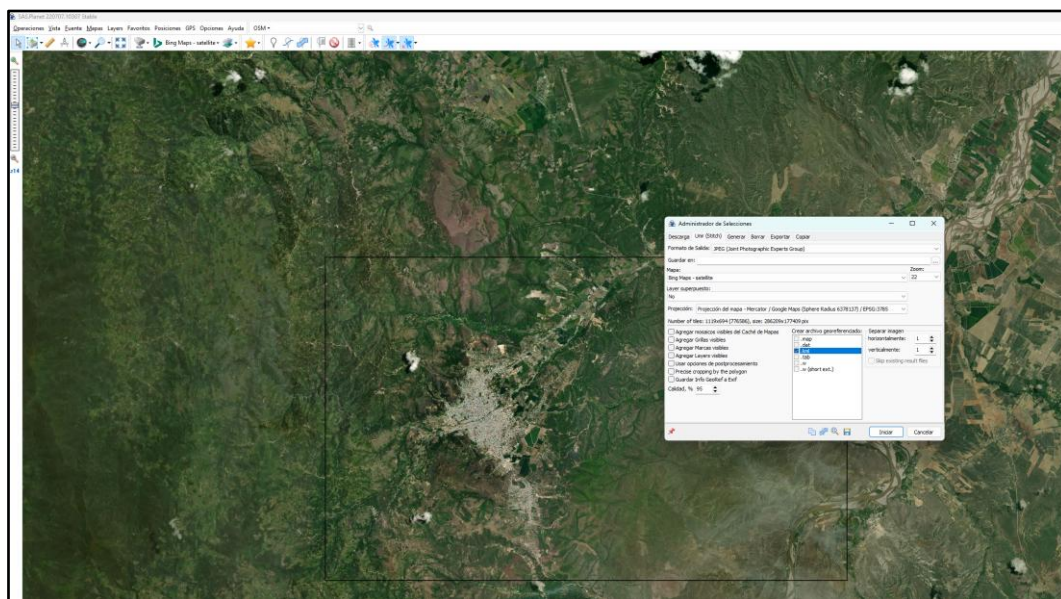
Fuente: Adaptado de CENEPRED (2014).

- Uso actual del suelo

La información del uso actual de suelo en la zona de estudio fue obtenida mediante el mapa de uso actual de la tierra de la zonificación Económica y Ecológica del proyecto Ordenamiento Territorial de la Región Cajamarca en escala de 1:250 000 en su última actualización en 2010. Debido a la escala regional de la información, para la elaboración del mapa en la zona de estudio se complementó el análisis con el apoyo visual de imágenes satelitales obtenidas mediante el software SAS.Planet Release 220707, empleando como mapa base el servicio Bing Maps – satellite.

Figura 15

Portal de descargas de Bing Maps – satellite

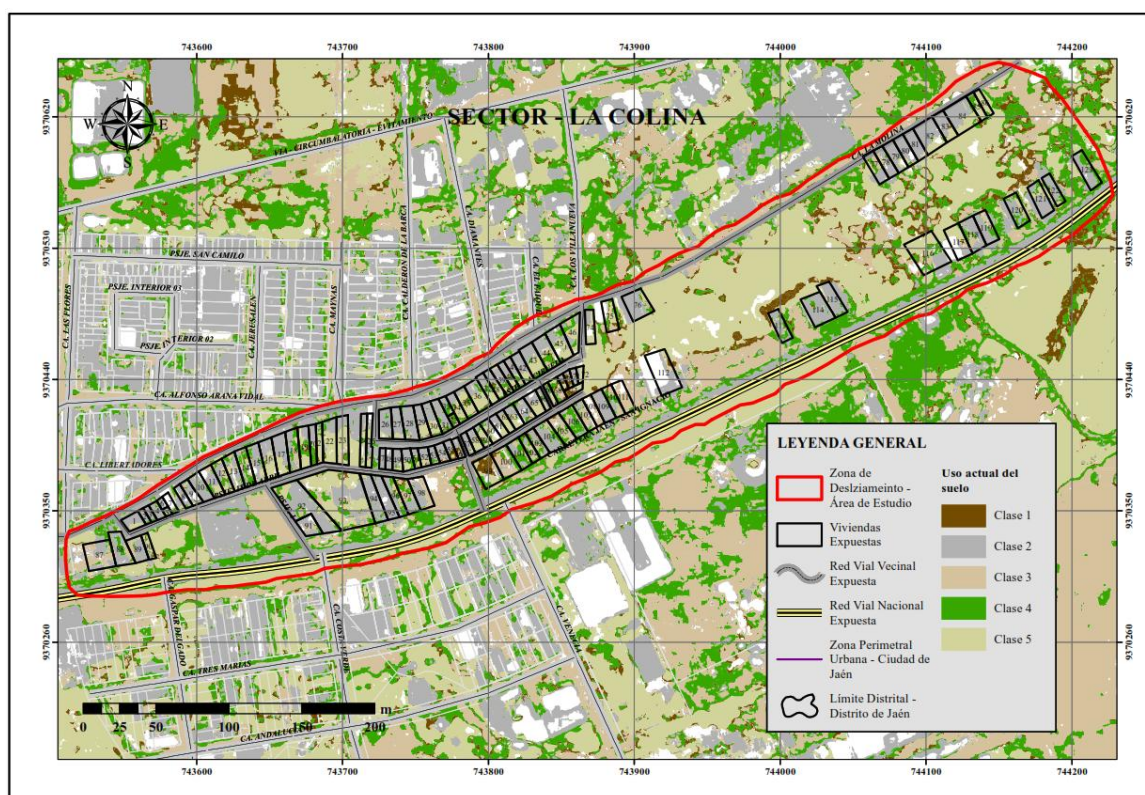


Nota: La figura muestra el portal SAS.Planet utilizado para obtener imágenes satelitales de Bing Maps – Satellite, empleadas como apoyo en la elaboración del mapa de uso actual del suelo.

La imagen fue descargada y procesada en el software QGIS, donde se realizó la delimitación y clasificación supervisada de las principales coberturas presentes en el área de estudio.

Figura 16

Mapa de Uso actual del suelo en el sector La Colina procesado en software QGIS



Fuente: La figura muestra la clasificación del uso actual del suelo en el sector La Colina, obtenida mediante imágenes satelitales y procesada en QGIS.

La ponderación del factor Uso actual del suelo se determinó de acuerdo a la metodología CENEPRED basado en el Proceso de Análisis Jerárquico.

Tabla 6

Peso ponderado factor condicionante – Uso Actual del Suelo

Parámetro			Peso Ponderado	
Descriptores	UT-01	Clase 1 (Áreas erosionadas, cárcavas, surcos y suelo desnudo degradado)	PS-1	0.501
	UT-02	Clase 2 (Zona Urbanizada en ladera)	PS-2	0.278
	UT-03	Clase 3 (Áreas abiertas sin vegetación)	PS-3	0.119
	UT-04	Clase 4 (Áreas agrícolas heterogéneas)	PS-4	0.064
	UT-05	Clase 5 (Vegetación arbustiva)	PS-5	0.037

Fuente: Adaptado de CENEPRED (2014).

2.9.2.3.2 Factor desencadenante

El factor de precipitación fue seleccionado como el desencadenante debido a la evidencia de surcos y cárcavas y material granular suelto en cortes de Talud y pasajes en pendiente del sector, evidencias asociadas a las intensas lluvias en la zona de estudio

Para su caracterización, se emplearon los umbrales de lluvias extremas establecidas por el SENAMHI en su informe técnico “Umbrales y precipitaciones absolutas” basadas en el análisis percentil de series históricas de precipitación diaria, utilizando la metodología sustentada por la nota técnica N°001-SENAMHI-DGM-2014.

Para el factor de precipitación se analizaron registros de la Estación de Jaén desde el año 2016 hasta el año 2025 en la que se determinó una precipitación máxima diaria de 72 mm/día. Comparando la precipitación máxima acumulada con los umbrales del SENAMHI (ver tabla 7), el evento evaluado se clasificó como extremadamente lluvioso.

Tabla 7

Rangos de precipitación

Umbrales de Precipitación	Caracterización de lluvias extremas	Umbrales calculados para la Estación: Jaén
RR/día>99p	Extremadamente lluvioso	RR>41,5 mm
95p<RR/día≤99p	Muy lluvioso	21,3 mm<RR≤41,5 mm
90p<RR/día≤95p	Lluvioso	14,8 mm<RR≤21,3 mm
75p<RR/día≤90p	Moderadamente lluvioso	7,2 mm<RR≤14,8 mm

Fuente: Adaptado de SENAMHI 2014, Umbrales y precipitaciones absolutas, tabla 6ª, estación Jaén.

La ponderación del factor Uso actual del suelo se determinó de acuerdo a la metodología CENEPRED basado en el Proceso de Análisis Jerárquico.

Tabla 8*Peso ponderado del factor desencadenante – Precipitación*

Parámetro			Peso Ponderado	
Descriptores	PP-01	Extremadamente lluvioso	PS-1	0.495
	PP-02	Muy lluvioso	PS-2	0.285
	PP-03	Lluvioso	PS-3	0.117
	PP-04	Moderadamente lluvioso	PS-4	0.069
	PP-05	Ligeramente lluvioso	PS-5	0.035

*Fuente: Adaptado de CENEPRED (2014).***2.9.2.4 Estratificación del nivel de peligro****Tabla 9***Estratificación del nivel de peligro*

NIVELES DE PELIGRO	DESCRIPCIÓN	RANGOS
Peligro muy alto	En la zona estudiada predominan áreas correspondientes a la Formación Bellavista. Estas se encuentran principalmente en colinas bajas fuertemente inclinadas, con pendientes entre 30° y 40°. También se caracterizan por la presencia de áreas abiertas con poca o ninguna cobertura vegetal. Las precipitaciones son lluviosas y los eventos de lluvia presentan periodos de retorno entre 15 y 25 años. Estas condiciones favorecen la erosión del suelo y pueden generar procesos de inestabilidad en las laderas, especialmente durante las temporadas de lluvias.	$0.287 \leq P \leq 0.365$
Peligro alto	En la zona estudiada predominan sectores desarrollados sobre la Formación Tamborapa, asociados a unidades geomorfológicas de colinas altas empinadas en rocas del Jurásico inferior, con pendientes empinadas entre 20° y 30°. Se caracterizan por la presencia de zonas urbanizadas en ladera, sometidas a condiciones muy lluviosas y a eventos de precipitación con periodos de retorno entre 5 y 15 años. Estas características generan condiciones favorables para la ocurrencia de movimientos en masa, especialmente durante temporadas de intensas precipitaciones.	$0.220 \leq P < 0.287$
Peligro medio	En la zona estudiada predominan áreas desarrolladas sobre la Formación Bellavista, ubicadas en unidades geomorfológicas de colinas bajas fuertemente inclinadas en rocas del Neógeno, con pendientes moderadamente empinadas entre 20° y 30°. Presentan principalmente áreas abiertas sin vegetación, sometidas a condiciones lluviosas y asociadas a eventos con periodos de retorno	$0.194 \leq P < 0.220$

NIVELES DE PELIGRO	DESCRIPCIÓN	RANGOS
	entre 15 y 25 años. Aunque las condiciones de inestabilidad son moderadas, la ocurrencia de precipitaciones persistentes puede generar procesos de erosión y movimientos de ladera localizados.	
Peligro bajo	En la zona estudiada predominan sectores desarrollados sobre la Formación Chulec y el Grupo Goyllarisquizga, asociados a unidades geomorfológicas de colinas moderadamente empinadas en rocas del Neógeno y complejos de terrazas inundables y no inundables, con pendientes ligeramente inclinadas o planas (0°–20°). Presentan principalmente áreas agrícolas heterogéneas y vegetación arbustiva, sometidas a condiciones moderadamente lluviosas o ligeramente lluviosas, asociadas a eventos con periodos de retorno superiores a 25 años. Estas características proporcionan una mayor estabilidad relativa del terreno, reduciendo la probabilidad de ocurrencia de deslizamientos.	$0.177 \leq P < 0.194$

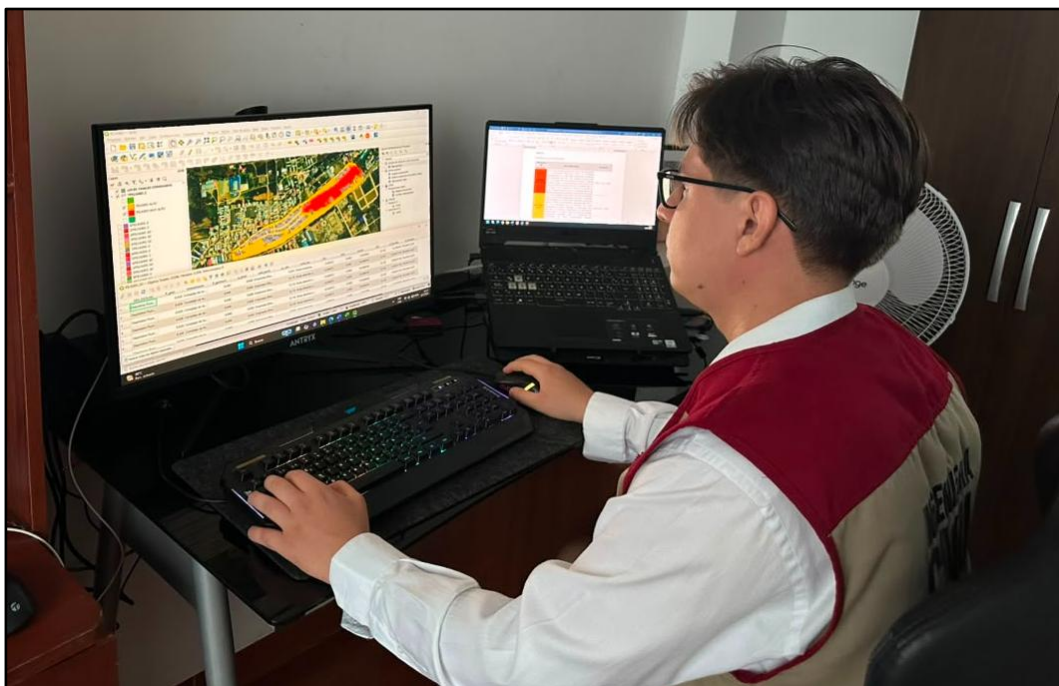
Fuente: Adaptado de CENEPRED

2.9.2.5 Procesamiento de la información de Peligro

Con toda la información analizada se procesó en el software QGIS en concordancia con las ponderaciones calculadas para identificar los niveles de peligro de la zona de estudio.

Figura 17

Procesamiento en el software QGIS de los niveles de Peligro en la zona de Estudio



Nota: La figura muestra el procesamiento en QGIS de la información ponderada para identificar los niveles de peligro por deslizamiento en el sector La Colina.

2.9.3 Análisis del nivel de vulnerabilidad

Se analizó la vulnerabilidad en sus dimensiones, social, económica y ambiental; y estas a su vez, en indicadores de exposición, fragilidad y resiliencia. A continuación, se detalla los pesos obtenidos tras la aplicación de la metodología del CENEPRED basado en el análisis de proceso jerárquico.

2.9.3.1 Factores de la vulnerabilidad

- Vulnerabilidad Social

Para la vulnerabilidad social se buscó identificar qué cantidad de población está expuesta, se analizó su fragilidad ante deslizamientos y su posible respuesta ante esta eventualidad, es decir la resiliencia. Se muestran las tablas de cada parámetro seleccionado con sus ponderaciones calculadas:

Tabla 10*Peso ponderado del parámetro cantidad de habitantes por vivienda*

Parámetro	Código	Cantidad de habitantes por vivienda	Peso Ponderado
Descriptores	CH-01	6 a más personas	0.504
	CH-02	De 4 a 5 personas	0.260
	CH-03	3 personas	0.132
	CH-04	2 personas	0.062
	CH-05	1 persona	0.042

Fuente: Adaptado de CENEPRED (2014).

Las viviendas con seis o más habitantes recibieron el mayor peso (0.504), con lo que se asignó mayor importancia a la concentración de personas que podrían requerir evacuación y asistencia.

Tabla 11*Descriptores del parámetro grupo etario más vulnerable en la vivienda*

Parámetro	Código	Grupo etario más vulnerable en la vivienda	Peso Ponderado
Descriptores	GE-01	Menor de 6 años y mayor de 60 años	0.489
	GE-02	De 51 a 60 años	0.288
	GE-03	De 7 a 12 años	0.125
	GE-04	De 13 a 19 años	0.064
	GE-05	De 20 a 50 años	0.033

Fuente: Adaptado de CENEPRED (2014).

La ponderación priorizó a los menores de seis años y mayores de sesenta, se consideró sus posibles necesidades de apoyo durante la evacuación y atención de una emergencia.

Tabla 12*Peso ponderado del parámetro personas con dificultad permanente*

Parámetro	Código	Personas con dificultad permanente	Peso Ponderado
Descriptores	D-01	Motriz: para usar brazos y/o piernas	0.499
	D-02	Cognitiva o mental	0.257
	D-03	Visual: Para ver, incluso con lentes.	0.144
	D-04	Para oír y/o hablar	0.065
	D-05	No tiene	0.035

Fuente: Adaptado de CENEPRED (2014).

La dificultad motriz concentró el mayor peso (0.499), seguida de la cognitiva o mental (0.257), se priorizó las condiciones que pueden requerir asistencia para desplazarse o seguir indicaciones de evacuación.

Tabla 13*Peso ponderado del parámetro abastecimiento de agua en la vivienda*

Parámetro	Código	Abastecimiento de agua en la vivienda	Peso Ponderado
Descriptores	A-01	No tiene	0.507
	A-02	Pozo, río o acequia	0.263
	A-03	Camión cisterna u otro	0.123
	A-04	Pilón	0.072
	A-05	Red de agua potable	0.035

Fuente: Adaptado de CENEPRED (2014).

La falta de abastecimiento de agua recibió un peso de 0.507, frente a 0.035 de la red de agua potable, se destacó esta carencia dentro de las necesidades básicas del hogar.

Tabla 14*Peso ponderado del parámetro disposición sanitaria de excretas en la vivienda*

Parámetro	Código	Disposición sanitaria de excretas en la vivienda	Peso Ponderado
Descriptores	EX-01	No tiene	0.509
	EX-02	Letrina	0.265
	EX-03	Pozo ciego/ negro	0.117
	EX-04	Conexión de desagüe improvisada, expuesta o en mal estado	0.073
	EX-05	Red de desagüe	0.036

Fuente: Adaptado de CENEPRED (2014).

La ausencia de disposición sanitaria de excretas presentó el mayor peso (0.509), mientras que la red de desagüe recibió el menor (0.036), se resaltó el saneamiento como parte de las condiciones de habitabilidad.

Tabla 15*Peso ponderado del parámetro fuente de energía eléctrica*

Parámetro	Código	Fuente de energía eléctrica	Peso Ponderado
Descriptores	AL-01	No tiene	0.444
	AL-02	Conexión improvisada desde otra vivienda.	0.262
	AL-03	Generador eléctrico	0.153
	AL-04	Panel solar	0.089
	AL-05	Red Pública	0.053

Fuente: Adaptado de CENEPRED (2014).

La falta de electricidad (0.444) y las conexiones improvisadas (0.262) recibieron los mayores pesos, esto debido a que la carencia del suministro puede dificultar la iluminación, comunicación, y evacuación durante una emergencia.

Tabla 16

Peso ponderado del parámetro nivel educativo más alto en la vivienda

Parámetro	Código	Nivel educativo más alto en la vivienda.	Peso Ponderado
Descriptores	ED-01	Ningún nivel y/o inicial	0.489
	ED-02	Primaria	0.288
	ED-03	Secundaria	0.125
	ED-04	Superior no universitaria	0.064
	ED-05	Superior universitaria/ Postgrado	0.033

Fuente: Adaptado de CENEPRED (2014).

La categoría sin nivel educativo o con nivel inicial recibió la mayor valoración (0.489). Con este criterio, se consideraron las posibles necesidades de apoyo para comprender la información preventiva

Tabla 17

Peso ponderado del parámetro capacitación de la población en temas de GRD

Parámetro	Código	Capacitación de la población en temas de GRD	Peso Ponderado
Descriptores	CP-01	No tiene	0.496
	CP-02	Hace 2 años o más	0.255
	CP-03	Hace 1 año	0.150
	CP-04	Hace 6 meses	0.064
	CP-05	Hace 3 meses	0.034

Fuente: Adaptado de CENEPRED (2014).

La ausencia de capacitación recibió el mayor peso (0.496), mientras que la capacitación realizada hacía tres meses obtuvo 0.034. Esta diferencia destacó la formación reciente como una condición favorable para la preparación.

Tabla 18

Peso ponderado del parámetro tipo de seguro de salud de la población

Parámetro	Código	Tipo de seguro de salud de la población	Peso Ponderado
Descriptores	TS-01	No tiene	0.499
	TS-02	SIS	0.251
	TS-03	ES SALUD	0.155
	TS-04	FF. AA y PNP	0.062
	TS-05	Privado	0.033

Fuente: Adaptado de CENEPRED (2014).

La ausencia de seguro presentó el mayor peso (0.499), por encima de todas las modalidades de aseguramiento. Con ello, se destacó la falta de cobertura de servicio médico como una condición desfavorable para afrontar una emergencia.

Tabla 19

Peso ponderado del parámetro percepción de la población acerca de los deslizamientos

Parámetro	Código	Percepción de la población acerca de los deslizamientos	Peso Ponderado
Descriptores	P-01	Fatalista	0.503
	P-02	Desidia	0.260
	P-03	Preocupado	0.134
	P-04	Positivo	0.068
	P-05	Optimista	0.035

Fuente: Adaptado de CENEPRED (2014).

El fatalismo (0.503) y la desidia (0.260) recibieron los mayores pesos. Estas posturas se consideraron menos favorables para la prevención que las percepciones clasificadas como positivas u optimistas.

- Vulnerabilidad económica

Para analizar la vulnerabilidad económica se identificó la cercanía de las viviendas a zonas de deslizamientos, las características de las viviendas y su protección ante deslizamientos.

Tabla 20

Peso ponderado del parámetro cercanía de la vivienda a la zona de deslizamientos

Parámetro	Código	Cercanía de la vivienda a la zona de deslizamientos	Peso Ponderado
Descriptores	C-01	Muy cercana: 0m a 40m	0.489
	C-02	Cercana: > 40m - 80m	0.288
	C-03	Medianamente cerca: > 80m - 150m	0.125
	C-04	Alejada: > 150m - 400m	0.064
	C-05	Muy alejada: > 400m	0.033

Fuente: Adaptado de CENEPRED (2014).

Las viviendas ubicadas hasta cuarenta metros de la zona de deslizamientos presentaron el mayor peso con 0.489, por lo que se priorizó la proximidad como criterio de exposición.

Tabla 21

Peso ponderado del parámetro antigüedad de la vivienda

Parámetro	Código	Antigüedad de la vivienda	Peso Ponderado
Descriptores	A-01	> 40 años	0.445
	A-02	30 - 40 años	0.297
	A-03	20 - 30 años	0.147
	A-04	10 -20 años	0.073
	A-05	< 10 años	0.037

Fuente: Adaptado de CENEPRED (2014).

Las viviendas de más de cuarenta años recibieron 0.445 de peso, siendo el mayor, frente a 0.037 de aquellas con menos de diez años. La clasificación atribuyó mayor vulnerabilidad a las construcciones más antiguas.

Tabla 22

Peso ponderado del parámetro material de construcción de paredes de la vivienda

Parámetro	Código	Material de construcción de paredes de la vivienda	Peso Ponderado
Descriptores	PR-01	Quincha (caña con barro)	0.507
	PR-02	Calamina y/o Triplay	0.263
	PR-03	Piedra con barro	0.123
	PR-04	Adobe o tapia	0.072
	PR-05	Ladrillo o bloque de cemento	0.035

Fuente: Adaptado de CENEPRED (2014).

La clasificación otorgó mayor peso a la quincha (0.507) y menor al ladrillo o bloque de cemento (0.035). Esta diferencia reflejó la fragilidad atribuida a cada material dentro de la evaluación.

Tabla 23

Peso ponderado del parámetro material de construcción en techo de la vivienda

Parámetro	Código	Material de construcción en techo de la vivienda	Peso Ponderado
Descriptores	T-01	Paja, hoja de palmera y similares	0.515
	T-02	Caña o estera con torta de barro	0.233
	T-03	Planchas de calamina	0.145
	T-04	Teja	0.071
	T-05	Concreto armado	0.035

Fuente: Adaptado de CENEPRED (2014).

Los techos de paja, hojas de palmera y similares recibieron el mayor peso (0.515), mientras que el concreto armado presentó el menor (0.035). Así, la evaluación consideró más desfavorables las primeras coberturas.

Tabla 24

Peso ponderado del parámetro material de construcción en piso de la vivienda

Parámetro	Código	Material de construcción en piso de la vivienda	Peso Ponderado
Descriptores	PS-01	Tierra	0.507
	PS-02	Cemento	0.263
	PS-03	Madera, entablados	0.123
	PS-04	Láminas asfálticas, vinílicas	0.072
	PS-05	Losetas, terrazas, cerámicos o similares	0.035

Fuente: Adaptado de CENEPRED (2014).

El piso de tierra obtuvo un peso de 0.507, frente a 0.035 de losetas, terrazas y cerámicos. El acabado del piso se utilizó como referencia de las condiciones de acondicionamiento de la vivienda.

Tabla 25

Peso ponderado del parámetro estado de conservación de la vivienda

Parámetro	Código	Estado de conservación de la vivienda	Peso Ponderado
Descriptores	EC-01	Destruído o muy mal conservado	0.514
	EC-02	Malo o con servicios muy deteriorados	0.273
	EC-03	Regular o con servicios funcionales	0.118
	EC-04	Bueno. Infraestructura y servicios funcionales	0.062
	EC-05	Muy bien conservado	0.033

Fuente: Adaptado de CENEPRED (2014).

Las viviendas destruidas o muy mal conservadas tuvieron el mayor peso (0.514), seguidas de aquellas en mal estado (0.273). Esta valoración destacó el deterioro como una condición prioritaria de atención.

Tabla 26*Peso ponderado del parámetro altura de la vivienda*

Parámetro	Código	Altura de la vivienda	Peso Ponderado
Descriptores	H-01	De 5 pisos a más	0.505
	H-02	4 pisos	0.262
	H-03	3 pisos	0.136
	H-04	2 pisos	0.060
	H-05	1 piso	0.037

Fuente: Adaptado de CENEPRED (2014).

La ponderación aumentó desde 0.037 para viviendas de un piso hasta 0.505 para aquellas de cinco pisos o más. Dentro de esta clasificación, se atribuyó mayor vulnerabilidad a las edificaciones de mayor altura.

Tabla 27*Peso ponderado del parámetro protección de la vivienda*

Parámetro	Código	Protección de la vivienda	Peso Ponderado
Descriptores	PV-01	Ninguna	0.503
	PV-02	Vegetación ribereña	0.260
	PV-03	Sacos de arena	0.134
	PV-04	Drenajes	0.068
	PV-05	Muros de contención	0.035

Fuente: Adaptado de CENEPRED (2014).

La ausencia de protección obtuvo 0.503, el peso mayor, mientras que los muros de contención presentaron el menor con 0.035. La clasificación valoró más favorablemente esta medida de protección frente a las demás alternativas.

Tabla 28*Peso ponderado del parámetro ingreso familiar promedio mensual*

Parámetro	Código	Ingreso familiar promedio mensual	Peso Ponderado
Descriptores	IF-01	Igual o menor a S/500	0.487
	IF-02	Entre S/500 a S/1000	0.286
	IF-03	Entre S/1000 a S/ 1500	0.124
	IF-04	S/ 1500 - S/ 2000	0.071
	IF-05	Mayor de S/ 2000	0.032

Fuente: Adaptado de CENEPRED (2014).

Los ingresos de hasta S/500 recibieron el mayor peso (0.487), frente a 0.032 de los superiores a S/2000. Esta diferencia destacó las restricciones económicas para financiar la prevención y afrontar la recuperación.

Tabla 29*Peso ponderado del parámetro condición laboral del jefe/a de hogar*

Parámetro	Código	Condición laboral del jefe/a de hogar	Peso Ponderado
Descriptores	CL-01	Desempleado	0.489
	CL-02	Trabajador familiar no remunerado/ Obrero	0.288
	CL-03	Empleado	0.125
	CL-04	Trabajador independiente	0.064
	CL-05	Empleador	0.033

Fuente: Adaptado de CENEPRED (2014).

El desempleo presentó el mayor peso (0.489), mientras que la condición de empleador recibió el menor (0.033). La falta de trabajo se consideró la situación más desfavorable dentro de esta clasificación.

- **Vulnerabilidad ambiental**

Para la vulnerabilidad ambiental se identificaron la cercanía de las viviendas a zonas agrícolas y botaderos, se analizó las malas prácticas de la población y su conocimiento de actividades de conservación ambiental.

Tabla 30

Peso ponderado del parámetro localización de áreas agrícolas frente a la zona de deslizamiento

Parámetro	Código	Localización de áreas agrícolas frente a la zona de deslizamiento.	Peso Ponderado
Descriptores	C-01	Muy cercana: 0m a 40m	0.491
	C-02	Cercana: > 40m - 80m	0.269
	C-03	Medianamente cerca: > 80m - 150m	0.146
	C-04	Alejada: > 150m - 400m	0.062
	C-05	Muy alejada: > 400m	0.032

Fuente: Adaptado de CENEPRED (2014).

Las áreas agrícolas situadas hasta cuarenta metros de la zona de deslizamiento recibieron el mayor peso (0.491). Con ello, se priorizó la cercanía para valorar la exposición de los terrenos productivos.

Tabla 31

Peso ponderado del parámetro malas prácticas de población que degradan el suelo

Parámetro	Código	Malas prácticas de población que degradan el suelo	Peso Ponderado
Descriptores	MP-01	Excavaciones, cortes de terreno o escombros en pendientes.	0.504
	MP-02	Presencia de agua que se filtra al suelo.	0.260
	MP-03	Tala de vegetación o limpieza de plantas.	0.134
	MP-04	Siembra en dirección a la pendiente.	0.067
	MP-05	No realiza malas prácticas.	0.035

Fuente: Adaptado de CENEPRED (2014).

Las excavaciones, cortes o escombros en pendientes fueron ponderadas en 0.504, seguidos de las filtraciones de agua con 0.260. Ambas condiciones concentraron la mayor valoración dentro de las prácticas de deterioro del terreno.

Tabla 32*Peso ponderado del parámetro cercanía de botaderos de basura*

Parámetro	Código	Cercanía de botaderos de basura	Peso Ponderado
Descriptores	B-01	Menor a 20 metros	0.493
	B-02	Entre 20 a 50 metros	0.271
	B-03	Entre 50 a 100 metros	0.147
	B-04	Entre 100 a 200 metros	0.055
	B-05	Mayor a 200 metros	0.034

Fuente: Adaptado de CENEPRED (2014).

Las distancias menores de veinte metros recibieron la mayor ponderación (0.493), frente a 0.034 de las superiores a doscientos metros. Se consideró más desfavorable la presencia de botaderos próximos a las viviendas.

Tabla 33*Peso ponderado del parámetro conocimiento de actividades de conservación ambiental*

Parámetro	Código	Conocimiento de actividades de conservación ambiental	Peso Ponderado
Descriptores	C-01	No conoce	0.481
	C-02	Escaso conocimiento	0.281
	C-03	Conoce, pero no aplica	0.146
	C-04	Conoce y aplica	0.060
	C-05	Aplica y difunde	0.031

Fuente: Adaptado de CENEPRED (2014).

El desconocimiento recibió 0.481, el mayor peso, mientras que la aplicación y difusión de prácticas obtuvo el menor con 0.031. La valoración destacó la puesta en práctica y transmisión del conocimiento como condiciones favorables para la conservación ambiental.

2.9.3.2 Estratificación de la vulnerabilidad

Tabla 34

Estratificación de los niveles de vulnerabilidad en el área de estudio – Sector La Colina

NIVELES DE VULNERABILIDAD	DESCRIPCIÓN	RANGOS
Vulnerabilidad Muy Alta	Viviendas con 6 a más personas o de 4 a 5 personas; población menor de 6 años, mayor de 60 años o de 51 a 60 años; dificultad motriz, cognitiva o mental; sin abastecimiento de agua o con agua de pozo, río o acequia; sin disposición sanitaria o con letrina; sin energía eléctrica o con conexión improvisada; sin nivel educativo o con primaria; sin capacitación o capacitación mayor a 2 años; sin seguro o con SIS; percepción fatalista o de desidia; vivienda ubicada entre 0 m y 80 m de la zona de deslizamiento; antigüedad mayor a 30 años; paredes de quinchá, caña con barro, calamina o triplay; techos de paja, hojas de palmera, caña o estera con barro; pisos de tierra o cemento; estado destruido, muy malo o malo; viviendas de 4 a 5 pisos o más; sin protección o con vegetación ribereña; ingresos menores o iguales a S/ 1000; jefe de hogar desempleado, trabajador familiar no remunerado u obrero; áreas agrícolas entre 0 m y 80 m; presencia de excavaciones, cortes, escombros o filtraciones de agua; botaderos a menos de 50 m; sin conocimiento o con escaso conocimiento de conservación ambiental.	$0.26 \leq 0.49$ 8 4
Vulnerabilidad Alta	Viviendas con 3 personas; población de 7 a 12 años; dificultad visual; abastecimiento mediante camión cisterna u otro; disposición sanitaria con pozo ciego o negro; energía eléctrica mediante generador; nivel educativo secundario; capacitación realizada hace 1 año; seguro EsSalud; percepción preocupada; vivienda ubicada entre más de 80 m y 150 m de la zona de deslizamiento; antigüedad de 20 a 30 años; paredes de piedra con barro; techo de planchas de calamina; piso de madera o entablado; estado de conservación regular; vivienda de 3 pisos; protección con sacos de arena; ingreso familiar entre S/ 1000 y S/ 1500; jefe de hogar empleado; áreas agrícolas entre más de 80 m y 150 m; tala de vegetación o limpieza de plantas; botaderos entre 50 m y 100 m; conoce, pero no aplica actividades de conservación ambiental.	$0.13 < 0.26$ 2 8
Vulnerabilidad Media	Viviendas con 2 personas; población de 13 a 19 años; dificultad para oír o hablar; abastecimiento mediante pilón; conexión de desagüe improvisada, expuesta o	$0.06 < 0.13$ 4 2

NIVELES DE VULNERABILIDAD	DESCRIPCIÓN	RANGOS
	en mal estado; energía eléctrica mediante panel solar; educación superior no universitaria; capacitación realizada hace 6 meses; seguro de FF. AA. o PNP; percepción positiva; vivienda ubicada entre más de 150 m y 400 m de la zona de deslizamiento; antigüedad de 10 a 20 años; paredes de adobe o tapia; techo de teja; piso de láminas asfálticas o vinílicas; buen estado de conservación; vivienda de 2 pisos; protección mediante drenajes; ingreso familiar entre S/ 1500 y S/ 2000; jefe de hogar trabajador independiente; áreas agrícolas entre más de 150 m y 400 m; siembra en dirección a la pendiente; botaderos entre 100 m y 200 m; conoce y aplica actividades de conservación ambiental.	
Vulnerabilidad Baja	Viviendas con 1 persona; población de 20 a 50 años; sin dificultad permanente; abastecimiento mediante red de agua potable; disposición sanitaria mediante red de desagüe; energía eléctrica mediante red pública; educación superior universitaria o posgrado; capacitación realizada hace 3 meses; seguro privado; percepción optimista; vivienda ubicada a más de 400 m de la zona de deslizamiento; antigüedad menor a 10 años; paredes de ladrillo o bloque de cemento; techo de concreto armado; piso de losetas, terrazos, cerámicos o similares; muy buen estado de conservación; vivienda de 1 piso; protección mediante muros de contención; ingreso familiar mayor a S/ 2000; jefe de hogar empleador; áreas agrícolas a más de 400 m; no realiza malas prácticas; botaderos a más de 200 m; aplica y difunde actividades de conservación ambiental.	0.03 6 S < 0.06 4

Fuente: Adaptado de CENEPRED (2014)

2.9.3.3 Recopilación de información de la vulnerabilidad en campo

Para recopilar la información de los parámetros de vulnerabilidad seleccionados se diseñó como instrumento una ficha de recolección de datos, y un cuestionario, los datos objetivos como distancias, fueron medidos utilizando los sistemas de información geográfica y los datos topográficos de las viviendas del sector La Colina.

- **Validez de instrumentos**

Para la validación de los instrumentos de vulnerabilidad, se utilizó el formato propuesto por Milla et al. (2023). Los instrumentos fueron validados por 3 expertos. Los formatos detallados se presentan en anexos.

- **Confiabilidad**

La confiabilidad del cuestionario se evaluó mediante estabilidad temporal por test-retest, procedimiento que permite verificar la consistencia de las mediciones en dos momentos bajo condiciones similares. Su aplicación fue coherente con la estructura del instrumento, conformado por indicadores diferenciados y ponderaciones específicas según la metodología del CENEPRED. La reaplicación se realizó a una submuestra aleatoria de 30 viviendas, tamaño aceptable para estudios piloto de confiabilidad de cuestionarios (Bujang et al., 2024). La concordancia del índice continuo se analizó mediante el Coeficiente de Correlación Intraclass (ICC), recomendado para mediciones cuantitativas repetidas (Koo & Li, 2016). Para los niveles categóricos se empleó el Kappa de Cohen, adecuado para evaluar el acuerdo en variables clasificadas por categorías (McHugh, 2012).

Tabla 35

Cálculo de ICC.

Indicador	Valor
Número de lotes evaluados	30
Número de mediciones	2
Media del test	0,242403
Media del retest	0,242299
Media general	0,242351
Cuadrado medio entre lotes (MS_R)	0,006168411
Cuadrado medio entre mediciones (MS_C)	0,000000163
Cuadrado medio del error (MS_E)	0,000002970
ICC obtenido	0,9991
Interpretación	Confiabilidad excelente

Fuente: Elaboración propia.

Tabla 36*Coefficiente Kappa de Cohen*

Indicador	Cálculo	Resultado
Acuerdo observado	$30 / 30$	1,000
Acuerdo esperado por azar	$[(22 \times 22) + (8 \times 8)] / 30^2$	0,6089
Kappa de Cohen	$((1,0000 - 0,6089) / (1 - 0,6089))$	1,000
Resultado del coeficiente	$(K = 1,0000)$	Concordancia perfecta
Interpretación	$(K = 1,0000)$	El instrumento presenta confiabilidad categórica excelente entre el test y retest.
Concordancia categórica	$(30 / 30 \times 100)$	100%

Fuente: Elaboración propia.

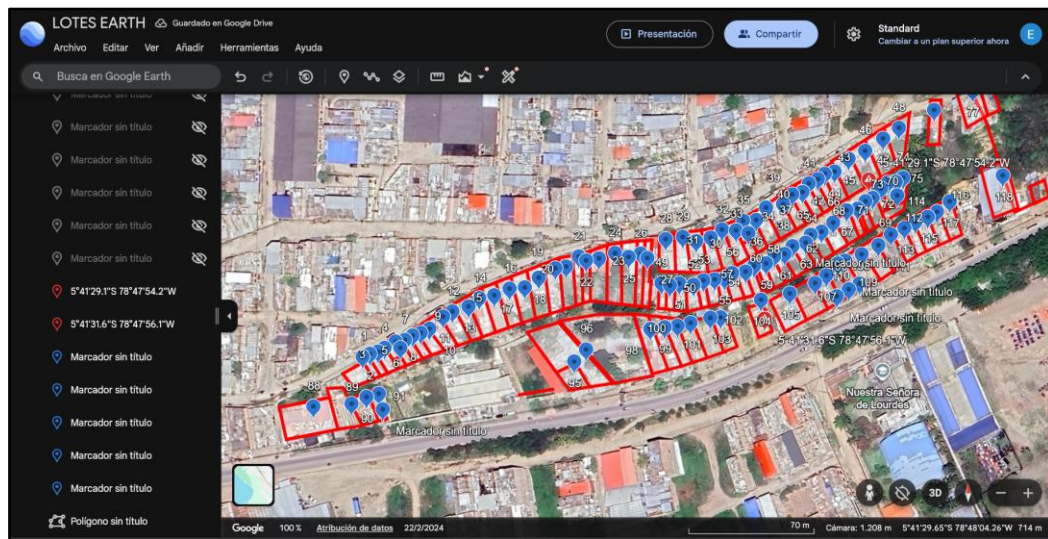
Nota: El ICC de 0.9991 y el coeficiente de Kappa de Cohen igual a 1.0 indican que el instrumento evaluado por el test retest arroja datos concordantes y se demuestra su excelente confiabilidad.

- **Digitalización de los instrumentos**

Para optimizar el flujo de trabajo se optó por digitalizar los instrumentos, primero para tener referencia de los lotes en campo, se insertaron los lotes de las viviendas en Google Earth, luego se digitalizó el cuestionario en AppSheet, esto permitió enlazar las respuestas del cuestionario con la ficha de recolección de datos a través de una base de datos.

Figura 18

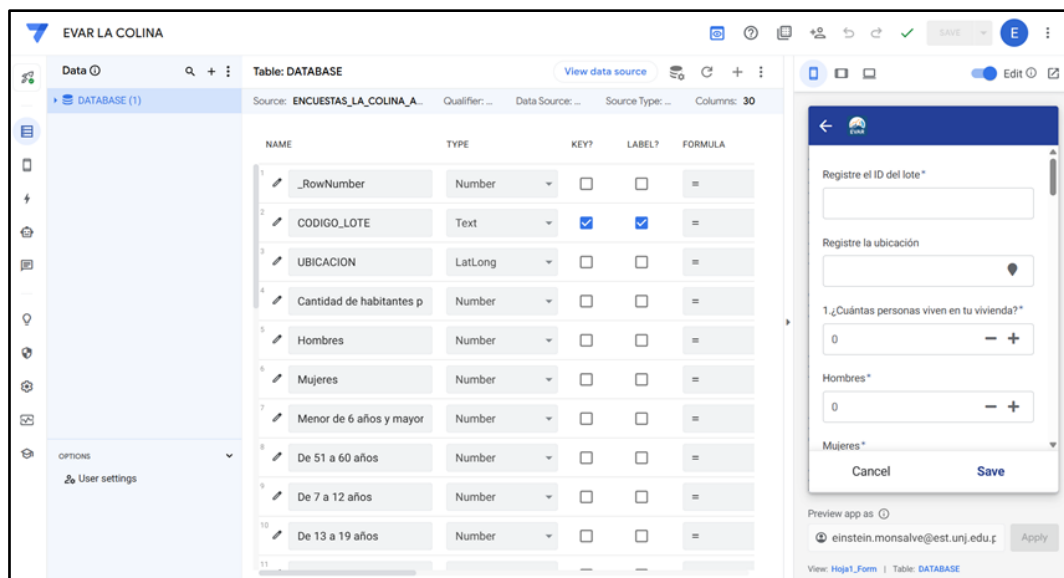
Referenciación de lotes en Google Earth para trabajo en campo



Nota: La figura muestra la referenciación de los lotes de las viviendas en Google Earth, utilizada para facilitar su identificación durante el trabajo de campo.

Figura 19

Uso de AppSheet para digitalizar el cuestionario



Nota: La figura muestra la digitalización del cuestionario en AppSheet para registrar y vincular la información recopilada en campo con la base de datos.

- **Aplicación de los instrumentos**

Se visitó vivienda a vivienda y se aplicó el cuestionario, asimismo se identificaron aquellos lotes vacíos que no eran viviendas y se excluyeron del análisis.

Figura 20

Aplicación de cuestionario en campo



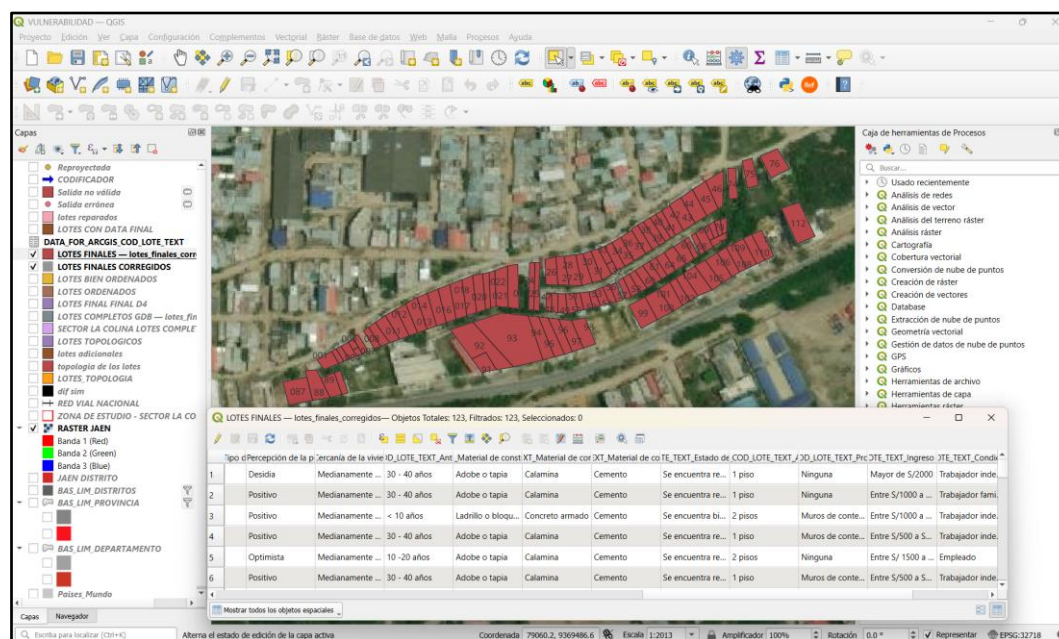
Nota: La figura muestra la aplicación del cuestionario a los residentes del sector La Colina durante el trabajo de campo.

2.9.3.4 Procesamiento de la información de la vulnerabilidad

Finalmente, la información de campo se procesó en el software QGIS de acuerdo a las ponderaciones calculadas con la metodología CENEPRED.

Figura 21

Procesamiento de información de campo a QGIS.



Nota: La figura muestra el procesamiento en QGIS de la información recopilada en campo para el análisis de la vulnerabilidad del sector La Colina.

2.9.4 Establecimiento y modelamiento del nivel de riesgo

2.9.4.1 Cálculo del riesgo por CENEPRED

Para cumplir el objetivo específico 3, se calculó el valor de riesgo de acuerdo con el Manual para la Evaluación de Riesgos Originados por Fenómenos Naturales del CENEPRED (2014), en función de los niveles de peligro y vulnerabilidad previamente identificados y analizados. Posteriormente, el valor obtenido permitió establecer el nivel de riesgo correspondiente.

Ecuación del riesgo CENEPRED (2014):

$$R = P \times V \quad (4)$$

Donde; R es el riesgo: P es el peligro y V es la vulnerabilidad

Nota: Adaptado del manual de evaluación de riesgos originados por fenómenos naturales 02 versión, CENEPRED, 2014.

- **Matriz de riesgo por CENEPRED**

Para el cálculo de los valores se resolvió la matriz de riesgo.

Tabla 37

Matriz de riesgo

NIVEL DE PELIGRO	VALOR DE PELIGRO	NIVELES DE RIESGO			
PMA	0.365	0.023	0.048	0.098	0.180
PA	0.287	0.018	0.038	0.077	0.142
PM	0.220	0.014	0.029	0.059	0.109
PB	0.194	0.012	0.026	0.052	0.096
VALOR DE VULNERABILIDAD		0.064	0.132	0.268	0.494
NIVEL DE VULNERABILIDAD		VB	VM	VA	VMA

Fuente: Adaptado CENEPRED (2014).

- **Estratificación de niveles de riesgo**

Tabla 38

Estratificación de los niveles de riesgo en el área de estudio – Sector La Colina

NIVELES DE RIESGO	DESCRIPCIÓN	RANGOS
Riesgo Muy Alto	<p>Predominan áreas de la Formación Bellavista, ubicadas en colinas bajas fuertemente inclinadas, con pendientes entre 20° y 30°, áreas abiertas con poca o ninguna cobertura vegetal, condiciones lluviosas y eventos de precipitación con periodos de retorno entre 15 y 25 años.</p> <p>Se caracterizan por viviendas con 4 a más personas, presencia de población menor de 6 años, mayor de 60 años o entre 51 y 60 años, dificultad motriz, cognitiva o mental, ausencia de abastecimiento de agua o uso de pozo, río o acequia, falta de disposición sanitaria o uso de letrina, falta de energía eléctrica o conexión improvisada, nivel educativo sin instrucción o primaria, sin capacitación o con capacitación mayor a 2 años, sin seguro o con SIS, percepción fatalista o desidia, ubicación entre 0 m y 80 m de la zona de deslizamiento, antigüedad mayor a 30 años, paredes de quincha, caña con barro, calamina o triplay, techos de paja, hojas de palmera, caña o estera con barro, pisos de tierra o cemento, estado destruido,</p>	$0.077 \leq S \leq 0.18$

	<p>muy malo o malo, viviendas de 4 a 5 pisos o más, ausencia de protección o presencia de vegetación ribereña, ingresos menores o iguales a S/ 1000, jefe de hogar desempleado, trabajador familiar no remunerado u obrero, áreas agrícolas entre 0 m y 80 m, presencia de excavaciones, cortes, escombros o filtraciones de agua, botaderos a menos de 50 m y escaso o nulo conocimiento de conservación ambiental.</p>	
Riesgo Alto	<p>Predominan áreas de la Formación Tamborapa, asociadas a colinas altas empinadas en rocas del Jurásico inferior, pendientes entre 30° y 40°, zonas urbanizadas en ladera, condiciones muy lluviosas y precipitaciones con periodos de retorno entre 5 y 15 años.</p> <p>Presentan 3 personas por vivienda, población de 7 a 12 años, dificultad visual, abastecimiento mediante camión cisterna u otro, disposición sanitaria con pozo ciego, energía eléctrica mediante generador, nivel educativo secundario, capacitación hace 1 año, seguro EsSalud, percepción preocupada, viviendas entre más de 80 m y 150 m de la zona de deslizamiento, antigüedad de 20 a 30 años, paredes de piedra con barro, techo de calamina, piso de madera o entablado, estado regular, viviendas de 3 pisos, protección con sacos de arena, ingreso familiar entre S/ 1000 y S/ 1500, jefe de hogar empleado, áreas agrícolas entre más de 80 m y 150 m, tala de vegetación o limpieza de plantas, botaderos entre 50 m y 100 m y conocimiento ambiental sin aplicación práctica.</p>	$0.029 \leq S < 0.077$
Riesgo Medio	<p>Predominan áreas de la Formación Bellavista, ubicadas en colinas bajas fuertemente inclinadas en rocas del Neógeno, con pendientes moderadamente empinadas entre 20° y 30°, presencia de áreas abiertas sin vegetación, condiciones lluviosas y eventos con periodos de retorno entre 15 y 25 años. Pese a la inestabilidad moderada, las precipitaciones persistentes pueden generar erosión y movimientos de ladera.</p> <p>Presentan 2 personas por vivienda, población de 13 a 19 años, dificultad para oír o hablar, abastecimiento mediante pilón, conexión de desagüe improvisada, expuesta o en mal estado, educación superior no universitaria, capacitación realizada hace 6 meses, seguro de FF. AA. o PNP, percepción positiva, viviendas ubicadas entre 150 m y 400 m de la zona de deslizamiento, antigüedad de 10 a 20 años, paredes de adobe o tapia, techo de teja, piso de láminas asfálticas o vinílicas, buen estado de conservación, viviendas de 2 pisos, protección mediante drenajes, ingreso familiar entre S/ 1500 y S/ 2000, jefe de hogar trabajador independiente, áreas agrícolas entre más de 150 m y 400 m, siembra en dirección a la pendiente, botaderos entre 100 m y 200 m, y conocimiento con aplicación de actividades de conservación ambiental.</p>	$0.012 \leq S < 0.029$
Riesgo Bajo	<p>Corresponde a zonas con valores de peligro entre $0.177 \leq P < 0.194$, desarrolladas sobre la Formación Chulec y el Grupo Goyllarisquizga, asociadas a colinas moderadamente</p>	$0.006 \leq S < 0.012$

empinadas en rocas del Neógeno y a complejos de terrazas inundables y no inundables, con pendientes entre 0° y 20°, presencia de áreas agrícolas heterogéneas y vegetación arbustiva, condiciones moderadamente lluviosas o ligeramente lluviosas y eventos con periodos de retorno superiores a 25 años. Estas características proporcionan mayor estabilidad relativa del terreno y reducen la probabilidad de deslizamientos.

