

UNIVERSIDAD NACIONAL DEL SANTA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E
INFORMÁTICA



**Modelo predictivo basado en inteligencia artificial y
oversampling para accidentes laborales en Corporación de
Alimentos Marítimo S.A.C**

**TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE
INGENIERO DE SISTEMAS E INFORMÁTICA**

AUTORES:

Bach. Leyva Cordova, German David

(orcid.org/0009-0008-0233-9207)

Bach. Kano Leiva, Brayan Marcelo

(orcid.org/0009-0009-1368-5727)

ASESOR:

Dr. Daza Vergaray, Alfredo

Código ORCID: [0000-0002-2259-1070](https://orcid.org/0000-0002-2259-1070)

Nuevo Chimbote – Perú

2026

UNIVERSIDAD NACIONAL DEL SANTA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E
INFORMÁTICA



CONFORMIDAD DEL ASESOR

La presente Tesis ha sido revisada y desarrollada en cumplimiento del objetivo propuesto y reúne las condiciones formales y metodológicas, estando en cuadrado dentro de las áreas y líneas de investigación conforme al reglamento general para obtener el Título Profesional en la Universidad Nacional del Santa de acuerdo a la denominación siguiente:

TESIS PARA OBTENER EL TÍTULO PROFESIONAL DE INGENIERO DE
SISTEMAS E INFORMÁTICA

“Modelo predictivo basado en inteligencia artificial y oversampling para
accidentes laborales en Corporación de Alimentos Marítimo S.A.C”

AUTORES: Bach. Leyva Cordova, German David (orcid.org/0009-0008-0233-9207)
Bach. Kano Leiva, Brayan Marcelo (orcid.org/0009-0009-1368-5727)

Revisado y Aprobado por Asesor:



Dr. Daza Vergaray, Alfredo

DNI: 40466240

Cód. ORCID 0000-0002-2259-1070

UNIVERSIDAD NACIONAL DEL SANTA
FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E
INFORMÁTICA



UNS
UNIVERSIDAD
NACIONAL DEL SANTA

CONFORMIDAD DEL JURADO EVALUADOR

El presente Jurado Evaluador da la conformidad del presente informe, desarrollado en cumplimiento del objetivo propuesto y presentado con forme al Reglamento General para obtener el Título Profesional en la Universidad Nacional del Santa, titulado:

“Modelo predictivo basado en inteligencia artificial y oversampling para accidentes laborales en Corporación de Alimentos Marítimo S.A.C”

AUTORES: Bach. Leyva Cordova, German David (orcid.org/0009-0008-0233-9207)
Bach. Kano Leiva, Brayan Marcelo (orcid.org/0009-0009-1368-5727)

Revisado y Aprobado por el Jurado Evaluador:

Dr. Ramírez Milla, Luis Enrique

Presidente

DNI N° 32956519

Cód. ORCID: 0000-0001-9064-5395

Mg. Escalante Espinoza, Walter Jesús

Secretario

DNI N° 32763644

Cód. ORCID: 0009-0004-7699-0044

Dr. Daza Vergaray, Alfredo

Integrante

DNI N° 40466240

Cód. ORCID: 0000-0002-2259-1070



UNS
UNIVERSIDAD
NACIONAL DEL SANTA

FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

ACTA DE EVALUACIÓN PARA SUSTENTACIÓN DE TESIS

En el Campus Universitario de la Universidad Nacional del Santa, siendo las 10:00 a.m. del día jueves 09 de julio de 2026, en el Pabellón de CECOMP, Aula S6, y en cumplimiento de lo dispuesto mediante Resolución Decanal N° 343-2026-UNS-FI, que declara expedito el presente acto académico con fecha 07 de julio de 2026, se procedió a la instalación del Jurado Evaluador, designado por Resolución N° 190-2026-UNS-CFI de fecha 03 de junio de 2026. Dicho Jurado se encuentra integrado por el **Dr. Luis Enrique Ramírez Milla (Presidente)**, el **Mg. Walter Jesús Escalante Espinoza (Secretario)** y el **Dr. Alfredo Daza Vergaray (Integrante)**, con la finalidad de dar inicio a la sustentación del Informe Final de Tesis titulado: **"MODELO PREDICTIVO BASADO EN INTELIGENCIA ARTIFICIAL Y OVERSAMPLING PARA ACCIDENTES LABORALES EN CORPORACIÓN DE ALIMENTOS MARÍTIMO S.A.C."**, presentado por los Bachilleres **LEYVA CÓRDOVA GERMAN DAVID**, con código de matrícula N° 0201914008, y **KANO LEIVA BRAYAN MARCELO**, con código de matrícula N° 0201914003, quienes cuentan con la asesoría del **Dr. Alfredo Daza Vergaray**, designado mediante Resolución N° 371-2025-UNS-FI, de fecha 20 de junio de 2026.

Concluida la exposición, los sustentantes absolvieron las preguntas, observaciones y comentarios formulados por los miembros del Jurado Evaluador, así como por el público asistente.

Luego de una deliberación exhaustiva sobre el contenido del trabajo, la calidad de la exposición y las respuestas brindadas, así como las sugerencias pertinentes, y en concordancia con lo estipulado en los artículos 71° y 111° del Reglamento General de Grados y Títulos vigente de la Universidad Nacional del Santa (aprobado mediante Resolución N° 337-2024-CU-R-UNS del 12 de abril de 2024), el Jurado acuerda otorgar la siguiente nota final de Evaluación:

BACHILLER	CALIFICACIÓN	CONDICIÓN
LEYVA CORDOVA GERMAN DAVID	18	BUENO

Siendo las 10:40 a.m. se dio por terminado el acto de sustentación y, en señal de conformidad, los miembros del Jurado firman la presente Acta.


Dr. Luis Enrique Ramírez Milla
Presidente


Mg. Walter Jesús Escalante Espinoza
Secretario


Dr. Alfredo Daza Vergaray
Integrante



UNS
UNIVERSIDAD
NACIONAL DEL SANTA

FACULTAD DE INGENIERÍA
ESCUELA PROFESIONAL DE INGENIERÍA DE SISTEMAS E INFORMÁTICA

ACTA DE EVALUACIÓN PARA SUSTENTACIÓN DE TESIS

En el Campus Universitario de la Universidad Nacional del Santa, siendo las 10:00 a.m. del día jueves 09 de julio de 2026, en el Pabellón de CECOMP, Aula S6, y en cumplimiento de lo dispuesto mediante Resolución Decanal N° 343-2026-UNS-FI, que declara expedito el presente acto académico con fecha 07 de julio de 2026, se procedió a la instalación del Jurado Evaluador, designado por Resolución N° 190-2026-UNS-CFI de fecha 03 de junio de 2026. Dicho Jurado se encuentra integrado por el **Dr. Luis Enrique Ramírez Milla (Presidente)**, el **Mg. Walter Jesús Escalante Espinoza (Secretario)** y el **Dr. Alfredo Daza Vergaray (Integrante)**, con la finalidad de dar inicio a la sustentación del Informe Final de Tesis titulado: **"MODELO PREDICTIVO BASADO EN INTELIGENCIA ARTIFICIAL Y OVERSAMPLING PARA ACCIDENTES LABORALES EN CORPORACIÓN DE ALIMENTOS MARÍTIMO S.A.C."**, presentado por los Bachilleres **LEYVA CÓRDOVA GERMAN DAVID**, con código de matrícula N° 0201914008, y **KANO LEIVA BRAYAN MARCELO**, con código de matrícula N° 0201914003, quienes cuentan con la asesoría del **Dr. Alfredo Daza Vergaray**, designado mediante Resolución N° 371-2025-UNS-FI, de fecha 20 de junio de 2026.

Concluida la exposición, los sustentantes absolvieron las preguntas, observaciones y comentarios formulados por los miembros del Jurado Evaluador, así como por el público asistente.

Luego de una deliberación exhaustiva sobre el contenido del trabajo, la calidad de la exposición y las respuestas brindadas, así como las sugerencias pertinentes, y en concordancia con lo estipulado en los artículos 71° y 111° del Reglamento General de Grados y Títulos vigente de la Universidad Nacional del Santa (aprobado mediante Resolución N° 337-2024-CU-R-UNS del 12 de abril de 2024), el Jurado acuerda otorgar la siguiente nota final de Evaluación:

BACHILLER	CALIFICACIÓN	CONDICIÓN
KANO LEIVA BRAYAN MARCELO	18	BUENO

Siendo las 10:40 a.m. se dio por terminado el acto de sustentación y, en señal de conformidad, los miembros del Jurado firman la presente Acta.


Dr. Luis Enrique Ramírez Milla
Presidente


Dr. Alfredo Daza Vergaray
Integrante


Mg. Walter Jesús Escalante Espinoza
Secretario



Recibo digital

Este recibo confirma que su trabajo ha sido recibido por Turnitin. A continuación podrá ver la información del recibo con respecto a su entrega.

La primera página de tus entregas se muestra abajo.

Autor de la entrega: ALFREDO DAZA VERGARAY_PreviewUser
Título del ejercicio: Turnitin
Título de la entrega: EMPASTADO GERMAN LEYVA- BRAYAN v2 (2).pdf
Nombre del archivo: EMPASTADO_GERMAN_LEYVA-_BRAYAN_v2_2_.pdf
Tamaño del archivo: 977.05K
Total páginas: 76
Total de palabras: 18,821
Total de caracteres: 105,887
Fecha de entrega: 11-ago-2026 08:41p. m. (UTC-0500)
Identificador de la entrega: 3011351547



EMPASTADO GERMAN LEYVA- BRAYAN v2 (2).pdf

INFORME DE ORIGINALIDAD

20%

INDICE DE SIMILITUD

17%

FUENTES DE INTERNET

5%

PUBLICACIONES

7%

TRABAJOS DEL ESTUDIANTE

FUENTES PRIMARIAS

1	Submitted to Universidad Cesar Vallejo	2%
	Trabajo del estudiante	
2	repositorio.unfv.edu.pe	2%
	Fuente de Internet	
3	repositorio.ucv.edu.pe	2%
	Fuente de Internet	
4	hdl.handle.net	1%
	Fuente de Internet	
5	repositorio.uns.edu.pe	1%
	Fuente de Internet	
6	Submitted to Universidad Continental	<1%
	Trabajo del estudiante	
7	repository.uniminuto.edu	<1%
	Fuente de Internet	
8	www.coursehero.com	<1%
	Fuente de Internet	
9	www.unite.ai	<1%
	Fuente de Internet	
10	Submitted to Shaheed Benazir Bhutto Women University, Peshawar	<1%
	Trabajo del estudiante	
11	qdoc.tips	<1%
	Fuente de Internet	
12	Submitted to Universidad Nacional Jorge Basadre Grohmann	<1%
	Trabajo del estudiante	
13	repositorio.uch.edu.pe	<1%
	Fuente de Internet	
14	prezi.com	<1%
	Fuente de Internet	
15	Submitted to Universidad Internacional de la Rioja	<1%
	Trabajo del estudiante	

DEDICATORIA

A nuestros padres, por su trabajo y sacrificio en todos estos años, gracias a su apoyo incondicional hemos logrado llegar hasta aquí y continuar esforzándonos para culminar este proyecto.

AGRADECIMIENTO

Agradecemos a Dios por habernos dado su bendición de vida y habernos guiado a lo largo de nuestra experiencia, por darnos el apoyo espiritual que necesitábamos. También agradecer a las personas que estuvieron involucradas directamente con el desarrollo del proyecto, en especial al Dr. Alfredo Daza Vergaray, por habernos orientado en todo momento en que necesitábamos de consejos y motivación a lo largo del presente proyecto.

Índice de contenidos

RESUMEN.....	1
ABSTRACT.....	2
I. INTRODUCCIÓN	3
1.1. Descripción y formulación del Problema.....	3
1.1.1. Planteamiento del problema	3
1.1.2. Análisis del problema.....	4
1.2. Formulación del Problema	7
1.2.1. Problema General.....	7
1.2.2. Problemas específicos.....	7
1.3. Objetivos de la investigación	8
1.3.1. Objetivo General.....	8
1.3.2. Objetivos Específicos.....	8
1.4. Hipótesis.....	9
1.4.1. Hipótesis General	9
1.4.2. Hipótesis específicas	9
II. MARCO TEÓRICO	11
2.1. Antecedentes.....	11
2.2. Marco Conceptual	17
2.2.1. Definición de Sistema inteligente	17
2.2.2. Definición de Inteligencia artificial	17
2.2.3. Oversampling	18
2.2.4. Metodologías en Data Mining	18
2.2.5. Definición de Machine Learning	20
2.2.6. Algoritmos de Machine Learning.....	20
2.2.7. Métricas de evaluación.....	23
2.5.6. Herramienta	25
2.5.7. Lenguaje de programación	26
2.5.9. Definición de Accidentes Laborales	27
III. Materiales y métodos.....	28
3.1. Materiales	28
3.2. Metodología	30
3.2.1. Tipo de investigación	30
3.3. Población y muestra.....	31

3.4. Técnicas e instrumentos de recolección de datos.....	32
3.5. Procedimientos	33
3.6. Análisis de datos	33
3.7. Consideraciones éticas.....	34
IV. Resultados y discusión	35
V. Conclusiones y recomendaciones.....	55
5.1. Conclusiones.....	55
5.2. Recomendaciones.....	57
Referencias	58
ANEXO	64

Índice de tablas

Tabla 1.....	6
<i>Accidentes laborales dentro de las áreas de la Corporación de Alimentos Marítimo S.A.C. durante los meses de setiembre y diciembre de 2024</i>	<i>6</i>
Tabla 2.....	18
<i>Comparación entre las Metodologías KDD, SEMMA y CRISP</i>	<i>18</i>
Tabla 3.....	28
<i>Materiales</i>	<i>28</i>
Tabla 4.....	28
<i>Detalle de materiales</i>	<i>28</i>
Tabla 5.....	28
<i>Software</i>	<i>28</i>
Tabla 6.....	29
<i>Detalle de software</i>	<i>29</i>
Tabla 7.....	29
<i>Hardware</i>	<i>29</i>
Tabla 8.....	29
<i>Detalle hardware.....</i>	<i>29</i>
Tabla 9.....	30
<i>Otros</i>	<i>30</i>
Tabla 10.....	30
<i>Detalle otros.....</i>	<i>30</i>
Tabla 11. <i>Rendimiento de los modelos según la métrica Accuracy.....</i>	<i>35</i>
Tabla 12. <i>ANOVA de la métrica Accuracy.....</i>	<i>36</i>
Tabla 13. <i>Tukey de la métrica Accuracy.....</i>	<i>37</i>
Tabla 14. <i>Rendimiento de los modelos según la métrica Precision.....</i>	<i>38</i>
Tabla 15. <i>ANOVA de la métrica Precision</i>	<i>39</i>
Tabla 16. <i>Tukey de la métrica Precision.....</i>	<i>40</i>
Tabla 17. <i>Rendimiento de los modelos según la métrica Recall</i>	<i>41</i>
Tabla 18. <i>ANOVA de la métrica Recall</i>	<i>42</i>
Tabla 19. <i>Tukey de la métrica Recall</i>	<i>43</i>
Tabla 20. <i>Rendimiento de los modelos según la métrica F1 Score.....</i>	<i>44</i>
Tabla 21. <i>ANOVA de la métrica F1 Score</i>	<i>46</i>
Tabla 22. <i>Tukey de la métrica F1 Score.....</i>	<i>46</i>
Tabla 23. <i>Rendimiento de los modelos según la métrica ROC AUC</i>	<i>48</i>
Tabla 24. <i>ANOVA la métrica ROC AUC.....</i>	<i>49</i>

Tabla 25. <i>Tukey de la métrica ROC AUC</i>	49
Tabla 26. <i>Rendimiento de los modelos según la métrica MCC</i>	51
Tabla 27. <i>ANOVA de la métrica MCC</i>	52
Tabla 28. <i>Tukey de la métrica MCC</i>	53
Tabla 29.....	64
<i>Matriz de consistencia</i>	64
Tabla 30.....	65
Matriz de operacionalización de variable.....	65

Índice de figuras

Figura 1	6
<i>Distribución de los accidentes laborales registrados en la Corporación de Alimentos Marítimo S.A.C. (noviembre-diciembre 2024)</i>	<i>6</i>
Figura 2	24
<i>Curva ROC</i>	<i>24</i>
Figura 3	31
<i>Diseño no experimental</i>	<i>31</i>
Figura 4. <i>Promedio de los modelos según la métrica Accuracy</i>	<i>35</i>
Figura 5. <i>Promedio de los modelos según la métrica Precision</i>	<i>38</i>
Figura 6. <i>Promedio de los modelos según la métrica Recall</i>	<i>41</i>
Figura 7. <i>Promedio de los modelos según la métrica F1 Score</i>	<i>45</i>
Figura 8. <i>Promedio de los modelos según la métrica ROC AUC.....</i>	<i>48</i>
Figura 9. <i>Promedio de los modelos según la métrica MCC.....</i>	<i>51</i>
Figura 10	72
<i>Cronograma de actividades para ejecución del proyecto de investigación</i>	<i>72</i>
Figura 11	73
<i>Cronograma de actividades lineal para ejecución del proyecto de investigación</i>	<i>73</i>
Figura 12	75
<i>Mapa mental de antecedentes.....</i>	<i>75</i>
Figura 13	76
<i>Mapa mental de marco teórico</i>	<i>76</i>
Figura 14	78
<i>Estructura Orgánica de la Corporación De Alimentos Marítimo S.A.C.</i>	<i>78</i>
Figura 15. <i>Importación de librerías y carga de datos</i>	<i>78</i>
Figura 16. <i>Identificación de la variable objetivo</i>	<i>79</i>
Figura 17. <i>Particionamiento del conjunto de datos</i>	<i>79</i>
Figura 18. <i>Código de hiperparámetros y construcción del modelo Árbol de Decisión.....</i>	<i>79</i>
Figura 19. <i>Código para el cálculo de métricas de evaluación</i>	<i>80</i>
Figura 20. <i>Código para generar la matriz de confusión y curva ROC</i>	<i>80</i>
Figura 21. <i>Matriz de Confusión – Test (CV = 2) del modelo Árbol de Decisión</i>	<i>80</i>
Figura 22. <i>Curva ROC – Test (CV = 2) del modelo Árbol de Decisión</i>	<i>81</i>
Figura 23. <i>Matriz de Confusión – Test (CV = 3) del modelo Árbol de Decisión</i>	<i>81</i>
Figura 24. <i>Curva ROC – Test (CV = 3) del modelo Árbol de Decisión</i>	<i>82</i>
Figura 25. <i>Matriz de Confusión – Test (CV = 4) del modelo Árbol de Decisión</i>	<i>82</i>
Figura 26. <i>Curva ROC – Test (CV = 4) del modelo Árbol de Decisión</i>	<i>83</i>

Figura 27. <i>Matriz de Confusión – Test (CV = 5) del modelo Árbol de Decisión</i>	83
Figura 28. <i>Curva ROC – Test (CV = 5) del modelo Árbol de Decisión</i>	84
Figura 29. <i>Matriz de Confusión – Test (CV = 6) del modelo Árbol de Decisión</i>	84
Figura 30. <i>Curva ROC – Test (CV = 6) del modelo Árbol de Decisión</i>	85
Figura 31. <i>Matriz de Confusión – Test (CV = 7) del modelo Árbol de Decisión</i>	85
Figura 32. <i>Curva ROC – Test (CV = 7) del modelo Árbol de Decisión</i>	86
Figura 33. <i>Matriz de Confusión – Test (CV = 8) del modelo Árbol de Decisión</i>	86
Figura 34. <i>Curva ROC – Test (CV = 8) del modelo Árbol de Decisión</i>	87
Figura 35. <i>Matriz de Confusión – Test (CV = 9) del modelo Árbol de Decisión</i>	87
Figura 36. <i>Curva ROC – Test (CV = 9) del modelo Árbol de Decisión</i>	88
Figura 37. <i>Matriz de Confusión – Test (CV = 10) del modelo Árbol de Decisión</i>	88
Figura 38. <i>Curva ROC – Test (CV = 10) del modelo Árbol de Decisión</i>	89
Figura 39. <i>Matriz de Confusión – Test (CV = 11) del modelo Árbol de Decisión</i>	89
Figura 40. <i>Curva ROC – Test (CV = 11) del modelo Árbol de Decisión</i>	90
Figura 41. <i>Importación de librerías y carga de datos</i>	90
Figura 42. <i>Identificación de la variable objetivo</i>	91
Figura 43. <i>Particionamiento del conjunto de datos</i>	91
Figura 44. <i>Código de hiperparámetros y construcción del modelo KNN</i>	91
Figura 45. <i>Código para el cálculo de métricas de evaluación</i>	92
Figura 46. <i>Código para generar la matriz de confusión y curva ROC</i>	92
Figura 47. <i>Matriz de Confusión – Test (CV = 2) del modelo KNN</i>	92
Figura 48. <i>Curva ROC – Test (CV = 2) del modelo KNN</i>	93
Figura 49. <i>Matriz de Confusión – Test (CV = 3) del modelo KNN</i>	93
Figura 50. <i>Curva ROC – Test (CV = 3) del modelo KNN</i>	94
Figura 51. <i>Matriz de Confusión – Test (CV = 4) del modelo KNN</i>	94
Figura 52. <i>Curva ROC – Test (CV = 4) del modelo KNN</i>	95
Figura 53. <i>Matriz de Confusión – Test (CV = 5) del modelo KNN</i>	95
Figura 54. <i>Curva ROC – Test (CV = 5) del modelo KNN</i>	96
Figura 55. <i>Matriz de Confusión – Test (CV = 6) del modelo KNN</i>	96
Figura 56. <i>Curva ROC – Test (CV = 6) del modelo KNN</i>	97
Figura 57. <i>Matriz de Confusión – Test (CV = 7) del modelo KNN</i>	97
Figura 58. <i>Curva ROC – Test (CV = 7) del modelo KNN</i>	98
Figura 59. <i>Matriz de Confusión – Test (CV = 8) del modelo KNN</i>	98
Figura 60. <i>Curva ROC – Test (CV = 8) del modelo KNN</i>	99
Figura 61. <i>Matriz de Confusión – Test (CV = 9) del modelo KNN</i>	99
Figura 62. <i>Curva ROC – Test (CV = 9) del modelo KNN</i>	100

Figura 63. <i>Matriz de Confusión – Test (CV = 10) del modelo KNN</i>	100
Figura 64. <i>Curva ROC – Test (CV = 10) del modelo KNN</i>	101
Figura 65. <i>Matriz de Confusión – Test (CV = 11) del modelo KNN</i>	101
Figura 66. <i>Curva ROC – Test (CV = 11) del modelo KNN</i>	102
Figura 67. <i>Importación de librerías y carga de datos</i>	102
Figura 68. <i>Identificación de la variable objetivo</i>	103
Figura 69. <i>Particionamiento del conjunto de datos</i>	103
Figura 70. <i>Código de hiperparámetros y construcción del modelo Random Forest</i>	103
Figura 71. <i>Código para el cálculo de métricas de evaluación</i>	104
Figura 72. <i>Código para generar la matriz de confusión y curva ROC</i>	104
Figura 73. <i>Matriz de Confusión – Test (CV = 2) del modelo Random Forest</i>	104
Figura 74. <i>Curva ROC – Test (CV = 2) del modelo Random Forest</i>	105
Figura 75. <i>Matriz de Confusión – Test (CV = 3) del modelo Random Forest</i>	105
Figura 76. <i>Curva ROC – Test (CV = 3) del modelo Random Forest</i>	106
Figura 77. <i>Matriz de Confusión – Test (CV = 4) del modelo Random Forest</i>	106
Figura 78. <i>Curva ROC – Test (CV = 4) del modelo Random Forest</i>	107
Figura 79. <i>Matriz de Confusión – Test (CV = 5) del modelo Random Forest</i>	107
Figura 80. <i>Curva ROC – Test (CV = 5) del modelo Random Forest</i>	108
Figura 81. <i>Matriz de Confusión – Test (CV = 6) del modelo Random Forest</i>	108
Figura 82. <i>Curva ROC – Test (CV = 6) del modelo Random Forest</i>	109
Figura 83. <i>Matriz de Confusión – Test (CV = 7) del modelo Random Forest</i>	109
Figura 84. <i>Curva ROC – Test (CV = 7) del modelo Random Forest</i>	110
Figura 85. <i>Matriz de Confusión – Test (CV = 8) del modelo Random Forest</i>	110
Figura 86. <i>Curva ROC – Test (CV = 8) del modelo Random Forest</i>	111
Figura 87. <i>Matriz de Confusión – Test (CV = 9) del modelo Random Forest</i>	111
Figura 88. <i>Curva ROC – Test (CV = 9) del modelo Random Forest</i>	112
Figura 89. <i>Matriz de Confusión – Test (CV = 10) del modelo Random Forest</i>	112
Figura 90. <i>Curva ROC – Test (CV = 10) del modelo Random Forest</i>	113
Figura 91. <i>Matriz de Confusión – Test (CV = 11) del modelo Random Forest</i>	113
Figura 92. <i>Curva ROC – Test (CV = 11) del modelo Random Forest</i>	114
Figura 93. <i>Importación de librerías y carga de datos</i>	114
Figura 94. <i>Identificación de la variable objetivo</i>	114
Figura 95. <i>Particionamiento del conjunto de datos</i>	115
Figura 96. <i>Código de hiperparámetros y construcción del modelo SVM</i>	115
Figura 97. <i>Código para el cálculo de métricas de evaluación</i>	115
Figura 98. <i>Código para generar la matriz de confusión y curva ROC</i>	116

Figura 99. <i>Matriz de Confusión – Test (CV = 2) del modelo SVM</i>	116
Figura 100. <i>Curva ROC – Test (CV = 2) del modelo SVM</i>	117
Figura 101. <i>Matriz de Confusión – Test (CV = 3) del modelo SVM</i>	117
Figura 102. <i>Curva ROC – Test (CV = 3) del modelo SVM</i>	118
Figura 103. <i>Matriz de Confusión – Test (CV = 4) del modelo SVM</i>	118
Figura 104. <i>Curva ROC – Test (CV = 4) del modelo SVM</i>	119
Figura 105. <i>Matriz de Confusión – Test (CV = 5) del modelo SVM</i>	119
Figura 106. <i>Curva ROC – Test (CV = 5) del modelo SVM</i>	120
Figura 107. <i>Matriz de Confusión – Test (CV = 6) del modelo SVM</i>	120
Figura 108. <i>Curva ROC – Test (CV = 6) del modelo SVM</i>	121
Figura 109. <i>Matriz de Confusión – Test (CV = 7) del modelo SVM</i>	121
Figura 110. <i>Curva ROC – Test (CV = 7) del modelo SVM</i>	122
Figura 111. <i>Matriz de Confusión – Test (CV = 8) del modelo SVM</i>	122
Figura 112. <i>Curva ROC – Test (CV = 8) del modelo SVM</i>	123
Figura 113. <i>Matriz de Confusión – Test (CV = 9) del modelo SVM</i>	123
Figura 114. <i>Curva ROC – Test (CV = 9) del modelo SVM</i>	124
Figura 115. <i>Matriz de Confusión – Test (CV = 10) del modelo SVM</i>	124
Figura 116. <i>Curva ROC – Test (CV = 10) del modelo SVM</i>	125
Figura 117. <i>Matriz de Confusión – Test (CV = 11) del modelo SVM</i>	125
Figura 118. <i>Curva ROC – Test (CV = 11) del modelo SVM</i>	126
Figura 119. <i>Importación de librerías y carga de datos</i>	126
Figura 120. <i>Identificación de la variable objetivo</i>	126
Figura 121. <i>Particionamiento del conjunto de datos</i>	127
Figura 122. <i>Código de hiperparámetros y construcción del modelo Redes Bayesianas</i>	127
Figura 123. <i>Código para el cálculo de métricas de evaluación</i>	127
Figura 124. <i>Código para generar la matriz de confusión y curva ROC</i>	128
Figura 125. <i>Matriz de Confusión – Test (CV = 2) del modelo Redes Bayesianas</i>	128
Figura 126. <i>Curva ROC – Test (CV = 2) del modelo Redes Bayesianas</i>	129
Figura 127. <i>Matriz de Confusión – Test (CV = 3) del modelo Redes Bayesianas</i>	129
Figura 128. <i>Curva ROC – Test (CV = 3) del modelo Redes Bayesianas</i>	130
Figura 129. <i>Matriz de Confusión – Test (CV = 4) del modelo Redes Bayesianas</i>	130
Figura 130. <i>Curva ROC – Test (CV = 4) del modelo Redes Bayesianas</i>	131
Figura 131. <i>Matriz de Confusión – Test (CV = 5) del modelo Redes Bayesianas</i>	131
Figura 132. <i>Curva ROC – Test (CV = 5) del modelo Redes Bayesianas</i>	132
Figura 133. <i>Matriz de Confusión – Test (CV = 6) del modelo Redes Bayesianas</i>	132
Figura 134. <i>Curva ROC – Test (CV = 6) del modelo Redes Bayesianas</i>	133

Figura 135. <i>Matriz de Confusión – Test (CV = 7) del modelo Redes Bayesianas</i>	133
Figura 136. <i>Curva ROC – Test (CV = 7) del modelo Redes Bayesianas</i>	134
Figura 137. <i>Matriz de Confusión – Test (CV = 8) del modelo Redes Bayesianas</i>	134
Figura 138. <i>Curva ROC – Test (CV = 8) del modelo Redes Bayesianas</i>	135
Figura 139. <i>Matriz de Confusión – Test (CV = 9) del modelo Redes Bayesianas</i>	135
Figura 140. <i>Curva ROC – Test (CV = 9) del modelo Redes Bayesianas</i>	136
Figura 141. <i>Matriz de Confusión – Test (CV = 10) del modelo Redes Bayesianas</i>	136
Figura 142. <i>Curva ROC – Test (CV = 10) del modelo Redes Bayesianas</i>	137
Figura 143. <i>Matriz de Confusión – Test (CV = 11) del modelo Redes Bayesianas</i>	137
Figura 144. <i>Curva ROC – Test (CV = 11) del modelo Redes Bayesianas</i>	137
Figura 145. <i>Importación de librerías y carga de datos</i>	138
Figura 146. <i>Identificación de la variable objetivo</i>	138
Figura 147. <i>Particionamiento del conjunto de datos</i>	139
Figura 148. <i>Código de hiperparámetros y construcción del modelo Gradiente</i>	139
Figura 149. <i>Código para el cálculo de métricas de evaluación</i>	139
Figura 150. <i>Código para generar la matriz de confusión y curva ROC</i>	140
Figura 151. <i>Matriz de Confusión – Test (CV = 2) del modelo Gradiente</i>	140
Figura 152. <i>Curva ROC – Test (CV = 2) del modelo Gradiente</i>	141
Figura 153. <i>Matriz de Confusión – Test (CV = 3) del modelo Gradiente</i>	141
Figura 154. <i>Curva ROC – Test (CV = 3) del modelo Gradiente</i>	142
Figura 155. <i>Matriz de Confusión – Test (CV = 4) del modelo Gradiente</i>	142
Figura 156. <i>Curva ROC – Test (CV = 4) del modelo Gradiente</i>	143
Figura 157. <i>Matriz de Confusión – Test (CV = 5) del modelo Gradiente</i>	143
Figura 158. <i>Curva ROC – Test (CV = 5) del modelo Gradiente</i>	144
Figura 159. <i>Matriz de Confusión – Test (CV = 6) del modelo Gradiente</i>	144
Figura 160. <i>Curva ROC – Test (CV = 6) del modelo Gradiente</i>	145
Figura 161. <i>Matriz de Confusión – Test (CV = 7) del modelo Gradiente</i>	145
Figura 162. <i>Curva ROC – Test (CV = 7) del modelo Gradiente</i>	146
Figura 163. <i>Matriz de Confusión – Test (CV = 8) del modelo Gradiente</i>	146
Figura 164. <i>Curva ROC – Test (CV = 8) del modelo Gradiente</i>	147
Figura 165. <i>Matriz de Confusión – Test (CV = 9) del modelo Gradiente</i>	147
Figura 166. <i>Curva ROC – Test (CV = 9) del modelo Gradiente</i>	148
Figura 167. <i>Matriz de Confusión – Test (CV = 10) del modelo Gradiente</i>	148
Figura 168. <i>Curva ROC – Test (CV = 10) del modelo Gradiente</i>	149
Figura 169. <i>Matriz de Confusión – Test (CV = 11) del modelo Gradiente</i>	149
Figura 170. <i>Curva ROC – Test (CV = 11) del modelo Gradiente</i>	150

RESUMEN

La prevención de accidentes laborales representa un desafío para la empresa del sector pesquero Corporación de Alimentos Marítimo S.A.C, siendo fundamental implementar herramientas tecnológicas que permitan predecir situaciones de riesgo. La presente tesis tuvo como objetivo general aplicar un modelo basado en inteligencia artificial y Oversampling para predecir los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. Se evaluaron los desempeños de los modelos mediante las métricas Accuracy (Exactitud), Precision (Precisión), Recall (Sensibilidad), F1-Score (Puntuación F1), ROC-AUC (Área bajo la curva ROC) y Matthews Correlation Coefficient (Coeficiente de correlación de Matthews, MCC). El estudio se desarrolló bajo un enfoque cuantitativo, de tipo aplicado, con un diseño no experimental y subtipo transversal correlacional-predictivo. La metodología estuvo basada en el proceso KDD (Knowledge Discovery in Databases), que permitió identificar patrones relevantes a partir de los datos. La población estuvo conformada por 110 registros de accidentes laborales de trabajadores de la empresa pesquera. Se empleó la técnica de análisis de registros y se utilizaron fichas de registro como instrumento. Como resultado, el modelo de Gradiente se mostró óptimo, alcanzando 100% en Accuracy, Precision, F1-Score, ROC-AUC y Matthews Correlation Coefficient, mientras que en Recall destacaron con 100% tanto Gradiente como Random Forest, demostrando su efectividad y confiabilidad para predecir y prevenir accidentes laborales en la pesquera. Como conclusión, el uso de modelos de Inteligencia Artificial y Oversampling permite predecir de manera confiable los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C., siendo Gradiente el modelo con un desempeño superior.

Palabras clave: Accidentes laborales, Predicción, Inteligencia Artificial, Oversampling, Seguridad laboral, Pesquera.

ABSTRACT

Preventing workplace accidents presents a challenge for the fishing company Corporación de Alimentos Marítimo S.A.C., making it essential to implement technological tools that allow for predicting risk situations. This thesis aimed to apply a model based on Artificial Intelligence and Oversampling to predict workplace accidents at Corporación de Alimentos Marítimo S.A.C. Model performance was evaluated using the metrics Accuracy, Precision, Recall, F1 Score, ROC-AUC (Area Under the ROC Curve), and Matthews Correlation Coefficient (MCC). The study employed a quantitative, applied approach with a non-experimental, cross-sectional correlational-predictive design. The methodology was based on the Knowledge Discovery in Databases (KDD) process, which allowed for the identification of relevant patterns within the data. The study population consisted of 110 records of workplace accidents involving employees of the fishing company. Document analysis was employed, and accident records were used as the data collection instrument. The results showed that the Gradient model was the most effective, achieving 100% accuracy, precision, F1-score, ROC-AUC, and Matthews correlation coefficient. In terms of recall, both the Gradient and Random Forest models achieved 100%, demonstrating their effectiveness and reliability in predicting and preventing workplace accidents at the fishing company. In conclusion, the use of Artificial Intelligence and Oversampling models allows for the reliable prediction of workplace accidents at Corporación de Alimentos Marítimo S.A.C., with the Gradient model demonstrating superior performance.

Keywords: Workplace accidents, Prediction, Artificial Intelligence, Oversampling, Occupational safety, Fishing company.

I. INTRODUCCIÓN

1.1. Descripción y formulación del Problema

1.1.1. Planteamiento del problema

Los accidentes laborales en entornos productivos continúan siendo una preocupación constante, ya que afectan directamente el bienestar del personal y generan importantes pérdidas económicas para las empresas. A pesar de los avances en medidas de seguridad y en la mecanización de procesos, la combinación de maquinaria, herramientas cortantes y superficies húmedas representa un riesgo constante para los trabajadores. En este contexto, los métodos tradicionales de gestión de seguridad suelen actuar de forma reactiva, basándose en el análisis posterior de los incidentes (Rahman et al., 2025).

A nivel internacional, se calcula que cada año se registran aproximadamente tres millones de muertes relacionadas con la actividad laboral, de las cuales 330.000 se deben a accidentes. Los sectores considerados más peligrosos incluyen la agricultura, la construcción, la silvicultura, la pesca y la industria manufacturera. En conjunto, estos representan el 63% de todas las muertes laborales, con aproximadamente 200,000 lesiones fatales al año (Organización Internacional del Trabajo, 2023) De manera similar, Nazarihaghighipashaki et al. (2024), destacan que las industrias primarias, como las empresas pesqueras, registran un elevado número de lesiones fatales, lo que hace relevante la necesidad de que las empresas dentro de estas industrias puedan implementar medidas de seguridad más estrictas en estos sectores de alto riesgo.

En el contexto peruano, durante diciembre de 2024 se reportaron 3,568 notificaciones de incidentes laborales. De este total, el 98.43% correspondió a accidentes de trabajo que no son mortales, por otro lado, el 0.53% a accidentes que fueron mortales, el 0.90% a incidentes considerados peligrosos y, por último, el 0.14% a enfermedades ocupacionales. Específicamente en el sector pesquero, se notificaron 31 incidentes laborales: 30 fueron accidentes no mortales y uno fue mortal (Ministerio de Trabajo y Promoción del Empleo, 2024).

Ante este panorama, el uso de tecnologías emergentes tales como el aprendizaje automático (machine learning) y la analítica de datos se presenta como una alternativa innovadora y más eficaz para anticipar y prevenir accidentes laborales. Estas herramientas permiten identificar patrones de riesgo y fortalecer la seguridad en las líneas de producción antes de que ocurran los incidentes (Gil de Zúñiga et al., 2024; Rahman et al., 2025).

1.1.2. Análisis del problema

La investigación se llevó a cabo en la empresa Corporación de Alimentos Marítimo S.A.C., ubicada en Jirón Huancavelica 1191, Pueblo Joven Florida Baja - Chimbote. Esta compañía está constituida como una Sociedad Anónima Cerrada (S.A.C.) y se especializa en el procesamiento de conservas de pescado. En las áreas pertenecientes al fileteado y envasado se identifica una alta frecuencia de accidentes laborales. En la Tabla 1 y la Figura 1 se muestran los tipos de accidentes ocurridos en la empresa entre setiembre y diciembre de 2024, en la cual se identifica que los cortes con cuchillos, atrapamientos, resbalones y caídas son los incidentes más comunes, en donde laboran 19 personas

La seguridad y salud ocupacional constituye un factor crítico dentro de las empresas del sector industrial y de procesamiento de alimentos, debido a la constante exposición de los trabajadores a riesgos mecánicos, físicos y ergonómicos durante el desarrollo de sus actividades laborales. En ese contexto, la Corporación de Alimentos Marítimo S.A.C. presenta diversos accidentes laborales asociados principalmente a las áreas de fileteado y envasado, los cuales afectan tanto la integridad del trabajador como la continuidad operativa de la empresa.

De acuerdo con los registros internos de la empresa correspondientes al periodo comprendido entre setiembre y diciembre de 2024, se reportaron 35 accidentes laborales, siendo los más frecuentes los cortes con cuchillos (8 casos), atrapamientos (6 casos), resbalones y caídas (6 casos), fatiga por movimientos repetitivos (5 casos) e inhalación de sustancias (4 casos). Asimismo, se identificaron accidentes relacionados con lesiones por máquinas y quemaduras, con 3 casos cada uno. Estos incidentes se presentan principalmente durante actividades de manipulación de cuchillos afilados, operación de maquinaria,

exposición a líquidos calientes y movimientos repetitivos propios del proceso industrial.

La ocurrencia recurrente de estos accidentes evidencia dificultades en la capacidad de anticipación y prevención de riesgos laborales dentro de la empresa, lo que genera consecuencias como lesiones físicas, interrupciones en la producción, ausentismo laboral, incremento de costos operativos y posibles afectaciones al cumplimiento de normas de seguridad ocupacional. Aunque la empresa dispone de registros históricos de accidentes y protocolos básicos de seguridad, actualmente no cuenta con mecanismos analíticos avanzados que permitan identificar patrones de riesgo ni predecir oportunamente la probabilidad de ocurrencia de accidentes laborales.

Además, los registros de accidentes presentan desbalance en la distribución de clases, debido a que algunos tipos de accidentes ocurren con menor frecuencia que otros, dificultando el desempeño de modelos tradicionales de predicción y limitando la identificación adecuada de eventos críticos poco frecuentes. Esta situación reduce la capacidad de generar estrategias preventivas basadas en datos y limita la toma de decisiones oportunas en materia de seguridad ocupacional.

Diversos estudios recientes señalan que los modelos basados en inteligencia artificial permiten analizar datos e identificar patrones asociados a accidentes laborales, contribuyendo a mejorar la prevención y gestión de riesgos dentro de las organizaciones. En ese sentido, surge la necesidad de desarrollar un modelo predictivo basado en inteligencia artificial y técnicas de oversampling que permita mejorar la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C., fortaleciendo así las acciones preventivas y la gestión de seguridad ocupacional.

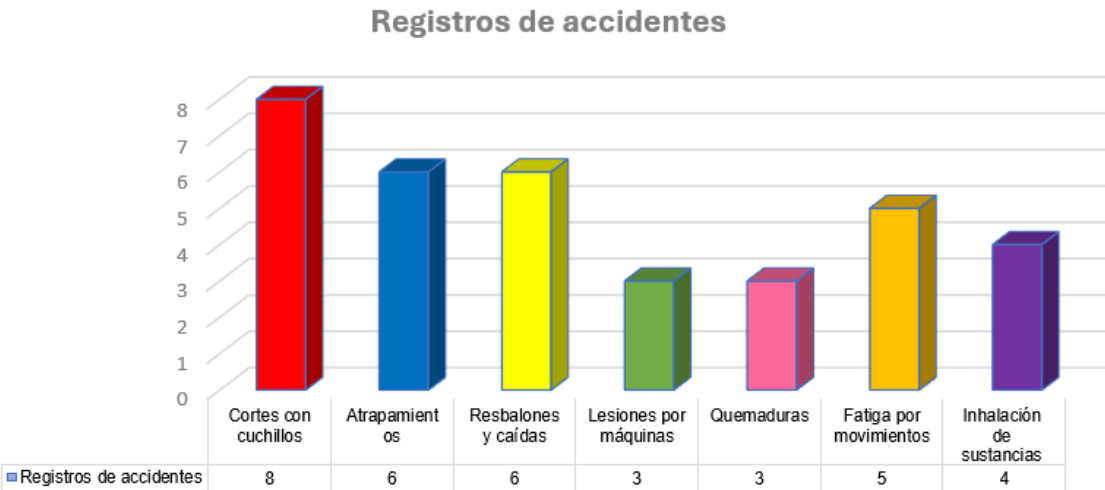
Tabla 1

Accidentes laborales dentro de las áreas de la Corporación de Alimentos Marítimo S.A.C. durante los meses de setiembre y diciembre de 2024

Tipo de accidente	Área	Descripción	Accidentes laborales registrados (setiembre-diciembre 2024)
Cortes con cuchillos	Fileteado	Lesiones al manipular cuchillos afilados para filetear	8
Atrapamientos	Envasado y Fileteado	Partes del cuerpo que han sido atrapados en maquinaria en funcionamiento	6
Resbalones y caídas	Envasado y Fileteado	Pérdida de equilibrio por condiciones en el entorno laboral	6
Lesiones por máquinas	Envasado y Fileteado	Golpes con objetos, piezas de pescado o equipos	3
Quemaduras	Envasado	Exposición a líquidos calientes durante el proceso	3
Fatiga por movimientos	Fileteado	Lesiones musculares o esguinces por movimientos repetitivos	5
Inhalación de sustancias	Envasado	Exposición a polvo de pescado u otras sustancias químicas	4
Total			35

Figura 1

Distribución de los accidentes laborales registrados en la Corporación de Alimentos Marítimo S.A.C. (noviembre-diciembre 2024)



1.2. Formulación del Problema

1.2.1. Problema General

- ¿En qué medida se predicen los accidentes laborales mediante un modelo basado en Inteligencia Artificial y Oversampling en la Corporación de Alimentos Marítimo S.A.C.?

1.2.2. Problemas específicos

- ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con Accuracy los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.?
- ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con precisión los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.?
- ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con Recall (Sensibilidad) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.?
- ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con F1 Score los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.?
- ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con ROC AUC los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.?
- ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con Matthews Correlation Coefficient (MCC) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.?

1.3. Objetivos de la investigación

1.3.1. Objetivo General

- Predecir los accidentes laborales mediante un modelo basado en Inteligencia Artificial y Oversampling en la Corporación de Alimentos Marítimo S.A.C.

1.3.2. Objetivos Específicos

- Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Accuracy los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con precisión los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Recall (Sensibilidad) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con F1 Score los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con ROC AUC los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Matthews Correlation Coefficient (MCC) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

1.4. Hipótesis

1.4.1. Hipótesis General

- Se predicen los accidentes laborales mediante un modelo basado en Inteligencia Artificial y Oversampling en la Corporación de Alimentos Marítimo S.A.C.

1.4.2. Hipótesis específicas

- Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Accuracy los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Precision los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Recall (Sensibilidad) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con F1 Score los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con ROC AUC los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.
- Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Matthews Correlation Coefficient (MCC) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

1.5. Justificación e importancia de la investigación

La importancia del presente estudio se fundamenta en los siguientes niveles:

A nivel teórico, aportó nuevos conocimientos sobre un modelo basado en inteligencia artificial y Oversampling para prevenir accidentes dentro de una empresa del sector pesquero, alineándose con la idea de que una investigación debe contribuir al desarrollo de teorías existentes, apoyarlas o cuestionarlas. El análisis del modelo predictivo permitió proporcionar información valiosa para mejorar la seguridad laboral en entornos industriales.

A nivel metodológico, se sustentó en la necesidad de diseñar nuevos métodos y herramientas que permitan un mejor entendimiento de la realidad (Ruiz & Valenzuela, 2022), por lo que este estudio implementó un sistema inteligente basado en oversampling y algoritmos de machine learning. Este enfoque metodológico no solo optimiza la predicción de incidentes laborales, sino que también puede ser de referencia para futuras investigaciones en inteligencia artificial y su uso para la seguridad laboral.

Desde la perspectiva social, la investigación cobra relevancia al beneficiar directamente a los trabajadores de la empresa del estudio, mejorando su seguridad y bienestar en áreas de alto riesgo. Esto se alinea con la idea de que una investigación debe tener un impacto bueno en la sociedad y en quienes serán los beneficiados por sus resultados (Ruiz & Valenzuela, 2022).

Finalmente, a nivel práctico, esta investigación responde a la necesidad de aplicar soluciones tecnológicas para resolver problemas concretos en el ámbito laboral (Ruiz & Valenzuela, 2022). Implementar un modelo inteligente basado en inteligencia artificial representa una innovación clave en la industria pesquera, optimizando los procesos de prevención de accidentes y contribuyendo a la sostenibilidad y eficiencia operativa.

II. MARCO TEÓRICO

2.1. Antecedentes

Diversos autores han documentado hallazgos a nivel internacional (véase Figura 12), los cuales permitieron contextualizar y fundamentar el presente estudio, los cuales son:

Luo et al. (2023) Tuvieron como objetivo general de investigación realizar la optimización de los algoritmos de aprendizaje automático para el desarrollo de un modelo de predicción de la gravedad de los accidentes laborales. La investigación se basa en una muestra de 264 informes de accidentes por colapso ocurridos dentro de la construcción en China entre 2013 y 2021. El estudio implementó un modelo basado en el algoritmo de Random Forest (RF), aplicando un preprocesamiento de datos mediante la herramienta Jieba en Python para segmentar los textos de los informes de manera automática. La optimización del modelo se realizó en tres fases: la primera, utilizando todos los atributos disponibles; la segunda, aplicando una selección de atributos críticos basados en la importancia de las características; y la tercera, ajustando los hiperparámetros del modelo. Los resultados indicaron que, tras estos ajustes, el modelo alcanzó un accuracy de hasta el 82%, lo que demuestra la efectividad del algoritmo RF para predecir la gravedad de los accidentes laborales en la construcción.

Zhou et al., (2025) llevaron a cabo un estudio donde tuvo como objetivo principal de desarrollar un Grafo de Conocimiento de accidentes, específicamente de Caídas desde Altura (FFH-KG), para respaldar el diseño de Sistemas de Protección contra Caídas (FPS). Se combinaron la recolección de datos con el uso de algoritmos de Machine Learning (ML) y Modelos de Lenguaje Grande (LLM) para procesar datos no estructurados. La muestra estuvo compuesta por 14 documentos de Especificaciones de Equipos de Protección (FPE-S) y 941 Informes de Accidentes laborales por Caídas desde Altura (FFH-AR) provenientes de China. Los instrumentos utilizados incluyeron el procesamiento de datos con algoritmos de ML y LLM para categorizar los Informes de Accidentes por Caídas desde Altura y extraer relaciones relevantes. Como resultados más relevantes, se destacó la creación de un Grafo de Conocimiento

de accidentes, en este caso Caídas desde Altura (FFH-KG), mejorando significativamente la eficiencia en el diseño de Sistemas de Protección contra Caídas (FPS). En cuanto a Precisión, el modelo Bi-LSTM-CRF alcanzó el valor más alto con 97.48 %. Para la métrica de Recall, Bi-LSTM-CRF destacó con 98.10. Finalmente, en el F1 Score, el mejor desempeño lo presentó Bi-LSTM-CRF con 97.79 %, seguido de LSTM-CRF con 96.57 % y CRF con 87.09 %. Además, el accuracy de extracción de información mediante el modelo de sistema de inteligencia artificial Kimi alcanzó un 96,18 % para escenarios de trabajo en altura (WAH) y un 94,66 % para las causas de caídas. Este avance proporcionó un modelo valioso para que se pueda tomar decisiones informadas en el diseño de sistemas de protección contra caídas, contribuyendo así a la mejora de la seguridad en el entorno laboral.

En la investigación de Sarkar et al. (2019) el objetivo principal fue desarrollar un modelo para la predicción de accidentes laborales haciendo uso de técnicas de machine learning. La muestra estuvo compuesta por datos de accidentes que fueron recopilados de la base de datos del sistema electrónico de gestión de seguridad de una planta siderúrgica integrada en India, entre 2010 y 2013, utilizando un total de 1500 registros de accidentes laborales. En el procedimiento se utilizó la red neuronal artificial (ANN) y la máquina de vectores de soporte (SVM), para predecir los resultados de incidentes ocupacionales. Para lograr la mejor precisión, se emplearon técnicas de optimización tanto la optimización por enjambre de partículas (PSO) como el algoritmo genético (GA), en los clasificadores. Los resultados del análisis revelaron que la máquina de vectores de soporte SVM basado en PSO superó a otros clasificadores en términos de precisión, alcanzando un 90,67% de accuracy. Los hallazgos también destacaron que utilizar algoritmos de aprendizaje automático (ML) permite predecir los accidentes laborales.

En el artículo elaborado por Tamascelli et al. (2022) tienen como objetivo determinar si los métodos de machine learning pueden aprovechar el conocimiento integrado en las bases de datos de accidentes para hacer la predicción de los resultados de accidentes e incidentes nuevos. Para ello, se utilizó una muestra de 7,100 eventos, y un conjunto de datos de evaluación que contiene 1,872 eventos. Donde el método consta del preprocesamiento de datos

y las simulaciones de aprendizaje automático. Se ha propuesto un enfoque genérico basado en algoritmos de clasificación para predecir los resultados de accidentes en términos de fallecidos y heridos. El método ha sido probado en una base de datos específica, MHIDAS, y para ello se han utilizado tres modelos de clasificación: Wide, Deep y Wide&Deep. Los resultados indican que el modelo Deep logra un Recall de 0.91 y una Precisión de 0.29, lo que significa que el modelo puede identificar 9 de cada 10 eventos que causaron entre 1 y 10 muertos, con una memoria de precisión del 29%. Los resultados sugieren que se pueden utilizar métodos de análisis avanzados, como el aprendizaje automático, para realizar predicciones sobre la gravedad de nuevos accidentes.

Qi et al. (2025) realizaron un estudio con el objetivo de predecir y diagnosticar los riesgos de accidentes en los lugares de trabajo, en este caso caídas, utilizando un enfoque basado en datos de accidentes y técnicas híbridas de inteligencia artificial (IA), centrándose en 914 accidentes laborales en la República Popular China. Emplearon seis algoritmos base de aprendizaje automático: árbol de decisión, k vecinos más cercanos, perceptrón multicapa, bayesiano ingenuo, bosque aleatorio y máquina de vectores de soporte. Estos modelos fueron combinados en un modelo de aprendizaje por conjuntos mediante un algoritmo de votación suave con pesos desiguales. Los factores causales extraídos de los informes de accidentes se adoptaron como nodos en la Red Bayesiana de Riesgo de Caídas (FRBN). Para el aprendizaje de su estructura, se utilizaron los algoritmos de escalada de colinas (HC) y Bayes ingenuo aumentado con árbol (TAN). Además, el modelo fue validado mediante validación cruzada y tres funciones de puntuación (K2, BDeu y BIC). Los resultados mostraron que los modelos basados en redes bayesianas alcanzaron una alta precisión (>0.95), las otras dos redes bayesianas de FRBN basadas en HC-BDeu y TAN presentaron accuracy de 0,978 y 0,979, respectivamente, lo que demuestra la eficacia de las técnicas de IA híbrida para predecir y diagnosticar los riesgos de accidentes en el ámbito laboral.

Özkan y Ulaş (2024) realizaron un estudio con el objetivo de desarrollar un marco predictivo basado en aprendizaje automático (ML) utilizando datos de accidentes laborales para identificar las causas de amputación y muertes en el sector metalúrgico recopilados por el Ministerio de Trabajo y Seguridad Social de

Turquía (MLSS) entre 2013 y 2018. Para esto, se utilizó un conjunto de datos compuesto por 234 variables relacionadas con accidentes laborales, aplicando un proceso de preprocesamiento y limpieza antes de emplear los modelos de Machine learning: Random Forest (RF), K-Nearest Neighbors (KNN), Recursive Partitioning and Regression Trees (RPART) y Gradient Boosting Machine (GBM). Posteriormente, se utilizó validación cruzada K-fold (10-fold) para evaluar el rendimiento de los modelos, dividiendo el conjunto de datos en un 80% para entrenamiento y un 20% para validación, repitiendo el proceso 50 veces para garantizar la robustez de los resultados. Los hallazgos indicaron que el modelo Random Forest (RF) obtuvo el mejor desempeño con un accuracy de 0.9172, una precisión de 0.9618, una recall de 0.9518 y una puntuación F de 0.9568, superando a los demás modelos en la clasificación de la gravedad de los accidentes. Se recomienda la implementación del modelo RF en instituciones públicas y empresas para la prevención de accidentes laborales, permitiendo predecir incidentes futuros y tomar medidas preventivas más eficaces.

En el estudio de Gondia et al. (2023) tuvieron como principal objetivo el desarrollo de modelos predictivos con algoritmos de aprendizaje automático (ML) para predecir lesiones e indicadores adelantados de riesgo para la seguridad en diferentes zonas de la obra y a lo largo del ciclo de vida del proyecto. La investigación se basó en 1981 casos derivados de la base de datos de lesiones e incidentes de la OSHA. En este estudio, se emplearon modelos de ensemble (combinación de múltiples algoritmos), que incluyen máquinas de vectores de soporte (SVM), redes neuronales artificiales (ANN), bosques aleatorios (RF), árboles de decisión (DT) y Naïve Bayes (NB). En la evaluación del problema de tipo de lesión, el modelo Ensemble mostró el mayor desempeño en Accuracy (82.62 %), Sensitivity (85.39%), Precision (78.89%), Specificity (80.28%) y F-measure (81.98%). Sin embargo, considerando el ranking de modelos, el modelo ANN se posicionó como el mejor con un S Ranking de 79.51%. Por otro lado, para el problema de identificación de parte del cuerpo, el modelo Ensemble nuevamente obtuvo el mejor valor en Specificity (93.95%), mientras que también presentó buenos resultados en Accuracy (78.84%), Precision (79.01%), Sensitivity (78.84%) y F-measure (78.93%). No obstante, según el S Ranking, el modelo ANN fue considerado el mejor (80.15 %), Estos modelos pueden predecir

resultados de lesiones que actúan como indicadores principales de riesgo de seguridad en el lugar de trabajo.

Tixier et al. (2016) en su investigación, tiene como objetivo aplicar tanto Stochastic Gradient Tree Boosting (SGTB) como Random Forest (RF), dos algoritmos de aprendizaje automático para predecir resultados relacionados con la seguridad en la construcción. El uso de una lista avalados por investigaciones anteriores y un sistema de procesamiento del lenguaje natural (PLN) de alta precisión permitió constituir un gran conjunto de 4400 datos. En particular, los modelos SGTB lograron una habilidad más alta en comparación con los modelos RF. Los valores medianos de Puntuación de habilidad de probabilidad de rango (RPSS) fueron de 0.236 a 0.436 para los modelos SGTB, y de 0.172 a 0.319 para los modelos RF, lo que indica una habilidad de predicción de media-alta a muy alta. Los modelos son capaces de guiar la toma de decisiones bajo incertidumbre, lo que puede ser de gran utilidad para mejorar la planificación de las reuniones de seguridad previas para predicciones de accidentes laborales.

Chen & Luo (2016) en su estudio tiene como objetivo desarrollar algoritmos para la predicción del riesgo de caída y evaluar su rendimiento basándose en 1161 registros de accidentes obtenidos de la base de datos OSHA. Los datos fueron analizados mediante los algoritmos k-Modes, Árboles de Decisión (CART) y Redes Neuronales Artificiales (RBF). Se emplearon tres algoritmos de minería de datos para analizar los accidentes de caídas registrados por OSHA. Primero, se filtraron 1161 registros de accidentes de caídas con 15 parámetros específicos. Luego, se utilizó el algoritmo k-Modes para clasificar los accidentes en grupos, el algoritmo CART para construir un árbol de decisión que predijera la gravedad de las lesiones, y finalmente, una red neuronal de Funciones Radiales (RBF) se aplicó para predecir el nivel de gravedad, con la salida categórica procesada por el algoritmo k-Means. Los resultados mostraron que las características más importantes para predecir la gravedad de las caídas son la ocupación del trabajador, la distancia de la caída (FallDist), la parte del cuerpo afectada (PartBody) y la fuente de la lesión (SourceInjury). El modelo de Árbol de Decisión fue el más preciso, alcanzando un 75% de accuracy en la predicción de la gravedad de las lesiones. En comparación, k-Modes logró una accuracy de

62% y RBF una accuracy de 49%. Este estudio resalta la importancia de integrar estos modelos predictivos en sistemas de protección proactiva.

En su investigación, Khairuddin et al. (2022) propusieron un modelo predictivo optimizado para anticipar la gravedad de las lesiones ocupacionales, utilizando un conjunto de datos de trabajadores lesionados en diversas industrias. Aplicaron cinco algoritmos de clasificación, incluidos Árboles de Decisión (DT), Naïve Bayes (NB) y Bosques Aleatorios (RF), para predecir la gravedad de las lesiones. El preprocesamiento de los datos incluyó la eliminación de filas con valores faltantes y la estandarización de los datos. Posteriormente, el conjunto de datos fue dividido en entrenamiento (70%) y prueba (30%). Tras comparar el rendimiento de los modelos, se seleccionó el modelo con los mejores resultados para identificar las características clave en la gravedad de las lesiones. Los autores realizan un análisis comparativo de los modelos. En cuanto a Accuracy, el modelo que obtuvo el mejor resultado fue Random Forest con un valor de 0.949. Respecto a la precisión, el desempeño más alto lo presentó KNN, alcanzando 0.869. Para la métrica de Recall, SVM destacó con 0.967, mientras que en F1-Score RF logró el mayor valor con 0.909. Finalmente, en la métrica AUC, se observó que KNN, DT, SVM y RF tuvieron 0.95, siendo Naive Bayes el único modelo con 0.94.

Mosquera et al. (2021). En un estudio aplicado a una empresa del sector papelerero, se diseñó un sistema de clasificación orientado a la detección y prevención de accidentes laborales en bodegas donde se almacenan fibras. La propuesta se fundamentó en analizar las variables operativas como la circulación de peatones, el tránsito de maquinaria (bobcat, tractocamiones), accesos, zonas peatonales y presencia de barandas. Utilizando datos históricos de incidentes y accidentes registrados en planta, se realizaron el entrenamiento y la evaluación de distintos algoritmos de clasificación con la métrica accuracy, entre ellos redes bayesianas, Naive Bayes y árboles de decisión (J48). Los resultados mostraron que el modelo basado en árboles de decisión logró identificar correctamente el 90% de los casos, destacando su utilidad como modelo para la gestión preventiva de riesgos laborales. Este trabajo demuestra que la identificación de factores clave en la accidentalidad permite construir modelos predictivos efectivos que pueden integrarse en los procesos de seguridad industrial

2.2. Marco Conceptual

Se recopila una variedad de información utilizada para predecir accidentes laborales. Existen numerosos estudios (véase Figura 13) que proporcionan definiciones sobre el uso de sistemas inteligentes con inteligencia artificial, machine learning y técnicas de oversampling para prever eventos de riesgo. A continuación, se presentan los conceptos:

2.2.1. Definición de Sistema inteligente

El sistema inteligente se caracteriza por su capacidad para autoorganizarse, estar consciente del contexto y permitir la comunicación entre sus elementos. Además, estos sistemas son capaces de aprender de la experiencia, razonar, percibir tanto su entorno como a sí mismos y tomar decisiones para controlar su entorno. También tienen conocimiento integrado, lo que les permite tomar decisiones autónomas. Estas características permiten que los sistemas inteligentes se adapten a su entorno y optimicen sus funciones de manera eficiente, sin intervención humana directa (Romero et al., 2020).

2.2.2. Definición de Inteligencia artificial

La Inteligencia Artificial (IA) es entendida como la capacidad de los sistemas o máquinas creadas por el ser humano para ejecutar tareas, resolver problemas, comunicarse e interactuar de manera lógica, imitando el razonamiento y comportamiento humano (Gil de Zúñiga et al., 2024). Asimismo, La Inteligencia Artificial (IA) es un campo científico de gran relevancia debido a su carácter multidisciplinario, en el cual se crean sistemas capaces de llevar a cabo tareas y solucionar problemas de forma parecida a cómo lo haría una persona. Además, busca simular artificialmente los procesos de pensamiento y la manera en que el cerebro humano realiza la toma de decisiones (Vallejo et al., 2022).

2.2.3. Oversampling

Oversampling es una técnica utilizada para abordar el problema del desbalance de clases en conjuntos de datos, ampliando artificialmente la cantidad de muestras de la clase minoritaria. Este enfoque permite mejorar el rendimiento de los modelos de machine learning al proporcionar una distribución de datos más equilibrada, evitando que el modelo favorezca únicamente la clase mayoritaria (Luo et al., 2023)

2.2.4. Metodologías en Data Mining

Para llevar a cabo el estudio, existen varias metodologías, entre las que se destacan KDD, SEMMA y CRISP, todas ellas orientadas al proceso de minería de datos. Respecto a la investigación se optará por la metodología KDD, considerando las características descritas en la Tabla 2.