Caracterizadas por presentar 1 persona por vivienda, población de 20 a 50 años, ausencia de dificultad permanente, abastecimiento mediante red de agua potable, disposición sanitaria mediante red de desagüe, energía eléctrica por red pública, educación superior universitaria o posgrado, capacitación realizada hace 3 meses, seguro privado, percepción optimista, viviendas ubicadas a más de 400 m de la zona de deslizamiento, antigüedad menor a 10 años, paredes de ladrillo o bloque de cemento, techo de concreto armado, piso de losetas, terrazos, cerámicos o similares, muy buen estado de conservación, viviendas de 1 piso, protección mediante muros de contención, ingreso familiar mayor a S/ 2000, jefe de hogar empleador, áreas agrícolas a más de 400 m, ausencia de malas prácticas, botaderos a más de 200 m, y aplicación y difusión de actividades de conservación ambiental.

Fuente: Adaptado CENEPRED (2014).

2.9.4.2 Proceso de modelamiento del riesgo con algoritmos ensamblados

- Comprensión del problema

Clasificación supervisada:

La clasificación supervisada es una técnica de aprendizaje automático que utiliza datos previamente etiquetados para entrenar un modelo capaz de asignar nuevos casos a una categoría o clase definida (James et al., 2021).

En la fase de comprensión del problema se definió el propósito del modelamiento. La investigación fue formulada como un problema de clasificación supervisada multicategórica, debido a que la variable de salida correspondió al nivel de riesgo por deslizamiento clasificado en categorías a partir de la metodología CENEPRED. En este caso, los algoritmos ensamblados tuvieron como finalidad estimar directamente el nivel de riesgo a partir de variables asociadas al peligro, vulnerabilidad y exposición de los lotes o viviendas evaluadas.

Variable objetivo (dependiente)

El nivel de riesgo calculado por CENEPRED fue la salida en la base de datos para entrenar los modelos y generar los ensambles.

Variables predictoras (independientes)

Se utilizaron 29 variables de entrada, estas fueron las seleccionadas para el análisis del peligro y vulnerabilidad tras la aplicación de la metodología CENEPRED. Estas sirvieron para armar la matriz de la base de datos.

Figura 22

Variables de entrada de la base de datos

Variables de entrada (29 parámetros)	
1. Geología	16. Cercanía de la vivienda a la zona de deslizamientos
2. Geomorfología	17. Antigüedad de la vivienda
3. Pendiente	18. Material de construcción de paredes de la vivienda
4. Uso actual del suelo	19. Material de construcción en techo de la vivienda
5. Precipitación	20. Material de construcción en piso de la vivienda
6. Cantidad de habitantes por vivienda	21. Estado de conservación de la vivienda
7. Grupo etario más vulnerable en la vivienda	22. Altura de la vivienda
8. Personas con dificultad permanente	23. Protección de la vivienda
9. Abastecimiento de agua en la vivienda	24. Ingreso familiar promedio mensual
10. Disposición sanitaria de excretas en la vivienda	25. Condición laboral del jefe de hogar
11. Fuente de energía eléctrica	26. Localización de áreas agrícolas frente a la zona de deslizamiento
12. Nivel educativo más alto en la vivienda	27. Malas prácticas de población que degradan el suelo
13. Capacitación de la población en temas de GRD	28. Cercanía de botaderos de basura
14. Tipo de seguro de salud de la población	29. Conocimiento de actividades de conservación ambiental
15. Percepción de la población acerca de los deslizamientos	

Nota: La figura presenta las 29 variables de entrada empleadas para conformar la base de datos utilizada en el modelamiento del nivel de riesgo por deslizamiento.

Google Colab:

La investigación utilizó código de Python aplicado en la plataforma de Google Colab para modelar el riesgo. El proceso de modelamiento comenzó con la importación de librerías y recursos necesarios para la ejecución del código, así como la definición del “target” o variable objetivo.

Figura 23

Importación de librerías de Python y definición de la variable objetivo

```
import os
import re
import warnings
import unicodedata
from datetime import datetime

import joblib
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

from google.colab import files

from sklearn.base import clone
from sklearn.compose import ColumnTransformer
from sklearn.ensemble import ExtraTreesClassifier, GradientBoostingClassifier, RandomForestClassifier, StackingClassifier
from sklearn.inspection import permutation_importance
from sklearn.metrics import (
    ConfusionMatrixDisplay,
    accuracy_score,
    balanced_accuracy_score,
    classification_report,
    confusion_matrix,
    f1_score,
    make_scorer,
    precision_score,
    recall_score,
)
from sklearn.model_selection import StratifiedKFold, cross_val_predict, cross_validate
from sklearn.pipeline import Pipeline
from sklearn.preprocessing import LabelEncoder, OneHotEncoder
from sklearn.tree import DecisionTreeClassifier
from sklearn.linear_model import LogisticRegression

try:
    from skopt import BayesSearchCV
    from skopt.space import Categorical, Integer, Real
except ImportError:
    import sys
    import subprocess

    subprocess.check_call([sys.executable, "-m", "pip", "install", "scikit-optimize"])
    from skopt import BayesSearchCV
    from skopt.space import Categorical, Integer, Real

warnings.filterwarnings("ignore", category=FutureWarning)
warnings.filterwarnings("ignore", category=UserWarning)

RANDOM_STATE = 42
TARGET = "NIVEL-DE-RIESGO"
NOMBRE_ARCHIVO_ESPERADO = "DATA_TESIS.csv"
```

Nota: Se establecen las dependencias, la semilla de reproducibilidad y los parámetros iniciales.

- **Comprensión de los datos.**

Base de datos integrada:

La base de datos estuvo conformada por 1092 registros y 30 variables tratadas a nivel categórico: 29 predictoras y una variable objetivo. Incluyó 123 registros del sector La Colina y 969 provenientes del Plan de prevención y reducción del riesgo de desastres del distrito de Jangas 2023–2030, elaborado por la Municipalidad Distrital de Jangas y publicado en la biblioteca SIGRID del CENEPRED. Los registros de Jangas se incorporaron como información complementaria para el entrenamiento, manteniendo a La Colina como ámbito territorial de la investigación. El aprendizaje supervisado se orientó a identificar las relaciones entre los indicadores y las categorías de riesgo asignadas mediante la metodología CENEPRED.

La utilización conjunta de ambas bases se sustentó en el contraste de las variables y los descriptores de sus respectivas matrices de evaluación, considerando la equivalencia de significado y de los criterios de evaluación de los indicadores. Se identificó equivalencia completa en 24 de las 29 variables predictoras (82,76 %), equivalencia parcial en tres (10,34 %) y ausencia de descriptores coincidentes en dos (6,90 %). Asimismo, se verificó la equivalencia de las cuatro categorías de la variable objetivo, nivel de riesgo. Estos resultados sustentan una amplia correspondencia metodológica entre las fuentes utilizadas. El detalle de la comparación se presenta en el anexo 04.

Google Colab para la comprensión de los datos:

En Colab, la compresión de los datos abarcó la importación de la base de datos, su revisión inicial, la validación de la variable objetivo, la detección de categorías que estuvieran repetidas por espacios, mayúsculas o minúsculas.

Figura 24

Ubicación de la variable objetivo en la base de datos

```
if TARGET not in df_raw.columns:
    columnas_parecidas = [col for col in df_raw.columns if str(col).strip().upper() == TARGET]
    mensaje = f"No se encontró la columna objetivo exacta: {TARGET}."
    if columnas_parecidas:
        mensaje += f" Columnas parecidas encontradas: {columnas_parecidas}. Revise espacios o caracteres especiales en el nombre."
        raise ValueError(mensaje)

print(f"Variable objetivo encontrada correctamente: {TARGET}")
```

Variable objetivo encontrada correctamente: NIVEL-DE-RIESGO

Nota: Se verifica la presencia de la variable objetivo “NIVEL-DE-RIESGO”.

Figura 25

Detección duplicadas por formato

```
def detectar_variantes_categoricas(dataframe, columnas):
    # Detecta categorías con diferencias solo por espacios o mayúsculas/minúsculas.
    registros = []

    for col in columnas:
        valores = dataframe[col].dropna().map(str)
        grupos = {}

        for valor in valores.unique():
            clave_normalizada = valor.strip().lower()
            grupos.setdefault(clave_normalizada, set()).add(valor)

        for variantes in grupos.values():
            if len(variantes) > 1:
                registros.append({
                    "columna": col,
                    "variantes_detectadas": sorted(variantes),
                    "cantidad_variantes": len(variantes),
                })

    return pd.DataFrame(registros)

variantes_categoricas = detectar_variantes_categoricas(df_raw, df_raw.columns)

if variantes_categoricas.empty:
    print("No se detectaron categorías repetidas por diferencias de espacios o mayúsculas/minúsculas.")
else:
    print("Posibles categorías repetidas detectadas:")
    display(variantes_categoricas)
```

Nota: Se identifican duplicados con diferencias de espacios, mayúsculas o minúsculas.

Preparación de los Datos

En la fase de preparación de los datos se realizó la limpieza, transformación y organización de la base para el entrenamiento de los algoritmos. Primero, se uniformizaron las variables categóricas mediante la eliminación de espacios innecesarios y la corrección de diferencias de escritura. Posteriormente, se trataron los valores faltantes y se definieron las variables independientes y la variable dependiente.

Los registros sin información en la variable objetivo fueron eliminados, debido a que no podían ser utilizados en un proceso de aprendizaje supervisado, de la misma manera las variables que tenían datos faltantes. Luego, se definió la matriz X, conformada por las variables predictoras, y el vector y, correspondiente al nivel de riesgo por deslizamiento.

Figura 26

Eliminación de registros sin salidas y definición de x e y

```
df_modelo = df.copy()

registros_iniciales = len(df_modelo)

# Eliminar solo registros sin variable de salida
df_modelo = df_modelo.dropna(subset=[TARGET]).copy()
df_modelo = df_modelo[df_modelo[TARGET].astype(str).str.strip() != ""].copy()

registros_eliminados = registros_iniciales - len(df_modelo)

if registros_eliminados > 0:
    print(f"Se eliminaron {registros_eliminados} registros sin clase de riesgo en {TARGET}.")

# Separar variables de entrada y salida
X = df_modelo.drop(columns=[TARGET]).copy()
y = df_modelo[TARGET].copy()

# Limpieza básica de espacios en blanco
for columna in X.columns:
    X[columna] = X[columna].astype(str).str.strip()

y = y.astype(str).str.strip()

print("Dimensión de X:", X.shape)
print("Dimensión de y:", y.shape)
print("\nPrimeras filas de X:")
display(X.head())
```

Nota: Preparación de las variables de entrada y de la variable objetivo para el modelado.

Para la variable objetivo se utilizó un “label encoder”, de esta forma la variable textual se transforma a números para evaluar los modelos.

Figura 27

Label Encoder para codificación de la variable objetivo

```
label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)

mapa_clases = pd.DataFrame({
    "codigo": range(len(label_encoder.classes_)),
    "clase_original": label_encoder.classes_,
})

print("Correspondencia entre código numérico y clase original:")
display(mapa_clases)

conteo_clases = pd.Series(y_encoded).value_counts().sort_index()
conteo_clases.index = label_encoder.inverse_transform(conteo_clases.index)
print("\nFrecuencia por clase usada en modelado:")
display(conteo_clases.to_frame("frecuencia"))
```

Nota: El nivel de riesgo se codificó utilizando la herramienta Label Encoder.

- **Configuración de la validación cruzada estratificada**

Se configuró una validación cruzada estratificada para mantener aproximadamente la proporción de las clases del nivel de riesgo en cada partición. Previamente, se verificó que la clase minoritaria contaba con 21 registros, lo que permitió establecer diez particiones para la evaluación principal de los modelos. Asimismo, para el stacking se configuraron cinco particiones internas estratificadas, destinadas a generar las predicciones utilizadas como entradas del metamodelo. En esta etapa se definió el esquema de evaluación interna; su ejecución se realizó posteriormente durante el entrenamiento y la evaluación de los modelos.

Figura 28

Configuración del esquema de validación

```
frecuencia_minima = int(pd.Series(y_encoded).value_counts().min())

if frecuencia_minima < 2:
    raise ValueError(
        "Hay al menos una clase con menos de 2 registros. "
        "No es posible aplicar validación cruzada estratificada. Revise o agrupe clases muy escasas."
    )

if frecuencia_minima >= 10:
    n_splits = 10
else:
    n_splits = min(5, frecuencia_minima)

cv = StratifiedKFold(n_splits=n_splits, shuffle=True, random_state=RANDOM_STATE)

# Validación interna usada por StackingClassifier.
n_splits_stacking_interno = min(5, n_splits)
cv_stacking_interno = StratifiedKFold(
    n_splits=n_splits_stacking_interno,
    shuffle=True,
    random_state=RANDOM_STATE,
)

print(f"Frecuencia de la clase minoritaria: {frecuencia_minima}")
print(f"Número de particiones para evaluación principal: {n_splits}")
print(f"Número de particiones internas para stacking: {n_splits_stacking_interno}")
```

Nota: Se emplea validación cruzada estratificada para preservar la distribución de las clases.

Como controles frente al sobreajuste y la filtración de información, se excluyeron de los predictores la variable objetivo y los índices finales de peligro, vulnerabilidad y riesgo. OneHotEncoder se integró en los pipelines y se ajustó únicamente con los registros de entrenamiento de cada partición.

Figura 29

Preprocesamiento con OneHotEncoder

```
def crear_onehot_encoder():  
    # Compatible con versiones recientes y anteriores de scikit-learn.  
    try:  
        return OneHotEncoder(handle_unknown="ignore", sparse_output=False)  
    except TypeError:  
        return OneHotEncoder(handle_unknown="ignore", sparse=False)  
  
def crear_preprocesador(columnas_predictoras):  
    return ColumnTransformer(  
        transformers=[  
            ("onehot", crear_onehot_encoder(), list(columnas_predictoras)),  
        ],  
        remainder="drop",  
        verbose_feature_names_out=True,  
    )  
  
columnas_predictoras = list(X.columns)  
preprocesador_base = crear_preprocesador(columnas_predictoras)  
  
print(f"Número de variables predictoras categóricas: {len(columnas_predictoras)}")
```

Nota: Para codificar las variables predictoras categóricas se utilizó el One Hot Encoder.

- Modelado

En la fase de modelado se entrenaron algoritmos de clasificación supervisada para estimar el nivel de riesgo por deslizamiento. Se utilizaron modelos base y modelos ensamblados, con la finalidad de comparar su desempeño y seleccionar la alternativa con mejor capacidad predictiva.

Los modelos base considerados fueron Decision Tree, Random Forest, Extra Trees y Gradient Boosting. Posteriormente, estos modelos fueron optimizados mediante búsqueda bayesiana de hiperparámetros y utilizados en esquemas de ensamblaje mediante Stacking.

Los algoritmos de Machine Learning, seleccionados fueron:

Tabla 39

Algoritmos seleccionados de Machine Learning.

Algoritmo	Alcance	Limitación
Random Forest (RF)	Clasifica el riesgo con múltiples árboles, analiza relaciones no lineales entre los 29 parámetros y permite identificar variables más influyentes. (Breiman, 2001; Sun et al., 2020; Zhou et al., 2021)	Puede afectarse por clases desbalanceadas y su interpretación detallada es limitada. (Sun et al., 2020; Zhou et al., 2021)
Extra Trees (ET)	Usa árboles extremadamente aleatorizados, reduce el sobreajuste y mejora la generalización en la clasificación del riesgo. (Geurts et al., 2006)	Su alta aleatoriedad reduce la interpretabilidad y depende de la calidad de los datos. (Geurts et al., 2006)
Decision Tree (DT)	Clasifica mediante reglas jerárquicas fáciles de interpretar, permitiendo relacionar parámetros físicos y sociales con el nivel de riesgo. (Breiman et al., 1984; Chen & Zhang, 2021)	Tiende al sobreajuste y puede ser menos estable frente a modelos ensamblados. (Breiman et al., 1984; Chen & Zhang, 2021)
Gradient Boosting (GB)	Corrige errores de forma secuencial y capta interacciones complejas entre los 29 parámetros. (Friedman, 2001; Sahin, 2020)	Requiere mayor calibración y es sensible a hiperparámetros. (Friedman, 2001; Sahin, 2020)

Optimización de Hiperparámetros:

Se optimizaron los hiperparámetros de Decision Tree, Random Forest, Extra Trees y Gradient Boosting mediante búsqueda bayesiana con BayesSearchCV, utilizando 20 iteraciones por modelo y validación cruzada estratificada de diez particiones. Según el algoritmo, se consideraron parámetros como profundidad, número de estimadores, tasa de aprendizaje, criterio de clasificación y ponderación de clases. La selección se basó en el F1 macro de validación, que combina precisión y sensibilidad por clase y promedia sus puntuaciones con igual peso para cada categoría, independientemente de su frecuencia (scikit-learn, s. f.).

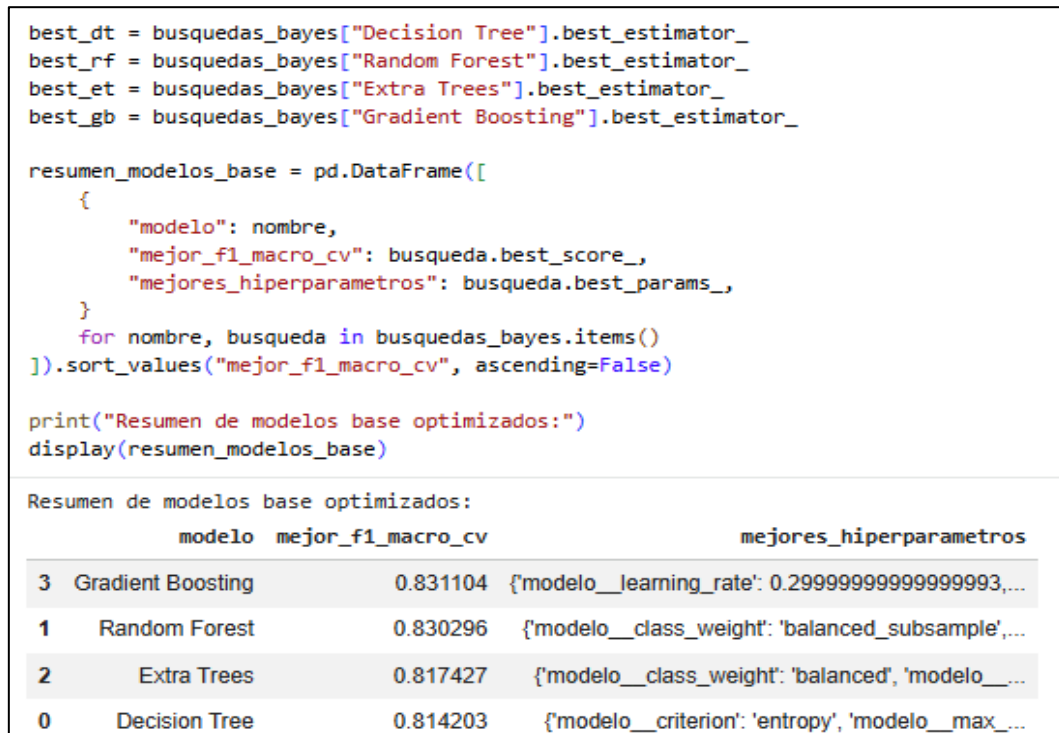
Posteriormente, los modelos individuales y ensambles stacking se compararon mediante métricas promedio por partición, complementadas con métricas y matrices de confusión obtenidas mediante cross_val_predict para las alternativas seleccionadas. Estos resultados de evaluación interna se diferenciaron del rendimiento sobre la base completa.

Mejores modelos base optimizados:

Después de la optimización bayesiana, se extrajeron los mejores estimadores de cada modelo base: Decision Tree, Random Forest, Extra Trees y Gradient Boosting.

Figura 30

Código para guardar los mejores modelos base optimizados



Nota: Se muestra los modelos con su F1 macro y los hiperparámetros óptimos.

Ensamblaje de algoritmos con el método Stacking:

Luego de entrenar y optimizar los modelos base, se aplicó el ensamblaje mediante Stacking o stacked generalization, técnica que combina las predicciones de varios modelos de primer nivel para utilizarlas como entrada en un meta-aprendiz encargado de realizar la clasificación final (Wolpert, 1992). En la investigación, este enfoque permitió integrar diferentes combinaciones de algoritmos base para clasificar el nivel de riesgo por deslizamiento a partir de los 29 parámetros de peligro y vulnerabilidad, empleando Regresión Logística como meta-aprendiz principal y Gradient Boosting como alternativa.

Combinaciones de algoritmos:

Tabla 40

Combinaciones de algoritmos para el ensamblaje con stacking

Combinación	Modelos de primer nivel	Modelo de segundo nivel
1	Decision Tree + Random Forest + Extra Trees + Gradient Boosting	Regresión Logística
2	Random Forest + Extra Trees + Gradient Boosting	Regresión Logística
3	Random Forest + Extra Trees	Regresión Logística
4	Random Forest + Gradient Boosting	Regresión Logística
5	Decision Tree + Random Forest + Extra Trees + Gradient Boosting	Gradient Boosting

El ensamblaje de algoritmos se realizó mediante Stacking, utilizando como modelos de primer nivel los algoritmos base previamente optimizados. Primero, se definió la estructura del clasificador Stacking y el meta-aprendiz encargado de realizar la clasificación final. Luego, se establecieron cinco combinaciones de modelos base, considerando Regresión Logística como meta-aprendiz principal y Gradient Boosting como alternativa.

Cada combinación fue evaluada mediante validación cruzada y métricas de clasificación, con la finalidad de comparar su desempeño en la estimación del nivel de riesgo por deslizamiento. La selección del mejor ensamble candidato se realizó considerando principalmente el F1-score macro y el balanced accuracy, debido a que estas métricas permiten valorar el rendimiento del modelo de manera más equilibrada entre las clases de riesgo.

Figura 31

Funciones para la creación del meta-aprendiz y del clasificador Stacking

```
def crear_meta_logistica():
    return LogisticRegression(
        max_iter=5000,
        class_weight="balanced",
        solver="lbfgs",
        random_state=RANDOM_STATE
    )

def crear_stacking(modelos_base, meta_aprendiz):
    return StackingClassifier(
        estimators=modelos_base,
        final_estimator=meta_aprendiz,
        stack_method="predict_proba",
        cv=cv_stacking_interno,
        n_jobs=-1,
        passthrough=False
    )
```

Nota: Código empleado para configurar el meta-aprendiz y el clasificador Stacking.

Figura 32

Definición de combinaciones de modelos ensamblados mediante Stacking

```
modelos_base = {
    "dt": best_dt,
    "rf": best_rf,
    "et": best_et,
    "gb": best_gb
}

combinaciones = {
    "C1: DT + RF + ET + GB | LR": ([ "dt", "rf", "et", "gb"], crear_meta_logistica()),
    "C2: RF + ET + GB | LR": ([ "rf", "et", "gb"], crear_meta_logistica()),
    "C3: RF + ET | LR": ([ "rf", "et"], crear_meta_logistica()),
    "C4: RF + GB | LR": ([ "rf", "gb"], crear_meta_logistica()),
    "C5: DT + RF + ET + GB | GB": (
        [ "dt", "rf", "et", "gb"],
        GradientBoostingClassifier(random_state=RANDOM_STATE)
    ),
}

ensambles = {
    nombre: crear_stacking(
        modelos_base=[(m, modelos_base[m]) for m in modelos],
        meta_aprendiz=meta
    )
    for nombre, (modelos, meta) in combinaciones.items()
}
```

Nota: Código empleado para establecer las combinaciones de modelos base y meta-aprendices.

Figura 33

Validación cruzada y métricas de evaluación para seleccionar mejor ensamble

```
scoring = {
    "accuracy": make_scorer(accuracy_score),
    "balanced_accuracy": make_scorer(balanced_accuracy_score),
    "precision_macro": make_scorer(precision_score, average="macro", zero_division=0),
    "recall_macro": make_scorer(recall_score, average="macro", zero_division=0),
    "f1_macro": make_scorer(f1_score, average="macro", zero_division=0),
    "f1_weighted": make_scorer(f1_score, average="weighted", zero_division=0),
}

resultados = []

for nombre, ensamble in ensambles.items():
    print(f"Evaluando {nombre}...")

    cv_result = cross_validate(
        ensamble, X, y_encoded,
        cv=cv,
        scoring=scoring,
        n_jobs=-1,
        error_score="raise"
    )

    resultados.append({
        "Ensamble": nombre,
        "Accuracy": np.mean(cv_result["test_accuracy"]),
        "Balanced accuracy": np.mean(cv_result["test_balanced_accuracy"]),
        "Precision macro": np.mean(cv_result["test_precision_macro"]),
        "Recall macro": np.mean(cv_result["test_recall_macro"]),
        "F1-score macro": np.mean(cv_result["test_f1_macro"]),
        "F1-score weighted": np.mean(cv_result["test_f1_weighted"]),
    })

resultados_ensambles = (
    pd.DataFrame(resultados)
    .sort_values(["F1-score macro", "Balanced accuracy"], ascending=False)
    .reset_index(drop=True)
)

display(resultados_ensambles)

mejor_nombre = resultados_ensambles.loc[0, "Ensamble"]
mejor_ensamble = ensambles[mejor_nombre]

print(f"Mejor ensamble seleccionado: {mejor_nombre}")
```

Nota: El código calcula y compara las métricas de desempeño de los modelos ensamblados.

- **Evaluación con validación cruzada estratificada**

Esta fase corresponde a una primera evaluación, donde se comparó el desempeño de los modelos individuales y ensamblados mediante métricas de clasificación supervisada tras el entrenamiento con validación cruzada estratificada. Esta evaluación permitió identificar el modelo individual y el ensamble con mejor comportamiento.

Debido al desbalance entre los niveles de riesgo, se utilizó el F1-score macro como métrica principal para seleccionar el mejor modelo, ya que promedia el F1 de cada clase con igual peso, independientemente de su frecuencia (Sokolova & Lapalme, 2009). Esto permite valorar adecuadamente las clases minoritarias y evita que la evaluación dependa principalmente de la accuracy o del F1 ponderado, métricas sensibles al predominio de las clases mayoritarias (He & Garcia, 2009). La evaluación se complementó con la Balanced Accuracy, para medir la capacidad promedio de reconocimiento de las clases (Brodersen et al., 2010), así como con la precisión y el recall por categoría, la desviación estándar y la matriz de confusión. Además, se consideró el coeficiente Kappa para medir la concordancia entre las clases observadas y predichas, descontando el acuerdo atribuible al azar (Cohen, 1960). La naturaleza categórica de las variables influyó en su codificación y preprocesamiento, pero no en la elección de la métrica. De manera referencial se incluyeron también otras métricas en el desarrollo de la investigación.

2.9.5 Validación del mejor ensamble de algoritmos.

- **Evaluación mediante predicciones fuera de pliegue**

En esta etapa se identificó el modelo individual y el ensamble stacking con mejor desempeño, considerando como criterio principal el mayor F1-score macro promedio obtenido mediante validación cruzada en la fase anterior. Posteriormente, ambos modelos fueron evaluados mediante la función “cross_val_predict”, utilizando el mismo esquema de validación cruzada estratificada.

Esta función generó una predicción fuera de pliegue para cada registro, de modo que cada observación fue clasificada por un modelo entrenado sin incluirla en el respectivo conjunto de entrenamiento. A partir de estas predicciones se calcularon nuevamente la accuracy, la balanced accuracy, la precisión macro, el recall macro, el F1-score macro, el F1 ponderado y el coeficiente Kappa de Cohen. Asimismo, se construyeron las matrices de confusión para analizar el comportamiento de los modelos en cada nivel de riesgo e identificar los principales errores de clasificación.

- Evaluación complementaria mediante soft voting ponderado

Se incorporó una evaluación complementaria mediante soft voting ponderado (SV) para determinar si la combinación de probabilidades de los modelos individuales con mejor desempeño permitía superar al ensamble stacking seleccionado. Los modelos se ordenaron según el F1-score macro y se conformaron dos combinaciones: SV_1, con los dos mejor posicionados, y SV_2, con los tres mejor posicionados.

Se conservaron las configuraciones seleccionadas de los modelos base y sus pipelines de preprocesamiento. Los pesos se optimizaron mediante GridSearchCV con validación cruzada estratificada de diez particiones y F1 macro como criterio, reajustando los modelos dentro de cada partición. La comparación se realizó mediante las métricas obtenidas con cross_val_predict, diferenciando estos resultados de evaluación interna del rendimiento sobre la base completa utilizada para el ajuste final.

Tabla 41

Configuración de los ensambles adicionales mediante soft voting ponderado

Ensamble	Modelos incluidos	Asignación de pesos	Criterio de selección
SV_1	Primer y segundo mejor modelo individual	GridSearchCV	Mayor F1-score macro
SV_2	Primer, segundo y tercer mejor modelo individual	GridSearchCV	Mayor F1-score macro

Una vez aplicado el soft voting se analizaron las métricas y los resultados con la finalidad de validar el mejor ensamble. Es decir, primero se evaluó el mejor ensamble de los propuestos con stacking, y luego se evaluaron ensambles adicionales teniendo en cuenta los resultados preliminares para validar el mejor ensamble.

III RESULTADOS

3.1 Identificación de los niveles de peligro del sector La Colina

Tabla 42

Rangos de niveles de peligro del sector La Colina

Nivel de peligro	Rango
Peligro muy alto	[0.287 – 0.365]
Peligro alto	[0.220 – 0.287)
Peligro medio	[0.194 – 0.220)
Peligro bajo	[0.177 – 0.194)

Fuente: Elaborado a partir de los resultados obtenidos mediante la metodología CENEPRED (2014).

La tabla 42 presenta los rangos empleados para la clasificación de los niveles de peligro. Los valores se agruparon en cuatro categorías: peligro bajo, medio, alto y muy alto. El rango más bajo corresponde a valores entre 0.177 y 0.194, mientras que el rango más alto comprende valores entre 0.287 y 0.365.

Tabla 43

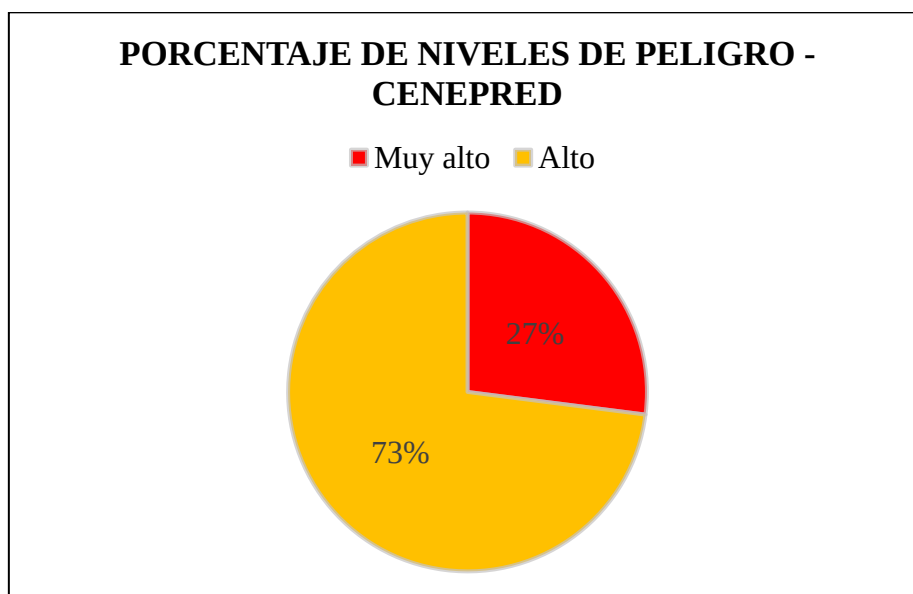
Peligro de deslizamientos por área de la zona de estudio - sector La Colina

Niveles	Área (hectáreas)	Porcentaje (%)
Muy alto	2.18	27%
Alto	5.89	73%
TOTAL	8.07	100%

La tabla 43 muestra la distribución de los niveles de peligro, se evidencia una mayor proporción de área afectada en la categoría de peligro alto y, en menor presencia, áreas afectadas en peligro muy alto. No se registraron áreas con peligro medio y bajo.

Figura 34

Distribución porcentual de niveles de peligro del área de estudio del sector La Colina

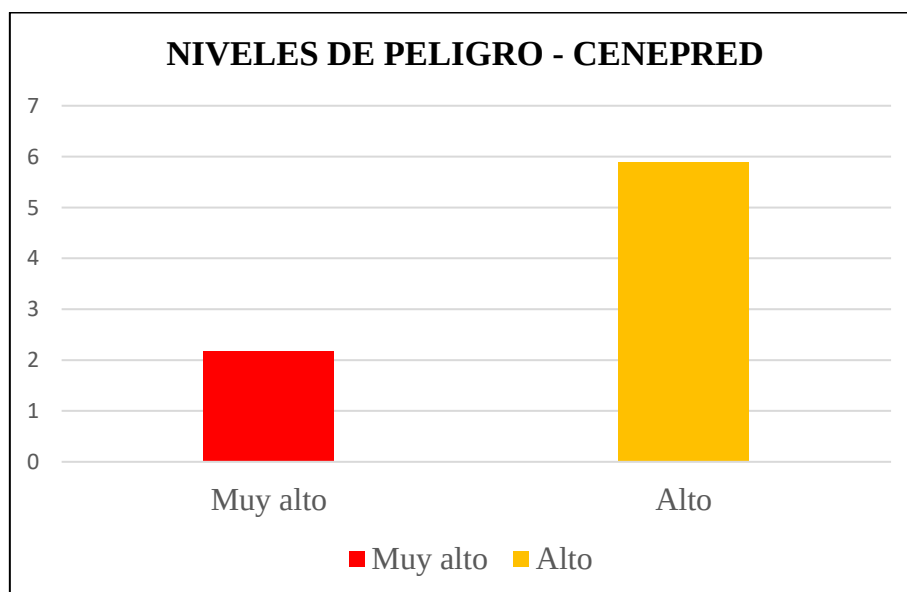


Nota: Niveles de peligro evaluados por la metodología del CENEPRED.

La figura 34 describe la distribución porcentual del peligro, donde, el 73% de la zona de estudio presenta peligro alto, y el 27% peligro muy alto.

Figura 35

Distribución de frecuencias de niveles de peligro del área de estudio del sector La Colina



Nota: Ninguna de las viviendas evaluadas estuvo en peligro bajo.

La figura 35 muestra que 5.89 hectáreas de la zona de estudio presenta un nivel de peligro alto, y 2.18 hectáreas en peligro muy alto.

3.2 Análisis de los niveles de vulnerabilidad del sector La Colina

Tabla 44

Rangos para la clasificación de niveles de vulnerabilidad en el sector La Colina

Nivel de vulnerabilidad	Rango
Vulnerabilidad muy alta	[0.268 – 0.494]
Vulnerabilidad alta	[0.132 – 0.268)
Vulnerabilidad media	[0.064 – 0.132)
Vulnerabilidad baja	[0.036 – 0.064)

Fuente: Elaborado a partir de los resultados obtenidos mediante la metodología CENEPRED (2014).

En la tabla 44 se presentan los rangos empleados para la clasificación de los niveles de vulnerabilidad. Los valores se agruparon en cuatro categorías: vulnerabilidad baja, media, alta y muy alta. El rango más bajo corresponde a valores entre 0.036 y 0.064, mientras que el rango más alto comprende valores entre 0.268 y 0.494.

Tabla 45

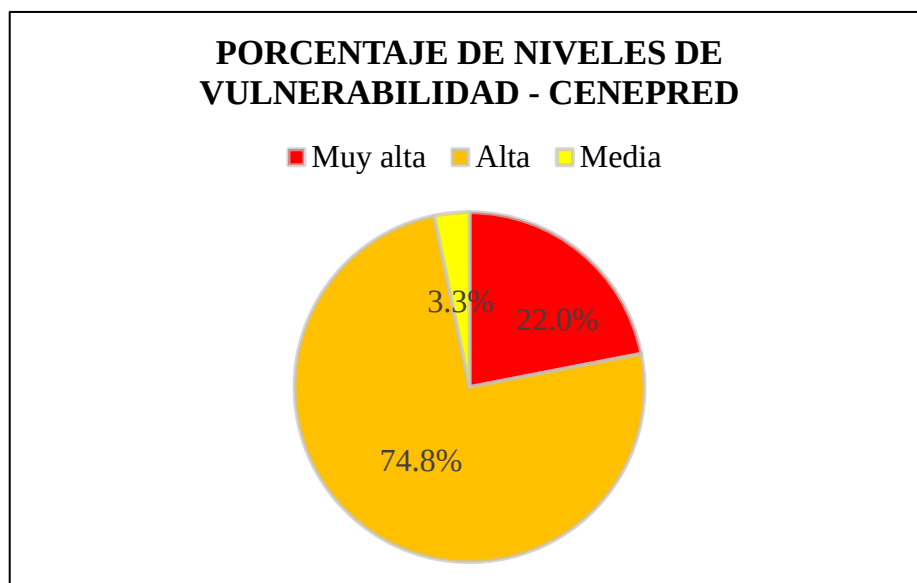
Distribución de los niveles de vulnerabilidad del sector La Colina

Niveles	Nº de viviendas	Porcentaje (%)
Muy alta	27	22.0%
Alta	92	74.8%
Media	4	3.3%
TOTAL	123	100%

La distribución de los niveles de vulnerabilidad, tabla 45, evidencia que la categoría alta concentra la mayor proporción de viviendas, seguida por la vulnerabilidad muy alta y, en menor presencia, por la vulnerabilidad media. No se registraron viviendas con vulnerabilidad baja.

Figura 36

Distribución porcentual de niveles de vulnerabilidad del área de estudio del sector La Colina

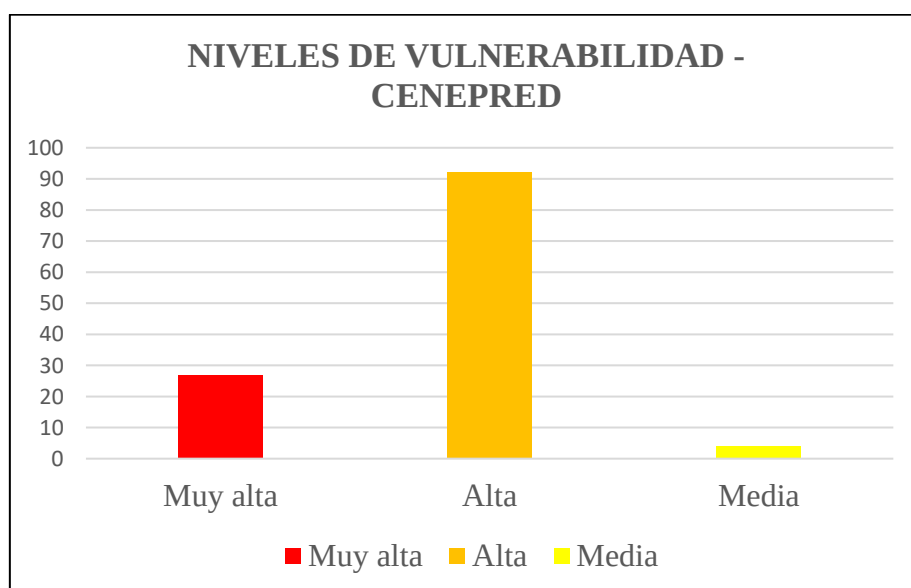


Nota: Niveles de vulnerabilidad evaluados por la metodología del CENEPRED.

La distribución porcentual de la figura 36 muestra que el 75% de las viviendas evaluadas presenta vulnerabilidad alta, el 22% vulnerabilidad muy alta y el 3% vulnerabilidad media.

Figura 37

Distribución de frecuencias de niveles de vulnerabilidad del sector La Colina



Nota: La evaluación no encontró viviendas en el nivel bajo de vulnerabilidad.

La distribución de frecuencias de la figura 37 muestra que 92 viviendas se ubican en el nivel de vulnerabilidad alta, 27 viviendas en vulnerabilidad muy alta y 4 viviendas en vulnerabilidad media.

3.3 Establecimiento y modelamiento de los niveles de riesgo

- Riesgo por metodología CENEPRED

Tabla 46

Rangos para la clasificación de niveles de riesgo en el sector La Colina

Nivel de riesgo	Rango
Riesgo muy alto	[0.077 – 0.180]
Riesgo alto	[0.029 – 0.077]
Riesgo medio	[0.012 – 0.029]
Riesgo bajo	[0.006 – 0.012]

Fuente: Elaborado a partir de los resultados obtenidos mediante la metodología CENEPRED (2014).

En la tabla 46 muestra los rangos empleados para la clasificación de los niveles de riesgo. Los valores se agruparon en cuatro categorías: riesgo bajo, medio, alto y muy alto. El rango más bajo corresponde a valores entre 0.006 y 0.012, mientras que el rango más alto comprende valores entre 0.077 y 0.18.

Tabla 47

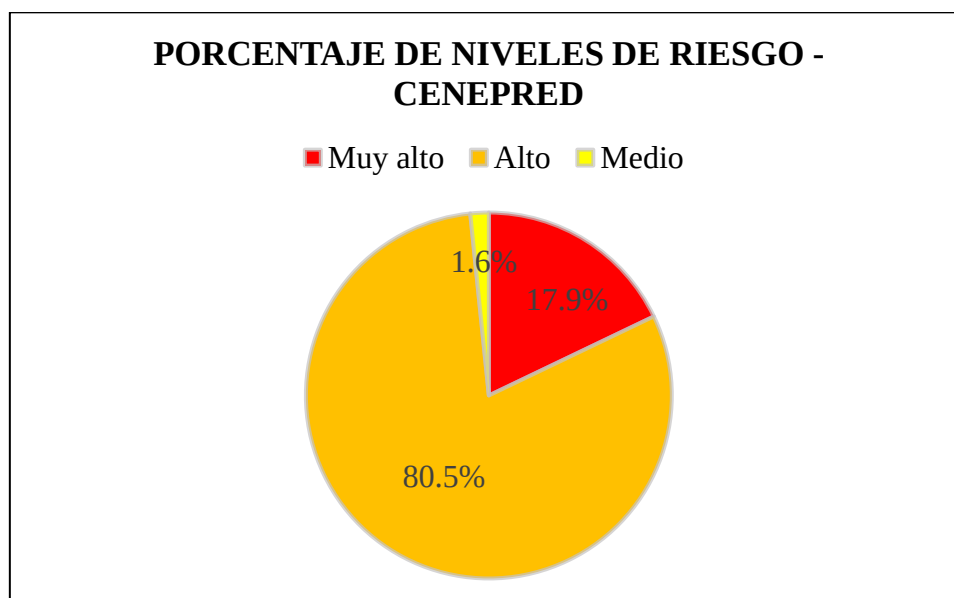
Distribución de los niveles de riesgo del sector La Colina

Niveles	N° de viviendas	Porcentaje (%)
Muy alta	22	17.9%
Alta	99	80.5%
Media	2	1.6%
TOTAL	123	100%

En la tabla 47 se observa que el riesgo alto agrupa la mayor cantidad de viviendas evaluadas. En segundo lugar, se presenta el riesgo muy alto, mientras que el riesgo medio aparece con una presencia mínima. No se identificaron viviendas en el nivel de riesgo bajo.

Figura 38

Distribución porcentual de niveles de riesgo del sector La Colina

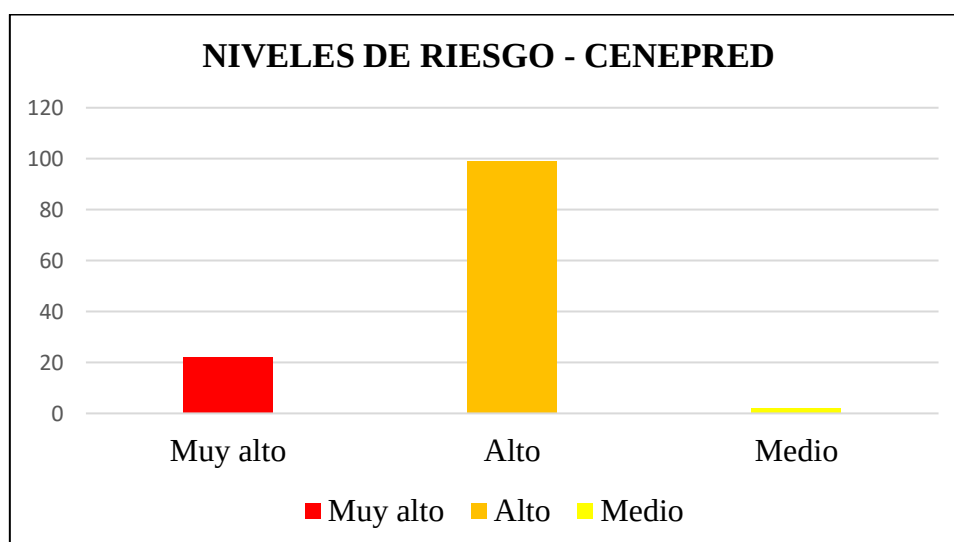


Nota: Niveles de riesgo evaluados con la metodología CENEPRED.

De acuerdo a la figura 38, el 80.5% de las viviendas corresponde al nivel de riesgo alto, el 17.9% al nivel de riesgo muy alto y el 1.6% al nivel de riesgo medio.

Figura 39

Distribución de frecuencias de niveles de riesgo del sector La Colina



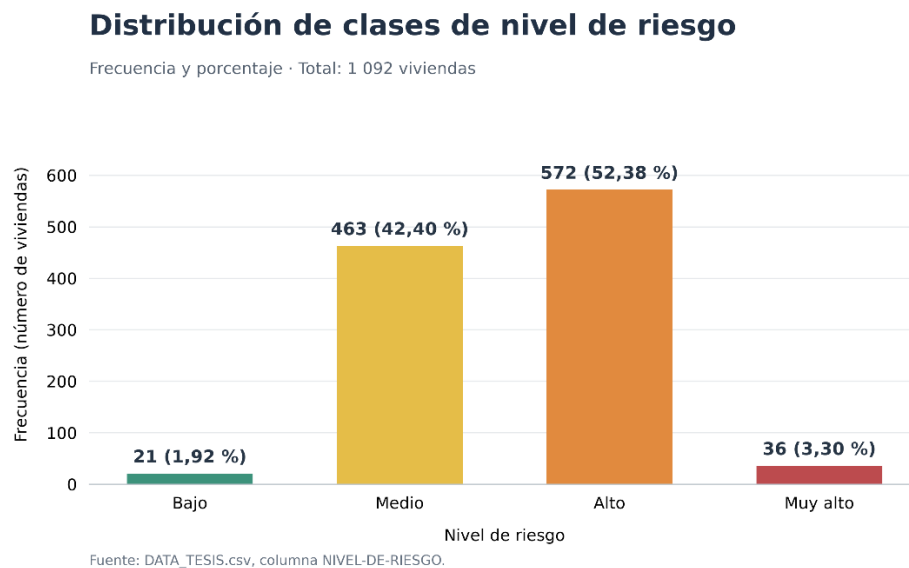
Nota: El nivel bajo no fue encontrado en las viviendas evaluadas.

Según la figura 39, gráfico de distribución de frecuencias, 99 viviendas se encuentran clasificadas con riesgo alto, 22 viviendas con riesgo muy alto y 2 viviendas con riesgo medio.

- Modelamiento del riesgo con algoritmos ensamblados

Figura 40

Distribución de los niveles de riesgo de la base de datos utilizada



Nota: Se observa en la figura que los registros de viviendas evaluadas están desbalanceados.

La figura 40 muestra la distribución de la variable objetivo de la base de datos con la que se entrenaron los modelos, la mayor concentración de registros estuvo en los niveles alto y medio. En menor proporción se registraron los niveles muy alto y bajo.

Tabla 48

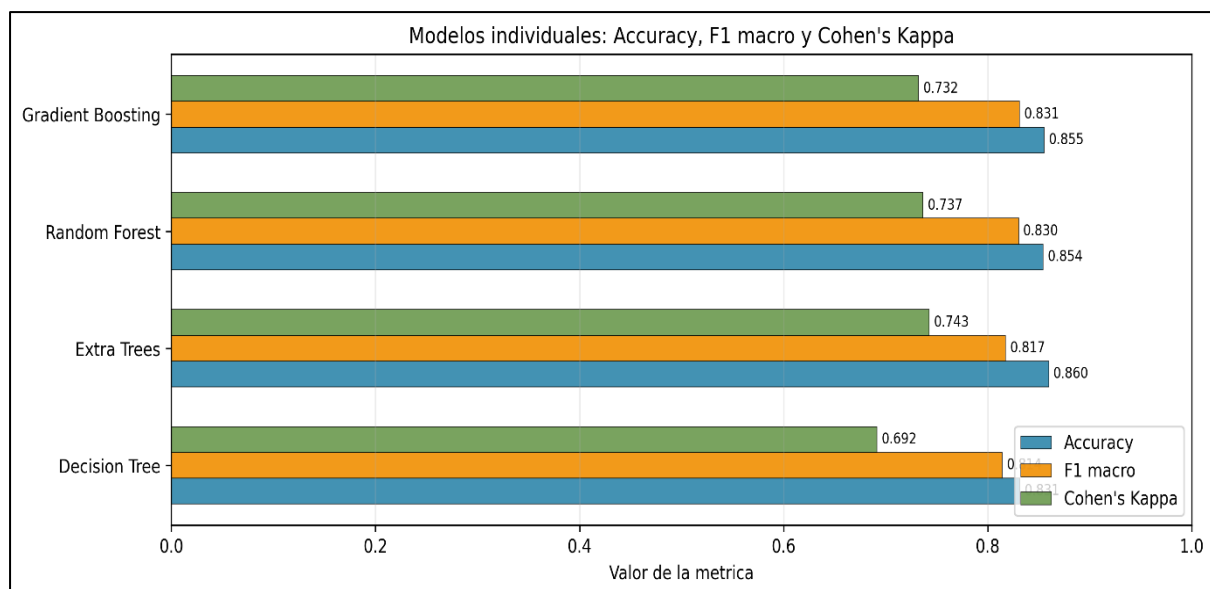
Rendimiento de los modelos individuales mediante validación cruzada de 10 particiones

Modelo	Accuracy (%)	Balanced Accuracy (%)	Precision Macro (%)	F1-score Macro (%)	Cohen's Kappa
Gradient Boosting	85.53	80.65	88.98	83.11	0.732
Random Forest	85.44	85.58	83.07	83.03	0.737
Extra Trees	85.99	80.92	86.88	81.74	0.743
Decision Tree	83.15	81.67	84.68	81.42	0.692

La Tabla 48 presenta los resultados de los modelos individuales evaluados mediante validación cruzada. Extra Trees obtuvo la mayor exactitud, con 85.99 %, y el mayor coeficiente Kappa, con 0.743; Random Forest alcanzó el mayor balanced accuracy, con 85.58 %; y Gradient Boosting registró la mayor precisión macro, con 88.98 %, y el mayor F1-score macro, con 83.11 %. Decision Tree mostró el menor desempeño general. Según el F1-score macro, Gradient Boosting fue seleccionado como el mejor modelo individual.