Tabla 2
Comparación entre las Metodologías KDD, SEMMA y CRISP

Metodología de Minería de Datos	Preprocesamiento	Procesamiento Principal	Postprocesamiento
CRISP-DM	Entendimiento del negocio, análisis de los datos y su preparación	Construcción del modelo	Validación del modelo, puesta en marcha
KDD	Selección y preprocesamiento de datos	Transformación y aplicación de técnicas de minería	Interpretación y evaluación de resultados
SEMMA	Muestreo, exploración y ajuste de datos	Desarrollo del modelo	Evaluación del desempeño

Fuente: Datos tomados del estudio de (Rogalewicz & Sika, 2016)

La metodología KDD suele emplearse en estudios de minería de datos y análisis de información; su aplicación no se restringe exclusivamente a grandes bases de datos, ya que su propósito principal consiste en estructurar el proceso de descubrimiento de conocimiento mediante etapas como la selección, preparación, transformación, minería e interpretación de datos, independientemente del tamaño de la muestra, siempre que existan variables susceptibles de análisis y modelamiento (Romero & Ventura). Este enfoque incluye varias etapas, que van desde la comprensión del problema y los objetivos del análisis, hasta la integración final de los resultados obtenidos (Safhi et al., 2019).

a) Selección: En esta fase se determinan las bases de datos que se emplearán.

b) Preprocesamiento: En esta etapa, se busca optimizar la calidad y la consistencia del conjunto de datos.

c) Transformación: Se lleva a cabo un análisis de las variables, donde se destaca la importancia que presentan, además de la interacción y las correlaciones.

d) Minería de Datos (Data Mining): Esta fase constituye el núcleo del proceso KDD, donde se utilizan algoritmos estadísticos y de machine learning para descubrir patrones, estructuras y reglas en los datos. Las principales actividades dentro de la minería de datos incluyen clasificación, agrupación, asociación, resumen y predicción.

e) Interpretación/Evaluación: Finalmente, se evalúa si los patrones identificados son realmente interesantes y útiles o si es necesario hacer ajustes adicionales (Safhi et al., 2019).

2.2.5. Definición de Machine Learning

Machine Learning proporciona modelos estadísticos y algoritmos que pueden realizar predicciones, clasificaciones y estimaciones, mejorando progresivamente su precisión a medida que aprenden de datos históricos. Además, permite el desarrollo de sistemas con capacidad de analizar volúmenes grandes de datos, identificar patrones y ofrecer soluciones eficientes para la reducción de riesgos en entornos laborales (Choi et al., 2020; Maheronnaghsh et al., 2023). Machine Learning se utiliza para enseñar a las máquinas a manejar datos de manera más eficiente. En muchas ocasiones, al analizar un conjunto de datos, no es posible interpretar directamente la información contenida en ellos. En estos casos, se aplica el aprendizaje automático. Dado el gran volumen de datos disponibles actualmente, la demanda de ML ha ido en aumento. Diversas industrias emplean esta tecnología para extraer información (Mahesh, 2020).

2.2.6. Algoritmos de Machine Learning

2.2.6.1. Árbol de decisión

Es una técnica utilizada para clasificar o predecir datos, que destaca por su capacidad para resolver problemas complejos a través de la creación de una estructura en forma de árbol. Esta estructura está formada por nodos y conexiones organizados jerárquicamente. En este árbol, cada nodo representa un atributo y la distribución de las instancias correspondientes (Choi et al., 2020).

$$Entropy(S) = \sum_{i=1}^c -P_i \log_2 P_i,$$

$$Gain(S, A) = Entropy(S) - \sum_{v \in Values(A)} \frac{|S_v|}{|S|} Entropy(S_v)$$

La entropía se utiliza para medir la incertidumbre en los conjuntos de datos y es clave en la toma de decisiones en el aprendizaje automático (Abolhosseini et al., 2024).

2.2.6.2. Vecinos más Cercanos (KNN)

En este tipo de clasificador, el conjunto de datos de entrenamiento está representado como un espacio con “m” dimensiones, donde se localizan las k instancias más próximas a un nuevo dato de prueba. La proximidad entre puntos se calcula usando una medida de distancia, como la distancia euclidiana (Ozturk et al., 2023).

$$d(x, y) = \sqrt{\sum_{i=1}^n (x_i - y_i)^2}$$

2.2.6.3. Random Forest

Random Forest es una combinación de árboles predictores, donde cada árbol tiene dependencia de un vector aleatorio muestreado de forma independiente y, por lo tanto, continúa con la misma distribución para todos los árboles dentro del bosque. Después de generar un gran número de árboles, estos realizan una votación por la clase más popular, a estos procedimientos se les llama bosques aleatorios (Breiman, 2001).

2.2.6.4. Máquina de Vector de Soporte

El SVM es un algoritmo eficaz para datos de alta dimensión, que encuentra un hiperplano óptimo para separar clases de datos. Su objetivo fundamental es identificar un hiperplano dentro de un espacio de “n” dimensiones que logra la separación de los diferentes puntos de datos según sus características. Este algoritmo se caracteriza por ignorar los valores atípicos y determinar el hiperplano óptimo que maximiza el margen entre las clases (Qi et al., 2025).

$$\text{Si } Y_i = +1; \quad wx_i + b \geq 0 \quad (1)$$

$$\text{Si } Y_i = -1; \quad wx_i + b \leq 0 \quad (2)$$

$$\text{Para todo } i; \quad Y_i (wx_i + b) \geq 0 \quad (3)$$

2.2.6.5. Red Bayesiana

Es un modelo gráfico que se estructura como un grafo dirigido sin ciclos (DAG), compuesto por un conjunto finito de nodos.

Este modelo muestra las relaciones de probabilidad entre variables (características), conectadas mediante arcos que representan influencias causales entre ellas (Sarkar & Maiti, 2020).

$$P\left(\frac{X}{Y}\right) = \frac{P\left(\frac{Y}{X}\right) \times P(X)}{P(Y)}$$

2.2.6.6. Gradiente

El descenso de gradiente es un algoritmo de optimización. Este es utilizado de forma que se mejora el rendimiento de una red neuronal mediante el ajuste de sus parámetros. Para llevar a cabo el descenso de gradiente, primero se deben calcular los gradientes. Para calcular el gradiente, necesitamos conocer la función de pérdida o de costo. Utilizaremos la función de costo para determinar la derivada. En cálculo, la derivada se refiere a la pendiente de una función en un punto dado; por lo tanto, básicamente estamos calculando la pendiente de la curva en función de la función de pérdida (Khasanov et al., 2021).

2.2.6.7. Redes neuronales artificiales

Las redes neuronales artificiales se usan para tareas de clasificación y regresión. Este tipo de modelo está inspirado en el funcionamiento del cerebro del ser humano, formado por una red de neuronas artificiales organizadas en tres tipos de capas (Yedla et al., 2020).

2.2.7. Métricas de evaluación

2.2.7.1. Accuracy

La exactitud es una medida que sirve para evaluar qué tan bien funciona un modelo de clasificación, y se calcula como el porcentaje de predicciones acertadas sobre el total de predicciones realizadas (Ozturk Kiyak et al., 2023).

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN}$$

2.2.7.2. Precisión

Mide la proporción de instancias relevantes entre todas las instancias recuperadas. En otras palabras, la precisión se calcula haciendo la división entre el número de verdaderos positivos (TP) con el total de predicciones positivas realizadas, que incluye tanto los verdaderos positivos como los falsos positivos (Ozturk Kiyak et al., 2023).

$$Precision = \frac{TP}{TP+FP}$$

2.2.7.3. Recall (Sensibilidad)

Indica qué proporción de los casos relevantes fueron identificados correctamente por el modelo. En otras palabras, recall evalúa cuántas veces el modelo ha identificado correctamente las instancias de la clase positiva, en relación con todas las instancias que realmente pertenecen a esa clase (Ozturk Kiyak et al., 2023).

$$Sensibilidad = \frac{(TP)}{(TP+FN)} * 100$$

2.2.7.4. F1-Score

Es una métrica que integra la precisión y el recall del modelo en un solo valor, lo que facilita balancear la relevancia de ambas medidas.

(Ozturk Kiyak et al., 2023)

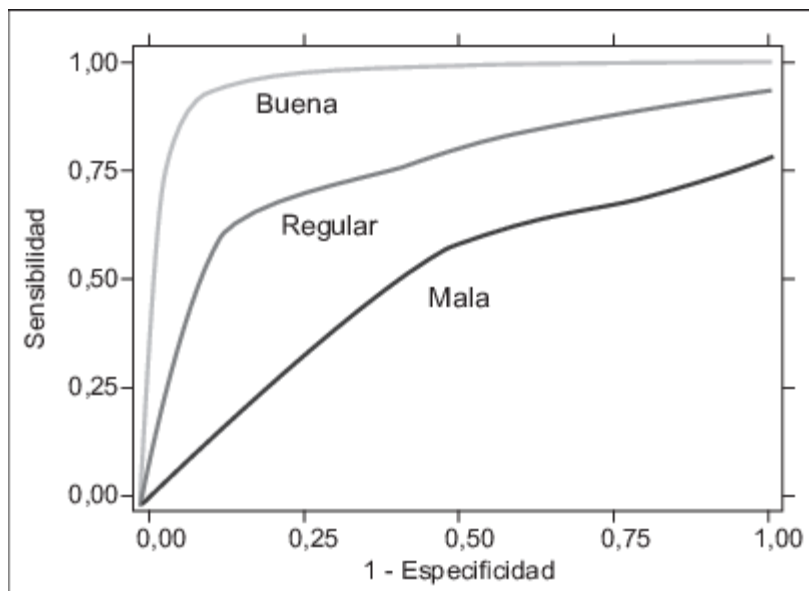
$$F1\ Score = \frac{2TP}{(2TP+FP+FN)}$$

2.2.7.5. Curva ROC

Muestra la relación entre la sensibilidad y (1 menos la especificidad). Un modelo ideal se ubica en la esquina superior izquierda (0,1), indicando 100% de sensibilidad y especificidad. Un modelo aleatorio sigue la diagonal, conocida como línea de no-discriminación. Si la curva sigue esta diagonal, el modelo no tiene capacidad predictiva; en cambio, si recorre los bordes izquierdo y superior, indica un modelo perfecto. Además, la curva ROC es útil para comparar diferentes modelos evaluando el área bajo la curva (Parra, 2019).

Figura 2

Curva ROC



Nota. (Burgos & Manterola, 2010)

2.2.7.6. Matthews correlation coefficient (MCC)

Matthews Correlation Coefficient o MCC es una métrica de que calcula la correlación entre las clases reales (c) y las etiquetas predichas (l), donde un valor

de -1 indica una predicción totalmente incorrecta, $+1$ representa una predicción perfecta y 0 sugiere una predicción aleatoria. El MCC se obtiene a partir de la covarianza entre las clases reales y las predichas, dividida entre el producto de sus desviaciones estándar (σ_c y σ_l) (Chicco et al. 2021)

$$MCC = \frac{Cov(c, l)}{\sigma_c \cdot \sigma_l} = \frac{TP \cdot TN - FP \cdot FN}{\sqrt{(TP + FP) \cdot (TP + FN) \cdot (TN + FP) \cdot (TN + FN)}}$$

2.2.7.7. Especificidad

Es una métrica que evalúa la capacidad de identificar correctamente las instancias de la clase negativa (o de fondo), y se expresa de la siguiente manera (Chicco & Jurman, 2020).

$$Especificidad = \frac{TN}{TN + FP}$$

2.5.6. Herramienta

2.5.6.1. Anaconda

Anaconda es una distribución de código abierto para Python y R, diseñada para la ciencia de datos, esto da un acceso y gestión fácil de bibliotecas y paquetes de Python necesarios para el análisis (Kovács et al., 2024)

2.5.6.1. SPSS 25

IBM SPSS Statistics es una herramienta informática empleada para realizar análisis estadísticos y para organizar, procesar y documentar información. (Čaplová & Šváblová, 2020).

2.5.6.1. Jupyter

Jupyter es un entorno informático interactivo basado en la web que admite numerosos lenguajes de programación de uso común.(Cai et al., 2025).

2.5.6.1. SQL Server 2019

SQL Server es un sistema utilizado para la gestión de bases de datos relacionales, creado por Microsoft, capaz de realizar actividades tales como almacenar, recuperar y administrar datos. Admite el lenguaje de consulta estructurado (SQL) para consultar bases de datos y se utiliza comúnmente en entornos empresariales para gestionar grandes volúmenes de datos transaccionales y analíticos (Landis, 2025).

2.5.7. Lenguaje de programación

2.5.7.1. Python

El lenguaje de programación Python, conocido por su naturaleza de código abierto, alto nivel y propósito general, ofrece una solución integral tanto para la entrada como para la manipulación y visualización de datos. Las versátiles bibliotecas de Python permiten garantizar la integridad de los datos, manipularlos de manera modular y reproducible, y crear visualizaciones personalizadas. (Kovács et al., 2024)

2.5.9. Definición de Accidentes Laborales

El accidente laboral es definida como todo suceso inesperado que ocurre como consecuencia de la actividad laboral y que provoca en el trabajador una discapacidad, alteración funcional, lesión física o la muerte (Díaz et al., 2020). Asimismo, se refiere a cualquier daño físico o alteración funcional que ocurra en el lugar de trabajo o como consecuencia de este. Estos accidentes pueden ser causados por eventos inesperados o repentinos, como una fuerza externa o el esfuerzo físico del trabajador, que ocurren de manera súbita y violenta (Gobierno del Perú, 2024).

2.5.9.1. Indicadores de Accidentes

2.5.9.1.1. Gravedad de accidentes

Este indicador mide cuántos días laborales se pierden a causa de accidentes por cada millón de horas trabajadas dentro de un período determinado. (Ministerio de Trabajo y Promoción del Empleo, 2024).

$$IGA = \frac{N^{\circ} \text{ de días perdidos}}{\text{Total horas-hombres trabajadas}} * 1000000$$

2.5.9.1.2. Frecuencia de accidentes

El indicador de frecuencia evalúa la cantidad de accidentes (tanto incapacitantes como fatales) en relación con el total de horas trabajadas por los empleados durante un periodo determinado (Ministerio de Trabajo y Promoción del Empleo, 2024).

$$IFA = \frac{N^{\circ} \text{ de accidentes}}{N^{\circ} \text{ total horas-hombres trabajadas}} * 1000000$$

III. Materiales y métodos

3.1. Materiales

Se llevó a cabo la identificación de los materiales para esta investigación. A continuación, se presenta un desglose detallado de estos aspectos.

Tabla 3

Materiales

Materiales
Hojas Bond A4
Cuaderno
Lapiceros
Plumones
Recarga de cartucho de impresión

Tabla 4

Detalle de materiales

N°	Descripción	Cantidad	Costo, S/	Total, S/
1	Hojas Bond A4	1 (paquete de 500 hojas)	S/15.70	15.70
2	Cuaderno	1	10.20	10.20
3	Lapiceros	4	2	8
4	Plumones	4	1.90	7.60
5	Recarga de cartucho de impresión	4	45	180
Total				221.50

Tabla 5

Software

Software	
SPSS 25	Anaconda
Microsoft Office 365	Python
SQL server 2019	Jupyter

Tabla 6**Detalle de software**

N°	Descripción	Cantidad	Costo, S/	Total, S/
01	SPSS 26	1	-	0,0
02	Microsoft Office 365	1	-	0,0
03	SQL server 2019	1	-	0,0
04	Anaconda	1	-	0,0
05	Python	1	-	0,0
06	Jupyter	1	-	0,0
Total				0,0

Tabla 7**Hardware**

Hardware	
Computadora: Laptop Asus TUF Gaming F16 Rtx4050 Intel Core 5 210h 8gb Ram 512gb Ssd F16 Fx607vu-rl037w 16" Fhd	Impresora: Impresora Multifuncional EcoTank L3250 WIFI

Tabla 8**Detalle hardware**

N°	Descripción	Cantidad	Costo S/	Total, S/
1	Computadora: Laptop Asus TUF Gaming F16 Rtx4050 Intel Core 5 210h 8gb Ram 512gb Ssd F16 Fx607vu-rl037w 16" Fhd	1	3100,0	3100,0
2	Impresora: Impresora Multifuncional EcoTank L3250 WIFI	1	650,0	650,0
Total				3750,0

Tabla 9**Otros**

Otros
Luz
Movilidad
Internet

Tabla 10**Detalle otros**

N°	Descripción	Número de meses	Costo, S/	Total, S/
01	Luz	10	110,0	1100,0
02	Internet	10	110,0	1100,0
03	Pasajes	10	30,0	300,0
Total				2500,0

3.2. Metodología**3.2.1. Tipo de investigación**

El enfoque de investigación fue cuantitativo ya que se fundamenta en el uso de la medición numérica. Las investigaciones bajo este enfoque emplean el recojo y análisis de datos con el fin de responder a preguntas de investigación y validar hipótesis previamente planteadas (Sánchez et al., 2018).

La investigación fue de tipo aplicado, puesto que su finalidad es dar solución a problemas concretos a través del uso de teorías existentes. La investigación aplicada se apoya en los resultados derivados de los objetivos de la investigación (Hadi et al., 2023).

Asimismo, se trató de un diseño no experimental, en el cual no se manipulan las variables ni se generan situaciones de manera intencional, sino que se observan hechos o fenómenos tal como ocurren en su entorno (Hernández et al., 2014).

Figura 3

Diseño no experimental

<p>M: representa la muestra seleccionada</p> <p>O: corresponde a la observación o medición de una variable</p> <p>T: indica el periodo de tiempo en el que se desarrolla la investigación</p> <p>R: expresa el coeficiente de correlación</p>

Fuente: (Ñaupas et al., 2018).

El subtipo de investigación es transversal correlacional-predictivo. Es transversal ya que se analizó un fenómeno en un solo momento determinado (Sánchez et al., 2018). El diseño correlacional permite establecer el grado de asociación entre una variable y otra, que no son dependientes entre sí (Ñaupas et al., 2018, p. 368). Por su parte, la investigación predictiva se orienta con el fin de predecir situaciones futuras (Ñaupas et al., 2018, p. 135).

La metodología estuvo basada en Knowledge Discovery in Databases (KDD), la cual se define como un proceso automatizado orientado al análisis y modelado de grandes volúmenes de datos. Este método comprende diversas fases que abarcan desde la comprensión del problema hasta la integración e interpretación de los resultados obtenidos (Safhi et al., 2019).

3.3. Población y muestra

La población se refiere al total de personas o elementos que forman parte de un estudio y comparten ciertas características en común (Arias & Covinos, 2021). Por lo que el estudio estuvo conformado por 110 datos de accidentes de trabajadores registrados en la empresa pesquera Corporación de Alimentos Marítimo S.A.C, **entre los meses de enero a diciembre del 2024**

La muestra se refiere al grupo de personas o casos elegidos de una población, utilizando algún tipo de muestreo, que puede ser al azar (probabilístico) o no aleatorio (no probabilístico), según el enfoque que se utilice (Sánchez et al.,

2018). En este caso, la muestra fue representado por la totalidad de la población, la cual el margen de error será cero por ciento.

El muestreo empleado fue no probabilístico se utiliza cuando el investigador selecciona la población basándose en características comunes o juicios personales, sin aplicar métodos estadísticos (Arias & Covinos, 2021).

Además, el muestreo fue intencional o por conveniencia ya que se basa en los criterios del investigador, como elegir estudiantes con buen rendimiento académico o incluir toda la población disponible, aplicando un juicio subjetivo para la selección (Arias & Covinos, 2021).

La unidad de análisis estuvo compuesta por un trabajador que labora en la empresa Corporación de Alimentos Marítimo S.A.C

3.4. Técnicas e instrumentos de recolección de datos

Como técnica se empleó el análisis de documentos y registros existentes, el cual consiste en examinar información contenida en documentos previamente elaborados, como es el caso de los registros históricos de accidentes laborales (Hadi et al., 2023).

Asimismo, se utilizaron fichas de registro (ver Anexos 3), las cuales serán aplicadas en el desarrollo de cada uno de los algoritmos de machine learning empleados en la investigación. Este instrumento permitió identificar la precisión de predicción de accidentes dentro de Corporación de Alimentos Marítimo S.A.C.

La validez hace referencia al nivel de precisión con el que un método o técnica logra medir aquello que realmente se propone evaluar. (Sánchez Carlessi et al., 2018). Para garantizar la validez del instrumento, se empleará el método del juicio de expertos. Se seleccionarán tres especialistas a quienes se les proporcionará una matriz de evaluación del cuestionario.

Un instrumento se considera confiable cuando las mediciones obtenidas son consistentes, es decir, no presentan variaciones significativas ni a lo largo del tiempo ni cuando son aplicadas a diferentes personas con el mismo nivel de instrucción (Ñaupas et al., 2018) . Asimismo, para determinar la confiabilidad, se

consideró las validaciones previamente realizadas a los instrumentos de medición utilizados en la investigación.

3.5. Procedimientos

En esta investigación se empleó el modelo basado en inteligencia artificial y oversampling para predecir los accidentes laborales en una empresa pesquera llamada Corporación de Alimentos Marítimo S.A.C., donde se toma la variable de accidentes laborales. Para esto, se revisaron a fondo distintos estudios internacionales con el objetivo de entender mejor el problema y buscar posibles soluciones. También se analizaron en detalle las dos variables del estudio, recopilando información de artículos y tesis que ayudaron a establecer antecedentes y bases teóricas, además de identificar las dimensiones necesarias para la investigación.

También se obtuvieron los datos de la población de estudio a partir de las fichas de registro de los empleados de la empresa. Una vez identificado el problema, se aplicó un estímulo (x), que consistió en el uso de un enfoque de machine learning, desarrollando la metodología Knowledge Discovery in Databases o KDD; esto permitió generar acciones concretas y aplicables. Para ello, se usó distintas técnicas de machine learning, como redes neuronales, redes bayesianas, máquinas de vectores de soporte (SVM), árboles de decisión y k vecinos más cercanos (KNN).

Con base en los resultados, se eligió el modelo que tuvo mejor desempeño para predecir accidentes. Luego, se implementó el enfoque utilizando el lenguaje de programación Python y la plataforma de desarrollo Anaconda. Para validar la hipótesis de la investigación, se utilizó la matriz de confusión a través de Python, con el fin de evaluar el rendimiento de los modelos y obtener un análisis preciso sobre la capacidad predictiva del sistema desarrollado.

3.6. Análisis de datos

Para el método de análisis de datos de esta investigación, se realizó la recopilación de los datos de las fichas de registro de los empleados de la empresa Pesquera Corporación de Alimentos Marítimo S.A.C., donde

posteriormente se realizó un proceso de depuración de los datos que son redundantes para proceder a la conversión de variables, transformando los valores numéricos en cualitativos. Este proceso se llevó a cabo utilizando tanto el software SPSS, Python y Anaconda. Asimismo, se efectuó un análisis de carácter predictivo haciendo uso de Python, que es un lenguaje de programación, donde se brindó información sobre las métricas. Respecto al análisis descriptivo, se utilizó el software SPSS para hacer el cálculo de la media a través de la prueba pre y post test.

3.7. Consideraciones éticas

El presente estudio se llevó a cabo siguiendo los lineamientos establecidos por la Universidad Nacional del Santa, garantizando el respeto por la integridad y autonomía de las personas. Se actuó con responsabilidad en cuanto a la relevancia, el alcance y los posibles impactos de la investigación. Además, se obtuvo el consentimiento informado, asegurando que las personas involucradas en la empresa en cuestión sean plenamente informadas sobre el objetivo y la naturaleza del estudio. El trabajo se sustenta en el principio de veracidad, garantizando que los datos presentados son auténticos y no han sido manipulados de ninguna manera. Finalmente, la estructura de las citas siguió los estándares de la séptima edición del formato APA y se respetaron los derechos de autor mediante las correspondientes referencias.

IV. Resultados y discusión

En esta sección se presentan los resultados obtenidos a partir de la implementación de los algoritmos de inteligencia artificial: árboles de decisión, KNN, Random Forest, SVM, redes bayesianas y gradiente para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. Asimismo, las figuras del 15 al 170 evidencian el procedimiento (véase figura 10) de ejecución experimental desarrollado para cada algoritmo, incluyendo las etapas de carga y preparación de datos, particionamiento del conjunto de datos, entrenamiento de los modelos y obtención de métricas de evaluación. Estas evidencias permiten garantizar la trazabilidad y reproducibilidad del proceso de modelamiento aplicado en la investigación.

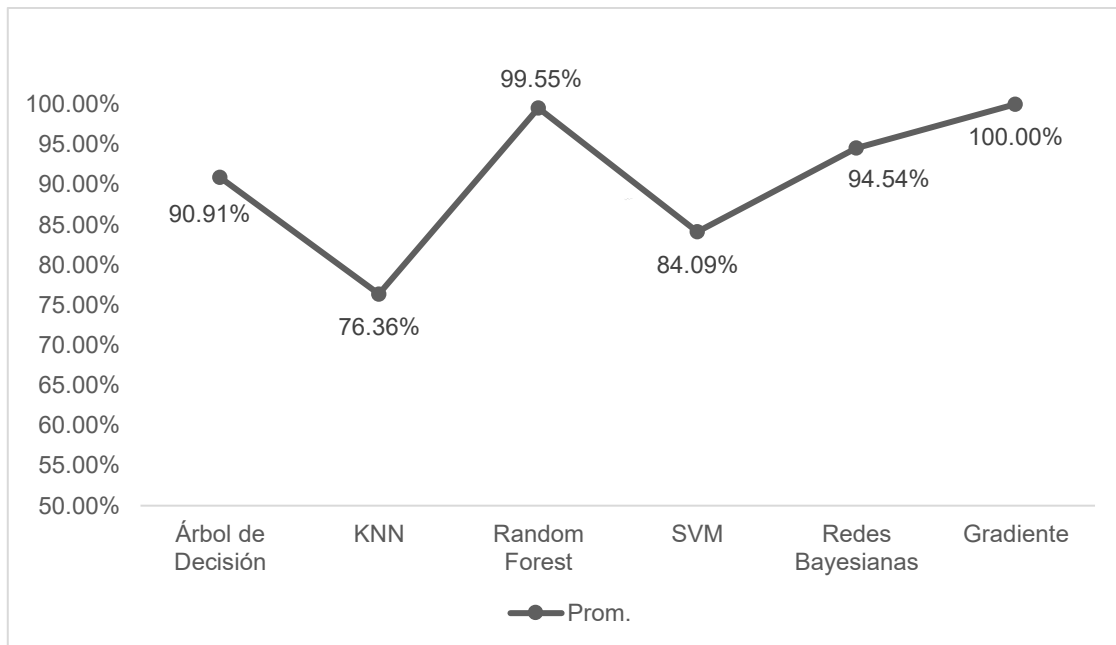
A continuación, presentaremos los resultados obtenidos por cada una de las hipótesis planteadas.

HE1: Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Accuracy los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

Tabla 11. Rendimiento de los modelos según la métrica Accuracy

CV	Árbol de Decisión	KNN	Random Forest	SVM	Redes Bayesianas	Gradiente
2	100.00%	72.73%	100.00%	77.27%	90.91%	100.00%
3	100.00%	72.73%	100.00%	90.91%	90.91%	100.00%
4	81.82%	77.27%	100.00%	90.91%	95.45%	100.00%
5	100.00%	77.27%	100.00%	90.91%	95.45%	100.00%
6	45.45%	77.27%	100.00%	90.91%	95.45%	100.00%
7	100.00%	77.27%	100.00%	90.91%	95.45%	100.00%
8	100.00%	77.27%	95.45%	77.27%	95.45%	100.00%
9	81.82%	77.27%	100.00%	77.27%	95.45%	100.00%
10	100.00%	77.27%	100.00%	77.27%	95.45%	100.00%
11	100.00%	77.27%	100.00%	77.27%	95.45%	100.00%
Prom.	90.91%	76.36%	99.55%	84.09%	94.54%	100.00%

Figura 4. Promedio de los modelos según la métrica Accuracy



Interpretación: Tal como se observa en la Tabla 11 y la Figura 4, la técnica con el porcentaje más óptimo según la métrica de accuracy para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. es el modelo Gradiente, con un valor promedio de 100.00%, seguido de Random Forest con 99.55%, y posteriormente Redes Bayesianas con 94.54%.

Comparando con estudios previos, el desempeño de Random Forest, nuestro segundo mejor modelo, es consistente con los hallazgos de Luo et al. (2023), quienes reportaron un accuracy de 82% en la predicción de la gravedad de accidentes en la construcción. De manera similar, Khairuddin et al. (2022) obtuvieron un accuracy de 0.949 con Random Forest para anticipar la gravedad de lesiones ocupacionales en diversas industrias, confirmando el alto porcentaje de accuracy de modelos basados en árboles.

Tabla 12. ANOVA de la métrica Accuracy

ANOVA					
	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Entre grupos	4285.394	5	857.079	13.775	0.000
Dentro de grupos	3359.978	54	62.222		
Total	7645.372	59			

Como se muestra en la Tabla 12, el nivel de significancia obtenido en la prueba ANOVA fue de 0.000, es decir, menor a 0.05. Por lo tanto, se procede a rechazar la hipótesis nula de igualdad y se llega a interpretar que al menos una de las medias difiere significativamente de las demás.

Tabla 13. Tukey de la métrica Accuracy

Comparaciones múltiples						
Variable dependiente:		valor				
HSD Tukey						
(I) algoritmo		Diferencia de medias (I-J)	Desv. Error	Sig.	Intervalo de confianza al 95%	
					Límite inferior	Límite superior
ARBOL DE DECISION	KNN	14,54700 ⁺	3.52766	0.002	4.1246	24.9694
	RANDOM FOREST	-8.63600	3.52766	0.158	-19.0584	1.7864
	SVM	6.81900	3.52766	0.394	-3.6034	17.2414
	REDES BAYESIANAS	-3.63300	3.52766	0.906	-14.0554	6.7894
KNN	GRADIENTE	-9.09100	3.52766	0.121	-19.5134	1.3314
	ARBOL DE DECISION	-14,54700 ⁺	3.52766	0.002	-24.9694	-4.1246
	RANDOM FOREST	-23,18300 ⁺	3.52766	0.000	-33.6054	-12.7606
	SVM	-7.72800	3.52766	0.259	-18.1504	2.6944
RANDOM FOREST	REDES BAYESIANAS	-18,18000 ⁺	3.52766	0.000	-28.6024	-7.7576
	GRADIENTE	-23,63800 ⁺	3.52766	0.000	-34.0604	-13.2156
	ARBOL DE DECISION	8.63600	3.52766	0.158	-1.7864	19.0584
	KNN	23,18300 ⁺	3.52766	0.000	12.7606	33.6054
SVM	SVM	15,45500 ⁺	3.52766	0.001	5.0326	25.8774
	REDES BAYESIANAS	5.00300	3.52766	0.716	-5.4194	15.4254
	GRADIENTE	-0.45500	3.52766	1.000	-10.8774	9.9674
	ARBOL DE DECISION	-6.81900	3.52766	0.394	-17.2414	3.6034
REDES BAYESIANAS	KNN	7.72800	3.52766	0.259	-2.6944	18.1504
	RANDOM FOREST	-15,45500 ⁺	3.52766	0.001	-25.8774	-5.0326
	REDES BAYESIANAS	-10,45200 ⁺	3.52766	0.049	-20.8744	-0.0296
	GRADIENTE	-15,91000 ⁺	3.52766	0.000	-26.3324	-5.4876
GRADIENTE	ARBOL DE DECISION	3.63300	3.52766	0.906	-6.7894	14.0554
	KNN	18,18000 ⁺	3.52766	0.000	7.7576	28.6024
	RANDOM FOREST	-5.00300	3.52766	0.716	-15.4254	5.4194
	SVM	10,45200 ⁺	3.52766	0.049	0.0296	20.8744
ARBOL DE DECISION	GRADIENTE	-5.45800	3.52766	0.636	-15.8804	4.9644
	ARBOL DE DECISION	9.09100	3.52766	0.121	-1.3314	19.5134
	KNN	23,63800 ⁺	3.52766	0.000	13.2156	34.0604
	RANDOM FOREST	0.45500	3.52766	1.000	-9.9674	10.8774
RANDOM FOREST	SVM	15,91000 ⁺	3.52766	0.000	5.4876	26.3324
	REDES BAYESIANAS	5.45800	3.52766	0.636	-4.9644	15.8804

Referente a la prueba Tukey que se observa en la Tabla 13 con un nivel de significancia de 0.05, se observa que Árbol de Decisión presenta Diferencia significativa con KNN (Sig. = 0.002), en cambio No es significativa con Random Forest (Sig. = 0.158), SVM (Sig. = 0.394), Redes Bayesianas (Sig. = 0.906) ni Gradiente (Sig. = 0.121).

Por otro lado, KNN muestra diferencias significativas con Árbol de Decisión (Sig. = 0.002), Random Forest (Sig. = 0.000), Redes Bayesianas (Sig. = 0.000) y Gradiente (Sig. = 0.000), mientras que no es significativa con SVM (Sig. = 0.259).

Asimismo, Random Forest presenta diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.001), pero no es significativa con Árbol de Decisión (Sig. = 0.158), Redes Bayesianas (Sig. = 0.716) ni Gradiente (Sig. = 1.000).

Igualmente, SVM muestra diferencias significativas con Random Forest (Sig. = 0.001), Redes Bayesianas (Sig. = 0.049) y Gradiente (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.394) ni KNN (Sig. = 0.259).

A su vez, Redes Bayesianas presenta diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.049), pero no es significativa con Árbol de Decisión (Sig. = 0.906), Random Forest (Sig. = 0.716) ni Gradiente (Sig. = 0.636).

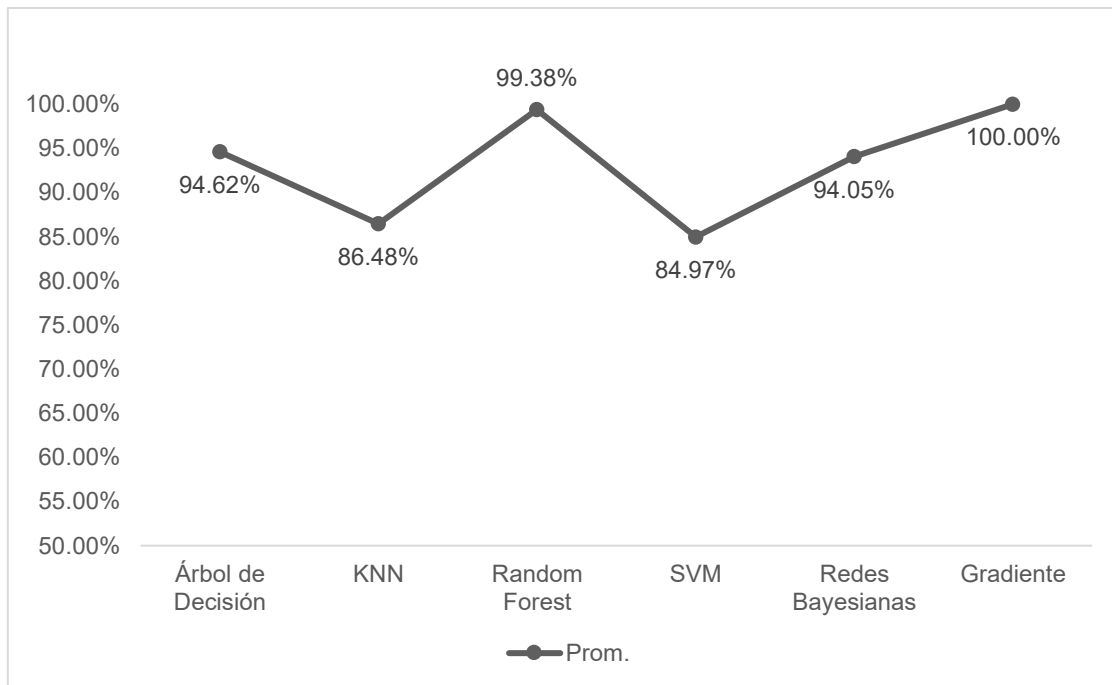
Cabe añadir que Gradiente muestra diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.121), Random Forest (Sig. = 1.000) ni Redes Bayesianas (Sig. = 0.636).

HE2: Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Precision los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

Tabla 14. Rendimiento de los modelos según la métrica Precision

CV	Árbol de Decisión	KNN	Random Forest	SVM	Redes Bayesianas	Gradiente
2	100.00%	85.71%	100.00%	76.19%	93.75%	100.00%
3	100.00%	85.71%	100.00%	93.75%	93.75%	100.00%
4	92.31%	86.67%	100.00%	93.75%	94.12%	100.00%
5	100.00%	86.67%	100.00%	93.75%	94.12%	100.00%
6	61.54%	86.67%	100.00%	93.75%	94.12%	100.00%
7	100.00%	86.67%	100.00%	93.75%	94.12%	100.00%
8	100.00%	86.67%	93.75%	76.19%	94.12%	100.00%
9	92.31%	86.67%	100.00%	76.19%	94.12%	100.00%
10	100.00%	86.67%	100.00%	76.19%	94.12%	100.00%
11	100.00%	86.67%	100.00%	76.19%	94.12%	100.00%
Prom.	94.62%	86.48%	99.38%	84.97%	94.05%	100.00%

Figura 5. Promedio de los modelos según la métrica Precision



Interpretación: Tal como se observa en la Tabla 14 y Figura 5, la técnica con el porcentaje más óptimo según la métrica de precisión para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. es el modelo Gradiente, con un valor promedio de 100.00%, seguido de Random Forest con 99.38%, y posteriormente Árbol de Decisión con 94.62%.

Estos resultados coinciden parcialmente con los hallazgos de Özkan & Ulaş (2024) realizaron un estudio con el objetivo de desarrollar un marco predictivo basado en aprendizaje automático (ML) utilizando datos de accidentes laborales para identificar las causas de amputación y muertes en el sector metalúrgico el modelo Random Forest (RF) obtuvo el mejor desempeño con una precision de 0.9618

Tabla 15. ANOVA de la métrica Precision

ANOVA					
	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Entre grupos	1999.960	5	399.992	10.211	0.000
Dentro de grupos	2115.303	54	39.172		
Total	4115.264	59			

Tal como se observa en la Tabla 15, el ANOVA muestra un valor de significancia de 0.000, es decir, menor a 0.05. Por lo tanto, se rechaza la hipótesis nula y se resalta que al menos uno de los modelos presenta un rendimiento significativamente diferente en términos de precisión.

Tabla 16. Tukey de la métrica Precision

Comparaciones múltiples						
Variable dependiente:		valor				
HSD Tukey						
(I) algoritmo		Diferencia de medias (I-J)	Desv. Error	Sig.	Intervalo de confianza al 95%	
					Límite inferior	Límite superior
ARBOL DE DECISION	KNN	8.13800	2.79901	0.056	-0.1316	16.4076
	RANDOM FOREST	-4.75900	2.79901	0.538	-13.0286	3.5106
	SVM	9.64600 [*]	2.79901	0.013	1.3764	17.9156
	REDES BAYESIANAS	0.57000	2.79901	1.000	-7.6996	8.8396
KNN	GRADIENTE	-5.38400	2.79901	0.399	-13.6536	2.8856
	ARBOL DE DECISION	-8.13800	2.79901	0.056	-16.4076	0.1316
	RANDOM FOREST	-12.89700 [*]	2.79901	0.000	-21.1666	-4.6274
	SVM	1.50800	2.79901	0.994	-6.7616	9.7776
RANDOM FOREST	REDES BAYESIANAS	-7.56800	2.79901	0.091	-15.8376	0.7016
	GRADIENTE	-13.52200 [*]	2.79901	0.000	-21.7916	-5.2524
	ARBOL DE DECISION	4.75900	2.79901	0.538	-3.5106	13.0286
	KNN	12.89700 [*]	2.79901	0.000	4.6274	21.1666
SVM	SVM	14.40500 [*]	2.79901	0.000	6.1354	22.6746
	REDES BAYESIANAS	5.32900	2.79901	0.411	-2.9406	13.5986
	GRADIENTE	-0.62500	2.79901	1.000	-8.8946	7.6446
	ARBOL DE DECISION	-9.64600 [*]	2.79901	0.013	-17.9156	-1.3764
REDES BAYESIANAS	KNN	-1.50800	2.79901	0.994	-9.7776	6.7616
	RANDOM FOREST	-14.40500 [*]	2.79901	0.000	-22.6746	-6.1354
	REDES BAYESIANAS	-9.07600 [*]	2.79901	0.024	-17.3456	-0.8064
	GRADIENTE	-15.03000 [*]	2.79901	0.000	-23.2996	-6.7604
GRADIENTE	ARBOL DE DECISION	-0.57000	2.79901	1.000	-8.8396	7.6996
	KNN	7.56800	2.79901	0.091	-0.7016	15.8376
	RANDOM FOREST	-5.32900	2.79901	0.411	-13.5986	2.9406
	SVM	9.07600 [*]	2.79901	0.024	0.8064	17.3456
REDES BAYESIANAS	GRADIENTE	-5.95400	2.79901	0.289	-14.2236	2.3156
	ARBOL DE DECISION	5.38400	2.79901	0.399	-2.8856	13.6536
	KNN	13.52200 [*]	2.79901	0.000	5.2524	21.7916
	RANDOM FOREST	0.62500	2.79901	1.000	-7.6446	8.8946
REDES BAYESIANAS	SVM	15.03000 [*]	2.79901	0.000	6.7604	23.2996
	REDES BAYESIANAS	5.95400	2.79901	0.289	-2.3156	14.2236

Según la prueba Tukey con un nivel de significancia de 0.05, se observa, en términos de la métrica de precisión que el algoritmo Árbol de Decisión presenta diferencia significativa con SVM (Sig. = 0.013), en cambio no es significativa con KNN (Sig. = 0.056), Random Forest (Sig. = 0.538), Redes Bayesianas (Sig. = 1.000) ni Gradiente (Sig. = 0.399).

Sumado a lo anterior, KNN muestra diferencias significativas con Random Forest (Sig. = 0.000) y Gradiente (Sig. = 0.000), pero no es significativa con Árbol de Decisión (Sig. = 0.056), SVM (Sig. = 0.994) ni Redes Bayesianas (Sig. = 0.091).

Adicionalmente, Random Forest presenta diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.538), Redes Bayesianas (Sig. = 0.411) ni Gradiente (Sig. = 1.000).

Además, SVM muestra diferencias significativas con Árbol de Decisión (Sig. = 0.013), Random Forest (Sig. = 0.000), Redes Bayesianas (Sig. = 0.024) y Gradiente (Sig. = 0.000), pero no es significativa con KNN (Sig. = 0.994).

Incluso Redes Bayesianas presenta diferencia significativa con SVM (Sig. = 0.024), en cambio no es significativa con Árbol de Decisión (Sig. = 1.000), KNN (Sig. = 0.091), Random Forest (Sig. = 0.411) ni Gradiente (Sig. = 0.289).

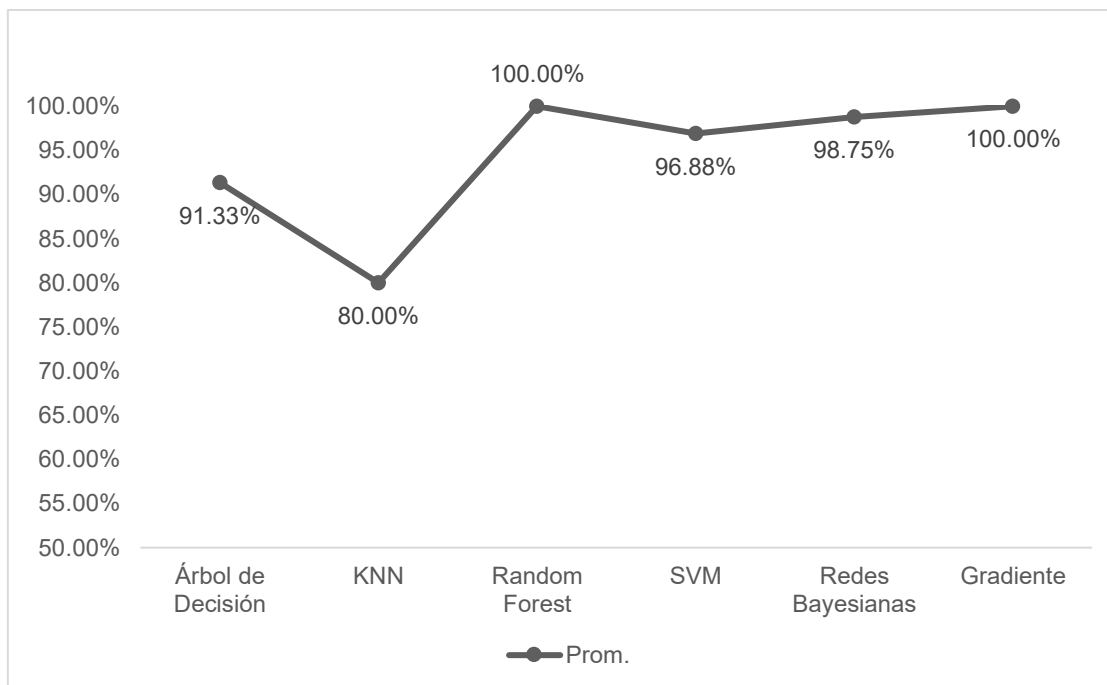
Del mismo modo, Gradiente muestra diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.399), Random Forest (Sig. = 1.000) ni Redes Bayesianas (Sig. = 0.289).

HE3: Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Recall (Sensibilidad) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

Tabla 17. Rendimiento de los modelos según la métrica Recall

CV	Árbol de Decisión	KNN	Random Forest	SVM	Redes Bayesianas	Gradiente
2	100.00%	75.00%	100.00%	100.00%	93.75%	100.00%
3	100.00%	75.00%	100.00%	93.75%	93.75%	100.00%
4	80.00%	81.25%	100.00%	93.75%	100.00%	100.00%
5	100.00%	81.25%	100.00%	93.75%	100.00%	100.00%
6	53.33%	81.25%	100.00%	93.75%	100.00%	100.00%
7	100.00%	81.25%	100.00%	93.75%	100.00%	100.00%
8	100.00%	81.25%	100.00%	100.00%	100.00%	100.00%
9	80.00%	81.25%	100.00%	100.00%	100.00%	100.00%
10	100.00%	81.25%	100.00%	100.00%	100.00%	100.00%
11	100.00%	81.25%	100.00%	100.00%	100.00%	100.00%
Prom.	91.33%	80.00%	100.00%	96.88%	98.75%	100.00%

Figura 6. Promedio de los modelos según la métrica Recall



Interpretación: Tal como se observa en la Tabla 17 y Figura 6, la técnica con el porcentaje más óptimo según la métrica de Recall o sensibilidad, para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. son los modelos Gradiente y Random Forest con un promedio de 100.00%, seguido de Redes Bayesianas con 98.75%. Posteriormente, se encuentran SVM con 96.88%, Árbol de Decisión con 91.33% y finalmente KNN con 80.00%.

Estos resultados son consistentes con el estudio de Özkan y Ulaş (2024) quienes desarrollaron un marco predictivo basado en aprendizaje automático utilizando datos de accidentes laborales del sector metalúrgico en Turquía, donde Random Forest alcanzó un recall de 0.9518, mostrando un alto desempeño para identificar eventos críticos. Asimismo, Tamascelli et al. (2022) destacan que los modelos de Deep Learning pueden aprovechar la información logrando un recall de 0.91.

Tabla 18. ANOVA de la métrica Recall

ANOVA					
valor	Suma de cuadrados	gl	Media cuadrática	F	Sig.

Entre grupos	3044.827	5	608.965	13.424	0.000
Dentro de grupos	2449.576	54	45.363		
Total	5494.403	59			

Tal como se observa en la Tabla 18, el ANOVA muestra un valor con un nivel de significancia de 0.000, siendo menor a 0.05. Por lo tanto, se rechaza la hipótesis nula, por ello, se puede mencionar que al menos uno de los modelos presenta un rendimiento significativamente diferente en términos de Recall.

Tabla 19. Tukey de la métrica Recall

Comparaciones múltiples						
Variable dependiente:		valor				
HSD Tukey						
(I) algoritmo		Diferencia de medias (I-J)	Desv. Error	Sig.	Intervalo de confianza al 95%	
					Límite inferior	Límite superior
ARBOL DE DECISION	KNN	11,33300*	3.01206	0.005	2.4339	20.2321
	RANDOM FOREST	-8.66700	3.01206	0.060	-17.5661	0.2321
	SVM	-5.54200	3.01206	0.450	-14.4411	3.3571
	REDES BAYESIANAS	-7.41700	3.01206	0.154	-16.3161	1.4821
KNN	GRADIENTE	-8.66700	3.01206	0.060	-17.5661	0.2321
	ARBOL DE DECISION	-11,33300*	3.01206	0.005	-20.2321	-2.4339
	RANDOM FOREST	-20,00000*	3.01206	0.000	-28.8991	-11.1009
	SVM	-16,87500*	3.01206	0.000	-25.7741	-7.9759
RANDOM FOREST	REDES BAYESIANAS	-18,75000*	3.01206	0.000	-27.6491	-9.8509
	GRADIENTE	-20,00000*	3.01206	0.000	-28.8991	-11.1009
	ARBOL DE DECISION	8.66700	3.01206	0.060	-0.2321	17.5661
	KNN	20,00000*	3.01206	0.000	11.1009	28.8991
SVM	SVM	3.12500	3.01206	0.903	-5.7741	12.0241
	REDES BAYESIANAS	1.25000	3.01206	0.998	-7.6491	10.1491
	GRADIENTE	0.00000	3.01206	1.000	-8.8991	8.8991
	ARBOL DE DECISION	5.54200	3.01206	0.450	-3.3571	14.4411
REDES BAYESIANAS	KNN	16,87500*	3.01206	0.000	7.9759	25.7741
	RANDOM FOREST	-3.12500	3.01206	0.903	-12.0241	5.7741
	REDES BAYESIANAS	-1.87500	3.01206	0.989	-10.7741	7.0241
	GRADIENTE	-3.12500	3.01206	0.903	-12.0241	5.7741
GRADIENTE	ARBOL DE DECISION	7.41700	3.01206	0.154	-1.4821	16.3161
	KNN	18,75000*	3.01206	0.000	9.8509	27.6491
	RANDOM FOREST	-1.25000	3.01206	0.998	-10.1491	7.6491
	SVM	1.87500	3.01206	0.989	-7.0241	10.7741
	GRADIENTE	-1.25000	3.01206	0.998	-10.1491	7.6491
	ARBOL DE DECISION	8.66700	3.01206	0.060	-0.2321	17.5661
	KNN	20,00000*	3.01206	0.000	11.1009	28.8991
	RANDOM FOREST	0.00000	3.01206	1.000	-8.8991	8.8991
	SVM	3.12500	3.01206	0.903	-5.7741	12.0241
	REDES BAYESIANAS	1.25000	3.01206	0.998	-7.6491	10.1491

Según la prueba Tukey, en la tabla 19, teniendo en cuenta nivel de significancia de 0.05, en términos de Recall, Árbol de Decisión presenta diferencia significativa con KNN (Sig. = 0.005), mientras que no es significativa con Random Forest (Sig.

= 0.060), SVM (Sig. = 0.450), Redes Bayesianas (Sig. = 0.154) ni Gradiente (Sig. = 0.060).

También, KNN muestra diferencias significativas con el resto de algoritmos, es decir, con todas las restantes, siendo Árbol de Decisión (Sig. = 0.005), Random Forest, SVM, Redes Bayesianas y Gradiente con (Sig. = 0.000).

Asimismo, se menciona que Random Forest presenta diferencia significativa únicamente con KNN (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.060), SVM (Sig. = 0.903), Redes Bayesianas (Sig. = 0.998) ni Gradiente (Sig. = 1.000).

De igual forma SVM muestra diferencia significativa únicamente con KNN (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.450), Random Forest (Sig. = 0.903), Redes Bayesianas (Sig. = 0.989) ni Gradiente (Sig. = 0.903).

Redes Bayesianas presenta diferencia significativa únicamente con KNN (Sig. = 0.000), y no es significativa con Árbol de Decisión (Sig. = 0.154), Random Forest (Sig. = 0.998), SVM (Sig. = 0.989) ni Gradiente (Sig. = 0.998).

Sumado a ello, Gradiente muestra diferencia significativa únicamente con KNN (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.060), Random Forest (Sig. = 1.000), SVM (Sig. = 0.903) ni Redes Bayesianas (Sig. = 0.998).

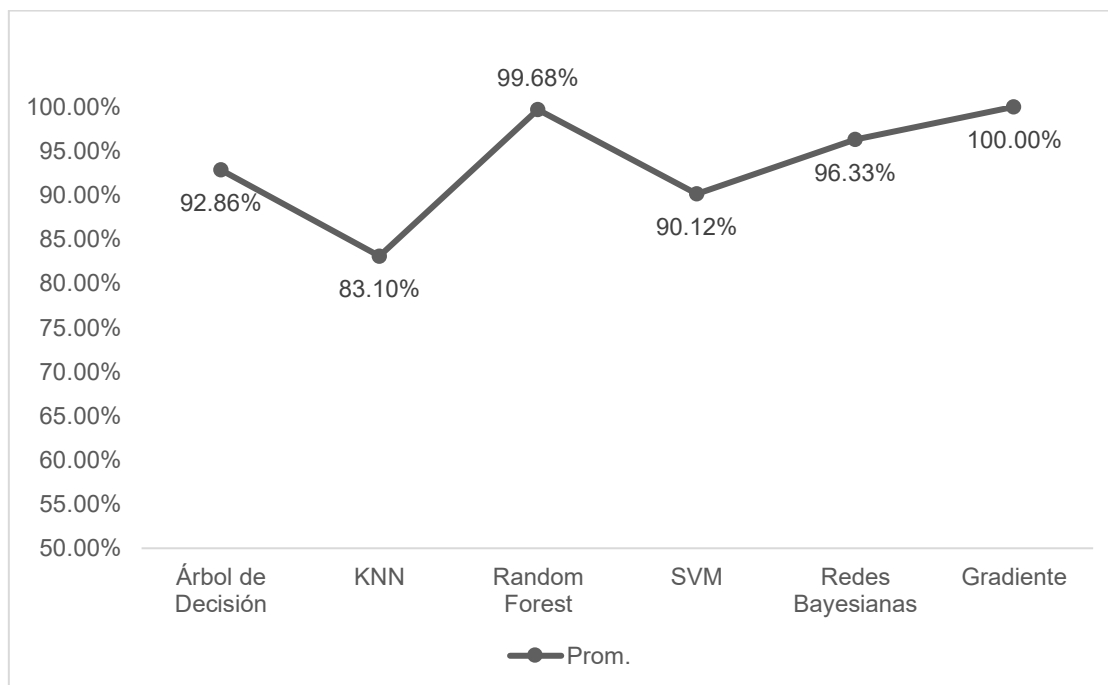
HE4: Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con F1 Score los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

Tabla 20. Rendimiento de los modelos según la métrica F1 Score

CV	Árbol de Decisión	KNN	Random Forest	SVM	Redes Bayesianas	Gradiente
2	100.00%	80.00%	100.00%	86.49%	93.75%	100.00%
3	100.00%	80.00%	100.00%	93.75%	93.75%	100.00%
4	85.71%	83.87%	100.00%	93.75%	96.97%	100.00%
5	100.00%	83.87%	100.00%	93.75%	96.97%	100.00%
6	57.14%	83.87%	100.00%	93.75%	96.97%	100.00%
7	100.00%	83.87%	100.00%	93.75%	96.97%	100.00%
8	100.00%	83.87%	96.77%	86.49%	96.97%	100.00%
9	85.71%	83.87%	100.00%	86.49%	96.97%	100.00%

10	100.00%	83.87%	100.00%	86.49%	96.97%	100.00%
11	100.00%	83.87%	100.00%	86.49%	96.97%	100.00%
Prom.	92.86%	83.10%	99.68%	90.12%	96.33%	100.00%

Figura 7. Promedio de los modelos según la métrica F1 Score



Interpretación: Como se observa en la Tabla 20 y Figura 7, la técnica con el porcentaje más óptimo según la métrica de F1 Score para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. es el modelo Gradiente, con un valor promedio de 100.00%, seguido de Random Forest con 99.68%, y posteriormente Redes Bayesianas con 96.33%. Posteriormente se encuentran Árbol de Decisión con 92.86%, SVM con 90.12% y KNN con 83.10%.

Estos resultados son consistentes con estudios previos, Khairuddin et al. (2022) también encontraron que Random Forest tuvo un alto desempeño, con un F1 Score de 0.909 para predecir la gravedad de lesiones ocupacionales en diferentes industrias, lo que coincide con nuestro segundo mejor modelo, mostrando que los métodos basados en árboles son muy efectivos para este tipo de predicciones. De manera complementaria, aunque los modelos sean diferentes, Zhou et al. (2025) reportaron un F1 Score de 97.79% usando Bi-

LSTM-CRF en un Grafo de Conocimiento de accidentes por caídas desde altura, lo que indica que es posible alcanzar altos niveles de F1 Score en la identificación de eventos críticos.

Tabla 21. ANOVA de la métrica F1 Score

ANOVA					
valor					
	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Entre grupos	2082.814	5	416.563	11.736	0.000
Dentro de grupos	1916.732	54	35.495		
Total	3999.545	59			

Tal como se observa en la Tabla 21, el ANOVA muestra un valor significancia de 0.000, es decir, menor a 0.05. Por lo tanto, se rechaza la hipótesis nula llegando a concluir que al menos los modelos presentan un rendimiento significativamente diferente en términos de F1 Score.

Tabla 22. Tukey de la métrica F1 Score

Comparaciones múltiples						
Variable dependiente: valor						
HSD Tukey						
(I) algoritmo		Diferencia de medias (I-J)	Desv. Error	Sig.	Intervalo de confianza al 95%	
					Límite inferior	Límite superior
ARBOL_DE_DECISION	KNN	9,76000*	2.66440	0.007	1.8881	17.6319
	RANDOM_FOREST	-6.82100	2.66440	0.125	-14.6929	1.0509
	SVM	2.73600	2.66440	0.907	-5.1359	10.6079
	REDES_BAYESIANAS	-3.47000	2.66440	0.782	-11.3419	4.4019
	GRADIENTE	-7.14400	2.66440	0.096	-15.0159	0.7279
KNN	ARBOL_DE_DECISION	-9,76000*	2.66440	0.007	-17.6319	-1.8881
	RANDOM_FOREST	-16,58100*	2.66440	0.000	-24.4529	-8.7091
	SVM	-7.02400	2.66440	0.106	-14.8959	0.8479
	REDES_BAYESIANAS	-13,23000*	2.66440	0.000	-21.1019	-5.3581
	GRADIENTE	-16,90400*	2.66440	0.000	-24.7759	-9.0321
RANDOM_FOREST	ARBOL_DE_DECISION	6.82100	2.66440	0.125	-1.0509	14.6929
	KNN	16,58100*	2.66440	0.000	8.7091	24.4529
	SVM	9,55700*	2.66440	0.009	1.6851	17.4289
	REDES_BAYESIANAS	3.35100	2.66440	0.806	-4.5209	11.2229
	GRADIENTE	-0.32300	2.66440	1.000	-8.1949	7.5489
SVM	ARBOL_DE_DECISION	-2.73600	2.66440	0.907	-10.6079	5.1359
	KNN	7.02400	2.66440	0.106	-0.8479	14.8959
	RANDOM_FOREST	-9,55700*	2.66440	0.009	-17.4289	-1.6851
	REDES_BAYESIANAS	-6.20600	2.66440	0.200	-14.0779	1.6659
	GRADIENTE	-9,88000*	2.66440	0.006	-17.7519	-2.0081
REDES_BAYESIANAS	ARBOL_DE_DECISION	3.47000	2.66440	0.782	-4.4019	11.3419

	KNN	13,23000'	2.66440	0.000	5.3581	21.1019
	RANDOM FOREST	-3.35100	2.66440	0.806	-11.2229	4.5209
	SVM	6.20600	2.66440	0.200	-1.6659	14.0779
	GRADIENTE	-3.67400	2.66440	0.739	-11.5459	4.1979
GRADIENTE	ARBOL DE DECISION	7.14400	2.66440	0.096	-0.7279	15.0159
	KNN	16,90400'	2.66440	0.000	9.0321	24.7759
	RANDOM FOREST	0.32300	2.66440	1.000	-7.5489	8.1949
	SVM	9,88000'	2.66440	0.006	2.0081	17.7519
	REDES_BAYESIANAS	3.67400	2.66440	0.739	-4.1979	11.5459

Según la prueba de Tukey, teniendo en cuenta la métrica F1 Score con un nivel de significancia de 0.05, el modelo Árbol de Decisión presenta diferencia significativa con KNN (Sig. = 0.007), mientras que no es significativa con Random Forest (Sig. = 0.125), SVM (Sig. = 0.907), Redes Bayesianas (Sig. = 0.782) ni Gradiente (Sig. = 0.096).

KNN muestra diferencias significativas con Árbol de Decisión (Sig. = 0.007), Random Forest (Sig. = 0.000), Redes Bayesianas y Gradiente (Sig. = 0.000), pero no es significativa con SVM (Sig. = 0.106).

Por otro lado, Random Forest presenta diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.009), mientras que no es significativa con Árbol de Decisión (Sig. = 0.125), Redes Bayesianas (Sig. = 0.806) ni Gradiente (Sig. = 1.000).

Se añade a SVM muestra diferencias significativas con Random Forest (Sig. = 0.009) y Gradiente (Sig. = 0.006), mientras que no es significativa con Árbol de Decisión (Sig. = 0.907), KNN (Sig. = 0.106) ni Redes Bayesianas (Sig. = 0.200).

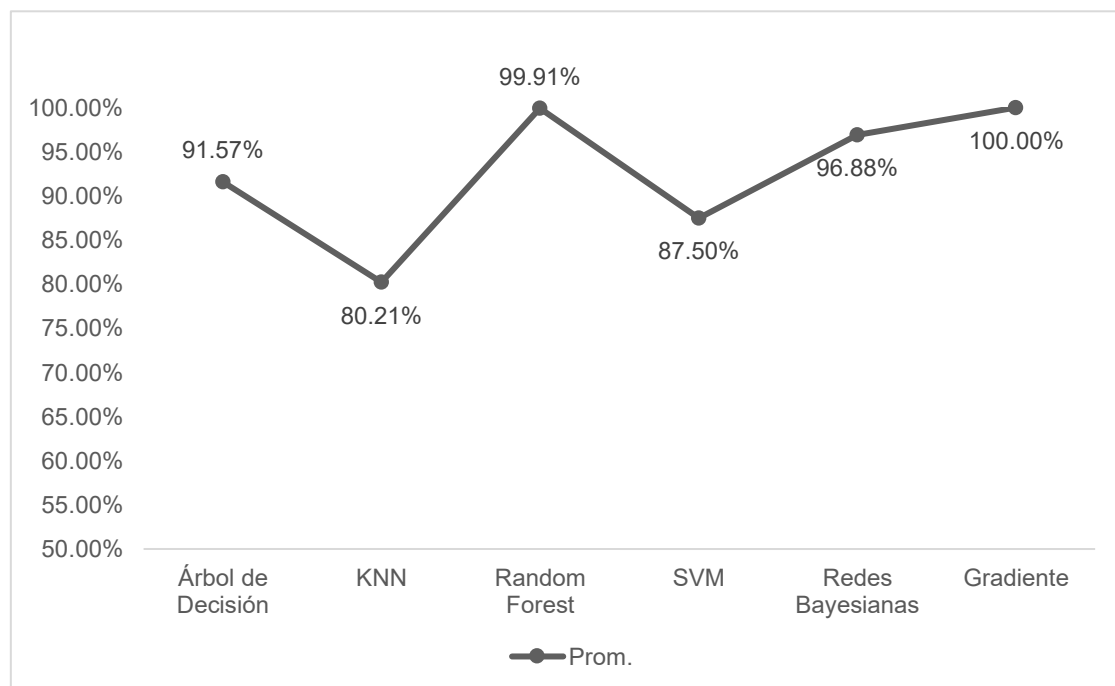
También, Redes Bayesianas presenta diferencia significativa únicamente con KNN (Sig. = 0.000), y no es significativa con Árbol de Decisión (Sig. = 0.782), Random Forest (Sig. = 0.806), SVM (Sig. = 0.200) ni Gradiente (Sig. = 0.739).

En paralelo, Gradiente muestra diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.006), mientras que no es significativa con Árbol de Decisión (Sig. = 0.096), Random Forest (Sig. = 1.000) ni Redes Bayesianas (Sig. = 0.739).

HE5: Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con ROC AUC los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

Tabla 23. Rendimiento de los modelos según la métrica ROC AUC

CV	Árbol de Decisión	KNN	Random Forest	SVM	Redes Bayesianas	Gradiente
2	100.00%	80.21%	100.00%	79.17%	96.88%	100.00%
3	100.00%	80.21%	100.00%	95.83%	96.88%	100.00%
4	85.71%	80.21%	100.00%	95.83%	96.88%	100.00%
5	100.00%	80.21%	100.00%	95.83%	96.88%	100.00%
6	48.57%	80.21%	100.00%	95.83%	96.88%	100.00%
7	100.00%	80.21%	100.00%	95.83%	96.88%	100.00%
8	100.00%	80.21%	99.05%	79.17%	96.88%	100.00%
9	81.43%	80.21%	100.00%	79.17%	96.88%	100.00%
10	100.00%	80.21%	100.00%	79.17%	96.88%	100.00%
11	100.00%	80.21%	100.00%	79.17%	96.88%	100.00%
Prom.	91.57%	80.21%	99.91%	87.50%	96.88%	100.00%

Figura 8. Promedio de los modelos según la métrica ROC AUC

Interpretación: Tal como se observa en la Tabla 23 y Figura 8, la técnica con el porcentaje óptimo según la métrica de ROC AUC para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. es el modelo Gradiente, con un valor promedio de 100.00%, seguido de Random Forest con 99.91%, y posteriormente Redes Bayesianas con 96.88%.

Posteriormente se encuentran Árbol de Decisión con 91.57%, SVM con 87.50% y KNN con 80.21%.

Estos resultados son similares a los encontrados por Khairuddin et al. (2022), haciéndose la comparación con nuestro segundo mejor modelo, Random Forest (99.91%). Los autores muestran un desempeño muy similar al de quienes reportaron que modelos como Random Forest, SVM y KNN lograron valores altos de AUC (0.95) para predecir la gravedad de lesiones ocupacionales. Esto confirma que los modelos basados en árboles suelen funcionar bien para este tipo de predicciones.

Tabla 24. ANOVA la métrica ROC AUC

ANOVA					
valor	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Entre grupos	3069.862	5	613.972	10.431	0.000
Dentro de grupos	3178.315	54	58.858		
Total	6248.176	59			

En la Tabla 24, el ANOVA muestra un nivel de significancia de 0.000. Por lo tanto, se rechaza la hipótesis nula, así que al menos uno de los modelos presenta un rendimiento significativamente diferente en términos de ROC AUC.

Tabla 25. Tukey de la métrica ROC AUC

Comparaciones múltiples						
Variable dependiente:	valor					
HSD Tukey						
(I) algoritmo		Diferencia de medias (I-J)	Desv. Error	Sig.	Intervalo de confianza al 95%	Límite superior
					Límite inferior	
ARBOL_DE_DECISION	KNN	11,36100*	3.43097	0.020	1.2243	21.4977
	RANDOM_FOREST	-8.33400	3.43097	0.165	-18.4707	1.8027
	SVM	4.07100	3.43097	0.841	-6.0657	14.2077
	REDES_BAYESIANAS	-5.30900	3.43097	0.636	-15.4457	4.8277
	GRADIENTE	-8.42900	3.43097	0.156	-18.5657	1.7077
KNN	ARBOL_DE_DECISION	-11,36100*	3.43097	0.020	-21.4977	-1.2243
	RANDOM_FOREST	-19,69500*	3.43097	0.000	-29.8317	-9.5583
	SVM	-7.29000	3.43097	0.290	-17.4267	2.8467
	REDES_BAYESIANAS	-16,67000*	3.43097	0.000	-26.8067	-6.5333
	GRADIENTE	-19,79000*	3.43097	0.000	-29.9267	-9.6533
RANDOM_FOREST	ARBOL_DE_DECISION	8.33400	3.43097	0.165	-1.8027	18.4707
	KNN	19,69500*	3.43097	0.000	9.5583	29.8317
	SVM	12,40500*	3.43097	0.008	2.2683	22.5417
	REDES_BAYESIANAS	3.02500	3.43097	0.949	-7.1117	13.1617
	GRADIENTE	-0.09500	3.43097	1.000	-10.2317	10.0417
SVM	ARBOL_DE_DECISION	-4.07100	3.43097	0.841	-14.2077	6.0657

REDES_BAYESIANAS	KNN	7.29000	3.43097	0.290	-2.8467	17.4267
	RANDOM_FOREST	-12.40500*	3.43097	0.008	-22.5417	-2.2683
	REDES_BAYESIANAS	-9.38000	3.43097	0.085	-19.5167	0.7567
	GRADIENTE	-12.50000*	3.43097	0.008	-22.6367	-2.3633
	ARBOL_DE_DECISION	5.30900	3.43097	0.636	-4.8277	15.4457
GRADIENTE	KNN	16.67000*	3.43097	0.000	6.5333	26.8067
	RANDOM_FOREST	-3.02500	3.43097	0.949	-13.1617	7.1117
	SVM	9.38000	3.43097	0.085	-0.7567	19.5167
	GRADIENTE	-3.12000	3.43097	0.942	-13.2567	7.0167
	ARBOL_DE_DECISION	8.42900	3.43097	0.156	-1.7077	18.5657
REDES_BAYESIANAS	KNN	19.79000*	3.43097	0.000	9.6533	29.9267
	RANDOM_FOREST	0.09500	3.43097	1.000	-10.0417	10.2317
	SVM	12.50000*	3.43097	0.008	2.3633	22.6367
	REDES_BAYESIANAS	3.12000	3.43097	0.942	-7.0167	13.2567
	GRADIENTE					

Según la prueba Tukey en la Tabla 25, con un nivel de significancia de 0.05, se observa que algunos modelos presentan diferencias significativas en términos de ROC AUC, con ello, Árbol de Decisión muestra diferencia significativa únicamente con KNN (Sig. = 0.020), mientras que no es significativa con Random Forest (Sig. = 0.165), SVM (Sig. = 0.841), Redes Bayesianas (Sig. = 0.636) ni Gradiente (Sig. = 0.156).

Además, KNN presenta diferencias significativas con Árbol de Decisión (Sig. = 0.020), Random Forest, Redes Bayesianas y Gradiente (Sig. = 0.000), pero no es significativa con SVM (Sig. = 0.290).

En esta misma línea, el algoritmo Random Forest tiene diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.008), mientras que no es significativo con Árbol de Decisión (Sig. = 0.165), Redes Bayesianas (Sig. = 0.949) ni Gradiente (Sig. = 1.000).

SVM muestra diferencias significativas con Random Forest y Gradiente (Sig. = 0.008), pero no es significativa con Árbol de Decisión (Sig. = 0.841), KNN (Sig. = 0.290) ni Redes Bayesianas (Sig. = 0.085).

Redes Bayesianas presenta diferencia significativa solamente con KNN (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.636), Random Forest (Sig. = 0.949), SVM (Sig. = 0.085) ni Gradiente (Sig. = 0.942).

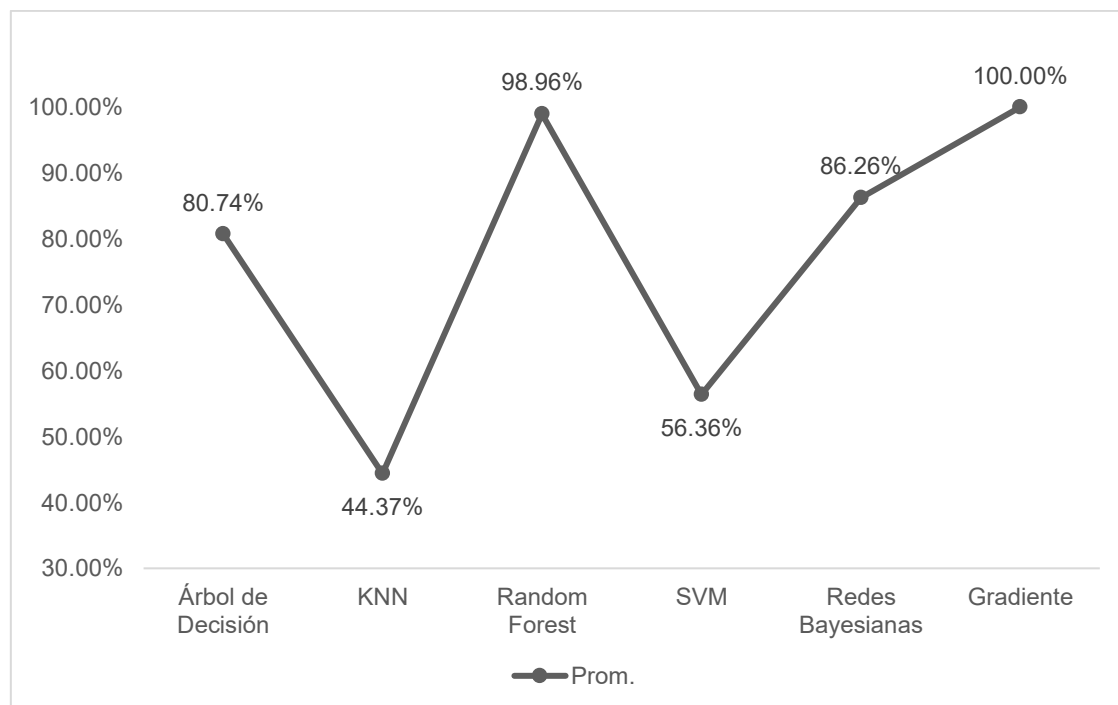
Asimismo, Gradiente tiene diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.008), mientras que no es significativa con Árbol de Decisión (Sig. = 0.156), Random Forest (Sig. = 1.000) ni Redes Bayesianas (Sig. = 0.942).

HE6: Un modelo basado en inteligencia artificial y Oversampling permite predecir con Matthews los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.

Tabla 26. Rendimiento de los modelos según la métrica MCC

CV	Árbol de Decisión	KNN	Random Forest	SVM	Redes Bayesianas	Gradiente
2	100.00%	38.58%	100.00%	35.63%	77.08%	100.00%
3	100.00%	38.58%	100.00%	77.08%	77.08%	100.00%
4	62.25%	45.82%	100.00%	77.08%	88.56%	100.00%
5	100.00%	45.82%	100.00%	77.08%	88.56%	100.00%
6	-17.14%	45.82%	100.00%	77.08%	88.56%	100.00%
7	100.00%	45.82%	100.00%	77.08%	88.56%	100.00%
8	100.00%	45.82%	89.64%	35.63%	88.56%	100.00%
9	62.25%	45.82%	100.00%	35.63%	88.56%	100.00%
10	100.00%	45.82%	100.00%	35.63%	88.56%	100.00%
11	100.00%	45.82%	100.00%	35.63%	88.56%	100.00%
Prom.	80.74%	44.37%	98.96%	56.36%	86.26%	100.00%

Figura 9. Promedio de los modelos según la métrica MCC



Interpretación: Tal como se observa en la Tabla 26 y Figura 9, la técnica con el porcentaje más óptimo según la métrica de Matthews para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. es el modelo Gradiente, con un valor promedio de 100.00%, seguido de Random Forest con 98.96%, y posteriormente Redes Bayesianas con 86.26%. Posteriormente se encuentran Árbol de Decisión con 80.74%, SVM con 56.36% y el porcentaje más bajo, con menos del 50% es KNN, contando con 44.37%.

Estos resultados son consistentes con el estudio de Özkan y Ulaş (2024), quienes aplicaron un Gradient Boosting (GBM) para predecir accidentes laborales en el sector metalúrgico en Turquía, alcanzando un accuracy de 0.9073 y un F1-score de 0.9513, mostrando que los modelos basados en Gradient Boosting obtienen un alto desempeño para identificar eventos críticos.

Tabla 27. ANOVA de la métrica MCC

ANOVA					
valor	Suma de cuadrados	gl	Media cuadrática	F	Sig.
Entre grupos	25983.315	5	5196.663	15.992	0.000
Dentro de grupos	17547.473	54	324.953		
Total	43530.788	59			

El ANOVA mostrado en la Tabla 27 muestra un valor de sig. 0.000. Por lo tanto, se rechaza la hipótesis nula (H_0) de igualdad de medias entre los modelos, concluyendo que al menos uno de los modelos presenta un rendimiento significativamente diferente en términos de Matthews Correlation Coefficient (MCC).

Tabla 28. Tukey de la métrica MCC

Comparaciones múltiples						
Variable dependiente: valor						
HSD Tukey						
(I) algoritmo		Diferencia de medias (I-J)	Desv. Error	Sig.	Intervalo de confianza al 95%	
					Límite inferior	Límite superior
ARBOL DE DECISION	KNN	36,36400*	8.06168	0.000	12.5459	60.1821
	RANDOM_FOREST	-18,22800	8.06168	0.228	-42.0461	5.5901
	SVM	24,38100*	8.06168	0.042	0.5629	48.1991
	REDES_BAYESIANAS	-5.52800	8.06168	0.983	-29.3461	18.2901
	GRADIENTE	-19.26400	8.06168	0.178	-43.0821	4.5541
KNN	ARBOL DE DECISION	-36,36400*	8.06168	0.000	-60.1821	-12.5459
	RANDOM_FOREST	-54,59200*	8.06168	0.000	-78.4101	-30.7739
	SVM	-11.98300	8.06168	0.674	-35.8011	11.8351
	REDES_BAYESIANAS	-41,89200*	8.06168	0.000	-65.7101	-18.0739
	GRADIENTE	-55,62800*	8.06168	0.000	-79.4461	-31.8099
RANDOM_FOREST	ARBOL DE DECISION	18.22800	8.06168	0.228	-5.5901	42.0461
	KNN	54,59200*	8.06168	0.000	30.7739	78.4101
	SVM	42,60900*	8.06168	0.000	18.7909	66.4271
	REDES_BAYESIANAS	12.70000	8.06168	0.618	-11.1181	36.5181
	GRADIENTE	-1.03600	8.06168	1.000	-24.8541	22.7821
SVM	ARBOL DE DECISION	-24,38100*	8.06168	0.042	-48.1991	-0.5629
	KNN	11.98300	8.06168	0.674	-11.8351	35.8011
	RANDOM_FOREST	-42,60900*	8.06168	0.000	-66.4271	-18.7909
	REDES_BAYESIANAS	-29,90900*	8.06168	0.006	-53.7271	-6.0909
	GRADIENTE	-43,64500*	8.06168	0.000	-67.4631	-19.8269
REDES_BAYESIANAS	ARBOL DE DECISION	5.52800	8.06168	0.983	-18.2901	29.3461
	KNN	41,89200*	8.06168	0.000	18.0739	65.7101
	RANDOM_FOREST	-12.70000	8.06168	0.618	-36.5181	11.1181
	SVM	29,90900*	8.06168	0.006	6.0909	53.7271
	GRADIENTE	-13.73600	8.06168	0.535	-37.5541	10.0821
GRADIENTE	ARBOL DE DECISION	19.26400	8.06168	0.178	-4.5541	43.0821
	KNN	55,62800*	8.06168	0.000	31.8099	79.4461
	RANDOM_FOREST	1.03600	8.06168	1.000	-22.7821	24.8541
	SVM	43,64500*	8.06168	0.000	19.8269	67.4631
	REDES_BAYESIANAS	13.73600	8.06168	0.535	-10.0821	37.5541

Tal como se observa en la Tabla 27, con un nivel de significancia de 0.05, se identifican diferencias significativas entre varios modelos en términos de Matthews Correlation Coefficient (MCC) para la predicción de accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. en la cual el Árbol de Decisión presenta diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.042), mientras que no hay diferencia significativa con Random Forest (Sig. = 0.228), Redes Bayesianas (Sig. = 0.983) ni Gradiente (Sig. = 0.178).

También el KNN muestra diferencias significativas con Árbol de Decisión, Random Forest, Redes Bayesianas y Gradiente, pero no significativa con SVM (Sig. = 0.674).

Por otro lado, Random Forest tiene diferencias significativas con KNN y SVM (Sig. = 0.000), mientras que no es significativa con Árbol de Decisión (Sig. = 0.228), Redes Bayesianas (Sig. = 0.618) ni Gradiente (Sig. = 1.000).

SVM presenta diferencias significativas con Árbol de Decisión (Sig. = 0.042), Random Forest (Sig. = 0.000), Redes Bayesianas (Sig. = 0.006) y Gradiente (Sig. = 0.000), pero no significativa con KNN (Sig. = 0.674).

Asimismo, Redes Bayesianas muestra diferencias significativas con KNN (Sig. = 0.000) y SVM (Sig. = 0.006), mientras que no hay diferencia significativa con Árbol de Decisión (Sig. = 0.983), Random Forest (Sig. = 0.618) ni Gradiente (Sig. = 0.535).

Finalmente, Gradiente presenta diferencias significativas con KNN y SVM (Sig. = 0.000), pero no significativa con Árbol de Decisión (Sig. = 0.178), Random Forest (Sig. = 1.000) ni Redes Bayesianas (Sig. = 0.535).

V. Conclusiones y recomendaciones

5.1. Conclusiones

1. Se determinó que la técnica que ofrece la mejor exactitud (Accuracy) para predecir los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. es el modelo de Gradiente. La prueba ANOVA indicó un nivel de significancia de 0.00, lo que evidencia que existen diferencias estadísticamente significativas entre los modelos evaluados. Asimismo, la prueba de comparaciones múltiples de Tukey HSD evidenció que el modelo Gradiente superó significativamente a KNN y SVM ($p < 0.05$); sin embargo, no se identificaron diferencias significativas respecto a árbol de decisión, Random Forest y redes bayesianas, lo que indica un comportamiento predictivo similar entre estos modelos.
2. En cuanto a la precisión (Precision), el modelo de Gradiente también mostró el mejor desempeño con un promedio de 100.00%. Asimismo, el resultado de la prueba ANOVA indicó que se presentan diferencias entre los modelos con significancia de $p = 0.00$. Esta métrica es relevante cuando se desea detectar la mayor cantidad de instancias positivas, especialmente en contextos donde los falsos negativos representan un impacto significativo. Asimismo, la prueba post hoc de Tukey HSD evidenció diferencias estadísticamente significativas del modelo de Gradiente frente a KNN y SVM ($p < 0.05$). Sin embargo, no se encontraron diferencias significativas respecto al árbol de decisión, random forest y regresión bayesiana ($p > 0.05$), indicando un desempeño comparable entre los modelos.
3. La métrica Recall (sensibilidad) indicó que los modelos Random Forest y Gradiente fueron los más efectivos, con valores de 100.00%. El ANOVA mostró diferencias significativas de 0.00. Esto significa que estos dos modelos identifican correctamente los accidentes, lo que es crucial para la prevención de riesgos. Asimismo, la prueba post hoc de Tukey HSD evidenció diferencias estadísticamente significativas del modelo gradiente

únicamente frente a knn ($p < 0.05$). En contraste, no se encontraron diferencias significativas respecto a árbol de decisiones, random forest , svm y redes bayesianas ($p > 0.05$) indicando un desempeño comparable entre dichos modelos en términos de recall

4. Para el F1-Score, el modelo de Gradiente obtuvo 100.00%, lo que refleja un equilibrio óptimo entre precisión y sensibilidad. Además, hubo un nivel de significancia en el ANOVA de 0.00. Se respalda que el modelo Gradiente es superior y confiable para la identificación de incidentes en la corporación pesquera. Asimismo, la prueba post hoc de Tukey HSD evidenció diferencias estadísticamente significativas del modelo gradiente frente a KNN y SVM ($p > 0.05$). Sin embargo, no se encontraron diferencias significativas respecto a árbol de decisión, random forest y redes bayesianas ($p > 0.05$), indicando un desempeño comparable entre estos modelos en términos de F1-score
5. Según la métrica ROC AUC, el modelo de Gradiente alcanzó 100.00%, evidenciando la capacidad que presenta el modelo para la predicción de accidentes. La prueba ANOVA demostró un nivel de significancia de 0.00, confirmando la superioridad estadística del modelo Gradiente sobre los demás. Asimismo, la prueba post hoc de Tukey evidenció diferencias estadísticamente significativas del modelo de gradiente frente a knn y svm ($p < 0.05$). No obstante, no se encontraron diferencias significativas respecto a árbol de decisiones, random forest y redes bayesianas ($p > 0.059$), indicando un desempeño comparable entre estos modelos en términos de ROC auc
6. Finalmente, con la métrica Matthews Correlation Coefficient (MCC), el modelo Gradiente fue el más sólido con un valor de 100.00%. Esto confirma que los modelos basados en Inteligencia Artificial y Oversampling son confiables para la predicción de accidentes laborales. La prueba ANOVA indicó significancia de 0.000, respaldando que las diferencias entre modelos son estadísticamente relevantes y que el modelo Gradiente es el más confiable para la prevención de riesgos en la pesquera. Asimismo, la prueba de Tukey HSD evidenció diferencias estadísticamente significativas del

modelo gradiente frente a knn y svm ($p < 0.05$). Sin embargo, no se encontraron diferencias significativas respecto a árbol de decisión, random forest y redes bayesianas ($p > 0.05$), indicando un desempeño comparable entre estos modelos en términos de la métrica MCC

5.2. Recomendaciones

1. Se recomienda a la corporación que pueda implementar los modelos de Gradiente y Random Forest como parte de un sistema de predicción de accidentes laborales, asimismo probar con diferentes enfoques de machine learning, priorizando la identificación temprana de riesgos y la prevención de incidentes en la empresa.
2. Se aconseja realizar el estudio con otra metodología como SEMMA y hacer comparaciones con la metodología actual, para determinar cuál es más eficiente en la predicción de accidentes laborales.
3. Se sugiere explorar enfoques combinados de Machine Learning, como Stacking o Cascading para mejorar aún más la precisión y sensibilidad de las predicciones.
4. Para futuros estudios, es recomendable ampliar la base de datos histórica de accidentes laborales actualizándola periódicamente, de modo que los modelos se entrenen con información reciente y se aumente la confiabilidad de las predicciones.

Referencias

- Arias Gonzáles, J., & Covinos Gallardo, M. (2021). *Diseño y metodología de la investigación* (1.^a ed.). ENFOQUES CONSULTING EIRL. https://gc.scalahed.com/recursos/files/r161r/w26022w/Arias_S2.pdf
- Breiman, L. (2001). Random forests. *Machine Learning*, 45(1), 5–32. <https://doi.org/10.1023/A:1010933404324/METRICS>
- Burgos, M., & Manterola, C. (2010). Cómo interpretar un artículo sobre pruebas diagnósticas. *ResearchGate*. <https://doi.org/10.4067/S0718-40262010000300018>
- Cai, Z., Davis, R. L., Mariétan, R., Tormey, R., & Dillenbourg, P. (2025). Jupyter Analytics: A Toolkit for Collecting, Analyzing, and Visualizing Distributed Student Activity in Jupyter Notebooks. *SIGCSE TS 2025 - Proceedings of the 56th ACM Technical Symposium on Computer Science Education*, 1, 172–178. <https://doi.org/10.1145/3641554.3701971;ISSUE:ISSUE:DOI>
- Čaplová, Z., & Švábová, P. (2020). IBM SPSS statistics. *Statistics and Probability in Forensic Anthropology*, 343–352. <https://doi.org/10.1016/B978-0-12-815764-0.00027-7>
- Chen, H., & Luo, X. (2016). Severity Prediction Models of Falling Risk for Workers at Height. *Procedia Engineering*, 164, 439-445. <https://doi.org/10.1016/j.proeng.2016.11.642>
- Chicco, D., & Jurman, G. (2020). The advantages of the Matthews correlation coefficient (MCC) over F1 score and accuracy in binary classification evaluation. *BMC Genomics*, 21(1), 6. <https://doi.org/10.1186/s12864-019-6413-7>
- Choi, J., Gu, B., Chin, S., & Lee, J.-S. (2020). Machine learning predictive model based on national data for fatal accidents of construction workers. *Automation in Construction*, 110, 102974. <https://doi.org/10.1016/j.autcon.2019.102974>
- Díaz Dumont, J. R., Suarez Mansilla, S. L., Santiago Martinez, R. N., & Bizarro Huaman, E. M. (2020). Accidentes laborales en el Perú: Análisis de la realidad a partir de datos estadísticos. *Revista Venezolana de Gerencia*, 25(89), 312-329.

Gil de Zúñiga, H., Goyanes ,Manuel, & and Durotoye, T. (2024). A Scholarly Definition of Artificial Intelligence (AI): Advancing AI as a Conceptual Framework in Communication Research. *Political Communication*, 41(2), 317-334. <https://doi.org/10.1080/10584609.2023.2290497>

Gobierno del Perú. (2024, enero 14). *¿Qué se considera un accidente de trabajo?* <https://www.gob.pe/12895-que-se-considera-un-accidente-de-trabajo>

Gondia, A., Moussa, A., Ezzeldin, M., & El-Dakhakhni, W. (2023). Machine learning-based construction site dynamic risk models. *Technological Forecasting and Social Change*, 189, 122347. <https://doi.org/10.1016/j.techfore.2023.122347>

Hernández Sampieri, R., Fernández Collado, C., & Baptista Lucio, M. del P. (2014). *Metodología de la investigación* (6ª ed.). McGraw-Hill Education. https://apiperiodico.jalisco.gob.mx/api/sites/periodicooficial.jalisco.gob.mx/files/metodologia_de_la_investigacion_-_roberto_hernandez_sampieri.pdf

Khairuddin, M. Z. F., Lu Hui, P., Hasikin, K., Abd Razak, N. A., Lai, K. W., Mohd Saudi, A. S., & Ibrahim, S. S. (2022). Occupational Injury Risk Mitigation: Machine Learning Approach and Feature Optimization for Smart Workplace Surveillance. *International Journal of Environmental Research and Public Health*, 19(21), 13962. <https://doi.org/10.3390/ijerph192113962>

Khasanov, D., Tojiyev, M., & Primkulov, O. (2021). Gradient descent in machine learning. *International Conference on Information Science and Communications Technologies: Applications, Trends and Opportunities, ICISCT 2021*. <https://doi.org/10.1109/ICISCT52966.2021.9670169>

Kovács, I., Székely, T., Pogány, P., Takács, S., Erős, M., & Járay, B. (2024). Utilizing the open-source programming language Python to create interactive Quality Assurance dashboards for diagnostic and screening performance in Cytology. *Journal of the American Society of Cytopathology*, 13(4), 309-318. <https://doi.org/10.1016/j.jasc.2024.03.007>

- Landis, H. (2025, October 10). *What Is SQL Server? Components, Editions and Use Cases*. <https://learn.g2.com/what-is-sql-server>
- Luo, X., Li, X., Goh, Y. M., Song, X., & Liu, Q. (2023). Application of machine learning technology for occupational accident severity prediction in the case of construction collapse accidents. *Safety Science*, 163, 106138. <https://doi.org/10.1016/j.ssci.2023.106138>
- Maheronnaghsh, S., Zolfagharnasab, H., Gorgich, M., & Duarte, J. (2023). Machine learning in Occupational Safety and Health - a systematic review: Review. *International Journal of Occupational and Environmental Safety*, 7(1), Article 1. https://doi.org/10.24840/2184-0954_007-001_001586
- Mahesh, B. (2020). Machine Learning Algorithms—A Review. *International Journal of Science and Research (IJSR)*, 9(1), 381-386. <https://doi.org/10.21275/ART20203995>
- Ministerio de Trabajo y Promoción del Empleo. (2024). *Notificaciones de accidentes de trabajo, incidentes peligrosos y enfermedades ocupacionales—Diciembre 2024* [Boletín Estadístico Mensual de Notificaciones de Accidentes de Trabajo, Incidentes Peligrosos y Enfermedades Ocupacionales]. Ministerio de Trabajo y Promoción del Empleo del Perú. <https://cdn.www.gob.pe/uploads/document/file/7632348/6471371-sat-diciembre-2024.pdf?v=1739543059>
- Mosquera, R., Parra, L., Ledesma, A. J., Bonilla, H. F., Mosquera, R., Parra, L., Ledesma, A. J., & Bonilla, H. F. (2021). Predicción de la accidentalidad laboral en la industria de pulpa y papel usando algoritmos de clasificación. *Información tecnológica*, 32(1), 133-142. <https://doi.org/10.4067/S0718-07642021000100133>
- Muñoz Muñoz, E. G., Verduga Alcívar, D. A., Yandri Francinet, G. A., Lapo Palacios, M. A., & Zorrilla Briones, O. (2024). Búsqueda de Patrones con Machine Learning en Datos de Siniestros de Tránsito. *Ciencia Latina: Revista Multidisciplinar*, 8(2), 1617-1638.

- Nazarihighipashaki, M., Moen, B. E., & Bråtveit, M. (2024). Fatal occupational injuries in fishing, farming and forestry 2010–2015. *Occupational Medicine*, 74(7), 523-529. <https://doi.org/10.1093/occmed/kqae073>
- Ñaupas Paitán, H., Valdivia Dueñas, M. R., Palacios Vilela, J. J., & Romero Delgado, H. E. (2018). *Metodología de la investigación cuantitativa-cualitativa y redacción de la tesis* (5.ª ed.). Ediciones de la U.
- Organización Internacional del Trabajo, [OIT]. (2023, noviembre 26). *Casi 3 millones de personas mueren por accidentes y enfermedades relacionadas con el trabajo*. <https://www.ilo.org/es/resource/news/casi-3-millones-de-personas-mueren-por-accidentes-y-enfermedades>
- Özkan, E. K., & Ulaş, H. B. (2024). Comparison of four machine learning methods for occupational accidents based on national data on metal sector in Turkey. *Safety Science*, 174, 106468. <https://doi.org/10.1016/j.ssci.2024.106468>
- Ozturk Kiyak, E., Ghasemkhani, B., & Birant, D. (2023). High-Level K-Nearest Neighbors (HLKNN): A Supervised Machine Learning Model for Classification Analysis. *Electronics*, 12(18), Article 18. <https://doi.org/10.3390/electronics12183828>
- Parra, F. (2019). *Estadística y Machine Learning con R*. <https://bookdown.org/content/2274/metodos-de-clasificacion.html>
- Pinedo Ramirez, J. A., & Grados Acosta, F. J. (2023). Modelo predictivo de planificación operativa bajo el enfoque de machine learning para gestionar los tiempos de ciclo del volquete en una mina subterránea. *Universidad Peruana de Ciencias Aplicadas (UPC)*. <https://repositorioacademico.upc.edu.pe/handle/10757/670723>
- Qi, H., Zhou, Z., Manu, P., & Li, N. (2025). Falling risk analysis at workplaces through an accident data-driven approach based upon hybrid artificial intelligence (AI) techniques. *Safety Science*, 185, 106814. <https://doi.org/10.1016/j.ssci.2025.106814>

- Rahman, M., Hossain, A., & Sikder, A. (2025). *Machine Learning Applications in Industry Safety: Analysis and Prediction of Industrial Accidents*. <http://dx.doi.org/10.2139/ssrn.5068169>
- Rogalewicz, M., & Sika, R. (2016). Methodologies of Knowledge Discovery from Data and Data Mining Methods in Mechanical Engineering. *Management and Production Engineering Review*, 7(4), 97-108. <https://doi.org/10.1515/mper-2016-0040>
- Romero, M., Guédria, W., Panetto, H., & Barafort, B. (2020). Towards a Characterisation of Smart Systems: A Systematic Literature Review. *Computers in Industry*, 120, 103224. <https://doi.org/10.1016/j.compind.2020.103224>
- Romero, C., & Ventura, S. (2024). *Educational data mining and learning analytics: An updated survey*. **WIREs Data Mining and Knowledge Discovery**. <https://doi.org/10.48550/arXiv.2402.07956>
- Ruiz Huaraz, C. B., & Valenzuela Ramos, M. R. (2022). Metodología de la investigación. En *Fondo Editorial UNAT*. Fondo Editorial UNAT. <https://doi.org/10.56224/EdiUnat.4>
- Safhi, H. M., Frikh, B., & Ouhbi, B. (2019). Assessing reliability of Big Data Knowledge Discovery process. *Procedia Computer Science*, 148, 30-36. <https://doi.org/10.1016/j.procs.2019.01.005>
- Sánchez Carlessi, H., Reyes Romero, C., & Mejía Sáenz, K. (2018). *Manual de términos en investigación científica, tecnológica y humanística* (1.^a ed.). Universidad Ricardo Palma. <https://repositorio.urp.edu.pe/handle/20.500.14138/1480>
- Sarkar, S., & Maiti, J. (2020). Machine learning in occupational accident analysis: A review using science mapping approach with citation network analysis. *Safety Science*, 131, 104900. <https://doi.org/10.1016/j.ssci.2020.104900>
- Sarkar, S., Vinay, S., Raj, R., Maiti, J., & Mitra, P. (2019). Application of optimized machine learning techniques for prediction of occupational accidents. *Computers & Operations Research*, 106, 210-224. <https://doi.org/10.1016/j.cor.2018.02.021>

- Tamascelli, N., Solini, R., Paltrinieri, N., & Cozzani, V. (2022). Learning from major accidents: A machine learning approach. *Computers & Chemical Engineering*, 162, 107786. <https://doi.org/10.1016/j.compchemeng.2022.107786>
- Tixier, A. J.-P., Hallowell, M. R., Rajagopalan, B., & Bowman, D. (2016). Application of machine learning to construction injury prediction. *Automation in Construction*, 69, 102-114. <https://doi.org/10.1016/j.autcon.2016.05.016>
- Vallejo Noguera, F. F., Rubio Endara, O. W., & Tello Moreira, J. A. (2022). Implementar el Uso de la Inteligencia Artificial para Detectar el Comportamiento del Trabajador en la Prevención de Accidentes Laborales en la Empresa. *Dominio de las Ciencias*, 8(1), 21.
- Vivian, G. A., Bauder, R. A., & Khoshgoftaar, T. M. (2025). A comprehensive survey on machine learning for workplace injury analysis: Risk prediction, return to work strategies, and demographic insights. *Journal of Big Data*, 12(1), 167. <https://doi.org/10.1186/s40537-025-01229-z>
- Yedla, A., Kakhki, F. D., & Jannesari, A. (2020). Predictive Modeling for Occupational Safety Outcomes and Days Away from Work Analysis in Mining Operations. *International Journal of Environmental Research and Public Health*, 17(19), Article 19. <https://doi.org/10.3390/ijerph17197054>
- Zhou, Z., Yu, X., Magoua, J. J., Cui, J., Luan, H., & Lin, D. (2025). Integrating machine learning and a large language model to construct a domain knowledge graph for reducing the risk of fall-from-height accidents. *Accident Analysis & Prevention*, 215, 108009. <https://doi.org/10.1016/j.aap.2025.108009>

ANEXO

Anexo 1: Matriz De consistencia

Tabla 29

Matriz de consistencia

Título	PROBLEMA	OBJETIVOS	HIPÓTESIS	VARIABLE	METODOLOGÍA
	General	General	General	Independiente	
	- ¿En qué medida se predicen los accidentes laborales mediante un modelo basado en Inteligencia Artificial y Oversampling en la Corporación de Alimentos Marítimo S.A.C.?	- Predecir los accidentes laborales mediante un modelo basado en Inteligencia Artificial y Oversampling en la Corporación de Alimentos Marítimo S.A.C.	- Se predicen los accidentes laborales mediante un modelo basado en Inteligencia Artificial y Oversampling en la Corporación de Alimentos Marítimo S.A.C.	Modelo Basado en Inteligencia Artificial y Oversampling Dimensiones: Modelos	- Enfoque: Cuantitativo - Tipo: Aplicada
Predicción de Accidentes Laborales mediante un Modelo de Inteligencia Artificial y Oversampling en la Corporación de Alimentos Marítimo S.A.C.	- Específicos	- Específicos	- Específicos	Dependiente	- Diseño: No experimental
	- ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con Accuracy los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.? - ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con Precision los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.? - ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con Recall (Sensibilidad) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.? - ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con F1 Score los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.? - ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con ROC AUC los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.? - ¿En qué medida un modelo basado en Inteligencia Artificial y Oversampling permitirá predecir con Matthews Correlation Coefficient (MCC) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.?	- Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Accuracy los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Precision los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Recall (Sensibilidad) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con F1 Score los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con ROC AUC los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Determinar en qué porcentaje un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Matthews Correlation Coefficient (MCC) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.	- Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Accuracy los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Precision los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Recall (Sensibilidad) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con F1 Score los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con ROC AUC los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C. - Un modelo basado en Inteligencia Artificial y Oversampling permite predecir con Matthews Correlation Coefficient (MCC) los accidentes laborales en la Corporación de Alimentos Marítimo S.A.C.	Predicción de Accidentes Laborales Dimensiones: Métricas de evaluación	- Subtipo: Transversal correlacional-predictivo - Metodología: Basada en KDD (Knowledge Discovery in Databases) - Población: 110 datos de accidentes de trabajadores registrados en la empresa pesquera Corporación De Alimentos Marítimo S.A.C. - Técnica: Análisis de documentos y registros existentes - Instrumento: Fichas de registro

Anexo 2: Operacionalización de variables

Tabla 30

Matriz de operacionalización de variable

Variable	Definición conceptual	Definición Operacional	Dimensiones	Indicadores	Escala de medición
Independiente: Modelo Basado en Inteligencia Artificial y Oversampling	Los sistemas inteligentes pueden considerarse como sistemas capaces de actualizar su conocimiento interno utilizado para razonar y tomar decisiones óptimas. Para ello, los sistemas inteligentes deben tener la capacidad de percibir continuamente su entorno, controlarlo y permitir la comunicación entre sus elementos (Romero et al., 2020).	Aplicación de un modelo de IA con técnicas de oversampling para generar predicciones de accidentes laborales.	Modelos	Árbol de Decisión KNN Random Forest SVM Redes Bayesianas Gradiente	
Dependiente: Predicción de Accidentes Laborales	Un accidente de trabajo se refiere a cualquier daño físico o alteración funcional que ocurra en el lugar de trabajo o como consecuencia de este. Estos accidentes pueden ser causados por eventos inesperados o repentinos, como una fuerza externa o el esfuerzo físico del trabajador, que ocurren de manera súbita y violenta (Gobierno del Perú, 2024).	Para medir la predicción de accidentes laborales se dan usos de las métricas de evaluación, la cual será obtenida a través de un modelo basado en inteligencia artificial y oversampling	Métricas de evaluación	<div>Accuracy = $\frac{((TP+TN)/(TP+TN+FP+FN)) * 100}{Precision= (TP/(TP+FP)) * 100}$<div>Sensibilidad = $(TP/(TP+FN)) * 100$</div><div>F1-Score = $2 * (Recall * Precisión) / (Recall + Precisión)$</div><div>ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)</div><div>Matthews correlation coefficient (MCC) = $(TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$</div></div>	Razón

ANEXO N° 3

Instrumento de observación para el modelo basado en inteligencia artificial y oversampling

A) Árbol de decisión

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})) * 100$	
2	Precision	Razón	$\text{Precision} = (\text{TP} / (\text{TP} + \text{FP})) * 100$	
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (\text{TP} / (\text{TP} + \text{FN})) * 100$	
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (\text{TP} * \text{TN} - \text{FP} * \text{FN}) / \sqrt{((\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN}))}$	

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

B) KNN

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

C) Random Forest

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

D) SVM

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

E) Redes Bayesianas

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	
6	Matthews correlation coefficient (MCC)	Razón	$Matthews\ correlation\ coefficient\ (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

F) Gradiente

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	
6	Matthews correlation coefficient (MCC)	Razón	$Matthews\ correlation\ coefficient\ (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

ANEXO N°4

Figura 10

Cronograma de actividades para ejecución del proyecto de investigación

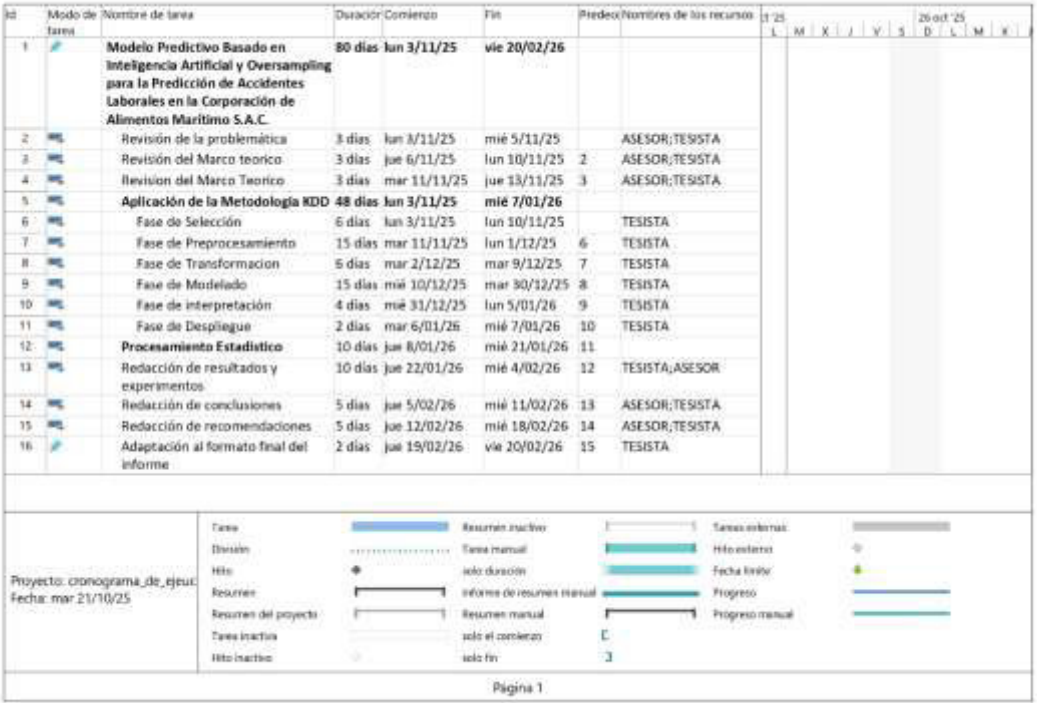
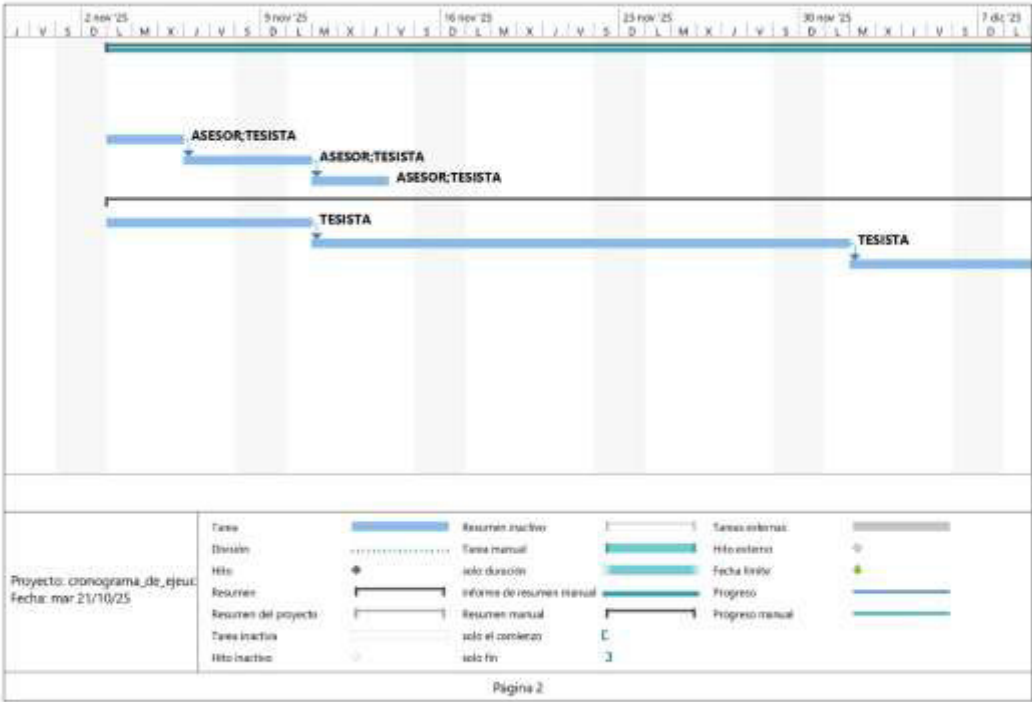
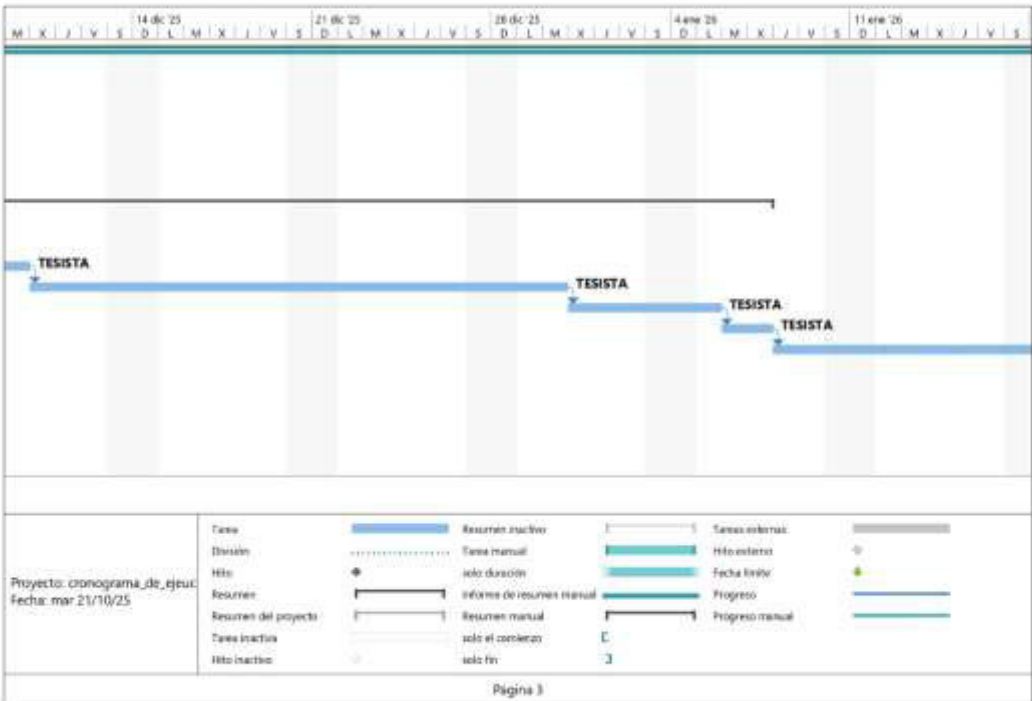
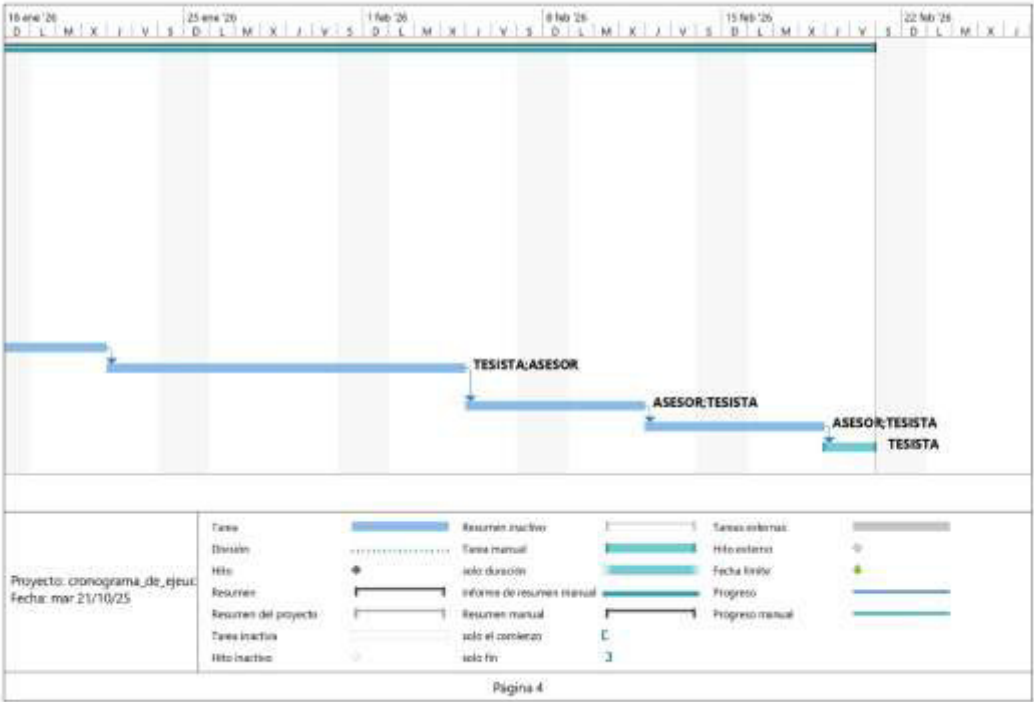


Figura 11

Cronograma de actividades lineal para ejecución del proyecto de investigación

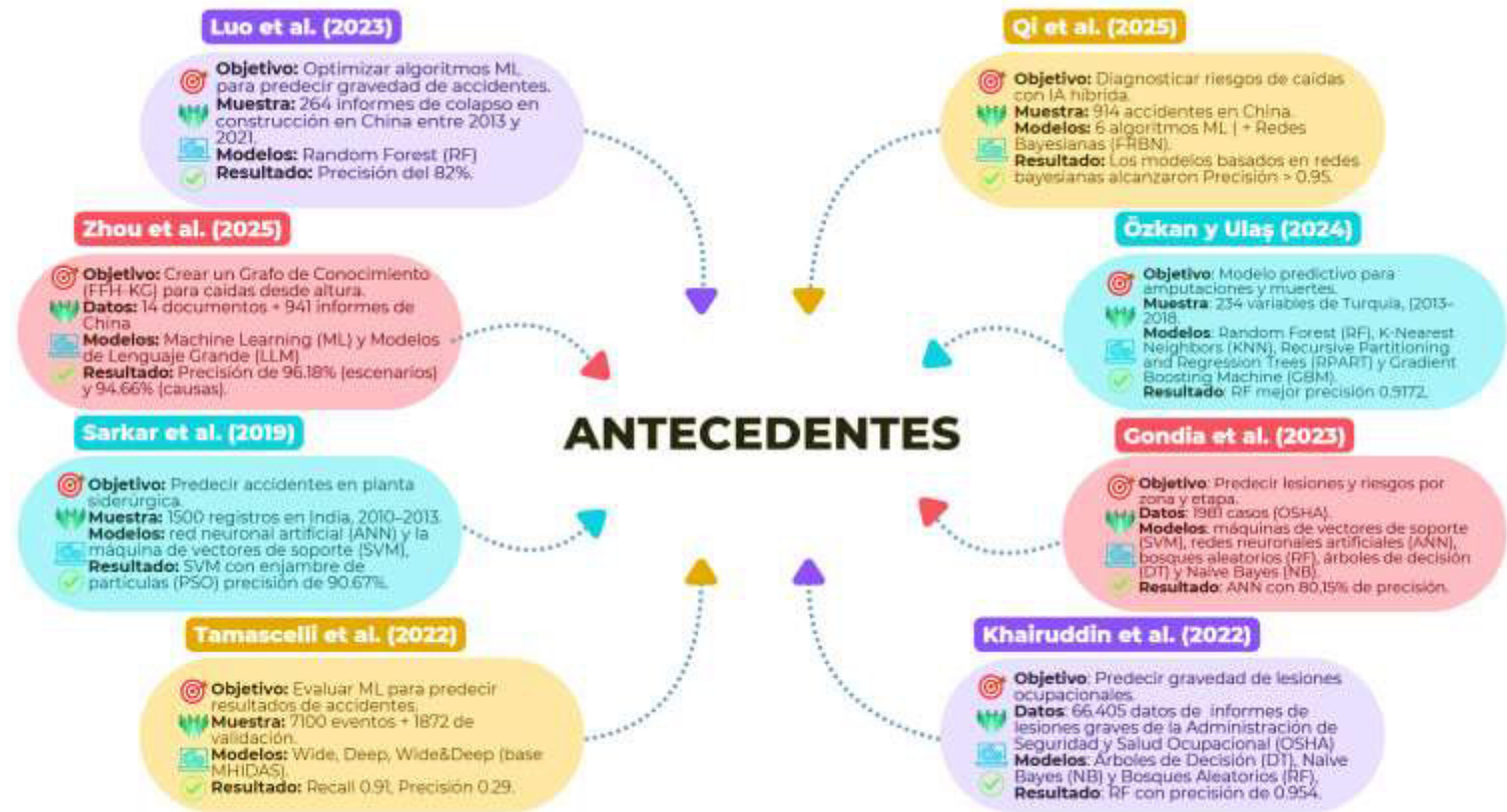




ANEXO N° 5

Figura 12

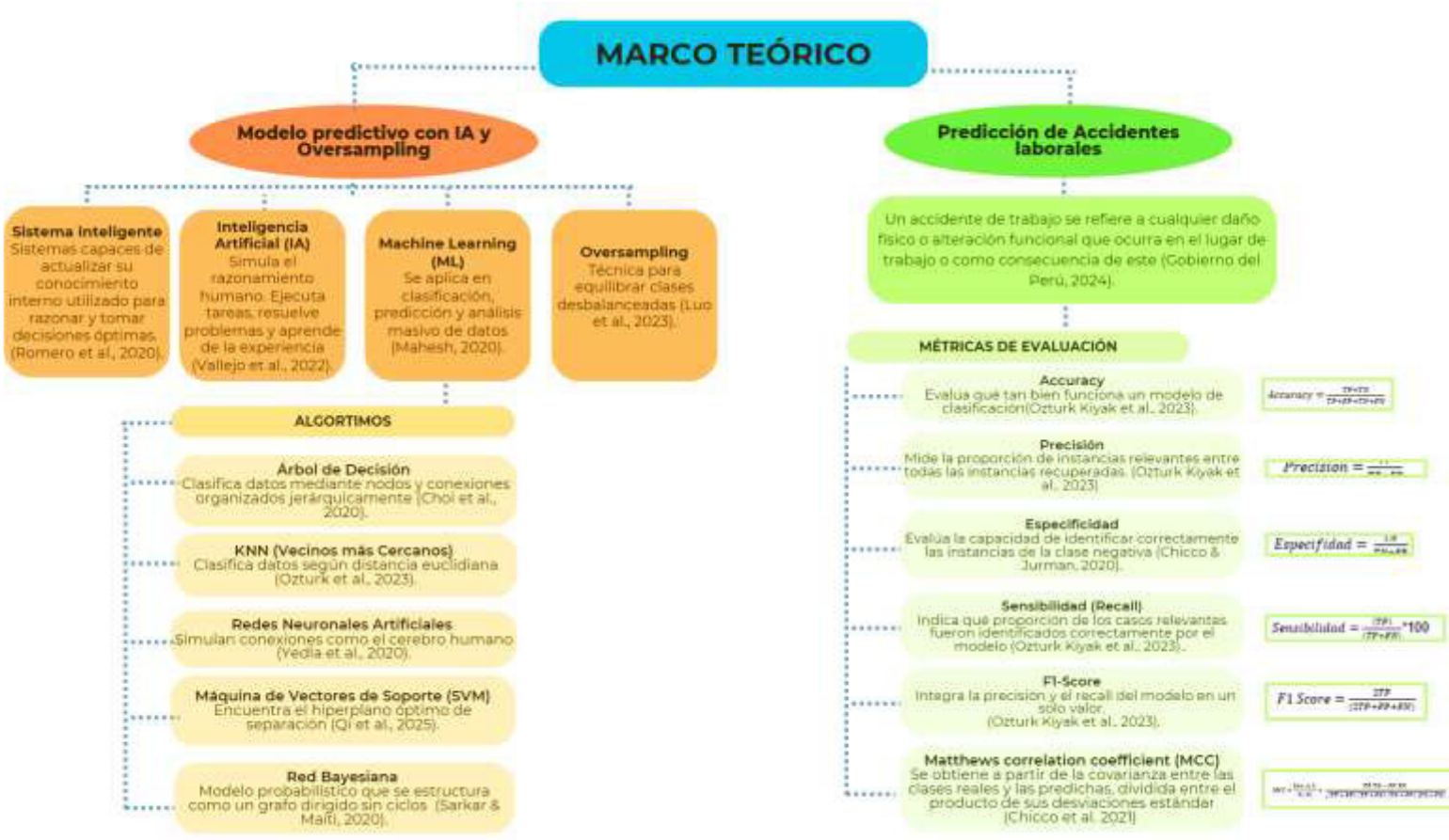
Mapa mental de antecedentes



ANEXO N°6

Figura 13

Mapa mental de marco teórico



ANEXO N° 7

Permiso de la empresa.



CORPORACIÓN DE ALIMENTOS MARÍTIMO S.A.C.

Producción y Comercialización de Productos
Hidrobiológicos y Agroindustriales

CARTA DE AUTORIZACIÓN DE USO DE INFORMACIÓN DE EMPRESA PARA OBTENCIÓN DE GRADO DE TÍTULO PROFESIONAL

Yo, JUAN CARLOS VELASQUEZ VALENTIN, identificado con DNI 41818704, en mi calidad de Gerente General de la empresa CORPORACION DE ALIMENTOS MARITIMO S.A.C. con R.U.C. N° 20600999797, ubicado en la ciudad de Chimbote, Jirón Huancavelica 1191 Pueblo Joven Florida Baja.

OTORGO LA AUTORIZACIÓN,

Al señor German David Leyva Cordova identificado con DNI N° 72084069 y el señor Brayan Marcelo Kano Leiva identificado con DNI N° 71848273 egresado/bachiller de la carrera de ingeniería de Sistemas e Informática para que utilice la siguiente información de la empresa:

Aplicación de encuestas, pruebas o información necesaria (*nombre, misión y visión e información adicional*), con la finalidad de que puedan desarrollar su Tesis para optar el Título Profesional.

Indicar si el Representante que autoriza la información de la empresa, solicita mantener el nombre o cualquier distintivo de la empresa en reserva, marcando con una "X" la opción seleccionada.

() Mantener en Reserva el nombre o cualquier distintivo de la empresa; o

(X) Mencionar el nombre de la empresa.

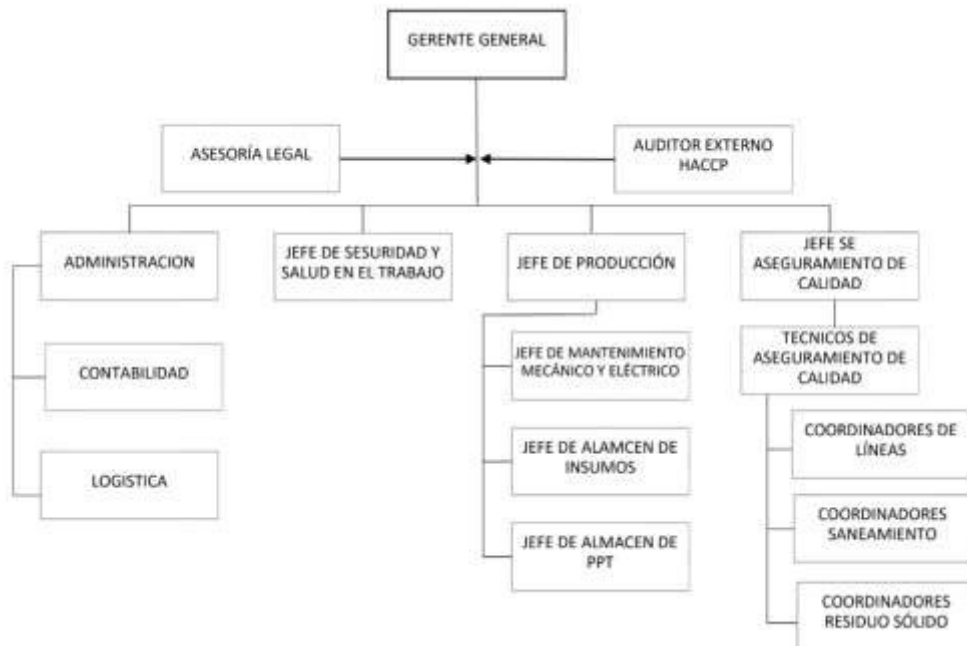
CORPORACION DE ALIMENTOS MARITIMO S.A.C.


Juan Carlos Velásquez Valentín
DNI: 41818704
GERENTE GENERAL

ANEXO N° 8

Figura 14

Estructura Orgánica de la Corporación De Alimentos Marítimo S.A.C.