Figura 41

Comparación de las métricas de evaluación de los modelos individuales



Nota: Los valores corresponden al desempeño obtenido por cada modelo individual.

La figura 41 muestra una comparación de las principales métricas de evaluación, se observa que en Accuracy, Extra Trees con 0.86 presentó mejor desempeño, en la métrica F1 macro el mejor resultado fue de Gradient Boosting con 0.831 y en la métrica del Kappa de Cohen, Extra Trees tuvo mejor rendimiento con 0.743. La métrica principal fue F1 macro, por ello se identifica a Gradient Boosting como el mejor modelo individual, seguido por Random Forest y Extra Trees.

Tabla 49

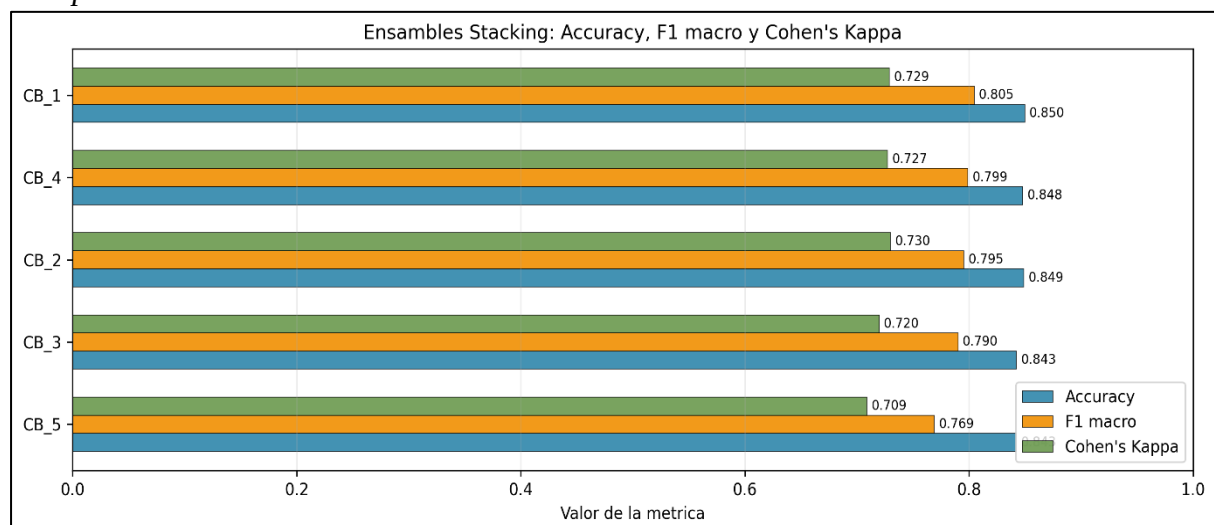
Rendimiento de los modelos ensamblados mediante validación cruzada de 10 particiones

Ensamble	Modelos de primer nivel	Meta-modelo	Accuracy (%)	Balanced Accuracy (%)	Precision Macro (%)	F1-score Macro (%)	Cohen's Kappa
CB_1	DT + RF + ET + GB	Regresión logística	84.98	85.07	79.46	80.48	0.729
CB_2	RF + ET + GB	Regresión logística	84.89	86.96	77.19	79.54	0.73
CB_3	RF + ET	Regresión logística	84.25	87.44	75.83	79.03	0.72
CB_4	RF + GB	Regresión logística	84.8	86.54	77.83	79.89	0.727
CB_5	DT + RF + ET + GB	Gradient Boosting	84.25	75.21	82.37	76.92	0.709

La Tabla 49 muestra que los ensambles stacking alcanzaron valores de exactitud entre 84.25 % y 84.98 %. El ensamble CB_1 obtuvo la mayor exactitud, con 84.98 %, y el mayor F1-score macro, con 80.48 %, por lo que fue seleccionado como el ensamble de mejor rendimiento. Asimismo, CB_3 alcanzó el mayor balanced accuracy, con 87.44 %, mientras que CB_2 presentó el coeficiente Kappa más elevado, con 0.730. En contraste, CB_5 registró el menor F1-score macro, con 76.92 %, y el menor coeficiente Kappa, con 0.709.

Figura 42

Comparación de las métricas de evaluación de los modelos ensamblados



Nota: Los valores corresponden al desempeño obtenido por cada configuración de stacking.

El gráfico de la figura 42 muestra que CB_1 presentó el mejor desempeño general al alcanzar la mayor accuracy (0.850) y el mayor F1-score macro (0.805), mientras que CB_2 obtuvo el coeficiente Kappa más alto (0.730). En contraste, CB_5 registró los valores más bajos de F1-score macro (0.769) y Kappa (0.709), por lo que CB_1 se considera el ensamble con el rendimiento más equilibrado.

3.4 Validación del mejor ensamble de algoritmos

El mejor modelo individual del entrenamiento con validación cruzada de la fase anterior fue el Gradient Boosting, y el mejor ensamble el CB-1, en este punto se hace un análisis posterior de los seleccionados con la función “cross_val_predict”.

Tabla 50

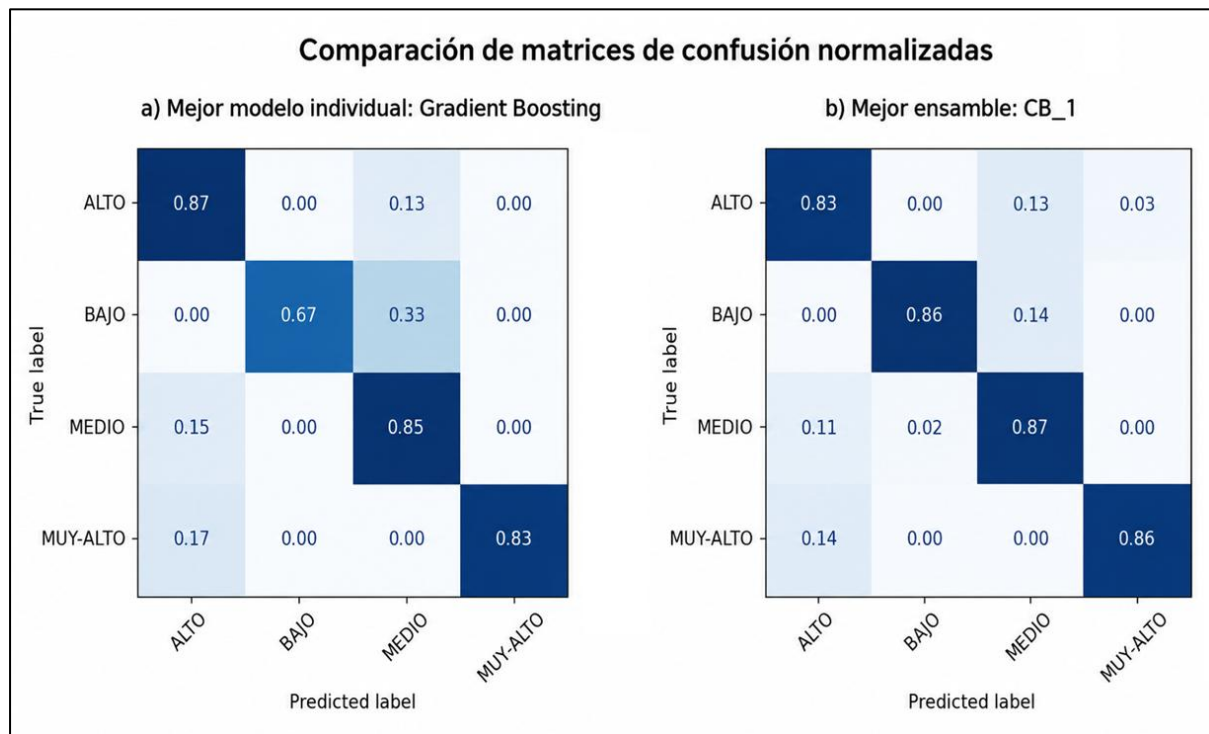
Métricas de la evaluación final con cross_val_predict

Categoría	Modelo	Accuracy (%)	Balanced Accuracy (%)	Precision Macro (%)	Recall Macro (%)	F1 Macro (%)	F1 Weighted (%)	Cohen's Kappa (%)
Modelo individual	Gradient Boosting	85.53	80.44	87.84	80.44	83.71	85.52	73.27
Ensamble	CB_1	84.98	85.52	76.64	85.52	80.41	85.09	72.91

La tabla 50 muestra que Gradient Boosting presentó el mejor rendimiento general, con mayor accuracy (85.53 %), precisión macro (87.84 %), F1 macro (83.71 %), F1 weighted (85.52 %) y coeficiente Kappa (73.27 %). En contraste, el ensamble CB_1 alcanzó mayor balanced accuracy y recall macro (85.52 %), lo que evidencia una identificación más equilibrada de las clases. Sin embargo, al considerar el F1 macro como métrica principal, Gradient Boosting fue seleccionado como el modelo de mejor desempeño.

Figura 43

Matrices de confusión de Gradient Boosting y el ensemble CB_1



Nota: Los valores representan proporciones normalizadas por clase verdadera.

La figura 43 muestra las matrices de confusión normalizadas evidenció que Gradient Boosting alcanzó un mayor porcentaje de clasificación correcta en la categoría ALTO, con 87 %, mientras que el ensemble CB_1 obtuvo mejores resultados en las categorías BAJO, MEDIO y MUY ALTO, con 86 %, 87 % y 86 %, respectivamente. Estos resultados muestran que el ensemble presentó una clasificación más equilibrada entre las categorías, aunque Gradient Boosting obtuvo el mayor F1-score macro general.

Como evaluación complementaria, se conformaron dos ensambles mediante soft voting ponderado, SV_1 de Gradient Boosting con Random Forest, y SV_2 de Gradient Boosting+Random Forest+Extra Trees.

Tabla 51

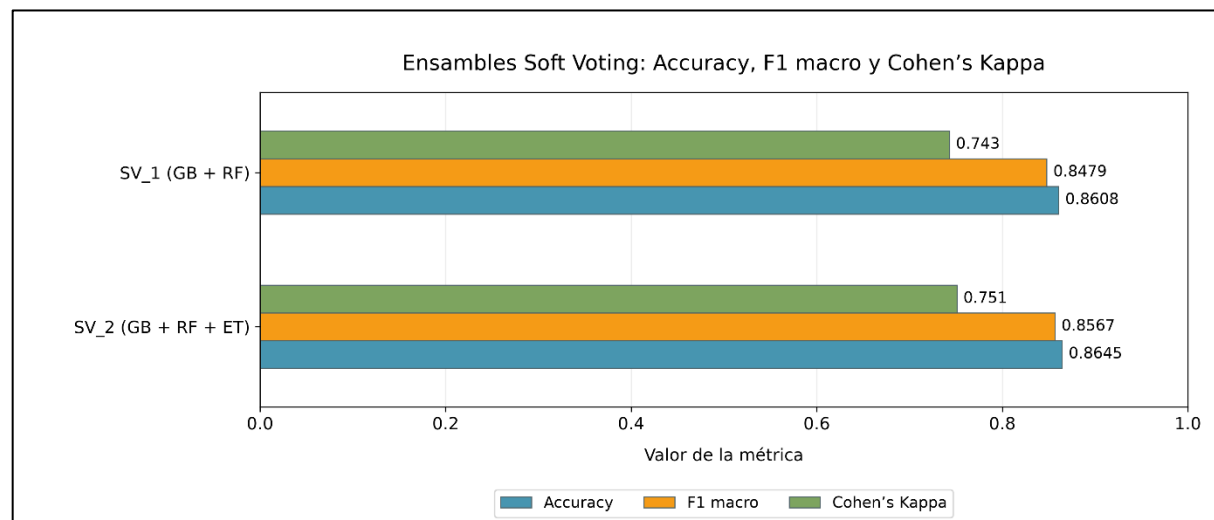
Métricas finales de los ensambles soft voting con la función “cross_val_predict”

	Ensamble	Accuracy (%)	Balanced Accuracy (%)	Precision Macro (%)	F1 Macro (%)	F1 Weighted (%)	Cohen's Kappa
SV_2	Soft Voting: GB+RF+ET	86.45	84.41	87.05	85.67	86.46	0.751
SV_1	Soft Voting: GB+RF	86.08	81.87	88.32	84.79	86.08	0.743

La tabla 51 muestra que el ensamble SV_2 alcanzó el mejor rendimiento general, con una exactitud de 86.45 %, un F1-score macro de 85.67 % y un coeficiente Kappa de 0.751. Aunque SV_1 obtuvo una mayor precisión macro, con 88.32 %, presentó valores inferiores en balanced accuracy y F1-score macro.

Figura 44

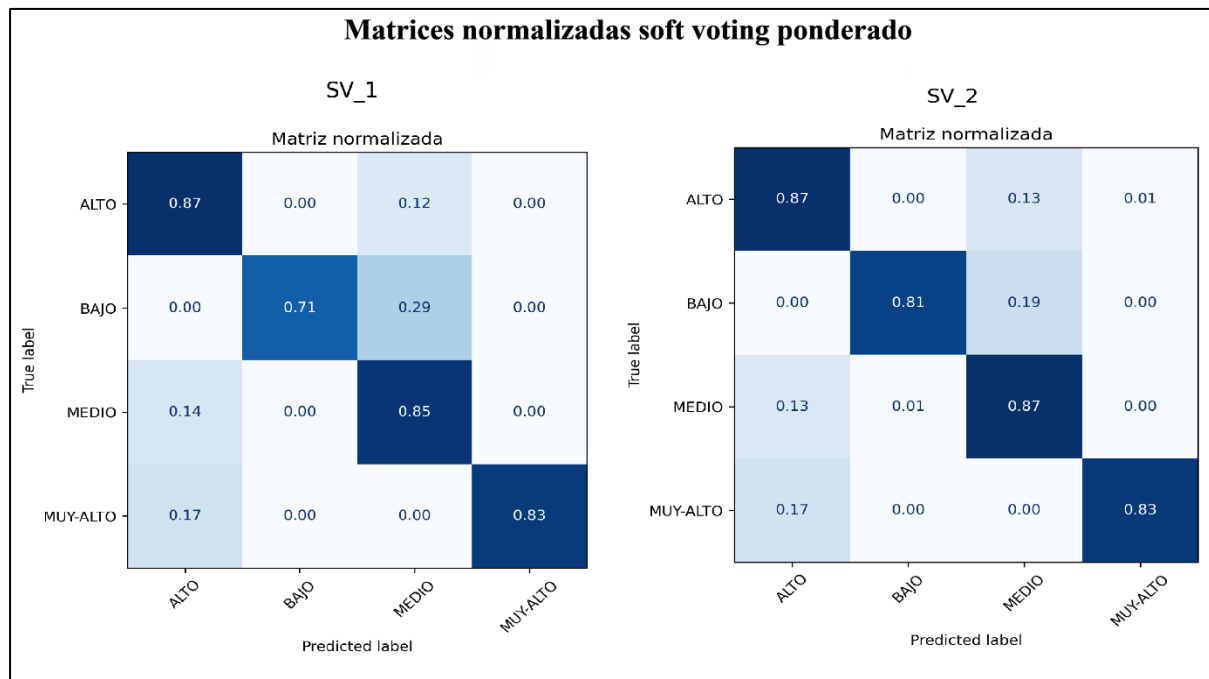
Comparación de ensambles soft voting con Accuracy, F1 macro, y Cohen's Kappa



El gráfico de la figura 44 muestra valores más altos de las métricas de F1 macro (0.8567), Accuracy (0.8645), y Cohen's Kappa (0.751) para el SV_2, superando a las métricas del ensamble SV_1.

Figura 45

Matrices de confusión de los ensambles soft voting ponderado



Nota: Las matrices de confusión normalizadas muestran el desempeño comparativo de los ensambles realizados con soft voting ponderado.

Las matrices normalizadas de la Figura 45 muestran que el ensamble SV_2 alcanzó porcentajes de clasificación correcta de 87 %, 81 %, 87 % y 83 % para los niveles alto, bajo, medio y muy alto, respectivamente. SV_2 presentó un desempeño más equilibrado, debido a que los porcentajes de clasificación correcta fueron relativamente similares entre las cuatro categorías, con una diferencia máxima de 6 puntos porcentuales entre la clase con mayor y menor porcentaje de aciertos. Asimismo, SV_2 obtuvo una balanced accuracy de 84.41 % y un F1-score macro de 85.67 %.

IV DISCUSIÓN

Los resultados del peligro en el sector La Colina mostraron una situación desfavorable, pues el 73% del área evaluada presentó peligro alto y el 27% peligro muy alto, sin identificarse niveles medios ni bajos. Estos resultados guardaron relación con Condorhuacho y Suni (2023), quienes identificaron 115 lotes con peligro muy alto y 57 con peligro alto en la quebrada Saqramayo, aunque también registraron categorías inferiores. Asimismo, Muñoz y Solís (2021) señalaron que el 32% de su zona de estudio presentó peligro muy alto ante deslizamientos, mientras que Tirado (2020) encontró un 68% del área con peligro muy alto y un 32% con peligro alto. La Colina coincidió con este último estudio en la presencia exclusiva de las categorías superiores, aunque predominó el peligro alto. Una posible explicación es la interacción entre las pendientes pronunciadas, los cortes de talud, la erosión y las lluvias intensas mencionadas en el diagnóstico. Los cortes pueden alterar el equilibrio de las laderas, mientras que la infiltración del agua aumenta la humedad y puede reducir la resistencia del suelo, facilitando su desplazamiento, como describe Ordóñez (2019). Por tanto, el resultado podría responder a la actuación conjunta de las condiciones del terreno y la precipitación; las diferencias entre estudios también deben interpretarse considerando sus ámbitos, unidades de análisis y criterios de clasificación.

Los resultados de vulnerabilidad mostraron que el 74.8% de las viviendas de La Colina presentó vulnerabilidad alta, el 22.0% muy alta y el 3.3% media, sin registrarse vulnerabilidad baja. Esta distribución fue semejante a la encontrada por Condorhuacho y Suni (2023), quienes identificaron 125 lotes con vulnerabilidad alta y 38 con vulnerabilidad muy alta. Sin embargo, difirió de Muñoz y Solís (2021), donde el 55% de la zona evaluada presentó vulnerabilidad muy alta, y de Tirado (2020), quien reportó un 78% de viviendas en vulnerabilidad muy alta y un 22% en vulnerabilidad alta. El predominio del nivel alto podría explicarse por la

coexistencia de condiciones desfavorables de exposición, fragilidad y resiliencia, sin que todas las viviendas alcancen simultáneamente los valores más críticos de estos componentes. Según el Centro Nacional de Estimación, Prevención y Reducción del Riesgo de Desastres (CENEPRED, 2014), las deficiencias físicas aumentan la susceptibilidad al daño, mientras que una menor capacidad de respuesta y recuperación incrementa la vulnerabilidad. En ese sentido, Tirado (2020) relacionó sus resultados con materiales de mala calidad y deficiencias estructurales. Esta comparación sugiere que las diferencias entre sectores pueden depender tanto de las características constructivas como de las capacidades de sus habitantes; por ello, una menor proporción de vulnerabilidad muy alta no implica condiciones suficientes de seguridad.

Los resultados del riesgo por deslizamiento mediante CENEPRED mostraron que, de las 123 viviendas evaluadas en La Colina, el 80.5% presentó riesgo alto, el 17.9% muy alto y el 1.6% medio, sin registrarse riesgo bajo. El predominio del riesgo alto coincidió con las conclusiones de Muñoz y Yamunaque (2024), quienes identificaron 47 lotes con riesgo alto y 20 con riesgo medio en Magllanal; esta coincidencia se limita a la distribución de categorías, pues evaluaron inundaciones fluviales. En cambio, Condorhuacho y Suni (2023) reportaron 109 lotes con riesgo muy alto y 61 con riesgo alto; Muñoz y Solís (2021) encontraron un 63% del predio con riesgo muy alto, y Tirado (2020) identificó un 76% de viviendas en riesgo muy alto y un 24% en riesgo alto. En La Colina, la concentración del riesgo en el nivel alto es compatible con el predominio del peligro y la vulnerabilidad en esa misma categoría. Sin embargo, su distribución final depende de cómo coinciden ambos componentes en cada vivienda y de los umbrales utilizados, por lo que los porcentajes generales no permiten atribuir el resultado exclusivamente a mejores materiales o condiciones de conservación. Así, las diferencias con los antecedentes pueden relacionarse con la localización de las viviendas y la intensidad de sus condiciones de vulnerabilidad (CENEPRED, 2014).

El modelamiento del riesgo por deslizamiento mostró un desempeño favorable de los algoritmos basados en árboles. Gradient Boosting obtuvo el mayor F1-score macro entre los modelos individuales, con 83.11%, mientras que Extra Trees alcanzó la mayor exactitud, con 85.99%, y el mayor coeficiente Kappa, con 0.743. Este comportamiento puede relacionarse con la capacidad de los árboles para representar interacciones y relaciones no lineales entre las variables evaluadas. En particular, Gradient Boosting incorpora árboles sucesivos orientados a reducir los errores del modelo acumulado, conforme al planteamiento de Friedman (2001). Los resultados coincidieron parcialmente con Zhang et al. (2024), quienes destacaron el desempeño de Random Forest y XGBoost en la evaluación de susceptibilidad a deslizamientos. Entre las combinaciones mediante stacking, CB_1 alcanzó 84.98% de exactitud y 80.48% de F1-score macro, sin superar los mejores valores individuales de estas métricas. Esto contrastó con Palomino et al. (2024), cuyos mejores ensambles superaron ligeramente al mejor modelo individual en vulnerabilidad sísmica. Una posible explicación es que los clasificadores de este estudio cometieran errores similares, reduciendo la información complementaria disponible para el metaclassificador; además, la menor representación de ciertas categorías pudo dificultar su aprendizaje. Por ello, combinar modelos no garantiza una mejora: también importa que sus predicciones aporten información complementaria.

La evaluación mediante soft voting ponderado mostró que SV_2 alcanzó 86.45 % de exactitud, 84.41 % de balanced accuracy, 85.67 % de F1-score macro y un Kappa de 0.751. Además, presentó porcentajes de clasificación correcta de 87 %, 81 %, 87 % y 83 % en los niveles alto, bajo, medio y muy alto, respectivamente, con una diferencia máxima de seis puntos porcentuales, lo que evidencia un desempeño relativamente equilibrado entre las categorías. Este resultado puede explicarse porque la combinación ponderada de probabilidades permite compensar errores particulares de los clasificadores. La mejora guarda

relación con Zhang et al. (2024), quienes destacaron el desempeño de Random Forest y XGBoost; con Palomino et al. (2024), quienes obtuvieron mejoras mediante stacking; y con Chen et al. (2024), quienes señalaron la importancia del diseño del modelo y la calidad de los datos. En esta investigación, SV_2 presentó el mejor desempeño entre las alternativas evaluadas. Sin embargo, los resultados corresponden únicamente a una validación interna mediante validación cruzada, por lo que su capacidad de generalización deberá comprobarse con datos independientes de otros sectores o periodos.

V CONCLUSIONES Y RECOMENDACIONES

5.1 Conclusiones

El sector La Colina presenta una condición general de riesgo elevado por deslizamiento, con predominio del nivel alto. La aplicación de algoritmos ensamblados permitió integrar las características asociadas al peligro y la vulnerabilidad y estimar de manera consistente los diferentes niveles de riesgo presentes en el área de estudio. El estudio demuestra la aplicabilidad de los algoritmos ensamblados como herramienta complementaria para la estimación del riesgo por deslizamiento en el sector La Colina, Jaén.

La zona de estudio del sector La Colina presenta niveles de peligro alto y muy alto ante deslizamientos. El nivel que más predomina es el peligro alto, con 5.89 hectáreas de área, lo que representa el 73% del área evaluada; mientras que el peligro muy alto abarca 2.18 hectáreas de área, equivalente al 27%. Por ello, se concluye que el sector presenta condiciones físicas y ambientales que pueden favorecer la inestabilidad de las laderas, sobre todo en zonas con pendientes pronunciadas, cortes de talud, erosión, suelo expuesto y presencia de lluvias intensas.

De las 123 viviendas evaluadas en la zona de estudio del sector La Colina, la mayoría presenta una condición de vulnerabilidad alta. En total, 92 viviendas se ubican en este nivel, lo que equivale al 74.8%; mientras que 27 viviendas presentan vulnerabilidad muy alta, equivalente al 22.0%, y solo 4 viviendas se encuentran en vulnerabilidad media, con el 3.3%. Además, no se registraron viviendas con vulnerabilidad baja. Por ello, se concluye que la población y las viviendas evaluadas presentan condiciones sociales, económicas, constructivas y ambientales que pueden aumentar su susceptibilidad frente a un posible deslizamiento.

El nivel de riesgo por deslizamiento en el sector La Colina, según la metodología CENEPRED, es principalmente alto. De las 123 viviendas evaluadas, 99 se encuentran en este nivel, lo que representa el 80.5%; además, 22 viviendas presentan riesgo muy alto, equivalente

al 17.9%, y solo 2 viviendas se ubican en riesgo medio, con el 1.6%. No se identificaron viviendas con riesgo bajo. Por ello, se concluye que el sector presenta una condición general de riesgo elevado, debido a la presencia de peligro alto y muy alto, junto con condiciones de vulnerabilidad que también son mayormente altas y muy altas.

El riesgo por deslizamiento se modeló mediante cuatro algoritmos individuales y cinco ensambles stacking, determinándose que Gradient Boosting fue el mejor modelo individual, con un F1 score macro de 83.11 %, mientras que CB_1 fue el stacking de mejor rendimiento, con 84.98 % de exactitud y 80.48 % de F1 score macro. Sin embargo, al no superar el ensamble al modelo individual en la métrica principal, se concluye que la combinación mediante stacking no produjo una mejora suficiente para la clasificación de los niveles de riesgo, debido a que la base de datos presentó desbalance entre categorías y los modelos integrados mostraron un comportamiento predictivo similar.

La validación permitió determinar que el ensamble SV_2 fue el modelo final de mejor desempeño, al alcanzar 86.45 % de exactitud, 84.41 % de balanced accuracy, 85.67 % de F1 score macro y un coeficiente Kappa de 0.751. En comparación con Gradient Boosting, SV_2 incrementó la exactitud en 0.92 puntos porcentuales, el F1 score macro en 1.96 puntos porcentuales, la balanced accuracy en 3.97 puntos porcentuales y el Kappa en aproximadamente 0.018. En consecuencia, se concluye que el soft voting ponderado proporcionó la clasificación más precisa y equilibrada de los cuatro niveles de riesgo por deslizamiento, por lo que constituye el ensamble final seleccionado.

5.2 Recomendaciones

Considerar la implementación progresiva de medidas estructurales y no estructurales en las zonas de peligro muy alto, realizar evaluaciones geotécnicas, análisis de estabilidad de taludes y revisión del drenaje pluvial. Asimismo, evaluar la conveniencia de limitar nuevas ocupaciones en los sectores de mayor criticidad.

Atender las viviendas con vulnerabilidad alta y muy alta mediante asistencia técnica constructiva, capacitación en gestión del riesgo y prevención de prácticas que deterioren el suelo, complementando estas acciones con mejoras en los servicios básicos y el fortalecimiento de la organización comunitaria en La Colina.

Priorizar la intervención integral en viviendas con riesgo alto y muy alto, articulando medidas de reducción del peligro y la vulnerabilidad. Incorporar los resultados de la investigación en la planificación urbana, el control de nuevas ocupaciones en laderas y las acciones preventivas de las autoridades locales en La Colina.

Ampliar las investigaciones mediante la comparación de técnicas y modelos de evaluación del riesgo, incorporando bases de datos con mayor detalle de variables cuantitativas. Evaluar distintas combinaciones de algoritmos de ensamblaje para determinar su desempeño y aplicabilidad en la zona de estudio.

A partir de la validación interna realizada en el presente estudio, se propone explorar nuevas combinaciones de algoritmos mediante ensambles de *soft voting* ponderado y evaluar la combinación SV_2 en otras zonas susceptibles a deslizamientos, con el fin de comprobar su desempeño y su capacidad para estimar los niveles de riesgo. Esta evaluación deberá incluir una comparación con Gradient Boosting para determinar si el ensamble mantiene su ventaja frente al modelo individual. Dado que el alcance actual se limita a una validación interna, queda pendiente contrastar ambos modelos con bases de datos externas para evaluar su capacidad de generalización.

VI REFERENCIAS BIBLIOGRÁFICAS

- Aguarón, J., & Moreno-Jiménez, J. M. (2003). The geometric consistency index: Approximated thresholds. *European Journal of Operational Research*, 147(1), 137–145. [https://doi.org/10.1016/S0377-2217\(02\)00255-2](https://doi.org/10.1016/S0377-2217(02)00255-2)
- Alfaro Lozano, L. (2014). *Estimación de umbrales de precipitaciones extremas para la emisión de avisos meteorológicos* (Nota Técnica N.º 001-SENAMHI-DGM-2014). Servicio Nacional de Meteorología e Hidrología del Perú.
- Anchelía, D., & Mori, X. (2020). *Determinación de zonas susceptibles a inundaciones y análisis comparativo del proceso de análisis jerárquico (AHP) y Random Forest (RF). Caso de estudio: Cuenca baja del río Chancay-Lambayeque* [Tesis de pregrado, Universidad Nacional Mayor de San Marcos]. Cybertesis. <https://hdl.handle.net/20.500.12672/15868>
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324>
- Breiman, L., Friedman, J. H., Olshen, R. A., & Stone, C. J. (1984). *Classification and regression trees*. Wadsworth International Group.
- Brodersen, K. H., Ong, C. S., Stephan, K. E., & Buhmann, J. M. (2010). The balanced accuracy and its posterior distribution. En *2010 20th International Conference on Pattern Recognition* (pp. 3121–3124). IEEE. <https://doi.org/10.1109/ICPR.2010.764>
- Bujang, M. A., Omar, E. D., Foo, D. H. P., & Hon, Y. K. (2024). Sample size determination for conducting a pilot study to assess reliability of a questionnaire. *Restorative Dentistry & Endodontics*, 49(1), e3. <https://doi.org/10.5395/rde.2024.49.e3>
- Centre for Research on the Epidemiology of Disasters. (2024). *2023 disasters in numbers: A significant year of disaster impact*. https://files.emdat.be/reports/2023_EMDAT_report.pdf

- Centro Nacional de Estimación, Prevención y Reducción del Riesgo de Desastres. (2014). *Manual para la evaluación de riesgos originados por fenómenos naturales* (2.^a versión). https://www.cenepred.gob.pe/web/wp-content/uploads/Guia_Manuales/Manual-Evaluacion-de-Riesgos_v2.pdf
- Chen, L., Ma, P., Fan, X., Wang, X., & Ng, C. W. W. (2024). A knowledge-aware deep learning model for landslide susceptibility assessment in Hong Kong. *Science of the Total Environment*, 941, 173557. <https://doi.org/10.1016/j.scitotenv.2024.173557>
- Chen, W., & Zhang, L. (2021). Predicting building damages in mega-disasters under uncertainty: An improved Bayesian network learning approach. *Sustainable Cities and Society*, 66, 102689. <https://doi.org/10.1016/j.scs.2020.102689>
- Chen, W., & Zhang, S. (2021). GIS-based comparative study of Bayes network, Hoeffding tree and logistic model tree for landslide susceptibility modeling. *Catena*, 203, 105344. <https://doi.org/10.1016/j.catena.2021.105344>
- Cohen, J. (1960). A coefficient of agreement for nominal scales. *Educational and Psychological Measurement*, 20(1), 37–46. <https://doi.org/10.1177/001316446002000104>
- Condorhuacho, R. F., & Suni, K. (2023). *Evaluación de riesgos por deslizamiento en la quebrada Saqramayo, en los distritos de Cusco, provincia y región del Cusco, 2020* [Tesis de licenciatura, Universidad Nacional de San Antonio Abad del Cusco]. Repositorio Institucional UNSAAC. <https://repositorio.unsaac.edu.pe/handle/20.500.12918/7338>
- Creswell, J. W. (2014). *Research design: Qualitative, quantitative, and mixed methods approaches* (4.^a ed.). SAGE Publications.
- Denzin, N. K., & Lincoln, Y. S. (Eds.). (2011). *The SAGE handbook of qualitative research* (4.^a ed.). SAGE Publications.

- Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. *The Annals of Statistics*, 29(5), 1189–1232. <https://doi.org/10.1214/aos/1013203451>
- Geurts, P., Ernst, D., & Wehenkel, L. (2006). Extremely randomized trees. *Machine Learning*, 63(1), 3–42. <https://doi.org/10.1007/s10994-006-6226-1>
- Gobierno Regional de Cajamarca. (2010). *Mapa de uso actual de la tierra* [Mapa, escala 1:250 000]. Zonificación Ecológica y Económica de la Región Cajamarca.
- Gobierno Regional de Cajamarca. (2012). *Mapa de geomorfología* [Mapa, escala 1:250 000]. Zonificación Ecológica y Económica de la Región Cajamarca.
- González-Bolaños, F., Olguín-López, J. L., Guevara-Gutiérrez, R. D., Palomera-García, C., Ramírez-Romero, J. M., & Villalvazo-López, V. M. (2023). Identificación espacial del riesgo por deslizamiento de tierra en el municipio de Autlán de Navarro, Jalisco, México. *Revista Geográfica de América Central*, 1(70), 349–376. <https://doi.org/10.15359/rgac.70-1.13>
- Goyes-Peñafiel, P., & Hernandez-Rojas, A. (2021). Landslide susceptibility index based on the integration of logistic regression and weights of evidence: A case study in Popayan, Colombia. *Engineering Geology*, 280, 105958. <https://doi.org/10.1016/j.enggeo.2020.105958>
- Grima, N., Edwards, D., Edwards, F., Petley, D., & Fisher, B. (2020). Landslides in the Andes: Forests can provide cost-effective landslide regulation services. *Science of the Total Environment*, 745, 141128. <https://doi.org/10.1016/j.scitotenv.2020.141128>
- Gumbel, E. J. (1958). *Statistics of Extremes*. Columbia University Press. <https://doi.org/10.7312/gumb92958>
- He, H., & Garcia, E. A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, 21(9), 1263–1284. <https://doi.org/10.1109/TKDE.2008.239>

- Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, M. del P. (2014). *Metodología de la investigación* (6.^a ed.). McGraw-Hill Education.
- Hussain, M. A., Chen, Z., Zheng, Y., Shoaib, M., Shah, S. U., Ali, N., & Afzal, Z. (2022). Landslide susceptibility mapping using machine learning algorithm validated by persistent scatterer InSAR technique. *Sensors*, 22(9), 3119. <https://doi.org/10.3390/s22093119>
- Idrogo Díaz, Y. E. (2023). *Niveles de riesgo por deslizamiento de tierra en la expansión urbana, sector 9 de la ciudad de Chota, aplicando la metodología del CENEPRED, 2021* [Tesis de grado, Universidad Nacional Autónoma de Chota]. Repositorio Institucional UNACH. <http://hdl.handle.net/20.500.14142/401>
- Instituto Geológico, Minero y Metalúrgico, Dirección de Geología Ambiental y Riesgo Geológico. (2024). *Reporte de monitoreo del deslizamiento de Higosbamba, Hichabamba, Huayllabamba y Churucana (Cajabamba), periodo 2022–2023. Distrito y provincia de Cajabamba, departamento de Cajamarca* [Informe técnico]. Repositorio Institucional del INGEMMET. <https://hdl.handle.net/20.500.12544/4995>
- Instituto Nacional de Defensa Civil. (2005). *Programa de prevención y medidas de mitigación ante desastres de la ciudad de Jaén*. https://bvpad.indeci.gob.pe/doc/estudios_CS/Region_cajamarca/jaen/jaen.pdf
- Instituto Nacional de Defensa Civil. (2019-2024). *Reportes preliminares, complementarios y de emergencias: Cajamarca*. Recuperado el 16 de julio de 2024, de <https://portal.indeci.gob.pe/informe/reportes-preliminares-complementarios-emergencias/>
- Instituto Nacional de Defensa Civil. (2021, 3 de junio). *Informe de emergencia N.º 824: Lluvias intensas en el distrito de Jaén, Cajamarca* (Informe N.º 6). Centro de Operaciones de Emergencia Nacional. <https://portal.indeci.gob.pe/emergencias/reporte-preliminar-n->

1110-8-4-2021-coen-indeci-0500-horas-lluvias-intensas-en-el-distrito-de-jaen-cajamarca/

International Organization for Standardization. (2018). *Security and resilience—Emergency management—Guidelines for implementation of a community-based landslide early warning system* (ISO Standard No. 22327:2018). <https://www.iso.org/standard/50064.html>

James, G., Witten, D., Hastie, T., & Tibshirani, R. (2021). *An introduction to statistical learning: With applications in R* (2.^a ed.). Springer. <https://doi.org/10.1007/978-1-0716-1418-1>

Koo, T. K., & Li, M. Y. (2016). A guideline of selecting and reporting intraclass correlation coefficients for reliability research. *Journal of Chiropractic Medicine*, 15(2), 155–163. <https://doi.org/10.1016/j.jcm.2016.02.012>

Lara, P., Bletery, Q., Ampuero, J.-P., Inza, A., & Tavera, H. (2023). Earthquake early warning starting from 3 s of records on a single station with machine learning. *Journal of Geophysical Research: Solid Earth*, 128(11), e2023JB026575. <https://doi.org/10.1029/2023JB026575>

López Vega, J. W., Torres Lázaro, J. C., & Herrera Quispe, J. (2021). Modelo predictivo basado en machine learning para la estimación de vulnerabilidades de riesgo de inundación y deslizamiento. Caso de estudio: Instituciones educativas del Perú. En *Actas del IV Congreso Internacional de Ingeniería de Sistemas* (pp. 206–207). Universidad de Lima. <https://doi.org/10.26439/ciis2021.5637>

McHugh, M. L. (2012). Interrater reliability: The kappa statistic. *Biochemia Medica*, 22(3), 276–282. <https://doi.org/10.11613/BM.2012.031>

Milla Pino, M. E., Rodríguez Ordoñez, F. R., Shimabuku Ysa, R. A., Jara Llanos, D. M., Torres Cruz, M. M., & Cayatopa Calderón, B. A. (2023). Role of expert in validation of

- information collection instruments for business purposes. *International Journal of Professional Business Review*, 8(8), e03122.
<https://doi.org/10.26668/businessreview/2023.v8i8.3122>
- Ministerio de Agricultura. (2009, 2 de septiembre). *Decreto Supremo N.º 017-2009-AG, que aprueba el Reglamento de Clasificación de Tierras por su Capacidad de Uso Mayor*. Diario Oficial *El Peruano*.
- Montgomery, D. C., & Runger, G. C. (2013). *Applied statistics and probability for engineers* (6.^a ed.). John Wiley & Sons.
- Municipalidad Distrital de Jangas. (2023). *Plan de prevención y reducción del riesgo de desastres del distrito de Jangas 2023–2030*. Sistema de Información para la Gestión del Riesgo de Desastres. <https://sigrid.cenepred.gob.pe/sigridv3/documento/17183>
- Municipalidad Provincial de Cusco. (2020). *Evaluación de riesgos originados por deslizamiento en el área de ubicación del proyecto “Mejoramiento y ampliación de la prestación del servicio de salud del Establecimiento de Salud de San Pedro APV Picchu Alto, distrito de Cusco, provincia de Cusco, Cusco”* [Informe técnico]. Sistema de Información para la Gestión del Riesgo de Desastres. <https://sigrid.cenepred.gob.pe/sigridv3/documento/10006>
- Muñoz, A., & Yamunaque, R. (2024). *Estimación del nivel de riesgo de inundación fluvial mediante el algoritmo Random Forest en el sector Magllanal, Jaén, Cajamarca, 2022* [Tesis de pregrado, Universidad Nacional de Jaén]. Repositorio Institucional UNJ. <https://repositorio.unj.edu.pe/handle/UNJ/692>
- Muñoz, G. A., & Solís, G. I. (2021). *Análisis de riesgos ante deslizamientos e inundaciones, utilizando la metodología del CENEPRED, para impulsar un desarrollo sostenible y planificado en la localidad de Izcuchaca, distrito y provincia de Anta, Cusco* [Tesis de

- licenciatura, Universidad Andina del Cusco]. Repositorio Digital UAC.
<https://hdl.handle.net/20.500.12557/4181>
- Naciones Unidas. (2023, 31 de mayo). *La IA puede impulsar los sistemas de alerta temprana*. Noticias ONU. <https://news.un.org/es/story/2023/05/1521512>
- Ordóñez, J. (2019). *Movimiento en masa por lluvias intensas en el Perú* (Documento técnico N.º 291). Servicio Nacional de Meteorología e Hidrología del Perú.
<https://hdl.handle.net/20.500.12542/291>
- Otzen, T., & Manterola, C. (2017). Técnicas de muestreo sobre una población a estudio. *International Journal of Morphology*, 35(1), 227–232. <https://doi.org/10.4067/S0717-95022017000100037>
- Palomino Ojeda, J. M., Quiñones Huatangari, L., Revilla Arce, J., Arce Fernández, N., Gonzales Santisteban, M. A., & Martínez Serrano, M. A. (2024). Assembly algorithms for seismic vulnerability estimation in confined masonry dwellings. *International Journal of Safety and Security Engineering*, 14(3), 967–984.
<https://doi.org/10.18280/ijssse.140327>
- Petley, D. (2012). Global patterns of loss of life from landslides. *Geology*, 40(10), 927–930.
<https://doi.org/10.1130/G33217.1>
- Sahin, E. K. (2020). Assessing the predictive capability of ensemble tree methods for landslide susceptibility mapping using XGBoost, gradient boosting machine, and random forest. *SN Applied Sciences*, 2, 1308. <https://doi.org/10.1007/s42452-020-3060-1>
- Sánchez Fernández, A. W., Dávila Manrique, D., & De la Cruz Bustamante, N. S. (1996). *Geología del cuadrángulo de Jaén, hoja 12-f* (Boletín N.º 62, Serie A: Carta Geológica Nacional). Instituto Geológico, Minero y Metalúrgico.
<https://hdl.handle.net/20.500.12544/184>

- Save the Children. (2024). *Respuesta humanitaria a la emergencia por deslizamiento en Arequipa*. <https://www.savethechildren.org.pe/proyectos/respuesta-humanitaria-a-la-emergencia-por-deslizamiento-en-arequipa/>
- scikit-learn. (s. f.). *f1_score*. https://scikit-learn.org/stable/modules/generated/sklearn.metrics.f1_score.html
- Servicio Nacional de Meteorología e Hidrología del Perú. (2014). *Umbrales y precipitaciones absolutas*. <https://web2.senamhi.gob.pe/pdf/clim/umbrales-precipitaciones-absol.pdf>
- Sokolova, M., & Lapalme, G. (2009). A systematic analysis of performance measures for classification tasks. *Information Processing & Management*, 45(4), 427–437. <https://doi.org/10.1016/j.ipm.2009.03.002>
- Sun, D., Wen, H., Wang, D., & Xu, J. (2020). A random forest model of landslide susceptibility mapping based on hyperparameter optimization using Bayes algorithm. *Geomorphology*, 362, 107201. <https://doi.org/10.1016/j.geomorph.2020.107201>
- Sun, D., Xu, J., Wen, H., & Wang, Y. (2020). An optimized random forest model and its generalization ability in landslide susceptibility mapping: Application in two areas of Three Gorges Reservoir, China. *Journal of Earth Science*, 31, 1068–1086. <https://doi.org/10.1007/s12583-020-1072-9>
- Tan, P.-N., Steinbach, M., & Kumar, V. (2013). *Introduction to data mining*. Pearson Addison-Wesley.
- Tirado Silva, M. K. (2020). *Evaluación del riesgo asociado a la vulnerabilidad física por laderas inestables en el tramo de carretera Cajamarca–Gavilán, 2018* [Tesis de licenciatura, Universidad Nacional de Cajamarca]. Repositorio Institucional UNC. <http://repositorio.unc.edu.pe/handle/20.500.14074/4012>

- Vargas Cordero, Z. R. (2009). La investigación aplicada: Una forma de conocer las realidades con evidencia científica. *Educación*, 33(1), 155–165. <https://doi.org/10.15517/revedu.v33i1.538>
- Wolpert, D. H. (1992). Stacked generalization. *Neural Networks*, 5(2), 241–259. [https://doi.org/10.1016/S0893-6080\(05\)80023-1](https://doi.org/10.1016/S0893-6080(05)80023-1)
- Zhang, A., Zhao, X.-W., Zhao, X.-Y., Zheng, X.-Z., Zeng, M., Huang, X., Wu, P., Jiang, T., Wang, S.-C., He, J., & Li, Y. (2024). Comparative study of different machine learning models in landslide susceptibility assessment: A case study of Conghua District, Guangzhou, China. *China Geology*, 7, 104–115. <https://doi.org/10.31035/cg2023056>
- Zhou, X., Wen, H., Zhang, Y., Xu, J., & Zhang, W. (2021). Landslide susceptibility mapping using hybrid random forest with GeoDetector and RFE for factor optimization. *Geoscience Frontiers*, 12(5), 101211. <https://doi.org/10.1016/j.gsf.2021.101211>
- Zhou, Z.-H. (2012). *Ensemble methods: Foundations and algorithms*. Taylor & Francis.

AGRADECIMIENTO

A Dios, por brindarnos la fortaleza y la perseverancia necesarias para culminar esta investigación.

A nuestros asesores, Mg. Billy Alexis Cayatopa Calderón y Dr. Lenin Quiñones Huatangari, por el tiempo dedicado, su permanente disposición, orientación y valiosos aportes académicos durante el desarrollo de la investigación.

Al Dr. Manuel Milla Pino, presidente del jurado evaluador, por su disposición para absolver nuestras consultas y compartir sus conocimientos y experiencia en estadística y ciencia de datos, cuyos aportes contribuyeron al fortalecimiento metodológico del trabajo.

DEDICATORIA

A mi padre, Fernando; a mi madre, Yolanda; a mi abuela, Fredesvinda; a mis tíos, María y Elver; y a mi pareja, Marcia, por su apoyo incondicional, sus enseñanzas y por acompañarme a lo largo de mi formación profesional. A ellos, con profunda gratitud, dedico este logro.

Bach. Franklin Fernando Vélchez Silva

A Dios, por guiarme y permitirme contar con personas valiosas; a mis padres, Eustaquio y Dionila, por su confianza y constante apoyo a lo largo de mi formación profesional; y a mi pareja, Gina, por su motivación y respaldo durante el desarrollo de esta investigación. Su presencia fue fundamental para alcanzar esta meta y continuar con entusiasmo hacia nuevos desafíos profesionales.

Bach. Einstein Edinson Monsalve Ticliahuanca

ANEXOS

Anexo 1

Información de la precipitación del SENAMHI de la estación de Jaén

Información de la precipitación de la estación jaén del año 2016 - 2025

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
		DPTO: CAJAMARCA					PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2016	1	0.0	0.0	2.2	0.0	0.0	0.0	0.0	0.0	7.0	15.6	0.6	0.0
2016	2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.8	0.0	0.0	0.0
2016	3	0.0	14.5	13.2	3.5	0.0	0.0	0.0	0.2	0.0	0.0	0.0	1.0
2016	4	0.0	0.0	17.0	9.5	0.0	0.0	0.0	0.0	0.0	0.0	1.2	0.0
2016	5	0.0	0.0	1.6	16.8	12.5	0.0	0.0	0.0	0.0	0.7	0.0	0.0
2016	6	6.0	0.0	4.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2016	7	0.0	3.0	19.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	21.0
2016	8	0.0	0.0	13.6	0.0	2.4	0.0	16.8	0.0	0.0	5.3	0.0	0.0
2016	9	0.0	0.0	4.8	5.2	0.8	0.0	1.4	0.0	0.0	0.0	0.0	1.4
2016	10	20.8	5.8	0.0	0.0	0.0	0.0	2.2	0.0	0.0	0.0	0.0	0.0
2016	11	0.0	0.0	0.0	0.0	0.0	0.0	0.6	2.3	0.0	5.4	0.0	0.0
2016	12	0.0	3.7	0.0	28.6	0.0	0.0	0.0	0.0	0.0	2.5	1.6	0.0
2016	13	13.0	0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2016	14	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.7	0.0	0.0
2016	15	0.0	0.0	12.0	0.0	0.0	0.0	0.4	0.0	11.6	0.0	0.0	0.3
2016	16	0.0	2.2	0.6	0.0	4.2	0.0	0.0	0.0	0.0	0.0	0.0	3.6
2016	17	0.0	1.8	0.0	20.2	1.0	0.0	0.8	2.5	0.0	0.0	0.0	0.0
2016	18	0.0	0.0	0.0	7.8	0.0	0.0	6.0	13.2	0.0	0.0	0.0	0.0
2016	19	2.2	0.0	3.0	0.0	0.0	9.2	4.0	0.0	0.0	2.0	0.0	15.5
2016	20	0.0	0.0	23.6	0.0	0.0	0.4	0.7	0.0	6.6	0.0	0.0	0.0
2016	21	1.5	0.0	8.3	3.2	1.5	1.0	0.0	0.0	6.0	0.0	0.0	12.8
2016	22	0.0	0.0	0.4	1.8	0.6	0.0	0.0	0.0	0.0	0.0	0.0	6.2
2016	23	0.0	10.4	0.0	4.5	0.8	1.4	0.0	0.0	0.0	0.0	0.0	0.0
2016	24	0.0	0.0	0.0	0.0	3.6	0.0	0.0	0.0	2.5	0.0	0.0	0.0
2016	25	0.0	13.0	0.0	0.0	0.0	7.8	0.0	0.0	0.0	4.0	0.0	0.0
2016	26	25.6	0.0	0.0	0.0	0.0	0.0	2.5	0.0	0.0	0.0	11.7	0.7
2016	27	0.0	5.5	3.8	0.0	0.0	1.8	0.0	0.0	4.0	0.0	0.0	0.0
2016	28	0.0	0.0	0.0	15.8	0.0	2.0	0.0	0.0	0.0	0.0	0.0	23.0
2016	29	0.0	4.4	0.0	0.0	7.5	0.0	0.0	0.0	0.0	0.7	0.0	4.4
2016	30	0.0	0.0	1.8	0.0	0.5	0.0	0.7	0.0	0.0	0.0	0.0	0.0
2016	31	9.5	0.0	0.0	0.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0	0.0
2017	1	1.2	0.0	0.0	0.0	1.5	0.0	0.0	0.0	0.0	3.4	0.0	3.4
2017	2	2.0	0.0	4.0	1.6	0.0	0.0	0.0	0.0	0.0	0.0	2.0	4.0
2017	3	9.4	0.0	0.0	0.0	0.0	7.5	0.0	0.0	0.0	0.0	0.0	0.0

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
		DPTO: CAJAMARCA					PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2017	4	0.4	2.9	1.0	0.0	0.0	11.9	0.0	0.0	0.0	0.0	0.0	0.0
2017	5	0.8	6.5	0.3	0.0	0.0	0.0	0.0	0.0	0.0	8.0	0.0	11.8
2017	6	5.4	0.0	0.3	0.0	1.0	1.6	0.0	0.0	0.0	0.0	15.0	1.2
2017	7	1.8	0.4	0.0	4.0	0.8	0.0	0.0	0.0	0.0	0.0	7.6	6.6
2017	8	0.0	0.3	1.7	0.0	0.0	0.0	0.0	2.0	0.0	0.0	0.0	0.0
2017	9	0.0	0.0	1.4	2.8	22.5	0.0	0.0	0.0	0.0	0.0	0.8	0.0
2017	10	9.0	0.0	1.5	26.8	14.5	0.0	0.0	0.0	0.5	0.0	0.0	0.0
2017	11	4.0	0.4	4.6	6.0	0.0	12.2	0.0	0.0	0.0	1.4	0.0	7.6
2017	12	0.0	0.0	0.0	4.5	0.0	0.0	2.6	0.0	2.0	0.0	0.0	0.0
2017	13	1.4	0.0	0.8	0.8	1.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2017	14	0.8	0.0	5.6	0.6	0.0	9.0	0.0	0.0	0.0	0.0	0.0	0.0
2017	15	15.6	0.0	12.8	0.0	1.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2017	16	0.0	0.0	0.0	0.0	1.6	0.0	0.0	0.0	0.0	10.0	0.0	0.0
2017	17	0.0	0.0	0.0	0.0	0.6	0.0	0.0	0.0	2.0	0.0	0.0	0.0
2017	18	0.4	0.0	22.6	0.0	2.5	0.0	0.8	0.0	2.0	0.0	0.0	0.0
2017	19	0.0	0.0	37.8	5.0	2.8	0.0	6.3	0.0	0.0	0.0	0.0	0.0
2017	20	0.0	0.0	0.0	16.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2017	21	0.8	0.6	6.2	13.2	5.8	12.8	0.0	21.7	0.0	0.0	0.0	0.0
2017	22	0.4	1.0	0.0	0.0	0.0	2.2	0.0	4.2	0.0	9.6	0.0	0.0
2017	23	0.0	0.0	2.6	0.0	0.0	0.0	0.0	0.0	0.0	17.1	4.6	2.6
2017	24	0.0	11.2	1.8	0.0	1.4	0.0	0.0	0.0	0.0	1.2	0.0	0.0
2017	25	5.4	6.8	34.6	0.0	0.0	0.0	0.0	0.0	0.5	0.0	3.4	0.0
2017	26	18.1	0.0	18.4	0.0	6.3	0.0	0.0	0.0	0.0	0.0	0.0	39.6
2017	27	19.2	0.8	0.0	5.2	7.5	0.0	9.4	0.0	0.0	0.0	0.0	6.0
2017	28	0.0	4.0	0.0	0.0	0.0	0.4	0.0	1.0	0.0	13.0	0.0	1.0
2017	29	0.0	0.0	0.0	2.4	2.0	0.8	1.2	7.6	1.5	17.4	0.4	14.0
2017	30	0.0	0.0	0.0	4.0	3.0	0.0	2.8	0.0	0.3	0.0	0.0	0.0
2017	31	0.0	0.0	34.4	2.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2018	1	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.6	0.0	0.0	0.0	0.0
2018	2	0.0	0.0	0.0	3.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2018	3	0.0	1.4	0.0	0.0	0.7	4.8	0.0	0.0	0.0	0.0	0.0	0.0
2018	4	0.0	1.8	0.0	6.0	0.0	3.0	16.3	0.0	1.6	5.0	1.9	0.0
2018	5	0.0	2.0	0.0	5.4	3.5	1.4	0.0	0.0	0.0	0.0	0.0	0.0
2018	6	15.7	2.5	2.8	12.2	10.2	0.0	0.0	0.0	0.0	1.4	0.0	0.0
2018	7	15.2	10.5	0.0	0.0	1.0	0.0	0.0	0.0	0.0	2.6	0.0	15.6
2018	8	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	20.5	2.2
2018	9	0.0	0.0	0.6	7.8	2.8	6.8	0.0	6.7	0.0	1.2	2.2	0.0

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
		DPTO: CAJAMARCA					PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2018	10	2.8	4.2	0.0	0.0	23.6	0.9	2.7	0.0	0.0	11.1	8.6	0.0
2018	11	0.0	7.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	54.4	0.0
2018	12	0.6	0.0	4.0	0.0	4.0	0.0	0.0	0.0	0.0	0.0	13.8	0.0
2018	13	0.0	0.0	0.0	0.0	11.3	1.3	0.0	0.0	0.0	0.0	2.0	0.0
2018	14	3.5	0.0	0.0	1.6	0.0	0.0	0.0	0.0	0.0	0.0	12.7	0.2
2018	15	9.2	1.6	0.0	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.5	6.0
2018	16	2.4	1.4	0.0	0.0	0.0	2.6	0.0	7.4	0.0	0.0	0.0	4.0
2018	17	0.4	1.2	1.0	0.0	15.5	0.0	0.0	0.0	0.0	0.0	0.0	0.7
2018	18	0.5	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	5.8	0.6
2018	19	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	6.8	0.0
2018	20	8.0	63.8	0.0	2.9	2.2	0.0	3.0	0.0	0.0	0.0	2.8	0.6
2018	21	10.3	0.0	4.0	2.0	4.2	0.0	0.0	0.0	0.0	1.3	0.0	0.0
2018	22	6.6	0.0	3.0	0.0	4.4	0.0	2.1	0.0	0.0	0.8	27.4	1.2
2018	23	0.0	0.0	0.0	2.0	0.0	0.0	1.8	0.0	1.0	0.0	0.0	1.5
2018	24	0.0	0.0	10.0	17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2018	25	0.0	0.0	0.0	0.5	0.0	0.0	0.6	0.0	0.0	0.0	0.0	0.0
2018	26	0.0	0.0	0.0	0.0	5.0	6.8	2.0	0.0	0.0	0.0	2.5	0.0
2018	27	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2018	28	0.0	13.0	0.0	0.0	0.0	0.0	5.5	0.0	0.0	0.0	0.0	0.0
2018	29	0.0	0.0	0.0	4.5	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0
2018	30	0.0	0.0	2.6	3.2	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0
2018	31	0.0	0.0	1.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2019	1	0.0	1.6	0.0	0.0	0.0	1.2	0.0	0.0	0.5	3.5	0.5	0.0
2019	2	0.0	0.8	1.4	6.0	1.0	0.0	0.0	0.0	0.0	0.0	4.0	11.8
2019	3	0.0	0.0	0.0	17.4	0.0	0.0	0.0	0.0	0.0	1.6	16.5	0.0
2019	4	0.0	7.8	0.0	2.5	0.0	6.0	0.0	0.0	0.0	0.0	5.4	2.4
2019	5	0.0	6.0	0.0	0.0	14.0	0.0	1.0	0.0	5.0	3.0	1.8	15.0
2019	6	0.0	1.0	1.2	0.0	6.4	0.0	0.8	0.0	0.0	0.0	0.0	4.4
2019	7	0.0	11.8	1.6	0.0	0.0	0.0	5.6	0.0	0.0	0.0	1.0	2.2
2019	8	0.0	2.5	0.0	0.4	0.0	0.0	0.0	0.0	0.0	0.4	6.4	5.8
2019	9	0.0	8.6	0.0	0.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	25.3
2019	10	0.0	5.5	0.2	0.0	0.0	5.4	0.0	0.0	0.0	0.0	0.0	7.6
2019	11	0.0	0.0	1.5	0.0	0.0	0.0	0.0	0.8	0.0	0.0	2.6	0.0
2019	12	0.0	11.4	0.0	4.5	0.0	2.0	0.0	2.6	0.3	0.0	0.8	0.0
2019	13	0.0	2.8	0.0	12.0	19.4	0.0	0.0	0.0	6.2	1.8	0.0	9.6
2019	14	5.0	0.0	1.4	3.4	5.1	1.2	20.4	0.0	1.0	0.0	3.4	0.0
2019	15	0.8	0.0	0.0	0.0	1.5	2.2	12.4	0.0	0.0	0.0	3.8	0.0

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
		DPTO: CAJAMARCA					PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2019	16	10.2	0.0	22.4	4.5	0.0	1.6	1.2	0.0	0.0	4.4	0.0	0.0
2019	17	2.5	2.8	0.0	0.0	0.0	6.0	0.0	0.0	0.3	1.6	0.0	0.0
2019	18	1.5	0.0	1.1	0.0	0.0	0.0	1.6	0.0	0.0	0.3	0.0	2.6
2019	19	0.0	0.0	6.4	0.0	0.0	0.0	13.6	0.0	0.0	6.5	0.0	10.0
2019	20	5.5	1.5	4.4	12.0	0.0	1.4	41.2	0.0	0.0	0.3	0.8	1.9
2019	21	5.2	6.5	0.0	0.0	0.0	0.0	11.5	1.0	0.4	0.0	0.0	3.4
2019	22	3.8	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	1.2
2019	23	1.6	3.8	5.2	0.0	0.0	0.0	0.6	0.0	0.0	8.4	2.5	1.2
2019	24	0.0	6.0	0.0	2.8	3.4	0.0	0.0	0.0	0.4	5.0	0.0	0.8
2019	25	0.0	44.0	0.0	7.8	6.5	0.0	0.0	0.0	1.5	0.0	0.0	0.0
2019	26	0.0	0.0	0.0	0.0	5.6	0.0	0.0	0.0	0.0	0.0	7.0	0.5
2019	27	0.0	2.6	0.0	0.0	0.0	2.5	0.0	0.0	0.0	2.0	0.0	2.0
2019	28	0.0	0.0	0.0	0.0	3.5	3.1	0.0	3.5	0.0	0.3	0.0	0.0
2019	29	0.0	0.0	0.0	0.0	0.0	2.5	1.8	0.0	0.0	0.0	0.0	2.0
2019	30	1.2	0.0	0.0	1.6	0.0	0.0	0.0	0.0	10.6	0.0	0.0	4.1
2019	31	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.6	0.0	5.5
2020	1	29.4	0.0	0.0	5.0	5.5	0.0	0.0	1.6	0.0	3.2	0.0	0.0
2020	2	25.5	6.0	15.0	11.8	1.6	0.0	0.0	0.0	0.0	0.0	0.0	2.4
2020	3	1.5	0.6	0.0	0.6	3.2	0.0	0.8	0.0	0.0	0.0	0.0	0.0
2020	4	0.0	0.0	0.0	0.0	14.2	0.0	1.2	0.0	0.0	1.0	0.0	2.0
2020	5	0.0	15.0	18.0	0.3	0.0	0.0	0.0	0.0	0.5	0.0	0.0	2.2
2020	6	0.0	0.0	0.0	9.4	0.0	6.0	0.0	0.0	1.4	0.0	20.0	3.2
2020	7	0.0	0.0	0.0	4.8	0.0	0.0	0.0	0.0	0.0	0.0	9.4	2.4
2020	8	0.0	0.0	0.0	6.8	1.6	0.0	0.0	0.5	0.0	0.0	0.0	3.4
2020	9	0.0	2.0	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2020	10	2.2	10.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2020	11	0.0	2.5	2.0	0.5	0.0	0.0	0.0	0.0	0.0	0.0	8.6	52.4
2020	12	0.0	0.0	2.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	9.4
2020	13	1.5	0.0	1.5	2.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.2
2020	14	0.0	0.0	3.0	12.0	0.0	0.4	0.0	0.0	0.0	0.0	0.0	0.0
2020	15	0.5	11.4	0.2	1.4	0.0	0.0	12.4	0.0	0.0	0.0	1.2	0.0
2020	16	0.0	1.5	0.0	0.3	15.2	0.0	0.0	0.0	0.0	0.0	0.0	4.0
2020	17	0.0	4.8	35.0	0.0	0.0	1.2	0.0	0.0	0.0	1.4	0.0	12.0
2020	18	22.4	0.0	1.8	1.2	3.0	0.7	0.0	0.0	0.0	0.0	15.2	17.0
2020	19	7.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	44.8	11.0
2020	20	0.0	0.0	4.8	0.0	0.0	0.0	0.0	0.0	0.9	0.0	0.0	17.1
2020	21	0.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	1.5	2.8	0.0

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
DPTO: CAJAMARCA							PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA									UNIDAD: (mm)				
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2020	22	0.0	0.0	0.0	0.0	0.0	3.4	0.0	5.6	0.0	1.5	1.5	0.6
2020	23	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.6	0.0	0.4	0.0
2020	24	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2020	25	0.0	11.8	0.0	0.0	5.4	0.3	0.0	0.0	2.6	2.5	0.0	1.5
2020	26	20.5	19.4	0.0	0.0	0.0	15.5	0.3	0.0	0.0	0.0	2.0	0.0
2020	27	0.0	0.0	0.0	3.0	1.0	14.5	1.6	0.0	0.0	0.0	28.8	0.0
2020	28	0.3	0.0	0.0	0.5	0.0	0.4	0.0	0.0	0.0	2.4	3.5	0.0
2020	29	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0	0.0
2020	30	0.0	0.0	1.6	0.0	0.0	0.0	5.0	0.0	5.6	0.0	0.0	0.0
2020	31	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0	0.0
2021	1	0.0	0.0	4.6	0.3	2.8	0.0	0.0	0.0	0.0	0.0	72.0	34.5
2021	2	5.8	0.0	21.6	0.0	0.0	0.5	0.0	0.0	0.0	0.0	0.0	11.0
2021	3	7.2	0.0	7.0	0.0	0.0	0.0	0.0	0.0	0.0	10.5	0.0	0.0
2021	4	6.0	9.7	0.5	3.0	0.0	19.0	0.0	0.0	0.0	7.2	0.0	0.0
2021	5	2.2	0.4	1.0	0.6	0.0	4.5	0.0	0.0	0.0	10.5	16.0	0.0
2021	6	0.0	0.0	0.0	3.2	0.0	0.0	0.0	0.0	2.0	0.0	3.4	6.2
2021	7	16.8	2.0	0.0	0.0	14.6	1.0	0.0	0.0	0.0	10.0	0.0	1.2
2021	8	0.0	0.0	0.0	5.6	0.5	1.0	0.0	0.0	0.0	0.0	0.0	1.5
2021	9	0.0	0.0	60.6	2.4	6.4	1.0	0.0	0.0	0.0	0.0	0.0	0.0
2021	10	0.0	0.0	18.6	5.2	5.2	0.0	0.0	1.6	0.0	10.0	7.0	0.0
2021	11	0.0	0.0	14.0	0.8	3.2	0.0	4.0	0.5	0.0	0.0	0.0	3.0
2021	12	0.0	0.0	9.6	0.0	0.0	0.0	5.5	0.6	0.8	0.0	0.0	0.0
2021	13	0.6	0.0	4.0	4.2	0.0	5.6	17.1	0.0	24.0	0.0	0.0	0.0
2021	14	0.0	0.0	1.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2021	15	0.0	8.4	0.7	15.6	3.4	0.0	0.0	1.8	0.0	20.5	0.0	0.0
2021	16	5.2	0.0	10.0	5.0	27.2	0.0	0.0	2.0	0.0	26.0	1.0	0.0
2021	17	0.0	0.0	9.0	4.5	1.8	0.0	4.6	0.0	6.8	0.0	2.4	4.5
2021	18	0.0	0.0	0.0	0.0	6.2	0.0	2.4	0.0	0.0	1.0	0.0	0.0
2021	19	3.8	0.0	13.5	0.0	0.8	0.0	0.0	0.0	0.0	3.0	0.0	10.5
2021	20	1.5	0.0	0.0	0.0	0.0	5.0	0.0	0.0	1.6	2.6	0.0	2.8
2021	21	12.6	0.0	0.0	0.0	0.0	4.8	0.0	0.0	1.5	4.5	0.0	12.0
2021	22	2.0	0.0	19.6	0.0	5.0	0.0	11.2	0.0	0.5	15.0	0.0	1.4
2021	23	1.2	0.0	0.0	0.0	7.0	0.0	0.0	0.0	1.4	2.6	0.0	19.0
2021	24	0.0	4.4	1.0	0.1	0.0	0.0	0.0	0.0	5.0	0.0	30.5	0.0
2021	25	0.0	4.9	1.4	0.0	1.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2021	26	5.8	3.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.5	2.0
2021	27	5.6	0.0	3.0	0.4	0.0	0.0	0.0	0.0	0.0	0.0	3.8	0.0

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
DPTO: CAJAMARCA							PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2021	28	1.0	4.5	8.1	3.0	0.0	0.0	0.0	10.2	0.0	11.5	0.2	0.0
2021	29	0.0	0.0	2.2	0.0	0.0	0.8	0.0	1.0	0.0	1.0	0.0	0.0
2021	30	0.0	0.0	9.0	7.6	0.0	2.0	0.0	1.0	1.0	0.0	0.0	0.0
2021	31	2.0	0.0	7.1	5.2	10.5	0.0	0.0	0.0	0.0	1.0	0.0	0.0
2022	1	0.0	0.0	0.6	17.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2022	2	0.0	0.0	1.6	1.8	0.0	0.0	2.8	2.6	0.0	3.4	8.0	0.0
2022	3	0.0	0.0	30.2	0.0	0.0	0.7	1.2	0.0	6.2	0.0	0.0	0.0
2022	4	0.0	0.0	2.5	0.0	0.0	13.0	2.0	5.3	0.0	0.0	0.0	0.0
2022	5	0.0	0.0	0.0	0.0	1.1	0.0	0.0	1.5	0.0	0.0	0.0	0.0
2022	6	0.0	9.6	0.0	0.0	11.2	0.0	0.0	1.4	0.0	0.6	0.0	0.0
2022	7	0.0	1.4	1.6	5.5	0.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2022	8	0.0	14.8	0.0	0.0	4.2	17.4	1.6	0.0	0.0	25.2	0.0	0.0
2022	9	0.0	2.0	0.0	0.0	0.0	2.2	1.2	2.6	0.0	0.0	0.0	0.5
2022	10	0.0	0.8	0.0	0.0	2.0	5.4	0.0	1.8	0.0	0.0	22.0	3.0
2022	11	0.0	1.5	0.3	0.0	0.0	9.2	0.0	3.2	0.0	0.0	0.0	27.0
2022	12	0.0	2.6	0.0	0.0	5.0	4.4	0.0	0.0	0.0	0.0	0.0	0.5
2022	13	0.0	0.0	8.2	2.6	7.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2022	14	0.4	0.0	42.2	0.0	3.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2022	15	6.5	8.0	0.0	0.0	32.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2022	16	5.8	0.0	0.0	0.0	1.6	0.0	0.0	0.0	0.0	3.0	20.0	0.0
2022	17	0.0	0.0	47.4	0.0	0.0	0.0	0.2	0.0	0.0	1.2	23.5	5.2
2022	18	0.0	0.0	0.0	11.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	3.0
2022	19	0.0	0.0	2.4	1.0	0.0	4.6	1.0	0.0	0.0	22.2	0.0	0.0
2022	20	0.0	0.0	0.0	4.6	0.0	0.0	0.0	1.2	0.0	0.0	0.0	0.0
2022	21	6.2	0.0	0.0	5.2	0.0	0.0	2.5	0.0	5.4	0.0	0.0	0.0
2022	22	0.0	0.0	13.2	0.0	0.0	0.0	0.0	0.0	0.0	8.4	0.0	1.6
2022	23	0.0	4.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.8
2022	24	1.2	0.0	6.5	1.0	0.0	0.0	9.0	0.0	0.0	3.4	0.0	0.0
2022	25	2.9	0.0	0.0	1.8	0.0	18.2	7.6	0.0	0.0	1.6	0.0	0.0
2022	26	4.0	4.0	0.0	0.0	2.6	0.0	4.1	0.0	0.0	0.0	0.0	0.0
2022	27	20.6	25.4	0.0	0.0	8.2	0.0	0.0	0.0	0.0	20.0	0.0	4.0
2022	28	4.5	0.4	3.6	1.6	2.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2022	29	0.2	1.0	0.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	2.6
2022	30	0.0	0.0	0.0	8.4	0.0	0.0	0.6	0.0	37.2	0.0	0.0	4.0
2022	31	0.0	0.0	0.0	0.0	0.0	0.0	33.8	0.0	9.5	0.0	0.0	0.0
2023	1	2.0	0.0	0.0	0.0	0.6	4.6	2.0	0.0	0.0	0.0	0.0	2.2
2023	2	4.4	6.4	0.0	0.0	0.0	4.4	0.0	5.4	0.0	0.0	0.0	3.0

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
DPTO: CAJAMARCA							PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2023	3	0.0	0.0	8.5	3.0	6.7	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2023	4	0.0	0.0	13.0	0.0	3.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2023	5	0.0	0.0	2.8	11.0	2.5	0.6	0.0	2.2	12.4	0.0	0.0	0.0
2023	6	0.0	3.5	0.0	0.0	0.0	0.0	1.4	0.0	0.0	0.0	9.2	0.5
2023	7	0.0	3.0	2.6	0.0	0.0	6.8	0.0	0.0	0.0	0.0	0.0	0.0
2023	8	0.0	0.0	0.7	13.6	0.0	8.2	2.8	0.0	0.0	0.0	0.0	0.0
2023	9	5.0	2.0	0.0	0.6	4.5	1.8	5.0	0.0	0.0	1.5	0.0	0.0
2023	10	2.5	0.0	9.0	0.0	2.0	0.0	0.0	0.0	0.0	7.8	1.4	0.0
2023	11	10.8	4.8	4.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	12.0	0.0
2023	12	7.2	6.4	0.6	0.0	1.2	0.0	0.0	0.0	0.0	0.0	0.0	2.6
2023	13	0.0	0.0	12.6	0.0	0.0	0.0	0.0	1.0	0.0	0.0	2.0	7.8
2023	14	0.0	0.0	20.2	0.0	0.0	0.0	0.0	6.7	0.0	11.8	0.0	2.4
2023	15	6.6	0.0	0.0	0.0	0.0	2.6	5.6	1.3	2.0	4.0	9.5	0.0
2023	16	13.4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.6	0.0	5.0	6.5
2023	17	0.0	0.0	0.0	16.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0
2023	18	0.0	2.0	0.0	0.0	0.0	0.0	2.5	0.0	0.0	0.0	0.0	1.8
2023	19	3.4	7.0	0.0	38.0	0.0	0.0	2.2	0.0	0.0	0.0	0.0	0.0
2023	20	3.4	0.0	0.0	11.0	19.6	0.0	0.0	0.0	0.0	0.0	0.0	4.6
2023	21	0.0	0.0	0.0	23.5	0.7	0.0	0.0	17.5	0.0	0.0	0.0	0.0
2023	22	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.5
2023	23	0.5	6.0	0.0	2.0	3.5	0.0	0.7	0.0	0.0	0.0	0.0	1.4
2023	24	0.0	0.0	0.0	0.8	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2023	25	10.7	0.0	49.2	0.4	0.0	0.0	0.0	0.0	0.0	0.0	3.0	0.0
2023	26	1.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	37.0	1.0	0.0
2023	27	0.0	0.0	0.5	0.0	0.6	0.0	0.0	0.0	0.0	0.0	4.8	22.4
2023	28	0.0	14.5	0.0	0.5	2.2	1.5	0.0	0.0	0.0	0.0	4.5	2.0
2023	29	0.0	0.0	0.0	2.4	0.0	0.0	0.0	0.0	0.0	0.0	4.6	0.0
2023	30	0.0	0.0	8.8	6.6	0.0	2.0	4.5	0.0	2.8	0.0	0.0	2.6
2023	31	0.0	0.0	0.0	0.0	9.0	0.0	0.0	0.0	0.0	11.0	3.0	0.0
2024	1	0.0	4.0	1.4	9.4	2.8	0.0	4.0	0.0	0.0	0.0	0.8	0.0
2024	2	0.0	0.8	17.9	0.0	0.0	0.0	1.5	0.0	0.0	0.0	2.2	0.0
2024	3	0.5	42.5	0.3	0.6	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2024	4	7.3	0.0	15.0	0.2	6.4	0.0	0.0	0.0	0.0	0.0	0.0	2.0
2024	5	0.0	0.6	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	7.5	0.0
2024	6	0.0	0.0	0.0	0.0	0.0	1.8	0.5	0.0	0.0	0.0	0.0	0.0
2024	7	0.0	0.0	0.0	0.0	0.0	2.2	4.2	0.0	0.0	0.0	3.3	2.9
2024	8	0.0	0.0	4.4	0.0	0.0	0.0	2.8	0.0	0.0	0.0	0.0	0.4

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
DPTO: CAJAMARCA							PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2024	9	0.0	6.6	11.5	0.0	2.2	1.6	0.0	0.0	0.0	0.0	0.0	1.5
2024	10	2.0	37.4	1.0	0.0	0.0	0.0	4.2	0.0	0.0	0.0	0.0	0.0
2024	11	3.0	0.0	1.8	0.0	0.0	4.4	2.0	0.0	0.0	0.0	0.0	0.0
2024	12	0.0	2.0	0.0	0.0	0.0	1.8	0.0	0.0	0.0	14.7	0.0	0.0
2024	13	0.0	0.0	3.5	0.0	0.0	9.6	1.3	0.0	0.0	0.0	0.0	2.3
2024	14	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.3	1.2	0.0	0.2
2024	15	0.0	0.0	1.7	0.0	0.0	0.0	0.0	0.0	0.0	1.0	0.0	0.0
2024	16	12.0	0.0	0.5	0.0	0.0	7.6	0.0	0.0	0.0	0.0	0.0	0.0
2024	17	0.0	0.0	0.0	0.0	0.0	7.4	0.0	0.0	0.0	0.0	4.5	1.2
2024	18	0.7	10.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.7
2024	19	4.2	2.0	18.6	30.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	4.7
2024	20	1.0	0.0	2.4	0.4	0.5	0.0	0.8	0.0	0.0	0.0	0.0	0.0
2024	21	1.4	0.0	0.0	2.6	0.2	7.4	0.0	0.0	0.0	0.0	0.0	0.5
2024	22	1.0	0.0	4.0	0.0	0.0	9.2	0.0	0.0	0.0	0.0	0.0	0.0
2024	23	0.7	0.0	0.2	1.4	0.0	0.0	1.2	0.0	0.0	1.0	5.8	0.0
2024	24	0.3	1.2	0.7	1.0	0.0	0.0	1.0	0.0	0.0	1.5	0.0	0.7
2024	25	0.0	0.2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0
2024	26	0.0	5.8	0.5	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2024	27	0.0	0.0	0.0	2.2	3.4	0.0	0.0	0.0	0.0	0.0	0.6	1.0
2024	28	0.2	0.0	0.0	0.5	2.4	0.0	0.0	0.0	2.2	6.4	2.8	0.8
2024	29	2.8	5.0	0.0	0.0	0.0	0.0	0.0	0.0	3.8	1.0	1.8	2.9
2024	30	0.0	0.0	0.0	3.0	0.0	0.0	0.8	0.0	0.0	1.5	0.0	2.4
2024	31	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0	0.0	0.0	0.0
2025	1	5.2	0.0	0.4	14.4	19.4	0.0	0.0	0.0	0.0	0.0	7.2	0.0
2025	2	9.2	1.4	29.4	3.6	0.0	0.0	0.5	0.0	0.6	0.0	0.4	0.0
2025	3	7.5	0.0	10.2	0.0	0.0	0.0	0.0	0.0	0.0	17.6	3.6	0.0
2025	4	22.0	0.0	3.8	0.0	0.3	0.0	0.0	1.2	0.0	0.0	7.7	0.0
2025	5	0.0	6.6	1.0	8.3	0.5	0.0	0.0	10.4	0.0	0.0	6.4	0.0
2025	6	0.7	19.0	3.0	0.6	0.6	0.5	0.0	0.4	0.0	0.0	0.0	0.0
2025	7	0.0	9.4	0.0	0.0	0.0	6.8	0.0	0.0	0.0	0.0	0.0	0.0
2025	8	0.3	7.4	7.3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
2025	9	1.5	0.0	0.0	0.2	0.0	5.4	0.0	0.3	0.0	0.0	0.0	0.0
2025	10	10.8	2.8	1.5	0.7	0.0	2.5	0.0	0.6	0.0	4.4	0.8	0.0
2025	11	0.0	11.0	0.0	10.8	0.0	1.4	0.0	0.5	0.0	7.8	1.0	0.0
2025	12	0.7	19.3	0.0	18.8	0.0	4.0	0.0	3.2	0.0	0.0	0.0	1.5
2025	13	0.0	8.8	0.0	0.5	0.0	0.0	0.0	0.0	0.0	15.2	1.4	0.0
2025	14	0.4	38.0	0.8	10.0	0.0	1.4	0.0	0.0	0.0	0.0	0.0	0.8

ESTACION: JAÉN / DZ-02		LAT: 05°40'35.99					LON: 78°46'27.05 W			ALT: 618 msnm			
		DPTO: CAJAMARCA					PROV: JAEN			DIST: JAEN			
PARAMETRO: PRECIPITACION TOTAL DIARIA										UNIDAD: (mm)			
Año	Día	Ene.	Feb.	Mar.	Abr.	May.	Jun.	Jul.	Ago.	Set.	Oct.	Nov.	Dic.
2025	15	17.8	0.0	0.0	0.0	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0
2025	16	0.0	0.0	3.2	0.0	0.0	3.4	0.0	0.0	0.0	0.0	2.8	0.0
2025	17	0.0	0.0	0.6	0.0	0.2	1.8	0.0	0.0	0.0	0.7	0.0	2.0
2025	18	14.0	55.0	0.3	1.3	0.6	0.9	6.4	0.8	0.0	0.0	4.3	1.4
2025	19	0.0	46.4	6.2	7.4	0.0	0.0	0.0	0.0	0.0	0.0	4.6	10.8
2025	20	0.0	17.1	1.2	0.0	13.4	0.0	1.4	0.0	0.7	0.6	0.0	0.0
2025	21	8.7	0.0	1.4	0.0	1.2	0.0	0.0	0.0	0.0	0.0	0.0	0.8
2025	22	2.0	0.2	3.2	0.0	0.0	0.0	0.0	0.0	0.0	0.2	0.0	0.0
2025	23	0.8	0.0	0.3	0.0	0.0	0.0	0.0	2.5	0.0	1.0	3.7	0.0
2025	24	0.0	29.8	3.4	0.0	0.8	0.0	0.0	0.3	0.2	0.3	0.5	0.0
2025	25	0.0	6.8	0.0	0.0	0.0	0.0	0.0	1.2	0.0	2.2	2.6	0.0
2025	26	0.0	14.0	0.0	0.0	0.0	24.0	0.0	0.0	0.0	4.0	0.0	0.0
2025	27	2.3	9.2	9.7	36.4	0.0	0.0	0.0	0.3	0.0	0.0	0.0	0.0
2025	28	2.8	0.4	0.8	19.8	0.0	12.2	7.6	2.3	0.0	0.0	6.0	0.0
2025	29	6.5	10.0	3.2	6.2	0.0	0.4	5.2	0.0	0.3	6.6	4.0	0.0
2025	30	5.3	0.0	1.8	0.3	1.7	0.3	0.0	0.0	0.0	5.4	8.2	0.0
2025	31	0.0	0.0	0.8	0.0	0.0	0.0	0.5	0.0	3.5	2.0	0.0	0.0
MEDIA		2.4	3.3	3.8	2.8	2.0	1.5	1.3	0.6	0.7	1.9	2.5	2.4
DESV. EST		5.0	8.0	8.4	5.8	4.5	3.4	4.1	2.2	3.0	4.7	7.2	5.9
MAX		29.4	63.8	60.6	38.0	32.6	24.0	41.2	21.7	37.2	37.0	72.0	52.4
MIN		0.2	0.2	0.2	0.1	0.2	0.3	0.2	0.2	0.2	0.2	0.2	0.2

Fuente: Adaptado de SENAMHI (2014).

Anexo 2

Instrumentos para la recolección de datos de vulnerabilidad

INSTRUMENTOS N°01: Ficha de recolección de datos

FICHA DE RECOLECCIÓN DE DATOS				
N° DE PREDIO		Fecha		Hora
Departamento	Provincia	Distrito	Sector	
Cajamarca	Jaén	Jaén	La Colina	
Coordenadas UTM	E:		N:	
DIMENSIÓN SOCIAL				
1. Cantidad de habitantes por vivienda				
6 a mas personas	de 4 a 5 personas	3 personas	2 personas	1 persona
2. Grupo etario más vulnerable de la vivienda				
Menor de 6 años y mayor de 60 años	De 51 a 60 años	De 7 a 12 años	De 13 a 19 años	De 20 a 50 años
3. Personas con dificultad permanente				
Motriz: para usar brazos y/o piernas	Cognitiva o mental	Visual: para ver, incluso con lentes	Para oír y/o hablar	No tiene
4. Abastecimiento de agua en la vivienda				
Red de agua potable	Pilón	Camión	Pozo, río o acequia	No tiene
5. Disposición sanitaria de excretas en la vivienda				
Red de desagüe	Conexión improvisada	Pozo ciego/ negro	Letrina	No tiene
6. Fuente de energía eléctrica				
Red pública	Panel Solar	Generador eléctrico	Conexión improvisada desde otra vivienda	No tiene
7. Nivel Educativo más alto en la vivienda				
Ningún nivel y/o inicial	Primaria	Secundaria	Superior no universitaria	Superior universitaria/ Postgrado
8. Capacitación en temas de Gestión del Riesgo de Desastres				
No tiene	Hace 2 años o más	Hace 1 año	Hace 6 meses	Hace 3 meses
9. Tipo de seguro de la población				
SIS	ESSALUD	FFAA-PNP	Otro seguro privado	No tiene
10. Percepción de la población acerca de los deslizamientos				
Fatalista	Desidia	Preocupado	Positivo	Optimista
DIMENSIÓN ECONÓMICA				
11. Cercanía de la vivienda a taludes				
Muy cercana: 0m a 40m	Cercana: > 40m - 80m	Medianamente cerca: > 80m - 150m	Alejada: > 150m - 400m	Muy alejada: > 400m
12. Antigüedad de la vivienda				
> 40 años	30 - 40 años	20 - 30 años	10 - 20 años	< 10 años

13. Material de construcción de las paredes de la vivienda				
Ladrillo o bloqueta de cemento	Adobe o tapia	Piedra con barro	Madera y/o triplay	Quincha
14. Material de construcción del techo de la vivienda				
Concreto armado	Teja	Calamina	Paja	Madera
15. Material de construcción del piso de la vivienda				
Tierra	Cemento	Madera	Láminas asfálticas, vinílicas	Losetas, cerámicos o similares
16. Estado de conservación de la vivienda				
	Destruído o muy mal conservado			
	Mal conservado o con servicios muy deteriorados			
	Regular o con servicios funcionales			
	Bien conservado y con servicios funcionales			
	Muy bien conservado			
17. Altura de la vivienda				
De 5 pisos a más	4 pisos	3 pisos	2 pisos	1 piso
18. Protección de la vivienda				
Ninguna	Vegetación ribereña	Sacos de arena	Drenajes	Muros de contención
19. Ingreso familiar promedio mensual				
Igual o menor a S/500	Entre S/500 a S/1000	Entre S/1000 a S/1500	Entre S/1500 a S/2000	Mayor de S/2000
20. Condición laboral del jefe/a del hogar				
Desempleado	Trabajador familiar no remunerado/ Obrero	Empleado	Trabajador independiente	Empleador
DIMENSIÓN AMBIENTAL				
21. Localización de áreas agrícolas frente a la zona de deslizamiento				
Muy cercana: 0m a 40m	Cercana: > 40m - 80m	Medianamente cerca: > 80m - 150m	Alejada: > 150m - 400m	Muy alejada: > 400m
22. Malas prácticas que degradan el suelo				
Excavaciones, cortes de terreno o escombros en pendientes	Presencia de agua que se filtra al suelo.	Tala de vegetación o limpieza de plantas.	Siembra en dirección a la pendiente	No realiza malas prácticas.
23. Cercanía a botaderos de basura				
Menor a 20 metros	Entre 20 a 50 metros	Entre 50 a 100 metros	Entre 100 a 200 metros	Mayor a 200 metros
24. Conocimiento de actividades de conservación ambiental				
No conoce	Escaso conocimiento	Conoce, pero no aplica	Conoce y aplica	Aplica y difunde

INSTRUMENTOS N°02: Cuestionario

CUESTIONARIO		
N° DE PREDIO	Fecha	Hora

I. LOCALIZACIÓN DEL AREA DE ESTUDIO

Departamento	Provincia	Distrito	Sector
Cajamarca	Jaén	Jaén	La Colina
Coordenadas UTM	E:	N:	

II. DIAGNÓSTICO DE ELEMENTOS POR DIMENSIÓN SOCIAL

1. ¿Cuántas personas viven en tu vivienda?	H:	M:
--	----	----

2. ¿Cuál es la edad de cada una de las personas que habitan permanentemente la vivienda?

Menor de 6 años y mayor de 60 años	De 51 a 60 años	De 7 a 12 años	De 13 a 19 años	De 20 a 50 años
N°: _____	N°: _____	N°: _____	N°: _____	N°: _____

3. ¿Presentas dificultad permanente?

Motriz: para usar brazos y/o piernas	Cognitiva o mental: Para comprender, Recordar o Concentrarse.	Visual: para ver, incluso con lentes	Para oír y/o hablar	No tiene
--------------------------------------	---	--------------------------------------	---------------------	----------

4. ¿El agua que abastece tu vivienda proviene de?

Red de agua potable	Pilón	Camión cisterna u otro	Pozo, río o acequia	No tiene
---------------------	-------	------------------------	---------------------	----------

5. ¿Cómo dispones las excretas en tu vivienda?

Red de desagüe	Conexión improvisada	Pozo ciego/ negro	Letrina	No tiene
----------------	----------------------	-------------------	---------	----------

6. ¿La energía eléctrica que utilizas en tu vivienda procede de?

Red Publica	Panel Solar	Generador eléctrico	Conexión improvisada desde otra vivienda	No tiene
-------------	-------------	---------------------	--	----------

7. ¿Cuál es el nivel educativo más alto que has alcanzado?

Ningún nivel y/o inicial	Primaria	Secundaria	Superior no universitaria	Superior universitaria/ Postgrado
--------------------------	----------	------------	---------------------------	-----------------------------------

8. ¿Has recibido capacitación en temas de Gestión del Riesgo de Desastres?				
No tiene	Hace 2 años o más	Hace 1 año	Hace 6 meses	Hace 3 meses

9. ¿Con qué tipo de seguro de salud cuentas?				
SIS	ESSALUD	FFAA-PNP	Otro seguro privado	No tiene

10. ¿Cuál es tu percepción acerca de la formación de los deslizamientos en el sector La Colina?	
	Existen otros peligros que afectan más. No considera a los deslizamientos
	El peligro de los deslizamientos, no me afecta
	Me afecta los deslizamientos, pero no puedo hacer nada
	Estoy dispuesto a acatar algunas medidas respecto a los deslizamientos, pero parcialmente
	Estoy dispuesto a acatar las disposiciones del gobierno

IV. DIAGNÓSTICO DE ELEMENTOS POR DIMENSIÓN ECONÓMICA

11. ¿Hace cuánto tiempo construiste tu vivienda?	
--	----------------

12. ¿De qué material son las paredes de tu vivienda?				
Ladrillo o bloqueta de cemento	Adobe o tapia	Piedra con barro	Madera y/o triplay	Quincha

13. ¿De qué material es el techo de tu vivienda?				
Concreto armado	Teja	Calamina	Paja	Madera

14. ¿De qué material es el piso de tu vivienda?				
Tierra	Cemento	Madera	Láminas asfálticas, vinílicas	Losetas, cerámicos o similares

15. ¿Cuál es el estado de conservación de tu vivienda?	
	Se encuentra destruido o muy mal conservado
	Se encuentra mal conservado o con servicios muy deteriorados
	Se encuentra regular o con servicios funcionales
	Se encuentra bien conservado y con servicios funcionales
	Se encuentra muy bien conservado

16. ¿Cuántos pisos tiene tu vivienda?	
---------------------------------------	----------------

17. ¿Cuenta con alguna protección tu vivienda, para dar respuesta ante un posible deslizamiento?				
Ninguna	Vegetación ribereña	Sacos de arena	Drenajes	Muros de contención

18. ¿Cuánto es el ingreso familiar promedio al mes en tu hogar?				
Igual o menor a S/500	Entre S/500 a S/1000	Entre S/1000 a S/ 1500	Entre S/ 1500 a S/ 2000	Mayor de S/ 2000

19. ¿Cuál es la condición laboral del jefe de tu hogar?				
Desempleado	Trabajador familiar no remunerado/ Obrero	Empleado	Trabajador independiente	Empleador

IV. DIAGNÓSTICO DE ELEMENTOS POR DIMENSIÓN AMBIENTAL

20. ¿Cuál de estas prácticas que degradan el suelo, has aplicado y/o observado en tu familia?				
Excavaciones, cortes de terreno o escombros en pendientes.	Presencia de agua que se filtra al suelo.	Tala de vegetación o limpieza de plantas.	Siembra en dirección a la pendiente.	No realiza malas prácticas.

21. ¿Conoces acerca de actividades de conservación ambiental?				
No conoce	Escaso conocimiento	Conoce, pero no aplica	Conoce y aplica	Aplica y difunde

Anexo 3

Validez y Confiabilidad de instrumentos.

Validez: La ficha utilizada para la validación por juicio de experto en la presente investigación, fue adaptada de Milla Pino et al. (2023).

Ficha de evaluación de instrumentos aplicada al instrumento 01, experto 01.

8. FICHA DE EVALUACIÓN INSTRUMENTO N°01

HOJA DE VALIDACIÓN FICHA DE RECOLECCIÓN DE DATOS																								
Título de la investigación: ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025																								
Elementos	Criterios															Juicio								
	Coherencia					Pertinencia					Independencia					Impacto					Aceptar	Modificar	Rechazar	
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5				
1				✓	✓					✓					✓						✓			
2					✓					✓					✓						✓			
3					✓					✓					✓						✓			
4				✓						✓					✓						✓			
5					✓					✓					✓						✓			
6				✓						✓					✓						✓			
7					✓					✓					✓						✓			
8					✓					✓					✓						✓			
9					✓					✓					✓						✓			
10					✓					✓					✓						✓			
11					✓					✓					✓						✓			
12					✓					✓					✓						✓			
13				✓					✓						✓						✓			
14					✓					✓					✓						✓			
15					✓					✓					✓						✓			
16					✓					✓					✓						✓			
17				✓						✓					✓						✓			
18					✓					✓					✓						✓			
19					✓					✓					✓						✓			
20					✓					✓					✓						✓			
21				✓						✓					✓						✓			
22					✓					✓					✓						✓			
23					✓					✓					✓						✓			
24					✓					✓					✓						✓			

Observaciones:

Lugar y fecha: JAEN	 <p>MUNICIPALIDAD PROVINCIAL DE JAEN SUB GERENCIA DE GESTION DE RIESGOS Y DESASTRES Lesley Miler Rodríguez Trigoso RUC 201010152990 Firma</p>
Nombre del experto: LESLEY MILER RODRIGUEZ TRIGOSO	
Área de especialización: GESTION DE RIESGOS DE DESASTRES	
Afiliación institucional: MUNICIPALIDAD PROVINCIAL DE JAEN - SUB GERENCIA DE GESTION DE RIESGOS DE DESASTRES	
Título Profesional y Grado Académico: INGENIERO CIVIL	

Ficha de evaluación de instrumentos aplicada al instrumento 01, experto 02.

9. FICHA DE EVALUACIÓN INSTRUMENTO N°02

HOJA DE VALIDACIÓN CUESTIONARIO																							
Título de la investigación: ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025																							
Elementos	Criterios																				Juicio		
	Coherencia					Pertinencia					Independencia					Impacto					Aceptar	Modificar	Rechazar
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5			
1					X					X					X					X			
2					X					X					X					X			
3					X					X					X					X			
4				X					X						X				X				
5				X					X						X				X				
6				X					X						X			X					
7				X					X						X			X					
8				X					X						X				X				
9				X					X						X			X					
10				X					X						X			X					
11				X					X						X			X					
12				X					X						X			X					
13				X					X						X			X					
14				X					X						X			X					
15				X					X						X			X					
16				X					X						X			X					
17				X					X						X			X					
18				X					X						X			X					
19				X					X						X			X					
20				X					X						X			X					
21				X					X						X			X					

Observaciones: *El cuestionario presenta adecuada claridad, coherencia interna y pertinencia técnica para el diagnóstico de la vulnerabilidad social, económica y ambiental frente al riesgo por deslizamientos. Los ítems son comprensibles para la población objetivo y guardan correspondencia con los lineamientos metodológicos del CENEPRED, por lo que el instrumento se considera válido para su aplicación en campo.*

Lugar y fecha: *Jaén, enero del 2026*

Nombre del experto: *Ing. Giam Carlos Campos Cardozo*

Área de especialización: *Evaluador del Riesgo Originados por Fenómenos Naturales.*

Afiliación institucional: *CENEPRED*

Título Profesional y Grado Académico: *Ingeniero Civil, con Postgrado en Transportes y Gestión Vial*


ING. GIAM CARLOS CAMPOS CARDOZO
 EVALUADOR DE RIESGO DE DESASTRES
 R.J. N° 090-2022-CENEPRED/J
 CIP N° 734984

Ficha de evaluación de instrumentos aplicada al instrumento 02, experto 01.

9. FICHA DE EVALUACIÓN INSTRUMENTO N°02

HOJA DE VALIDACIÓN_CUESTIONARIO																						
Título de la investigación: ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN – CAJAMARCA, 2025																						
Elementos	Criterios															Juicio						
	Coherencia					Pertinencia					Independencia					Impacto					Aceptar	Modificar
1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5			
1					✓					✓					✓					✓		
2					✓					✓					✓				✓			
3					✓					✓					✓				✓			
4					✓				✓					✓					✓			
5				✓						✓					✓				✓			
6					✓					✓					✓				✓			
7					✓					✓					✓				✓			
8					✓					✓					✓				✓			
9				✓					✓						✓				✓			
10					✓					✓					✓				✓			
11					✓					✓				✓					✓			
12					✓				✓					✓					✓			
13					✓					✓					✓				✓			
14				✓						✓					✓				✓			
15					✓					✓					✓				✓			
16					✓				✓					✓					✓			
17					✓					✓				✓					✓			
18					✓					✓				✓					✓			
19					✓					✓				✓					✓			
20					✓					✓				✓					✓			
21					✓					✓				✓				✓				

Observaciones:

Lugar y fecha: JAÉN	 MUNICIPALIDAD PROVINCIAL DE JAÉN SUB GERENCIA DE GESTIÓN DE RIESGO Y DESASTRES Lesley Milner Rodríguez Trigo ING. CIVIL - C.R. 132090 SUB GERENTE
Nombre del experto: LESLEY MILER RODRÍGUEZ TRIGOSO	
Área de especialización: GESTIÓN DE RIESGOS DE DESASTRES	
Afiliación institucional: MUNICIPALIDAD PROVINCIAL DE JAÉN - SUB GERENCIA DE GESTIÓN DE RIESGOS DE DESASTRES	
Título Profesional y Grado Académico: INGENIERO CIVIL	 Firma

Ficha de evaluación de instrumentos aplicada al instrumento 01, experto 02.

8. FICHA DE EVALUACIÓN INSTRUMENTO N°01

HOJA DE VALIDACIÓN FICHA DE RECOLECCIÓN DE DATOS																								
Título de la investigación: ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025																								
Elementos	Criterios															Juicio								
	Coherencia					Pertinencia					Independencia					Impacto					Aceptar	Modificar	Rechazar	
1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5					
1					X					X					X					X				
2					X					X					X					X				
3					X					X					X					X				
4				X					X						X				X					
5				X					X						X				X					
6				X					X						X			X						
7				X					X						X			X						
8				X					X						X				X					
9				X					X						X			X						
10				X					X						X				X					
11				X					X						X				X					
12				X					X						X			X						
13				X					X						X				X					
14				X					X						X			X						
15				X					X						X			X						
16				X					X						X				X					
17				X					X						X			X						
18				X					X						X				X					
19				X					X						X			X						
20				X					X						X			X						
21				X					X						X				X					
22				X					X						X				X					
23				X					X						X			X						
24				X					X						X			X						

Observaciones: El instrumento presenta adecuada coherencia interna, pertinencia técnica e impacto para la graduación de la Vulnerabilidad frente al riesgo por deslizamientos. Se recomienda precisar algunos rangos y criterios operativos para optimizar la recolección de datos en campo.

Lugar y fecha: Jaén, Enero del 2026	 ING. GIAM CARLOS CAMPOS CARDOZO EVALUADOR DE RIESGO DE DESASTRES R.J N° 090-2022-CENEPRED/J N° 234984 Firma
Nombre del experto: Ing. Giam Carlos Campos Cardozo	
Área de especialización: Evaluador del Riesgo Originado por fenómenos naturales.	
Afiliación institucional: CENEPRED	
Título Profesional y Grado Académico: Ingeniero Civil, con Maestría en Transportes y Conservación Vial	

8. FICHA DE EVALUACIÓN INSTRUMENTO N°01

HOJA DE VALIDACIÓN FICHA DE RECOLECCIÓN DE DATOS																									
Título de la investigación: ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025																									
Elementos	Criterios																				Juicio				
	Coherencia					Pertinencia					Independencia					Impacto					Aceptar	Modificar	Rechazar		
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5					
1					✓					✓					✓					✓					✓
2					✓					✓					✓					✓					✓
3					✓					✓					✓					✓					✓
4					✓					✓					✓					✓					✓
5					✓					✓					✓					✓					✓
6					✓					✓					✓					✓					✓
7					✓					✓					✓					✓					✓
8					✓					✓					✓					✓					✓
9					✓					✓					✓					✓					✓
10					✓					✓					✓					✓					✓
11					✓					✓					✓					✓					✓
12					✓					✓					✓					✓					✓
13					✓					✓					✓					✓					✓
14				✓						✓					✓					✓					✓
15				✓						✓					✓					✓					✓
16				✓						✓					✓					✓					✓
17				✓						✓					✓					✓					✓
18				✓						✓					✓					✓					✓
19				✓						✓					✓					✓					✓
20				✓						✓					✓					✓					✓
21				✓						✓					✓					✓					✓
22				✓						✓					✓					✓					✓
23				✓						✓					✓					✓					✓
24				✓						✓					✓					✓					✓

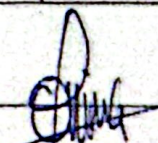
Observaciones:

Lugar y fecha: <u>Jaén, enero 2026.</u>	 CRISTIAN DARWIN PEREZ VASQUEZ INGENIERO CIVIL RÉG CIP 360418
Nombre del experto: <u>Cristian Darwin Perez Vasquez.</u>	
Área de especialización: <u>Supervisión - Residencia de Obras Civiles</u>	
Afiliación institucional: <u></u>	
Título Profesional y Grado Académico: <u>Ingeniero Civil.</u>	Firma

9. FICHA DE EVALUACIÓN INSTRUMENTO N°02

HOJA DE VALIDACIÓN CUESTIONARIO																								
Título de la investigación: ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN – CAJAMARCA, 2025																								
Elementos	Criterios																				Juicio			
	Coherencia					Pertinencia					Independencia					Impacto					Aceptar	Modificar	Rechazar	
	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5	1	2	3	4	5				
1					✓					✓					✓					✓	✓			
2					✓					✓					✓					✓	✓			
3					✓					✓					✓					✓	✓			
4					✓					✓					✓					✓	✓			
5					✓					✓					✓					✓	✓			
6					✓					✓					✓					✓	✓			
7					✓					✓					✓					✓	✓			
8					✓					✓					✓					✓	✓			
9				✓						✓					✓					✓	✓			
10				✓						✓					✓					✓	✓			
11				✓						✓					✓					✓	✓			
12				✓						✓					✓					✓	✓			
13				✓						✓					✓					✓	✓			
14				✓					✓	✓					✓					✓	✓			
15				✓					✓	✓					✓					✓	✓			
16				✓					✓	✓					✓					✓	✓			
17				✓					✓	✓					✓					✓	✓			
18				✓					✓	✓					✓					✓	✓			
19				✓					✓	✓					✓				✓	✓	✓			
20				✓					✓	✓					✓				✓	✓	✓			
21				✓					✓	✓					✓				✓	✓	✓			

Observaciones:

Lugar y fecha: Jaén, enero 2026.	 CRISTIAN DARWIN PEREZ VASQUEZ INGENIERO CIVIL REG CIP 360418 Firma
Nombre del experto: Cristian Darwin Perez Vasquez	
Área de especialización: Supervisión - Residencia en Obras Civiles	
Afiliación institucional:	
Título Profesional y Grado Académico: Ingeniero Civil.	

Confiabilidad del cuestionario (Instrumento N°02):

Resultados del índice de vulnerabilidad en el test y retest.