ANEXO N° 9

Desarrollo de los algoritmos

Anexo A. Árbol de decisión

Figura 15. Importación de librerías y carga de datos

```
13  #librerías para el desarrollo del proyecto
import matplotlib.pyplot as plt #librería para graficas
from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches

import seaborn as sb

#librerías pandas para el manejo de los datos
import pandas as pd
import pandas as ps
import pandas as pf
import pandas as filtro_filas

#cargando los datos en el archivo excel
data = pd.read_excel("../data/data_pesquera.xlsx", index_col=None)
import numpy as np

#invocando a la librería de clasificación
from sklearn.tree import DecisionTreeClassifier
```

Figura 16. Identificación de la variable objetivo

```
11 Y = data['AccidenteLaboral']
12 #ya = data_datos['CALIFICACION']

13 Y
14 0    0
15 1    1
16 2    0
17 3    1
18 4    0
19 ..
20 105   0
21 106   1
22 107   1
23 108   0
24 109   1
25 Name: AccidenteLaboral, length: 110, dtype: int64

26 # 0 esta enfermo del corazon, 1 no esta enfermo del corazon
27 print(data.groupby('AccidenteLaboral').size())

28 AccidenteLaboral
29 0     51
30 1     59
31 dtype: int64

32 from sklearn.metrics import roc_curve, auc
33 from sklearn import datasets
34 from sklearn.model_selection import GridSearchCV
35 from sklearn.svm import LinearSVC
36 from sklearn.preprocessing import label_binarize
37 from sklearn.model_selection import train_test_split
38 import matplotlib.pyplot as plt
```

Figura 17. Particionamiento del conjunto de datos

```
11 #particionando la data en 0.8 para entrenar y 0.2 para validar
12 from sklearn.model_selection import train_test_split
13 X_trainset, X_testset, y_trainset, y_testset = train_test_split(X, Y, test_size=0.2, random_state=42)
14
15 print(X_trainset.shape, y_trainset.shape)
16 print(X_testset.shape, y_testset.shape)

17 (88, 7) (88,)
18 (22, 7) (22,)
```

Figura 18. Código de hiperparámetros y construcción del modelo Árbol de Decisión

```
# lista para almacenar los resultados
resultados = []

# INICIO DEL FOR
for i in range(1, 12):
    grid_search = GridSearchCV(
        DecisionTreeClassifier(),
        param_grid,
        refit=True,
        verbose=0,
        cv=5
    )

    # Entrenar modelo
    grid_search.fit(X_trainset, y_trainset)

    # AQUÍ SE AGREGA LA LÍNEA PARA CAPTURAR EL MEJOR MODELO
    #grid_search = grid_search.best_estimator_
    # .....
    # Predicciones
    pred_V1 = grid_search.predict(X_testset)
    y_prob = grid_search.predict_proba(X_testset)[:, 1] # Probabilidades clase positiva (1)

    # Métricas (para dos clases)
    accuracy = accuracy_score(y_testset, pred_V1)
    precision = precision_score(y_testset, pred_V1)
    recall = recall_score(y_testset, pred_V1)
    f1 = f1_score(y_testset, pred_V1)
    roc_auc = roc_auc_score(y_testset, y_prob)
    matthews = matthews_corrcoef(y_testset, pred_V1)

    resultados.append([
        "cv": i,
        "Accuracy": accuracy,
        "Precision": precision,
        "Recall": recall,
        "F1 score": f1,
        "ROC AUC": roc_auc,
        "Matthews": matthews,
        "Predicciones": pred_V1
    ])

11
```

Figura 19. Código para el cálculo de métricas de evaluación

```
resultados.append({
    "cv": i,
    "Accuracy": accuracy,
    "Precision": precision,
    "Recall": recall,
    "F1 score": f1,
    "ROC AUC": roc_auc,
    "Matthews": matthews,
    "Predicciones": pred_Y1
```

Figura 20. Código para generar la matriz de confusión y curva ROC

```
# Matriz de confusión y curva ROC
for resultado in resultados:
    # Matriz de confusión
    plt.figure(figsize=(6, 4))
    cmf_matrix = confusion_matrix(y_testset, resultado['Predicciones'])
    sb.heatmap(cmf_matrix, annot=True, fmt='d', cmap='Blues', cbar=False)
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Decision Tree Confusion Matrix (cv={resultado["cv"]})')
    plt.show()

    # Curva ROC
    plt.figure(figsize=(6, 4))
    y_prob = grid_search.predict_proba(X_testset)[:, 1]
    fpr, tpr, _ = roc_curve(y_testset, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.plot(fpr, tpr, lw=2, label=f'ROC curve (area = {roc_auc:0.2f})')
    plt.plot([0, 1], [0, 1], 'k--', lw=2)
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Decision Tree ROC Curve (cv={resultado["cv"]})')
    plt.legend(loc="lower right")
    plt.show()
```

Figura 21. Matriz de Confusión – Test (CV = 2) del modelo Árbol de Decisión

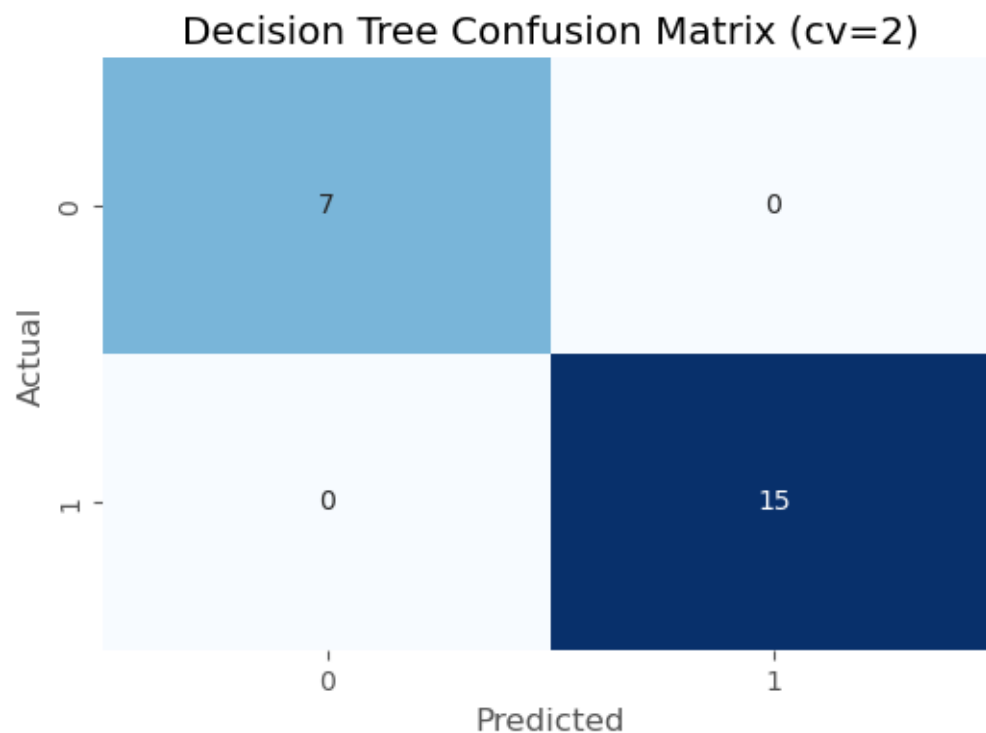


Figura 22. Curva ROC – Test (CV = 2) del modelo Árbol de Decisión

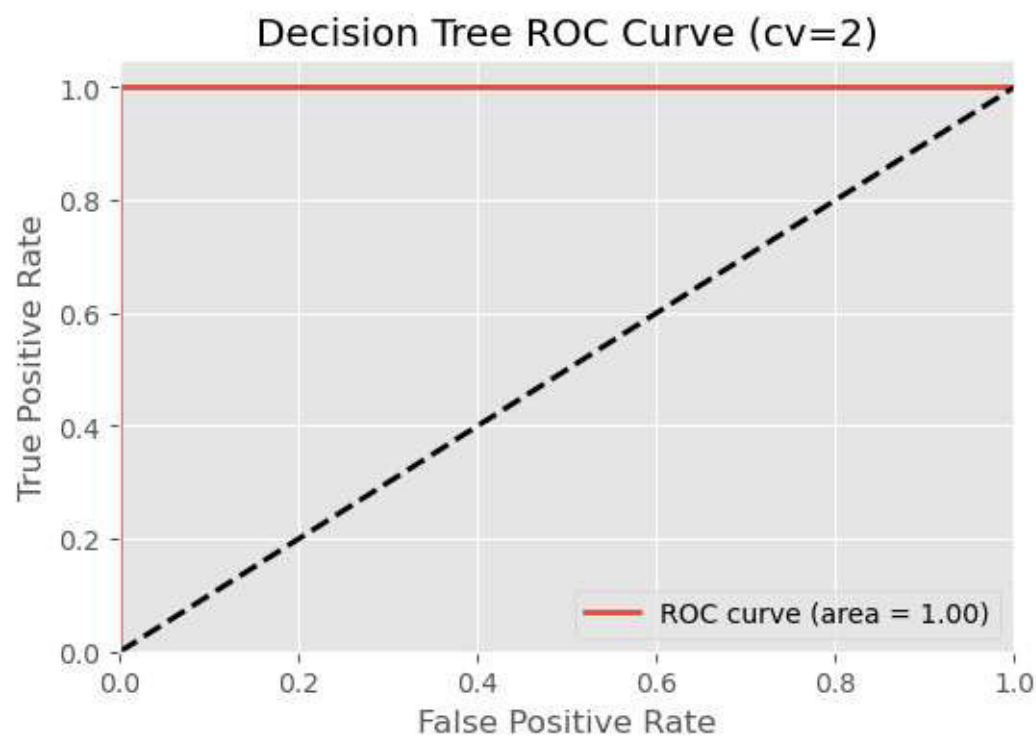


Figura 23. Matriz de Confusión – Test (CV = 3) del modelo Árbol de Decisión

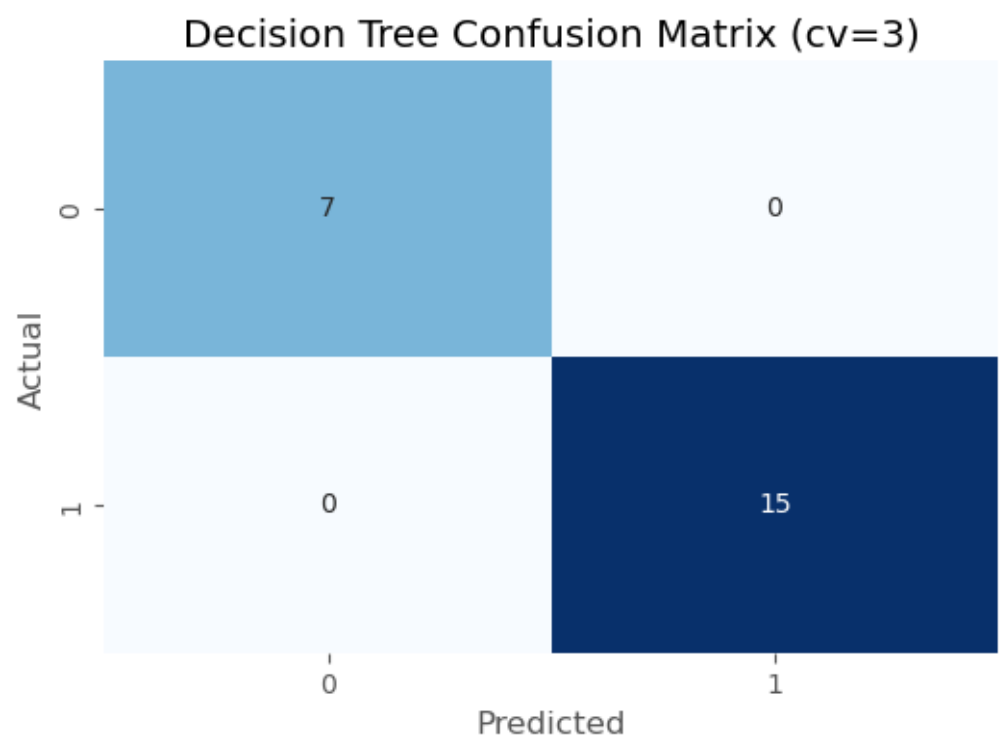


Figura 24. Curva ROC – Test (CV = 3) del modelo Árbol de Decisión

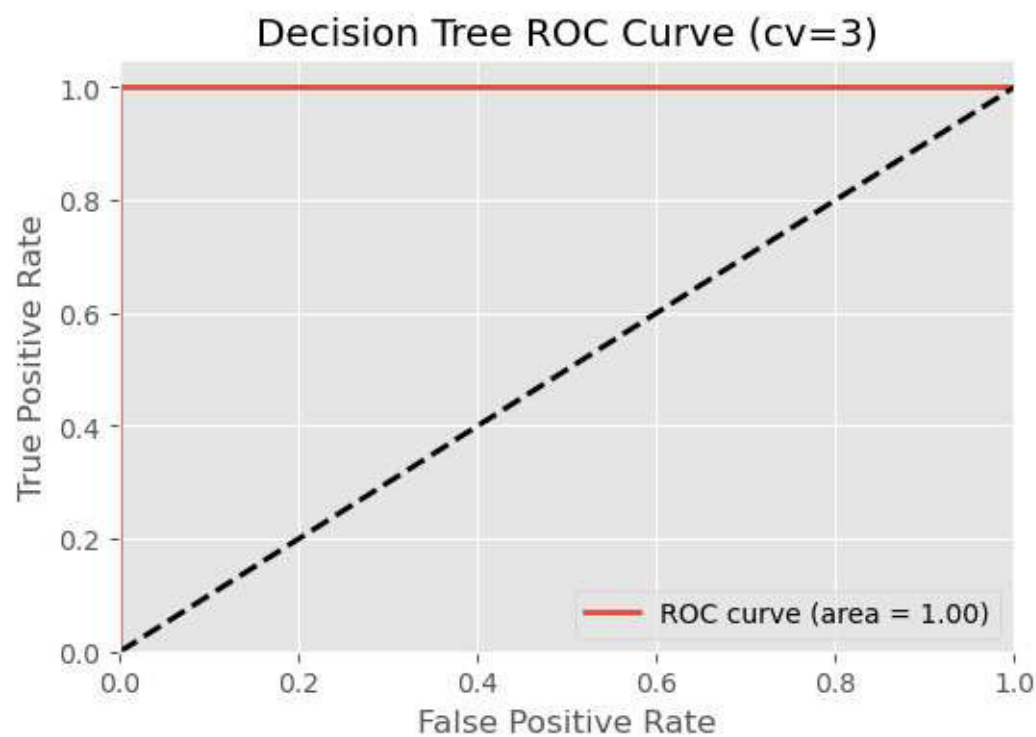


Figura 25. Matriz de Confusión – Test (CV = 4) del modelo Árbol de Decisión

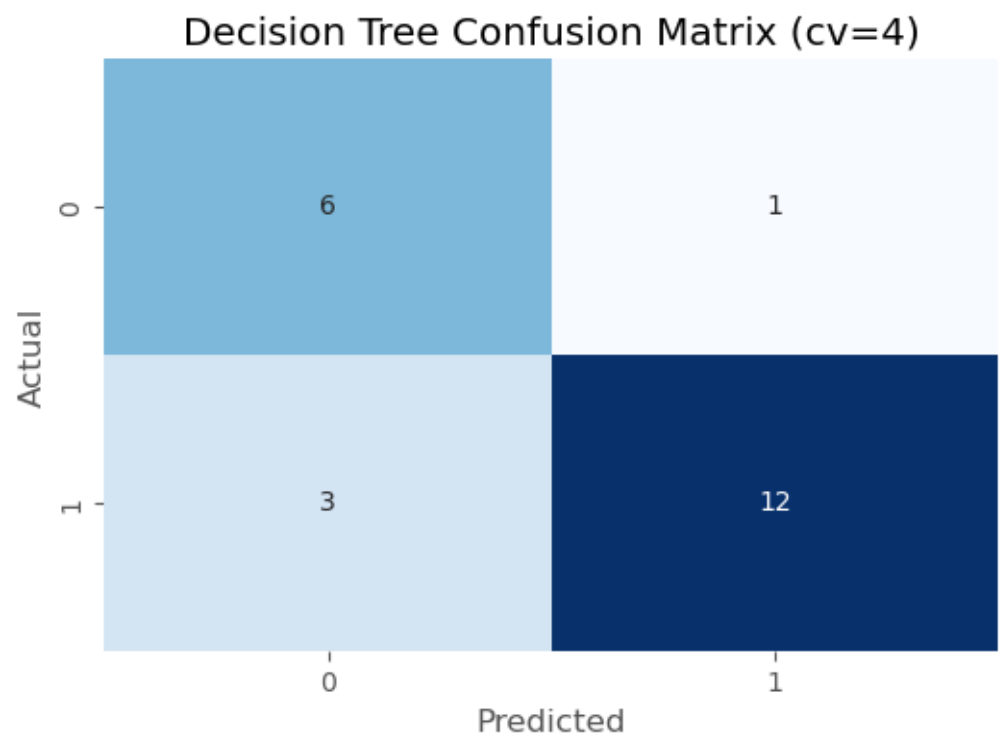


Figura 26. Curva ROC – Test (CV = 4) del modelo Árbol de Decisión

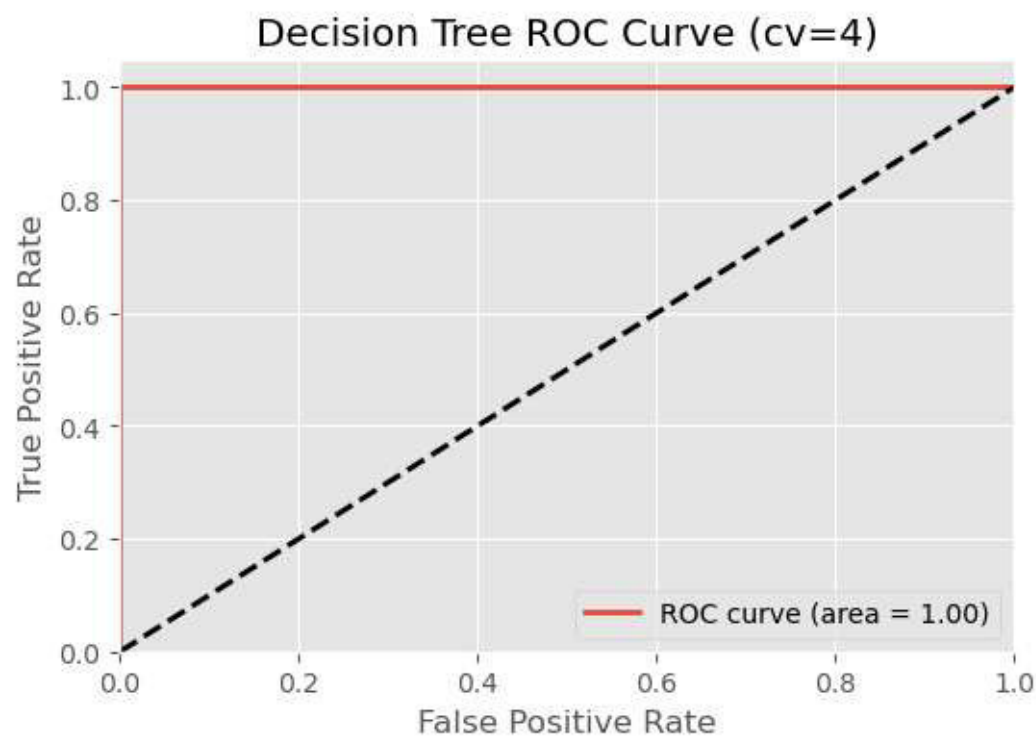


Figura 27. Matriz de Confusión – Test (CV = 5) del modelo Árbol de Decisión

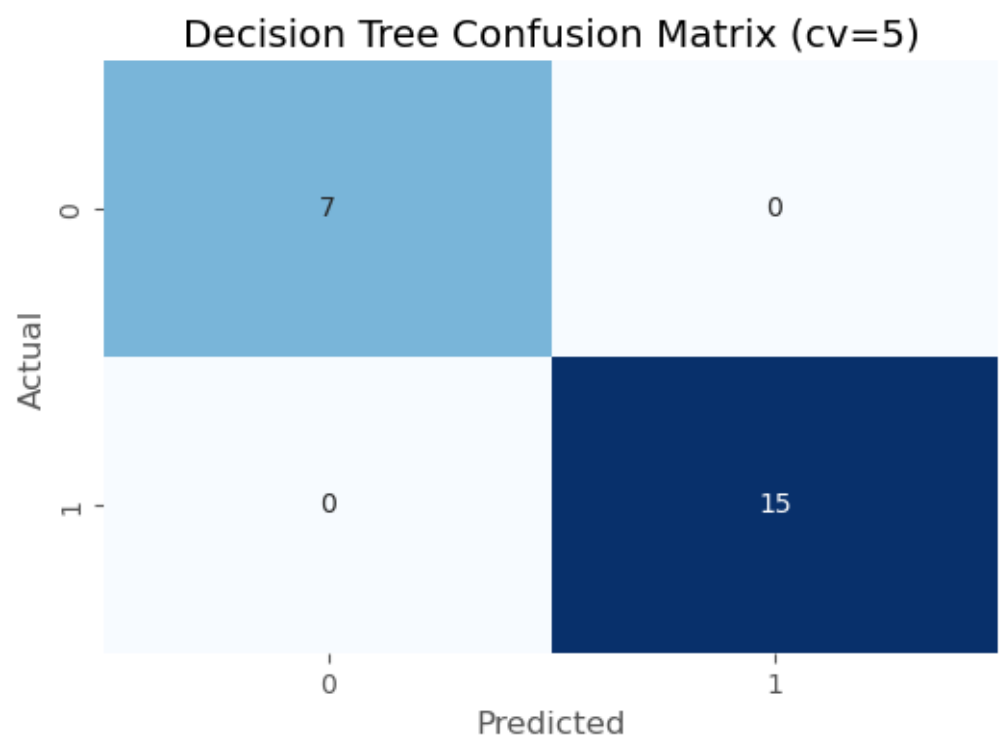


Figura 28. Curva ROC – Test (CV = 5) del modelo Árbol de Decisión

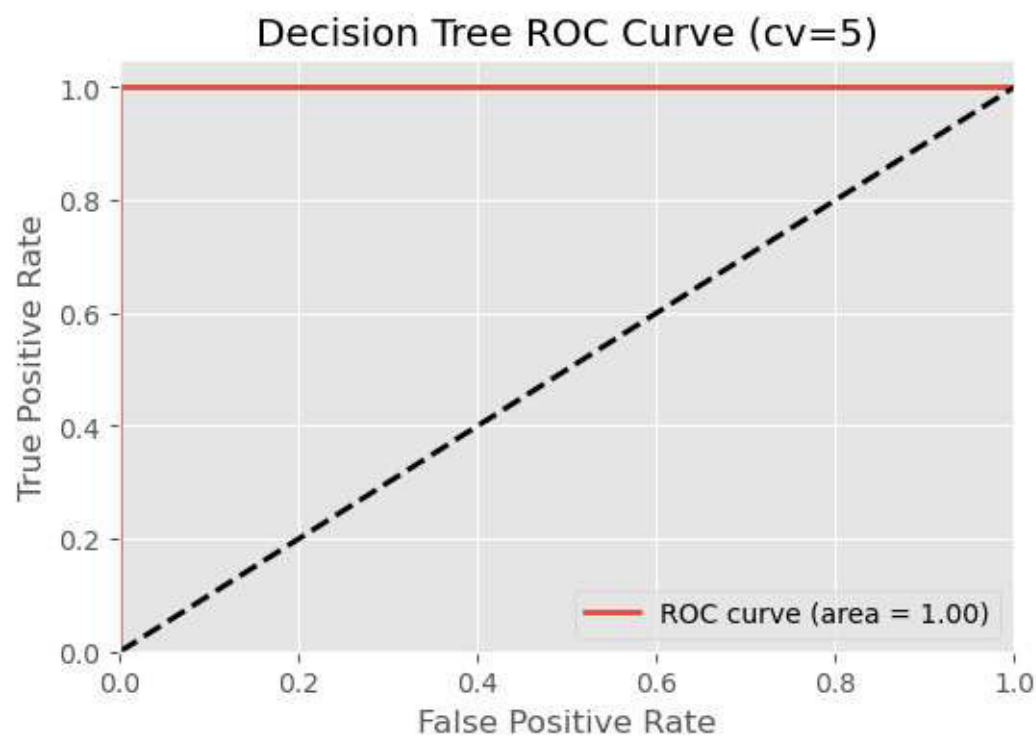


Figura 29. Matriz de Confusión – Test (CV = 6) del modelo Árbol de Decisión

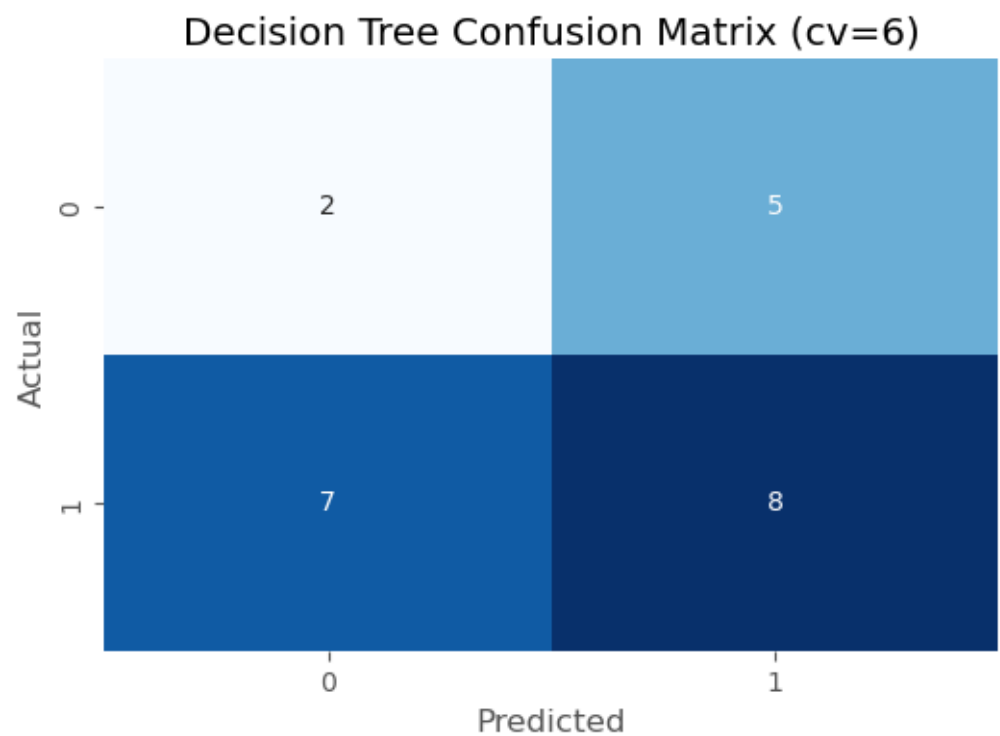


Figura 30. Curva ROC – Test (CV = 6) del modelo Árbol de Decisión

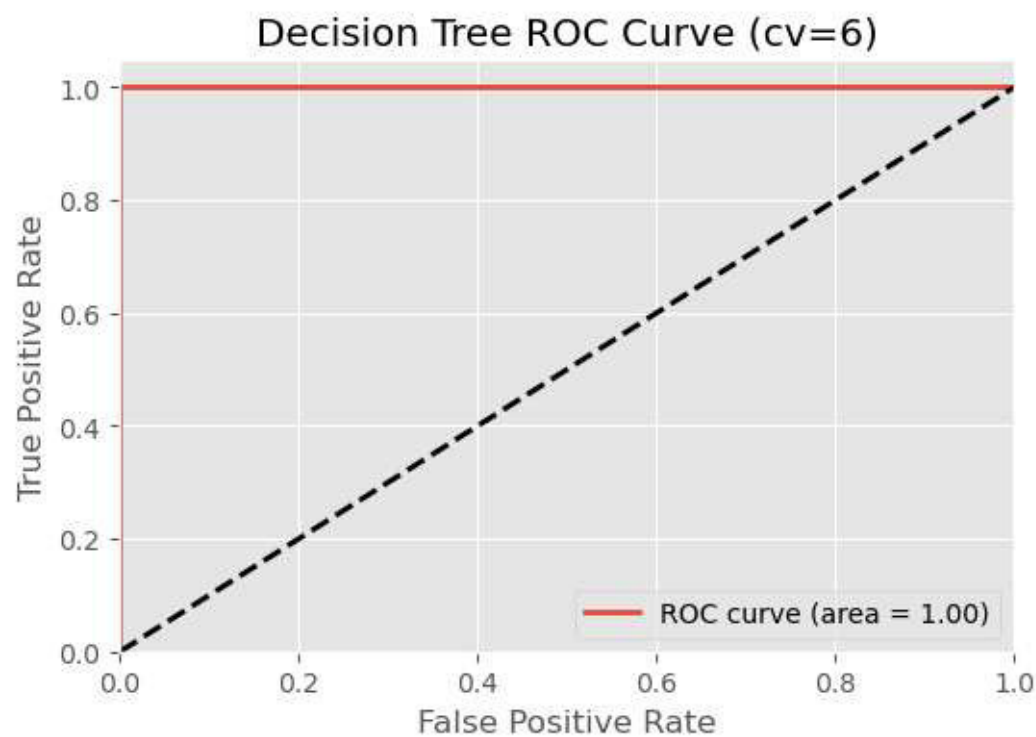


Figura 31. Matriz de Confusión – Test (CV = 7) del modelo Árbol de Decisión

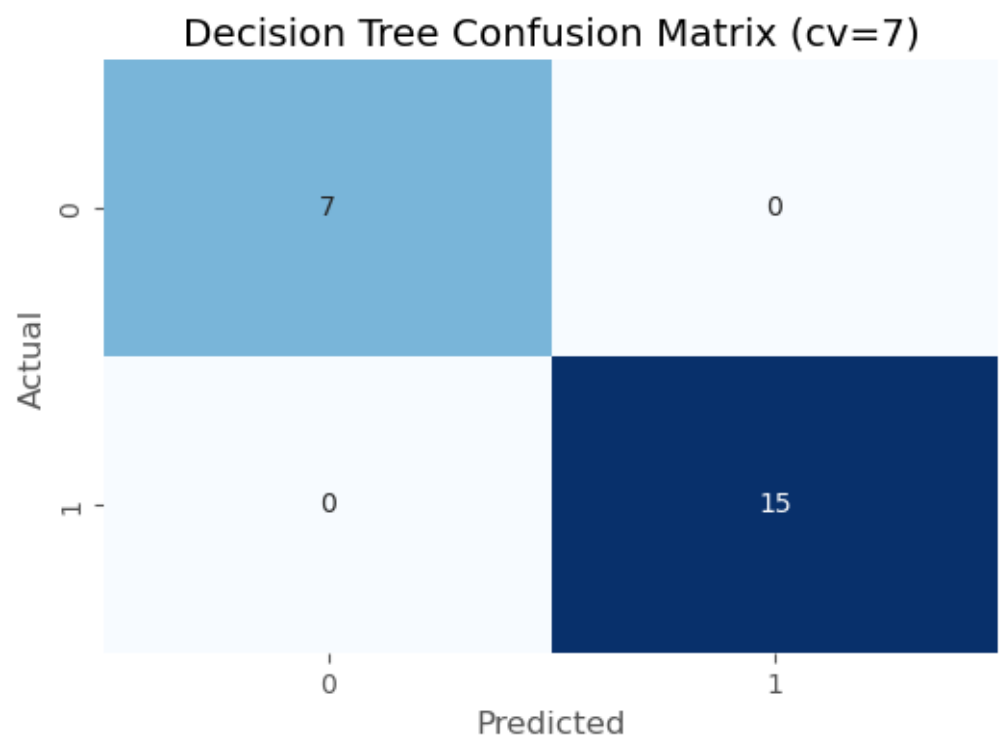


Figura 32. Curva ROC – Test (CV = 7) del modelo Árbol de Decisión

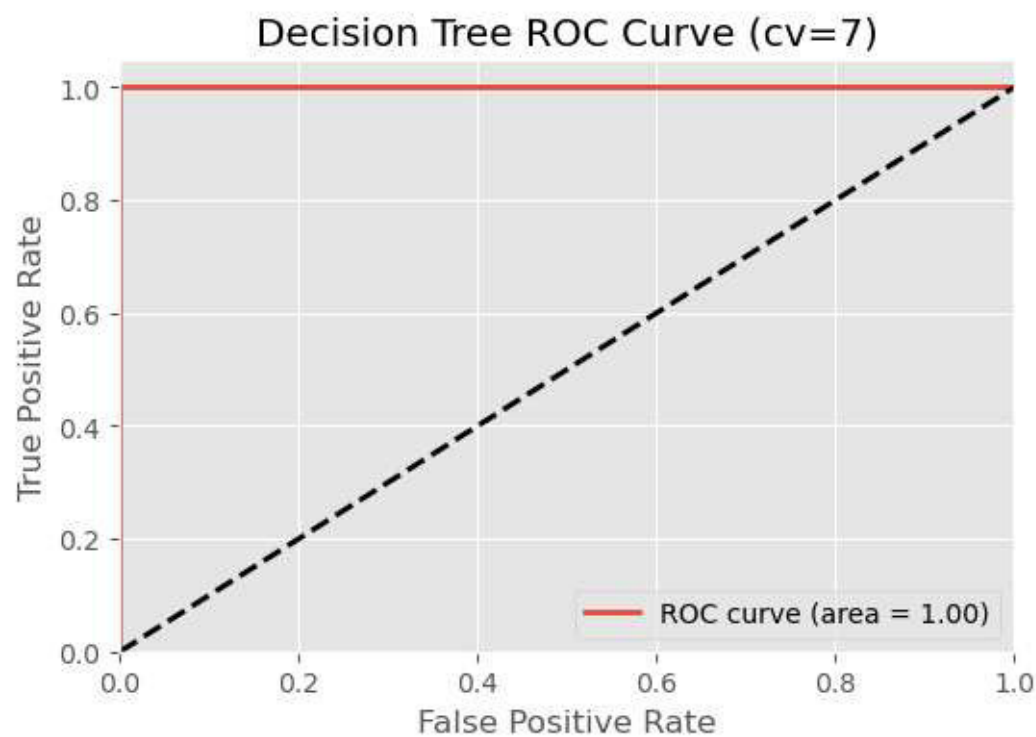


Figura 33. Matriz de Confusión – Test (CV = 8) del modelo Árbol de Decisión

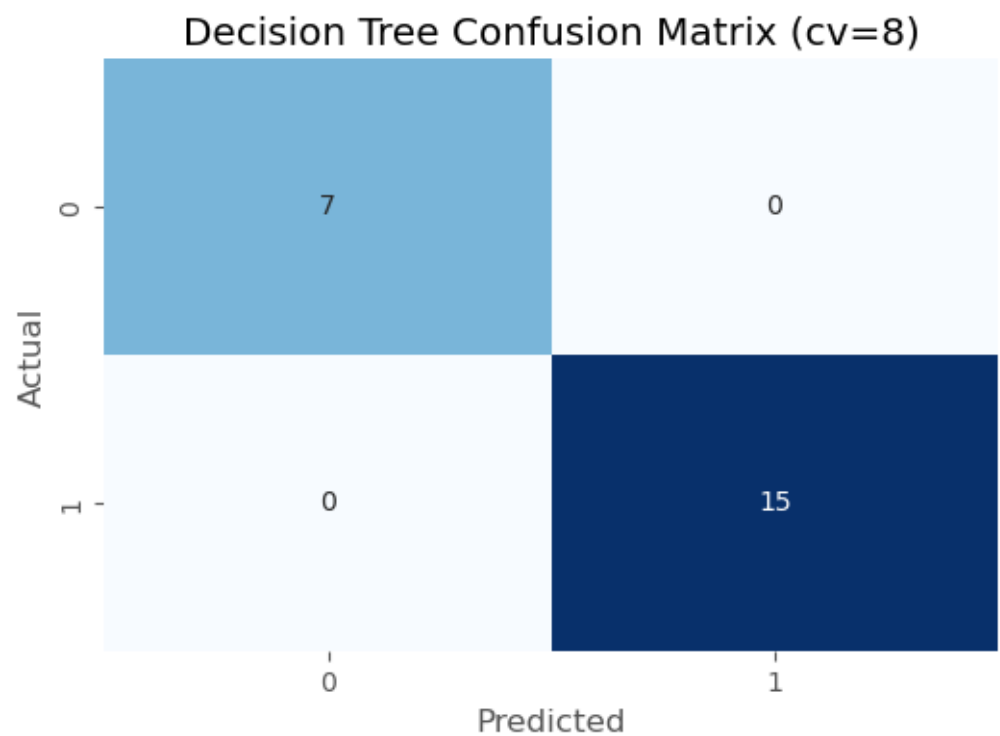


Figura 34. Curva ROC – Test (CV = 8) del modelo Árbol de Decisión

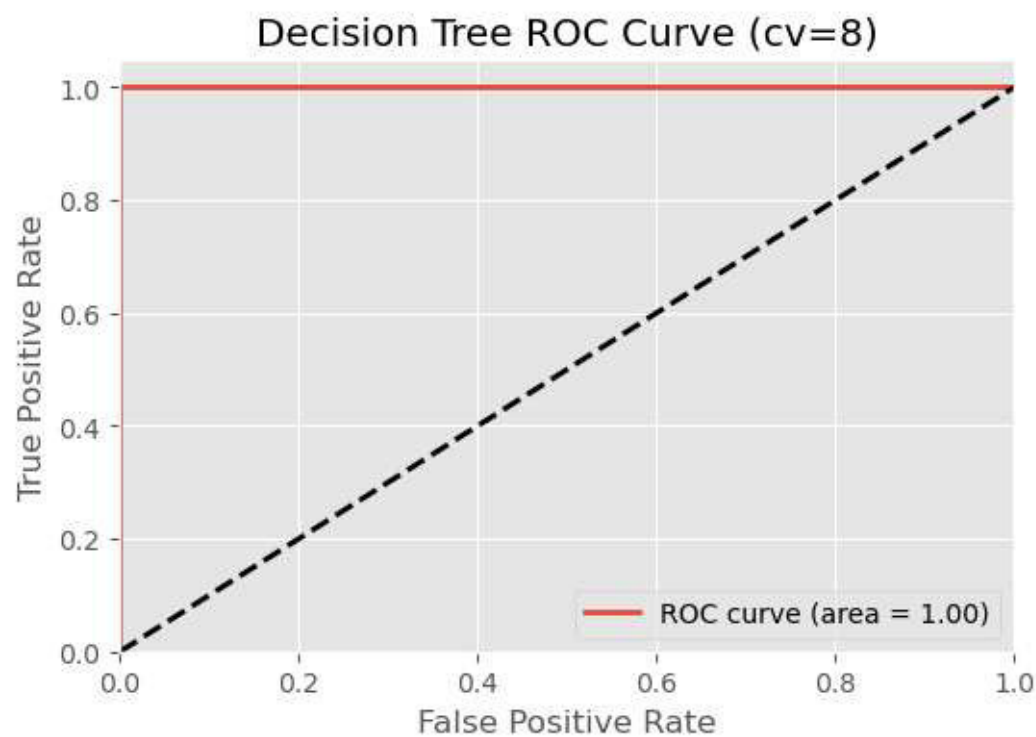


Figura 35. Matriz de Confusión – Test (CV = 9) del modelo Árbol de Decisión

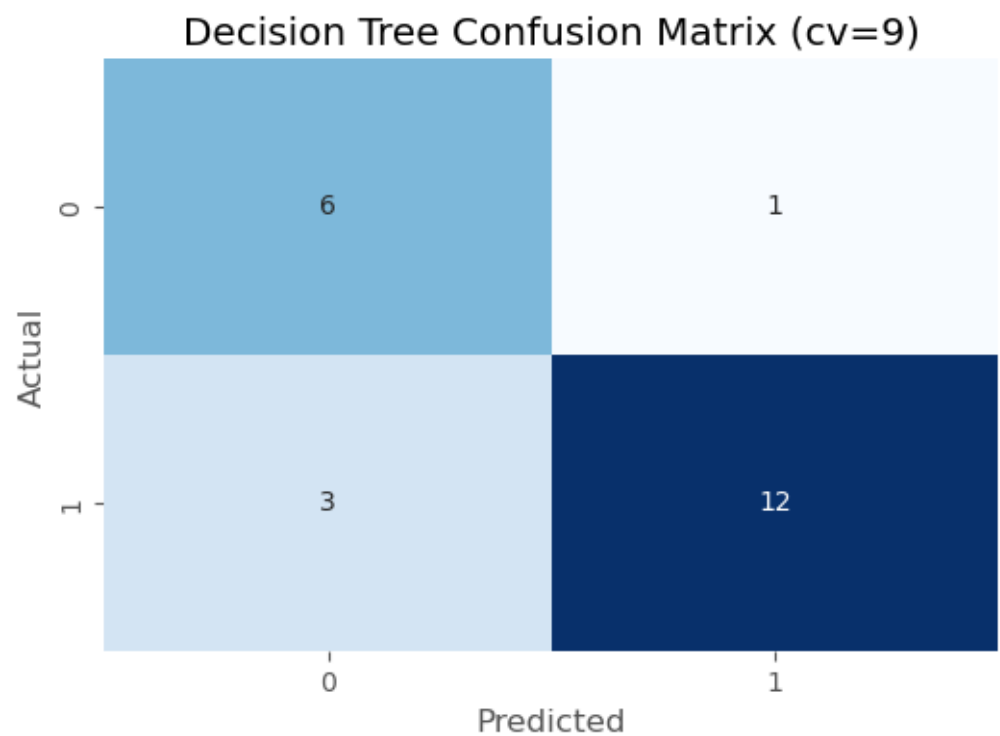


Figura 36. Curva ROC – Test (CV = 9) del modelo Árbol de Decisión

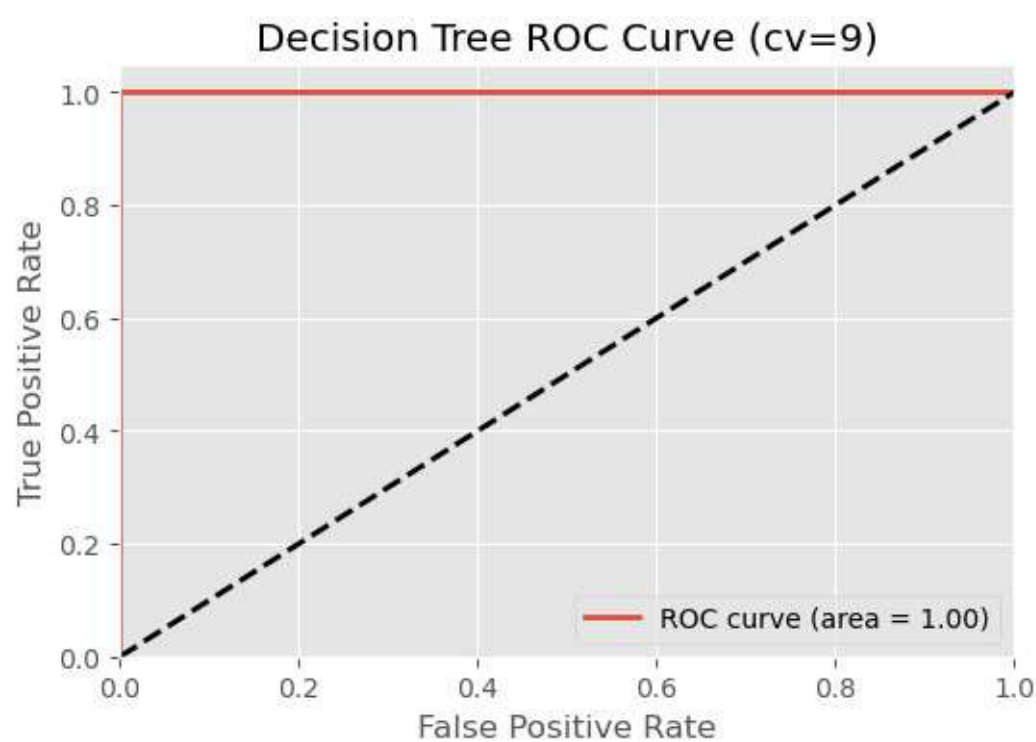


Figura 37. Matriz de Confusión – Test (CV = 10) del modelo Árbol de Decisión

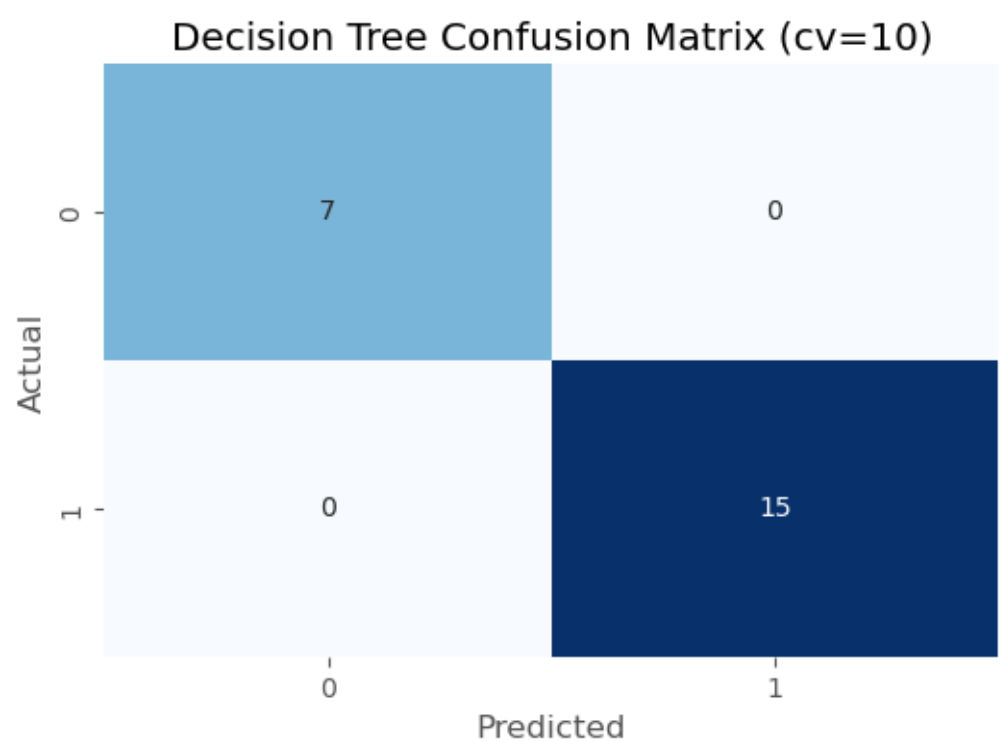


Figura 38. Curva ROC – Test (CV = 10) del modelo Árbol de Decisión

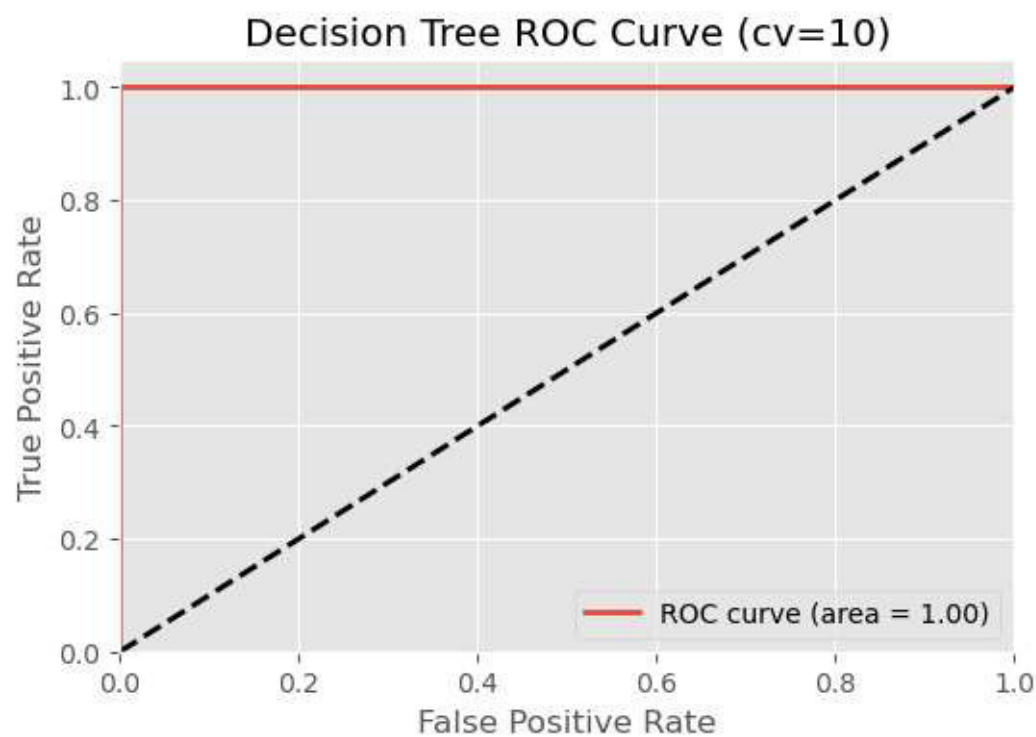


Figura 39. Matriz de Confusión – Test (CV = 11) del modelo Árbol de Decisión

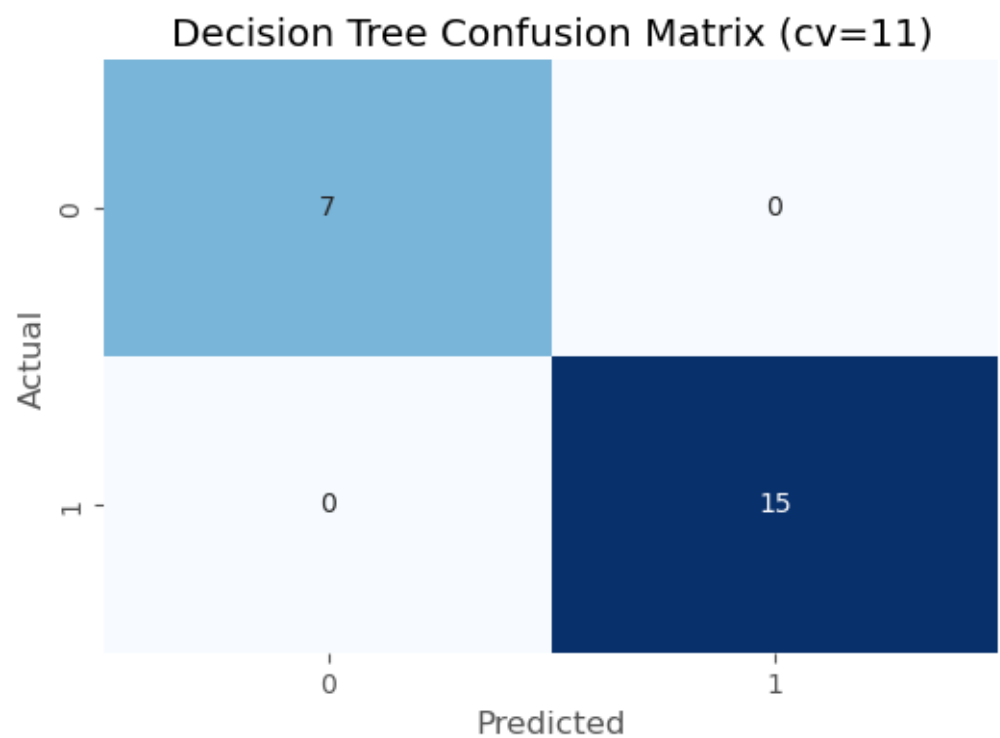
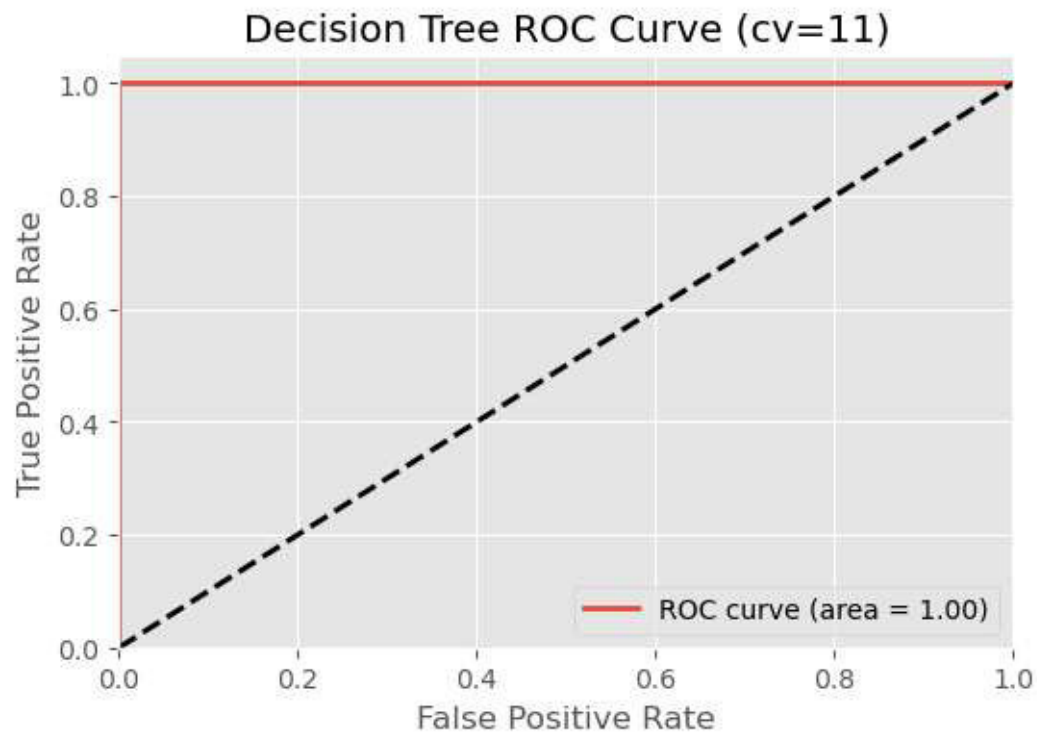


Figura 40. Curva ROC – Test (CV = 11) del modelo Árbol de Decisión



Anexo B. KNN

Figura 41. Importación de librerías y carga de datos

```
#librerías para el desarrollo del proyecto
import matplotlib.pyplot as plt #librería para graficas
from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches

import seaborn as sb

#librerías pandas para el manejo de los datos
import pandas as pd
import pandas as pa
import pandas as pf
import pandas as filtro_files

#enlazando los datos en el archivo excel
data = pd.read_excel("../data/data_pesquera.xlsx", index_col=None)
import numpy as np
#Invocando a la librería de clasificación
from sklearn.tree import DecisionTreeClassifier
```

Figura 42. Identificación de la variable objetivo

```

11 Y = data['Accidentalaboral']
    #y0 = data_datos['CALIFICACION']

12
13 Y
14
15 0    0
16 1    1
17 2    0
18 3    1
19 4    0
20 ...
21 105   0
22 106   1
23 107   1
24 108   0
25 109   1
26 Name: Accidentalaboral, length: 110, dtype: int64

27
28 # 0 esta enfermo del corazon, 1 no esta enfermo del corazon
    print(data.groupby('Accidentalaboral').size())
29
30 Accidentalaboral
31 0     11
32 1     99
33 dtype: int64

34
35 from sklearn.metrics import roc_curve, auc
    from sklearn import datasets
    from sklearn.multiclass import OneVsRestClassifier
    from sklearn.svm import LinearSVC
    from sklearn.preprocessing import label_binarize
    from sklearn.model_selection import train_test_split
    import matplotlib.pyplot as plt

```

Figura 43. Particionamiento del conjunto de datos

```

#particionando la data en 0.8 para entrenar y 0.2 para validar
from sklearn.model_selection import train_test_split
X_trainset, X_testset, y_trainset, y_testset = train_test_split(X,Y, test_size=0.2, random_state=3)

print(X_trainset.shape, y_trainset.shape)
print(X_testset.shape, y_testset.shape)

(88, 7) (88,)
(22, 7) (22,)

```

Figura 44. Código de hiperparámetros y construcción del modelo KNN

```

1 resultados = []

2 for i in range(2, 12):
3     grid_search = GridSearchCV(KNeighborsClassifier(),
4                               param_grid,
5                               refit=True,
6                               verbose=0,
7                               cv=i)
8
9     # Entrenar modelo
10    grid_search.fit(X_trainset, y_trainset)
11
12    # Predicciones
13    pred_Y1 = grid_search.predict(X_testset)
14    y_prob = grid_search.predict_proba(X_testset)[1, 1] # Probabilidades clase positiva (1)
15
16    # Métricas (para dos clases)
17    accuracy = accuracy_score(y_testset, pred_Y1)
18    precision = precision_score(y_testset, pred_Y1)
19    recall = recall_score(y_testset, pred_Y1)
20    f1 = f1_score(y_testset, pred_Y1)
21    roc_auc = roc_auc_score(y_testset, y_prob)
22    matthews = matthews_corrcoef(y_testset, pred_Y1)
23
24    resultados.append([
25        "cv": i,
26        "Accuracy": accuracy,
27        "Precision": precision,
28        "Recall": recall,
29        "F1 score": f1,
30        "ROC AUC": roc_auc,
31        "Matthews": matthews,
32        "Predicciones": pred_Y1
33    ])
34
35

```

Figura 45. Código para el cálculo de métricas de evaluación

```
resultados.append({
    "cv": i,
    "Accuracy": accuracy,
    "Precision": precision,
    "Recall": recall,
    "F1 score": f1,
    "ROC AUC": roc_auc,
    "Matthews": matthews,
    "Predicciones": pred_Y1
})
```

Figura 46. Código para generar la matriz de confusión y curva ROC

```
# Matriz de confusión y curva ROC
for resultado in resultados:
    # Matriz de confusión
    plt.figure(figsize=(6, 4))
    cmf_matrix = confusion_matrix(y_testset, resultado['Predicciones'])
    sb.heatmap(cmf_matrix, annot=True, fmt='d', cmap='Blues', cbar=False)
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Decision Tree Confusion Matrix (cv={resultado["cv"]})')
    plt.show()

    # Curva ROC
    plt.figure(figsize=(6, 4))
    y_prob = grid_search.predict_proba(X_testset)[1, 1]
    fpr, tpr, _ = roc_curve(y_testset, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.plot(fpr, tpr, lw=2, label=f'ROC curve (area = {roc_auc:0.2f})')
    plt.plot([0, 1], [0, 1], 'k--', lw=2)
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Decision Tree ROC Curve (cv={resultado["cv"]})')
    plt.legend(loc='lower right')
    plt.show()
```

Figura 47. Matriz de Confusión – Test (CV = 2) del modelo KNN

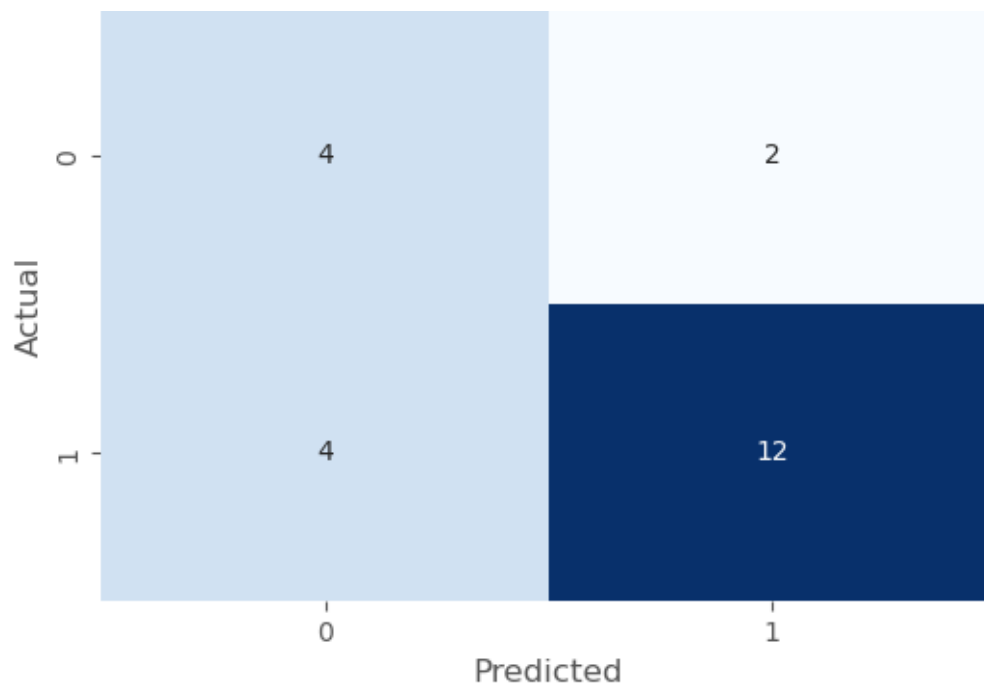


Figura 48. Curva ROC – Test (CV = 2) del modelo KNN

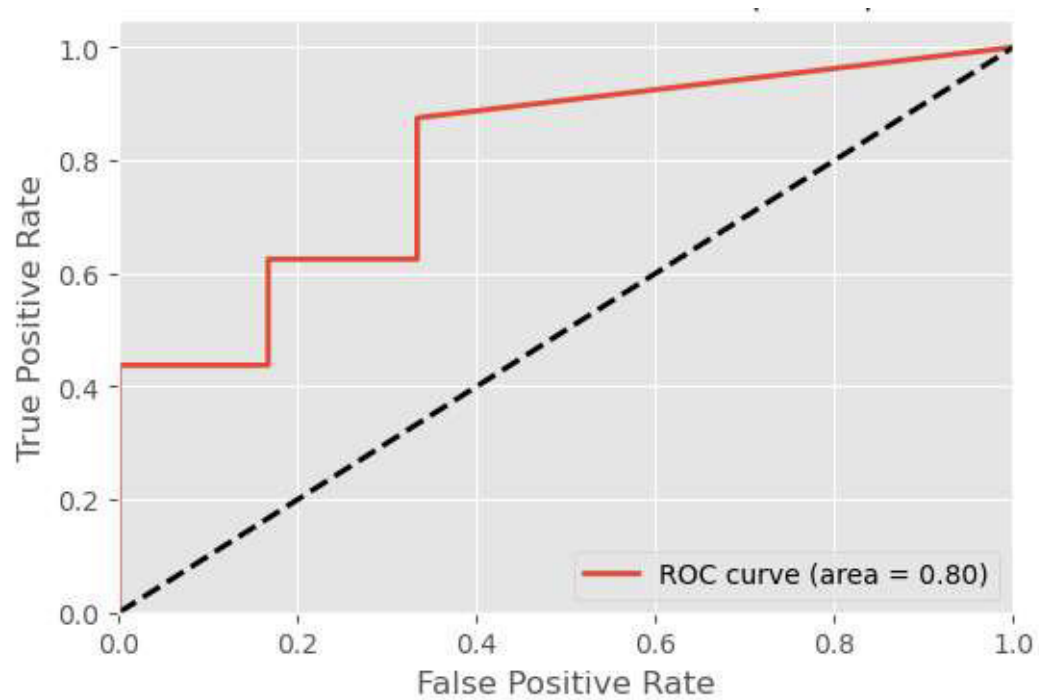


Figura 49. Matriz de Confusión – Test (CV = 3) del modelo KNN

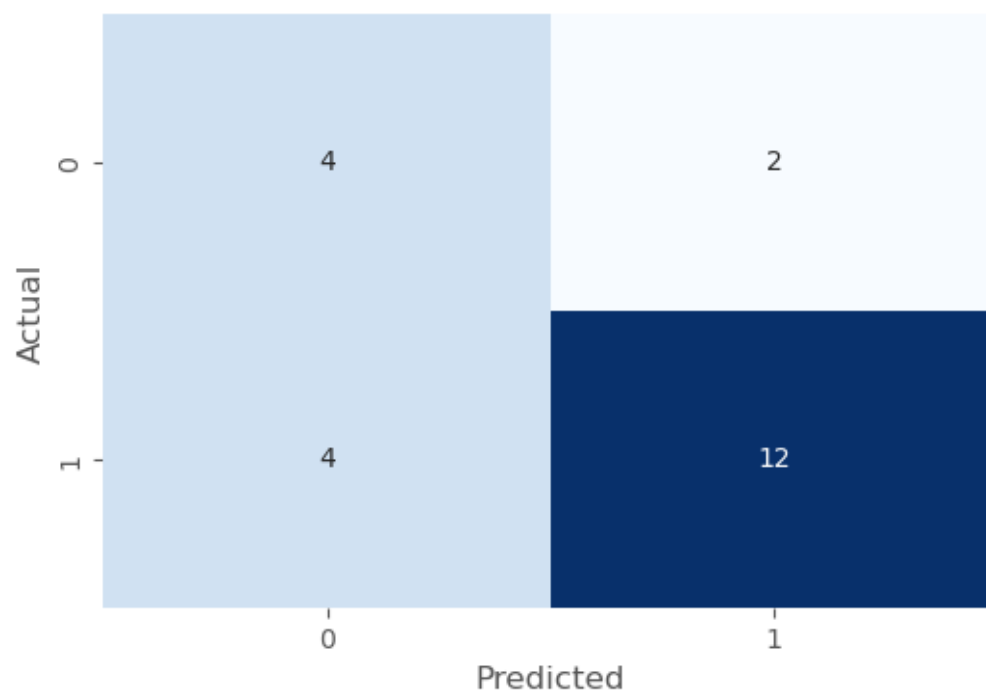


Figura 50. Curva ROC – Test (CV = 3) del modelo KNN

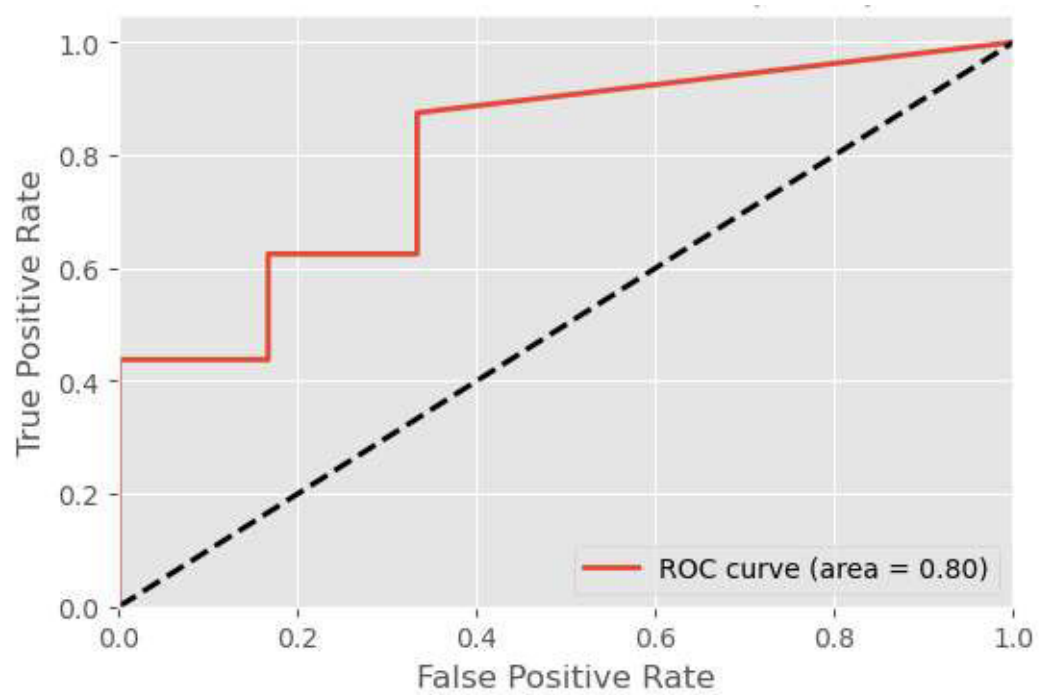


Figura 51. Matriz de Confusión – Test (CV = 4) del modelo KNN

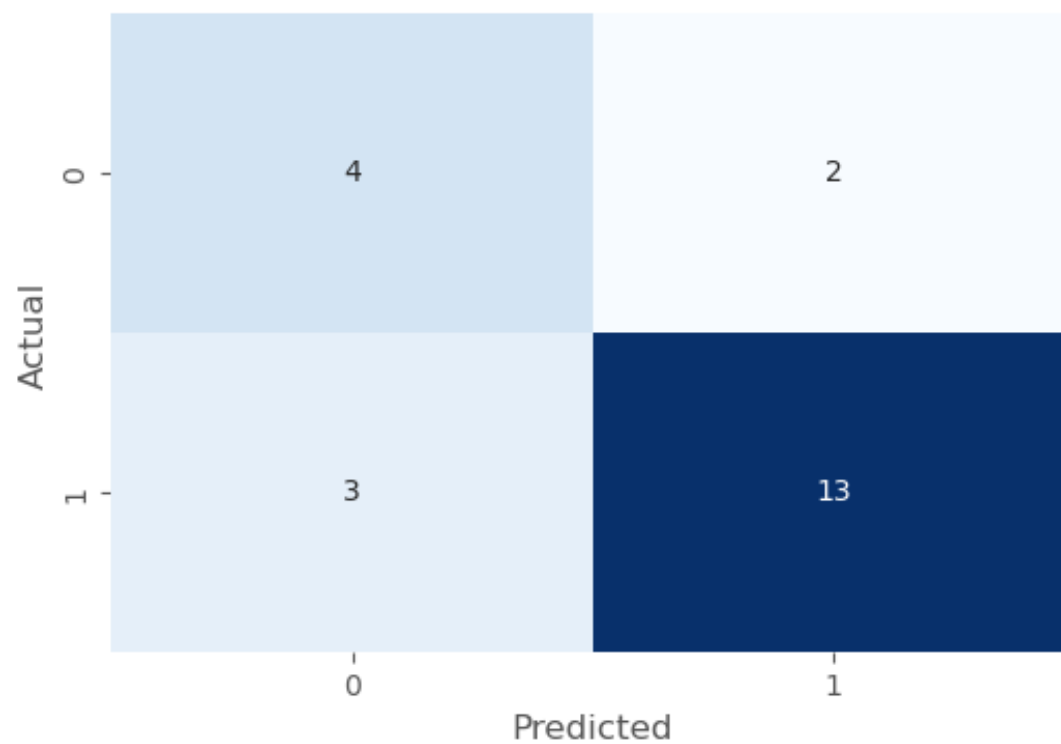


Figura 52. Curva ROC – Test (CV = 4) del modelo KNN

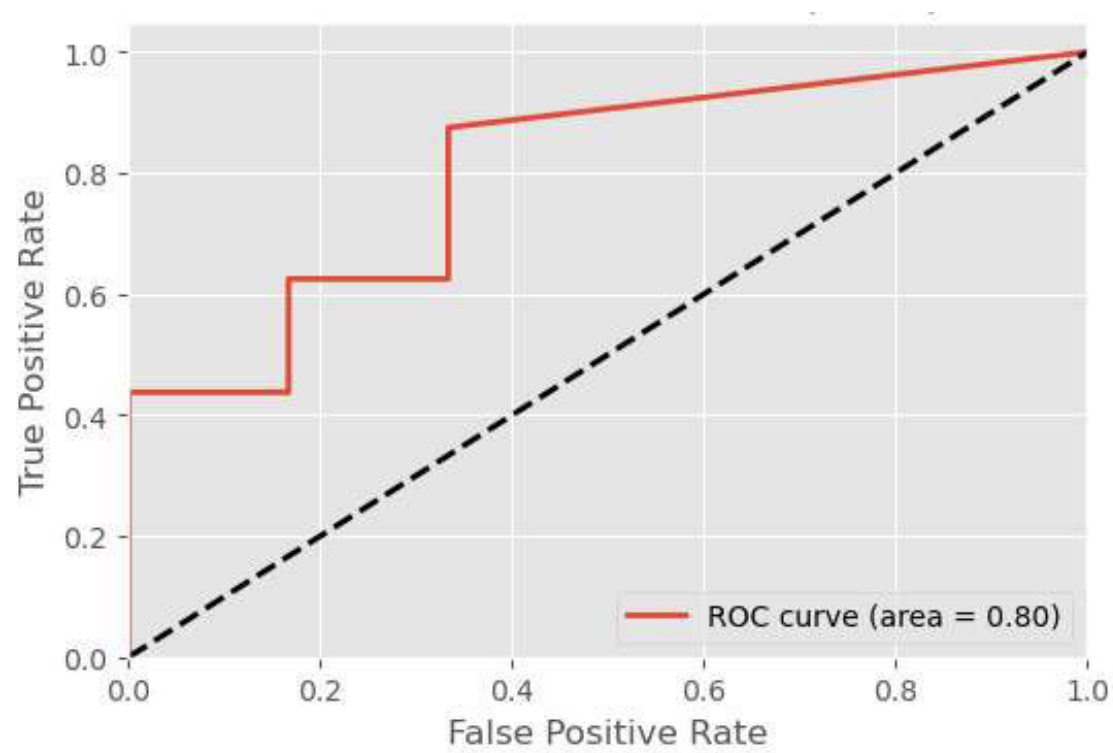


Figura 53. Matriz de Confusión – Test (CV = 5) del modelo KNN

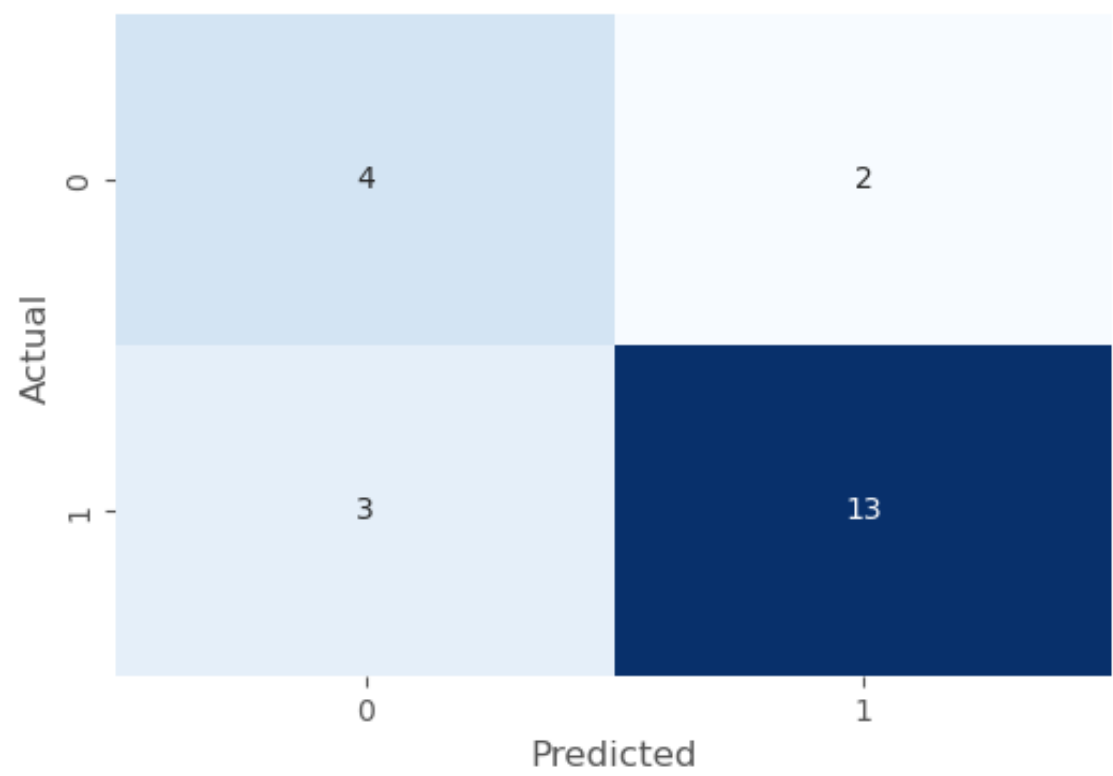


Figura 54. Curva ROC – Test (CV = 5) del modelo KNN

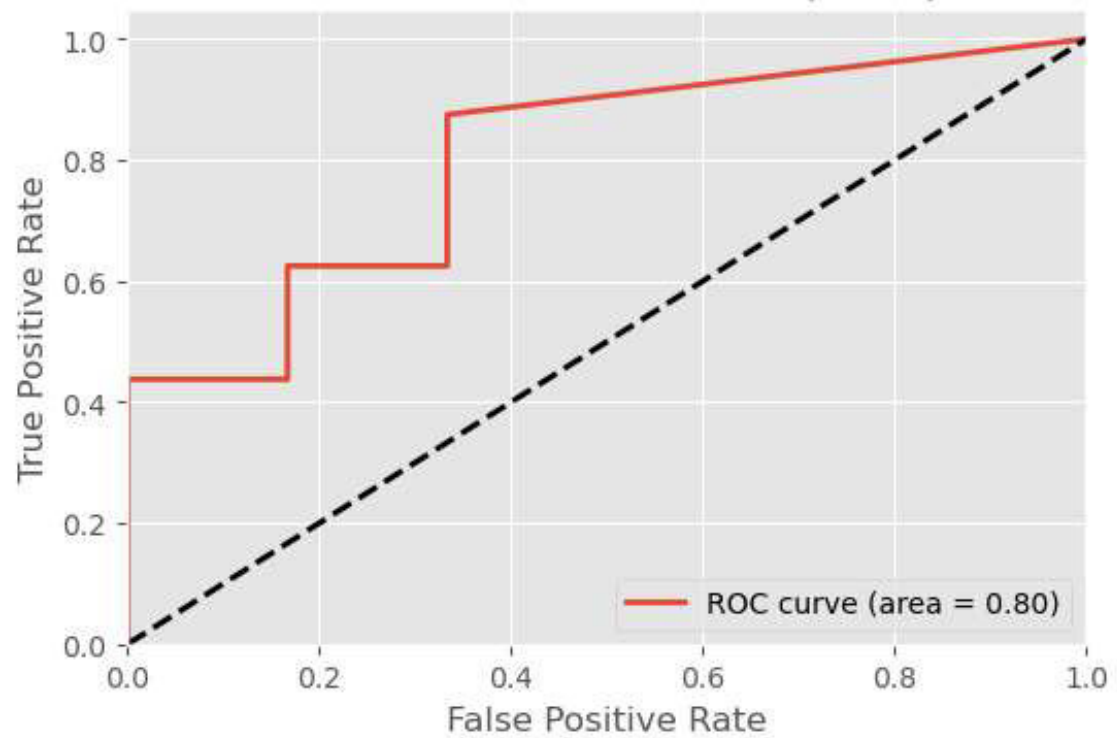


Figura 55. Matriz de Confusión – Test (CV = 6) del modelo KNN

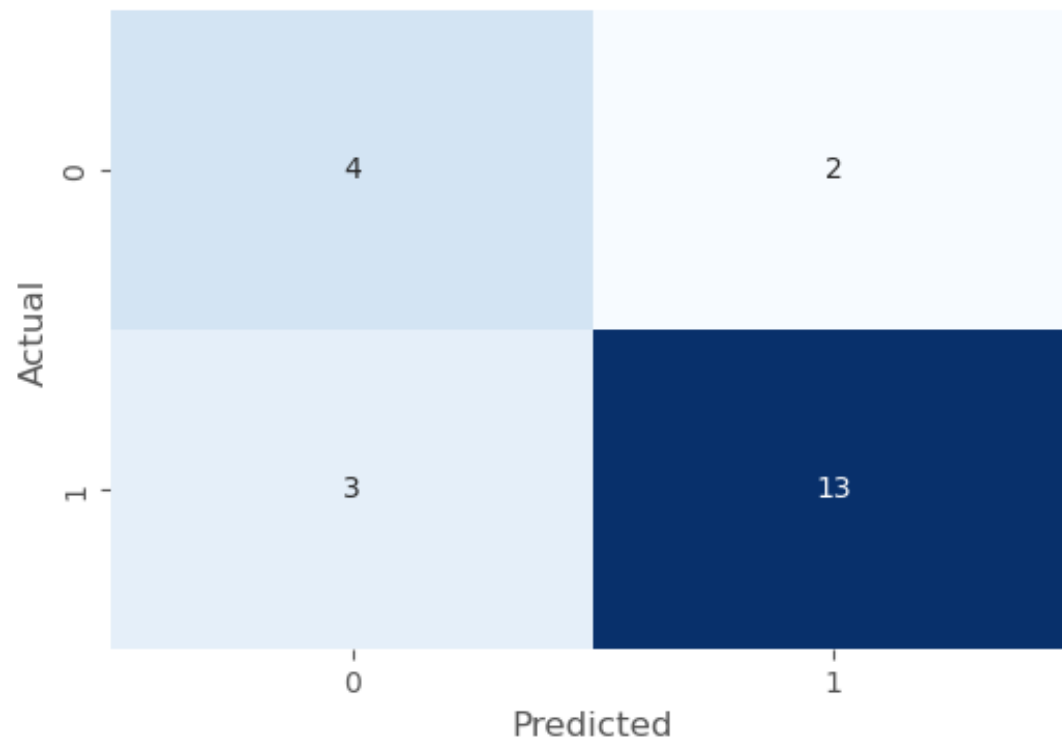


Figura 56. Curva ROC – Test (CV = 6) del modelo KNN

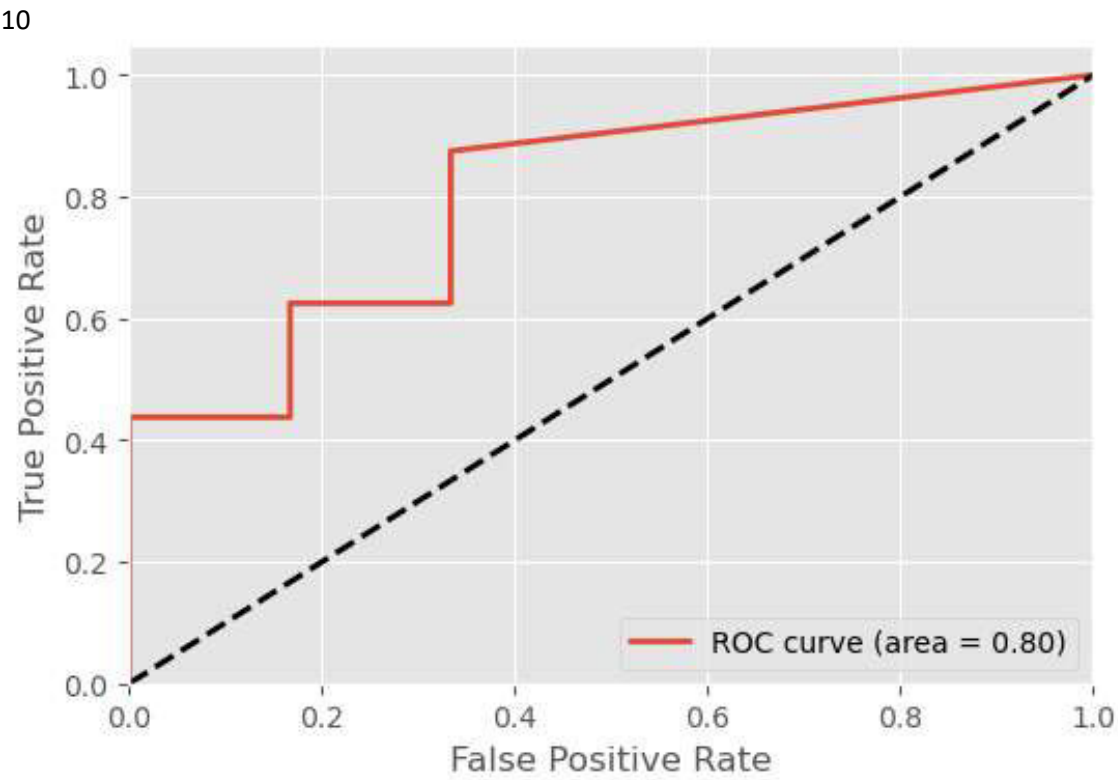


Figura 57. Matriz de Confusión – Test (CV = 7) del modelo KNN

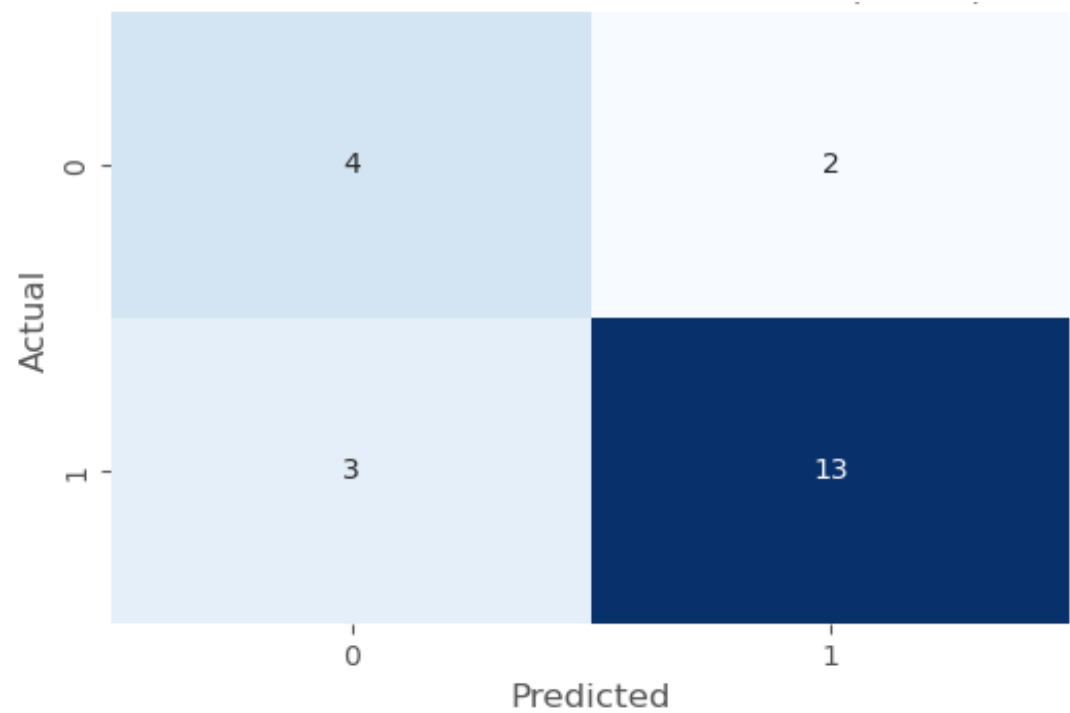


Figura 58. Curva ROC – Test (CV = 7) del modelo KNN

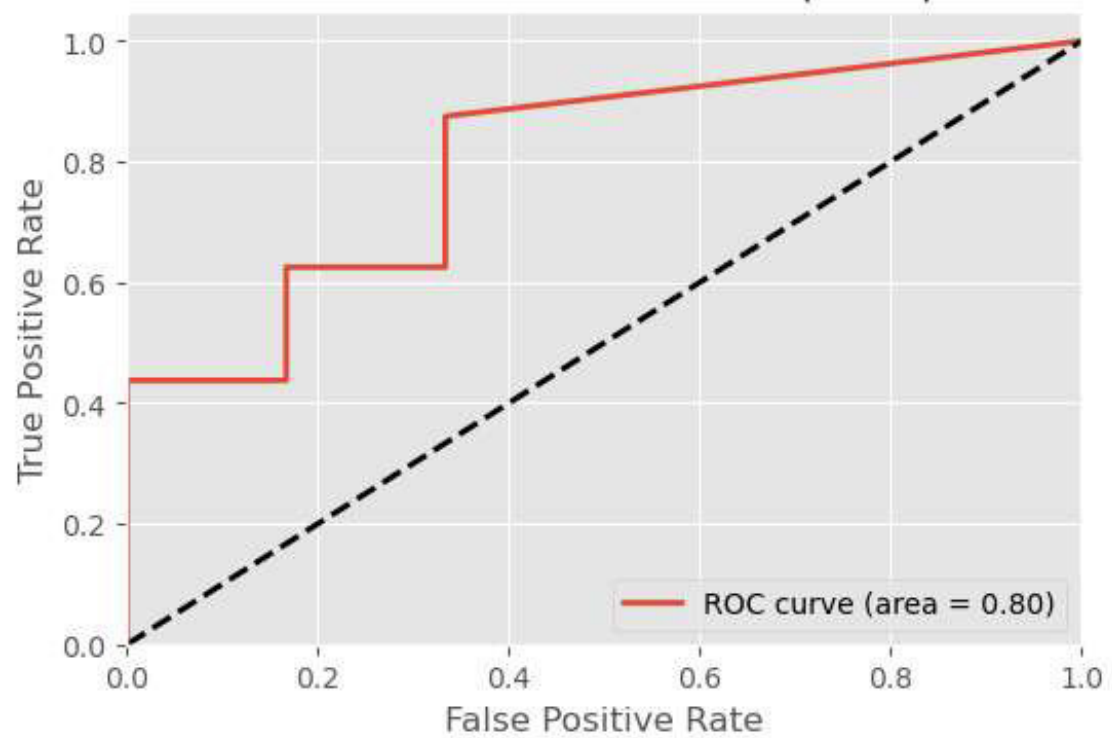


Figura 59. Matriz de Confusión – Test (CV = 8) del modelo KNN

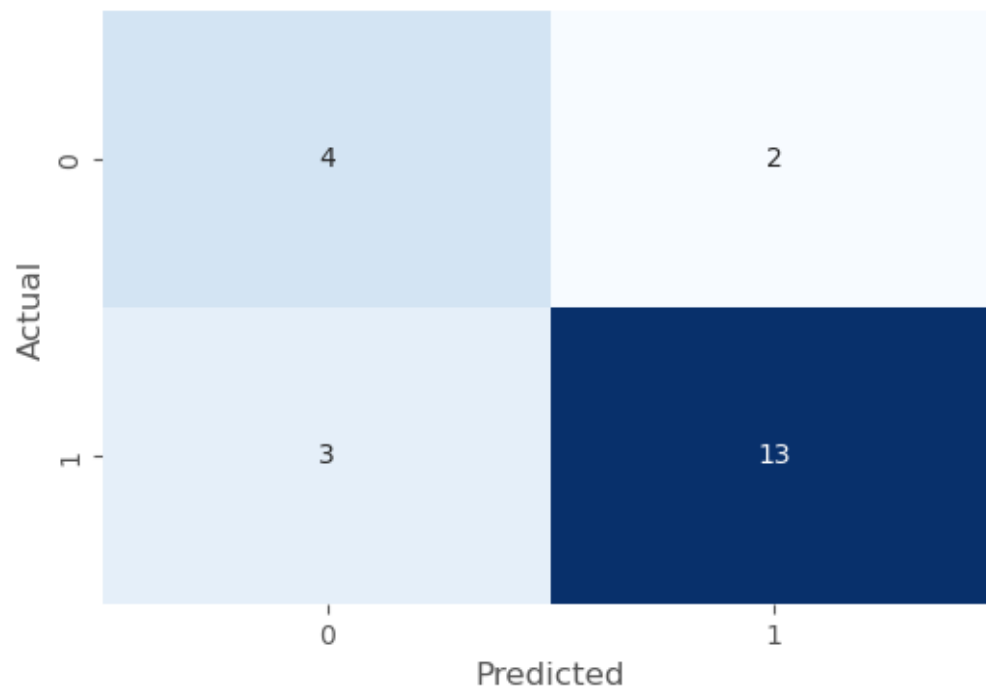


Figura 60. Curva ROC – Test (CV = 8) del modelo KNN

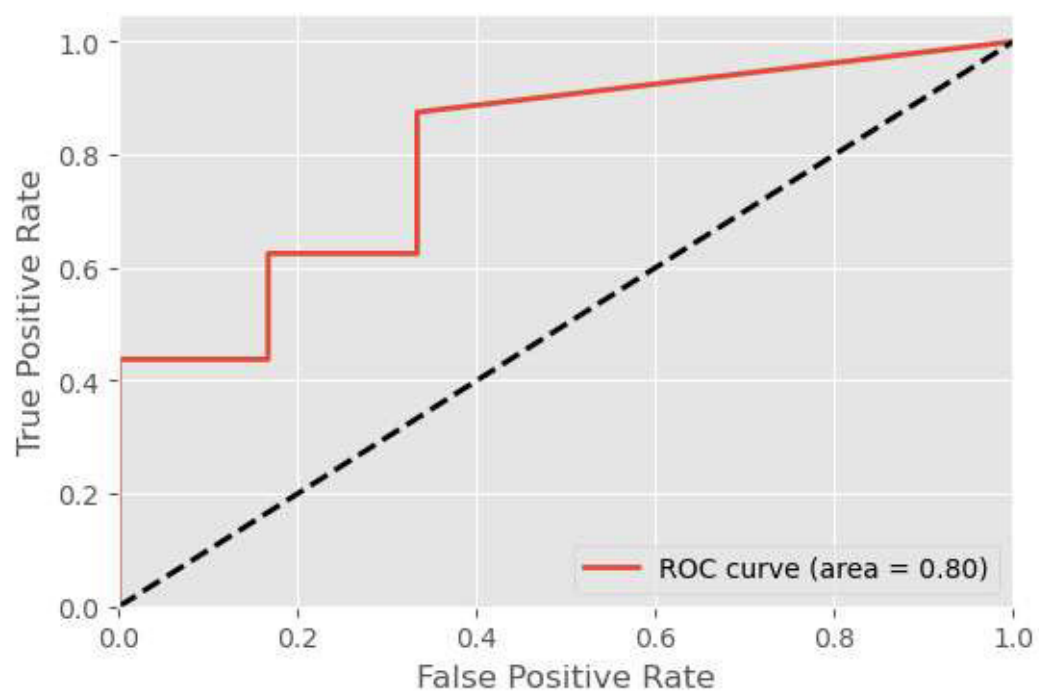


Figura 61. Matriz de Confusión – Test (CV = 9) del modelo KNN

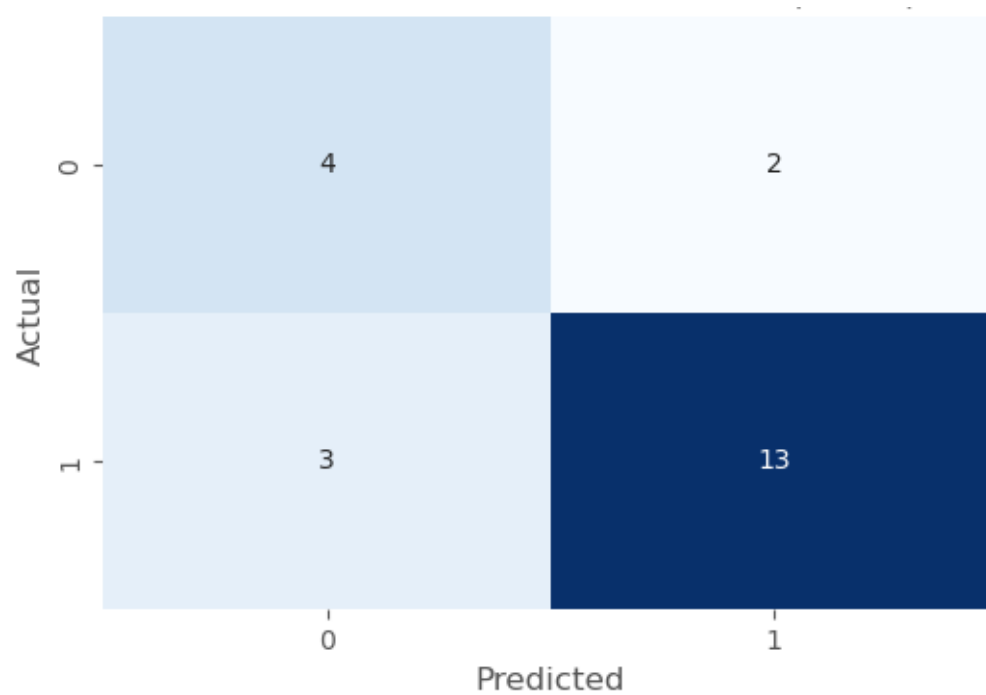


Figura 62. Curva ROC – Test (CV = 9) del modelo KNN

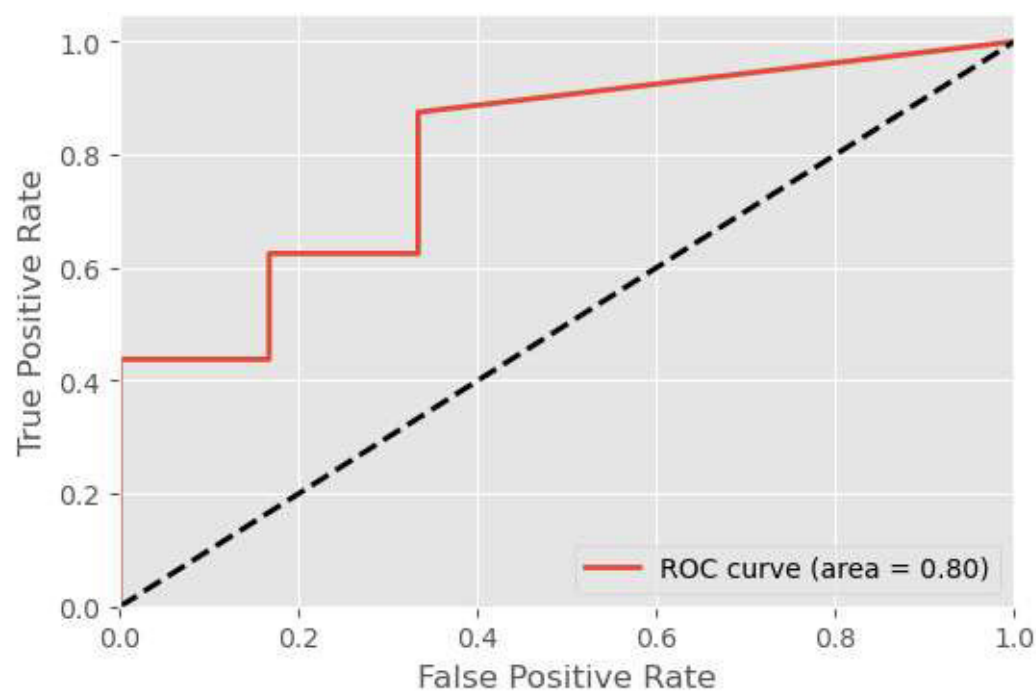


Figura 63. Matriz de Confusión – Test (CV = 10) del modelo KNN

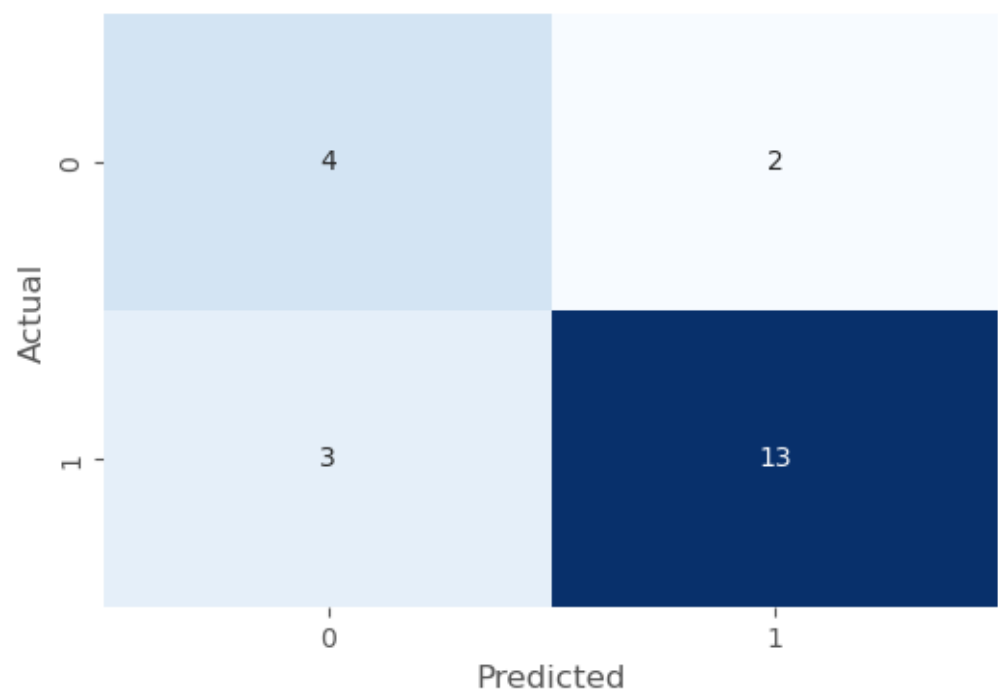


Figura 64. Curva ROC – Test (CV = 10) del modelo KNN

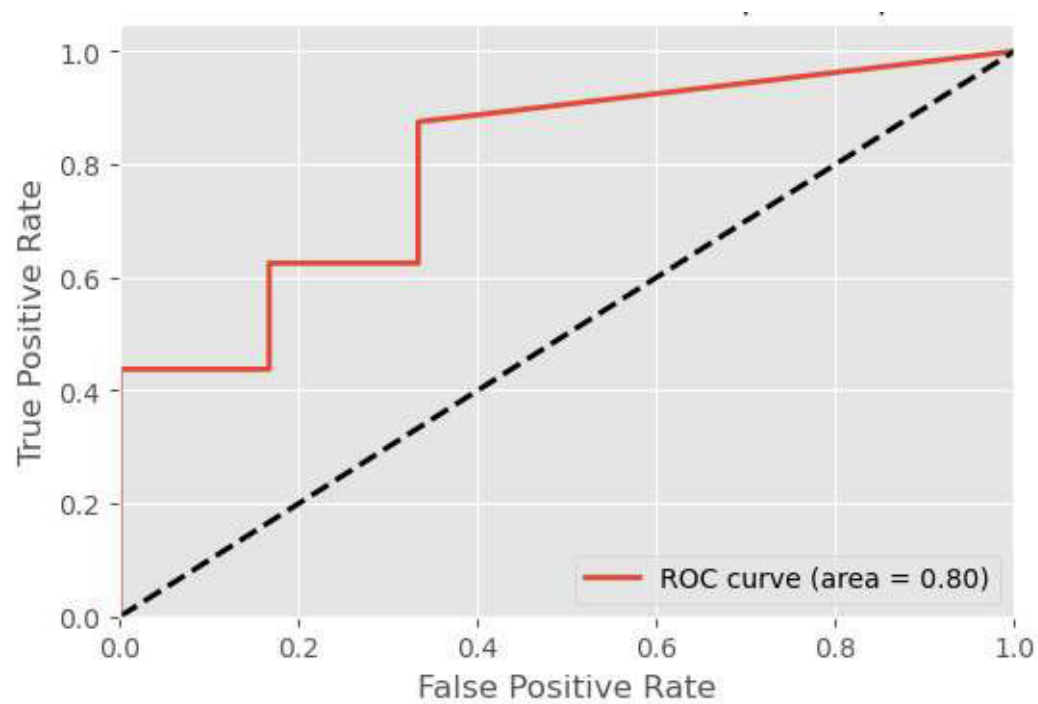


Figura 65. Matriz de Confusión – Test (CV = 11) del modelo KNN

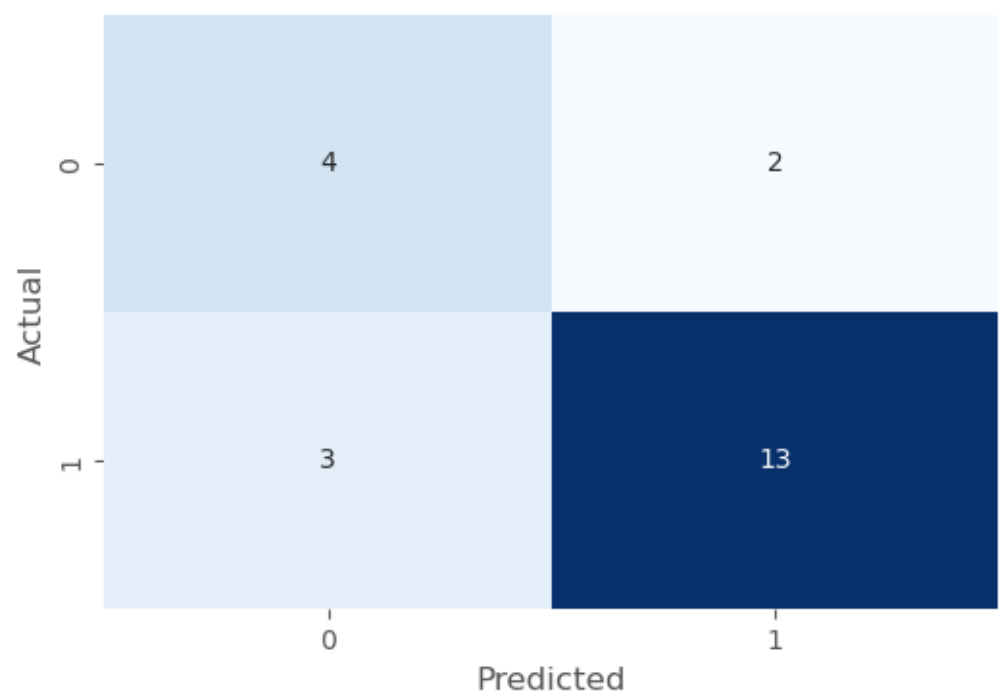
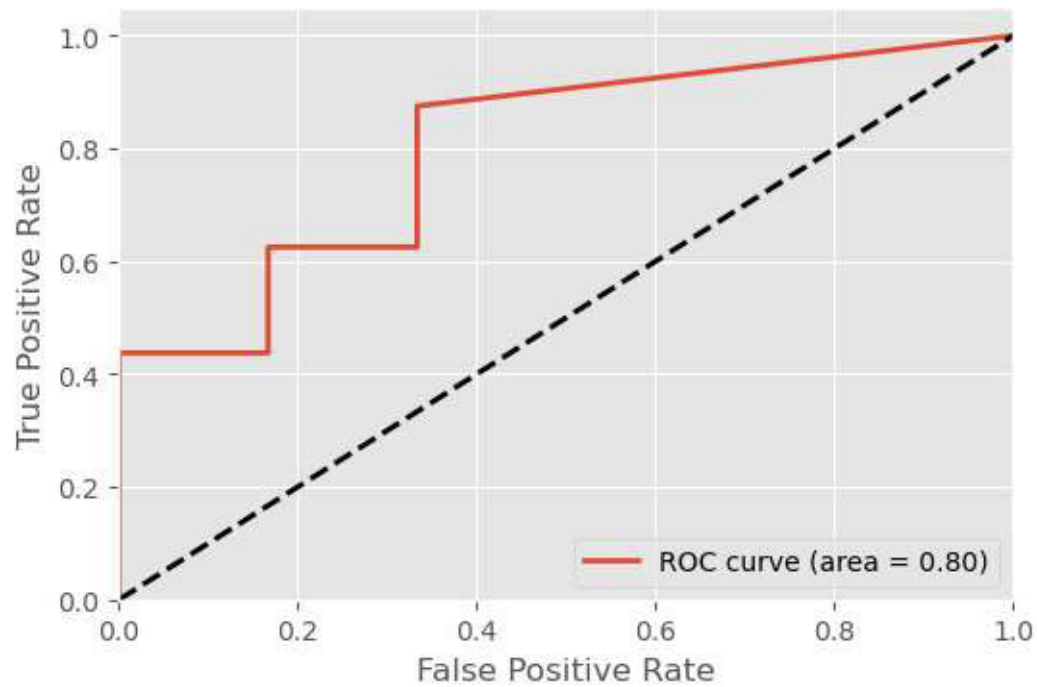


Figura 66. Curva ROC – Test (CV = 11) del modelo KNN



Anexo C. Random Forest

Figura 67. Importación de librerías y carga de datos

```
#librerías para el desarrollo del proyecto
import matplotlib.pyplot as plt #librería para graficos
from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches

import seaborn as sb

#librerías pandas para el manejo de los datos
import pandas as pd
import pandas as pq
import pandas as pf
import pandas as filtro_filas

#cargando los datos en el archivo excel
data = pd.read_excel("../data/data_pesquera.xlsx", index_col=None)
import numpy as np
#asociando a la librería de clasificación
from sklearn.tree import DecisionTreeClassifier
```

Figura 68. Identificación de la variable objetivo

```
Y = data['Accidental laboral']
Xya = data.dropna(["CALIFICACION"])

Y
0      0
1      1
2      0
3      1
4      0
...
105     0
106     1
107     1
108     0
109     1
Name: Accidental laboral, Length: 110, dtype: int64

# 0 este enfermo del corazon, 1 no este enfermo del corazon
print(data.groupby('Accidental laboral').size())

Accidental laboral
0      31
1      79
dtype: int64

from sklearn.metrics import roc_curve, auc
from sklearn import datasets
from sklearn.multiclass import OneVsRestClassifier
from sklearn.svm import LinearSVC
from sklearn.preprocessing import label_binarize
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

Figura 69. Particionamiento del conjunto de datos

```
#particionando la data en 0.8 para entrenar y 0.2 para validar
from sklearn.model_selection import train_test_split
X_trainset, X_testset, y_trainset, y_testset = train_test_split(X, Y, test_size=0.2, random_state=42)

print(X_trainset.shape, y_trainset.shape)
print(X_testset.shape, y_testset.shape)

(88, 7) (88,)
(22, 7) (22,)
```

Figura 70. Código de hiperparámetros y construcción del modelo Random Forest

```
# lista para almacenar los resultados
resultados = []

for i in range(2, 11):
    grid_search = GridSearchCV(RandomForestClassifier(),
                                parametros,
                                refit=True,
                                verbose=0,
                                cv=i)

    # Entrenar modelo
    grid_search.fit(X_trainset, y_trainset)

    # Predicciones
    pred_Y1 = grid_search.predict(X_testset)
    y_prob = grid_search.predict_proba(X_testset)[:, 1] # Probabilidades clase positiva (1)

    # Métricas (para dos clases)
    accuracy = accuracy_score(y_testset, pred_Y1)
    precision = precision_score(y_testset, pred_Y1)
    recall = recall_score(y_testset, pred_Y1)
    f1 = f1_score(y_testset, pred_Y1)
    roc_auc = roc_auc_score(y_testset, y_prob)
    matthews = matthews_corrcoef(y_testset, pred_Y1)

    resultados.append({
        "cv": i,
        "Accuracy": accuracy,
        "Precision": precision,
        "Recall": recall,
        "F1 score": f1,
        "ROC AUC": roc_auc,
        "Matthews": matthews,
        "Predicciones": pred_Y1
    })
```

Figura 71. Código para el cálculo de métricas de evaluación

```
resultados.append({
    "cv": i,
    "Accuracy": accuracy,
    "Precision": precision,
    "Recall": recall,
    "F1 score": f1,
    "ROC AUC": roc_auc,
    "Matthews": matthews,
    "Predicciones": pred_YI
})
```

Figura 72. Código para generar la matriz de confusión y curva ROC

```
# Matriz de confusión y curva ROC
for resultado in resultados:
    # Matriz de confusión
    plt.figure(figsize=(6, 4))
    cmf_matrix = confusion_matrix(y_testset, resultado['Predicciones'])
    sb.heatmap(cmf_matrix, annot=True, fmt='d', cmap='Blues', cbar=False)
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Decision Tree Confusion Matrix (cv={resultado["cv"]})')
    plt.show()