Lote	Vuln test	Vuln retest	Diferencia absoluta	Nivel test	Nivel retest	Concordancia
2	0,165097	0,164500	0,000597	Alta	Alta	Sí
21	0,219812	0,219844	0,000032	Alta	Alta	Sí
22	0,232099	0,232108	0,000009	Alta	Alta	Sí
23	0,282929	0,282404	0,000525	Muy alta	Muy alta	Sí
31	0,254050	0,254048	0,000002	Alta	Alta	Sí
34	0,226371	0,226380	0,000009	Alta	Alta	Sí
35	0,257045	0,257035	0,000010	Alta	Alta	Sí
39	0,209469	0,209467	0,000002	Alta	Alta	Sí
40	0,265380	0,265411	0,000031	Alta	Alta	Sí
58	0,283886	0,283874	0,000012	Muy alta	Muy alta	Sí
59	0,365813	0,364736	0,001077	Muy alta	Muy alta	Sí
61	0,142509	0,142506	0,000003	Alta	Alta	Sí
66	0,211091	0,211122	0,000031	Alta	Alta	Sí
67	0,170053	0,175694	0,005641	Alta	Alta	Sí
73	0,185782	0,185814	0,000032	Alta	Alta	Sí
75	0,348073	0,350972	0,002899	Muy alta	Muy alta	Sí
77	0,217562	0,217551	0,000011	Alta	Alta	Sí
79	0,218479	0,218511	0,000032	Alta	Alta	Sí
86	0,224328	0,218897	0,005431	Alta	Alta	Sí
91	0,246536	0,246546	0,000010	Alta	Alta	Sí

93	0,309600	0,309599	0,000001	Muy alta	Muy alta	Sí
101	0,251180	0,248937	0,002243	Alta	Alta	Sí
104	0,331852	0,331842	0,000010	Muy alta	Muy alta	Sí
106	0,226450	0,221234	0,005216	Alta	Alta	Sí
110	0,177045	0,177451	0,000406	Alta	Alta	Sí
112	0,328013	0,333648	0,005635	Muy alta	Muy alta	Sí
114	0,224624	0,224877	0,000253	Alta	Alta	Sí
119	0,179941	0,180347	0,000406	Alta	Alta	Sí
120	0,242440	0,244659	0,002219	Alta	Alta	Sí
121	0,274589	0,268953	0,005636	Muy alta	Muy alta	Sí

Matriz de concordancia entre test y retest

Test \ Retest	Vulnerabilidad alta	Vulnerabilidad muy alta	Total test
Vulnerabilidad alta	22	0	22
Vulnerabilidad muy alta	0	8	8
Total retest	22	8	30

Anexo 4

Matriz de comparabilidad de las dos bases de datos

Matriz de comparabilidad metodológica de las bases de datos de La Colina y Jangas

Tabla Anexo 4

Comparación de los descriptores de las matrices de evaluación de La Colina y Jangas

N.º	Variable	Descriptores La Colina	Descriptores Jangas	Descriptores coincidentes
1	Geología	5	3	0
2	Geomorfología	5	5	0
3	Pendiente	5	5	5
4	Uso actual del suelo	5	5	2
5	Precipitación	5	5	5
6	Cantidad de habitantes por vivienda	5	5	5
7	Grupo etario más vulnerable en la vivienda	5	5	5
8	Personas con dificultad permanente	5	5	5
9	Abastecimiento de agua en la vivienda	5	5	5
10	Disposición sanitaria de excretas en la vivienda	5	5	4
11	Fuente de energía eléctrica	5	5	4
12	Nivel educativo más alto en la vivienda	5	5	5
13	Capacitación de la población en temas de GRD	5	5	5
14	Tipo de seguro de salud de la población	5	5	5
15	Percepción de la población acerca de los deslizamientos	5	5	5
16	Cercanía de la vivienda a la zona de deslizamientos	5	5	5

Tabla Anexo 4

N.º	Variable	Descriptor La Colina	Descriptor Jangas	Descriptor coincidentes
17	Antigüedad de la vivienda	5	5	5
18	Material de construcción de paredes	5	5	5
19	Material de construcción en techo	5	5	5
20	Material de construcción en piso	5	5	5
21	Estado de conservación de la vivienda	5	5	5
22	Altura de la vivienda	5	5	5
23	Protección de la vivienda	5	5	5
24	Ingreso familiar promedio mensual	5	5	5
25	Condición laboral del jefe de hogar	5	5	5
26	Localización de áreas agrícolas frente a la zona de deslizamiento	5	5	5
27	Malas prácticas de población que degradan el suelo	5	5	5
28	Cercanía de botaderos de basura	5	5	5
29	Conocimiento de actividades de conservación ambiental	5	5	5
30	Nivel de riesgo (variable objetivo)	4	4	4

Nota. Los descriptores coincidentes corresponden a indicadores cuya equivalencia de significado y criterios de evaluación fue verificada por los autores de la investigación. La coincidencia expresa equivalencia metodológica y no implica igualdad en las frecuencias o distribuciones de los registros. GRD = gestión del riesgo de desastres.

Elaboración propia a partir del *Plan de prevención y reducción del riesgo de desastres del distrito de Jangas 2023-2030*, de la Municipalidad Distrital de Jangas (2023), y de la matriz de evaluación de La Colina correspondiente a la presente investigación, *Estimación de riesgo por deslizamiento empleando algoritmos ensamblados en el sector La Colina 2025*.

Anexo 5

Código utilizado en la investigación

PARTE I. MODELOS INDIVIDUALES Y ENSAMBLE STACKING

Instalación de dependencias

```
!pip -q install scikit-learn pandas numpy matplotlib  
joblib openpyxl scikit-optimize
```

0.1 Importación de librerías

```
import os  
import re  
import shutil  
import warnings  
import unicodedata  
from datetime import datetime  
from pathlib import Path  
  
import joblib  
import matplotlib.pyplot as plt  
import numpy as np  
import pandas as pd  
  
from google.colab import files  
IN_COLAB = True  
  
from sklearn.base import clone  
from sklearn.compose import ColumnTransformer  
from sklearn.ensemble import ExtraTreesClassifier,  
GradientBoostingClassifier, RandomForestClassifier,  
StackingClassifier  
from sklearn.inspection import  
permutation_importance  
from sklearn.metrics import (  
    ConfusionMatrixDisplay,  
    accuracy_score,  
    balanced_accuracy_score,  
    classification_report,  
    cohen_kappa_score,  
    confusion_matrix,  
    f1_score,  
    make_scorer,  
    precision_score,  
    recall_score,  
)  
from sklearn.model_selection import StratifiedKFold,  
cross_val_predict, cross_validate  
from sklearn.pipeline import Pipeline  
from sklearn.preprocessing import LabelEncoder,  
OneHotEncoder  
from sklearn.tree import DecisionTreeClassifier  
from sklearn.linear_model import LogisticRegression  
  
try:  
    from skopt import BayesSearchCV  
    from skopt.space import Categorical, Integer,  
    Real  
except ImportError:  
    import sys  
    import subprocess  
  
    subprocess.check_call([sys.executable, "-m",  
    "pip", "install", "scikit-optimize"])  
    from skopt import BayesSearchCV  
    from skopt.space import Categorical, Integer,  
    Real  
  
warnings.filterwarnings("ignore",  
category=FutureWarning)  
warnings.filterwarnings("ignore",  
category=UserWarning)  
  
RANDOM_STATE = 42  
TARGET = "NIVEL-DE-RIESGO"  
NOMBRE_ARCHIVO_ESPERADO = "DATA_TESIS.csv"  
  
1. Comprensión de los datos  
1.1 Carga manual del archivo CSV desde Google Colab  
def leer_csv_con_separador_automatico(ruta):  
    # Lee un CSV probando separadores comunes y  
    elige el que genere mas columnas.  
    candidatos = [";", ",", "\t"]  
    mejores = []  
  
    for encoding in ["utf-8", "utf-8-sig", "latin-  
1"]:  
        for sep in candidatos:  
            try:  
                temporal = pd.read_csv(ruta,  
sep=sep, encoding=encoding)
```

```
                mejores.append((temporal.shape[1],  
sep, encoding, temporal))  
            except Exception:  
                continue  
  
        if not mejores:  
            return pd.read_csv(ruta, sep=None,  
engine="python"), "inferido", "automatico"  
  
        mejores.sort(key=lambda x: x[0], reverse=True)  
        columnas, sep, encoding, dataframe = mejores[0]  
  
        if columnas <= 1:  
            dataframe = pd.read_csv(ruta, sep=None,  
engine="python", encoding=encoding)  
            sep = "inferido"  
  
        return dataframe, sep, encoding  
  
print(f"Suba manualmente el archivo CSV esperado:  
{NOMBRE_ARCHIVO_ESPERADO}")  
uploaded = files.upload()  
  
if not uploaded:  
    raise FileNotFoundError("No se subio ningun  
archivo. Vuelva a ejecutar la celda y cargue el  
CSV.")  
  
archivos_subidos = list(uploaded.keys())  
  
if NOMBRE_ARCHIVO_ESPERADO in archivos_subidos:  
    ruta_csv = NOMBRE_ARCHIVO_ESPERADO  
elif len(archivos_subidos) == 1:  
    ruta_csv = archivos_subidos[0]  
    print(f"Advertencia: no se encontro el nombre  
exacto esperado. Se usara el archivo subido:  
{ruta_csv}")  
else:  
    raise FileNotFoundError(  
        f"No se encontro {NOMBRE_ARCHIVO_ESPERADO}.  
Archivos subidos: {archivos_subidos}"  
    )  
  
df_raw, separador_detectado, encoding_detectado =  
leer_csv_con_separador_automatico(ruta_csv)  
  
print("Archivo cargado correctamente.")  
print(f"Separador detectado: {separador_detectado}")  
print(f"Codificacion usada: {encoding_detectado}")  
  
1.2 Revisión inicial de la base de datos  
print("Dimensión de la base de datos (filas,  
columnas):")  
print(df_raw.shape)  
  
print("\nNombres de columnas:")  
for i, col in enumerate(df_raw.columns, start=1):  
    print(f"{i:02d}. {col}")  
  
print("\nPrimeras filas:")  
display(df_raw.head())  
  
print("\nTipos de datos detectados por pandas:")  
display(df_raw.dtypes.to_frame("tipo_dato"))  
  
print("\nValores nulos por columna:")  
display(df_raw.isna().sum().to_frame("nulos"))  
  
1.3 Validación de la variable objetivo  
if TARGET not in df_raw.columns:  
    columnas_parecidas = [col for col in  
df_raw.columns if str(col).strip().upper() ==  
TARGET]  
    mensaje = f"No se encontró la columna objetivo  
exacta: {TARGET}."  
    if columnas_parecidas:  
        mensaje += f" Columnas parecidas  
encontradas: {columnas_parecidas}. Revise espacios o  
caracteres especiales en el nombre."  
        raise ValueError(mensaje)  
  
print(f"Variable objetivo encontrada correctamente:  
{TARGET}")  
  
1.4 Detección de categorías repetidas por espacios o  
mayúsculas/minúsculas  
def detectar_variantes_categoricas(dataframe,  
columnas):  
    # Detecta categorías con diferencias solo por
```

```

espacios o mayúsculas/minúsculas.
registros = []

for col in columnas:
    valores = dataframe[col].dropna().map(str)
    grupos = {}

    for valor in valores.unique():
        clave_normalizada =
valor.strip().lower()
        grupos.setdefault(clave_normalizada,
set()).add(valor)

    for variantes in grupos.values():
        if len(variantes) > 1:
            registros.append({
                "columna": col,
                "variantes_detectadas":
sorted(variantes),
                "cantidad_variantes":
len(variantes),
            })

    return pd.DataFrame(registros)

variantes_categoricas =
detectar_variantes_categoricas(df_raw,
df_raw.columns)

if variantes_categoricas.empty:
    print("No se detectaron categorías repetidas por
diferencias de espacios o mayúsculas/minúsculas.")
else:
    print("Posibles categorías repetidas
detectadas:")
    display(variantes_categoricas)

```

1.5 Limpieza básica de textos categóricos

```

def limpiar_valor_texto(valor):
    # Convierte valores categóricos a texto y limpia
    espacios sin alterar valores nulos.
    if pd.isna(valor):
        return np.nan
    texto = str(valor).strip()
    return np.nan if texto == "" else texto

# Todas las variables se tratan como
categóricas/texto, según la descripción de la base.
df = df_raw.copy()
for columna in df.columns:
    df[columna] =
df[columna].map(limpiar_valor_texto)

print("Limpieza de espacios aplicada. Vista
preliminar:")
display(df.head())

print("\nValores nulos por columna después de
limpiar espacios:")
display(df.isna().sum().to_frame("nulos"))

```

1.6 Distribución de la variable objetivo

```

distribucion_abs =
df[TARGET].value_counts(dropna=False)
distribucion_pct =
df[TARGET].value_counts(normalize=True,
dropna=False).mul(100).round(2)

distribucion_objetivo = pd.DataFrame({
    "frecuencia": distribucion_abs,
    "porcentaje": distribucion_pct,
})

print(f"Distribución de la variable objetivo:
{TARGET}")
display(distribucion_objetivo)

plt.figure(figsize=(8, 5))
df[TARGET].value_counts(dropna=False).plot(kind="bar
", color="#2E86AB", edgecolor="black")
plt.title("Distribución de clases de nivel de
riesgo")
plt.xlabel("Nivel de riesgo")
plt.ylabel("Cantidad de registros")
plt.xticks(rotation=45, ha="right")
plt.grid(axis="y", alpha=0.3)
plt.tight_layout()
plt.show()

```

2. Preparación de los datos

2.1 Tratamiento de valores faltantes y definición de X e y

```
df_modelo = df.copy()
```

```

registros_iniciales = len(df_modelo)
df_modelo = df_modelo.dropna(subset=[TARGET]).copy()
registros_eliminados = registros_iniciales -
len(df_modelo)

if registros_eliminados > 0:
    print(f"Se eliminaron {registros_eliminados}
registros sin clase de riesgo en {TARGET}.")

X = df_modelo.drop(columns=[TARGET]).copy()
y = df_modelo[TARGET].copy()

# Reemplazo claro de faltantes en predictor
categóricas.
X = X.fillna("Sin-dato")
X = X.replace("", "Sin-dato")

for columna in X.columns:
    X[columna] = X[columna].map(lambda v: "Sin-dato"
if pd.isna(v) or str(v).strip() == "" else
str(v).strip())

y = y.map(lambda v: str(v).strip())

print("Dimensión de X:", X.shape)
print("Dimensión de y:", y.shape)
print("\nPrimeras filas de X:")
display(X.head())

```

2.2 Codificación de la variable objetivo

```

label_encoder = LabelEncoder()
y_encoded = label_encoder.fit_transform(y)

mapa_clases = pd.DataFrame({
    "codigo": range(len(label_encoder.classes_)),
    "clase_original": label_encoder.classes_,
})

print("Correspondencia entre código numérico y clase
original:")
display(mapa_clases)

conteo_clases =
pd.Series(y_encoded).value_counts().sort_index()
conteo_clases.index =
label_encoder.inverse_transform(conteo_clases.index)
print("\nFrecuencia por clase usada en modelado:")
display(conteo_clases.to_frame("frecuencia"))

```

2.3 Validación cruzada estratificada

```

frecuencia_minima =
int(pd.Series(y_encoded).value_counts().min())

if frecuencia_minima < 2:
    raise ValueError(
        "Hay al menos una clase con menos de 2
registros. "
        "No es posible aplicar validación cruzada
estratificada. Revise o agrupe clases muy escasas."
    )

if frecuencia_minima >= 10:
    n_splits = 10
else:
    n_splits = min(5, frecuencia_minima)

cv = StratifiedKFold(n_splits=n_splits,
shuffle=True, random_state=RANDOM_STATE)

# Validación interna usada por StackingClassifier.
n_splits_stacking_interno = min(5, n_splits)
cv_stacking_interno = StratifiedKFold(
    n_splits=n_splits_stacking_interno,
    shuffle=True,
    random_state=RANDOM_STATE,
)

print(f"Frecuencia de la clase minoritaria:
{frecuencia_minima}")
print(f"Número de particiones para evaluación
principal: {n_splits}")
print(f"Número de particiones internas para
stacking: {n_splits_stacking_interno}")

```

2.4 Preprocesamiento con One-Hot Encoding

```

def crear_onehot_encoder():
    # Compatible con versiones recientes y
    anteriores de scikit-learn.
    try:
        return
OneHotEncoder(handle_unknown="ignore",
sparse_output=False)

```

```

except TypeError:
    return
OneHotEncoder(handle_unknown="ignore", sparse=False)

def crear_preprocesador(columnas_predictoras):
    return ColumnTransformer(
        transformers=[
            ("onehot", crear_onehot_encoder(),
list(columnas_predictoras)),
        ],
        remainder="drop",
        verbose_feature_names_out=True,
    )

columnas_predictoras = list(X.columns)
preprocesador_base =
crear_preprocesador(columnas_predictoras)

print(f"Número de variables predictoras categóricas:
{len(columnas_predictoras)}")

```

3. Modelado

```

def crear_pipeline(clasificador):
    return Pipeline(steps=[
        ("preprocesamiento",
crear_preprocesador(columnas_predictoras)),
        ("modelo", clasificador),
    ])

pipelines_base = {
    "Decision Tree": crear_pipeline(

DecisionTreeClassifier(random_state=RANDOM_STATE,
class_weight="balanced")
    ),
    "Random Forest": crear_pipeline(

RandomForestClassifier(random_state=RANDOM_STATE,
n_jobs=-1)
    ),
    "Extra Trees": crear_pipeline(

ExtraTreesClassifier(random_state=RANDOM_STATE,
n_jobs=-1)
    ),
    "Gradient Boosting": crear_pipeline(

GradientBoostingClassifier(random_state=RANDOM_STATE
)
    ),
}

```

```

print("Modelos base definidos:")
for nombre in pipelines_base:
    print("-", nombre)

```

4. Optimización de hiperparámetros con búsqueda bayesiana

```
N_ITER_BAYES = 20
```

```

espacios_busqueda = {
    "Decision Tree": {
        "modelo__max_depth": Integer(2, 30),
        "modelo__min_samples_split": Integer(2, 20),
        "modelo__min_samples_leaf": Integer(1, 10),
        "modelo__criterion": Categorical(["gini",
"entropy"]),
    },
    "Random Forest": {
        "modelo__n_estimators": Integer(100, 400),
        "modelo__max_depth": Categorical([None, 5,
10, 15, 20, 30, 40]),
        "modelo__min_samples_split": Integer(2, 20),
        "modelo__min_samples_leaf": Integer(1, 10),
        "modelo__max_features": Categorical(["sqrt",
"log2", None]),
        "modelo__class_weight": Categorical([None,
"balanced", "balanced_subsample"]),
    },
    "Extra Trees": {
        "modelo__n_estimators": Integer(100, 400),
        "modelo__max_depth": Categorical([None, 5,
10, 15, 20, 30, 40]),
        "modelo__min_samples_split": Integer(2, 20),
        "modelo__min_samples_leaf": Integer(1, 10),
        "modelo__max_features": Categorical(["sqrt",
"log2", None]),
        "modelo__class_weight": Categorical([None,
"balanced", "balanced_subsample"]),
    },
    "Gradient Boosting": {

```

```

        "modelo__n_estimators": Integer(50, 250),
        "modelo__learning_rate": Real(0.01, 0.30,
prior="log-uniform"),
        "modelo__max_depth": Integer(1, 5),
        "modelo__min_samples_split": Integer(2, 20),
        "modelo__min_samples_leaf": Integer(1, 10),
        "modelo__subsample": Real(0.60, 1.00),
    },
}

```

```

def optimizar_modelo(nombre, pipeline, espacio):
    print(f"\nOptimizando: {nombre}")
    busqueda = BayesSearchCV(
        estimator=pipeline,
        search_spaces=espacio,
        n_iter=N_ITER_BAYES,
        scoring="f1_macro",
        cv=cv,
        n_jobs=-1,
        random_state=RANDOM_STATE,
        refit=True,
        verbose=0,
        error_score="raise",
    )
    busqueda.fit(X, y_encoded)
    print(f"Mejor F1-score macro CV:
{busqueda.best_score_:.4f}")
    print("Mejores hiperparámetros:")
    print(busqueda.best_params_)
    return busqueda

```

```

busquedas_bayes = {}
for nombre_modelo, pipeline in
pipelines_base.items():
    busquedas_bayes[nombre_modelo] =
optimizar_modelo(
        nombre_modelo,
        pipeline,
        espacios_busqueda[nombre_modelo],
    )

```

4.1 Mejores modelos base optimizados

```

best_dt = busquedas_bayes["Decision
Tree"].best_estimator_
best_rf = busquedas_bayes["Random
Forest"].best_estimator_
best_et = busquedas_bayes["Extra
Trees"].best_estimator_
best_gb = busquedas_bayes["Gradient
Boosting"].best_estimator_

```

```

resumen_modelos_base = pd.DataFrame([
    {
        "modelo": nombre,
        "mejor_f1_macro_cv": busqueda.best_score_,
        "mejores_hiperparametros":
busqueda.best_params_,
    }
    for nombre, busqueda in busquedas_bayes.items()
]).sort_values("mejor_f1_macro_cv", ascending=False)

```

```

print("Resumen de modelos base optimizados:")
display(resumen_modelos_base)

```

5. Ensamblaje mediante Stacking

```

def crear_meta_logistica():
    return LogisticRegression(
        max_iter=5000,
        class_weight="balanced",
        solver="lbfgs",
        random_state=RANDOM_STATE,
    )

def crear_stacking(estimadores_base, meta_aprendiz):
    return StackingClassifier(
        estimators=estimadores_base,
        final_estimator=meta_aprendiz,
        stack_method="predict_proba",
        cv=cv_stacking_interno,
        n_jobs=-1,
        passthrough=False,
    )

```

```

modelos_individuales = {
    "Decision Tree": best_dt,
    "Random Forest": best_rf,
    "Extra Trees": best_et,
    "Gradient Boosting": best_gb,
}

```

```

ensambles = {
    "Combinacion 1: DT + RF + ET + GB |

```

```

LogisticRegression": crear_stacking(
    estimadores_base=[("dt", best_dt), ("rf",
best_rf), ("et", best_et), ("gb", best_gb)],
    meta_aprendiz=crear_meta_logistica(),
),
"Combinacion 2: RF + ET + GB |
LogisticRegression": crear_stacking(
    estimadores_base=[("rf", best_rf), ("et",
best_et), ("gb", best_gb)],
    meta_aprendiz=crear_meta_logistica(),
),
"Combinacion 3: RF + ET | LogisticRegression":
crear_stacking(
    estimadores_base=[("rf", best_rf), ("et",
best_et)],
    meta_aprendiz=crear_meta_logistica(),
),
"Combinacion 4: RF + GB | LogisticRegression":
crear_stacking(
    estimadores_base=[("rf", best_rf), ("gb",
best_gb)],
    meta_aprendiz=crear_meta_logistica(),
),
"Combinacion 5: DT + RF + ET + GB |
GradientBoosting": crear_stacking(
    estimadores_base=[("dt", best_dt), ("rf",
best_rf), ("et", best_et), ("gb", best_gb)],
),
meta_aprendiz=GradientBoostingClassifier(random_stat
e=RANDOM_STATE),
),
)

claves_ensambles = {
    "CB_1": "Combinacion 1: DT + RF + ET + GB |
LogisticRegression",
    "CB_2": "Combinacion 2: RF + ET + GB |
LogisticRegression",
    "CB_3": "Combinacion 3: RF + ET |
LogisticRegression",
    "CB_4": "Combinacion 4: RF + GB |
LogisticRegression",
    "CB_5": "Combinacion 5: DT + RF + ET + GB |
GradientBoosting",
}

descripcion_ensambles = {
    "CB_1": {
        "Modelos_primer_nivel": "Decision Tree +
Random Forest + Extra Trees + Gradient Boosting",
        "Meta_modelo": "Regresion Logistica",
    },
    "CB_2": {
        "Modelos_primer_nivel": "Random Forest +
Extra Trees + Gradient Boosting",
        "Meta_modelo": "Regresion Logistica",
    },
    "CB_3": {
        "Modelos_primer_nivel": "Random Forest +
Extra Trees",
        "Meta_modelo": "Regresion Logistica",
    },
    "CB_4": {
        "Modelos_primer_nivel": "Random Forest +
Gradient Boosting",
        "Meta_modelo": "Regresion Logistica",
    },
    "CB_5": {
        "Modelos_primer_nivel": "Decision Tree +
Random Forest + Extra Trees + Gradient Boosting",
        "Meta_modelo": "Gradient Boosting",
    },
}

modelos_ensamblados = {
    combinacion: ensambles[clave_ensamble]
    for combinacion, clave_ensamble in
claves_ensambles.items()
}

tabla_ensambles = pd.DataFrame([
    {
        "Ensamble": combinacion,
        "Descripcion":
claves_ensambles[combinacion],
        "Modelos_primer_nivel":
descripcion_ensambles[combinacion]["Modelos_primer_n
ivel"],
        "Meta_modelo":
descripcion_ensambles[combinacion]["Meta_modelo"],

```

```

    }
    for combinacion in modelos_ensamblados
])

print("Modelos individuales optimizados:")
for nombre in modelos_individuales:
    print("-", nombre)

print("\nEnsamblajes definidos mediante Stacking:")
display(tabla_ensambles)

6. Entrenamiento y evaluación de los algoritmos sin
ensamblar con cross validation
OUTPUTS_TESIS = Path("outputs")
DIR_METRICAS = OUTPUTS_TESIS / "metricas_modelos"
DIR_GRAFICOS = OUTPUTS_TESIS / "graficos_metricas"
DIR_MATRICES = OUTPUTS_TESIS / "matrices_confusion"
DIR_IMPORTANCIA = OUTPUTS_TESIS /
"importancia_variables"
DIR_PREDICCIONES = OUTPUTS_TESIS / "predicciones"

for carpeta in [DIR_METRICAS, DIR_GRAFICOS,
DIR_MATRICES, DIR_IMPORTANCIA, DIR_PREDICCIONES]:
    carpeta.mkdir(parents=True, exist_ok=True)

def guardar_tabla(tabla, ruta_base):
    ruta_base = Path(ruta_base)
    csv_path = ruta_base.with_suffix(".csv")
    xlsx_path = ruta_base.with_suffix(".xlsx")

    tabla.to_csv(csv_path, index=False,
encoding="utf-8-sig")
    try:
        tabla.to_excel(xlsx_path, index=False)
    except Exception as exc:
        print(f"No se pudo guardar Excel para
{xlsx_path.name}. Se conserva el CSV. Detalle:
{exc}")

    return csv_path, xlsx_path

def nombre_archivo_seguro(texto):
    texto = str(texto).strip().replace(" ", "_")
    texto = unicodedata.normalize("NFKD",
texto).encode("ascii", "ignore").decode("ascii")
    texto = re.sub(r"[^A-Za-z0-9_]+", "_", texto)
    texto = re.sub(r"_+", "_", texto).strip("_")
    return texto or "modelo"

SCORING_CV = {
    "accuracy": make_scorer(accuracy_score),
    "balanced_accuracy":
make_scorer(balanced_accuracy_score),
    "precision_macro": make_scorer(precision_score,
average="macro", zero_division=0),
    "recall_macro": make_scorer(recall_score,
average="macro", zero_division=0),
    "f1_macro": make_scorer(f1_score,
average="macro", zero_division=0),
    "f1_weighted": make_scorer(f1_score,
average="weighted", zero_division=0),
    "cohen_kappa": make_scorer(cohen_kappa_score),
}

NOMBRES_METRICAS_CV = {
    "accuracy": "Accuracy",
    "balanced_accuracy": "Balanced Accuracy",
    "precision_macro": "Precision Macro",
    "recall_macro": "Recall Macro",
    "f1_macro": "F1 Macro",
    "f1_weighted": "F1 Weighted",
    "cohen_kappa": "Cohen Kappa",
}

def evaluar_modelos_con_cross_validation(modelos,
tipo_modelo, columna_nombre):
    filas = []

    for nombre, estimador in modelos.items():
        print(f"Evaluando con cross validation:
{nombre}")
        cv_result = cross_validate(
            estimador,
            X,
            y_encoded,
            cv=cv,
            scoring=SCORING_CV,
            n_jobs=-1,
            error_score="raise",
        )

```

```

        fila = {
            "Tipo": tipo_modelo,
            "columna_nombre": nombre,
            "Metodo_evaluacion": "cross_validate",
            "N_splits": n_splits,
        }

        for clave_scorer, nombre_metrica in
            NOMBRES_METRICAS_CV.items():
            valores =
            cv_result[f"test_{clave_scorer}"]
            fila[nombre_metrica] =
            float(np.mean(valores))
            fila[f"{nombre_metrica}_Std"] =
            float(np.std(valores))

        filas.append(fila)

    return pd.DataFrame(filas).sort_values(
        by=["F1_Macro", "Balanced_Accuracy"],
        ascending=False,
    ).reset_index(drop=True)

def graficar_metricas_agrupadas(tabla,
    columna_nombre, titulo, ruta_salida):
    metricas_grafico = ["Accuracy", "F1_Macro",
        "Cohen_Kappa"]
    etiquetas_metricas = {
        "Accuracy": "Accuracy",
        "F1_Macro": "F1 macro",
        "Cohen_Kappa": "Cohen's Kappa",
    }
    colores = {
        "Accuracy": "#2E86AB",
        "F1_Macro": "#F18F01",
        "Cohen_Kappa": "#6A994E",
    }

    datos = tabla.sort_values("F1_Macro",
        ascending=True).copy()
    y_pos = np.arange(len(datos))
    altura = 0.22

    fig, ax = plt.subplots(figsize=(11, max(4.5,
        len(datos) * 0.75)))

    for i, metrica in enumerate(metricas_grafico):
        valores =
        datos[metrica].astype(float).to_numpy()
        barras = ax.barh(
            y_pos + (i - 1) * altura,
            valores,
            height=altura,
            label=etiquetas_metricas[metrica],
            color=colores[metrica],
            alpha=0.9,
            edgecolor="black",
            linewidth=0.4,
        )
        ax.bar_label(barras, labels=[f"{valor:.3f}"
            for valor in valores], padding=3, fontsize=8)

    minimo =
    float(datos[metricas_grafico].min().min())
    maximo =
    float(datos[metricas_grafico].max().max())
    ax.set_xlim(min(0, minimo - 0.05), max(1.0,
        maximo + 0.08))
    ax.set_yticks(y_pos)
    ax.set_yticklabels(datos[columna_nombre])
    ax.set_xlabel("Valor de la metrica")
    ax.set_title(titulo)
    ax.grid(axis="x", alpha=0.25)
    ax.legend(loc="lower right")
    plt.tight_layout()

    fig.savefig(ruta_salida, dpi=300,
        bbox_inches="tight")
    plt.show()
    print(f"Grafico guardado en: {ruta_salida}")

    tabla_metricas_modelos_individuales =
    evaluar_modelos_con_cross_validation(
        modelos=modelos_individuales,
        tipo_modelo="Modelo individual",
        columna_nombre="Modelo",
    )

    guardar_tabla(
        tabla_metricas_modelos_individuales,

```

```

        DIR_METRICAS /
        "metricas_modelos_individuales_cross_validation",
    )

    print("Metricas de modelos individuales ordenadas
    por F1_Macro:")
    display(tabla_metricas_modelos_individuales)

    graficar_metricas_agrupadas(
        tabla=tabla_metricas_modelos_individuales,
        columna_nombre="Modelo",
        titulo="Modelos individuales: Accuracy, F1 macro
        y Cohen's Kappa",
        ruta_salida=DIR_GRAFICOS /
        "metricas_modelos_individuales_barras_horizontales.p
        ng",
    )

    mejor_modelo_individual_nombre =
    tabla_metricas_modelos_individuales.loc[0, "Modelo"]
    mejor_modelo_individual =
    modelos_individuales[mejor_modelo_individual_nombre]

    print(f"Mejor modelo individual segun F1 macro:
    {mejor_modelo_individual_nombre}")

7. Entrenamiento y evaluación de los algoritmos
    ensamblados con cross validation
    tabla_metricas_ensambles =
    evaluar_modelos_con_cross_validation(
        modelos=modelos_ensamblados,
        tipo_modelo="Ensamble Stacking",
        columna_nombre="Ensamble",
    )

    tabla_metricas_ensambles =
    tabla_metricas_ensambles.merge(
        tabla_ensambles,
        on="Ensamble",
        how="left",
    )

    columnas_principales = [
        "Tipo",
        "Ensamble",
        "Descripcion",
        "Modelos_primer_nivel",
        "Meta_modelo",
        "Metodo_evaluacion",
        "N_splits",
        "Accuracy",
        "Balanced_Accuracy",
        "Precision_Macro",
        "Recall_Macro",
        "F1_Macro",
        "F1_Weighted",
        "Cohen_Kappa",
    ]

    columnas_std = [col for col in
        tabla_metricas_ensambles.columns if
        col.endswith("_Std")]
    tabla_metricas_ensambles =
    tabla_metricas_ensambles[columnas_principales +
        columnas_std]

    guardar_tabla(
        tabla_metricas_ensambles,
        DIR_METRICAS /
        "metricas_ensambles_cross_validation",
    )

    print("Metricas de ensambles ordenadas por
    F1_Macro:")
    display(tabla_metricas_ensambles)

    graficar_metricas_agrupadas(
        tabla=tabla_metricas_ensambles,
        columna_nombre="Ensamble",
        titulo="Ensamblados Stacking: Accuracy, F1 macro y
        Cohen's Kappa",
        ruta_salida=DIR_GRAFICOS /
        "metricas_ensambles_barras_horizontales.png",
    )

    mejor_ensamble_nombre =
    tabla_metricas_ensambles.loc[0, "Ensamble"]
    mejor_ensamble =
    modelos_ensamblados[mejor_ensamble_nombre]

    # Alias conservados para compatibilidad con
    secciones posteriores.

```

```

mejor_nombre = mejor_ensamble_nombre
resultados_ensambles =
tabla_metricas_ensambles.copy()

print(f"Mejor ensamble segun F1 macro:
{mejor_ensamble_nombre}")
display(tabla_ensambles[tabla_ensambles["Ensamble"].
eq(mejor_ensamble_nombre)])

8. Evaluación final con cross_val_predict del mejor
modelo individual sin ensamblar
etiquetas_codificadas =
np.arange(len(label_encoder.classes_))
etiquetas_clases = label_encoder.classes_

def calcular_metricas_prediccion(y_real, y_pred):
    return {
        "Accuracy": accuracy_score(y_real, y_pred),
        "Balanced Accuracy":
balanced_accuracy_score(y_real, y_pred),
        "Precision_Macro": precision_score(y_real,
y_pred, average="macro", zero_division=0),
        "Recall_Macro": recall_score(y_real, y_pred,
average="macro", zero_division=0),
        "F1_Macro": f1_score(y_real, y_pred,
average="macro", zero_division=0),
        "F1_Weighted": f1_score(y_real, y_pred,
average="weighted", zero_division=0),
        "Cohen_Kappa": cohen_kappa_score(y_real,
y_pred),
    }

def evaluar_final_con_cross_val_predict(estimador,
nombre, tipo, prefijo_archivo):
    print(f"Generando predicciones fuera de muestra
con cross_val_predict: {nombre}")
    y_pred = cross_val_predict(
        estimador,
        X,
        y_encoded,
        cv=cv,
        n_jobs=-1,
        method="predict",
    )

    metricas =
calcular_metricas_prediccion(y_encoded, y_pred)
    tabla_metricas = pd.DataFrame([
        "Tipo": tipo,
        "Modelo": nombre,
        "Metodo_evaluacion": "cross_val_predict",
        **metricas,
    ])

    guardar_tabla(tabla_metricas, DIR_METRICAS /
f"metricas_finales_{prefijo_archivo}")

    print("Metricas finales:")
    display(tabla_metricas)

    reporte = classification_report(
        y_encoded,
        y_pred,
        labels=etiquetas_codificadas,
        target_names=etiquetas_clases,
        zero_division=0,
        output_dict=True,
    )

    tabla_reporte =
pd.DataFrame(reporte).transpose().reset_index().rename(
columns={"index": "Clase"})
    guardar_tabla(tabla_reporte, DIR_METRICAS /
f"classification_report_{prefijo_archivo}")

    print("\nClassification report:")
    display(tabla_reporte)

    matriz = confusion_matrix(y_encoded, y_pred,
labels=etiquetas_codificadas)
    matriz_norm = confusion_matrix(y_encoded,
y_pred, labels=etiquetas_codificadas,
normalize="true")

    matriz_df = pd.DataFrame(matriz,
index=etiquetas_clases, columns=etiquetas_clases)
    matriz_norm_df = pd.DataFrame(matriz_norm,
index=etiquetas_clases,
columns=etiquetas_clases).round(4)

    guardar_tabla(

```

```

matriz_df.rename_axis("Clase_real").reset_index(),
        DIR_MATRICES /
f"matriz_confusion_absoluta_{prefijo_archivo}",
    )
    guardar_tabla(

matriz_norm_df.rename_axis("Clase_real").reset_index
()),
        DIR_MATRICES /
f"matriz_confusion_normalizada_{prefijo_archivo}",
    )

    print("Matriz de confusion absoluta:")
    display(matriz_df)

    print("Matriz de confusion normalizada por clase
real:")
    display(matriz_norm_df)

    fig, axes = plt.subplots(1, 2, figsize=(15, 6))

    ConfusionMatrixDisplay(
        confusion_matrix=matriz,
        display_labels=etiquetas_clases,
    ).plot(cmap="Blues", ax=axes[0],
values_format="d", colorbar=False)
    axes[0].set_title("Matriz de confusion
absoluta")
    axes[0].tick_params(axis="x", rotation=45)

    ConfusionMatrixDisplay(
        confusion_matrix=matriz_norm,
        display_labels=etiquetas_clases,
    ).plot(cmap="Blues", ax=axes[1],
values_format=".2f", colorbar=False)
    axes[1].set_title("Matriz de confusion
normalizada")
    axes[1].tick_params(axis="x", rotation=45)

    fig.suptitle(f"{tipo}: {nombre}", fontsize=14)
    plt.tight_layout()

    ruta_figura = DIR_MATRICES /
f"matrices_confusion_{prefijo_archivo}.png"
    fig.savefig(ruta_figura, dpi=300,
bbox_inches="tight")
    plt.show()
    print(f"Figura de matrices guardada en:
{ruta_figura}")

    return {
        "nombre": nombre,
        "tipo": tipo,
        "y_true": y_encoded,
        "y_pred": y_pred,
        "metricas": metricas,
        "tabla_metricas": tabla_metricas,
        "classification_report": tabla_reporte,
        "matriz_confusion": matriz,
        "matriz_confusion_normalizada": matriz_norm,
    }

resultado_final_modelo_individual =
evaluar_final_con_cross_val_predict(
    estimador=mejor_modelo_individual,
    nombre=mejor_modelo_individual_nombre,
    tipo="Mejor modelo individual",

prefijo_archivo=f"mejor_modelo_individual_{nombre_ar
chivo_seguro(mejor_modelo_individual_nombre)}",
)

metricas_finales_modelo_individual =
resultado_final_modelo_individual["metricas"]
y_pred_cv_mejor_modelo_individual =
resultado_final_modelo_individual["y_pred"]
cm_mejor_modelo_individual =
resultado_final_modelo_individual["matriz_confusion"]
cm_norm_mejor_modelo_individual =
resultado_final_modelo_individual["matriz_confusion_
normalizada"]

9. Evaluación final con cross_val_predict del mejor
ensamble de algoritmos
resultado_final_ensamble =
evaluar_final_con_cross_val_predict(
    estimador=mejor_ensamble,
    nombre=mejor_ensamble_nombre,
    tipo="Mejor ensamble",

```

```

prefijo_archivo=f"mejor_ensamble_{nombre_archivo_seguro(mejor_ensamble_nombre)}",
)

metricas_finales_ensamble =
resultado_final_ensamble["metricas"]
y_pred_cv_mejor_ensamble =
resultado_final_ensamble["y_pred"]
cm_mejor_ensamble =
resultado_final_ensamble["matriz_confusion"]
cm_norm_mejor_ensamble =
resultado_final_ensamble["matriz_confusion_normalizada"]

# Alias conservados para el guardado final del modelo.
metricas_finales = metricas_finales_ensamble
y_pred_cv = y_pred_cv_mejor_ensamble
cm = cm_mejor_ensamble
cm_norm = cm_norm_mejor_ensamble

print("Comparacion sintetica entre ganadores:")
comparacion_ganadores = pd.DataFrame([
    {
        "Ganador": "Modelo individual",
        "Nombre": mejor_modelo_individual_nombre,
        **metricas_finales_modelo_individual,
    },
    {
        "Ganador": "Ensamble",
        "Nombre": mejor_ensamble_nombre,
        **metricas_finales_ensamble,
    },
])
display(comparacion_ganadores)
guardar_tabla(comparacion_ganadores, DIR_METRICAS /
"comparacion_ganadores_modelo_individual_vs_ensamble")

10. Importancia de variables
# Se entrena una copia del mejor ensamble con toda la base para estimar importancias.
mejor_ensamble_entrenado = clone(mejor_ensamble)
mejor_ensamble_entrenado.fit(X, y_encoded)

# Recuperacion de nombres one-hot para trazabilidad.
preprocesador_trazabilidad =
crear_preprocesador(columnas_predictoras)
preprocesador_trazabilidad.fit(X)

try:
    nombres_oh =
preprocesador_trazabilidad.get_feature_names_out()
except Exception:
    nombres_oh = []

print(f"Cantidad de variables generadas por One-Hot Encoding: {len(nombres_oh)}")
if len(nombres_oh) > 0:
    display(pd.DataFrame({"variable_codificada_oh": nombres_oh}).head(20))

# La permutacion se aplica sobre cada columna original de X; por eso la importancia queda agrupada por variable original.
resultado_permutacion = permutation_importance(
    mejor_ensamble_entrenado,
    X,
    y_encoded,
    scoring="f1_macro",
    n_repeats=10,
    random_state=RANDOM_STATE,
    n_jobs=-1,
)

importancia_variables = pd.DataFrame([
    "variable_original": columnas_predictoras,
    "importancia_media_f1_macro":
resultado_permutacion.importances_mean,
    "importancia_desviacion":
resultado_permutacion.importances_std,
]).sort_values("importancia_media_f1_macro",
ascending=False).reset_index(drop=True)

print("Importancia de variables agrupada por variable original:")
display(importancia_variables)

importancia_variables.to_csv("importancia_variables_riesgo.csv", index=False, encoding="utf-8-sig")
guardar_tabla(importancia_variables, DIR_IMPORTANCIA

```

```

/ "importancia_variables_riesgo")
print("Archivo exportado:
importancia_variables_riesgo.csv")
top_n = min(15, len(importancia_variables))
top_importancia =
importancia_variables.head(top_n).sort_values("importancia_media_f1_macro", ascending=True)

plt.figure(figsize=(9, max(5, top_n * 0.35)))
plt.barh(
    top_importancia["variable_original"],
    top_importancia["importancia_media_f1_macro"],
    xerr=top_importancia["importancia_desviacion"],
    color="#1F77B4",
    edgecolor="black",
    alpha=0.85,
)
plt.title(f"Top {top_n} variables mas importantes del mejor ensamble")
plt.xlabel("Disminucion media del F1 macro al permutar")
plt.ylabel("Variable")
plt.grid(axis="x", alpha=0.3)
plt.tight_layout()

ruta_importancia = DIR_IMPORTANCIA /
"importancia_variables_top.png"
plt.savefig(ruta_importancia, dpi=300,
bbox_inches="tight")
plt.show()

print(f"Grafico de importancia guardado en: {ruta_importancia}")

11. Predicciones de evaluación y modelo final para los mejores modelos
def normalizar_nombre_probabilidad(clase,
prefijo="PROB"):
    texto = str(clase).strip().upper().replace("-",
"_").replace(" ", "_")
    texto = unicodedata.normalize("NFKD",
texto).encode("ascii", "ignore").decode("ascii")
    texto = re.sub(r"^[A-Z0-9_]+", "_", texto)
    texto = re.sub(r"_+", "_", texto).strip("_")
    return f"{prefijo}_{texto}"

def exportar_predicciones_evaluacion_cv(y_pred_cv,
ruta_salida, columna_predicha, columna_correcta):
    predicciones_clase =
label_encoder.inverse_transform(np.asarray(y_pred_cv).astype(int))

    df_salida = df_modelo.copy()
    df_salida[columna_predicha] = predicciones_clase
    df_salida[columna_correcta] = (
        df_salida[TARGET].astype(str).to_numpy() ==
predicciones_clase
    )
    df_salida.to_csv(ruta_salida, index=False,
encoding="utf-8-sig")

    return df_salida, predicciones_clase

def exportar_predicciones_modelo_final(
    estimador,
    ruta_salida,
    columna_predicha,
    columna_correcta,
    prefijo_probabilidad,
):
    modelo_final = clone(estimador)
    modelo_final.fit(X, y_encoded)

    predicciones_codificadas =
modelo_final.predict(X)
    predicciones_clase =
label_encoder.inverse_transform(
np.asarray(predicciones_codificadas).astype(int)
)

    df_salida = df_modelo.copy()
    df_salida[columna_predicha] = predicciones_clase
    df_salida[columna_correcta] = (
        df_salida[TARGET].astype(str).to_numpy() ==
predicciones_clase
    )

    if hasattr(modelo_final, "predict_proba"):
        try:
            probabilidades =

```



```

modelo_final.predict_proba(X)
    for indice_clase, clase in
enumerate(label_encoder.classes_):
    nombre_columna =
normalizar_nombre_probabilidad(clase,
prefijo=prefijo_probabilidad)
    df_salida[nombre_columna] =
probabilidades[:, indice_clase]
    except Exception as exc:
        print("El modelo tiene predict_proba,
pero no fue posible calcular probabilidades.")
        print(f"Detalle tecnico: {exc}")
    else:
        print("El modelo no permite calcular
predict_proba. Se exportaran solo las clases
predichas.")

    df_salida.to_csv(ruta_salida, index=False,
encoding="utf-8-sig")

    return modelo_final, df_salida,
predicciones_clase

# 11.1 Mejor algoritmo individual: evaluacion
honesta con cross_val_predict
ruta_evaluacion_modelo_individual = (
    DIR_PREDICCIONES /
"01_evaluacion_CENEPRED_vs_mejor_modelo_individual_c
ross_val_predict.csv"
)
df_evaluacion_mejor_modelo_individual,
predicciones_cv_mejor_modelo_individual_clase = (
    exportar_predicciones_evaluacion_cv(
        y_pred_cv=y_pred_cv_mejor_modelo_individual,

ruta_salida=ruta_evaluacion_modelo_individual,
columna_predicha="NIVEL-DE-RIESGO-PREDICHO-
CV-MEJOR-MODELO-INDIVIDUAL",

columna_correcta="PREDICCION_CV_CORRECTA_MEJOR_MODEL
O_INDIVIDUAL",
    )
)

# 11.2 Mejor algoritmo individual: modelo final
entrenado con toda la base
ruta_modelo_final_modelo_individual = (
    DIR_PREDICCIONES /
"02_modelo_final_mejor_modelo_individual_entrenado_c
on_toda_la_base.csv"
)
mejor_modelo_individual_final,
df_modelo_final_mejor_modelo_individual,
predicciones_final_mejor_modelo_individual_clase = (
    exportar_predicciones_modelo_final(
        estimador=mejor_modelo_individual,

ruta_salida=ruta_modelo_final_modelo_individual,
columna_predicha="NIVEL-DE-RIESGO-PREDICHO-
MODELO-FINAL-MEJOR-MODELO-INDIVIDUAL",

columna_correcta="PREDICCION_MODELO_FINAL_CORRECTA_M
EJOR_MODELO_INDIVIDUAL",

prefijo_probabilidad="PROB_FINAL_MODELO_INDIVIDUAL",
    )
)

# 11.3 Mejor ensamble: evaluacion honesta con
cross_val_predict
ruta_evaluacion_ensamble = (
    DIR_PREDICCIONES /
"03_evaluacion_CENEPRED_vs_mejor_ensamble_cross_val_
predict.csv"
)
df_evaluacion_mejor_ensamble,
predicciones_cv_mejor_ensamble_clase =
exportar_predicciones_evaluacion_cv(
    y_pred_cv=y_pred_cv_mejor_ensamble,
    ruta_salida=ruta_evaluacion_ensamble,
    columna_predicha="NIVEL-DE-RIESGO-PREDICHO-CV-
MEJOR-ENSAMBLE",

columna_correcta="PREDICCION_CV_CORRECTA_MEJOR_ENSAM
BLE",
)

# 11.4 Mejor ensamble: modelo final entrenado con
toda la base
ruta_modelo_final_ensamble = (
    DIR_PREDICCIONES /
"04_modelo_final_mejor_ensamble_entrenado_con_toda_l
a_base.csv"
)
mejor_ensamble_final,
df_modelo_final_mejor_ensamble,
predicciones_final_mejor_ensamble_clase = (
    exportar_predicciones_modelo_final(
        estimador=mejor_ensamble,
        ruta_salida=ruta_modelo_final_ensamble,
        columna_predicha="NIVEL-DE-RIESGO-PREDICHO-
MODELO-FINAL-MEJOR-ENSAMBLE",

columna_correcta="PREDICCION_MODELO_FINAL_CORRECTA_M
EJOR_ENSAMBLE",
        prefijo_probabilidad="PROB_FINAL_ENSAMBLE",
    )
)

# Archivo para mapa metodologico: usa predicciones
fuera de muestra con cross_val_predict.
# Este archivo es el adecuado para representar la
evaluacion del algoritmo frente a CENEPRED.
df_predicciones_cv_para_mapa =
df_evaluacion_mejor_ensamble.copy()

archivo_resultado_mapa_cv =
"resultado_riesgo_predicho_cv_para_mapa.csv"
df_predicciones_cv_para_mapa.to_csv(
    archivo_resultado_mapa_cv,
    index=False,
    encoding="utf-8-sig",
)

ruta_resultado_mapa_cv_outputs = DIR_PREDICCIONES /
"resultado_riesgo_predicho_cv_para_mapa.csv"
df_predicciones_cv_para_mapa.to_csv(
    ruta_resultado_mapa_cv_outputs,
    index=False,
    encoding="utf-8-sig",
)

# Archivo del modelo final entrenado con toda la
base.

archivo_resultado_modelo_final =
"resultado_modelo_final_mejor_ensamble_entrenado_tod
a_base.csv"
df_modelo_final_mejor_ensamble.to_csv(
    archivo_resultado_modelo_final,
    index=False,
    encoding="utf-8-sig",
)

ruta_predicciones_outputs = DIR_PREDICCIONES /
"resultado_modelo_final_mejor_ensamble_entrenado_tod
a_base.csv"
df_modelo_final_mejor_ensamble.to_csv(
    ruta_predicciones_outputs,
    index=False,
    encoding="utf-8-sig",
)

print("Archivos de predicciones generados:")
for ruta in [
    ruta_evaluacion_modelo_individual,
    ruta_modelo_final_modelo_individual,
    ruta_evaluacion_ensamble,
    ruta_modelo_final_ensamble,
    ruta_resultado_mapa_cv_outputs,
    Path(archivo_resultado_mapa_cv),
    ruta_predicciones_outputs,
    Path(archivo_resultado_modelo_final),
]:
    print("-", ruta)

print("\nCorrectas CV mejor modelo individual:",
int(df_evaluacion_mejor_modelo_individual["PREDICCIO
N_CV_CORRECTA_MEJOR_MODELO_INDIVIDUAL"].sum()))
print("Correctas modelo final mejor modelo
individual:",
int(df_modelo_final_mejor_modelo_individual["PREDICC
ION_MODELO_FINAL_CORRECTA_MEJOR_MODELO_INDIVIDUAL"].
sum()))
print("Correctas CV mejor ensamble:",
int(df_evaluacion_mejor_ensamble["PREDICCION_CV_CORR
ECTA_MEJOR_ENSAMBLE"].sum()))
print("Correctas modelo final mejor ensamble:",
int(df_modelo_final_mejor_ensamble["PREDICCION_MODEL
O_FINAL_CORRECTA_MEJOR_ENSAMBLE"].sum()))