    # Curva ROC
    plt.figure(figsize=(6, 4))
    y_prob = grid_search.predict_proba(X_testset)[:, 1]
    fpr, tpr, _ = roc_curve(y_testset, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.plot(fpr, tpr, lw=2, label=f'ROC curve (area = {roc_auc:.2f})')
    plt.plot([0, 1], [0, 1], 'k--', lw=2)
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Decision Tree ROC Curve (cv={resultado["cv"]})')
    plt.legend(loc='lower right')
    plt.show()
```

Figura 73. Matriz de Confusión – Test (CV = 2) del modelo Random Forest

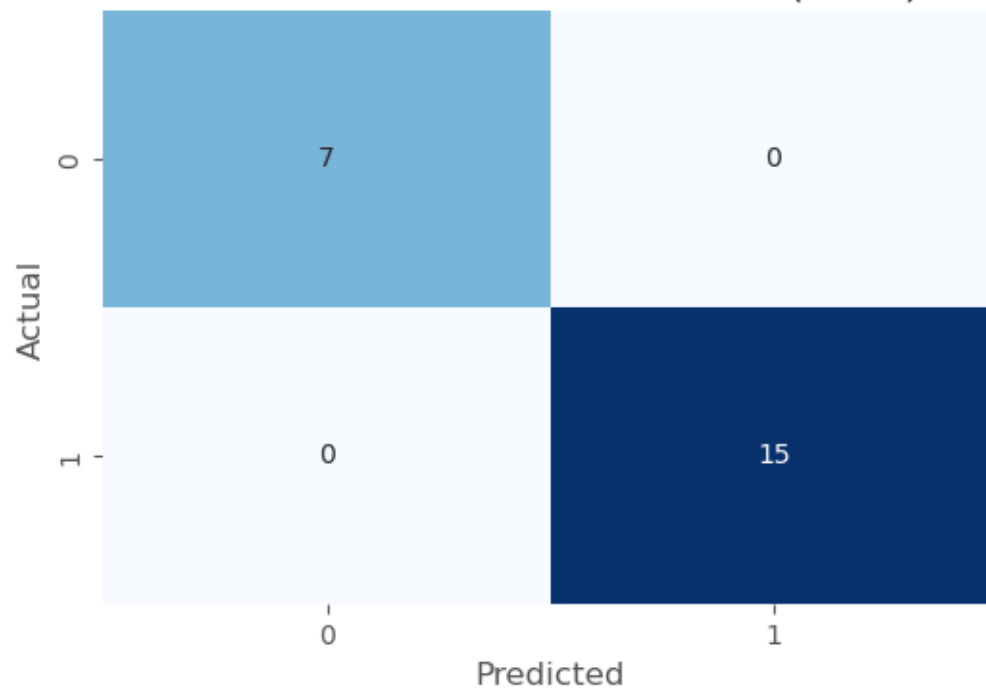


Figura 74. Curva ROC – Test (CV = 2) del modelo Random Forest

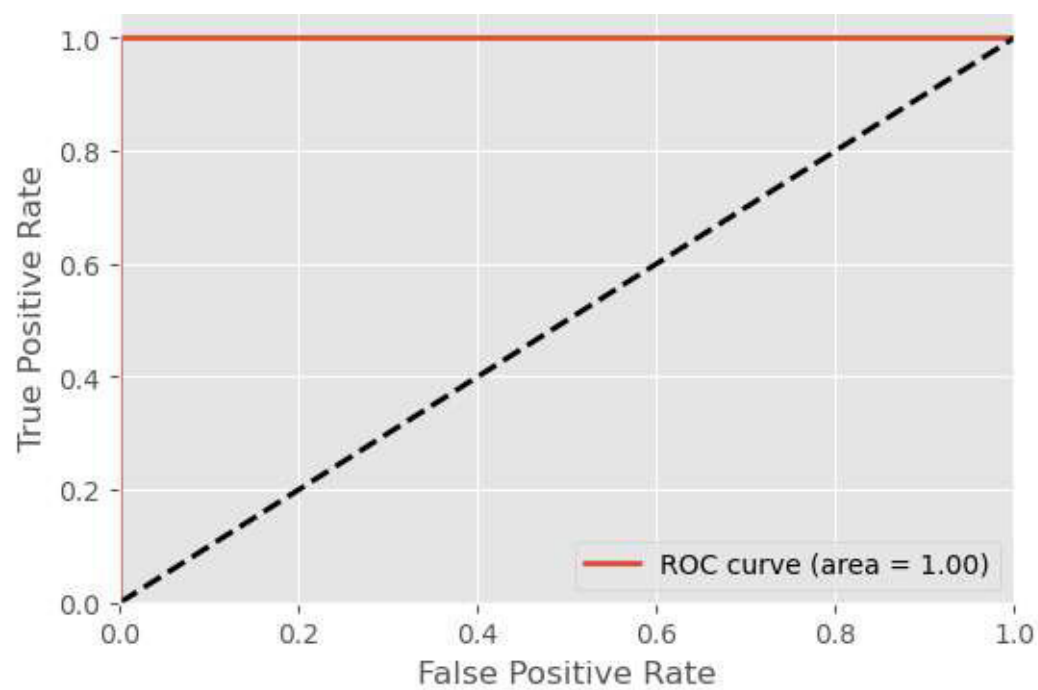


Figura 75. Matriz de Confusión – Test (CV = 3) del modelo Random Forest

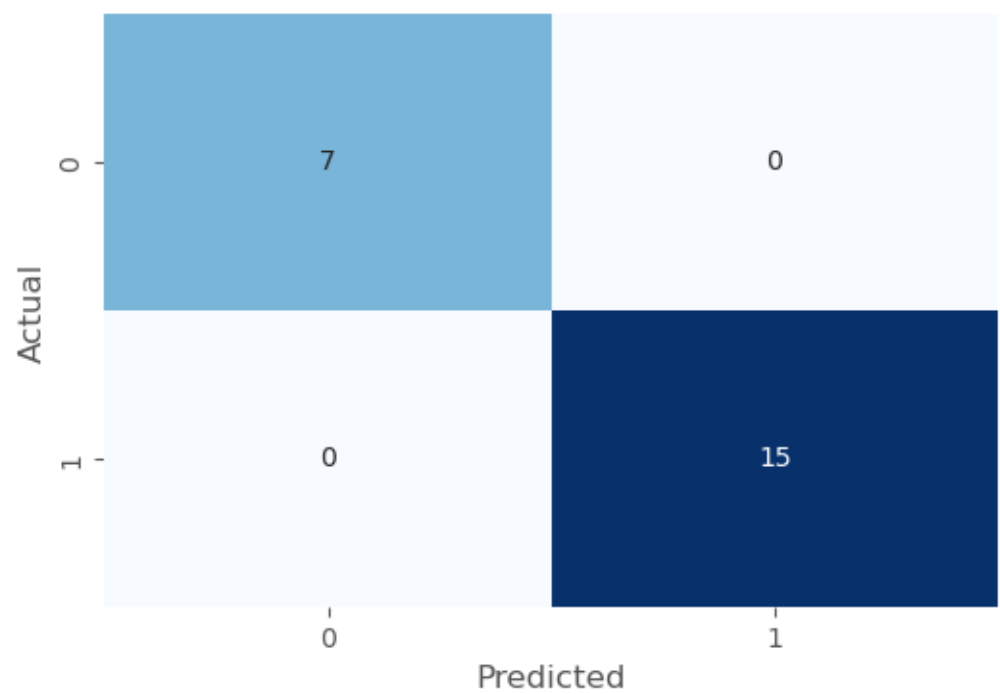


Figura 76. Curva ROC – Test (CV = 3) del modelo Random Forest

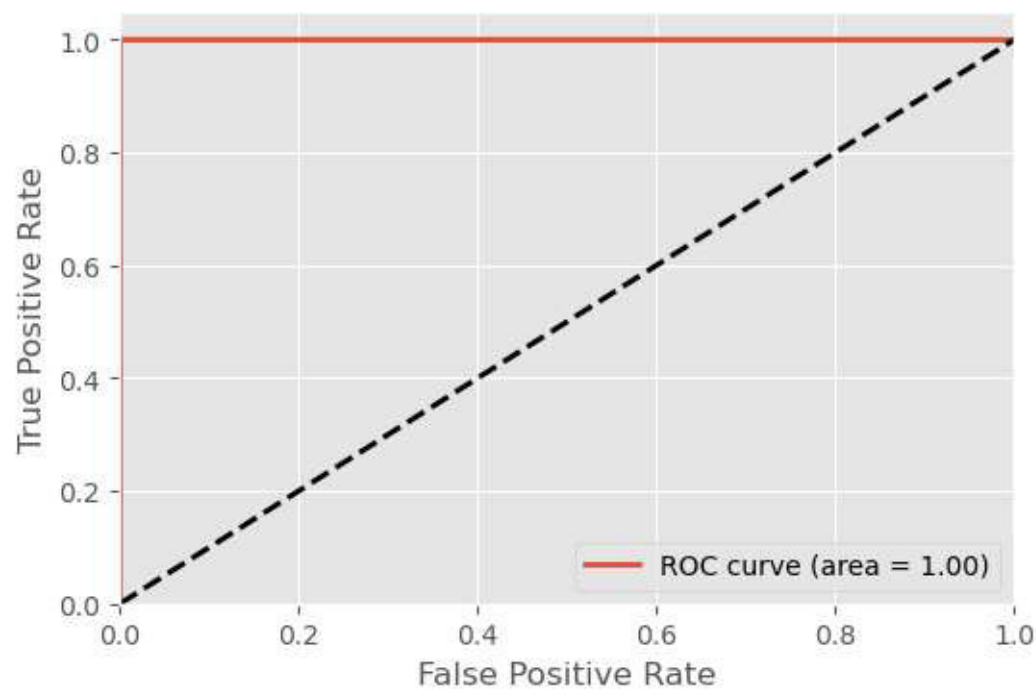


Figura 77. Matriz de Confusión – Test (CV = 4) del modelo Random Forest

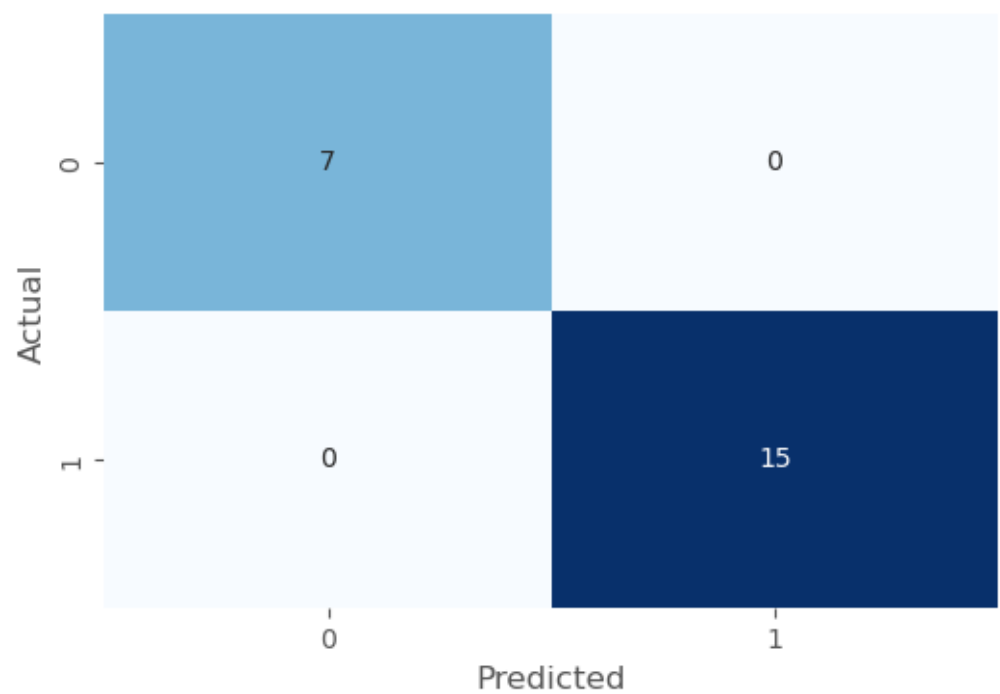


Figura 78. Curva ROC – Test (CV = 4) del modelo Random Forest

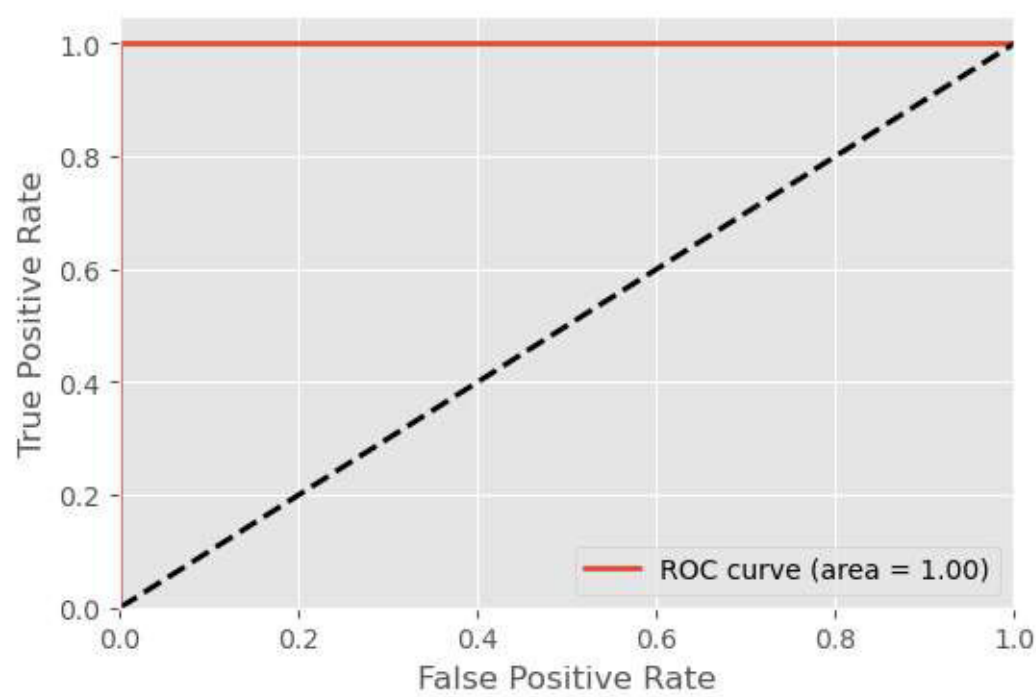


Figura 79. Matriz de Confusión – Test (CV = 5) del modelo Random Forest

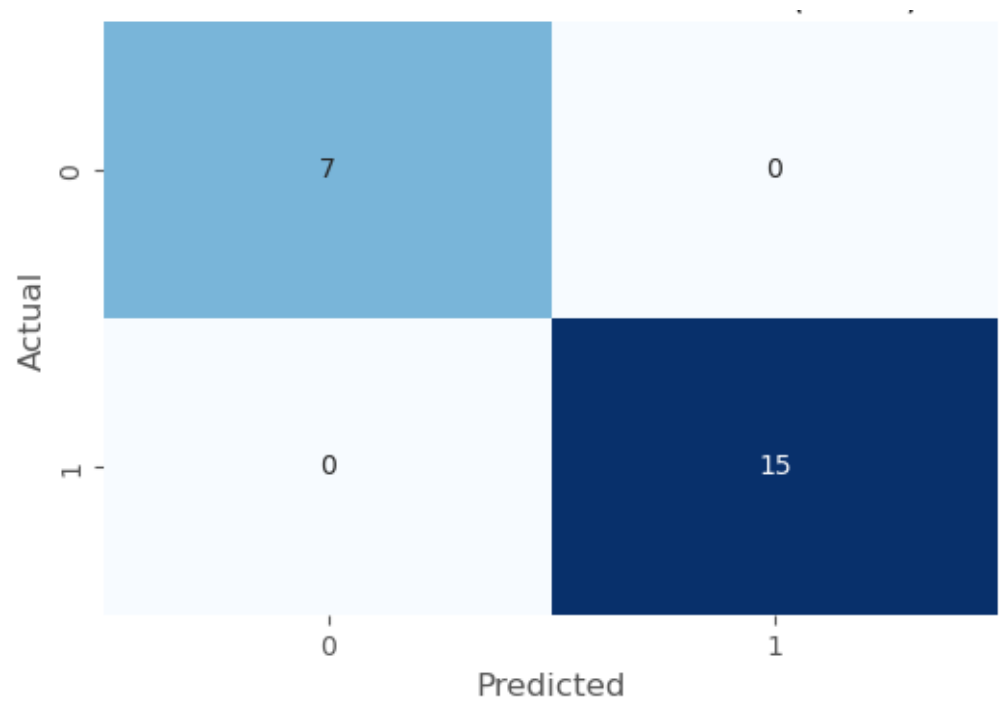


Figura 80. Curva ROC – Test (CV = 5) del modelo Random Forest

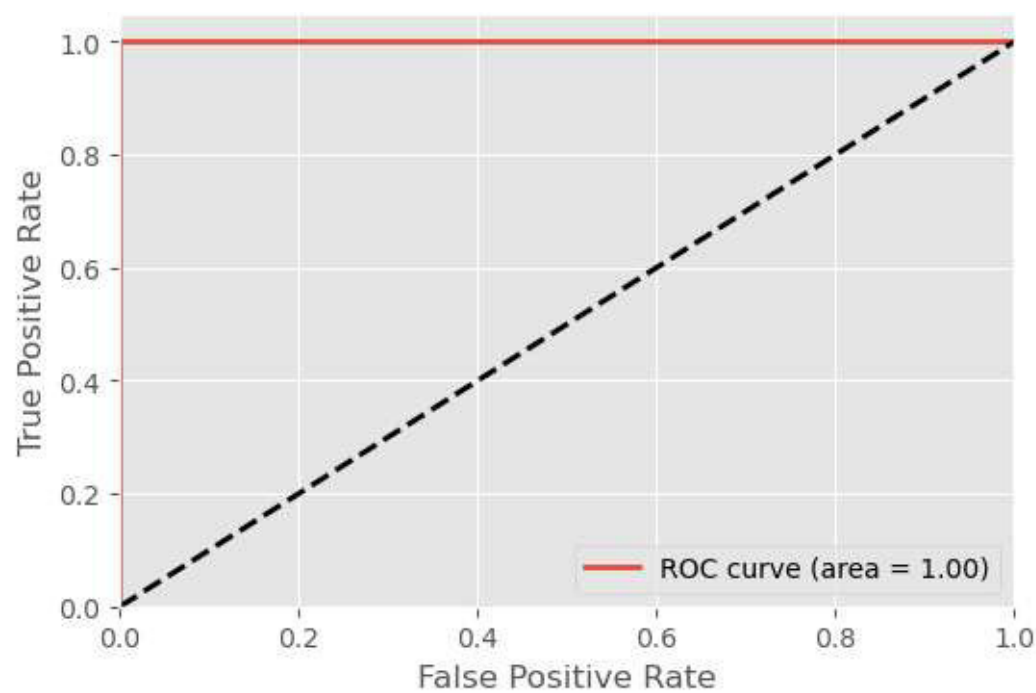


Figura 81. Matriz de Confusión – Test (CV = 6) del modelo Random Forest

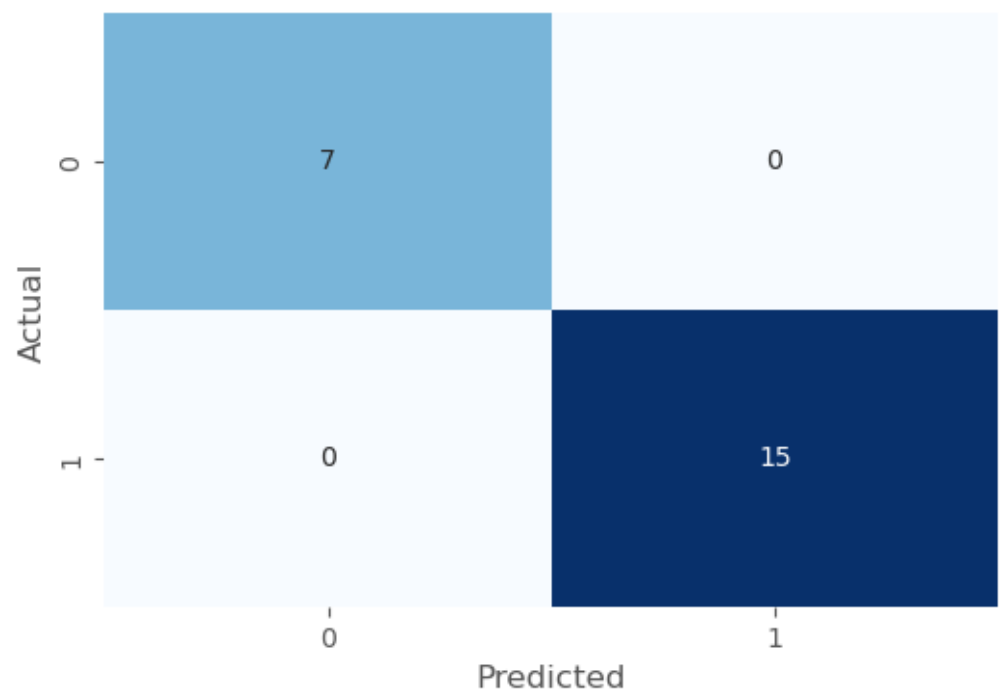


Figura 82. Curva ROC – Test (CV = 6) del modelo Random Forest

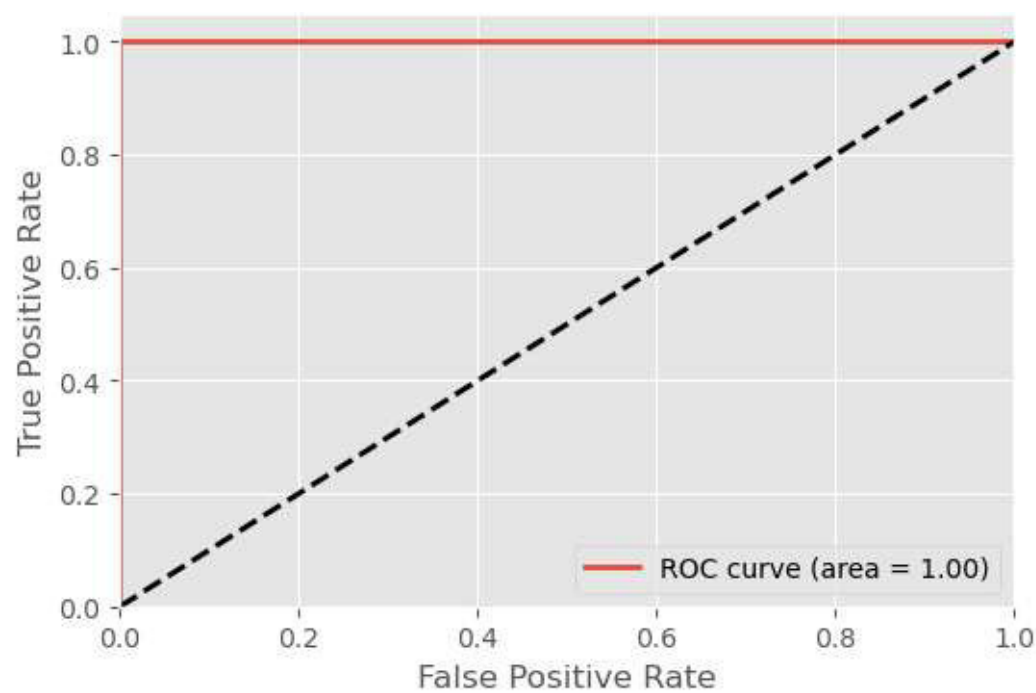


Figura 83. Matriz de Confusión – Test (CV = 7) del modelo Random Forest

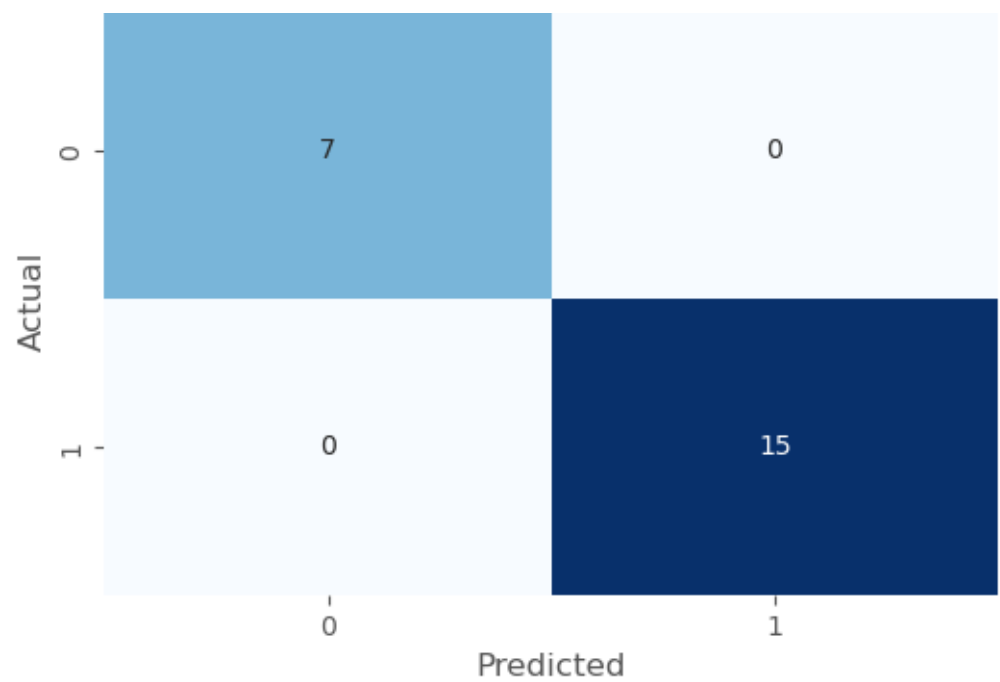


Figura 84. Curva ROC – Test (CV = 7) del modelo Random Forest

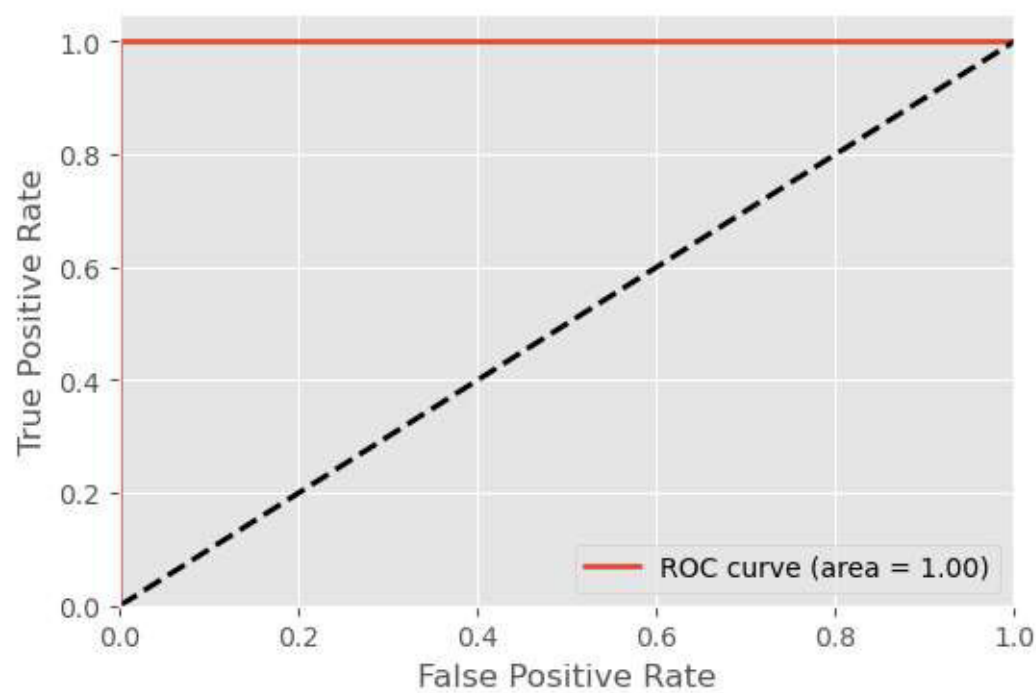


Figura 85. Matriz de Confusión – Test (CV = 8) del modelo Random Forest

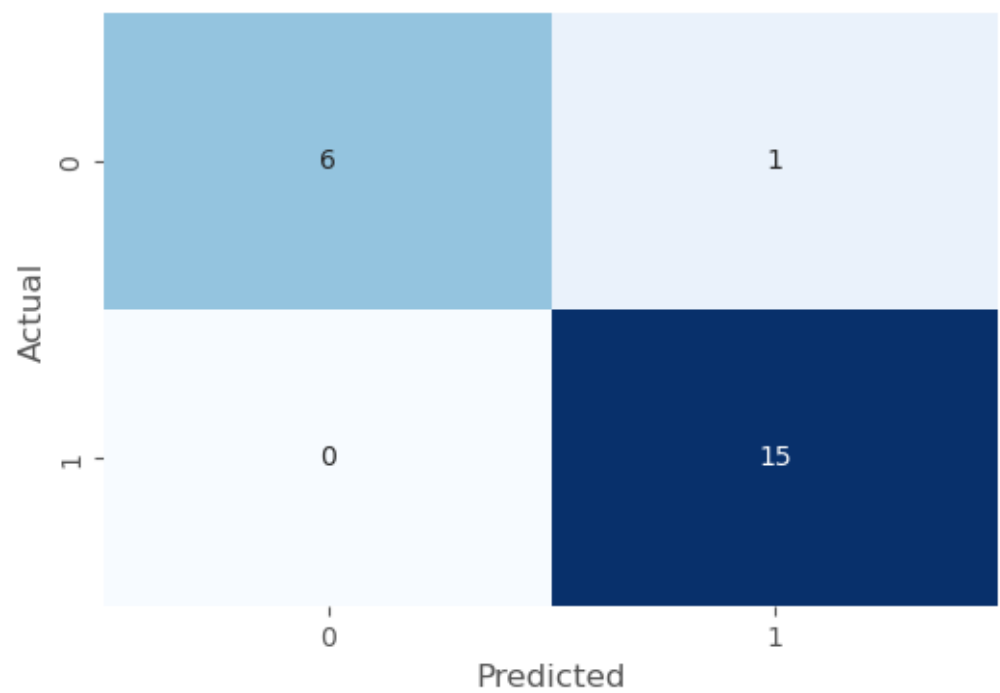


Figura 86. Curva ROC – Test (CV = 8) del modelo Random Forest

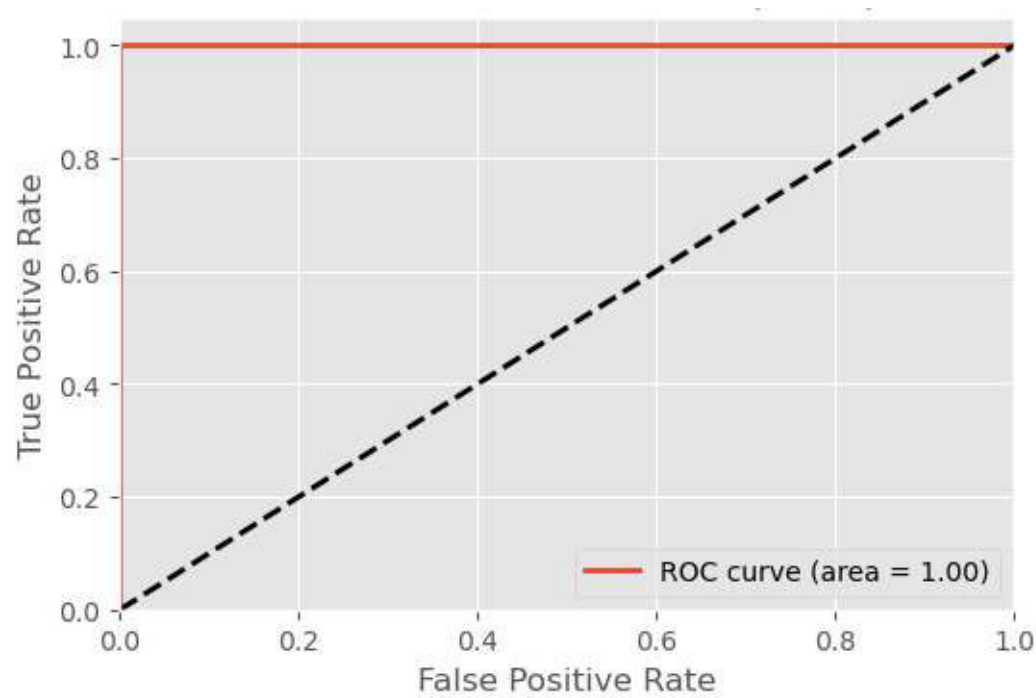


Figura 87. Matriz de Confusión – Test (CV = 9) del modelo Random Forest

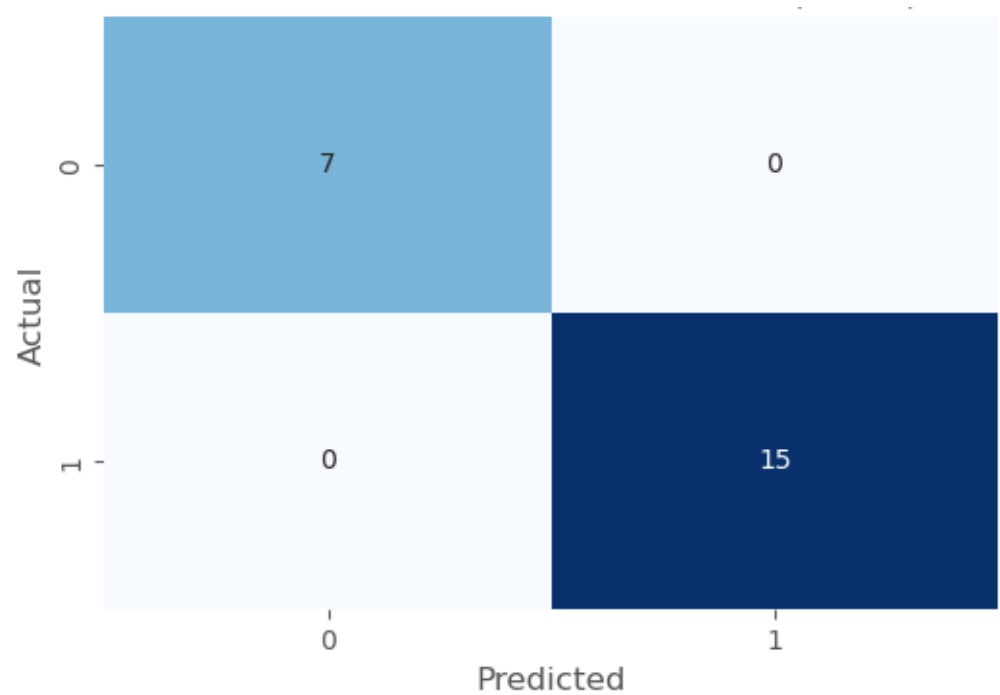


Figura 88. Curva ROC – Test (CV = 9) del modelo Random Forest

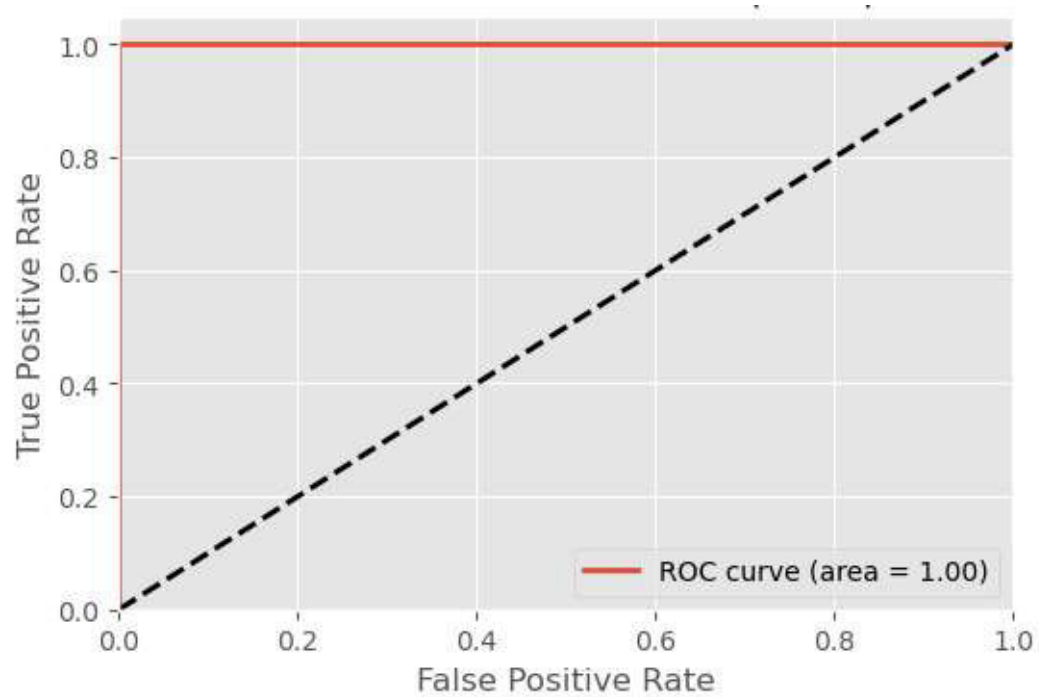


Figura 89. Matriz de Confusión – Test (CV = 10) del modelo Random Forest

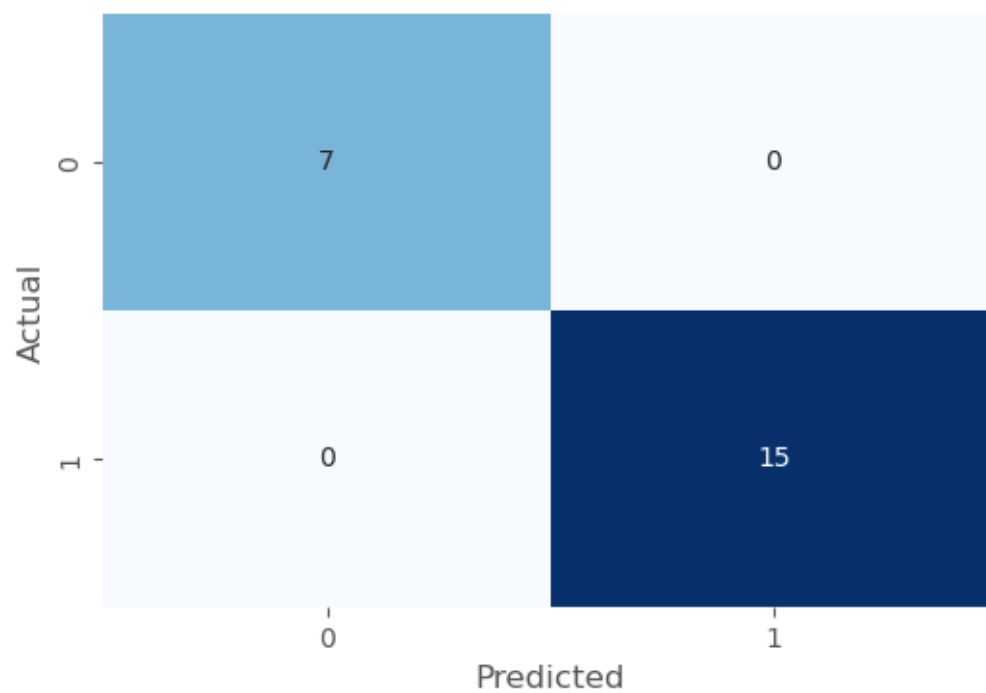


Figura 90. Curva ROC – Test (CV = 10) del modelo Random Forest

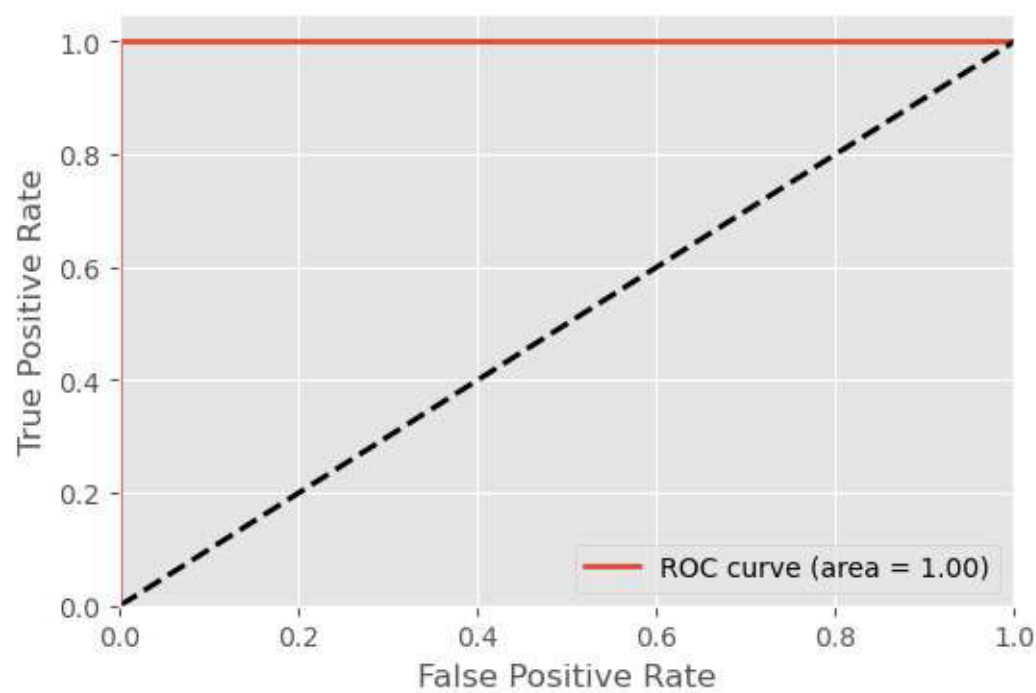


Figura 91. Matriz de Confusión – Test (CV = 11) del modelo Random Forest

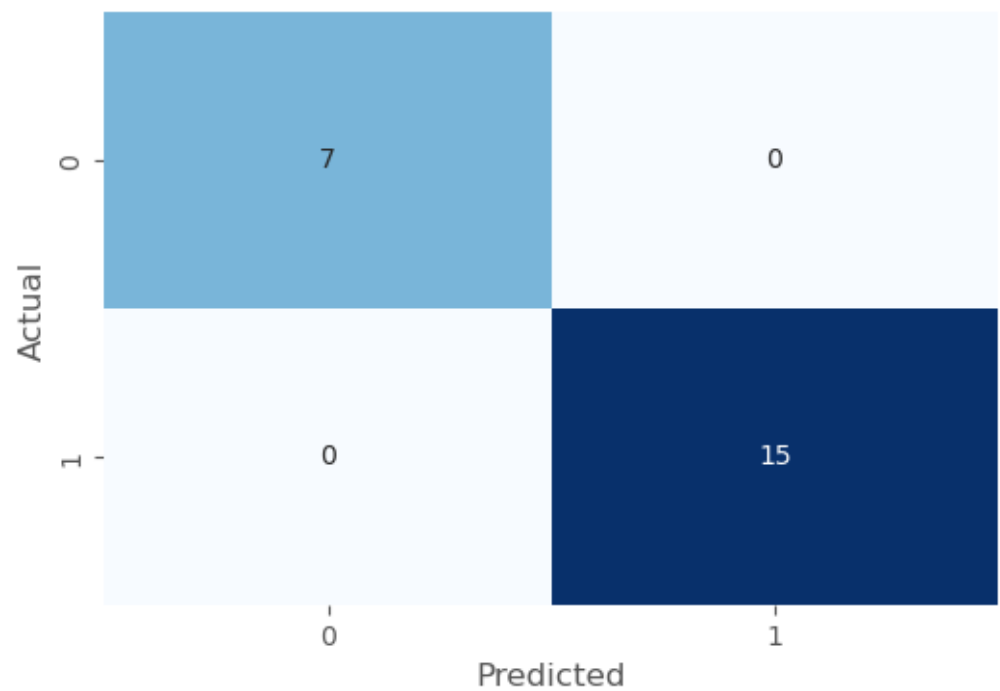
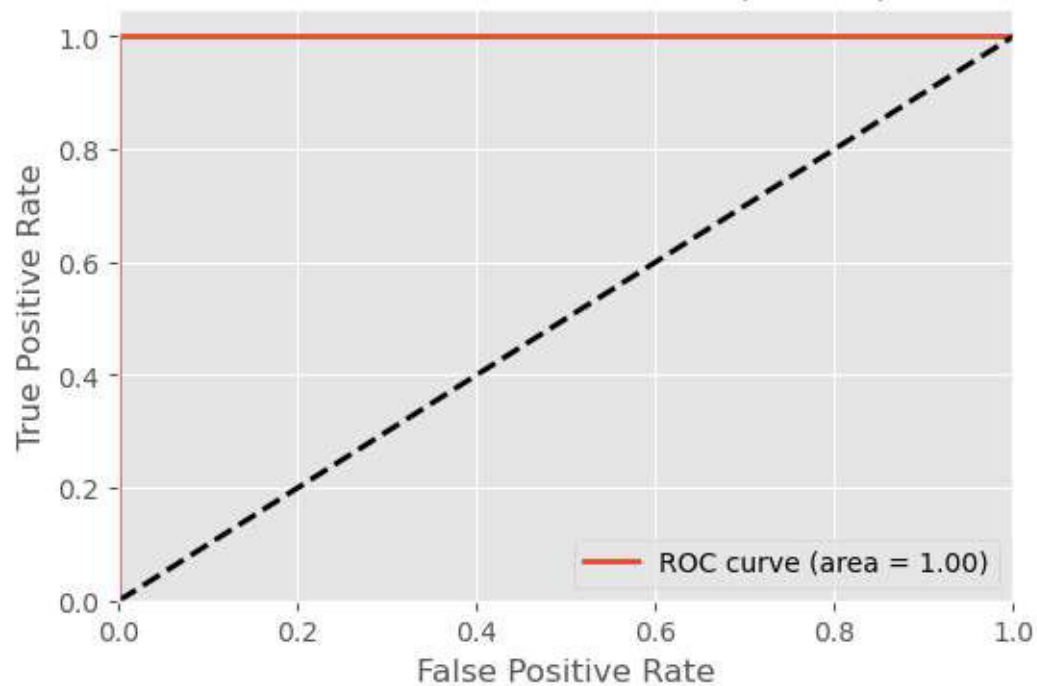


Figura 92. Curva ROC – Test (CV = 11) del modelo Random Forest



Anexo D. SVM

Figura 93. Importación de librerías y carga de datos

```
#librerías para el desarrollo del proyecto
import matplotlib.pyplot as plt #librería para graficas
from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches

import seaborn as sb

#librerías pandas para el manejo de los datos
import pandas as pd
import pandas as pq
import pandas as pf
import pandas as filtro_filas

#enlazando los datos en el archivo excel
data = pd.read_excel("../data/data_pesquero.xlsx", index_col=None)

# '../data/diametros.csv'

import numpy as np
#invocando a la librería de clasificación
from sklearn.tree import DecisionTreeClassifier
```

Figura 94. Identificación de la variable objetivo

```
print(data.groupby('Accidentalaboral').size())
```

```
Accidentalaboral
0    31
1    79
dtype: int64
```

```
from sklearn.metrics import roc_curve, auc
from sklearn import datasets
from sklearn.multiclass import OneVsRestClassifier
from sklearn.svm import LinearSVC
from sklearn.preprocessing import label_binarize
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

Figura 95. *Particionamiento del conjunto de datos*

```
1 #particionando la data en 0.8 para entrenar y 0.2 para validar
2 from sklearn.model_selection import train_test_split
3 X_trainset, X_testset, y_trainset, y_testset = train_test_split(X,Y, test_size=0.2, random_state=3)
4
5 print(X_trainset.shape, y_trainset.shape)
6 print (X_testset.shape, y_testset.shape)
7
8 (88, 7) (88,)
9 (22, 7) (22,)
```

Figura 96. *Código de hiperparámetros y construcción del modelo SVM*

```
1 # lista para almacenar los resultados
2 resultados = []
3
4 for i in range(2, 32):
5     grid_search = GridSearchCV(SVC(probability=True),
6                               parametros,
7                               refit=True,
8                               verbose=0,
9                               cv=i)
10
11     # Entrenar modelo
12     grid_search.fit(X_trainset, y_trainset)
13
14     # Predicciones
15     pred_Y1 = grid_search.predict(X_testset)
16     y_prob = grid_search.predict_proba(X_testset)[:, 1] # Probabilidades clase positiva (1)
17
18     # Métricas (para dos clases)
19     accuracy = accuracy_score(y_testset, pred_Y1)
20     precision = precision_score(y_testset, pred_Y1)
21     recall = recall_score(y_testset, pred_Y1)
22     f1 = f1_score(y_testset, pred_Y1)
23     roc_auc = roc_auc_score(y_testset, y_prob)
24     matthews = matthews_corrcoef(y_testset, pred_Y1)
25
26     resultados.append({
27         "cv": i,
28         "Accuracy": accuracy,
29         "Precision": precision,
30         "Recall": recall,
31         "F1 score": f1,
32         "ROC AUC": roc_auc,
33         "Matthews": matthews,
34         "Predicciones": pred_Y1
35     })
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

Figura 97. *Código para el cálculo de métricas de evaluación*

```
1 resultados.append({
2     "cv": i,
3     "Accuracy": accuracy,
4     "Precision": precision,
5     "Recall": recall,
6     "F1 score": f1,
7     "ROC AUC": roc_auc,
8     "Matthews": matthews,
9     "Predicciones": pred_Y1
10 })
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61
62
63
64
65
66
67
68
69
70
71
72
73
74
75
76
77
78
79
80
81
82
83
84
85
86
87
88
89
90
91
92
93
94
95
96
97
98
99
```

Figura 98. Código para generar la matriz de confusión y curva ROC

```
# Matriz de confusión y curva ROC
for resultado in resultados:
    # Matriz de confusión
    plt.figure(figsize=(6, 4))
    cmf_matrix = confusion_matrix(y_testset, resultado['Predicciones'])
    sb.heatmap(cmf_matrix, annot=True, fmt='d', cmap='Blues', cbar=False)
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Decision Tree Confusion Matrix (cv={resultado["cv"]})')
    plt.show()

    # Curva ROC
    plt.figure(figsize=(6, 4))
    y_prob = grid_search.predict_proba(X_testset)[:, 1]
    fpr, tpr, _ = roc_curve(y_testset, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.plot(fpr, tpr, lw=2, label=f'ROC curve (area = {roc_auc:0.2f})')
    plt.plot([0, 1], [0, 1], 'k--', lw=2)
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Decision Tree ROC Curve (cv={resultado["cv"]})')
    plt.legend(loc='lower right')
    plt.show()
```

Figura 99. Matriz de Confusión – Test (CV = 2) del modelo SVM

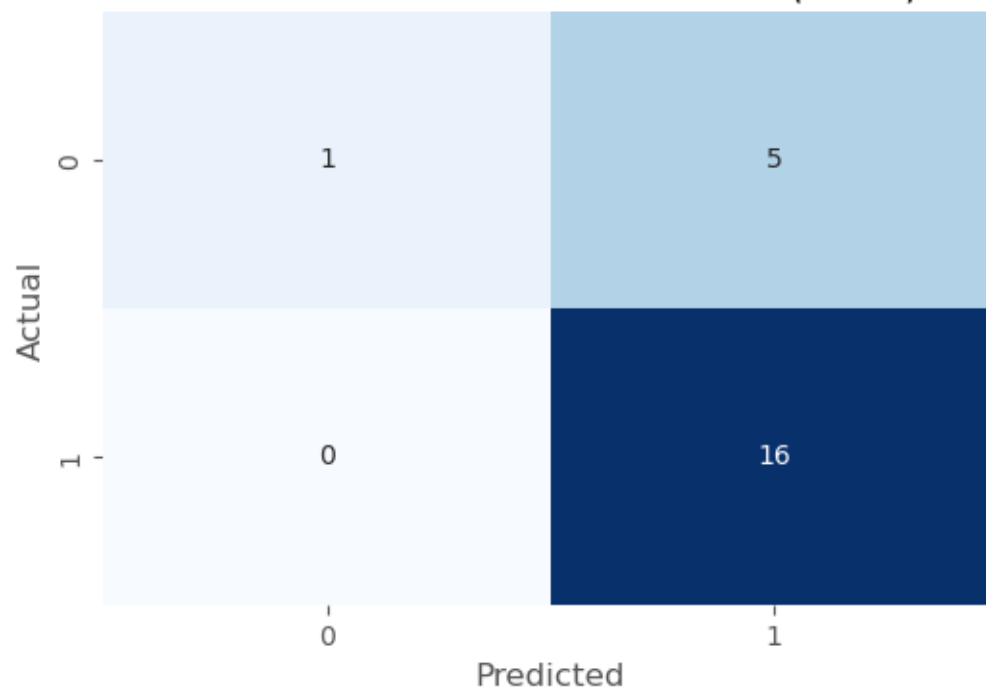


Figura 100. Curva ROC – Test (CV = 2) del modelo SVM

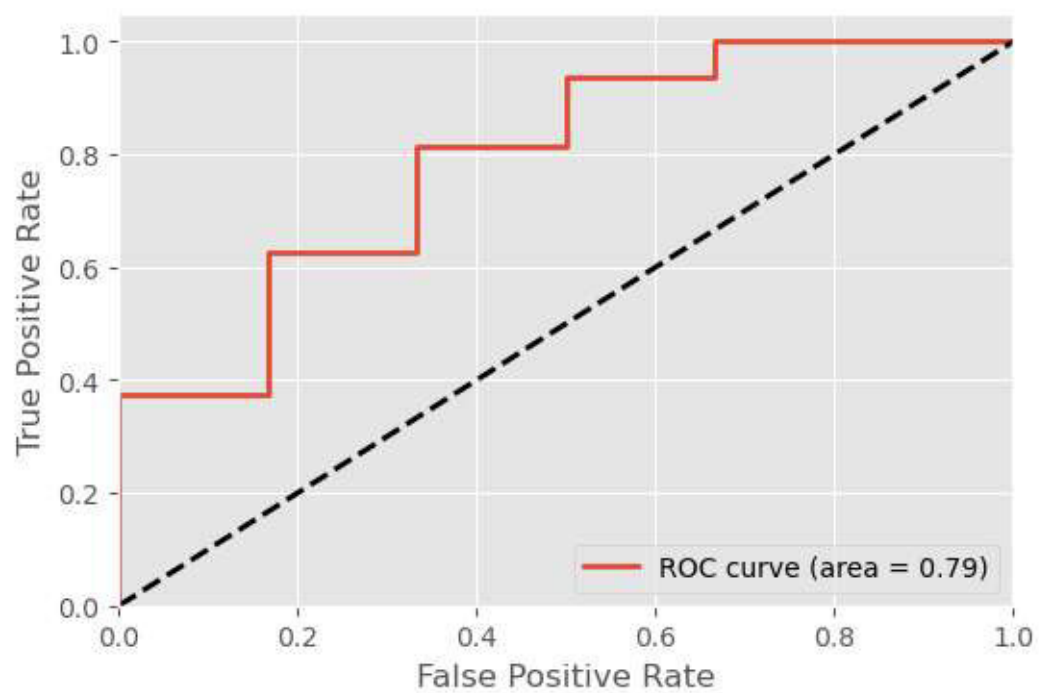


Figura 101. Matriz de Confusión – Test (CV = 3) del modelo SVM

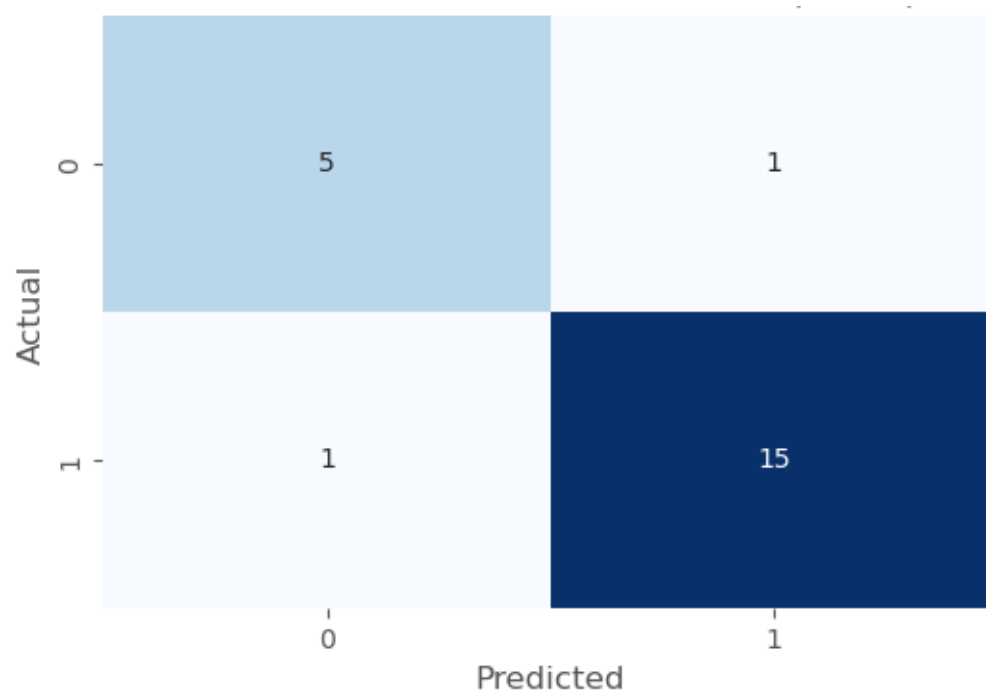


Figura 102. Curva ROC – Test (CV = 3) del modelo SVM

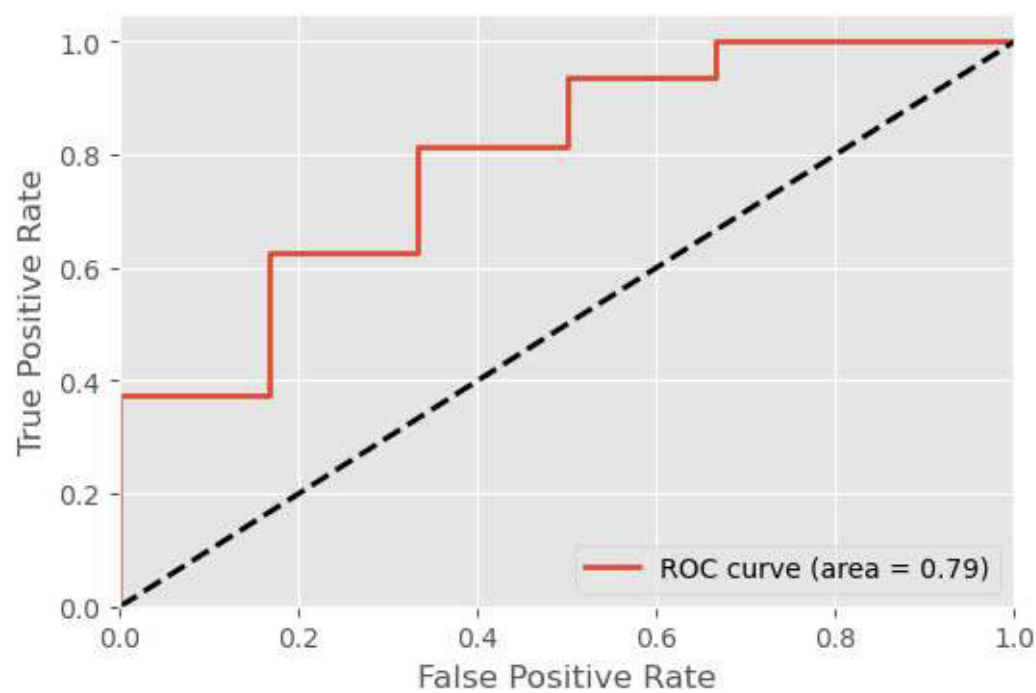


Figura 103. Matriz de Confusión – Test (CV = 4) del modelo SVM

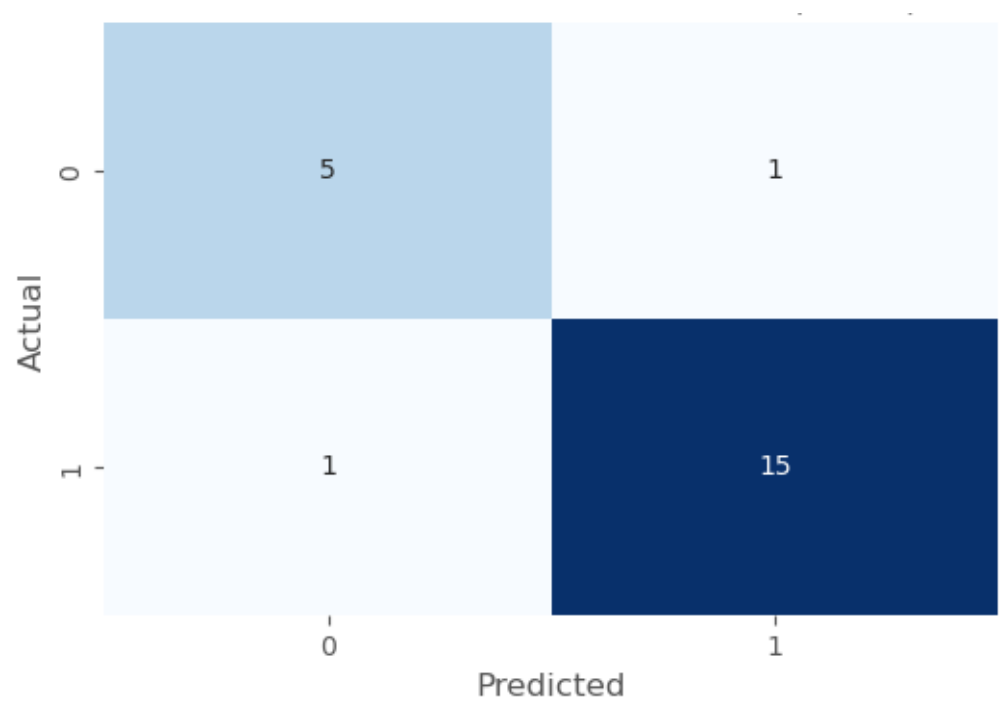


Figura 104. Curva ROC – Test (CV = 4) del modelo SVM

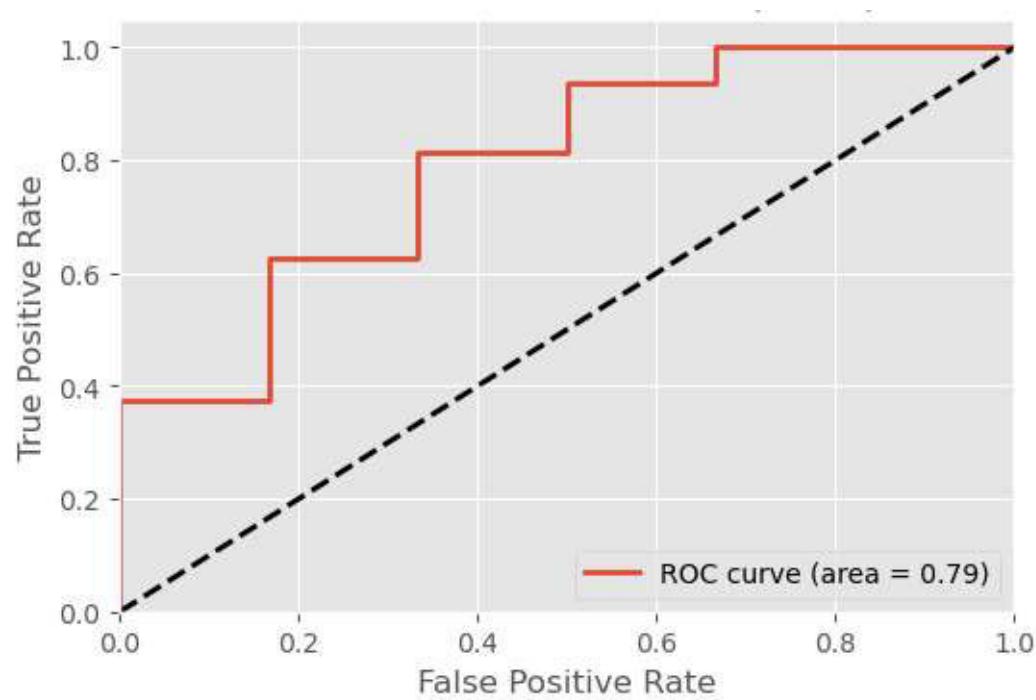


Figura 105. Matriz de Confusión – Test (CV = 5) del modelo SVM

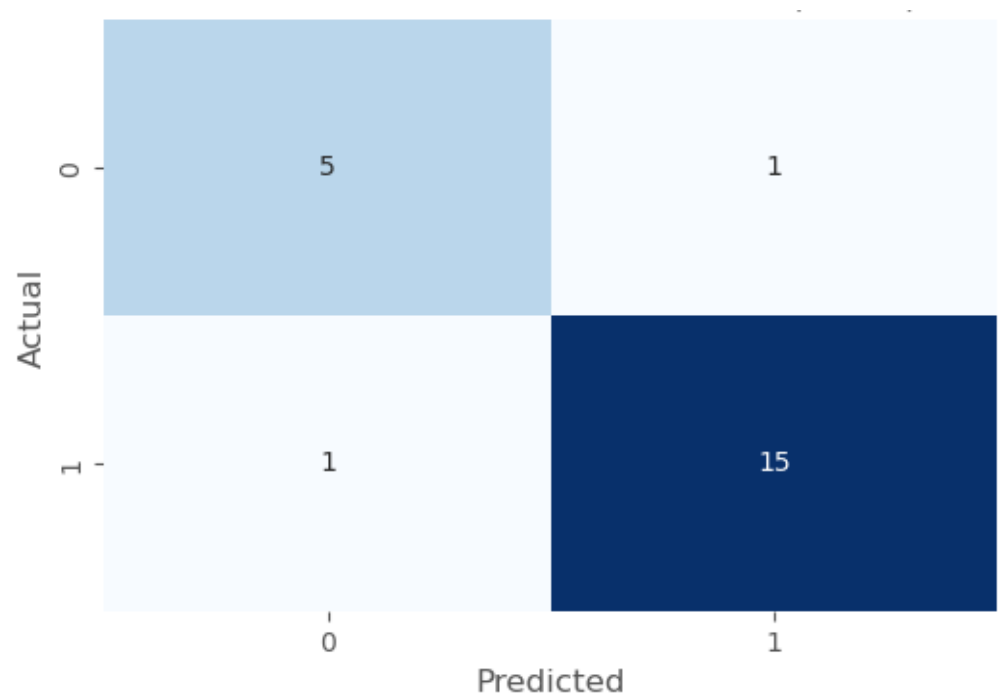


Figura 106. Curva ROC – Test (CV = 5) del modelo SVM

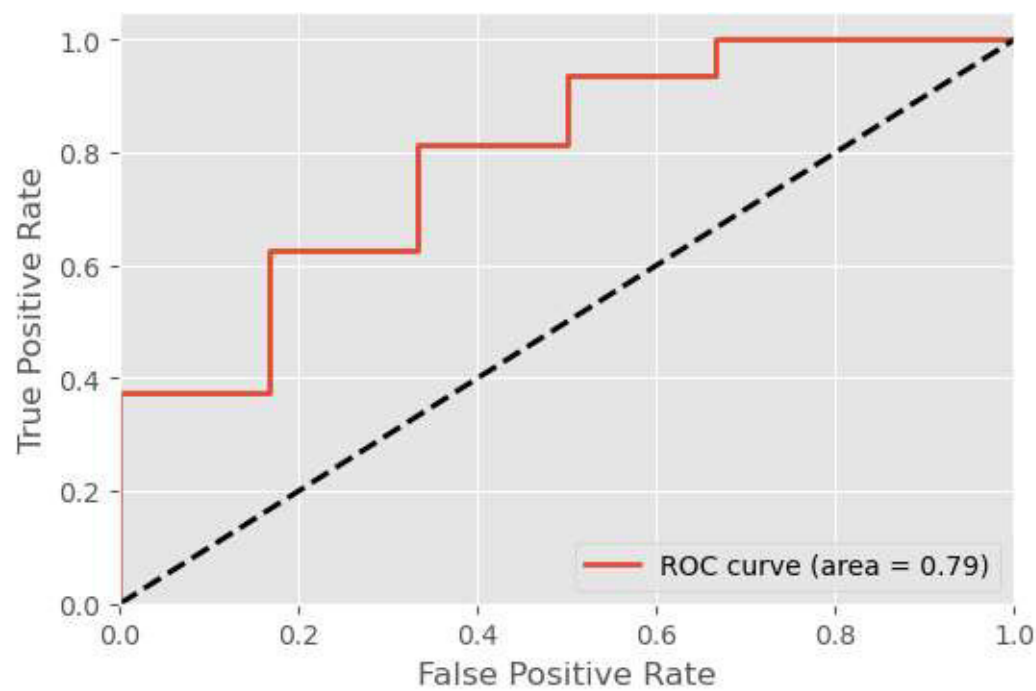


Figura 107. Matriz de Confusión – Test (CV = 6) del modelo SVM

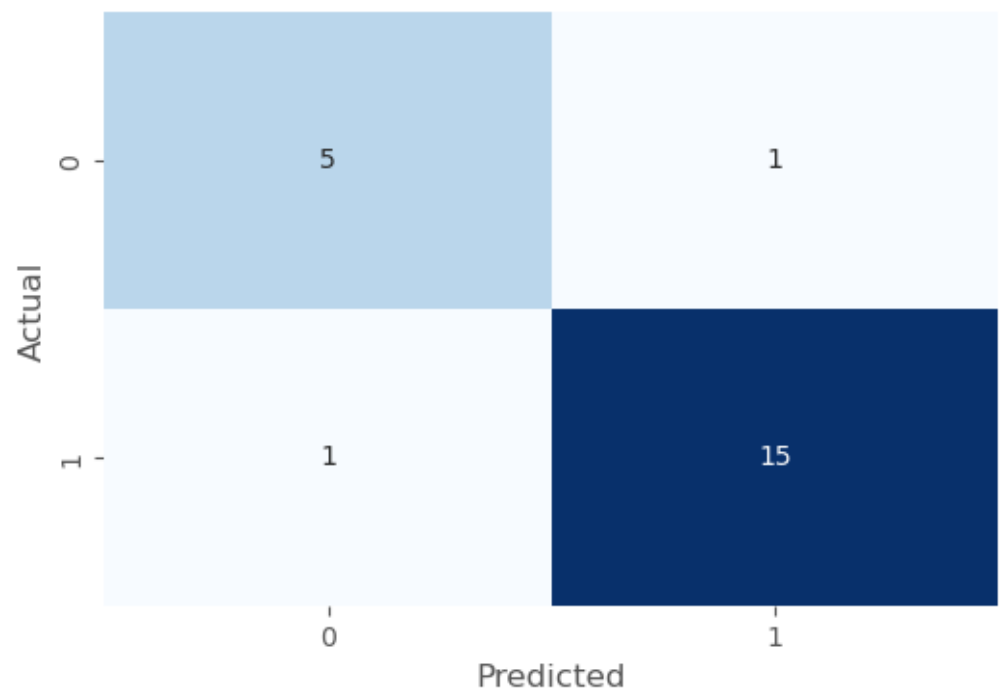


Figura 108. Curva ROC – Test (CV = 6) del modelo SVM

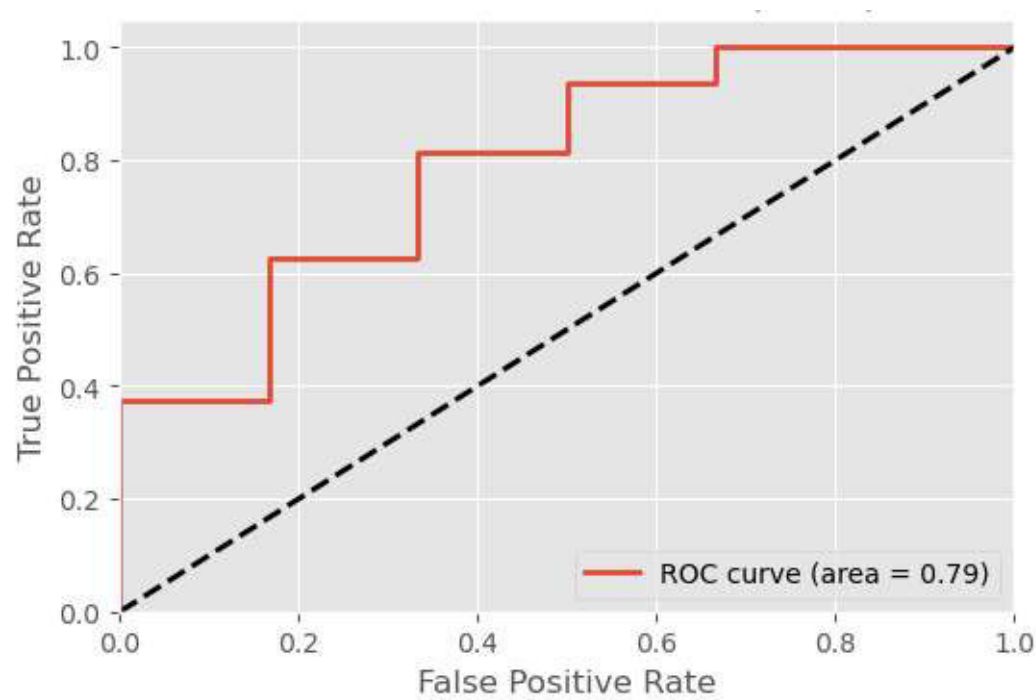


Figura 109. Matriz de Confusión – Test (CV = 7) del modelo SVM

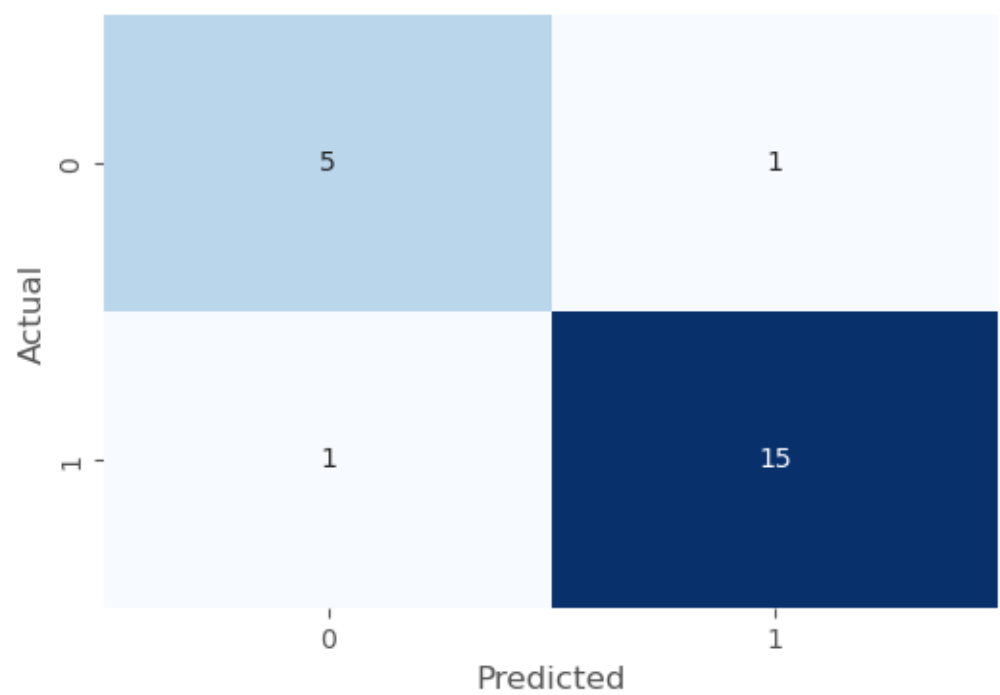


Figura 110. Curva ROC – Test (CV = 7) del modelo SVM

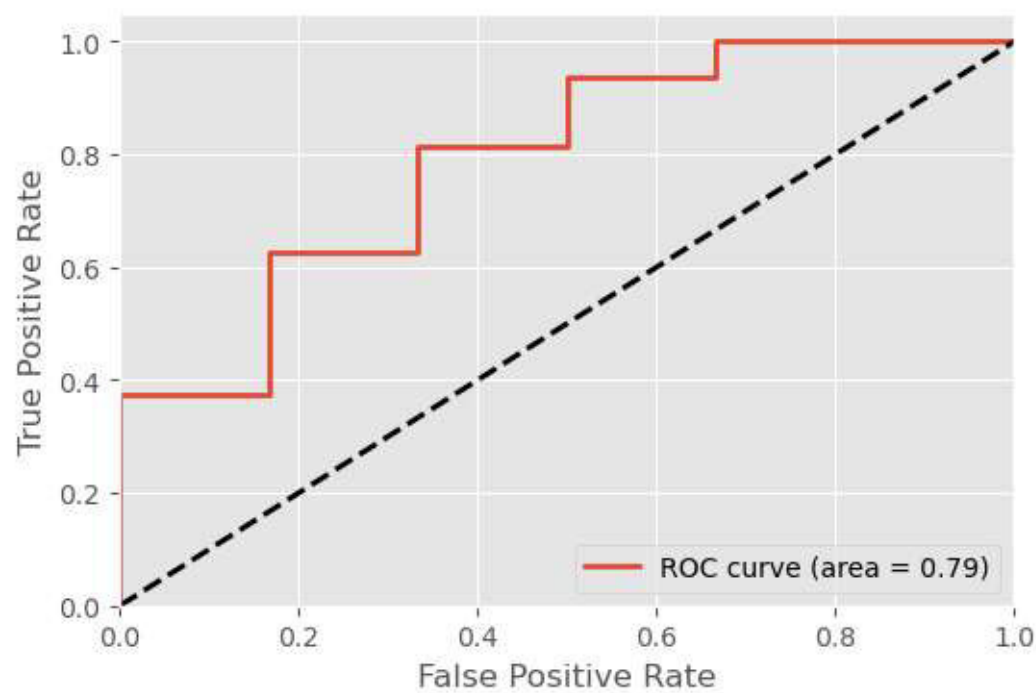


Figura 111. Matriz de Confusión – Test (CV = 8) del modelo SVM

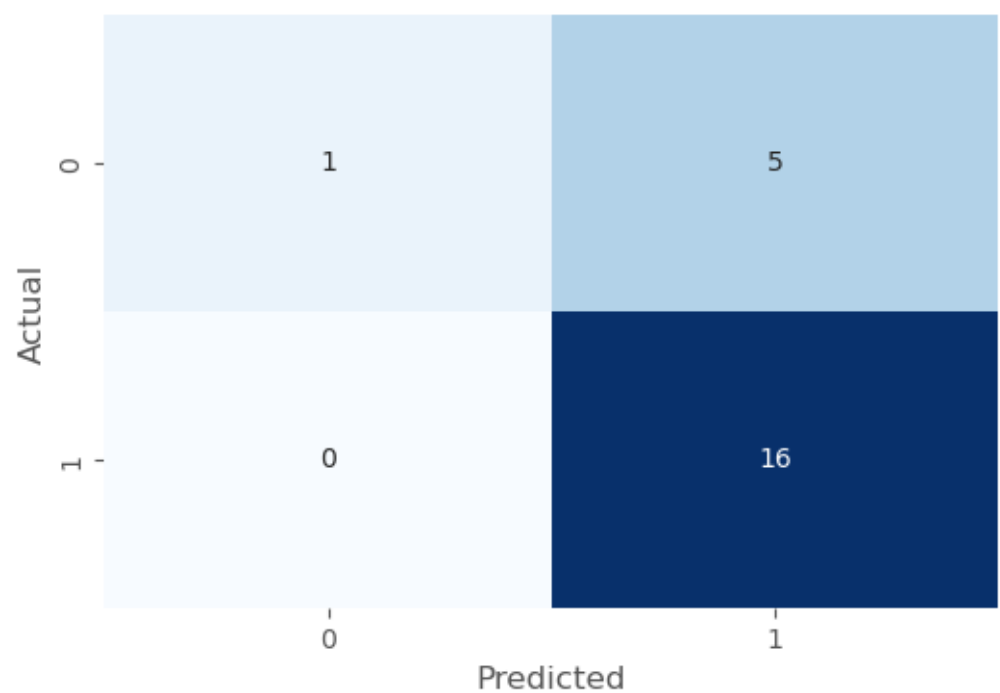


Figura 112. Curva ROC – Test (CV = 8) del modelo SVM

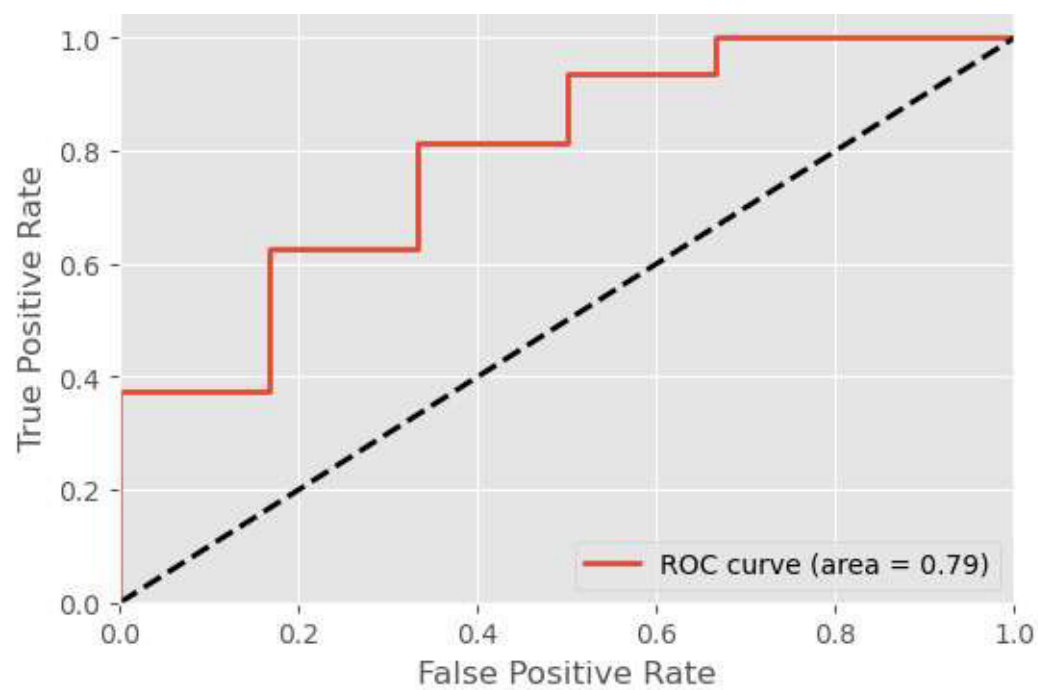


Figura 113. Matriz de Confusión – Test (CV = 9) del modelo SVM

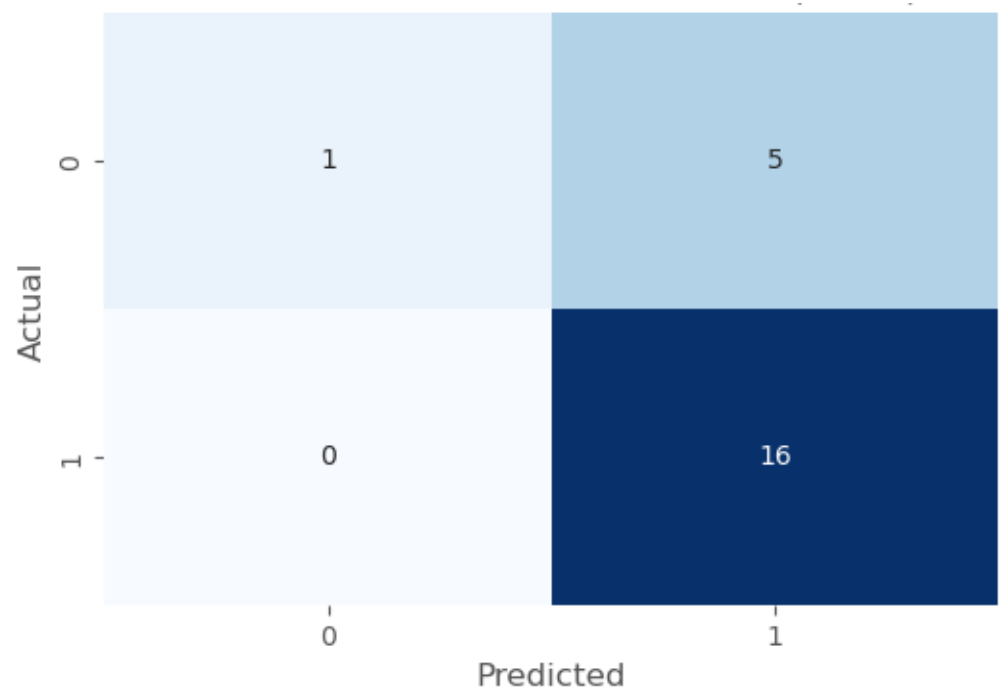


Figura 114. Curva ROC – Test (CV = 9) del modelo SVM

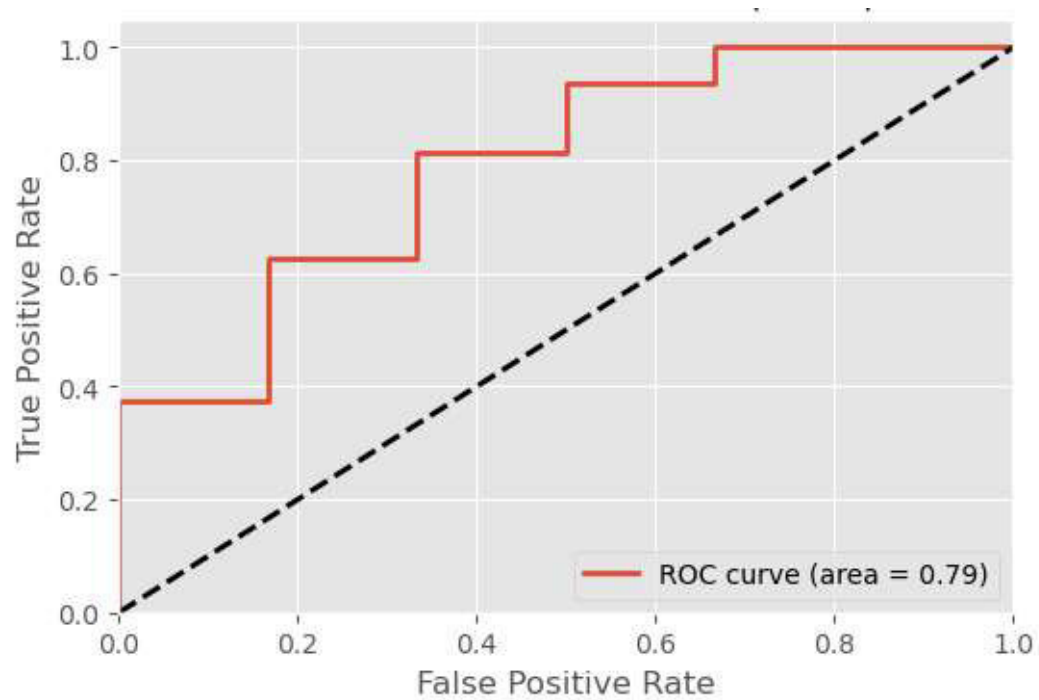


Figura 115. Matriz de Confusión – Test (CV = 10) del modelo SVM

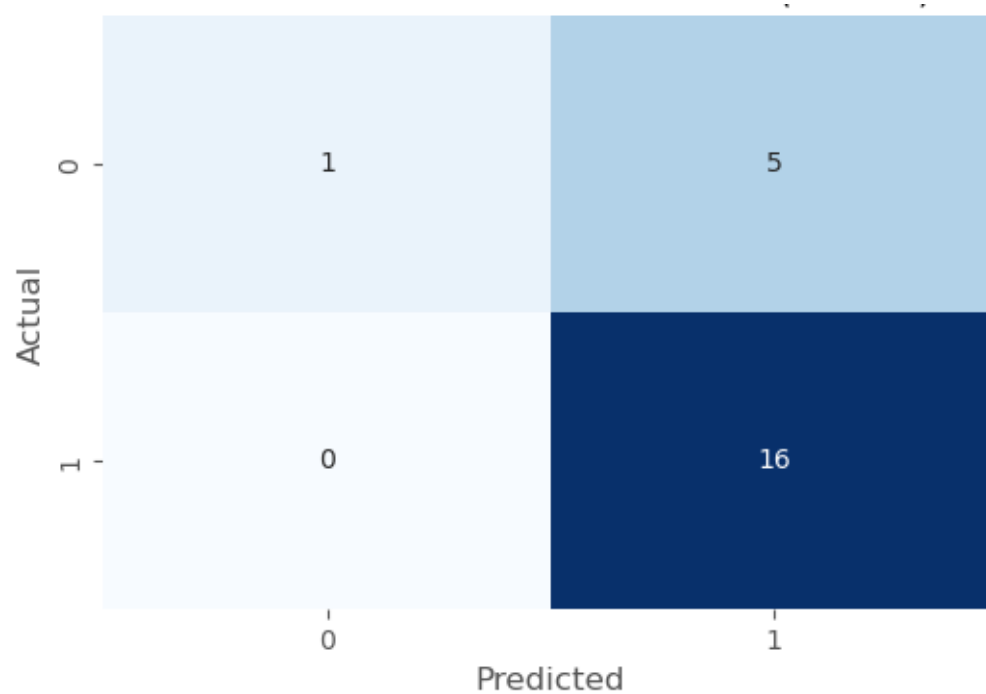


Figura 116. Curva ROC – Test (CV = 10) del modelo SVM

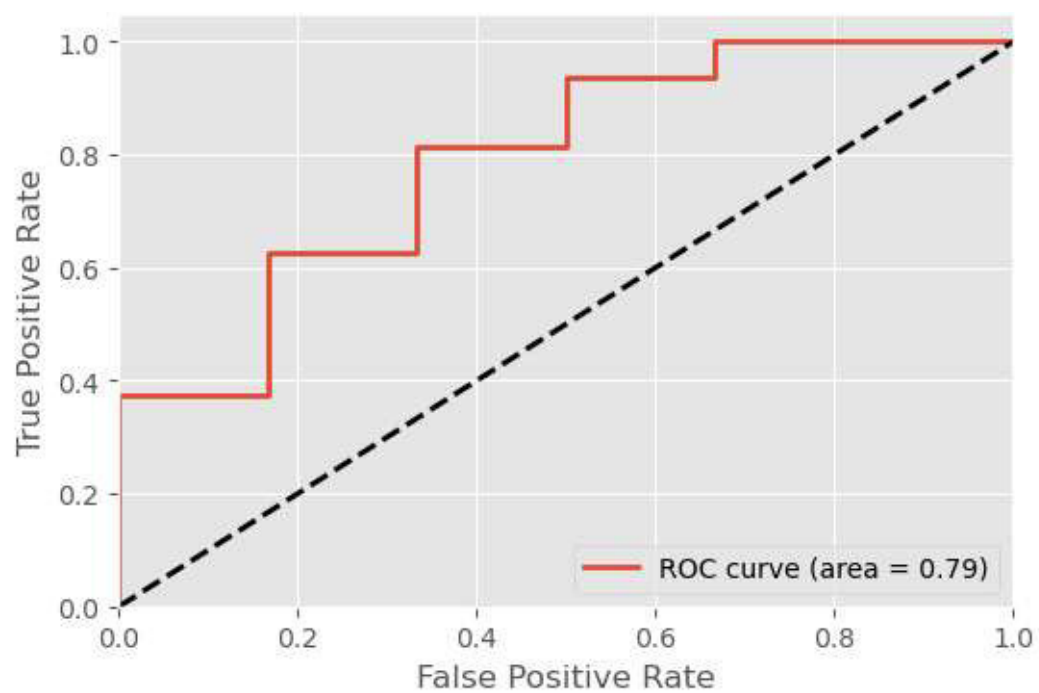


Figura 117. Matriz de Confusión – Test (CV = 11) del modelo SVM

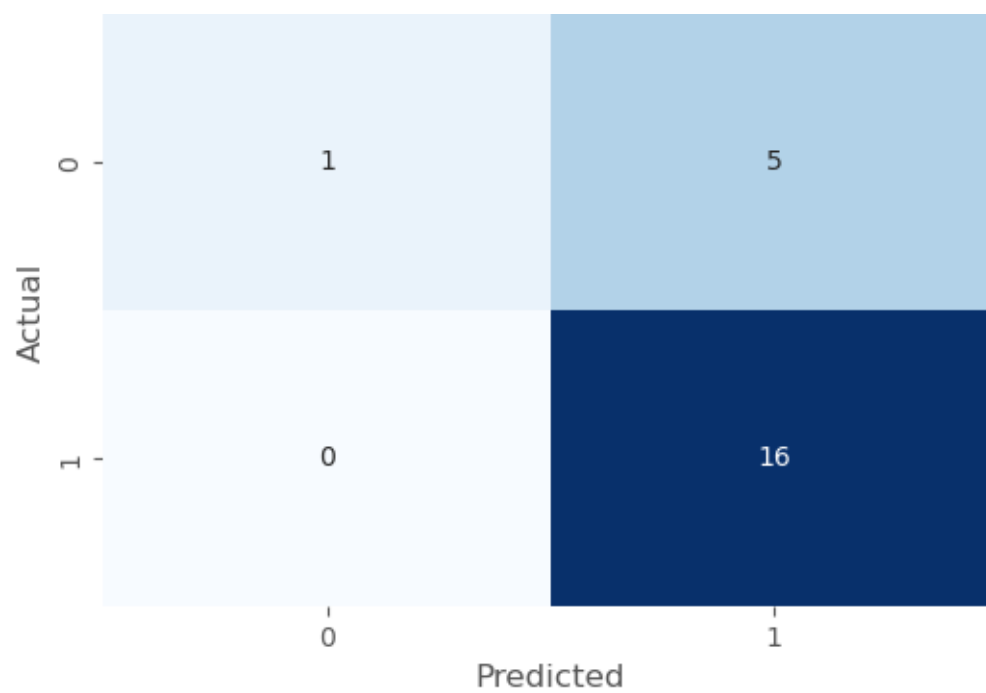
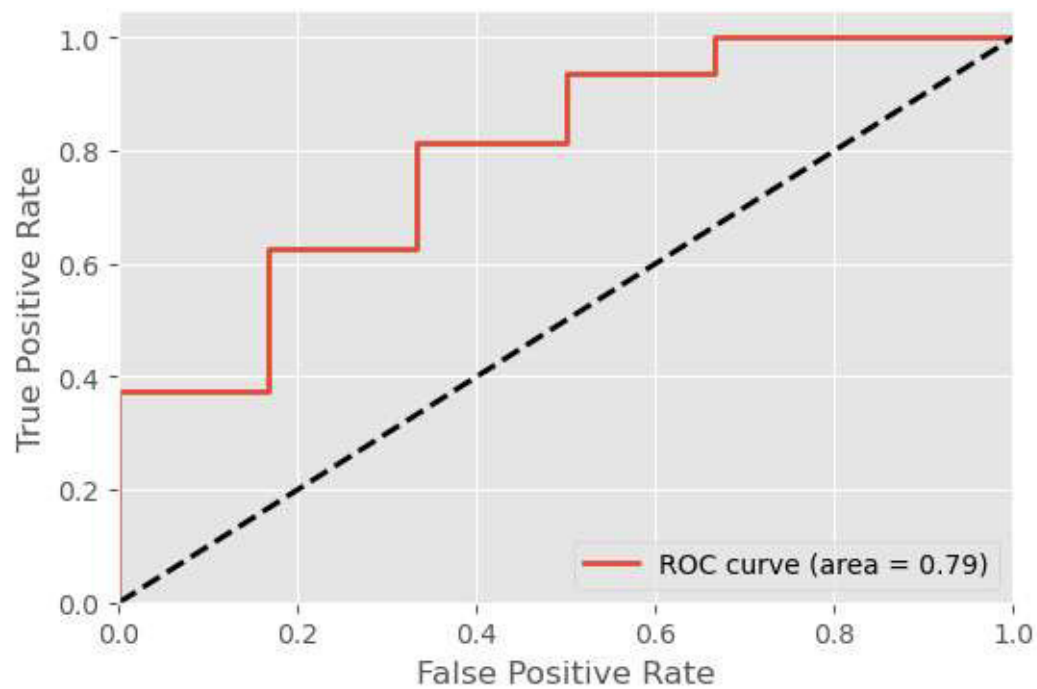


Figura 118. Curva ROC – Test (CV = 11) del modelo SVM



Anexo E. Redes Bayesianas

Figura 119. Importación de librerías y carga de datos

```
#librerías para el desarrollo del proyecto
import matplotlib.pyplot as plt #librería para graficas
from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches

import seaborn as sb

#librerías pandas para el manejo de los datos
import pandas as pd
import pandas as pq
import pandas as pf
import pandas as filtro_filas

#enlazando los datos en el archivo excel
data = pd.read_excel("../data/data_besqueras.xlsx", index_col=None)

# '../data/diabetes.csv'

import numpy as np
#invocando a la librería de clasificación
from sklearn.tree import DecisionTreeClassifier
```

Figura 120. Identificación de la variable objetivo

```
print(data.groupby('Accidentalaboral').size())

Accidentalaboral
0    31
1    79
dtype: int64

from sklearn.metrics import roc_curve, auc
from sklearn import datasets
from sklearn.multiclass import OneVsRestClassifier
from sklearn.svm import LinearSVC
from sklearn.preprocessing import label_binarize
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

Figura 121. *Particionamiento del conjunto de datos*

```
#particionando la data en 0.8 para entrenar y 0.2 para validar
from sklearn.model_selection import train_test_split
X_trainset, X_testset, y_trainset, y_testset = train_test_split(X, y, test_size=0.2, random_state=3)

print(X_trainset.shape, y_trainset.shape)
print (X_testset.shape, y_testset.shape)
```

```
(88, 7) (88,)
(22, 7) (22,)
```

Figura 122. *Código de hiperparámetros y construcción del modelo Redes Bayesianas*

```
# lista para almacenar los resultados
resultados = []

for i in range(2, 12):
    grid_search = GridSearchCV(GaussianNB(),
                                parametros,
                                refit=True,
                                verbose=0,
                                cv=i)

    # Entrenar modelo
    grid_search.fit(X_trainset, y_trainset)

    # Predicciones
    pred_Y1 = grid_search.predict(X_testset)
    y_prob = grid_search.predict_proba(X_testset)[:, 1] # Probabilidades clase positiva (1)

    # Métricas (para dos clases)
    accuracy = accuracy_score(y_testset, pred_Y1)
    precision = precision_score(y_testset, pred_Y1)
    recall = recall_score(y_testset, pred_Y1)
    f1 = f1_score(y_testset, pred_Y1)
    roc_auc = roc_auc_score(y_testset, y_prob)
    matthews = matthews_corrcoef(y_testset, pred_Y1)

    resultados.append({
        "cv": i,
        "Accuracy": accuracy,
        "Precision": precision,
        "Recall": recall,
        "F1 score": f1,
        "ROC AUC": roc_auc,
        "Matthews": matthews,
        "Predicciones": pred_Y1
    })
```

Figura 123. *Código para el cálculo de métricas de evaluación*

```
resultados.append({
    "cv": i,
    "Accuracy": accuracy,
    "Precision": precision,
    "Recall": recall,
    "F1 score": f1,
    "ROC AUC": roc_auc,
    "Matthews": matthews,
    "Predicciones": pred_Y1
})
```

Figura 124. Código para generar la matriz de confusión y curva ROC

```
# Matriz de confusión y curva ROC
for resultado in resultados:
    # Matriz de confusión
    plt.figure(figsize=(6, 4))
    cmf_matrix = confusion_matrix(y_testset, resultado['Predicciones'])
    sb.heatmap(cmf_matrix, annot=True, fmt='d', cmap='Blues', cbar=False)
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Decision Tree Confusion Matrix (cv={resultado["cv"]})')
    plt.show()

    # Curva ROC
    plt.figure(figsize=(6, 4))
    y_prob = grid_search.predict_proba(X_testset)[:, 1]
    fpr, tpr, _ = roc_curve(y_testset, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.plot(fpr, tpr, lw=2, label=f'ROC curve (area = {roc_auc:0.2f})')
    plt.plot([0, 1], [0, 1], 'k--', lw=2)
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.0])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Decision Tree ROC Curve (cv={resultado["cv"]})')
    plt.legend(loc="lower right")
    plt.show()
```

Figura 125. Matriz de Confusión – Test (CV = 2) del modelo Redes Bayesianas

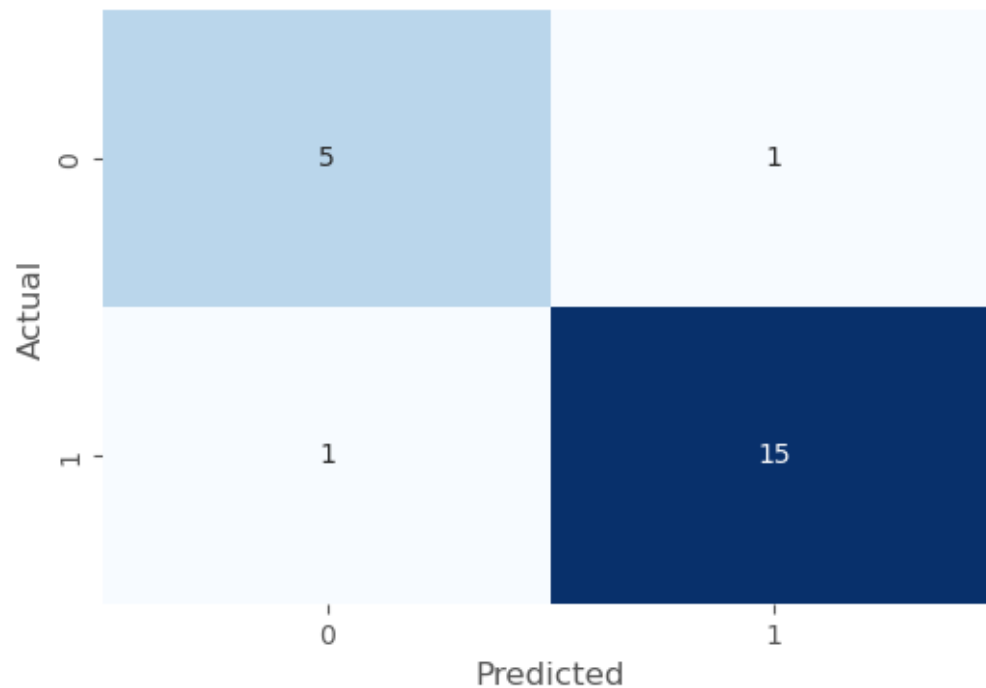


Figura 126. Curva ROC – Test (CV = 2) del modelo Redes Bayesianas

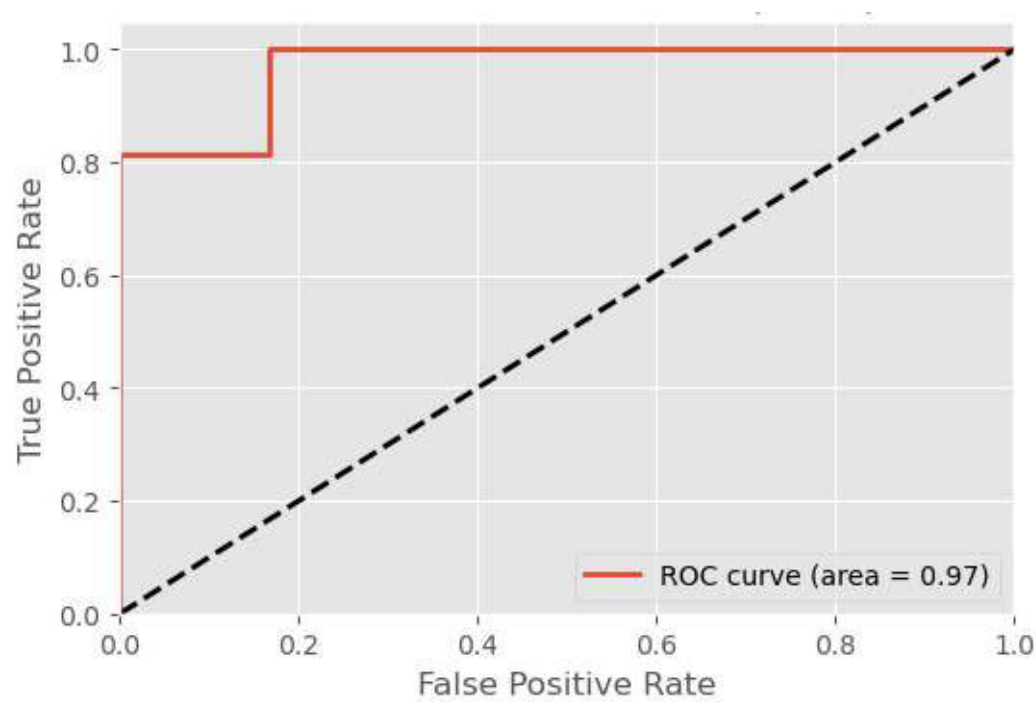


Figura 127. Matriz de Confusión – Test (CV = 3) del modelo Redes Bayesianas

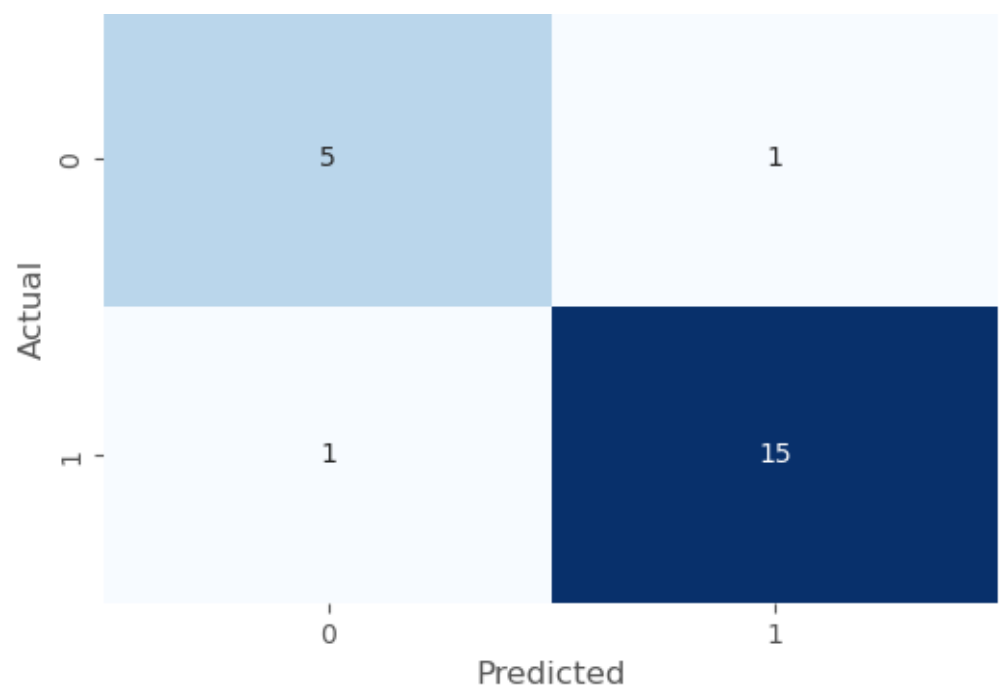


Figura 128. Curva ROC – Test (CV = 3) del modelo Redes Bayesianas

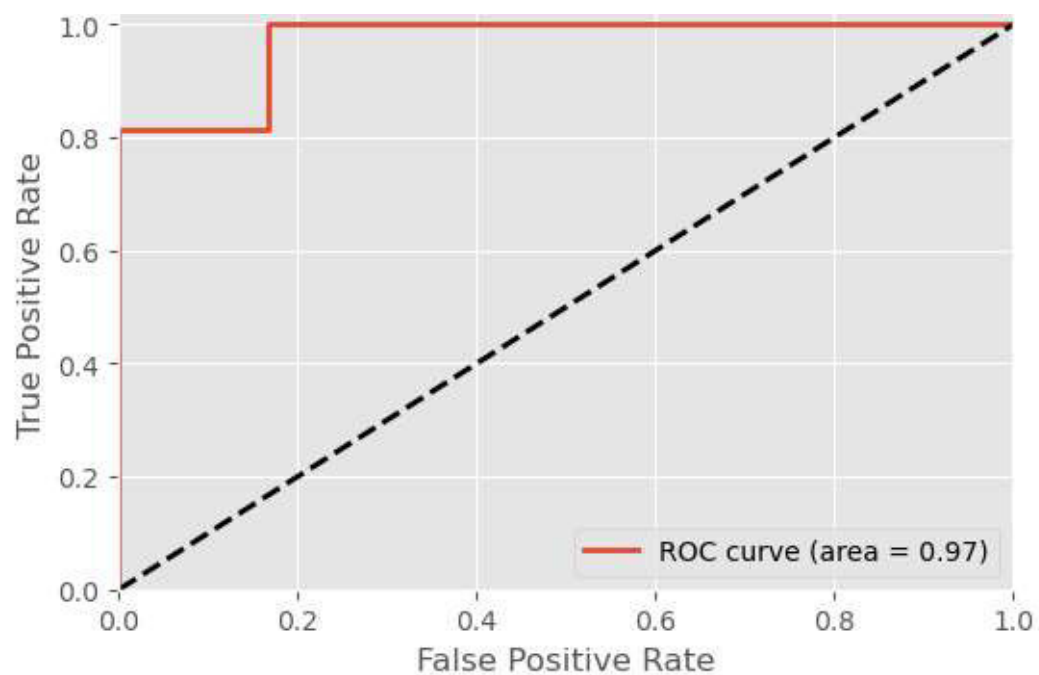


Figura 129. Matriz de Confusión – Test (CV = 4) del modelo Redes Bayesianas

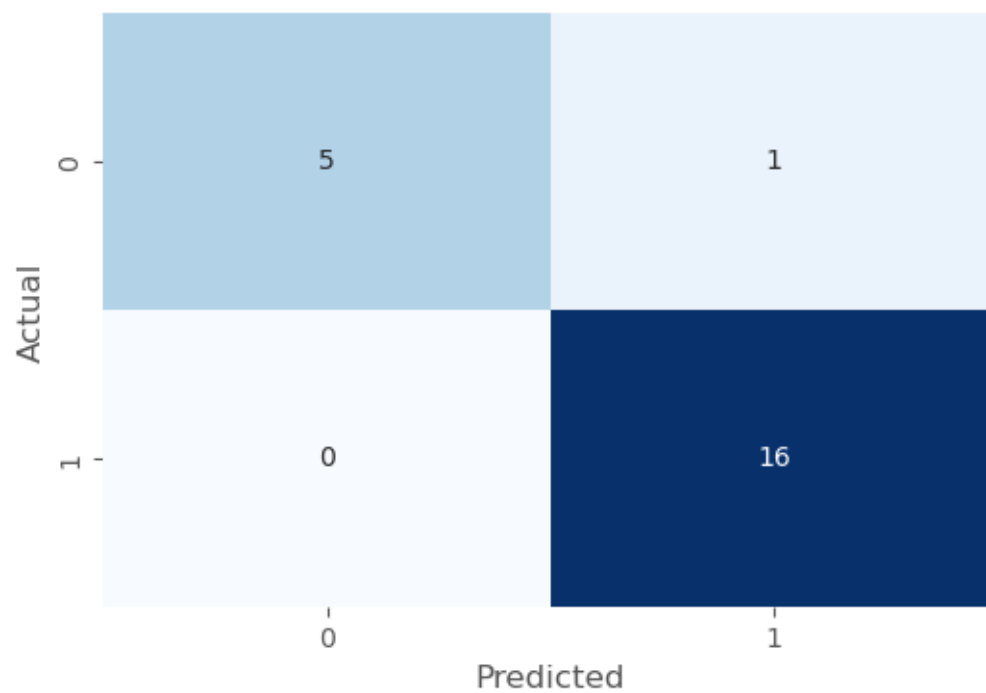


Figura 130. Curva ROC – Test (CV = 4) del modelo Redes Bayesianas

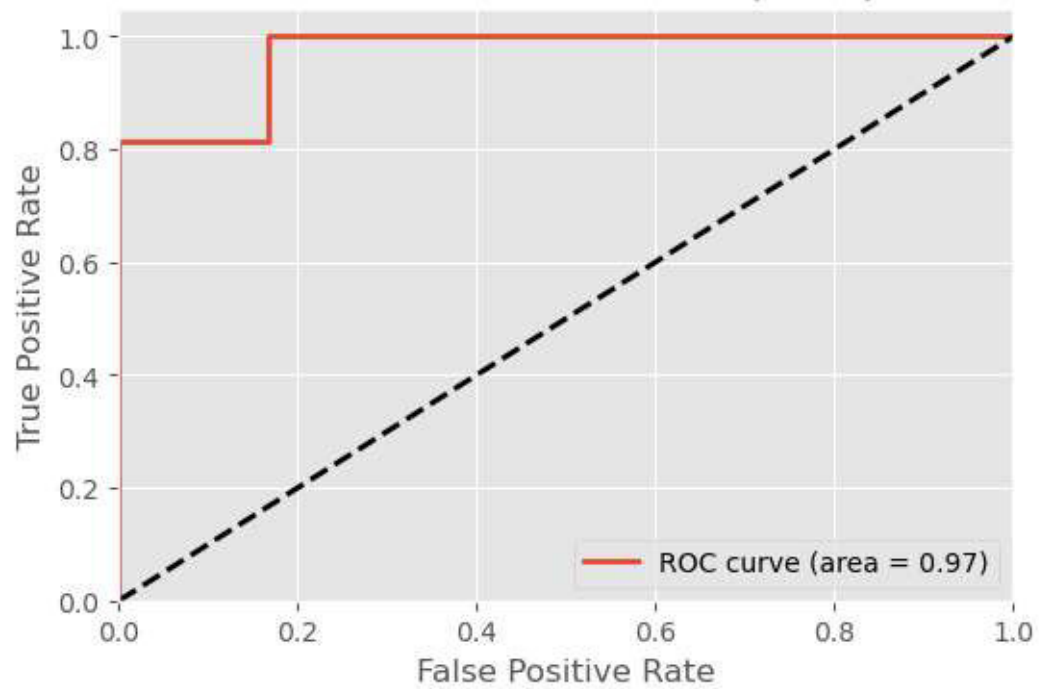


Figura 131. Matriz de Confusión – Test (CV = 5) del modelo Redes Bayesianas

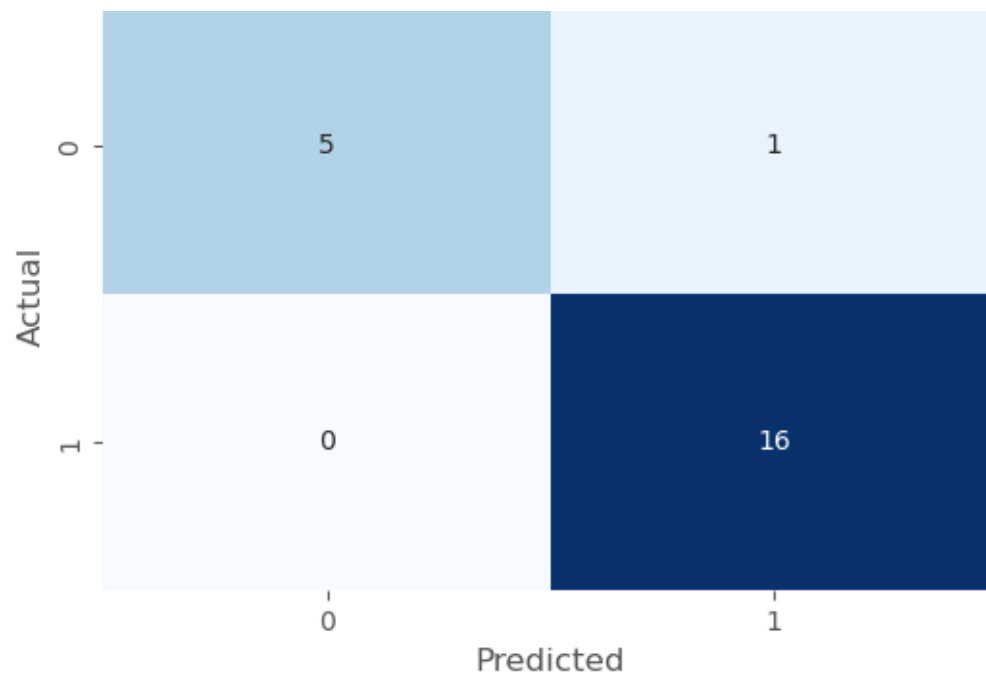


Figura 132. Curva ROC – Test (CV = 5) del modelo Redes Bayesianas

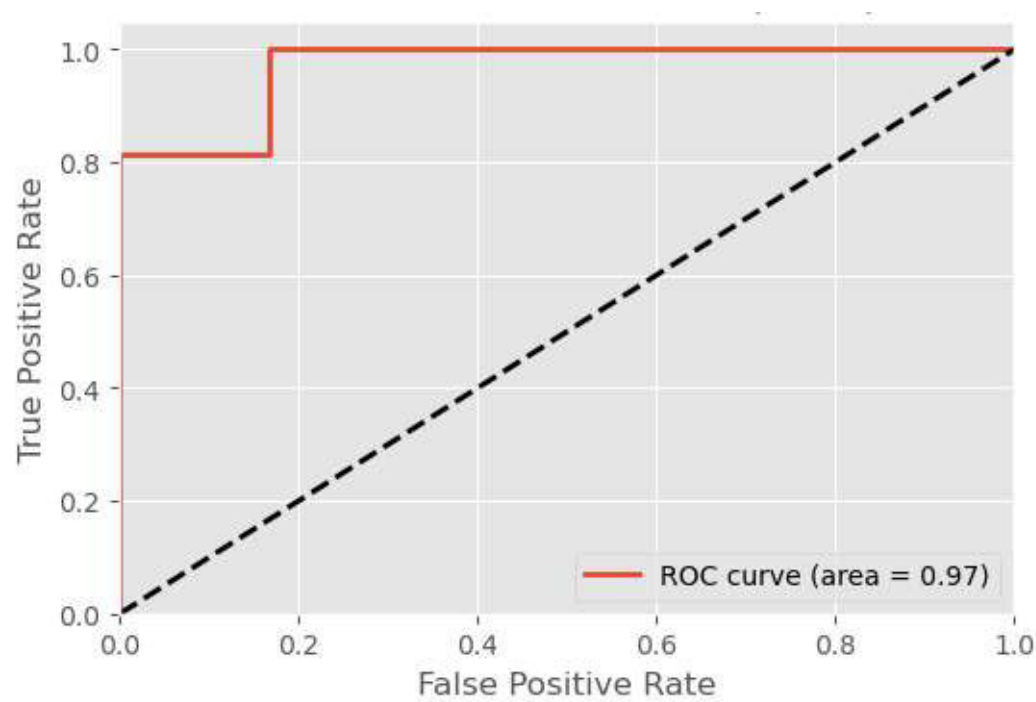


Figura 133. Matriz de Confusión – Test (CV = 6) del modelo Redes Bayesianas

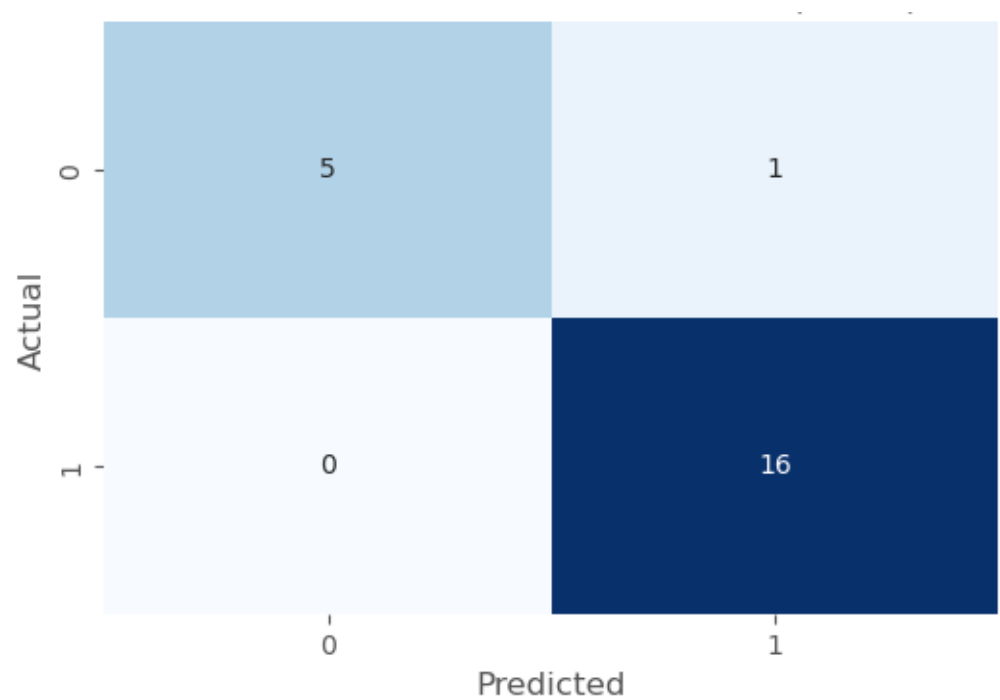


Figura 134. Curva ROC – Test (CV = 6) del modelo Redes Bayesianas

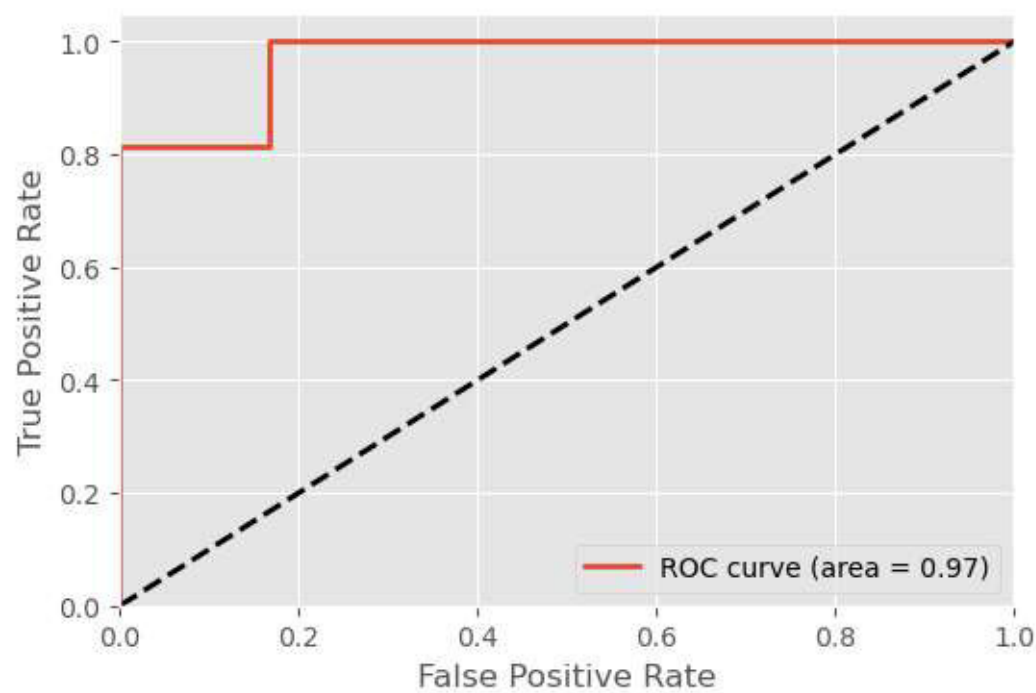


Figura 135. Matriz de Confusión – Test (CV = 7) del modelo Redes Bayesianas

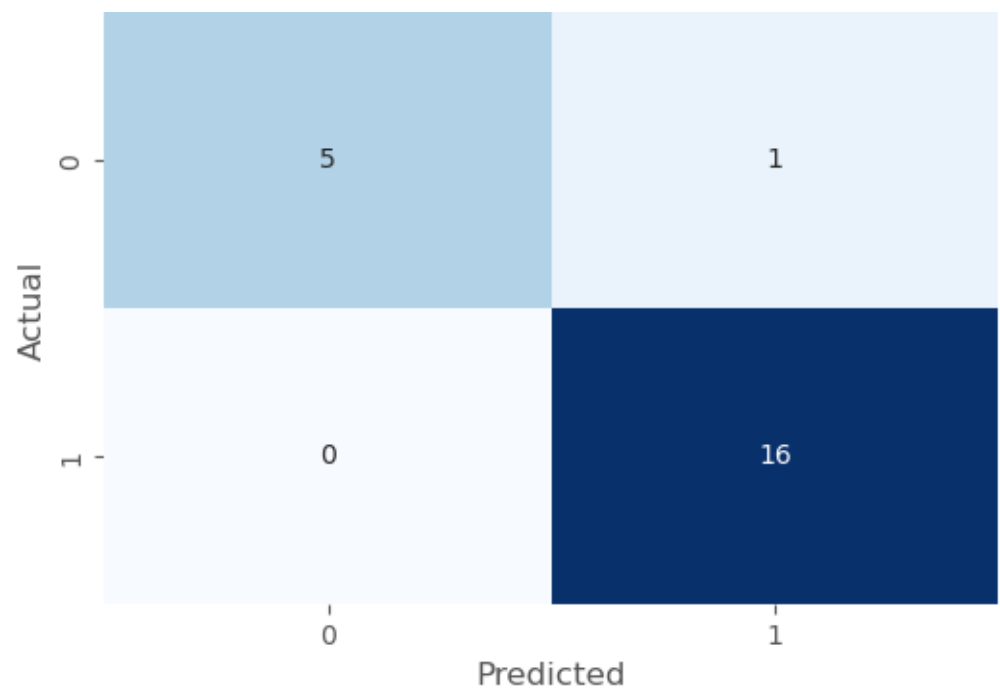


Figura 136. Curva ROC – Test (CV = 7) del modelo Redes Bayesianas

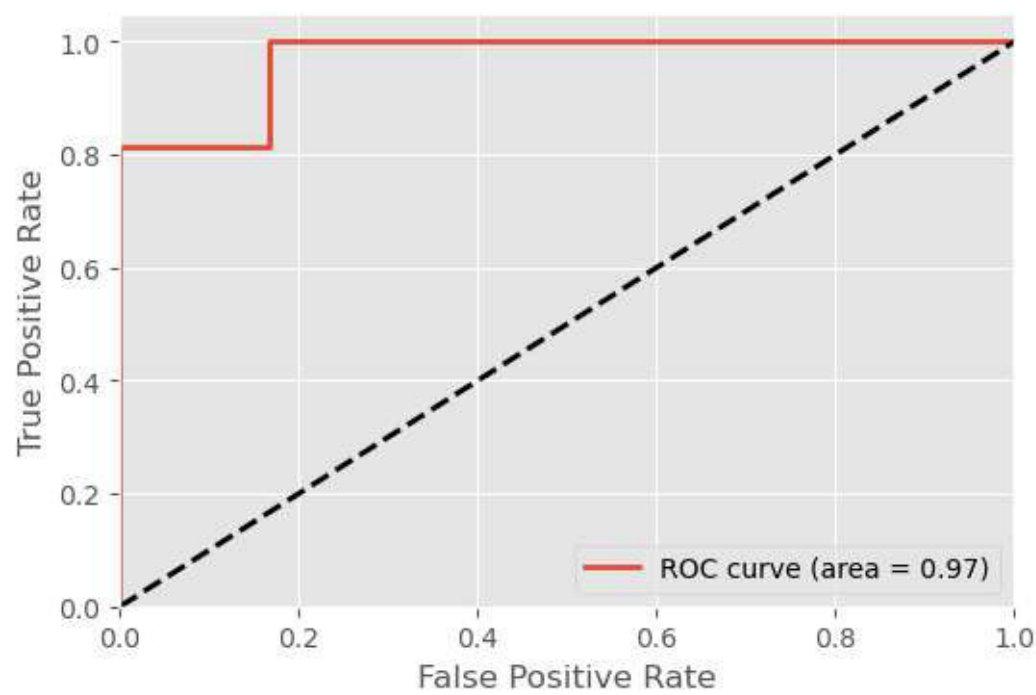


Figura 137. Matriz de Confusión – Test (CV = 8) del modelo Redes Bayesianas

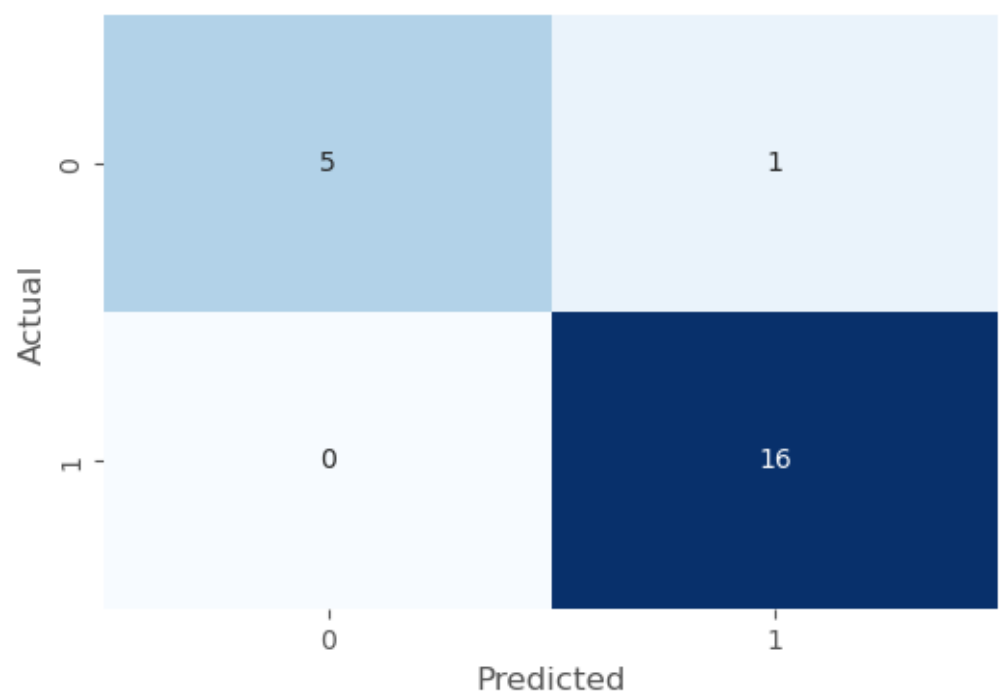


Figura 138. Curva ROC – Test (CV = 8) del modelo Redes Bayesianas

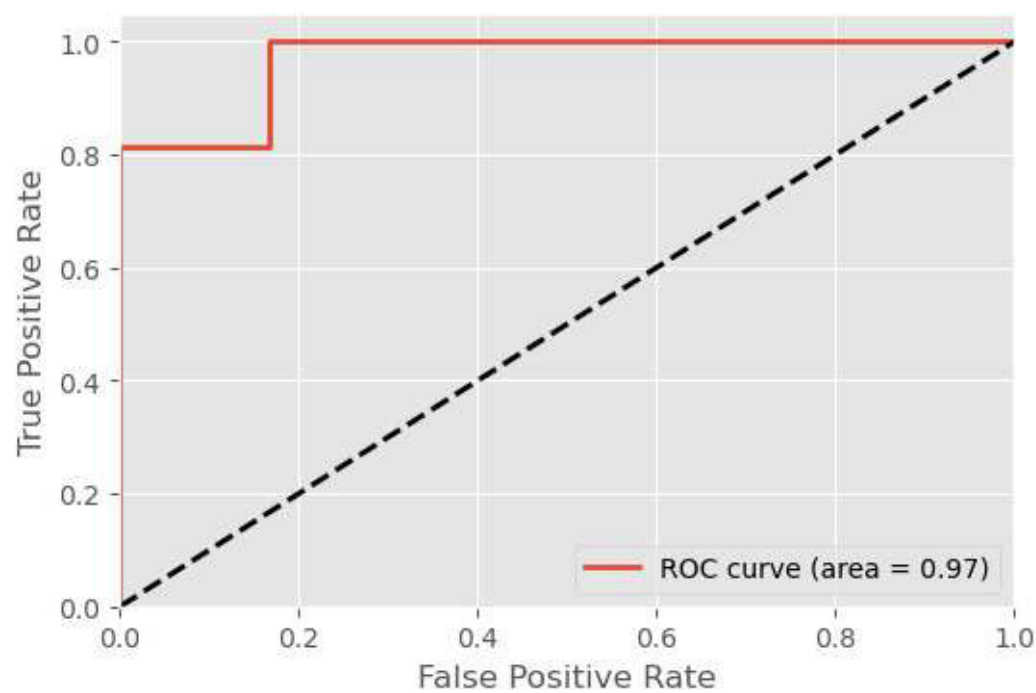


Figura 139. Matriz de Confusión – Test (CV = 9) del modelo Redes Bayesianas

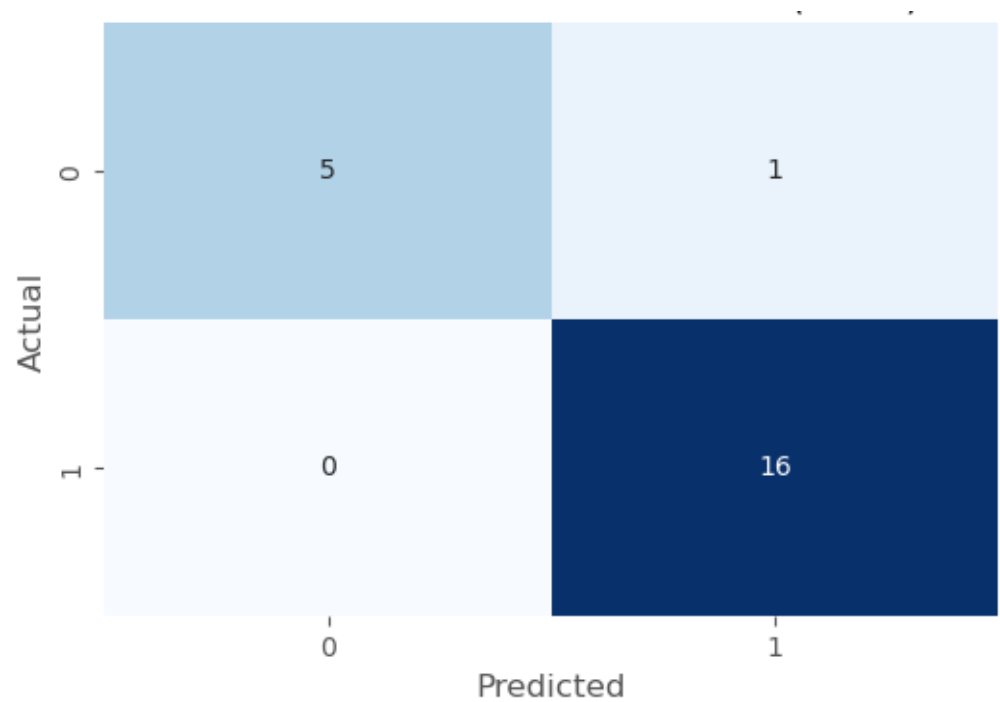


Figura 140. Curva ROC – Test (CV = 9) del modelo Redes Bayesianas

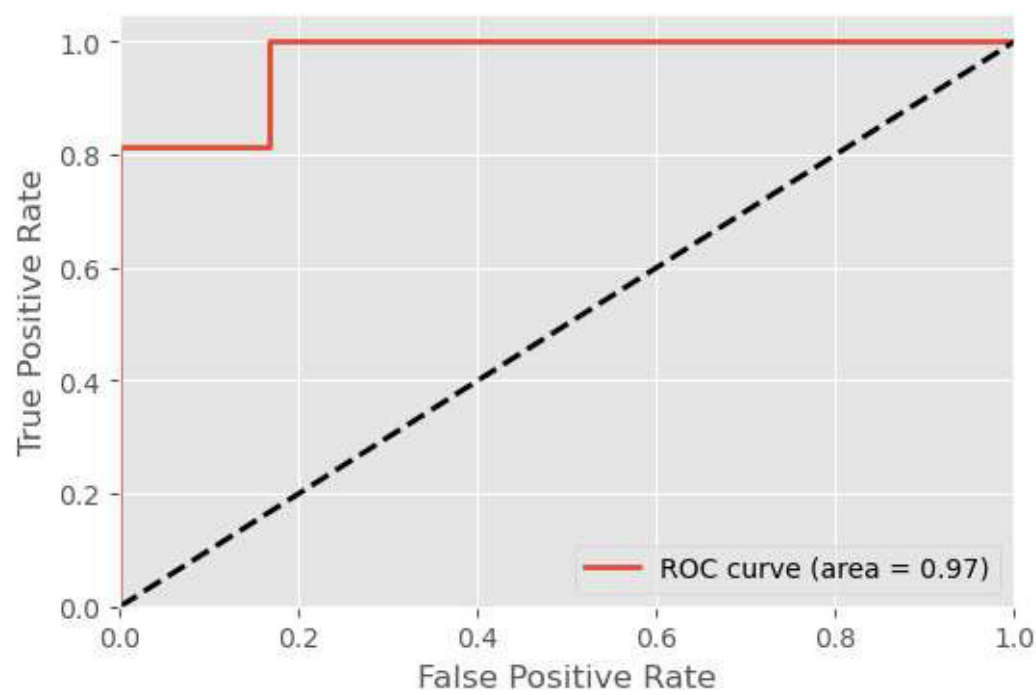


Figura 141. Matriz de Confusión – Test (CV = 10) del modelo Redes Bayesianas

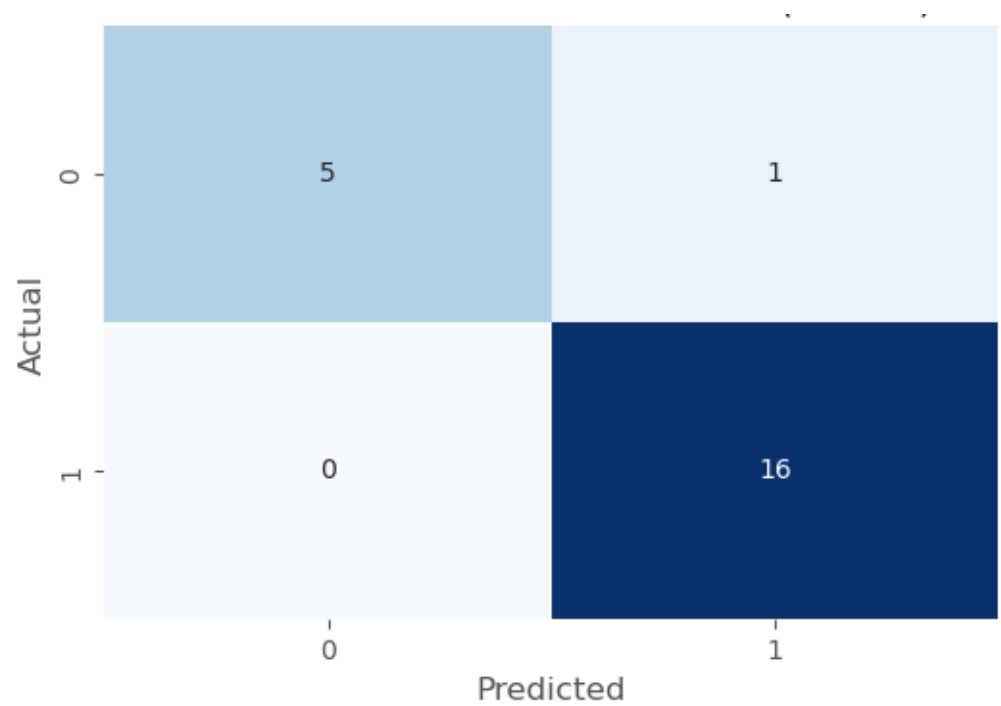


Figura 142. Curva ROC – Test (CV = 10) del modelo Redes Bayesianas

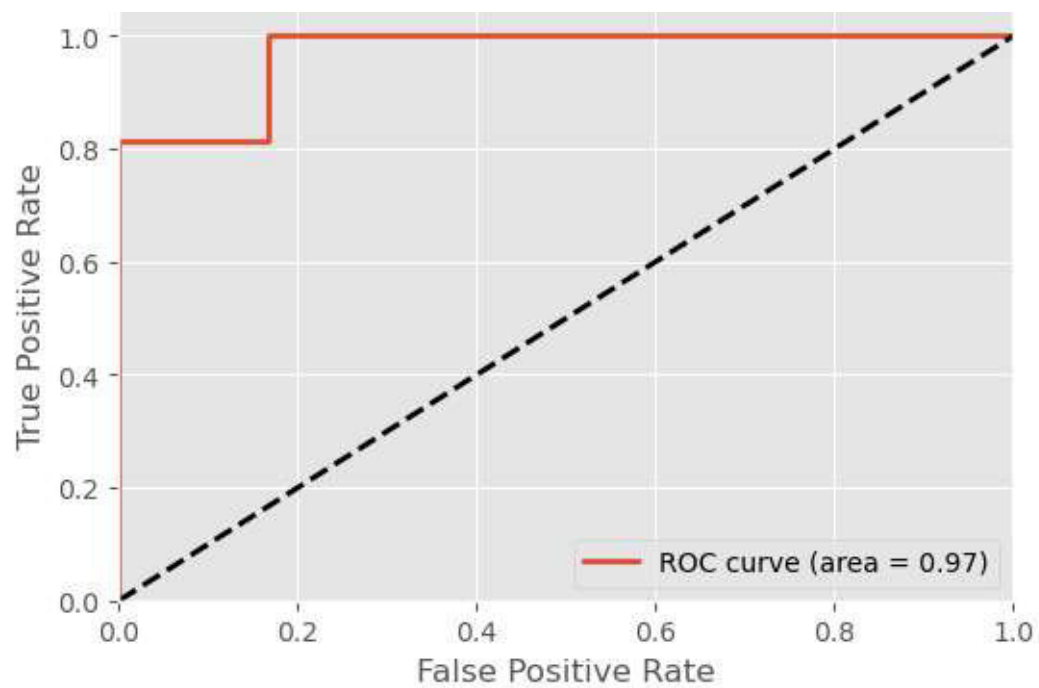


Figura 143. Matriz de Confusión – Test (CV = 11) del modelo Redes Bayesianas

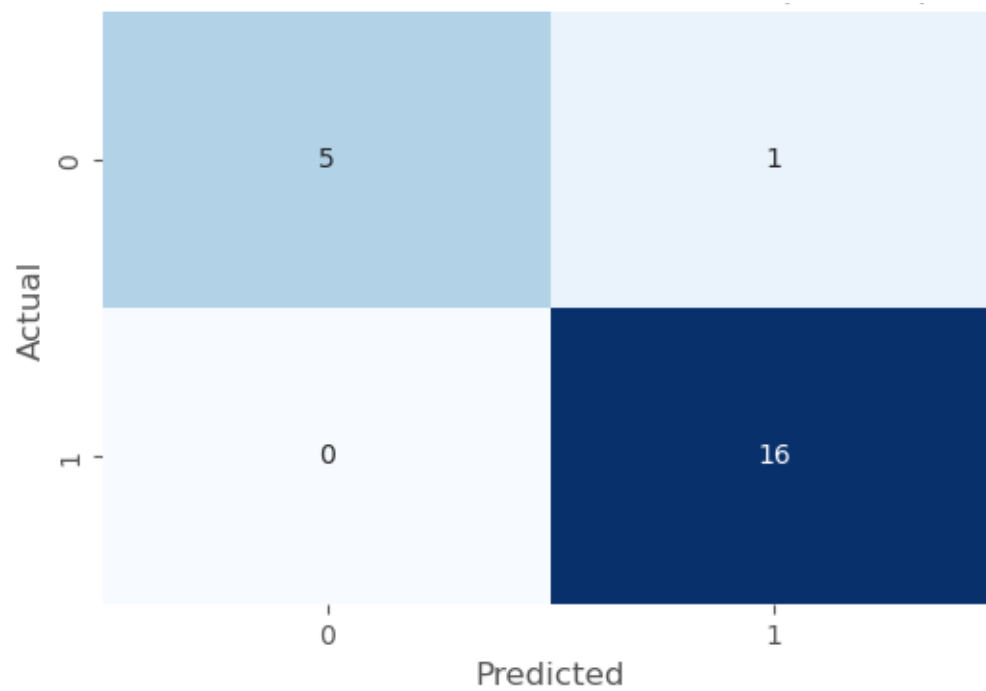
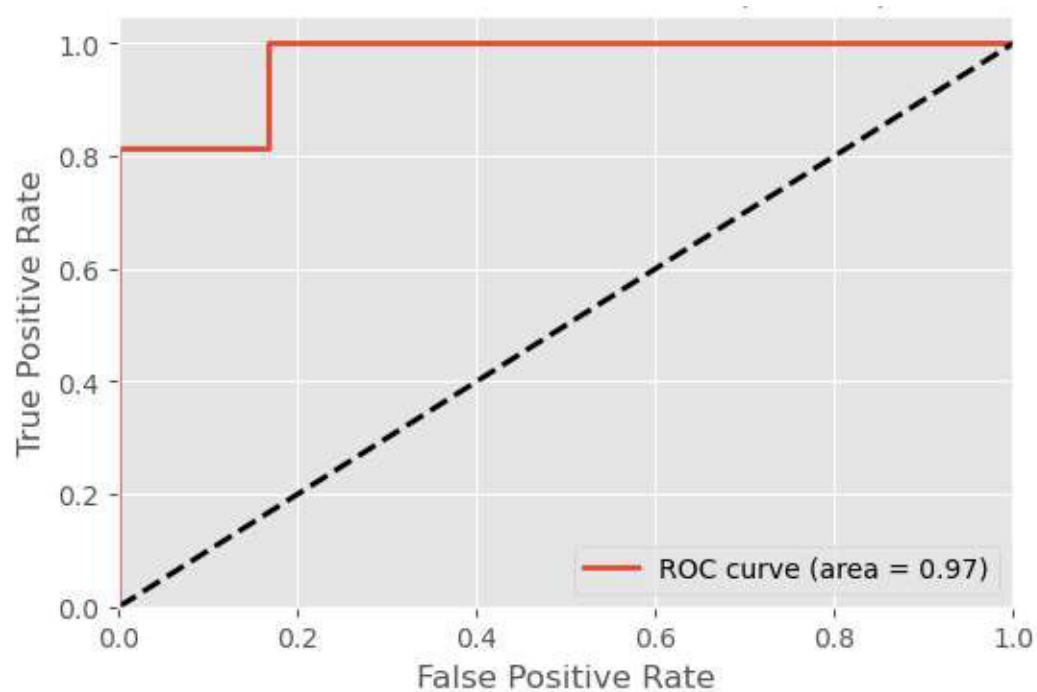