```

```
display(df_predicciones_cv_para_mapa.head())
display(df_modelo_final_mejor_ensemble.head())
```

12. Guardado del modelo

```
paquete_modelo = {
    "modelo": mejor_ensemble_final,
    "modelo_individual_final":
mejor_modelo_individual_final,
    "label_encoder": label_encoder,
    "columnas_predictoras": columnas_predictoras,
    "clases_objetivo": list(label_encoder.classes_),
    "variable_objetivo": TARGET,
    "mejor_modelo_individual":
mejor_modelo_individual_nombre,
    "mejor_ensemble": mejor_ensemble_nombre,
    "metricas_finales_modelo_individual":
metricas_finales_modelo_individual,
    "metricas_finales_ensemble":
metricas_finales_ensemble,
    "archivos_predicciones": {
        "evaluacion_cv_mejor_modelo_individual":
str(ruta_evaluacion_modelo_individual),
        "modelo_final_mejor_modelo_individual":
str(ruta_modelo_final_modelo_individual),
        "evaluacion_cv_mejor_ensemble":
str(ruta_evaluacion_ensemble),
        "modelo_final_mejor_ensemble":
str(ruta_modelo_final_ensemble),
        "mapa_evaluacion_cv_mejor_ensemble":
str(ruta_resultado_mapa_cv_outputs),

    "modelo_final_entrenado_toda_base_mejor_ensemble":
str(ruta_predicciones_outputs),
    },
    "fecha_entrenamiento":
datetime.now().strftime("%Y-%m-%d %H:%M:%S"),
}

archivo_modelo =
"modelo_ensemble_riesgo_deslizamiento.pkl"
joblib.dump(paquete_modelo, archivo_modelo)
```

```
print(f"Modelo guardado correctamente:
{archivo_modelo}")
```

13. Exportación del paquete base para Soft Voting

```
paquete_base_soft_voting = {
    "best_dt": best_dt,
    "best_rf": best_rf,
    "best_et": best_et,
    "best_gb": best_gb,
    "modelos_individuales": modelos_individuales,
    "resumen_modelos_base": resumen_modelos_base,
    "tabla_metricas_modelos_individuales":
tabla_metricas_modelos_individuales,
    "df_modelo": df_modelo,
    "X": X,
    "y": y,
    "y_encoded": y_encoded,
    "label_encoder": label_encoder,
    "clases_objetivo": list(label_encoder.classes_),
    "columnas_predictoras": columnas_predictoras,
    "variable_objetivo": TARGET,
    "random_state": RANDOM_STATE,
    "n_splits": n_splits,
    "cv_config": {
        "tipo": "StratifiedKfold",
        "n_splits": n_splits,
        "shuffle": True,
        "random_state": RANDOM_STATE,
    },
    "nota_metodologica": (
        "Paquete generado para continuar con
ensambles Soft Voting ponderados. "
        "No requiere repetir BayesSearchCV;
GridSearchCV solo optimiza pesos."
    ),
    "fecha_exportacion":
datetime.now().strftime("%Y-%m-%d %H:%M:%S"),
}

archivo_base_soft_voting =
"paquete_base_soft_voting.pkl"
joblib.dump(paquete_base_soft_voting,
archivo_base_soft_voting)
```

```
print(f"Paquete base para Soft Voting guardado
correctamente: {archivo_base_soft_voting}")
print("Contenido principal del paquete:")
for clave in [
    "best_gb",
```

```
"best_rf",
"best_et",
"best_dt",
"X",
"y_encoded",
"label_encoder",
"columnas_predictoras",
"n_splits",
]:
    valor = paquete_base_soft_voting[clave]
    if hasattr(valor, "shape"):
        print(f"- {clave}: shape={valor.shape}")
    else:
        print(f"- {clave}: {type(valor).__name__}")
```

14. Descarga de resultados desde Colab

```
import tempfile
import zipfile

IN_COLAB = globals().get("IN_COLAB", True)

def elegir_carpeta(titulo):
    import tkinter as tk
    from tkinter import filedialog

    root = tk.Tk()
    root.withdraw()
    root.attributes("-topmost", True)
    ruta = filedialog.askdirectory(title=titulo)
    root.destroy()

    if not ruta:
        raise FileNotFoundError("No selecciono
ninguna carpeta de destino.")

    return Path(ruta)

def buscar_archivo(archivo, carpetas):
    archivo = Path(archivo)
    candidatos = [archivo] if archivo.is_absolute()
    else [archivo] + [
        Path(carpeta) / archivo.name for carpeta in
carpetas
    ]
    return next((ruta for ruta in candidatos if
ruta.exists()), None)

def copiar_archivo(archivo, carpeta_destino,
carpetas_busqueda):
    origen = buscar_archivo(archivo,
carpetas_busqueda)

    if origen is None:
        print(f"No se encontro: {archivo}")
        return

    destino = carpeta_destino / origen.name
    if origen.resolve() != destino.resolve():
        shutil.copy2(origen, destino)

    print(f"Archivo guardado: {destino.resolve()}")

def crear_zip_sin_zips(carpeta_origen,
archivo_zip_destino):
    carpeta_origen = Path(carpeta_origen)
    archivo_zip_destino = Path(archivo_zip_destino)

    if not carpeta_origen.exists():
        raise FileNotFoundError(f"No existe la
carpeta de resultados: {carpeta_origen}")

    if archivo_zip_destino.exists():
        archivo_zip_destino.unlink()

    zip_temporal = Path(tempfile.gettempdir()) /
archivo_zip_destino.name
    if zip_temporal.exists():
        zip_temporal.unlink()

    archivos = [
        ruta for ruta in carpeta_origen.rglob("*")
        if ruta.is_file() and ruta.suffix.lower() !=
".zip"
    ]

    with zipfile.ZipFile(zip_temporal, "w",
compression=zipfile.ZIP_DEFLATED) as zipf:
        for ruta in archivos:
            zipf.write(ruta,
ruta.relative_to(carpeta_origen))
```

```

        shutil.move(str(zip_temporal),
str(archivo_zip_destino))
        print(f"ZIP guardado sin zips anidados:
{archivo_zip_destino.resolve()}")

archivo_zip_base =
"resultados_tesis_riesgo_deslizamiento"

archivos_para_exportar = [
    Path(archivo_resultado_mapa_cv),
    Path(archivo_resultado_modelo_final),
    Path("importancia_variables_riesgo.csv"),

    Path("modelo_ensamble_riesgo_deslizamiento.pkl"),
    Path("paquete_base_soft_voting.pkl"),
    Path(ruta_evaluacion_modelo_individual),
    Path(ruta_modelo_final_modelo_individual),
    Path(ruta_evaluacion_ensamble),
    Path(ruta_modelo_final_ensamble),
]

if IN_COLAB:
    archivo_zip = Path(f"{archivo_zip_base}.zip")
    crear_zip_sin_zips(OUTPUTS_TESIS, archivo_zip)

    for archivo in archivos_para_exportar +
[archivo_zip]:
        archivo = Path(archivo)
        if archivo.exists():
            print(f"Descargando: {archivo}")
            files.download(str(archivo))
        else:
            print(f"No se encontro: {archivo}")
    else:
        carpeta_destino = elegir_carpeta(
            "Seleccione carpeta para exportar resultados
            finales de Stacking"
        )
        carpeta_destino.mkdir(parents=True,
exist_ok=True)
        print(f"Carpeta seleccionada:
{carpeta_destino.resolve()}")

        carpetas_busqueda = [
            OUTPUTS_TESIS,
            DIR_PREDICCIONES,
            DIR_IMPORTANCIA,
            DIR_METRICAS,
            DIR_GRAFICOS,
            DIR_MATRICES,
        ]

        for archivo in archivos_para_exportar:
            copiar_archivo(archivo, carpeta_destino,
carpetas_busqueda)

        crear_zip_sin_zips(
            OUTPUTS_TESIS,
            carpeta_destino / f"{archivo_zip_base}.zip",
        )

```

PARTE II. ENSAMBLES SOFT VOTING PONDERADOS

Instalación de dependencias

```
!pip -q install scikit-learn pandas numpy matplotlib
openpyxl
```

0.1 Importación de librerías

```

import os
import re
import shutil
import warnings
import unicodedata
from datetime import datetime
from pathlib import Path

import joblib
import matplotlib.pyplot as plt
import numpy as np
import pandas as pd

from google.colab import files

```

```

from sklearn.base import clone
from sklearn.ensemble import VotingClassifier
from sklearn.metrics import (
    ConfusionMatrixDisplay,
    accuracy_score,
    balanced_accuracy_score,
    classification_report,
    cohen_kappa_score,

```

```

    confusion_matrix,
    f1_score,
    make_scorer,
    precision_score,
    recall_score,
)
from sklearn.model_selection import GridSearchCV,
StratifiedKFold, cross_val_predict, cross_validate

```

```

warnings.filterwarnings("ignore",
category=FutureWarning)
warnings.filterwarnings("ignore",
category=UserWarning)

```

```

ARCHIVO_BASE_SOFT_VOTING =
"paquete_base_soft_voting.pkl"
RANDOM_STATE = 42

```

1. Carga del paquete base generado por el notebook anterior

```

if not os.path.exists(ARCHIVO_BASE_SOFT_VOTING):
    print(f"No se encontró
{ARCHIVO_BASE_SOFT_VOTING} en la sesión actual.")
    print("Suba el archivo generado por
Notebook Stacking_base_SV.ipynb.")
    uploaded = files.upload()
    if ARCHIVO_BASE_SOFT_VOTING not in uploaded:
        raise FileNotFoundError(
            f"Debe subir exactamente el archivo
{ARCHIVO_BASE_SOFT_VOTING}."
        )

```

```
paquete_sv = joblib.load(ARCHIVO_BASE_SOFT_VOTING)
```

```

print("Paquete base cargado correctamente.")
print("Claves disponibles:")
for clave in sorted(paquete_sv.keys()):
    print("-", clave)

```

2. Recuperación de datos, modelos y validación cruzada

```

X = paquete_sv["X"]
y = paquete_sv["y"]
y_encoded = np.asarray(paquete_sv["y_encoded"])
df_modelo = paquete_sv["df_modelo"]
label_encoder = paquete_sv["label_encoder"]
columnas_predictoras =
paquete_sv["columnas_predictoras"]
TARGET = paquete_sv["variable_objetivo"]
RANDOM_STATE = int(paquete_sv.get("random_state",
RANDOM_STATE))
n_splits = int(paquete_sv.get("n_splits", 10))

best_gb = paquete_sv["best_gb"]
best_rf = paquete_sv["best_rf"]
best_et = paquete_sv["best_et"]
best_dt = paquete_sv.get("best_dt")

```

```

modelos_individuales = {
    "Gradient Boosting": best_gb,
    "Random Forest": best_rf,
    "Extra Trees": best_et,
}
if best_dt is not None:
    modelos_individuales["Decision Tree"] = best_dt

```

```

cv = StratifiedKFold(
    n_splits=n_splits,
    shuffle=True,
    random_state=RANDOM_STATE,
)

```

```

print("Datos recuperados:")
print(f"- X: {X.shape}")
print(f"- y_encoded: {y_encoded.shape}")
print(f"- Clases: {list(label_encoder.classes_)}")
print(f"- Validación cruzada: StratifiedKFold con
{n_splits} particiones")

```

```

print("\nModelos individuales cargados:")
for nombre in modelos_individuales:
    print("-", nombre)

```

3. Configuración de resultados y funciones auxiliares

```

OUTPUTS_SV = Path("outputs_soft_voting")
DIR_METRICAS = OUTPUTS_SV / "metricas"
DIR_GRAFICOS = OUTPUTS_SV / "graficos"
DIR_MATRICES = OUTPUTS_SV / "matrices_confusion"
DIR_MODELOS = OUTPUTS_SV / "modelos"
DIR_PREDICCIONES = OUTPUTS_SV / "predicciones"
DIR_GRID = OUTPUTS_SV / "grid_search"

```

```

for carpeta in [DIR_METRICAS, DIR_GRAFICOS,
DIR_MATRICES, DIR_MODELOS, DIR_PREDICCIONES,
DIR_GRID]:
    carpeta.mkdir(parents=True, exist_ok=True)

def guardar_tabla(tabla, ruta_base):
    ruta_base = Path(ruta_base)
    csv_path = ruta_base.with_suffix(".csv")
    xlsx_path = ruta_base.with_suffix(".xlsx")

    tabla.to_csv(csv_path, index=False,
encoding="utf-8-sig")
    try:
        tabla.to_excel(xlsx_path, index=False)
    except Exception as exc:
        print(f"No se pudo guardar Excel para
{xlsx_path.name}. Se conserva CSV. Detalle: {exc}")

    return csv_path, xlsx_path

def nombre_archivo_seguro(texto):
    texto = str(texto).strip().replace(" ", "_")
    texto = unicodedata.normalize("NFKD",
texto).encode("ascii", "ignore").decode("ascii")
    texto = re.sub(r"[^A-Za-z0-9_]+", "_", texto)
    texto = re.sub(r"_+", "_", texto).strip("_")
    return texto or "modelo"

def calcular_metricas(y_real, y_pred):
    return {
        "Accuracy": accuracy_score(y_real, y_pred),
        "Balanced Accuracy":
balanced_accuracy_score(y_real, y_pred),
        "Precision_Macro": precision_score(y_real,
y_pred, average="macro", zero_division=0),
        "Recall_Macro": recall_score(y_real, y_pred,
average="macro", zero_division=0),
        "F1_Macro": f1_score(y_real, y_pred,
average="macro", zero_division=0),
        "F1_Weighted": f1_score(y_real, y_pred,
average="weighted", zero_division=0),
        "Cohen_Kappa": cohen_kappa_score(y_real,
y_pred),
    }

4. Definición de los ensambles Soft Voting
sv_1 = VotingClassifier(
    estimators=[
        ("gb", clone(best_gb)),
        ("rf", clone(best_rf)),
    ],
    voting="soft",
    n_jobs=-1,
)

sv_2 = VotingClassifier(
    estimators=[
        ("gb", clone(best_gb)),
        ("rf", clone(best_rf)),
        ("et", clone(best_et)),
    ],
    voting="soft",
    n_jobs=-1,
)

ensambles_soft_voting = {
    "SV_1": sv_1,
    "SV_2": sv_2,
}

tabla_definicion_sv = pd.DataFrame([
    {
        "Ensamble": "SV_1",
        "Tipo": "Soft Voting ponderado",
        "Modelos": "Gradient Boosting + Random
Forest",
        "Parametro_optimizado": "weights",
    },
    {
        "Ensamble": "SV_2",
        "Tipo": "Soft Voting ponderado",
        "Modelos": "Gradient Boosting + Random
Forest + Extra Trees",
        "Parametro_optimizado": "weights",
    },
])

display(tabla_definicion_sv)
guardar_tabla(tabla_definicion_sv, DIR_METRICAS /
"definicion_ensambles_soft_voting")

```

5. Búsqueda automática de pesos con GridSearchCV

```

pesos_sv_1 = sorted(set(
    [(gb, rf) for gb in range(1, 9) for rf in
range(1, 4)]
    + [(1, 1), (2, 1), (3, 1), (4, 1), (5, 1), (6,
1), (8, 1), (10, 1)]
))

pesos_sv_2 = sorted(set(
    [(gb, rf, et) for gb in range(1, 8) for rf in
range(1, 4) for et in range(1, 4)]
    + [
        (1, 1, 1),
        (2, 1, 1),
        (3, 1, 1),
        (4, 1, 1),
        (5, 1, 1),
        (6, 1, 1),
        (8, 1, 1),
        (10, 1, 1),
        (4, 2, 1),
        (4, 1, 2),
        (5, 2, 1),
        (5, 1, 2),
    ]
))

param_grids = {
    "SV_1": {"weights": pesos_sv_1},
    "SV_2": {"weights": pesos_sv_2},
}

print(f"Combinaciones de pesos para SV_1:
{len(pesos_sv_1)}")
print(f"Combinaciones de pesos para SV_2:
{len(pesos_sv_2)}")

grid_searches_sv = {}
filas_resumen_grid = []

for nombre, estimador in
ensambles_soft_voting.items():
    print(f"\nOptimizando pesos de {nombre} con
GridSearchCV...")
    grid = GridSearchCV(
        estimator=estimador,
        param_grid=param_grids[nombre],
        scoring="f1_macro",
        cv=cv,
        n_jobs=-1,
        refit=True,
        return_train_score=True,
        error_score="raise",
        verbose=0,
    )
    grid.fit(X, y_encoded)
    grid_searches_sv[nombre] = grid

    resultados_grid =
pd.DataFrame(grid.cv_results_).sort_values("rank_tes
t_score")
    guardar_tabla(resultados_grid, DIR_GRID /
f"resultados_grid_{nombre}")

    filas_resumen_grid.append({
        "Ensamble": nombre,
        "Mejores_pesos":
grid.best_params_["weights"],
        "Mejor_F1_Macro_CV": grid.best_score_,
        "Combinaciones_evaluadas":
len(resultados_grid),
    })

    print(f"Mejores pesos para {nombre}:
{grid.best_params_['weights']}")
    print(f"Mejor F1 macro CV para {nombre}:
{grid.best_score_:.4f}")

tabla_resumen_grid =
pd.DataFrame(filas_resumen_grid).sort_values(
    "Mejor_F1_Macro_CV",
    ascending=False,
).reset_index(drop=True)

guardar_tabla(tabla_resumen_grid, DIR_GRID /
"resumen_gridsearch_soft_voting")
display(tabla_resumen_grid)

```

6. Evaluación final de SV_1 y SV_2 con cross_val_predict

```
etiquetas_codificadas =
np.arange(len(label_encoder.classes_))
etiquetas_clases = label_encoder.classes_

def evaluar_con_cross_val_predict(estimador, nombre,
tipo):
    print(f"Evaluando {nombre} con
cross_val_predict...")
    y_pred = cross_val_predict(
        estimador,
        X,
        y_encoded,
        cv=cv,
        n_jobs=-1,
        method="predict",
    )

    metricas = calcular_metricas(y_encoded, y_pred)
    matriz = confusion_matrix(y_encoded, y_pred,
labels=etiquetas_codificadas)
    matriz_norm = confusion_matrix(
        y_encoded,
        y_pred,
        labels=etiquetas_codificadas,
        normalize="true",
    )

    return {
        "Nombre": nombre,
        "Tipo": tipo,
        "Estimador": estimador,
        "y_pred": y_pred,
        "metricas": metricas,
        "matriz": matriz,
        "matriz_norm": matriz_norm,
    }

mejor_sv_1 =
grid_searches_sv["SV_1"].best_estimator_
mejor_sv_2 =
grid_searches_sv["SV_2"].best_estimator_

resultados_finales = [
    evaluar_con_cross_val_predict(best_gb, "Gradient
Boosting", "Modelo individual baseline"),
    evaluar_con_cross_val_predict(mejor_sv_1,
"SV_1", "Soft Voting"),
    evaluar_con_cross_val_predict(mejor_sv_2,
"SV_2", "Soft Voting"),
]

tabla_metricas_finales_sv = pd.DataFrame([
    {
        "Nombre": resultado["Nombre"],
        "Tipo": resultado["Tipo"],
        **resultado["metricas"],
    }
    for resultado in resultados_finales
]).sort_values(
    by=["F1_Macro", "Balanced_Accuracy"],
    ascending=False,
).reset_index(drop=True)

guardar_tabla(tabla_metricas_finales_sv,
DIR_METRICAS / "metricas_finales_soft_voting")

print("Métricas finales ordenadas por F1_Macro:")
display(tabla_metricas_finales_sv)

mejor_soft_voting_nombre =
tabla_metricas_finales_sv[
    tabla_metricas_finales_sv["Tipo"].eq("Soft
Voting")
].iloc[0]["Nombre"]

mejor_soft_voting = mejor_sv_1 if
mejor_soft_voting_nombre == "SV_1" else mejor_sv_2
print(f"Mejor ensamble Soft Voting según F1_Macro:
{mejor_soft_voting_nombre}")

7. Gráfico comparativo de métricas
metricas_grafico = ["Accuracy", "F1_Macro",
"Cohen_Kappa"]
etiquetas_metricas = {
    "Accuracy": "Accuracy",
    "F1_Macro": "F1 macro",
    "Cohen_Kappa": "Cohen's Kappa",
}
```

```
colores = {
    "Accuracy": "#2E86AB",
    "F1_Macro": "#F18F01",
    "Cohen_Kappa": "#6A994E",
}

datos =
tabla_metricas_finales_sv.sort_values("F1_Macro",
ascending=True).copy()
y_pos = np.arange(len(datos))
altura = 0.22

fig, ax = plt.subplots(figsize=(10, max(4.5,
len(datos) * 0.8)))

for i, metrica in enumerate(metricas_grafico):
    valores =
datos[metrica].astype(float).to_numpy()
    barras = ax.barh(
        y_pos + (i - 1) * altura,
        valores,
        height=altura,
        label=etiquetas_metricas[metrica],
        color=colores[metrica],
        alpha=0.9,
        edgecolor="black",
        linewidth=0.4,
    )
    ax.bar_label(barras, labels=[f"{valor:.3f}" for
valor in valores], padding=3, fontsize=8)

minimo = float(datos[metricas_grafico].min().min())
maximo = float(datos[metricas_grafico].max().max())
ax.set_xlim(min(0, minimo - 0.05), max(1.0, maximo +
0.08))
ax.set_yticks(y_pos)
ax.set_yticklabels(datos["Nombre"])
ax.set_xlabel("Valor de la métrica")
ax.set_title("Comparación: Gradient Boosting vs Soft
Voting")
ax.grid(axis="x", alpha=0.25)
ax.legend(loc="lower right")
plt.tight_layout()

ruta_grafico = DIR_GRAFICOS /
"comparacion_gradient_boosting_soft_voting.png"
fig.savefig(ruta_grafico, dpi=300,
bbox_inches="tight")
plt.show()

print(f"Gráfico guardado en: {ruta_grafico}")

8. Matrices de confusión y reportes por clase
for resultado in resultados_finales:
    nombre = resultado["Nombre"]
    nombre_seguro = nombre_archivo_seguro(nombre)
    y_pred = resultado["y_pred"]
    matriz = resultado["matriz"]
    matriz_norm = resultado["matriz_norm"]

    matriz_df = pd.DataFrame(matriz,
index=etiquetas_clases, columns=etiquetas_clases)
    matriz_norm_df = pd.DataFrame(matriz_norm,
index=etiquetas_clases,
columns=etiquetas_clases).round(4)

    guardar_tabla(
        matriz_df.rename_axis("Clase_real").reset_index(),
        DIR_MATRICES /
        f"matriz_confusion_absoluta_{nombre_seguro}",
    )
    guardar_tabla(
        matriz_norm_df.rename_axis("Clase_real").reset_index
(),
        DIR_MATRICES /
        f"matriz_confusion_normalizada_{nombre_seguro}",
    )

    reporte = classification_report(
        y_encoded,
        y_pred,
        labels=etiquetas_codificadas,
        target_names=etiquetas_clases,
        zero_division=0,
        output_dict=True,
    )
    tabla_reporte =
pd.DataFrame(reporte).transpose().reset_index().rena
me(columns={"index": "Clase"})
```

```

guardar_tabla(tabla_reporte, DIR_METRICAS /
f"classification_report_{nombre_seguro}")

fig, axes = plt.subplots(1, 2, figsize=(15, 6))
ConfusionMatrixDisplay(
    confusion_matrix=matriz,
    display_labels=etiquetas_clases,
).plot(cmap="Blues", ax=axes[0],
values_format="d", colorbar=False)
axes[0].set_title("Matriz absoluta")
axes[0].tick_params(axis="x", rotation=45)

ConfusionMatrixDisplay(
    confusion_matrix=matriz_norm,
    display_labels=etiquetas_clases,
).plot(cmap="Blues", ax=axes[1],
values_format=".2f", colorbar=False)
axes[1].set_title("Matriz normalizada")
axes[1].tick_params(axis="x", rotation=45)

fig.suptitle(nombre, fontsize=14)
plt.tight_layout()

ruta_matriz = DIR_MATRICES /
f"matrices_confusion_{nombre_seguro}.png"
fig.savefig(ruta_matriz, dpi=300,
bbox_inches="tight")
plt.show()

print(f"Resultados guardados para {nombre}")

```

9. Predicciones vs base de datos para el mejor Soft Voting

```

# 9.1 Predicciones de evaluación: CENEPRED vs
predicción fuera de muestra
resultado_mejor_soft_voting = next(
    resultado for resultado in resultados_finales
    if resultado["Nombre"] ==
mejor_soft_voting_nombre
)

y_pred_cv_mejor_soft_voting =
resultado_mejor_soft_voting["y_pred"]
predicciones_cv_clase =
label_encoder.inverse_transform(
    y_pred_cv_mejor_soft_voting.astype(int)
)

df_evaluacion_sv = df_modelo.copy()
df_evaluacion_sv["NIVEL-DE-RIESGO-PREDICHO-CV-SOFT-
VOTING"] = predicciones_cv_clase
df_evaluacion_sv["PREDICCION_CV_CORRECTA_SOFT_VOTING"] = (
    df_evaluacion_sv[TARGET].astype(str).to_numpy()
    == predicciones_cv_clase
)

ruta_evaluacion_sv = DIR_PREDICCIONES /
"01_evaluacion_CENEPRED_vs_SV2_cross_val_predict.csv"

df_evaluacion_sv.to_csv(ruta_evaluacion_sv,
index=False, encoding="utf-8-sig")

print("Archivo de evaluación generado:")
print(ruta_evaluacion_sv)
print("Correctas CV:")
int(df_evaluacion_sv["PREDICCION_CV_CORRECTA_SOFT_VO
TING"].sum())
print("Incorrectas CV:")
int((~df_evaluacion_sv["PREDICCION_CV_CORRECTA_SOFT_
VOTING"]).sum())
display(df_evaluacion_sv.head())

# 9.2 Predicciones del modelo final entrenado con
toda la base
mejor_soft_voting_final = clone(mejor_soft_voting)
mejor_soft_voting_final.fit(X, y_encoded)

predicciones_final_codificadas =
mejor_soft_voting_final.predict(X)
predicciones_final_clase =
label_encoder.inverse_transform(
    predicciones_final_codificadas.astype(int)
)

df_modelo_final_sv = df_modelo.copy()
df_modelo_final_sv["NIVEL-DE-RIESGO-PREDICHO-MODELO-
FINAL-SOFT-VOTING"] = predicciones_final_clase
df_modelo_final_sv["PREDICCION_MODELO_FINAL_CORRECTA_
SOFT_VOTING"] = (

```

```

df_modelo_final_sv[TARGET].astype(str).to_numpy() ==
predicciones_final_clase
)

def normalizar_nombre_probabilidad(clase):
    texto = str(clase).strip().upper().replace("-",
    "_").replace(" ", "_")
    texto = unicodedata.normalize("NFKD",
    texto).encode("ascii", "ignore").decode("ascii")
    texto = re.sub(r"^[^A-Z0-9_]+", "_", texto)
    texto = re.sub(r"_+", "_", texto).strip("_")
    return f"PROB_FINAL_{texto}"

if hasattr(mejor_soft_voting_final,
"predict_proba"):
    probabilidades =
mejor_soft_voting_final.predict_proba(X)
    for indice_clase, clase in
enumerate(label_encoder.classes_):
        nombre_columna =
normalizar_nombre_probabilidad(clase)
        df_modelo_final_sv[nombre_columna] =
probabilidades[:, indice_clase]

ruta_modelo_final_sv = DIR_PREDICCIONES /
"02_modelo_final_SV2_entrenado_con_toda_la_base.csv"
df_modelo_final_sv.to_csv(ruta_modelo_final_sv,
index=False, encoding="utf-8-sig")

print("\nArchivo de modelo final generado:")
print(ruta_modelo_final_sv)
print("Correctas modelo final:")
int(df_modelo_final_sv["PREDICCION_MODELO_FINAL_CORR
ECTA_SOFT_VOTING"].sum())
print("Incorrectas modelo final:")
int((~df_modelo_final_sv["PREDICCION_MODELO_FINAL_CO
RRECTA_SOFT_VOTING"]).sum())
display(df_modelo_final_sv.head())

```

10. Guardado del mejor Soft Voting

```

metricas_mejor_soft_voting =
tabla_metricas_finales_sv[

tabla_metricas_finales_sv["Nombre"].eq(mejor_soft_vo
ting_nombre)
].iloc[0].to_dict()

paquete_modelo_soft_voting = {
    "modelo": mejor_soft_voting_final,
    "nombre_modelo": mejor_soft_voting_nombre,
    "mejores_pesos":
grid_searches_sv[mejor_soft_voting_nombre].best_para
ms_["weights"],
    "label_encoder": label_encoder,
    "columnas_predictoras": columnas_predictoras,
    "clases_objetivo": list(label_encoder.classes_),
    "variable_objetivo": TARGET,
    "metricas": metricas_mejor_soft_voting,
    "tabla_metricas_finales_sv":
tabla_metricas_finales_sv,
    "tabla_resumen_grid": tabla_resumen_grid,
    "fecha_entrenamiento":
datetime.now().strftime("%Y-%m-%d %H:%M:%S"),
}

archivo_modelo_soft_voting =
"modelo_soft_voting_riesgo_deslizamiento.pkl"
joblib.dump(paquete_modelo_soft_voting,
archivo_modelo_soft_voting)
joblib.dump(paquete_modelo_soft_voting, DIR_MODELOS
/ archivo_modelo_soft_voting)

print(f"Modelo Soft Voting guardado correctamente:
{archivo_modelo_soft_voting}")
print(f"Mejor Soft Voting:
{mejor_soft_voting_nombre}")
print(f"Mejores pesos:
{paquete_modelo_soft_voting['mejores_pesos']}")

```

11. Descarga de resultados desde Colab

```

import tempfile
import zipfile

IN_COLAB = globals().get("IN_COLAB", True)

def elegir_carpeta(titulo):
    import tkinter as tk
    from tkinter import filedialog

    root = tk.Tk()
    root.withdraw()
    root.attributes("--topmost", True)

```

```

ruta = filedialog.askdirectory(title=titulo)
root.destroy()

if not ruta:
    raise FileNotFoundError("No selecciono
ninguna carpeta de destino.")

return Path(ruta)

def buscar_archivo(archivo, carpetas):
    archivo = Path(archivo)
    candidatos = [archivo] if archivo.is_absolute()
else [archivo] + [
    Path(carpeta) / archivo.name for carpeta in
carpetas
]
    return next((ruta for ruta in candidatos if
ruta.exists()), None)

def copiar_archivo(archivo, carpeta_destino,
carpetas_busqueda):
    origen = buscar_archivo(archivo,
carpetas_busqueda)

    if origen is None:
        print(f"No se encontro: {archivo}")
        return

    destino = carpeta_destino / origen.name
    if origen.resolve() != destino.resolve():
        shutil.copy2(origen, destino)

    print(f"Archivo guardado: {destino.resolve()}")

def crear_zip_sin_zips(carpeta_origen,
archivo_zip_destino):
    carpeta_origen = Path(carpeta_origen)
    archivo_zip_destino = Path(archivo_zip_destino)

    if not carpeta_origen.exists():
        raise FileNotFoundError(f"No existe la
carpeta de resultados: {carpeta_origen}")

    if archivo_zip_destino.exists():
        archivo_zip_destino.unlink()

    zip_temporal = Path(tempfile.gettempdir()) /
archivo_zip_destino.name
    if zip_temporal.exists():
        zip_temporal.unlink()

    archivos = [
        ruta for ruta in carpeta_origen.rglob("**")
        if ruta.is_file() and ruta.suffix.lower() !=
".zip"
    ]

    with zipfile.ZipFile(zip_temporal, "w",
compression=zipfile.ZIP_DEFLATED) as zipf:
        for ruta in archivos:
            zipf.write(ruta,
ruta.relative_to(carpeta_origen))

    shutil.move(str(zip_temporal),
str(archivo_zip_destino))
    print(f"ZIP guardado sin zips anidados:
{archivo_zip_destino.resolve()}")

archivo_zip_base =
"resultados_soft_voting_riesgo_deslizamiento"

archivos_para_exportar = [
    Path(ruta_evaluacion_sv),
    Path(ruta_modelo_final_sv),
    Path(archivo_modelo_soft_voting),
]

if IN_COLAB:
    archivo_zip = Path(f"{archivo_zip_base}.zip")
    crear_zip_sin_zips(OUTPUTS_SV, archivo_zip)

    for archivo in archivos_para_exportar +
[archivo_zip]:
        archivo = Path(archivo)
        if archivo.exists():
            print(f"Descargando: {archivo}")
            files.download(str(archivo))
        else:
            print(f"No se encontro: {archivo}")
else:

```

```

carpeta_destino = elegir_carpeta(
    "Selecione carpeta para exportar resultados
finales de Soft Voting"
)
carpeta_destino.mkdir(parents=True,
exist_ok=True)
print(f"Carpeta seleccionada:
{carpeta_destino.resolve()}")

carpetas_busqueda = [
    OUTPUTS_SV,
    DIR_PREDICCIONES,
    DIR_MODELOS,
    DIR_METRICAS,
    DIR_GRAFICOS,
    DIR_MATRICES,
    DIR_GRID,
]

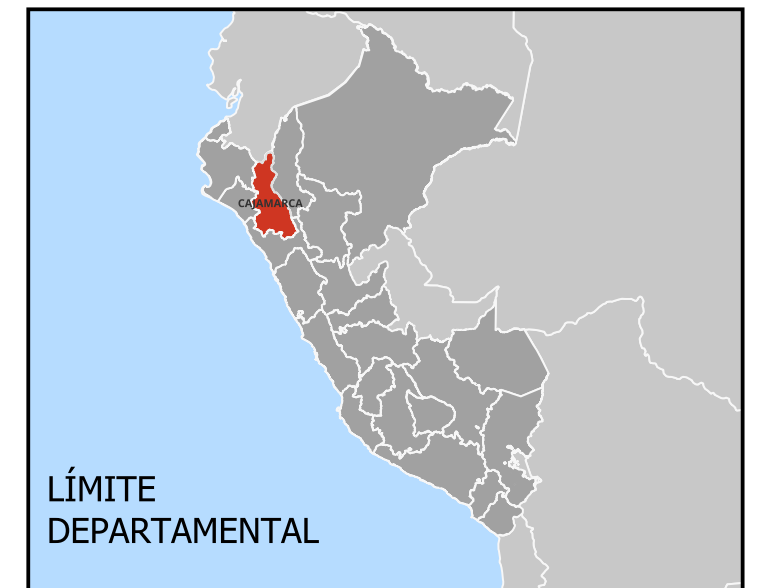
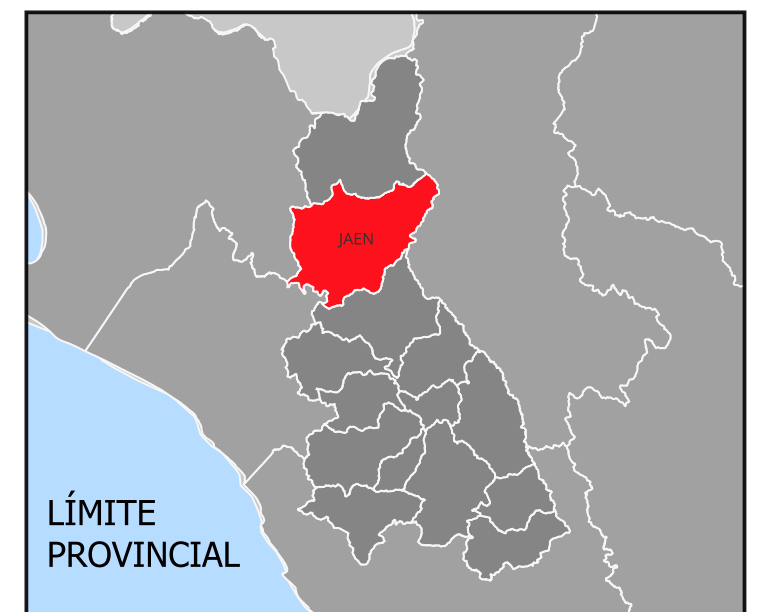
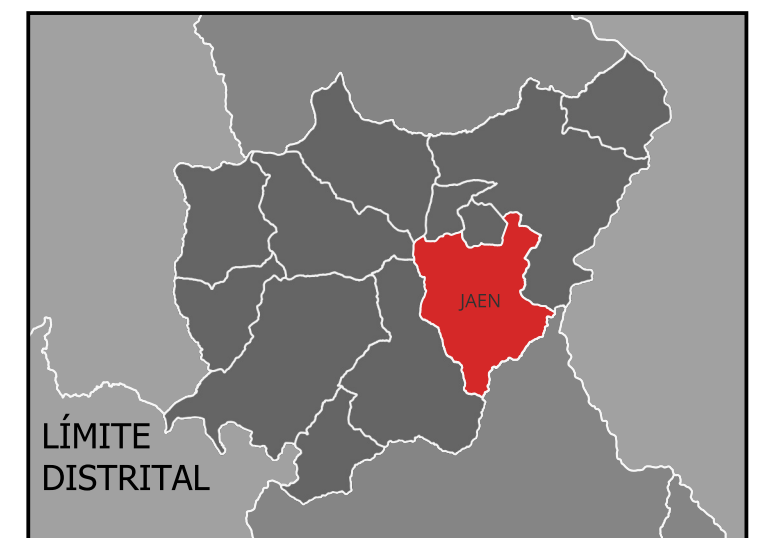
for archivo in archivos_para_exportar:
    copiar_archivo(archivo, carpeta_destino,
carpetas_busqueda)

crear_zip_sin_zips(
    OUTPUTS_SV,
    carpeta_destino / f"{archivo_zip_base}.zip",
)

```

Anexo 6

Mapa de ubicación de la zona de estudio



MAPA DE UBICACIÓN		
ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025	DATUM	WGS 84 / UTM 17S
DEPARTAMENTO	CAJAMARCA	MU -01
PROVINCIA	JAÉN	
DISTRITO	JAÉN	

Anexo 7

Mapa de elementos expuestos de la zona de estudio

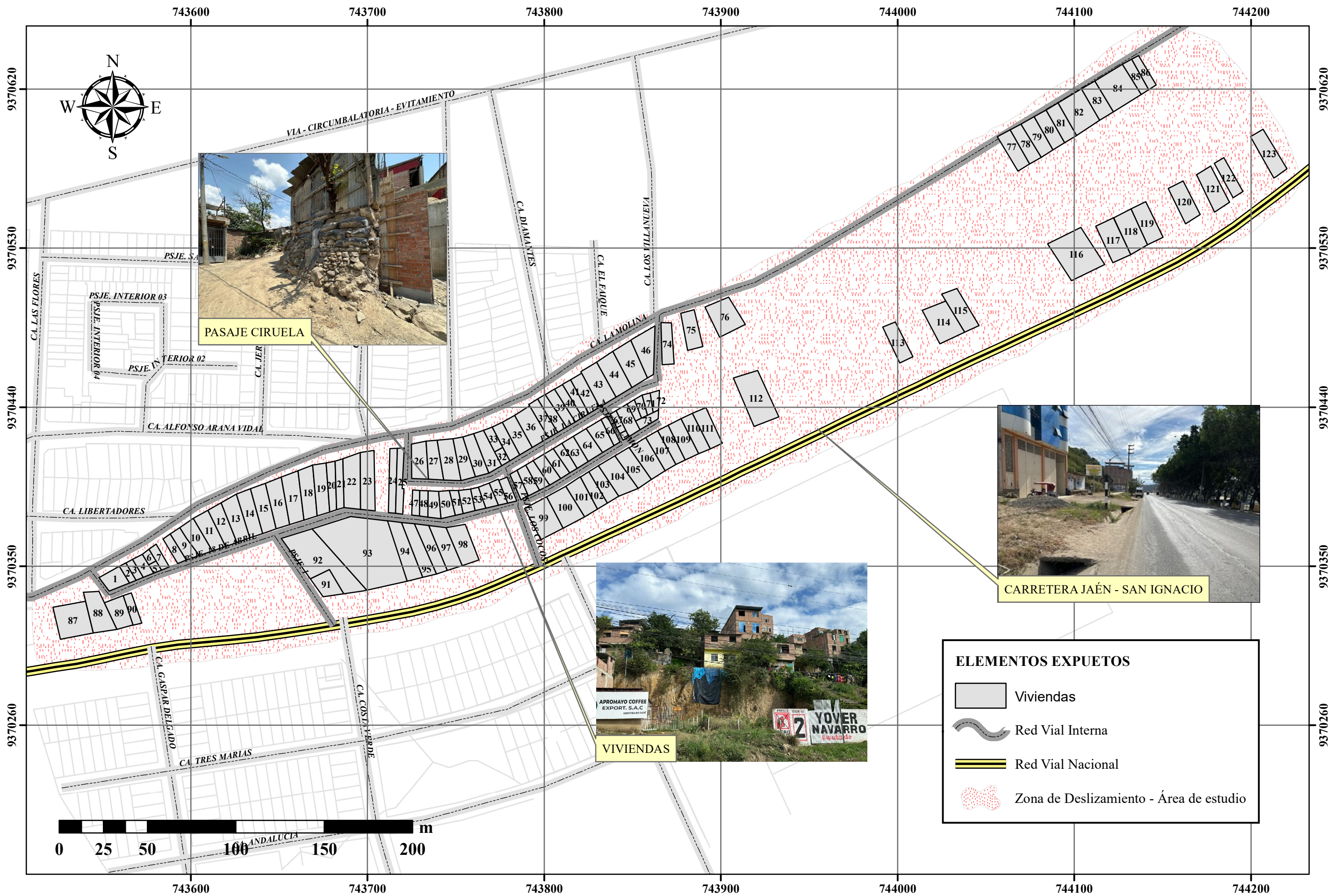
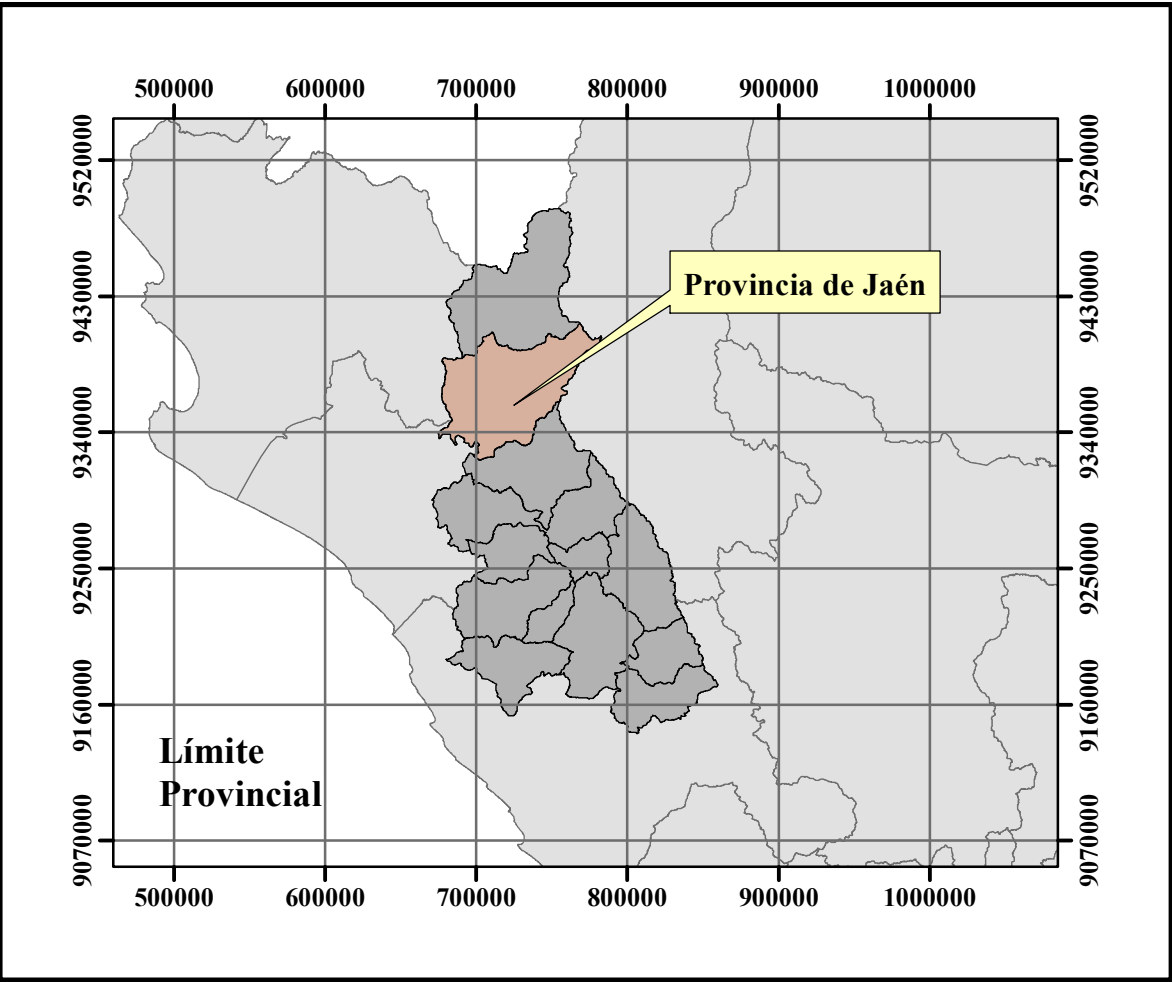
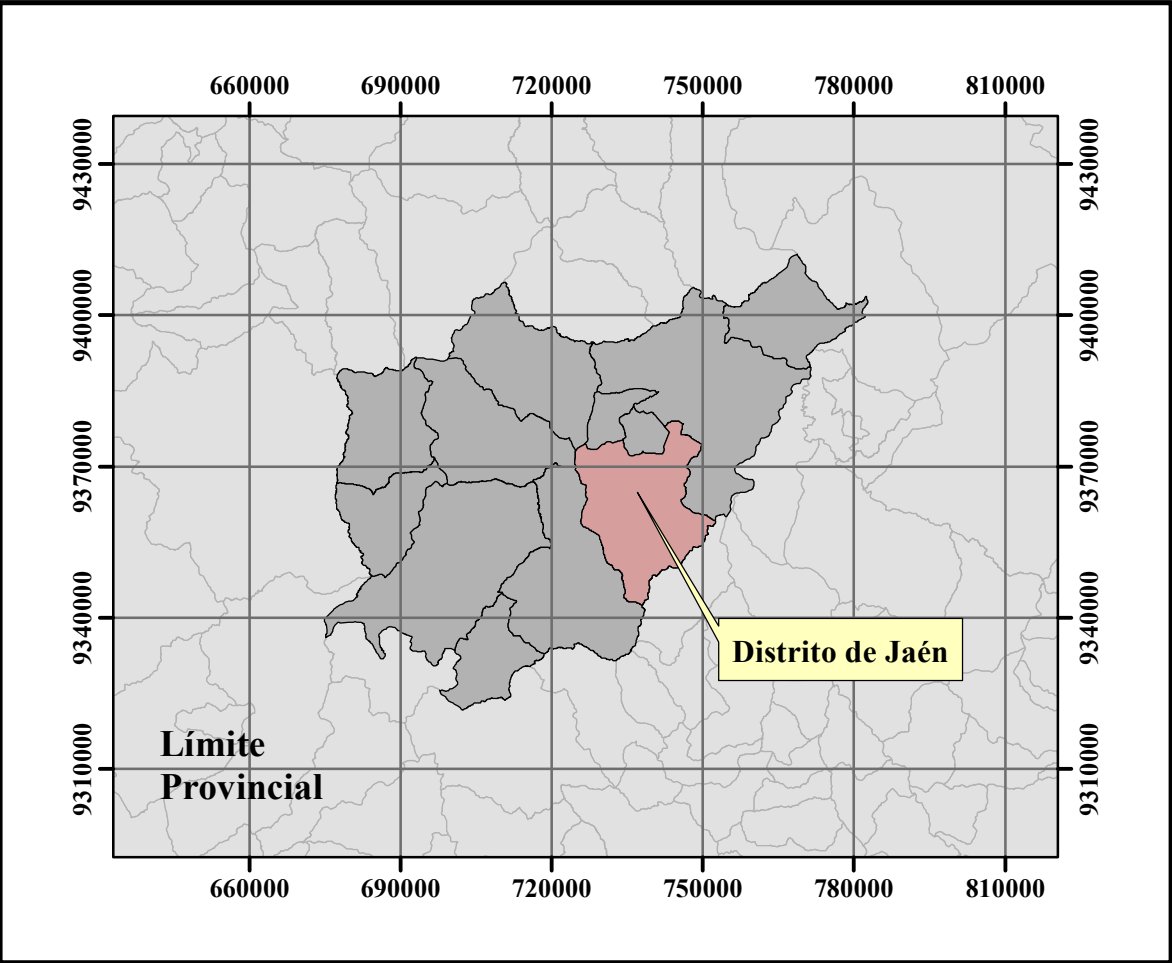


Tabla 1. Población y vivienda del Sector la Colina - Zona de estudio		
Descripción	N° Viviendas	N° Población
N° Viviendas encuestadas	123	508
Viviendas particulares	123	508
Viviendas Colectivas	0	0
Total	123	508

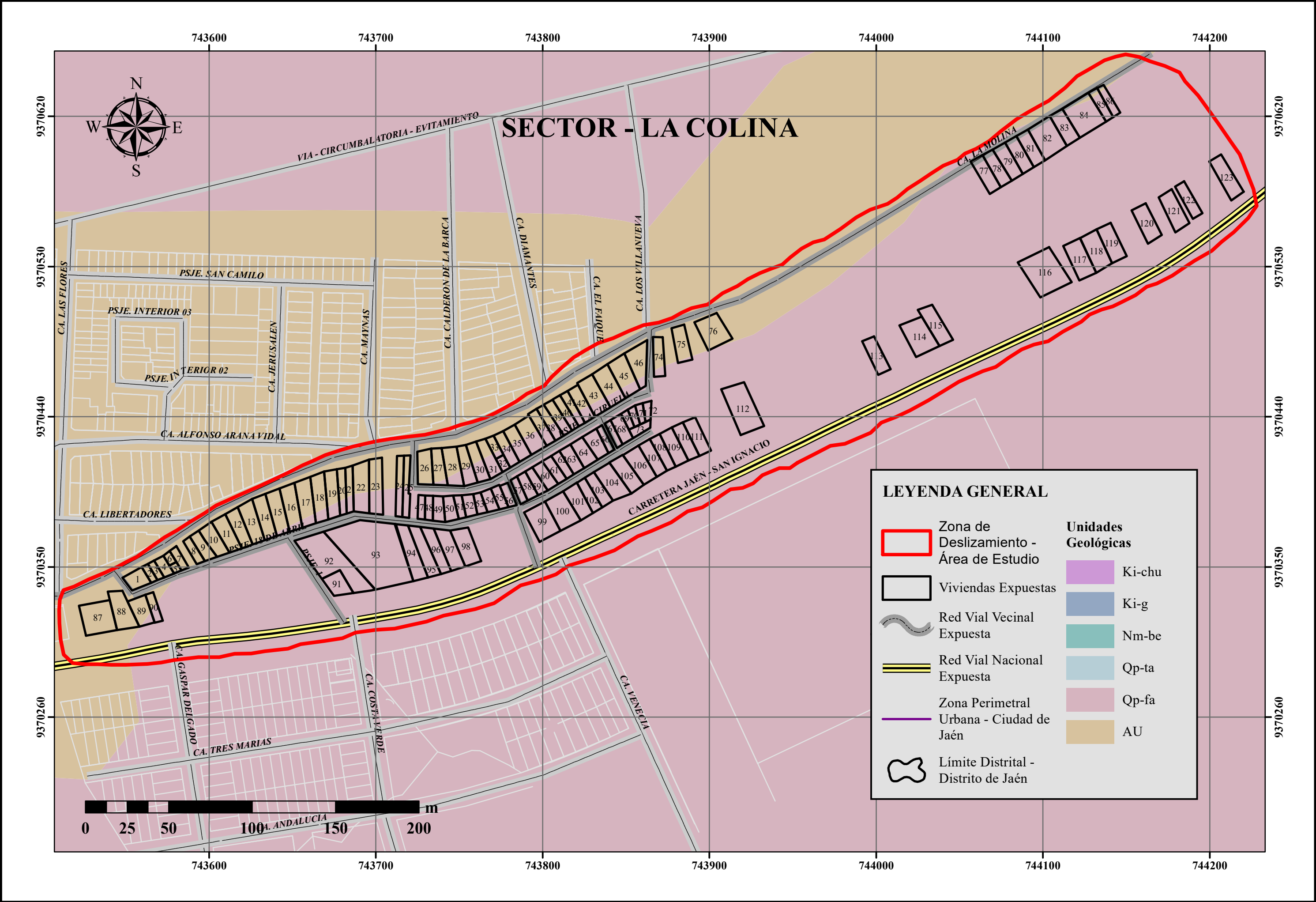
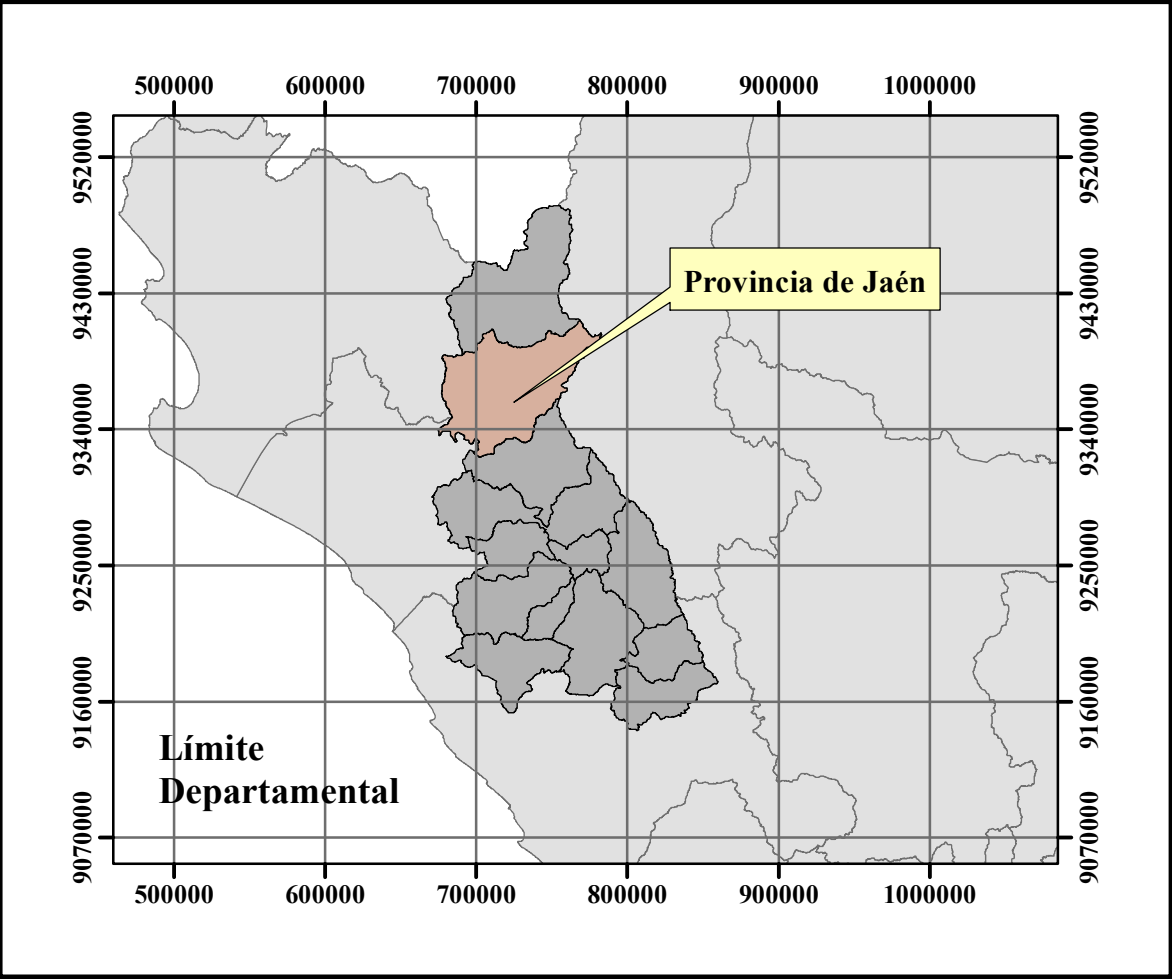
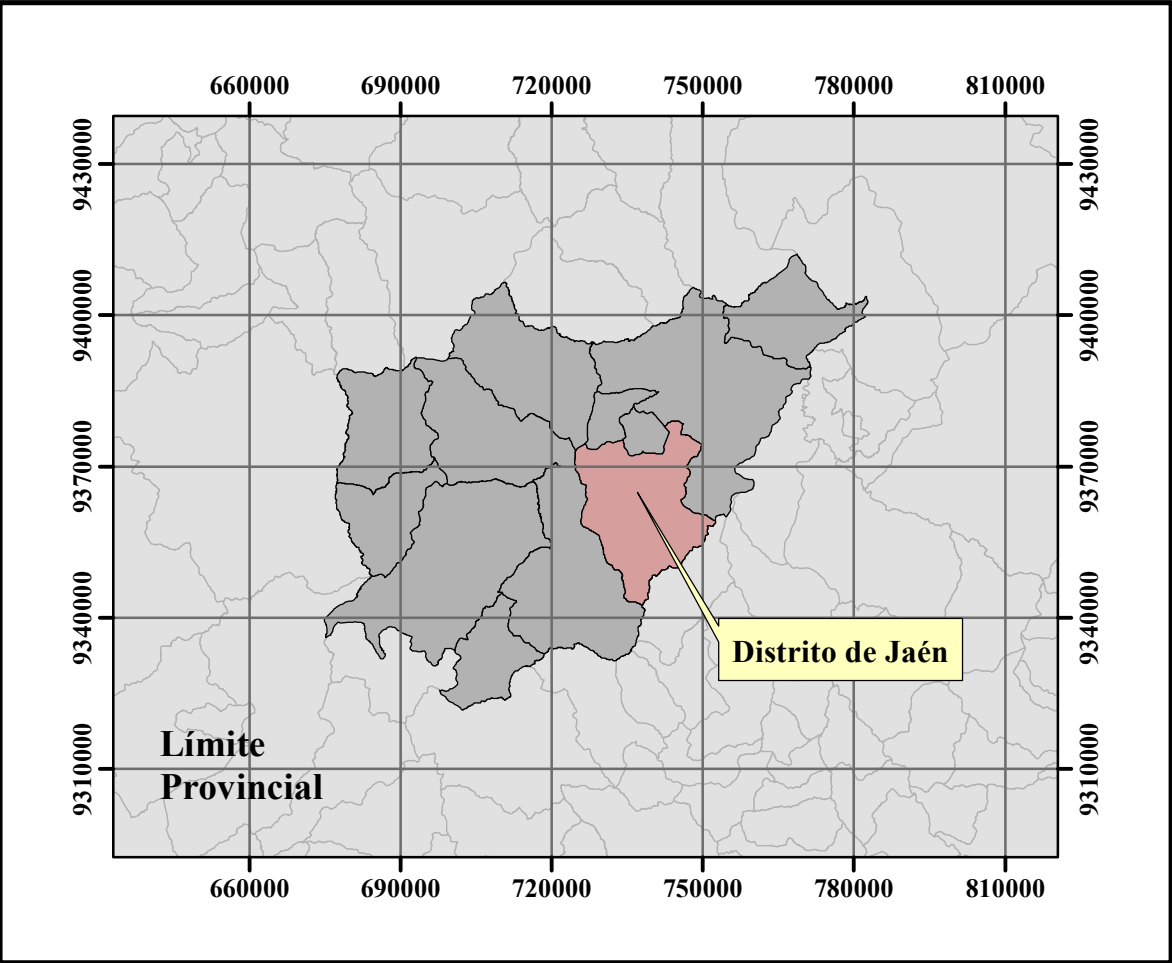
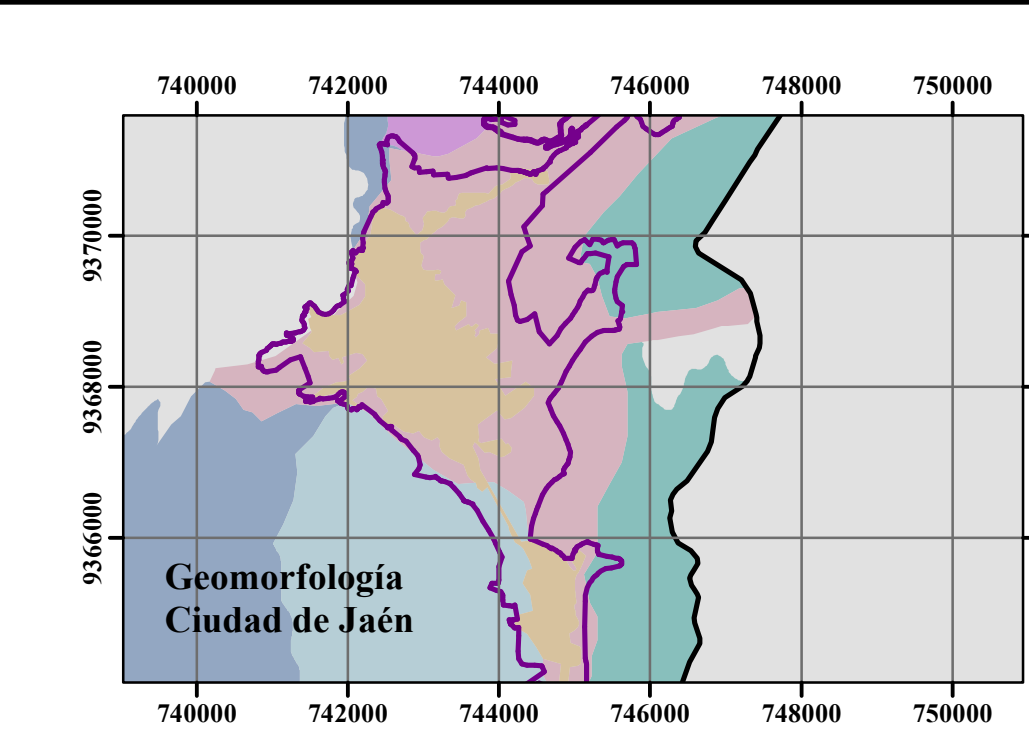
Tabla 2. Población por sexo del Sector la Colina - Zona de estudio		
Sexo	Población	%
Masculina	262	51.6%
Femenina	246	48.4%
Población total	508	100.0%

Tabla 3. Vías de acceso expuestas				
	Nombre	Tipo	Longitud (m)	Estado actual
Red vial nacional	Red vial nacional, Carretera Jaén- San Ignacio	Nacional	920.21	Bueno
	Ca. La Molina	Urbano	874.14	Bueno
Red vial interna	Psje. El Limon	Urbano	22.64	Regular
	Psje. 1	Urbano	54.15	Malo
	Psj. Los Cocos	Urbano	56.04	Malo
	Psje. 18 de Abril	Urbano	351.48	Malo
	Psje. La Ciruela	Urbano	221.93	Malo

UNIVERSIDAD NACIONAL DE JAÉN		
CARRERA PROFESIONAL DE INGENIERÍA CIVIL		
Tesis:	"ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN – CAJAMARCA, 2025"	
Lámina:	PLANO DE ELEMENTOS EXPUESTOS EN EL SECTOR LA COLINA DE LA CIUDAD DE JAÉN (ZONA DE ESTUDIO)	REGIÓN: CAJAMARCA PROVINCIA: JAÉN
Responsables:	MONSALVE TICLIAHUANCA EINSTEIN EDINSON VÍLCHEZ SILVA FRANKLIN FERNANDO	DISTRITO: JAÉN FECHA: Febrero-2026
Asesores:	Mg. Ing. Billy Alexis Cayatopa Calderón Dr. Lenin Quiñones Huatangari	ESCALA: INDICADA PLANO: ME - 01
DATUM: WGS - 84		SIST. DE PROYEC: UTM
		ZONA 17

Anexo 8

Mapa espacial del factor condicionante – Geología



CÓDIGO	UNIDAD GEOLÓGICA	ERA	PERIODO	EPOCA	TIPO ROCA	LITOLOGIA	UNIDAD HIDROGRÁFICA
Ki-chu	Formación Chulec	Mesozoico	Cretaceo	Inferior	Sedimentaria	Calizas, lutitas, margas	Acuitardo sedimentario
Ki-g	Grupo Goyllarisquiza	Mesozoico	Cretaceo	Inferior	Sedimentaria	Cuarzitas, areniscas	Acuífero fisurado sedimentario
Nm-be	Formación Bellavista	Cenozoico	Neógeno	Miocena	Sedimentaria	Conglomerados, areniscas, lutitas	Acuífero poroso consolidado
Qp-ta	Formación Tamborapa	Cenozoico	Cuaternario	Pleistoceno	Sedimentaria	Conglomerados, areniscas, lutitas	Acuífero poroso consolidado
Qp-fa	Depósitos Fluvioaluviales	Cenozoico	Cuaternario	Pleistoceno	Sedimentaria	Gravas, arenas, arcillas, limos	Acuífero poroso no consolidado
AU	Casco Urbano (Depósitos Fluvioaluviales)	Cenozoico	Cuaternario	Pleistoceno	Sedimentaria	Gravas, arenas, arcillas, limos	Acuífero poroso no consolidado



UNIVERSIDAD NACIONAL DE JAÉN

CARRERA PROFESIONAL DE INGENIERÍA CIVIL

Tesis: "ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025"

Lámina: PLANO DE UNIDADES GEOLÓGICAS EN EL SECTOR LA COLINA DE LA CIUDAD DE JAÉN (ZONA DE ESTUDIO)

Responsables: MONSALVE TICLIAHUANCA EINSTEIN EDINSON
VÍLCHEZ SILVA FRANKLIN FERNANDO

Asesores: Mg. Ing. Billy Alexis Cayatopa Calderón
Dr. Lenin Quiñones Huatangari

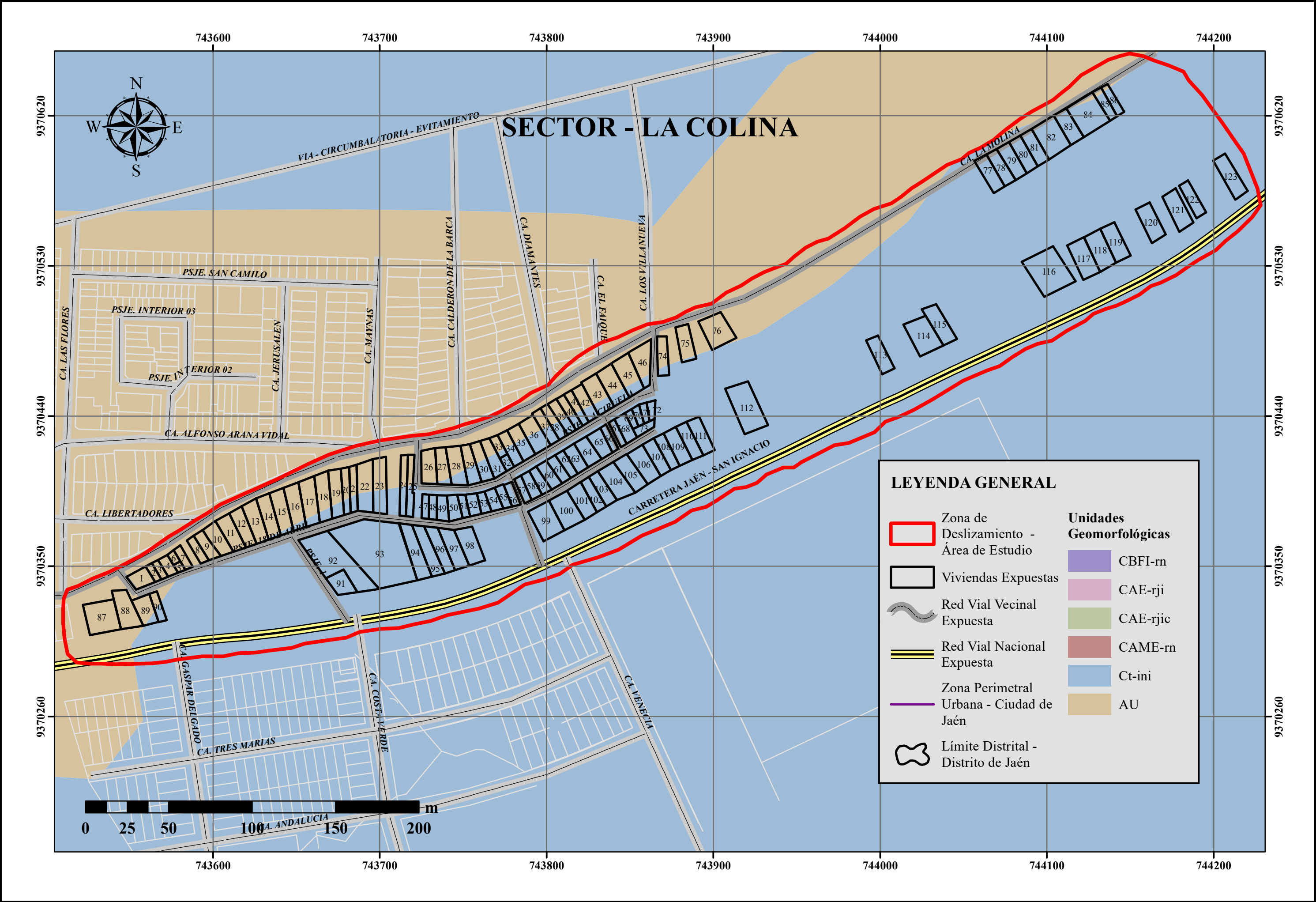
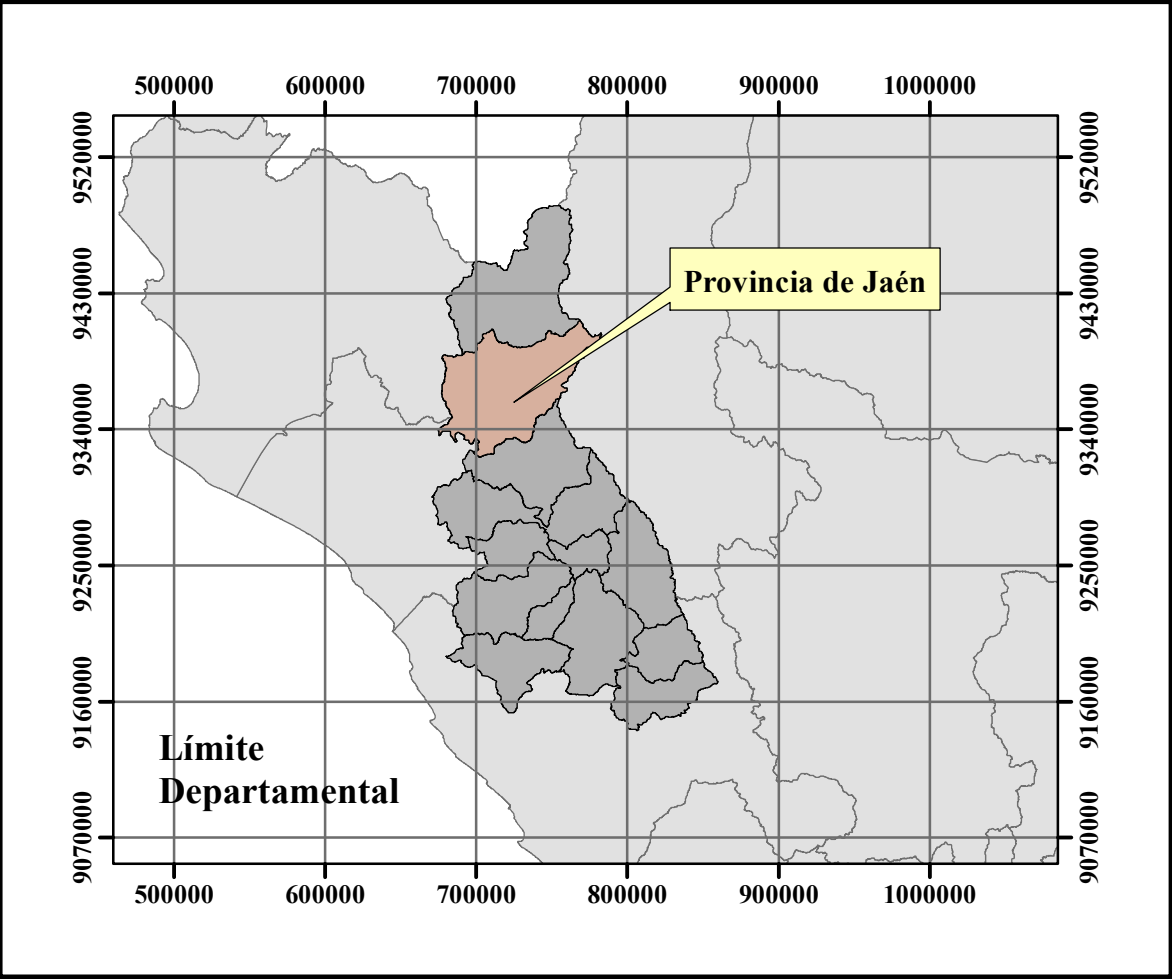
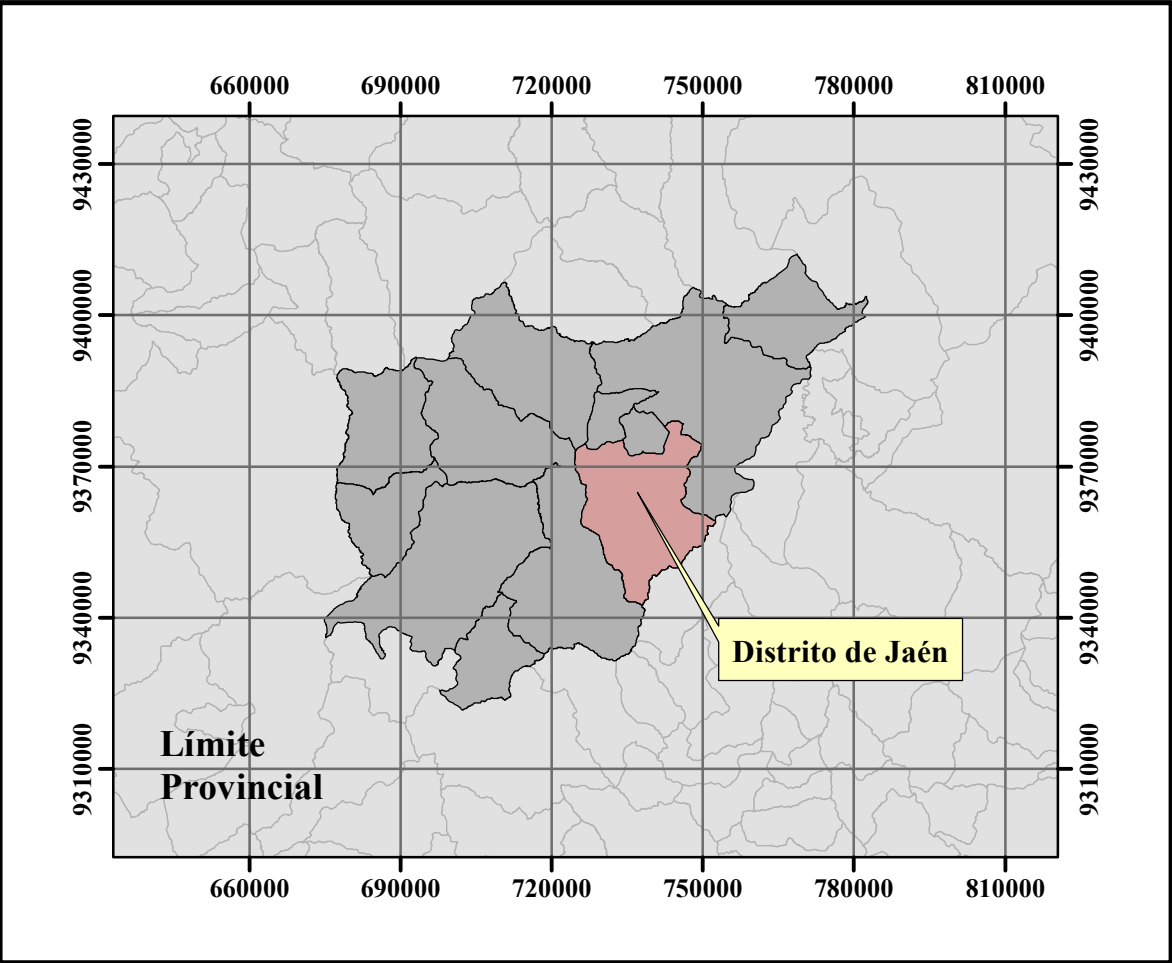
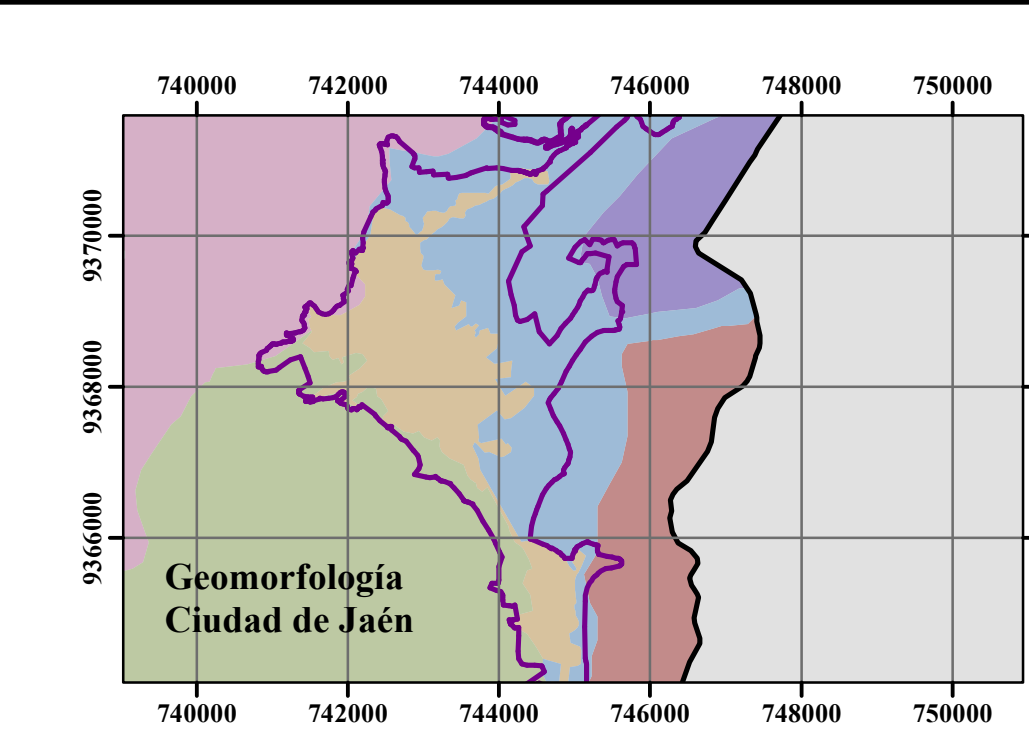
DATUM: WGS - 84 SIST. DE PROYEC: UTM ZONA 17



REGIÓN: CAJAMARCA
PROVINCIA: JAÉN
DISTRITO: JAÉN
FECHA: Febrero-2026
ESCALA: INDICADA
PLANO: **G - 01**

Anexo 9

Mapa espacial del factor condicionante – Geomorfología



CÓDIGO	UNIDAD GEOMORFOLÓGICA	ORIGEN	PAISAJE DOM.	LITOLOGÍA	PROCESO MORFOLÓGICO
CBFI-rn	Colina baja fuertemente inclinada en rocas del neogeno	Denudacional	Colinoso	Intercalaciones de areniscas y lutitas abigarradas	Erosion de laderas
CAE-rji	Colina alta empinada en rocas del jurasico inferior	Denudacional	Colinoso	Rocas volcanicas	Caída de rocas, derrumbes, erosion de laderas
CAE-rjic	Colina alta empinada en rocas del jurasico inferior y del cuaternario	Denudacional	Colinoso	Areniscas cuarzosas bien clasificadas de grano medio a grueso	Movimientos complejos y erosion de laderas
CAME-rn	Colina alta moderadamente empinada en rocas del neogeno	Denudacional	Colinoso	Rocas sedimentarias	Erosion de laderas y caída de rocas
Ct-ini	Complejo de terrazas inundable y no inundable	Agradacional	Planicie	Depositos fluvioaluviales	Erosion e inundacion fluvial, flujo
AU	Casco urbano (Complejo de terrazas inundable y no inundable)	Agradacional	Planicie	Depositos fluvioaluviales	Erosion e inundacion fluvial, flujo



UNIVERSIDAD NACIONAL DE JAÉN

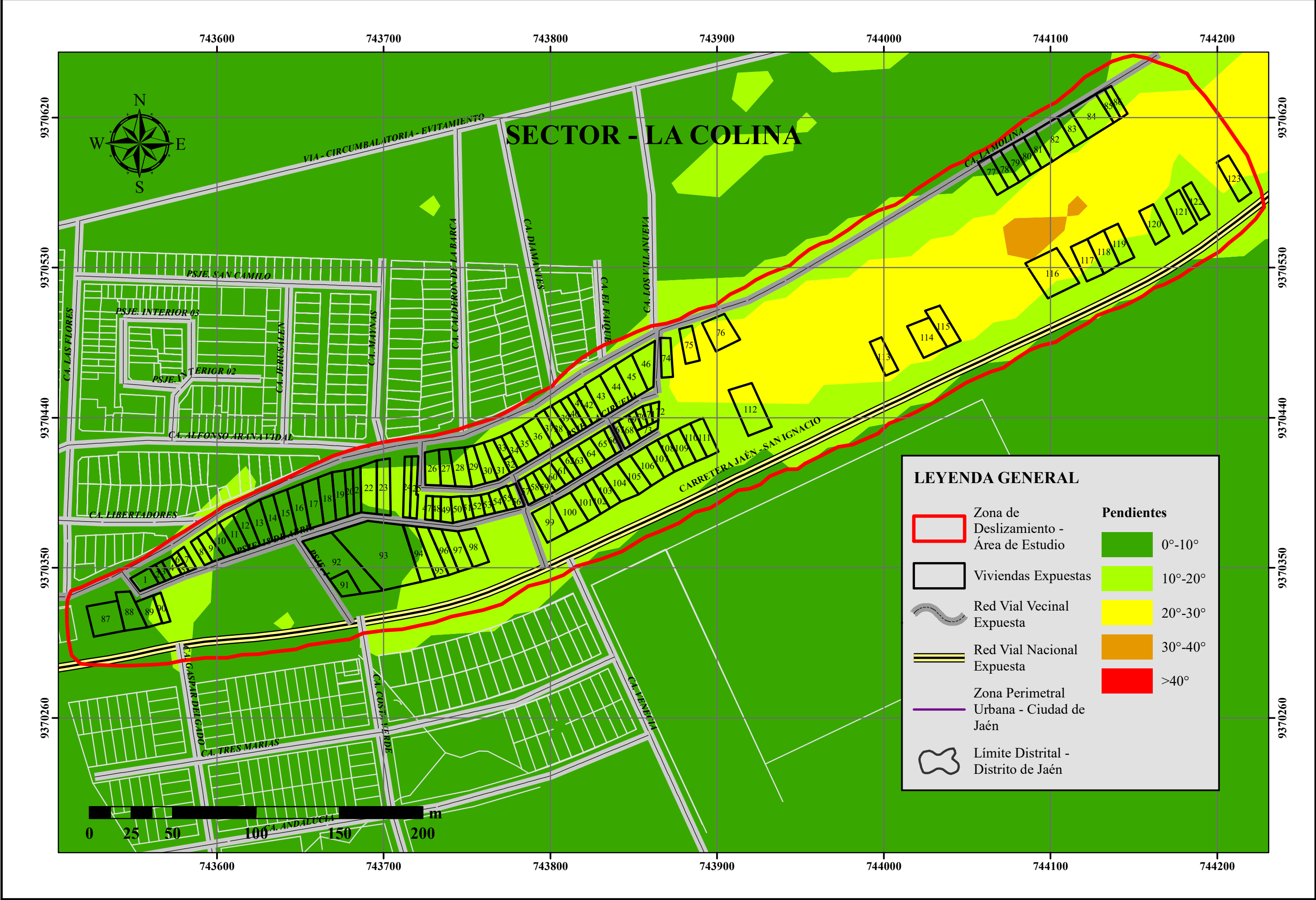
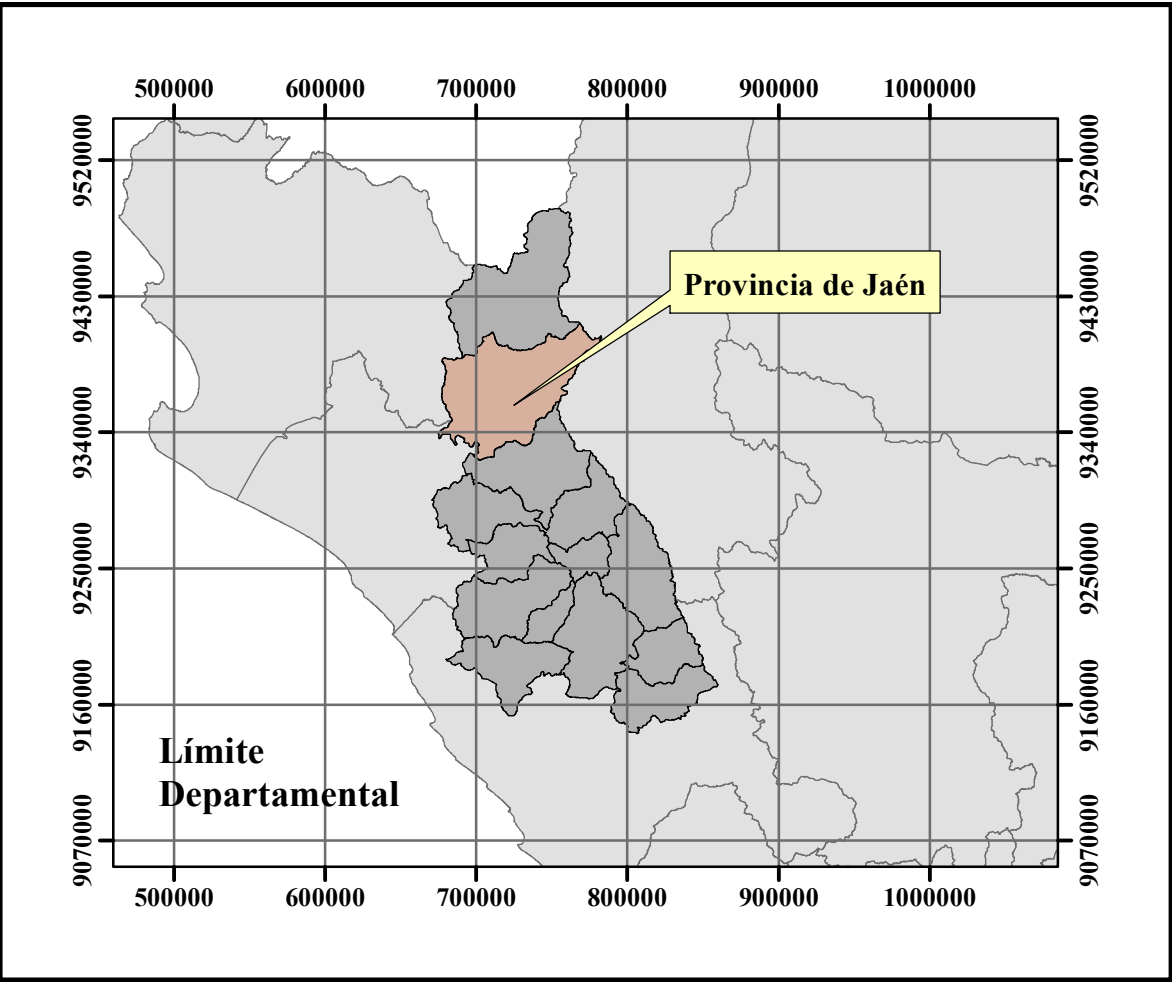
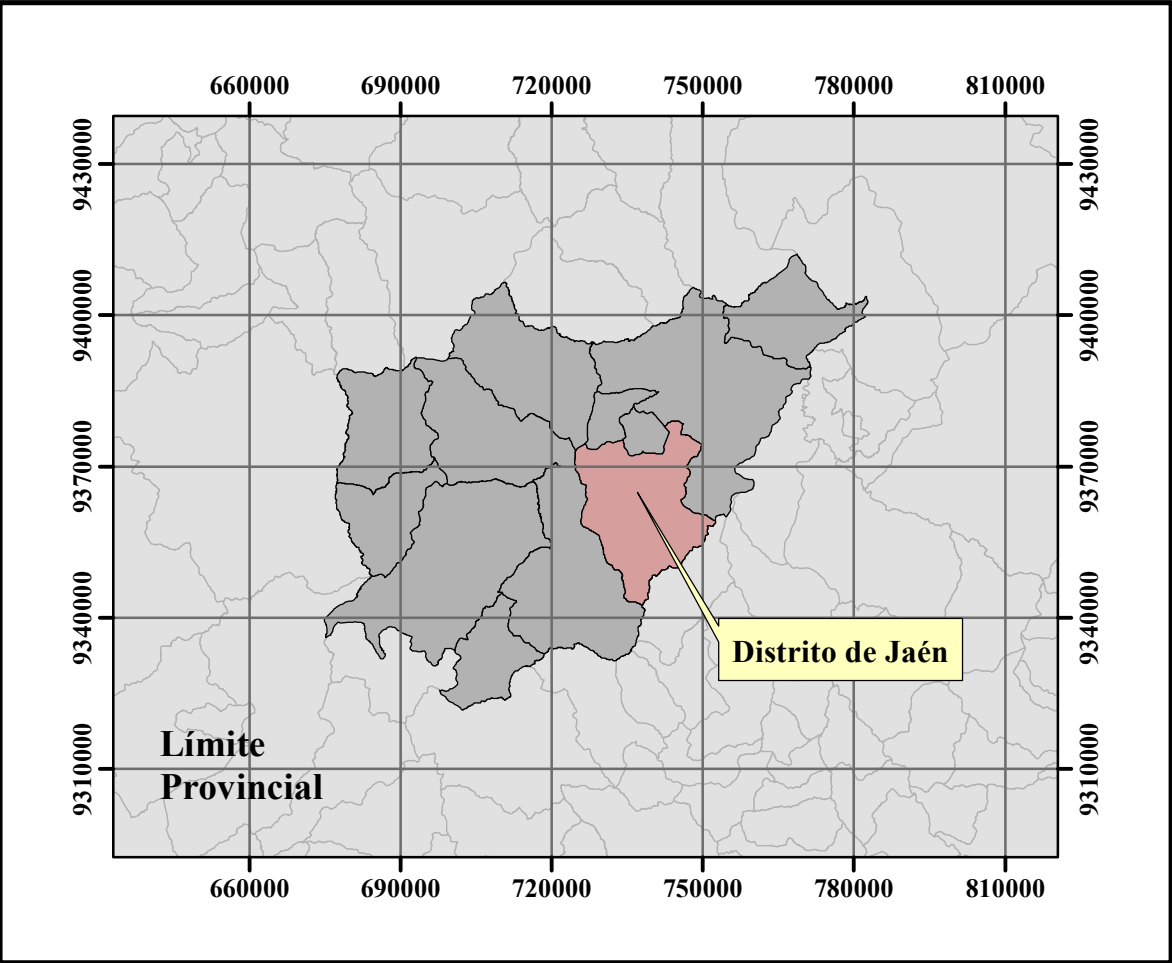
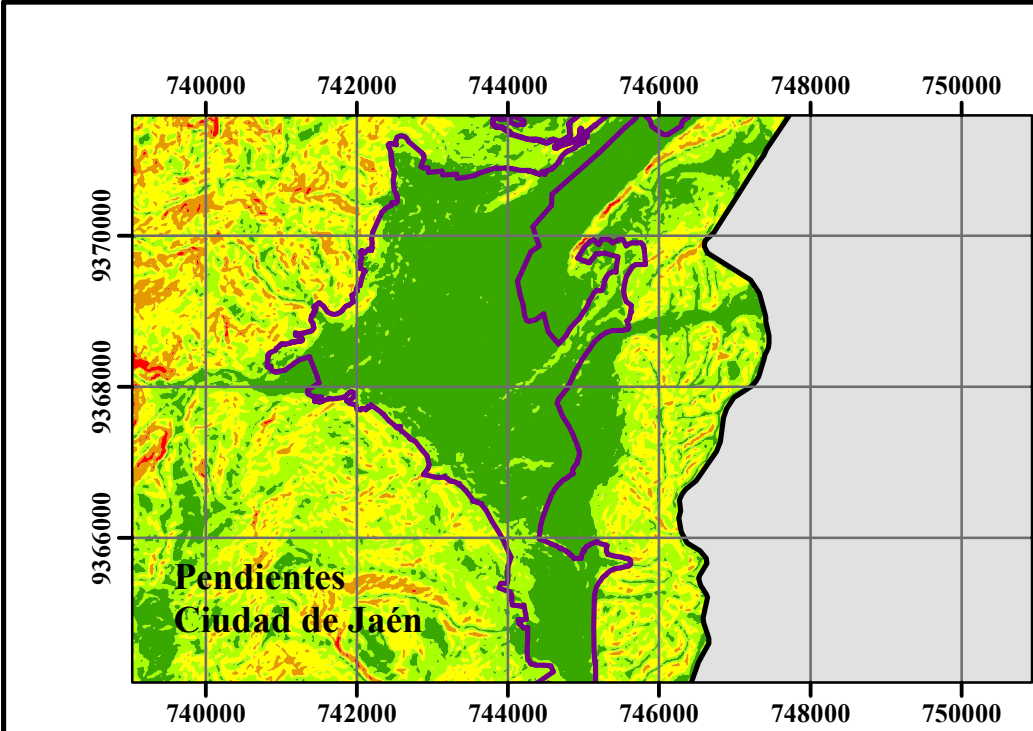
CARRERA PROFESIONAL DE INGENIERÍA CIVIL

Tesis: "ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025"

Lámina:	PLANO DE UNIDADES GEOMORFOLÓGICAS EN EL SECTOR LA COLINA DE LA CIUDAD DE JAÉN (ZONA DE ESTUDIO)	REGIÓN:	CAJAMARCA
Responsables:	MONSALVE TICLIAHUANCA EINSTEIN EDINSON VÍLCHEZ SILVA FRANKLIN FERNANDO	PROVINCIA:	JAÉN
Asesores:	Mg. Ing. Billy Alexis Cayatopa Calderón Dr. Lenin Quiñones Huatangari	DISTRITO:	JAÉN
		FECHA:	Febrero-2026
		ESCALA:	INDICADA
		PLANO:	GM - 01
	DATUM: WGS - 84	SIST. DE PROYEC: UTM	ZONA 17

Anexo 10

Mapa espacial del factor condicionante – Pendiente

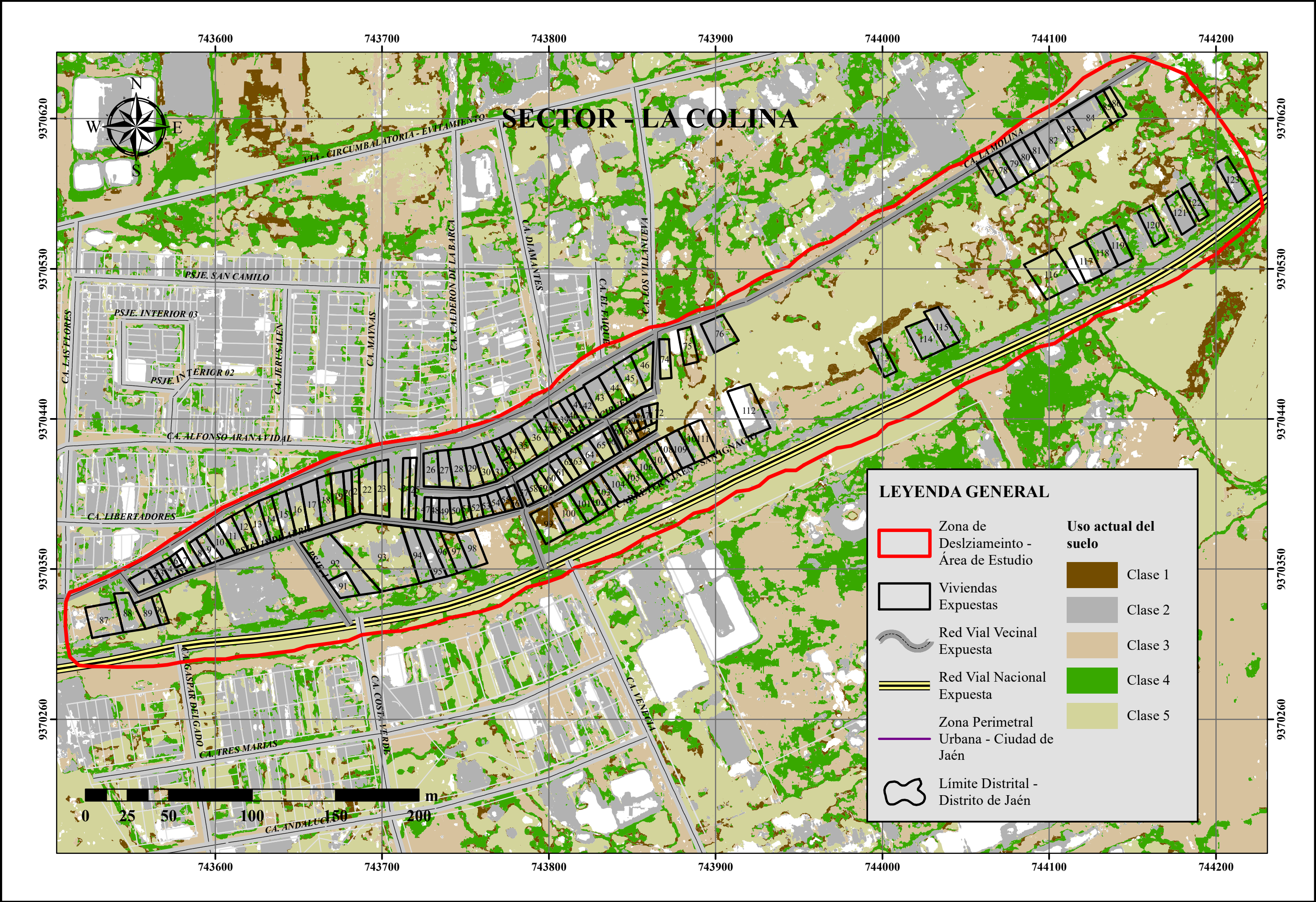
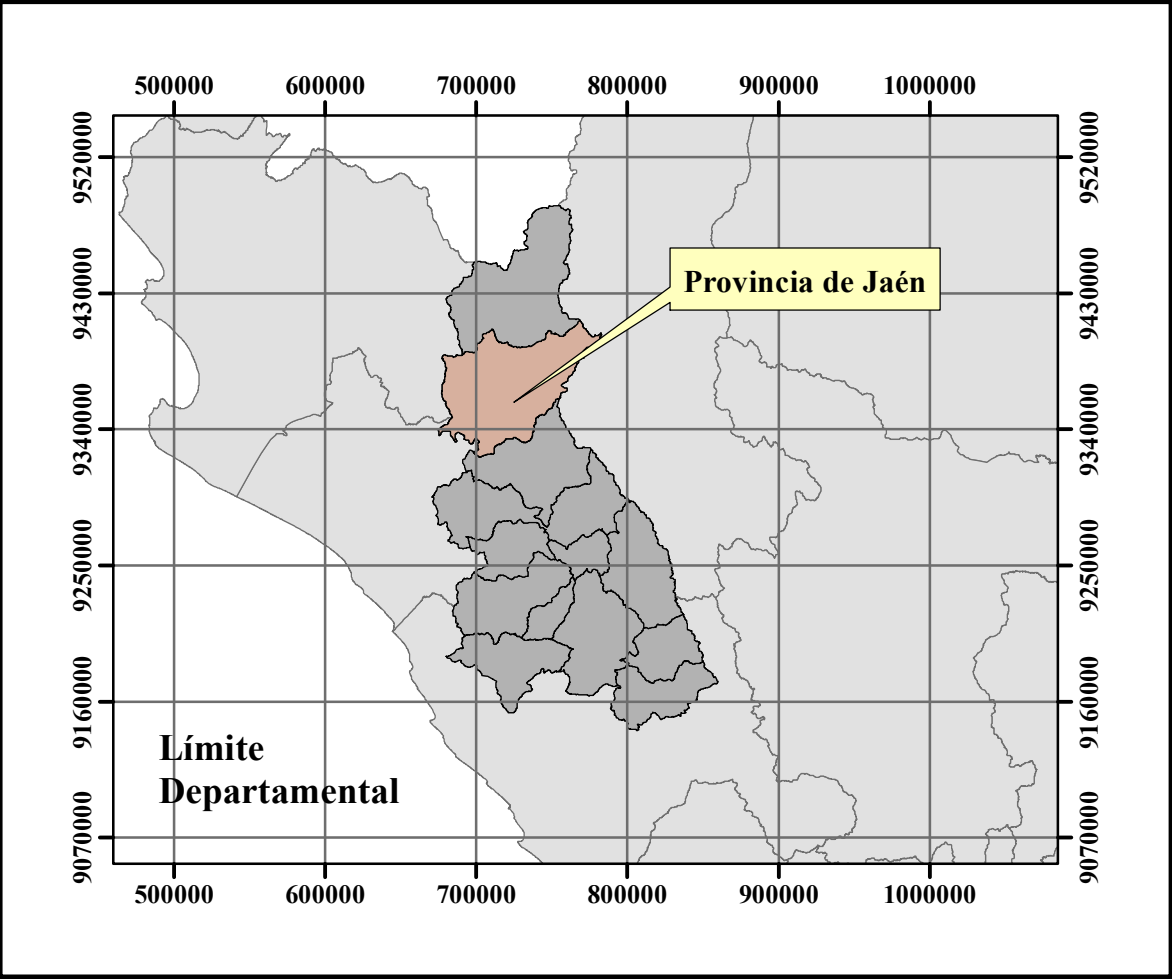
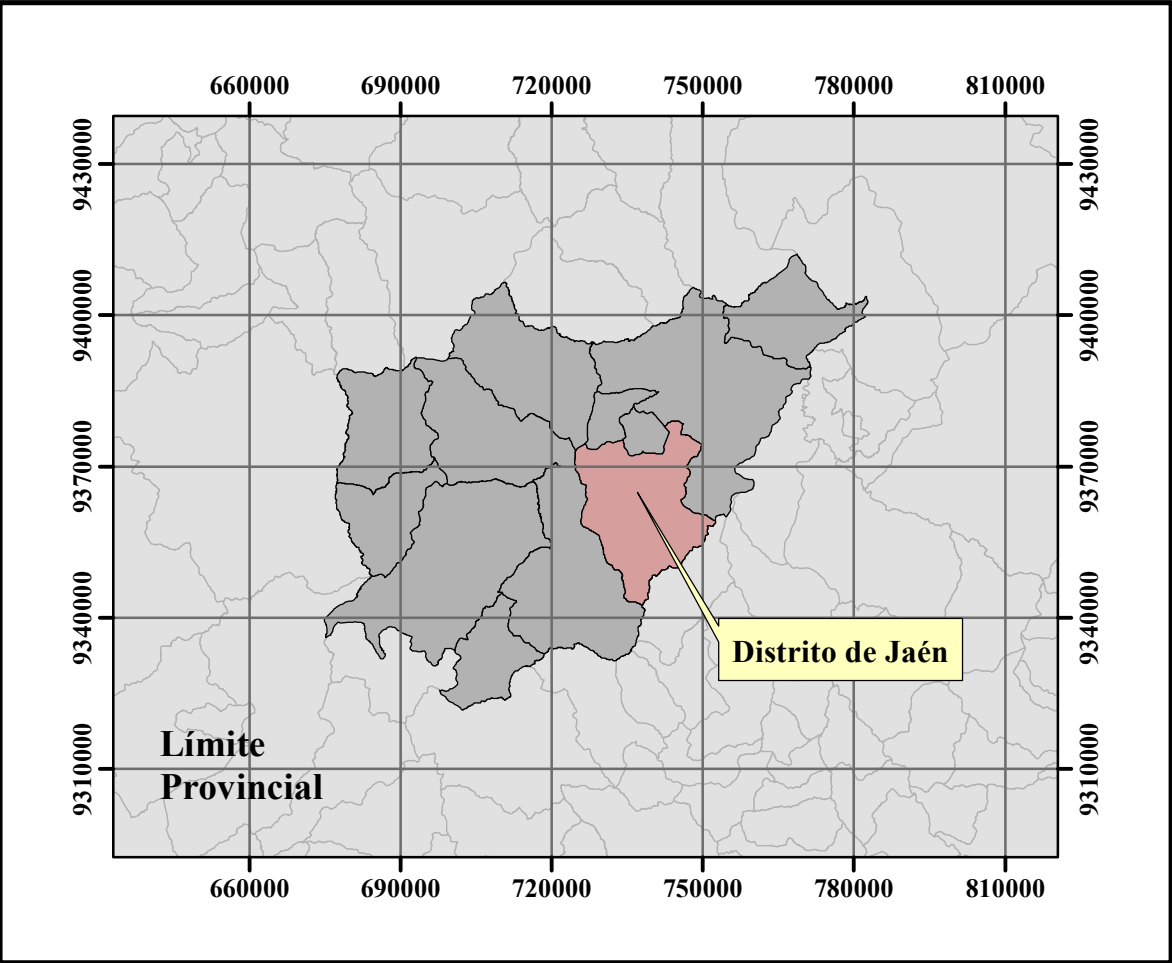
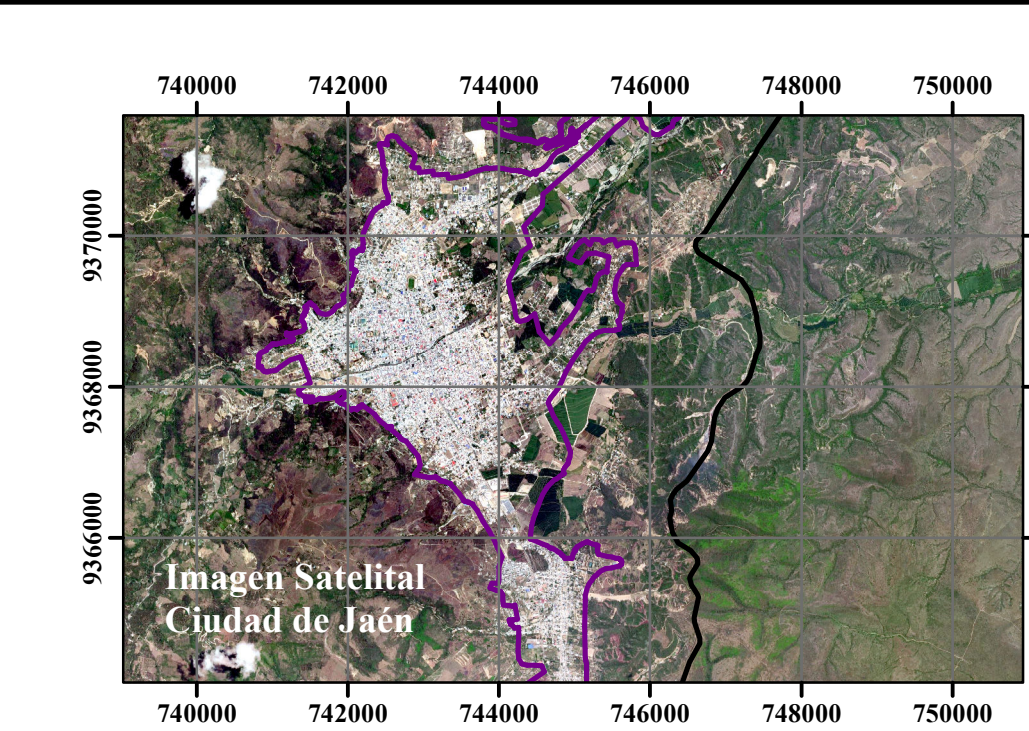