Figura 144. Curva ROC – Test (CV = 11) del modelo Redes Bayesianas



Anexo F. Gradiente

Figura 145. Importación de librerías y carga de datos

```
#librerías para el desarrollo del proyecto
import matplotlib.pyplot as plt #librería para graficas
from matplotlib.colors import ListedColormap
import matplotlib.patches as mpatches

import seaborn as sb

#librerías pandas para el manejo de los datos
import pandas as pd
import pandas as pq
import pandas as pf
import pandas as filtro_files

#enlazando los datos en el archivo excel
data = pd.read_excel("../data/data_pesquera.xlsx", index_col=None)
import numpy as np
#invocando a la librería de clasificación
from sklearn.tree import DecisionTreeClassifier
```

Figura 146. Identificación de la variable objetivo

```
print(data.groupby('Accidentelaboral').size())
```

```
Accidentelaboral
0    31
1    79
dtype: int64
```

```
from sklearn.metrics import roc_curve, auc
from sklearn import datasets
from sklearn.multiclass import OneVsRestClassifier
from sklearn.svm import LinearSVC
from sklearn.preprocessing import label_binarize
from sklearn.model_selection import train_test_split
import matplotlib.pyplot as plt
```

Figura 147. *Particionamiento del conjunto de datos*

Particionamiento de la data

```
[30]: #particionando la data en 0.8 para entrenar y 0.2 para validar
from sklearn.model_selection import train_test_split
X_trainset, X_testset, y_trainset, y_testset = train_test_split(X, Y, test_size=0.2, random_state=42)

print(X_trainset.shape, y_trainset.shape)
print(X_testset.shape, y_testset.shape)

(88, 7) (88,)
(22, 7) (22,)
```

Figura 148. *Código de hiperparámetros y construcción del modelo Gradiente*

```
# Lista para almacenar los resultados
resultados = []

# INICIO DEL FOR
for i in range(2, 12):
    grid_search = GridSearchCV(GradientBoostingClassifier(),
                               param_grid,
                               refit=True,
                               verbose=0,
                               cv=i)

    # Entrenar modelo
    grid_search.fit(X_trainset, y_trainset)

    # AQUÍ SE AGREGA LA LÍNEA PARA CAPTURAR EL MEJOR MODELO
    grid_search = grid_search.best_estimator_

    # Predicciones
    pred_Y1 = grid_search.predict(X_testset)
    y_prob = grid_search.predict_proba(X_testset)[:, 1] # Probabilidades clase positiva (1)

    # Métricas (para dos clases)
    accuracy = accuracy_score(y_testset, pred_Y1)
    precision = precision_score(y_testset, pred_Y1)
    recall = recall_score(y_testset, pred_Y1)
    f1 = f1_score(y_testset, pred_Y1)
    roc_auc = roc_auc_score(y_testset, y_prob)
    matthews = matthews_corrcoef(y_testset, pred_Y1)

    resultados.append({
        "cv": i,
        "Accuracy": accuracy,
        "Precision": precision,
        "Recall": recall,
        "F1 score": f1,
        "ROC AUC": roc_auc,
        "Matthews": matthews,
        "Predicciones": pred_Y1
    })

})
```

Figura 149. *Código para el cálculo de métricas de evaluación*

```
resultados.append({
    "cv": i,
    "Accuracy": accuracy,
    "Precision": precision,
    "Recall": recall,
    "F1 score": f1,
    "ROC AUC": roc_auc,
    "Matthews": matthews,
    "Predicciones": pred_Y1
})
```

Figura 150. Código para generar la matriz de confusión y curva ROC

```
# Matriz de confusión y curva ROC
for resultado in resultados:
    # Matriz de confusión
    plt.figure(figsize=(6, 4))
    cnf_matrix = confusion_matrix(y_testset, resultado['Predicciones'])
    sb.heatmap(cnf_matrix, annot=True, fmt='d', cmap='Blues', cbar=False)
    plt.xlabel('Predicted')
    plt.ylabel('Actual')
    plt.title(f'Decision Tree Confusion Matrix (cv={resultado["cv"]})')
    plt.show()

    # Curva ROC
    plt.figure(figsize=(6, 4))
    y_prob = grid_search.predict_proba(X_testset)[:, 1]
    fpr, tpr, _ = roc_curve(y_testset, y_prob)
    roc_auc = auc(fpr, tpr)

    plt.plot(fpr, tpr, lw=2, label=f'ROC curve (area = {roc_auc:0.2f})')
    plt.plot([0, 1], [0, 1], 'k--', lw=2)
    plt.xlim([0.0, 1.0])
    plt.ylim([0.0, 1.05])
    plt.xlabel('False Positive Rate')
    plt.ylabel('True Positive Rate')
    plt.title(f'Decision Tree ROC Curve (cv={resultado["cv"]})')
    plt.legend(loc='lower right')
    plt.show()
```

Figura 151. Matriz de Confusión – Test (CV = 2) del modelo Gradiente

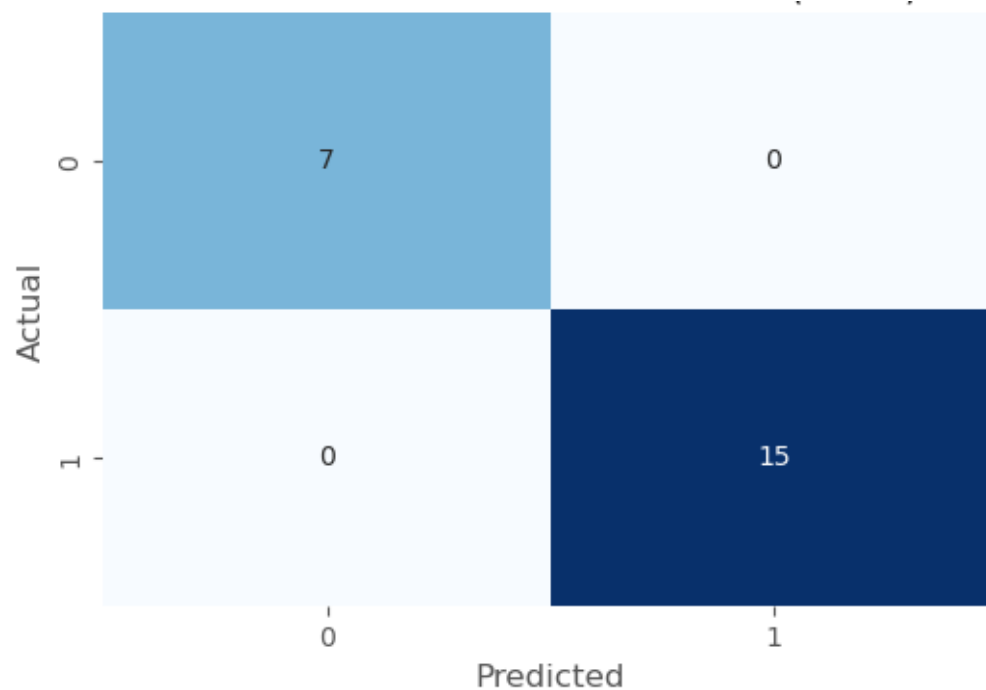


Figura 152. Curva ROC – Test (CV = 2) del modelo Gradiente

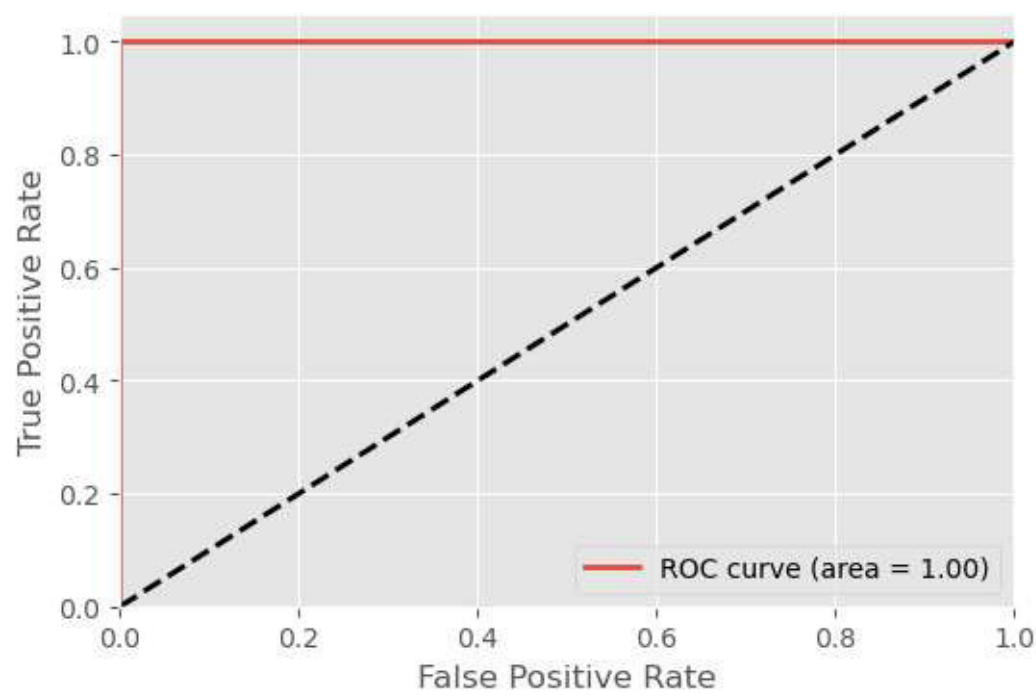


Figura 153. Matriz de Confusión – Test (CV = 3) del modelo Gradiente

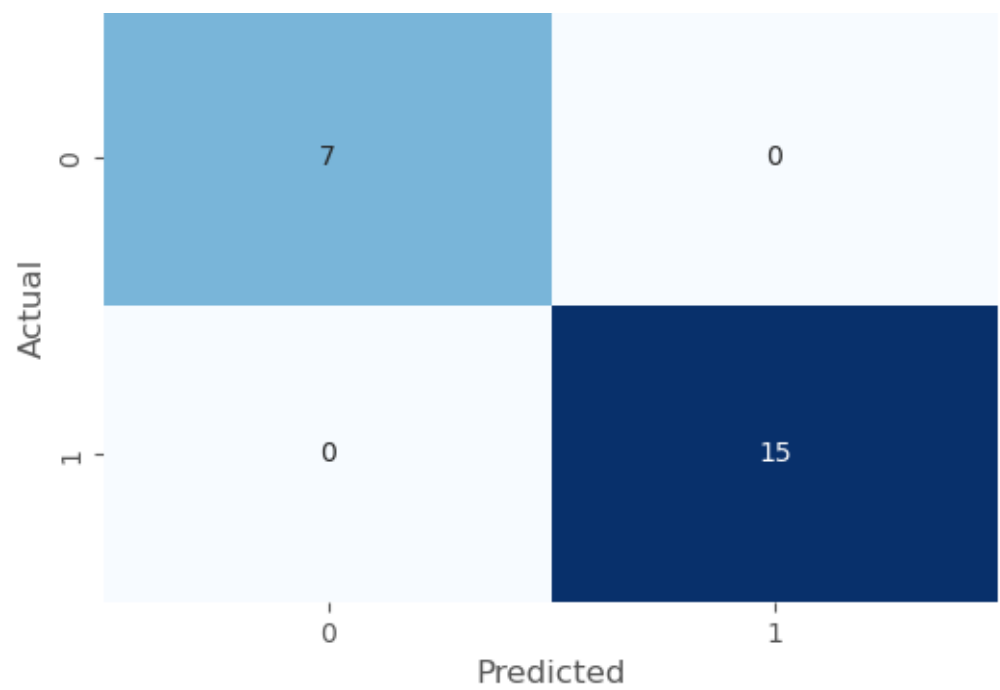


Figura 154. Curva ROC – Test (CV = 3) del modelo Gradiente

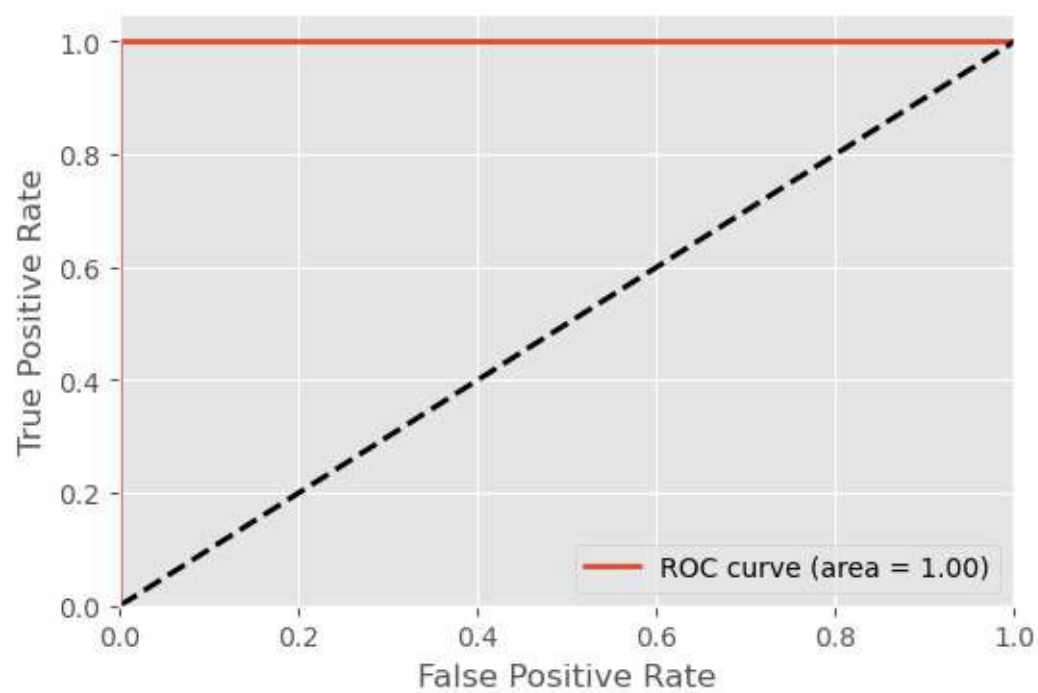


Figura 155. Matriz de Confusión – Test (CV = 4) del modelo Gradiente

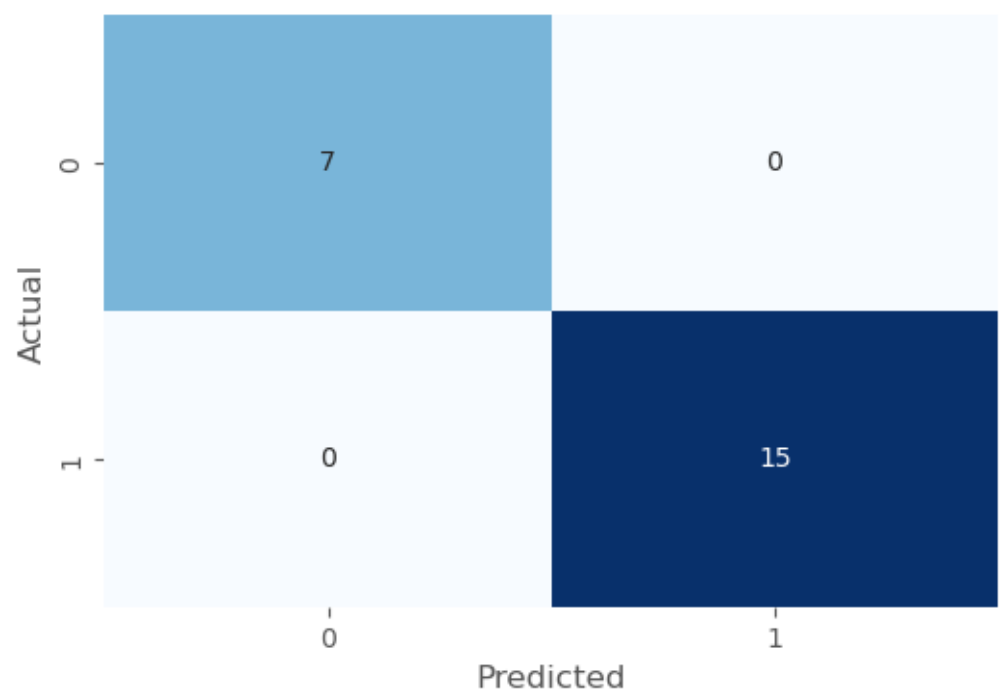


Figura 156. Curva ROC – Test (CV = 4) del modelo Gradiente

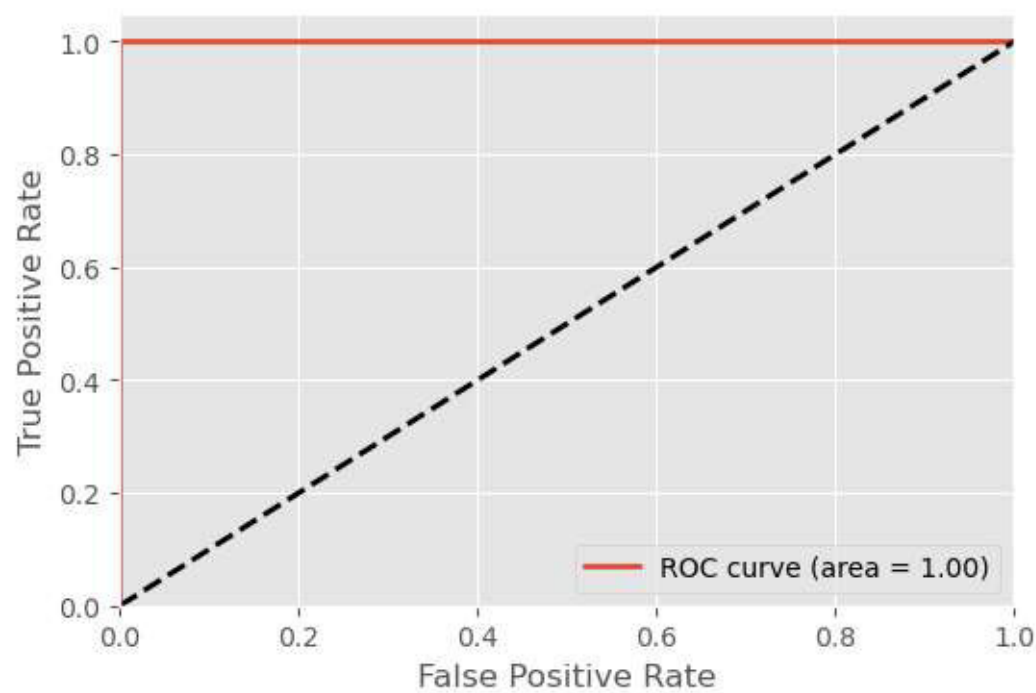


Figura 157. Matriz de Confusión – Test (CV = 5) del modelo Gradiente

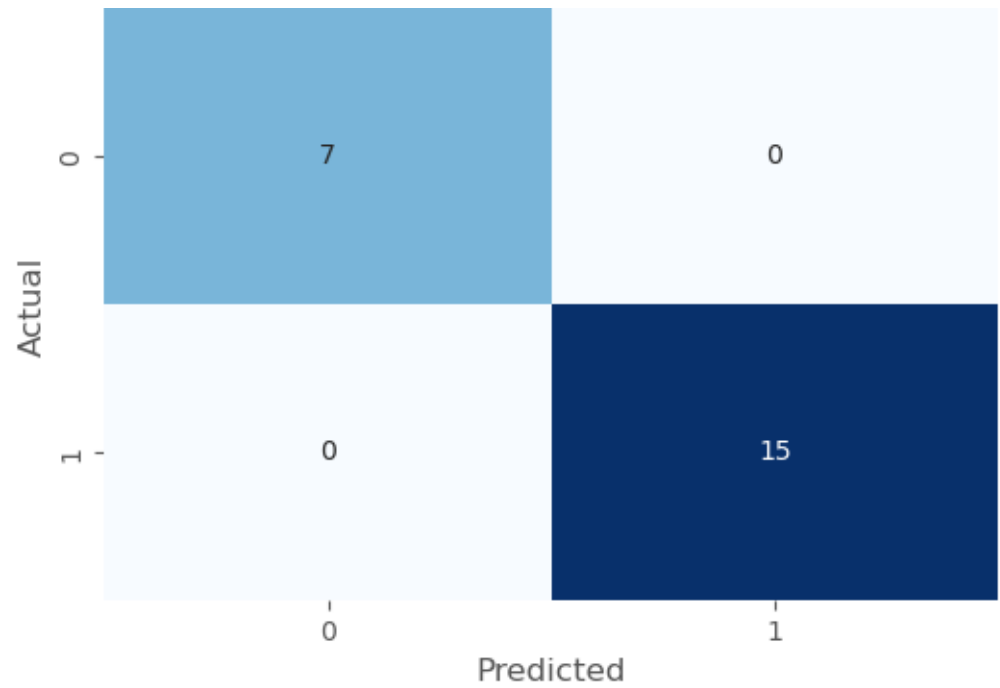


Figura 158. Curva ROC – Test (CV = 5) del modelo Gradiente

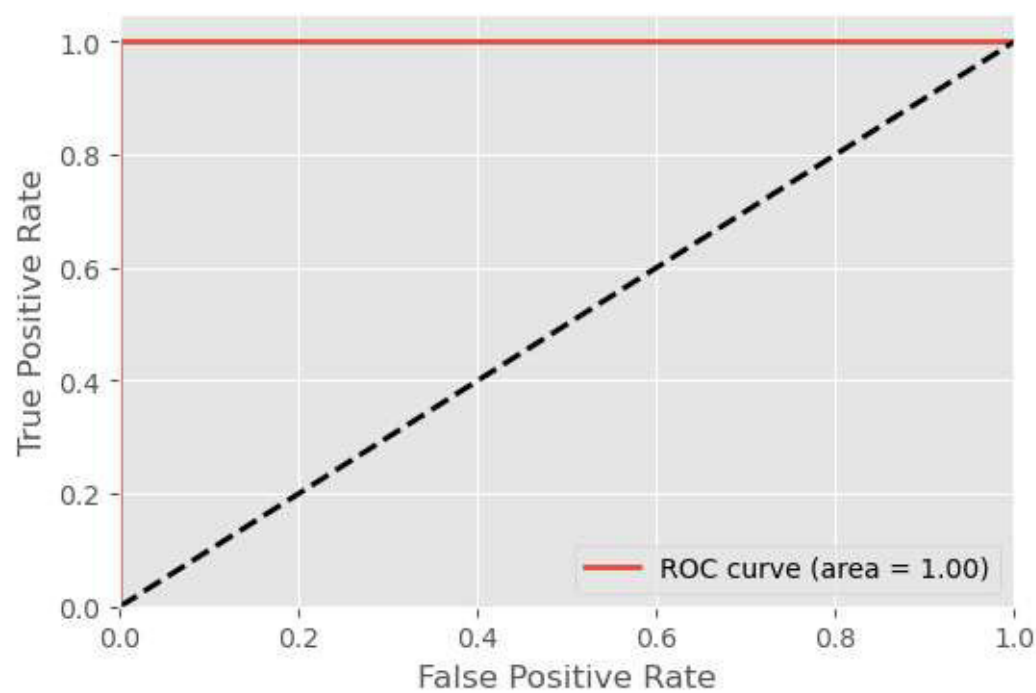


Figura 159. Matriz de Confusión – Test (CV = 6) del modelo Gradiente

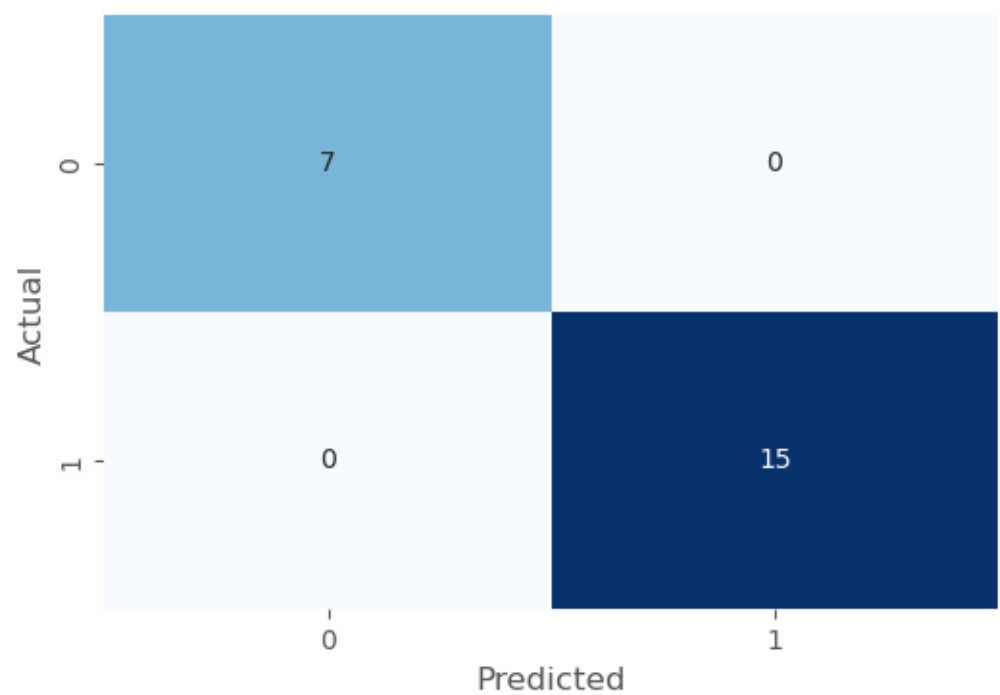


Figura 160. Curva ROC – Test (CV = 6) del modelo Gradiente

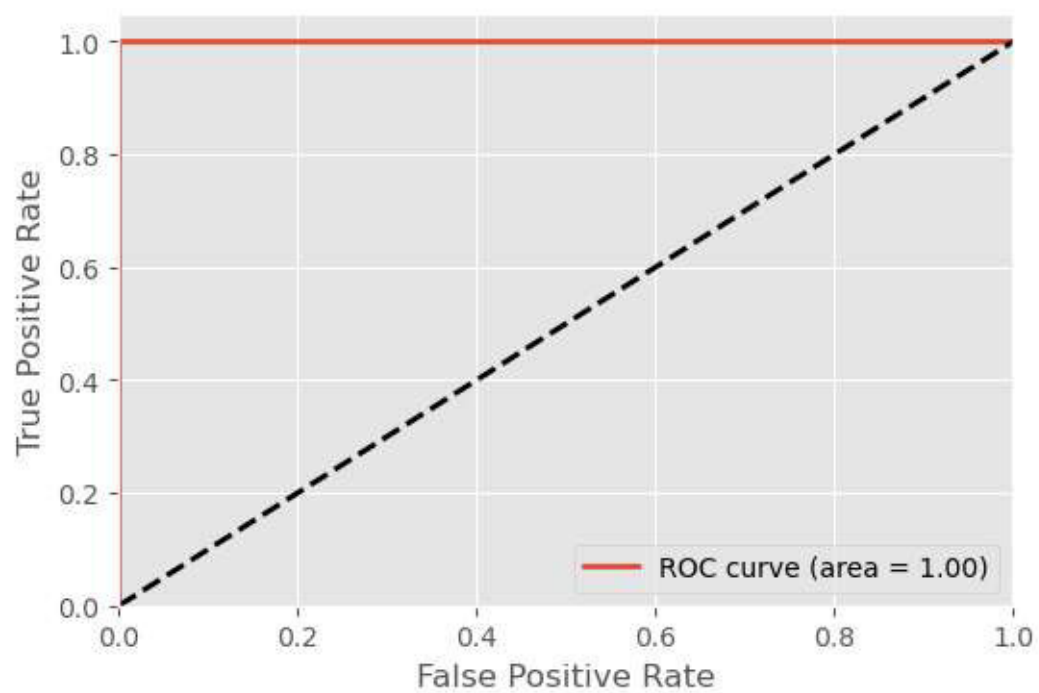


Figura 161. Matriz de Confusión – Test (CV = 7) del modelo Gradiente

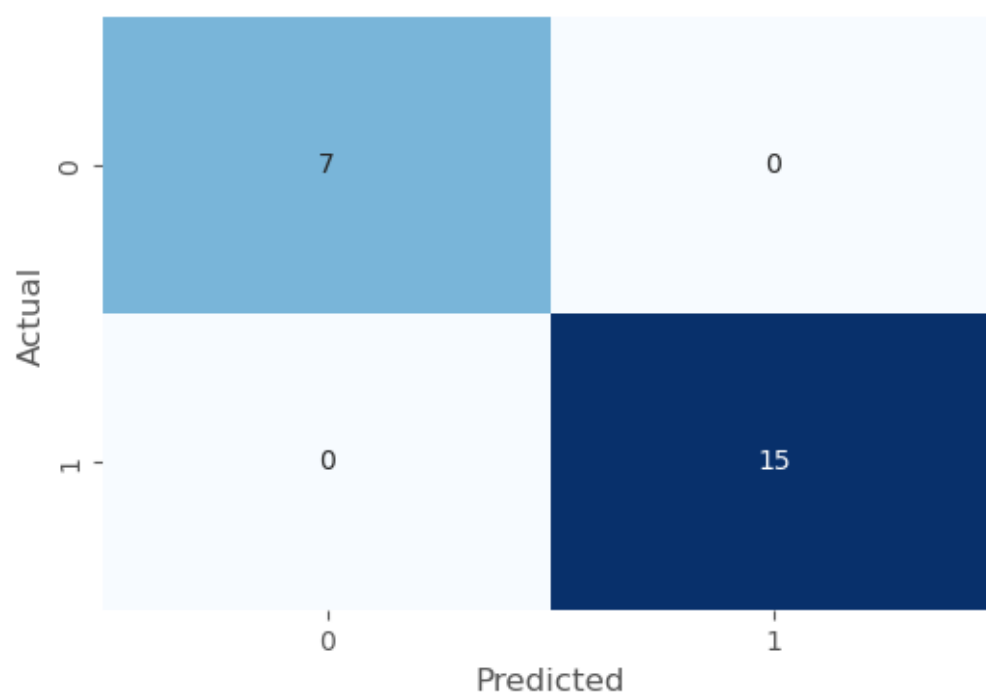


Figura 162. Curva ROC – Test (CV = 7) del modelo Gradiente

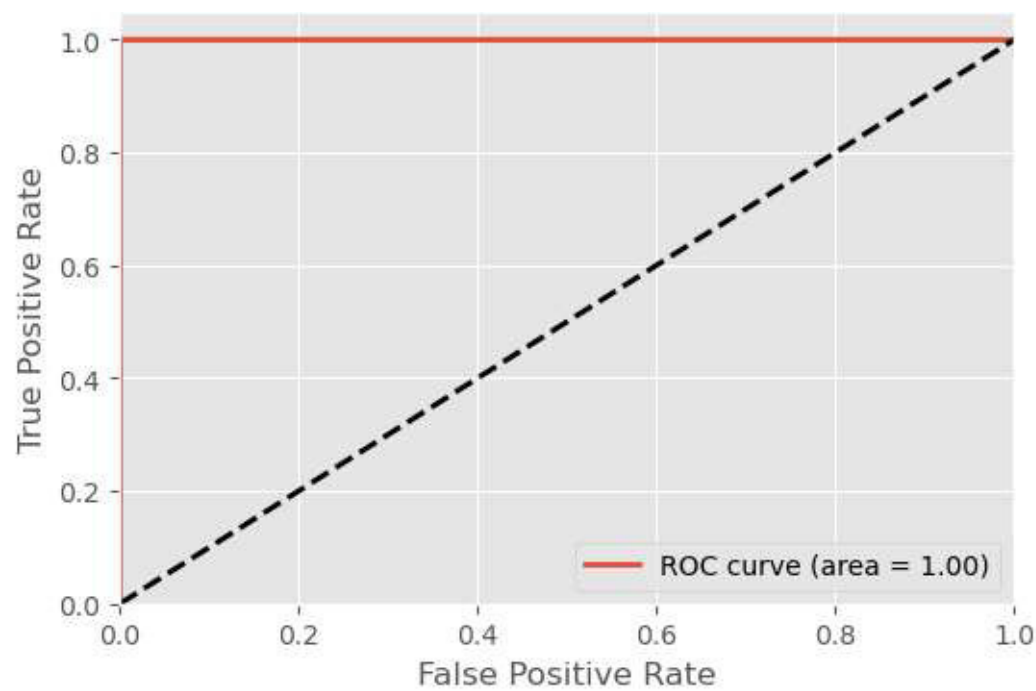


Figura 163. Matriz de Confusión – Test (CV = 8) del modelo Gradiente

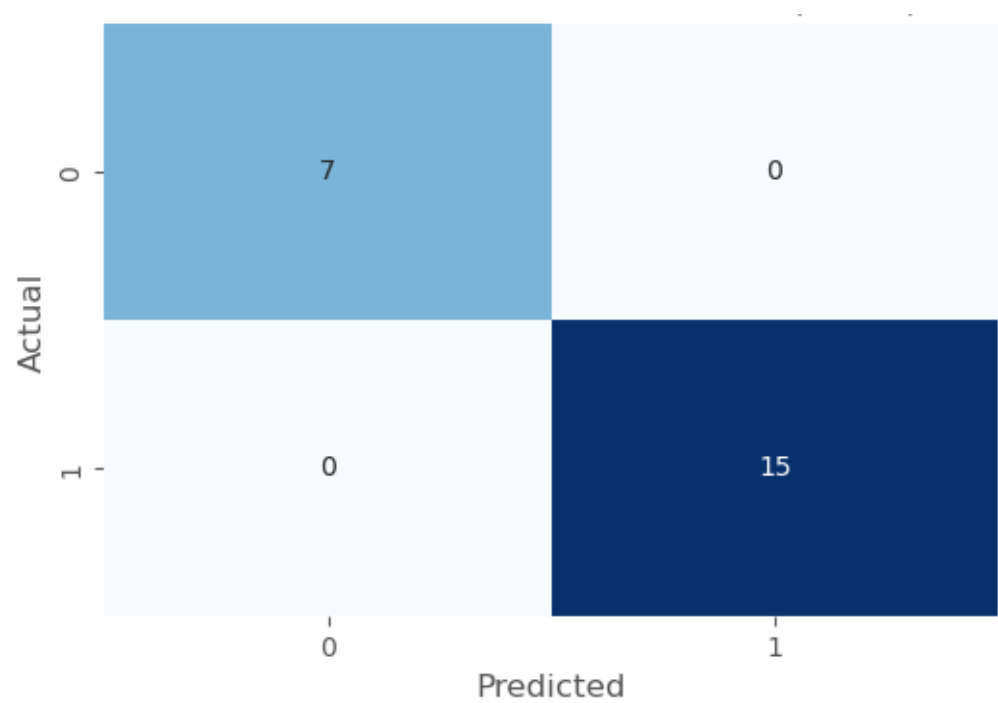


Figura 164. Curva ROC – Test (CV = 8) del modelo Gradiente

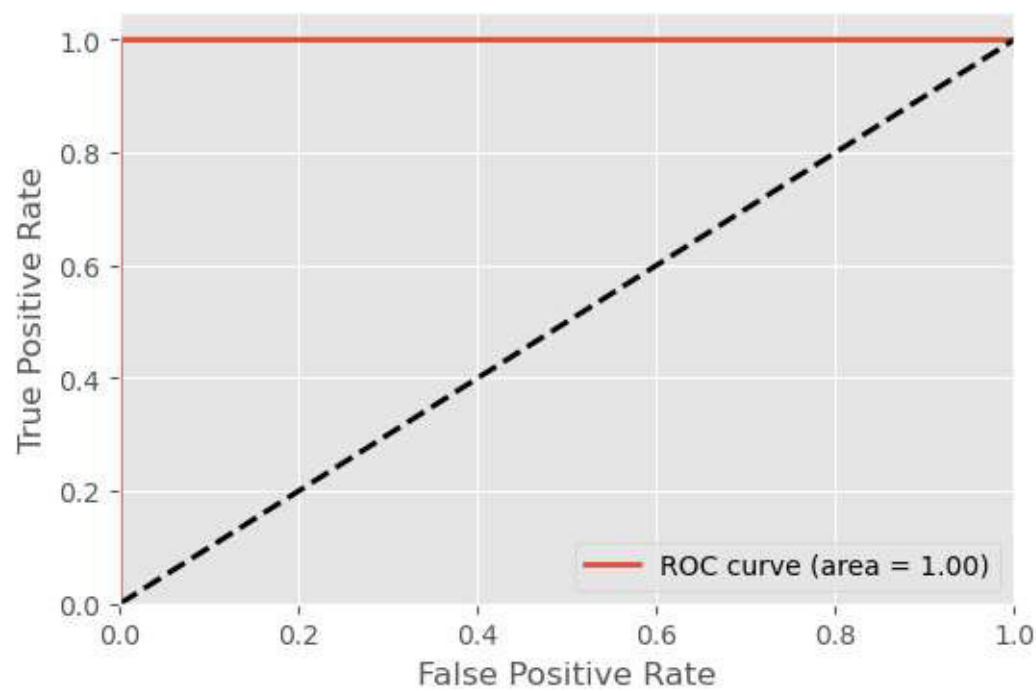


Figura 165. Matriz de Confusión – Test (CV = 9) del modelo Gradiente

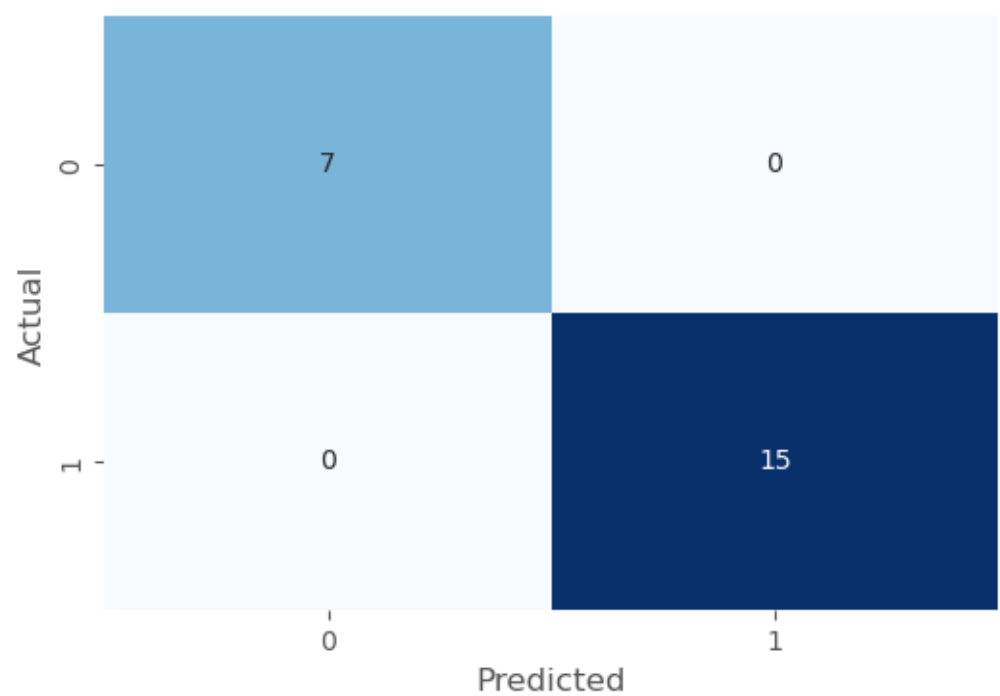


Figura 166. Curva ROC – Test (CV = 9) del modelo Gradiente

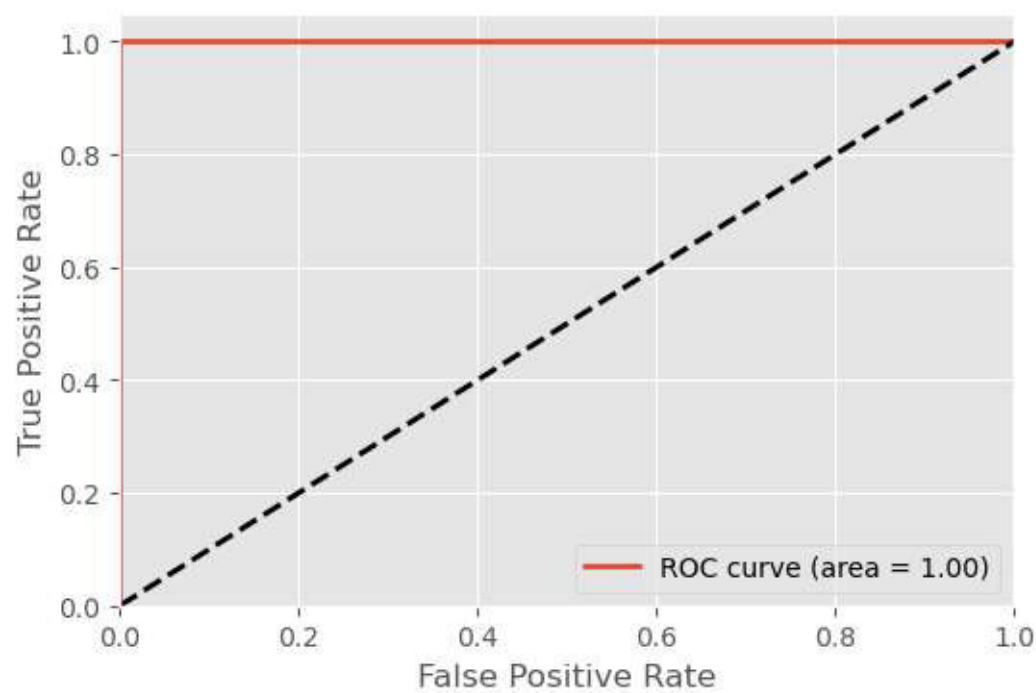


Figura 167. Matriz de Confusión – Test (CV = 10) del modelo Gradiente

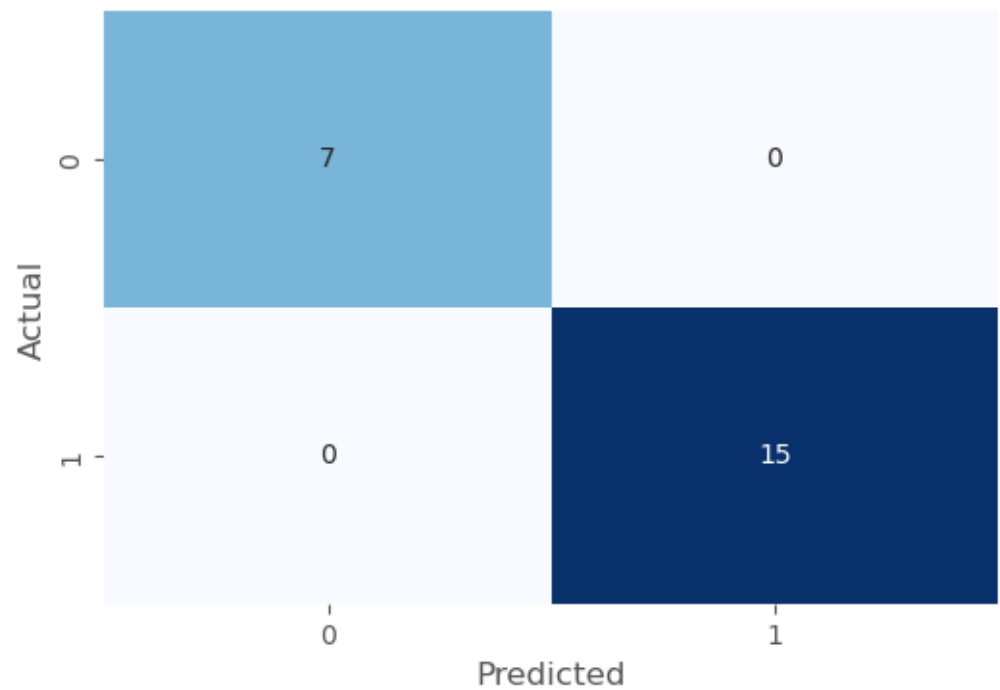


Figura 168. Curva ROC – Test (CV = 10) del modelo Gradiente

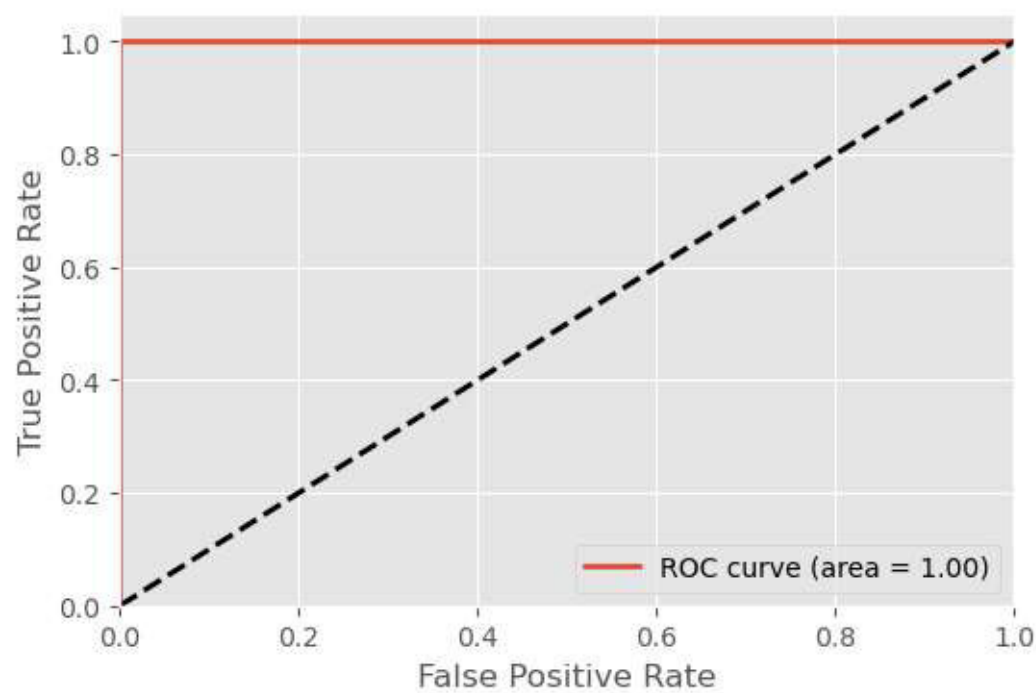


Figura 169. Matriz de Confusión – Test (CV = 11) del modelo Gradiente

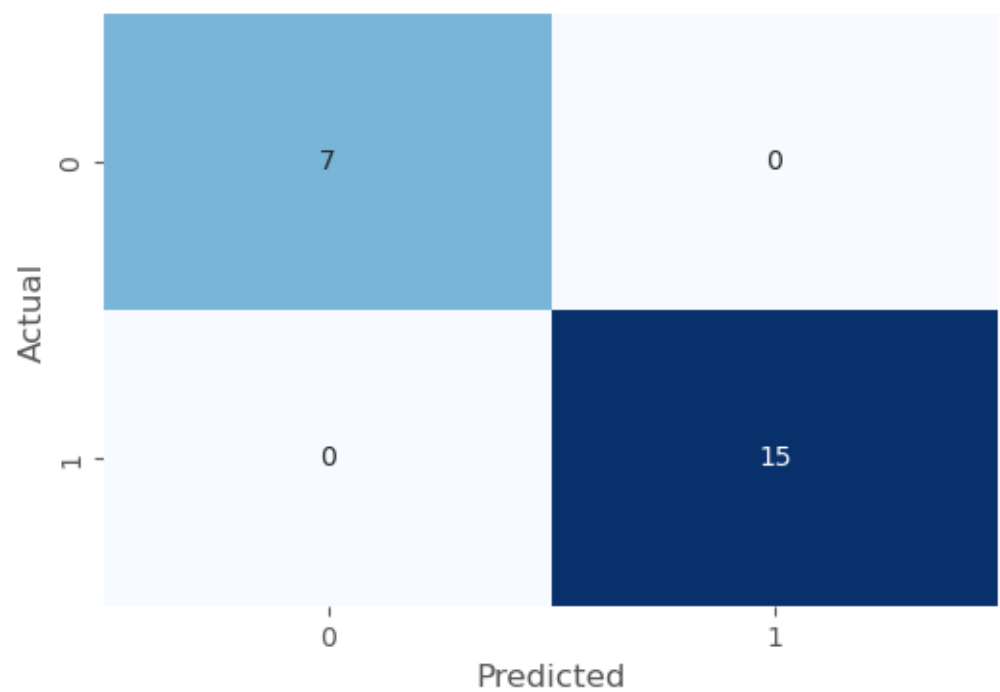
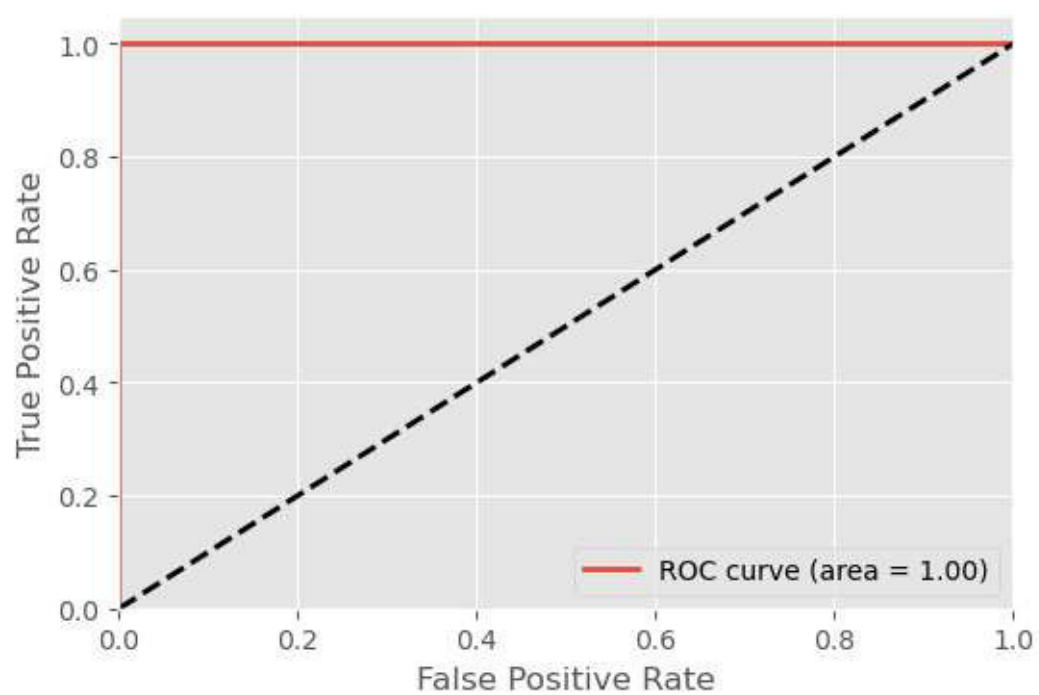


Figura 170. Curva ROC – Test (CV = 11) del modelo Gradiente



ANEXO N° 10

Resultados Juicio de Expertos (Especialista 01)

ANEXO N° 3

Instrumento de observación para el modelo basado en inteligencia artificial y oversampling

A) Árbol de decisión

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	22-01-2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	90,91%
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	94,62%
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	91,33%
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	92,86%
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	91,57%
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	80,74%

Firma:

Mg. Walter Jesus Escalante Espinoza
DNI 32763644

B) KNN

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	22-01-2026
Algoritmo	KNN

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	76,36%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	86,48%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	80,00%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precision) / (Recall + Precision)$	83,10%
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	80,21%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews\ correlation\ coefficient\ (MCC) = (TP * TN - FP * FN) / \sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}$	44,37%

Firma:

Mg. Walter Jesus Escalante Espinoza
DNI 32763644

C) Random Forest

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	22-01-2026
Algoritmo	Random Forest

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	99,55%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	99,38%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	100%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	99,68%
5	ROC AUC	Razón	$ROC AUC = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	99,91%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews correlation coefficient (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	98,96%

Firma:

Mg. Walter Jesus Escalante Espinoza
DNI 32763644

D) SVM

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	22-01-2026
Algoritmo	SVM

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	84,09%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	84,97%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	96,88%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	90,12%
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	87,50%
6	Matthews correlation coefficient (MCC)	Razón	Matthews correlation coefficient (MCC) = $(TP * TN - FP * FN) / \sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}$	56,36%

Firma:

Mg. Walter Jesus Escalante Espinoza
DNI 32763644

E) Redes Bayesianas

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	22-01-2026
Algoritmo	Redes Bayesianas

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	94,54%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	99,05%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	98,75%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	96,33%
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	96,88%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews\ correlation\ coefficient\ (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	86,26%

Firma:

Mg. Walter Jesus Escalante Espinoza
DNI 32763644

F) Gradiente

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	22-01-2026
Algoritmo	Gradiente

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	100 %
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	100 %
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	100 %
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	100 %
5	ROC AUC	Razón	$ROC AUC = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	100 %
6	Matthews correlation coefficient (MCC)	Razón	$Matthews correlation coefficient (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	100 %

Firma:

Mg. Walter Jesus Escalante Espinoza
DNI 32763644

Resultados Juicio de Expertos (Especialista 02)

ANEXO N° 3

Instrumento de observación para el modelo basado en inteligencia artificial y oversampling

A) Árbol de decisión

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	23 - 01 - 2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	90,91%
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	94,62%
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	91,33%
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	92,86%
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	91,59%
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	80,74%

Firma
Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

B) KNN

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	23-01-2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	76,36%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	86,48%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	80 %
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	83,10%
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	80,21%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews\ correlation\ coefficient\ (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	44,37%

Firma
Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

C) Random Forest

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	23-01-2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	99,55%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	99,38%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	100%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	99,68%
5	ROC AUC	Razón	$ROC AUC = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	99,91%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews correlation coefficient (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	98,96%

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

D) SVM

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	23-01-2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	84,09%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	84,97%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	96,88%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	90,12%
5	ROC AUC	Razón	ROC AUC = Área bajo la curva ROC (TP Rate vs FP Rate)	87,50%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews\ correlation\ coefficient\ (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	56,36%

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

E) Redes Bayesianas

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	23-01-2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	94,54%
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	94,05%
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	98,25%
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	96,23%
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	96,88%
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	86,26%

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

F) Gradiente

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	23-01-2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((\text{TP} + \text{TN}) / (\text{TP} + \text{TN} + \text{FP} + \text{FN})) * 100$	100%
2	Precision	Razón	$\text{Precision} = (\text{TP} / (\text{TP} + \text{FP})) * 100$	100%
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (\text{TP} / (\text{TP} + \text{FN})) * 100$	100%
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	100%
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	100%
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (\text{TP} * \text{TN} - \text{FP} * \text{FN}) / \sqrt{((\text{TP} + \text{FP})(\text{TP} + \text{FN})(\text{TN} + \text{FP})(\text{TN} + \text{FN}))}$	100%

Firma

Asesor: Dr. Alfredo Daza Vergaray
Investigador Renacyt

Resultados Juicio de Expertos (Especialista 03)

ANEXO N° 3

Instrumento de observación para el modelo basado en inteligencia artificial y oversampling

A) Árbol de decisión

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	24-01-2026
Algoritmo	Árbol de decisión

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	90,91%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	94,62%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	91,33%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	92,86%
5	ROC AUC	Razón	$ROC AUC = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	91,57%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews correlation coefficient (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	80,74%

Firma: 
Ing. Javier Luchio Utrilla Camones
DNI 45787621

B) KNN

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	24-01-2026
Algoritmo	KNN

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	76,36%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	86,48%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	80,00%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	83,10%
5	ROC AUC	Razón	$ROC AUC = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	80,21%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews correlation coefficient (MCC) = (TP * TN - FP * FN) / \sqrt{(TP+FP)(TP+FN)(TN+FP)(TN+FN)}$	44,37%

Firma: 
Ing. Javier Lucho Utrilla Camones
DNI 45787621

C) Random Forest

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	24-01-2026
Algoritmo	Random Forest

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Ítem	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	99,55%
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	99,38%
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	100%
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	99,68%
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	99,91%
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	98,96%

Firma:

Ing. Javier Lucho Utrilla Camones
DNI 45787621

D) SVM

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	24-01-2026
Algoritmo	SVM

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	84,09%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	84,97%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	96,88%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precisión) / (Recall + Precisión)$	90,12%
5	ROC AUC	Razón	$ROC AUC = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	87,50%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews correlation coefficient (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	56,36%

Firma:

Ing. Javier Luche Utrilla Camones
DNI 45787621

E) Redes Bayesianas

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	24-01-2024
Algoritmo	Redes Bayesianas

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$\text{Accuracy} = ((TP+TN)/(TP+TN+FP+FN)) * 100$	94,54%
2	Precision	Razón	$\text{Precision} = (TP/(TP+FP)) * 100$	94,05%
3	Recall (Sensibilidad)	Razón	$\text{Sensibilidad} = (TP/(TP+FN)) * 100$	98,75%
4	F1-Score	Razón	$\text{F1-Score} = 2 * (\text{Recall} * \text{Precisión}) / (\text{Recall} + \text{Precisión})$	96,33%
5	ROC AUC	Razón	$\text{ROC AUC} = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	96,88%
6	Matthews correlation coefficient (MCC)	Razón	$\text{Matthews correlation coefficient (MCC)} = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	86,26%

Firma:

Ing. Javier Lucho Utrilla Camones
DNI 45787621

F) Gradiente

Tipo de Prueba	Post Test
Empresa Investigada	Corporación de Alimentos Marítimo S.A.C.
Investigador	German David Leyva Cordova Brayan Marcelo Kano Leiva
Fecha de inicio	24-01-2026
Algoritmo	Gradiente

Matriz de confusión:

Observación		Predicción	
		Positive	Negative
	Positive	(TP) True Positive	(FN) False Negative
	Negative	(FP) False Positive	(TN) True Negative

Métricas a Evaluar:

Item	Indicador	Medida	Fórmula	Precisión
1	Accuracy (exactitud)	Razón	$Accuracy = ((TP+TN)/(TP+TN+FP+FN)) * 100$	100%
2	Precision	Razón	$Precision = (TP/(TP+FP)) * 100$	100%
3	Recall (Sensibilidad)	Razón	$Sensibilidad = (TP/(TP+FN)) * 100$	100%
4	F1-Score	Razón	$F1-Score = 2 * (Recall * Precision) / (Recall + Precision)$	100%
5	ROC AUC	Razón	$ROC AUC = \text{Área bajo la curva ROC (TP Rate vs FP Rate)}$	100%
6	Matthews correlation coefficient (MCC)	Razón	$Matthews correlation coefficient (MCC) = (TP * TN - FP * FN) / \sqrt{((TP+FP)(TP+FN)(TN+FP)(TN+FN))}$	100%

Firma:
Ing. Javier Lugo Utrilla Camones
DNI 45787621