RANGO	DESCRIPCIÓN
0°-10°	Plano o llano (pendiente muy baja)
10°-20	Ligeramente inclinado (pendiente moderada, deslizamientos ocasionales)
20°-30°	Moderadamente empinada (Pendiente fuerte, procesos de deslizamientos, peligro extremo de erosión de suelos)
30°-40°	Empinada (Pendiente muy fuerte, proceso de deslizamientos intensos)
>40°	Muy empinada (Pendiente muy fuerte, proceso de deslizamientos extremos)

 UNIVERSIDAD NACIONAL DE JAÉN CARRERA PROFESIONAL DE INGENIERÍA CIVIL 		
Tesis: "ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025"		
Lámina:	PLANO DE PENDIENTES EN EL SECTOR LA COLINA DE LA CIUDAD DE JAÉN (ZONA DE ESTUDIO)	REGIÓN: CAJAMARCA PROVINCIA: JAÉN
Responsables:	MONSALVE TICLIAHUANCA EINSTEIN EDINSON VÍLCHEZ SILVA FRANKLIN FERNANDO	DISTRITO: JAÉN FECHA: Febrero-2026
Asesores:	Mg. Ing. Billy Alexis Cayatopa Calderón Dr. Lenin Quiñones Huatangari	ESCALA: INDICADA PLANO:
DATUM: WGS - 84 SIST. DE PROYEC: UTM ZONA 17		P - 01

Anexo 11

Mapa espacial del factor condicionante – Uso Actual del Suelo



CLASES	DESCRIPCION
1	Áreas erosionadas, cárcavas, surcos y suelo desnudo degradado
2	Zona Urbanizada en ladera
3	Áreas abiertas sin vegetación
4	Áreas agrícolas heterogéneas
5	Vegetación arbustiva

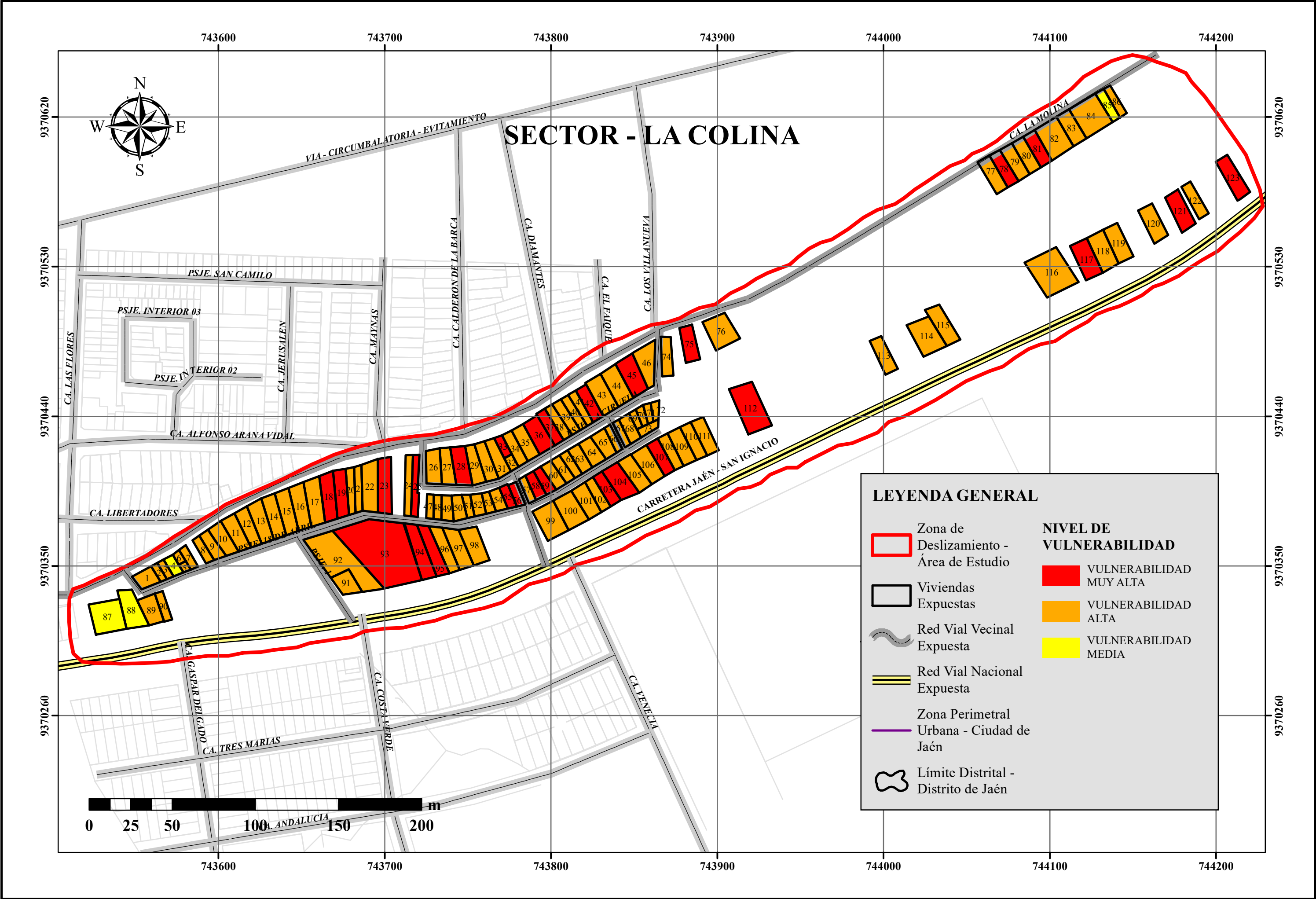
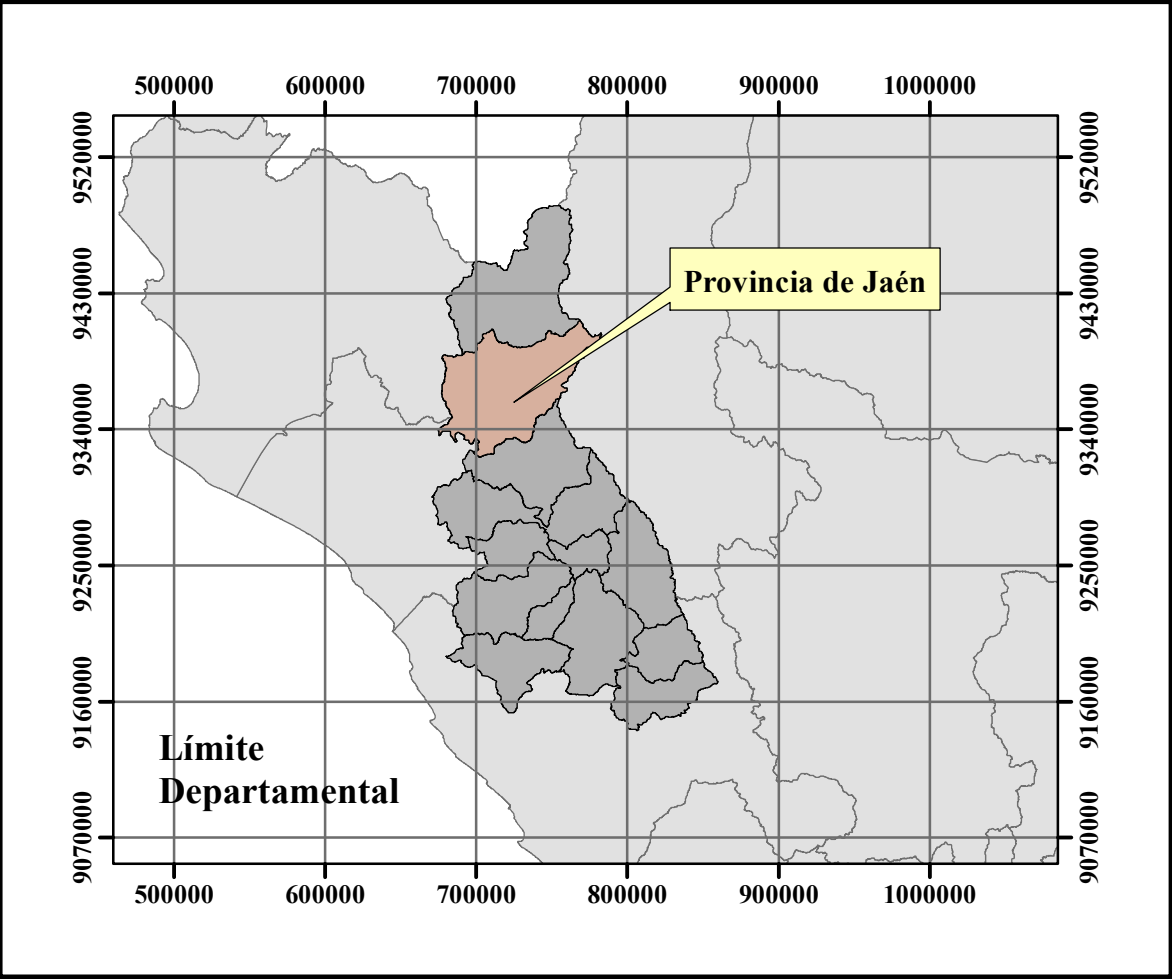
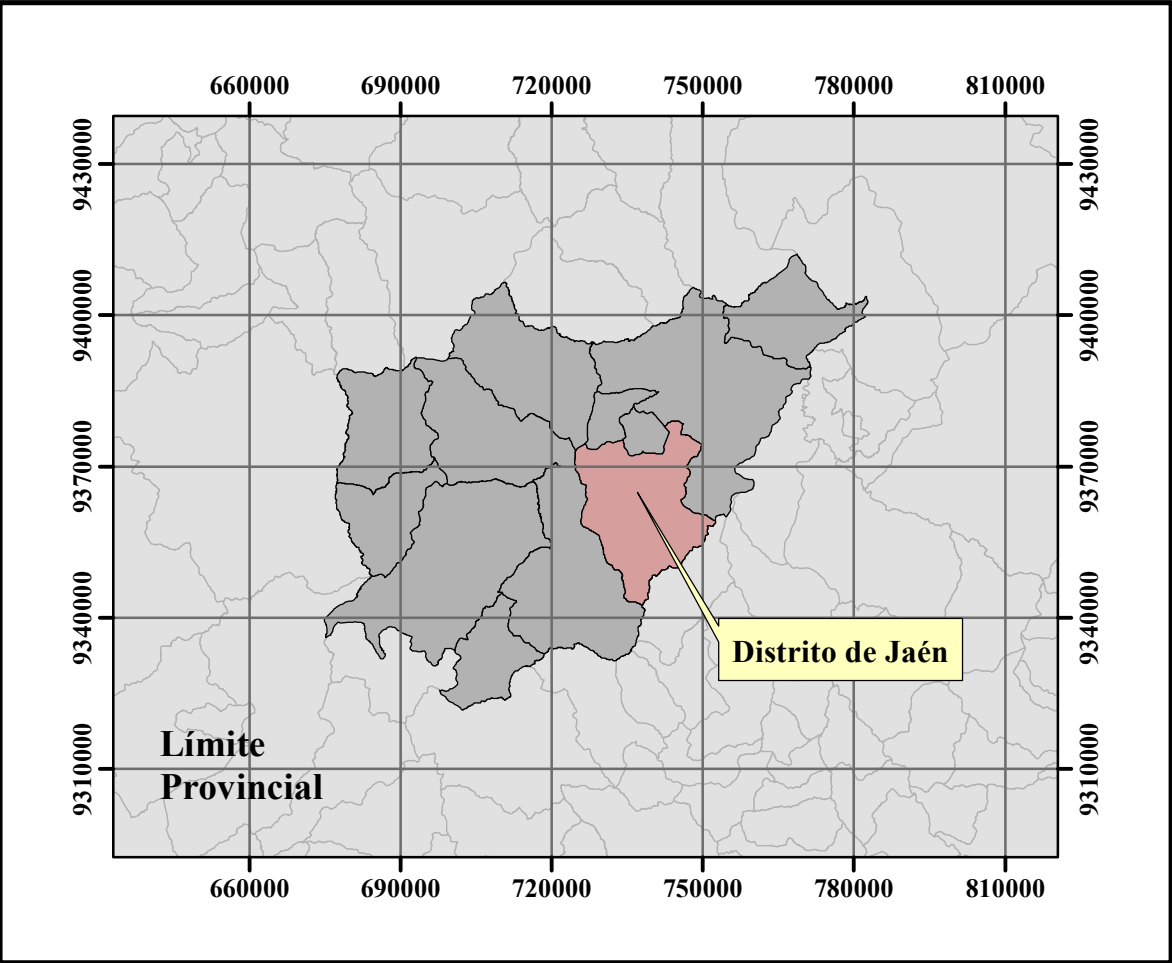
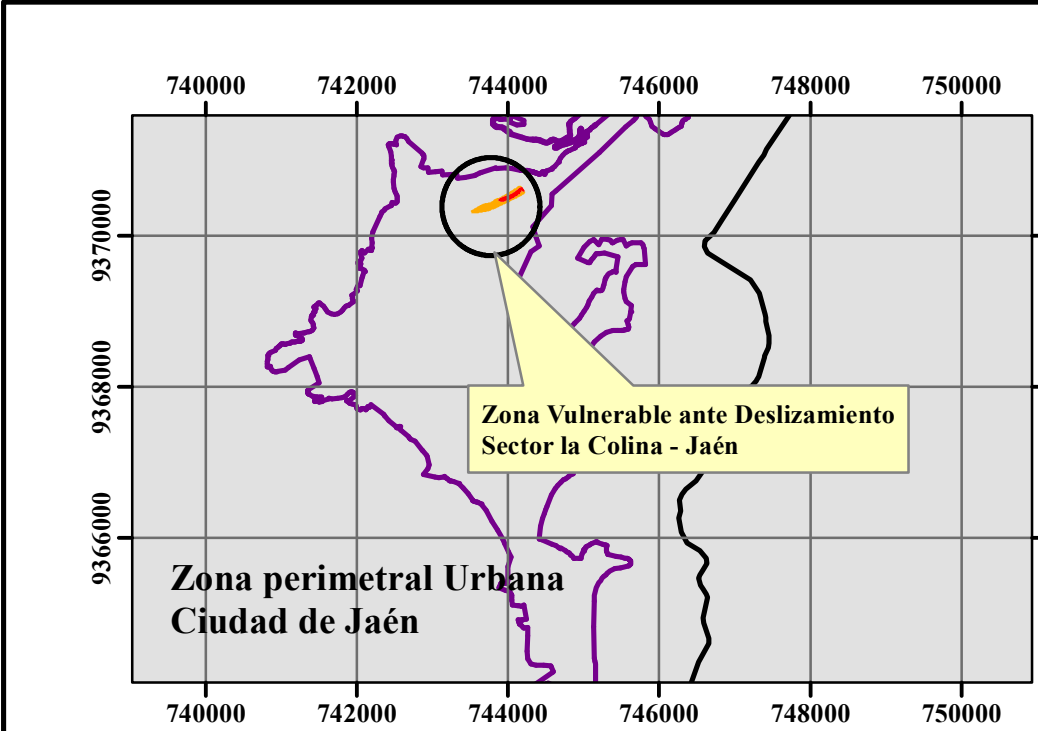
		UNIVERSIDAD NACIONAL DE JAÉN			
		CARRERA PROFESIONAL DE INGENIERÍA CIVIL			
Tesis:	"ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN – CAJAMARCA, 2025"				
Lámina:	PLANO DE USO ACTUAL DEL SUELO EN EL SECTOR LA COLINA DE LA CIUDAD DE JAÉN (ZONA DE ESTUDIO)			REGIÓN:	CAJAMARCA
Responsables:	MONSALVE TICLIAHUANCA EINSTEIN EDINSON VÍLCHEZ SILVA FRANKLIN FERNANDO			PROVINCIA:	JAÉN
Asesores:	Mg. Ing. Billy Alexis Cayatopa Calderón Dr. Lenin Quiñones Huatangari			DISTRITO:	JAÉN
DATUM: WGS - 84 SIST. DE PROYEC: UTM ZONA 17				FECHA:	Febrero-2026
				ESCALA:	INDICADA
				PLANO:	US - 01

Anexo 12

Mapa de zonificación del nivel de peligro del sector La Colina – CENEPRED

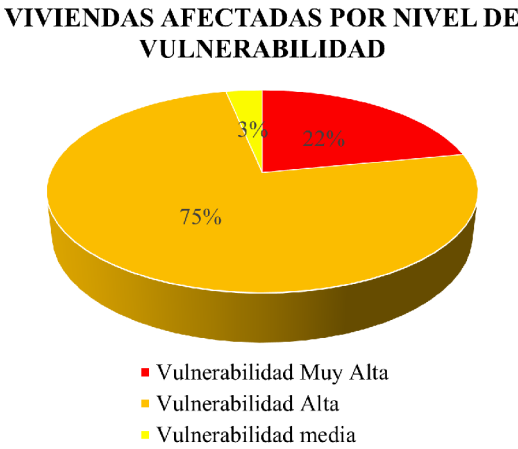
Anexo 13

Mapa de zonificación del nivel de vulnerabilidad del sector La Colina – CENEPRED



RANGO			NIVELES DE VULNERABILIDAD
0.268	≤ P ≤	0.494	VULNERABILIDAD MUY ALTA
0.132	≤ P <	0.268	VULNERABILIDAD ALTA
0.064	≤ P <	0.132	VULNERABILIDAD MEDIA
0.036	≤ P <	0.064	VULNERABILIDAD BAJA

NIVEL DE VULNERABILIDAD	Nº DE VIVIENDAS AFECTADAS
VULNERABILIDAD MUY ALTA	27
VULNERABILIDAD ALTA	92
VULNERABILIDAD MEDIA	4
TOTAL	123





UNIVERSIDAD NACIONAL DE JAÉN

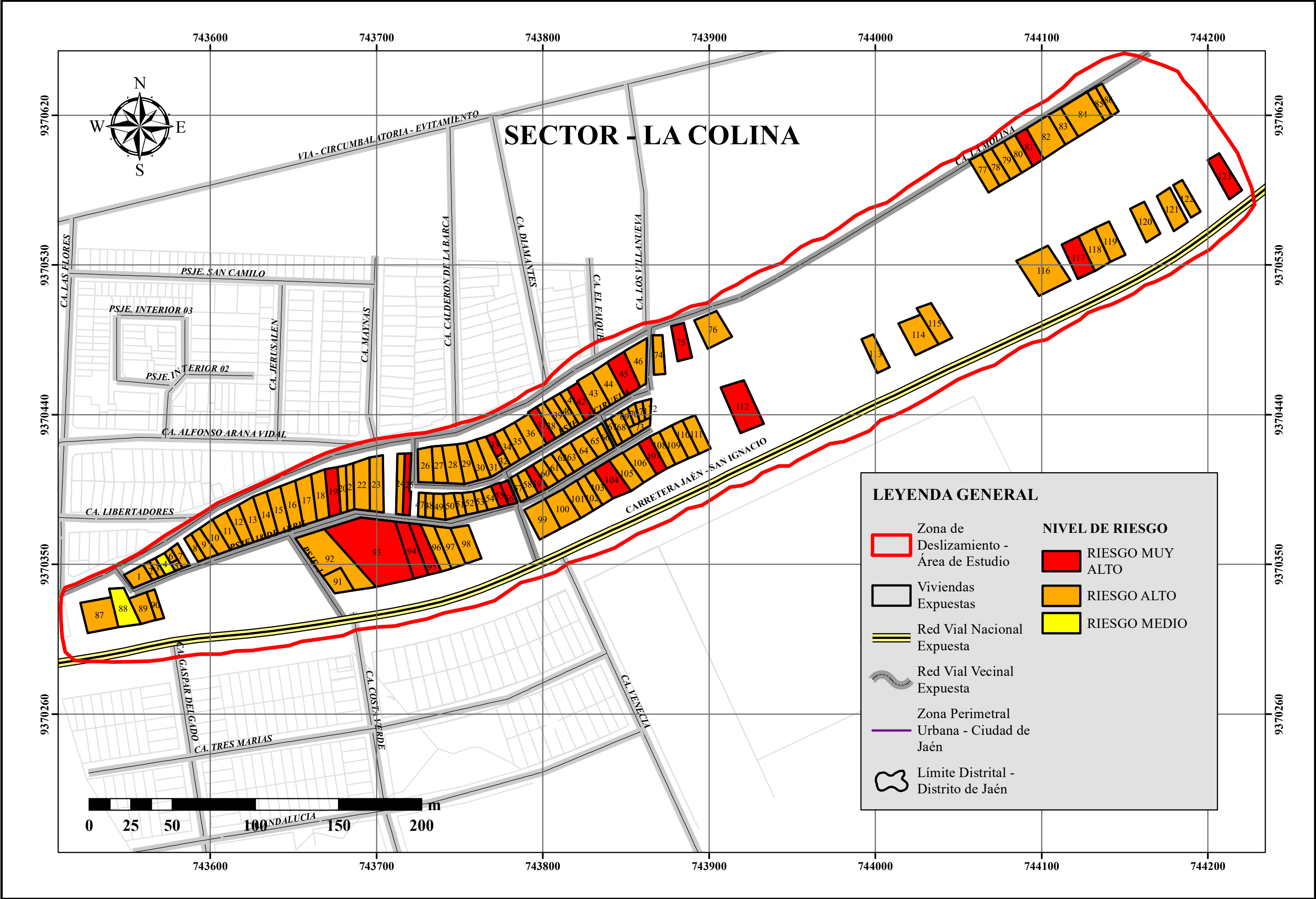
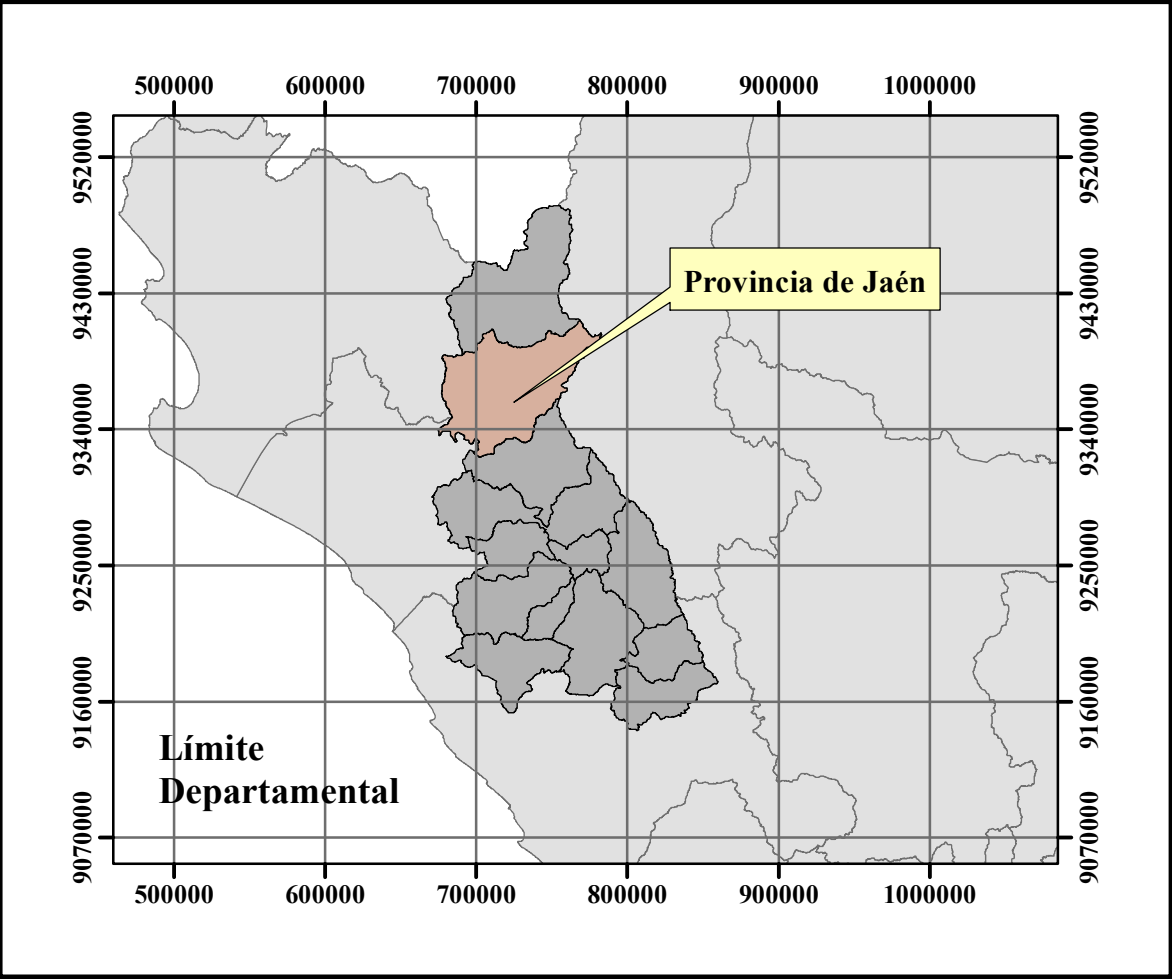
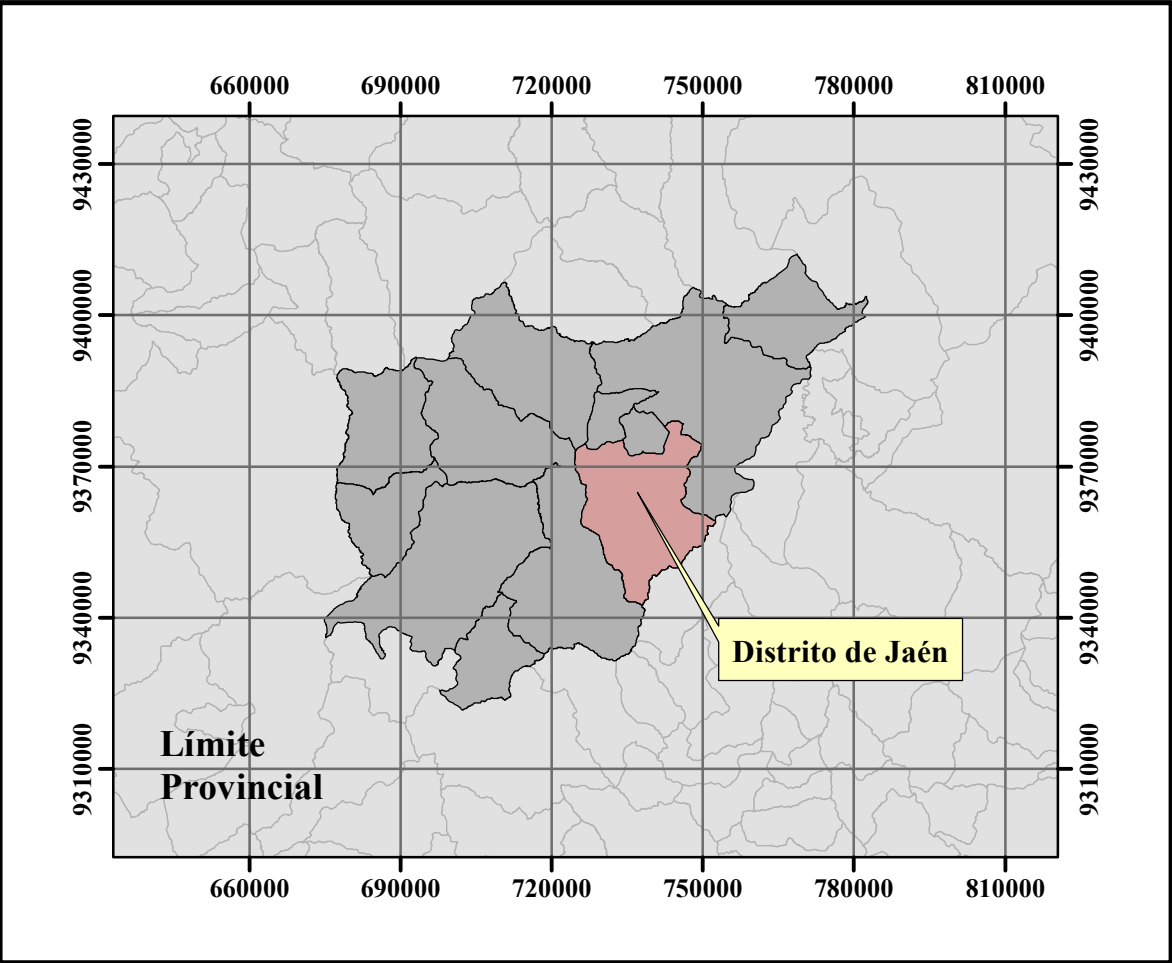
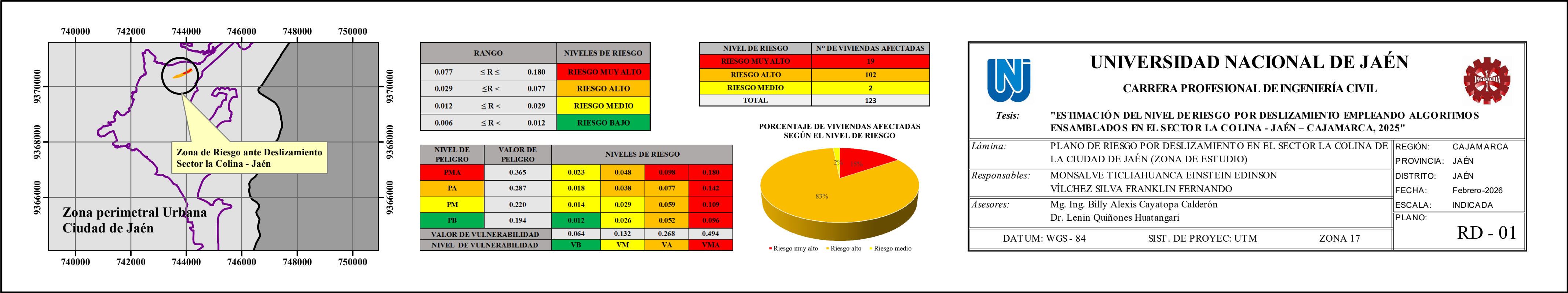
CARRERA PROFESIONAL DE INGENIERÍA CIVIL

Tesis: "ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025"

Lámina:	PLANO DE VULNERABILIDAD POR DESLIZAMIENTO EN EL SECTOR LA COLINA DE LA CIUDAD DE JAÉN (ZONA DE ESTUDIO)	REGIÓN: CAJAMARCA
Responsables:	MONSALVE TICLIAHUANCA EINSTEIN EDINSON VÍLCHEZ SILVA FRANKLIN FERNANDO	PROVINCIA: JAÉN
Asesores:	Mg. Ing. Billy Alexis Cayatopa Calderón Dr. Lenin Quiñones Huatangari	DISTRITO: JAÉN
DATUM: WGS - 84 SIST. DE PROYEC: UTM ZONA: 17		FECHA: Febrero-2026
		ESCALA: INDICADA
		PLANO: VD - 01

Anexo 14

Mapa de zonificación del nivel de riesgo del sector La Colina – CENEPRED

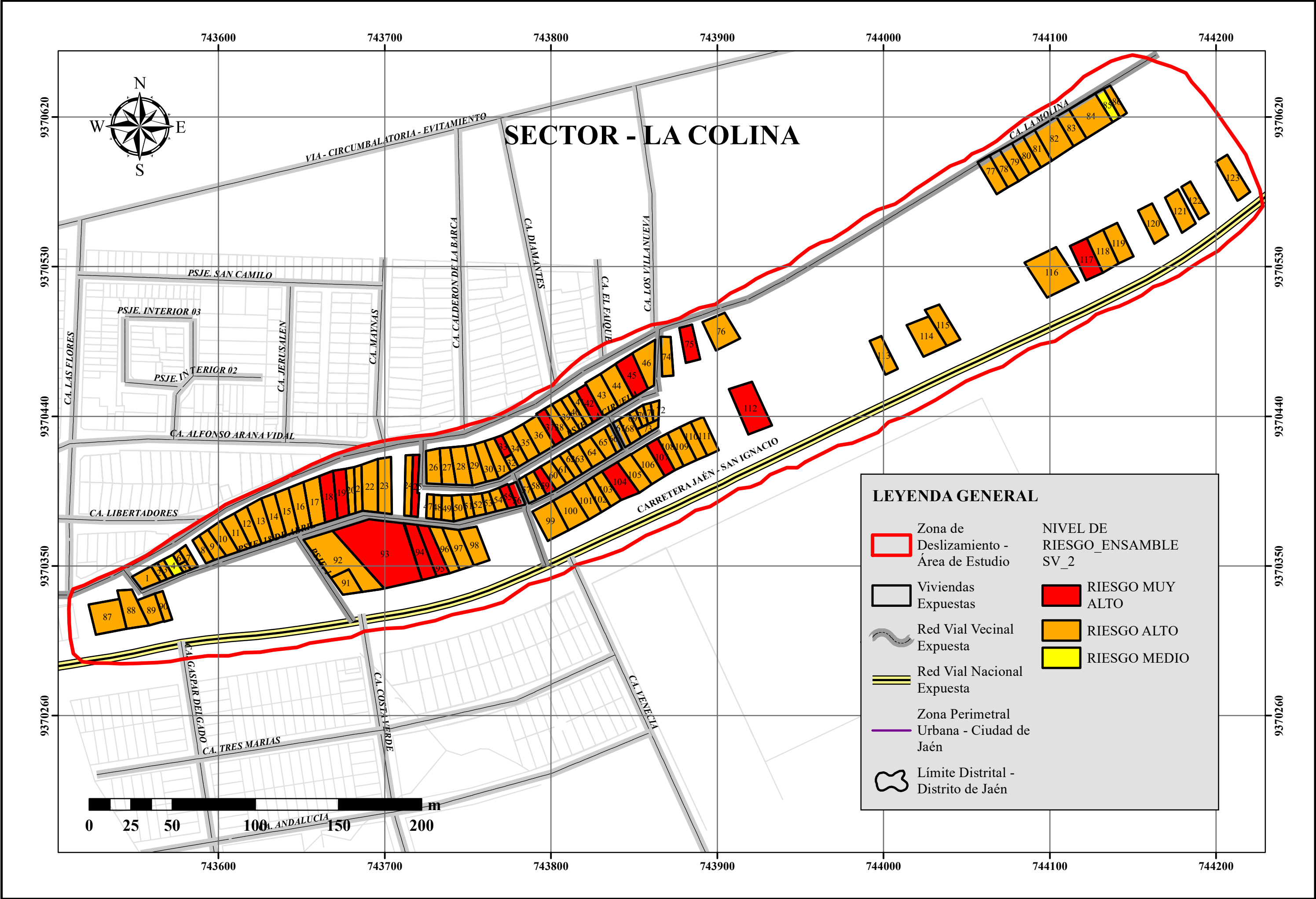
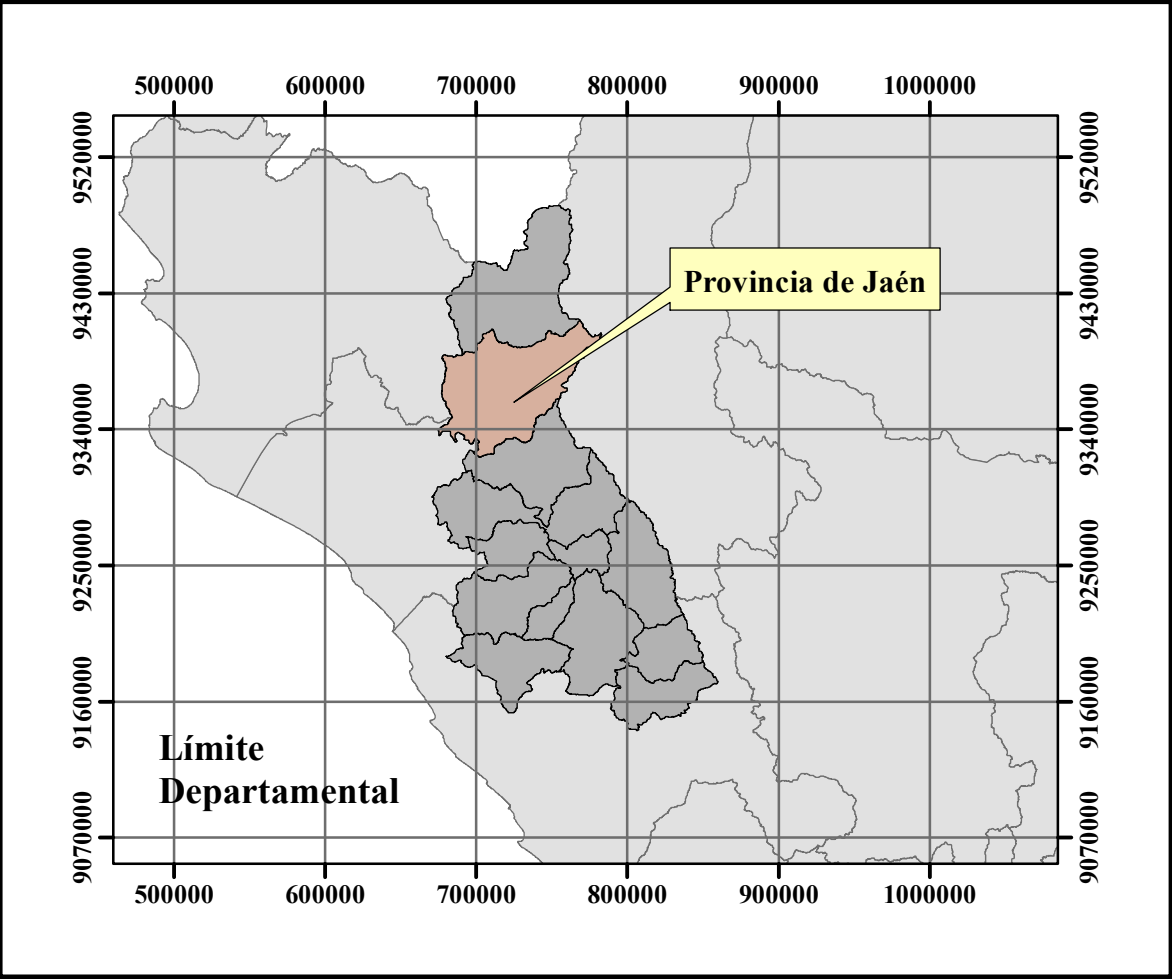
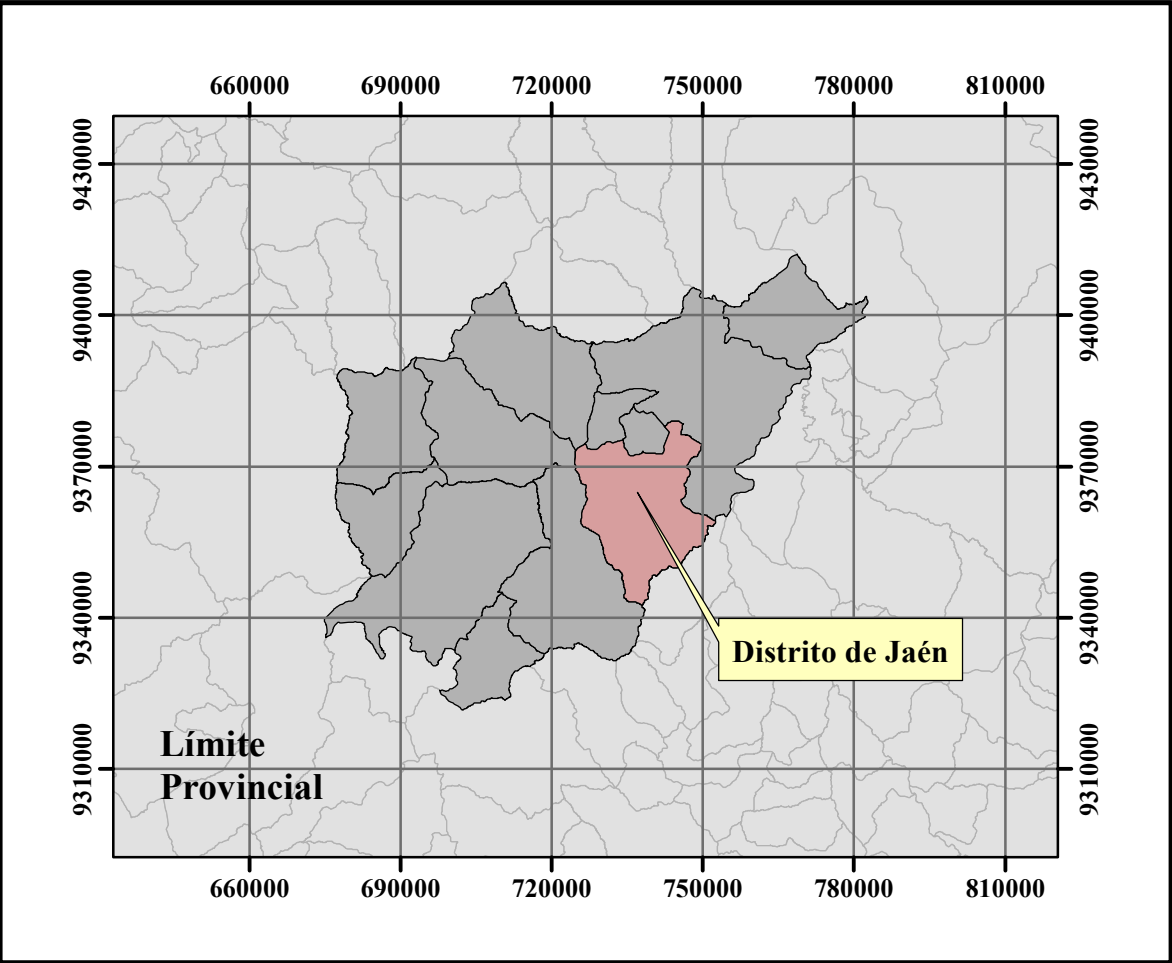
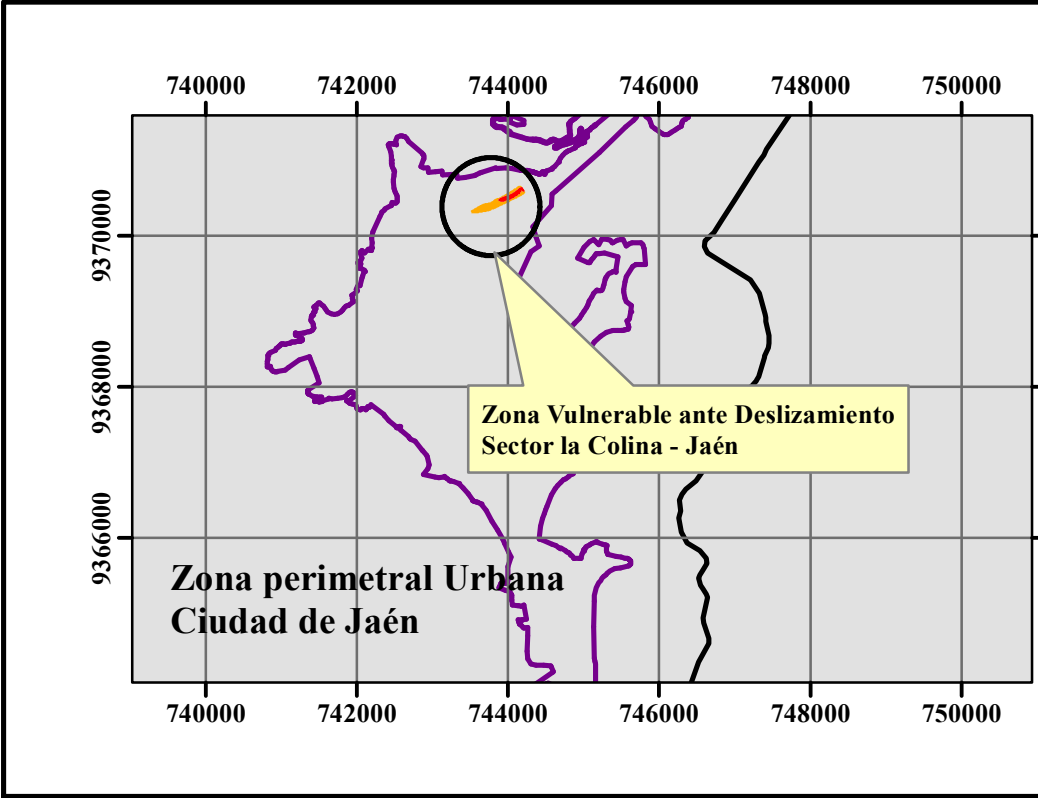


Anexo 15

Mapa de zonificación del nivel de riesgo del sector La

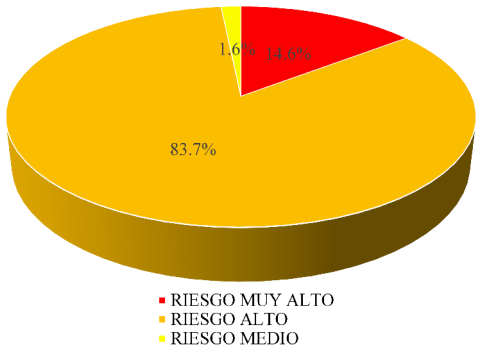
Colina – Algoritmo ensamblado SV_2: Soft Voting:

GB+RF+ET



NIVEL DE RIESGO	Nº DE VIVIENDAS AFECTADAS
RIESGO MUY ALTO	18
RIESGO ALTO	103
RIESGO MEDIO	2
TOTAL	123

PORCENTAJE DE VIVIENDAS AFECTADAS SEGÚN EL NIVEL DE RIESGO ENSAMBLE SV_2



 UNIVERSIDAD NACIONAL DE JAÉN CARRERA PROFESIONAL DE INGENIERÍA CIVIL			
Tesis: "ESTIMACIÓN DEL NIVEL DE RIESGO POR DESLIZAMIENTO EMPLEANDO ALGORITMOS ENSAMBLADOS EN EL SECTOR LA COLINA - JAÉN - CAJAMARCA, 2025"		REGIÓN: CAJAMARCA	
Lámina: PLANO DE RIESGO POR DESLIZAMIENTO EN EL SECTOR LA COLINA DE LA CIUDAD DE JAÉN (ZONA DE ESTUDIO) ENSAMBLE SV_2: GB + RF+ET		PROVINCIA: JAÉN	
Responsables: MONSALVE TICLIAHUANCA EINEIN EDINSON VÍLCHEZ SILVA FRANKLIN FERNANDO		DISTRITO: JAÉN	
Asesores: Mg. Ing. Billy Alexis Cayatopa Calderón Dr. Lenin Quiñones Huatangari		FECHA: Febrero-2026	
DATUM: WGS - 84		SIST. DE PROYEC: UTM	ZONA 17
		ESCALA: INDICADA	
		PLANO: RE - 01	