



UNIVERSIDAD
DE PIURA

FACULTAD DE INGENIERÍA

**Desarrollo de un sistema inteligente para la adecuada
gestión de mantenimiento en una flota de máquinas
PC4000-6**

Tesis para optar el Título de
Ingeniero Mecánico - Eléctrico

**Raul Guillermo Castro Lopez
Oscar Mario Garcia Timoteo**

**Asesor:
Mgtr. Ing. Jorge Arturo Yaksetig Castillo**

Piura, octubre de 2020



A Dios por cuidarme siempre; a mi madre Julia, por su paciencia y amor incondicional; a mis hermanos Jesús y Juan, por su apoyo; y al profesor Robeldo por enseñarme matemática.

Oscar

A Dios, por darme la vida; a mis padres, por su apoyo constante; y a mi universidad, por brindarme una excelente formación como profesional y como persona.

Raul





Resumen analítico-informativo

Desarrollo de un sistema inteligente para la adecuada gestión de mantenimiento en una flota de máquinas PC4000-6

Raul Guillermo Castro Lopez

Oscar Mario Garcia Timoteo

Asesor: Mgtr. Ing. Jorge Arturo Yaksetig Castillo

Tesis.

Ingeniero Mecánico - Eléctrico

Universidad de Piura. Facultad de Ingeniería

Piura, octubre de 2020

Palabras claves: *Machine learning* / Sistema inteligente / Teorema de Naive Bayes / Algoritmo de clasificación / Aplicativo móvil / Máquina PC4000-6 / Gestión de mantenimiento / Datos de Fallas.

Introducción: El proyecto tiene como objetivo disponer de un sistema inteligente que permita automatizar y realizar, de una manera sencilla y rápida, el proceso de clasificación de los datos de fallas de una flota de máquinas PC4000-6 en sus respectivos sistemas: hidráulico, eléctrico, motriz, mecánico, lubricación y refrigeración.

Metodología: Se realiza un estudio de la máquina PC4000-6 y se clasifica en sistemas, subsistemas y componentes, bajo la norma ISO 14224:2016. Luego se procede a elaborar la data de entrenamiento, para lo cual se toma como referencia la data histórica de fallas de la flota. Para la clasificación automática de datos de fallas se emplea un algoritmo de *machine learning*, denominado Clasificador de Naive Bayes Ingenuo, el cual se basa en el teorema estadístico de Naive Bayes; asimismo, se desarrolla un aplicativo móvil que permite registrar los datos de fallas que ocurren en la flota de máquinas PC4000-6.

Resultados: Se presenta la clasificación de datos de fallas en sus respectivos sistemas realizada por el sistema inteligente y se observa que clasifica correctamente los datos de fallas en un 85%.

Conclusiones: Se logra reducir el tiempo empleado en el proceso de clasificación de datos de falla de la flota de máquinas PC4000-6, haciendo uso de un algoritmo de *machine learning*, de esta manera el personal del área de mantenimiento puede disponer de una herramienta de apoyo que le permita obtener rápidamente información adecuada a fin de buscar estrategias para mejorar la gestión de mantenimiento de la flota de máquinas PC4000-6.

Fecha de elaboración del resumen: 10 de octubre de 2020

Analytical-Informative Summary

Desarrollo de un sistema inteligente para la adecuada gestión de mantenimiento en una flota de máquinas PC4000-6

Raul Guillermo Castro Lopez

Oscar Mario Garcia Timoteo

Asesor: Mgtr. Ing. Jorge Arturo Yaksetig Castillo

Tesis.

Ingeniero Mecánico - Eléctrico

Universidad de Piura. Facultad de Ingeniería

Piura, October de 2020

Keywords: Machine learning/ Smart system/ Naive Bayes Theorem/ Classification Algorithm/ Mobile Application/ PC4000-6 machine/ Maintenance Management/ Fault Data

Introduction: The objective of the project is to have an intelligent system that allows to automate and easily carry out the failure data of a fleet of PC4000-6 machines in their respective systems: Hydraulic, electrical, motor, mechanical, lubrication and refrigeration.

Methodology: Makes a study of PC4000-6 machines and is classified into systems, subsystems and components under ISO 14224:2016 then proceed to prepare the historical data on fleet failures. For the automatic classification of failure data, a machine learning algorithm called Naive Bayes classifier is used, which is based on the Naive Bayes statistical theorem, a mobile application is also developed that allows to record the data of the failures that occur in the fleet PC4000-6 machines.

Results: The classification of fault data in their respective systems by the intelligent system is presented, managing to observe that the intelligent system correctly classifies the fault data by 85%.

Conclusions: It is possible to reduce the time spent in the process of classifying the failure of the PC4000-6 machine fleet, using the machine learning algorithm, in this way, the maintenance staff can have a support tool that allows them to quickly obtain adequate information. In order to look for strategies to improve the maintenance management of the PC4000-6 machine fleet.

Summary date: October 10th, 2020

Agradecimientos

Les agradecemos de una manera muy especial a nuestro asesor Mgtr. Ing. Jorge Yaksetig y al Ing. Julián Ortiz, quienes amablemente con su experiencia nos proporcionaron sugerencias valiosas para el desarrollo de este proyecto de tesis. Asimismo, agradecemos a la Universidad de Piura (UDEP) por ser la sede de todo el conocimiento adquirido en estos años.





Tabla de contenido

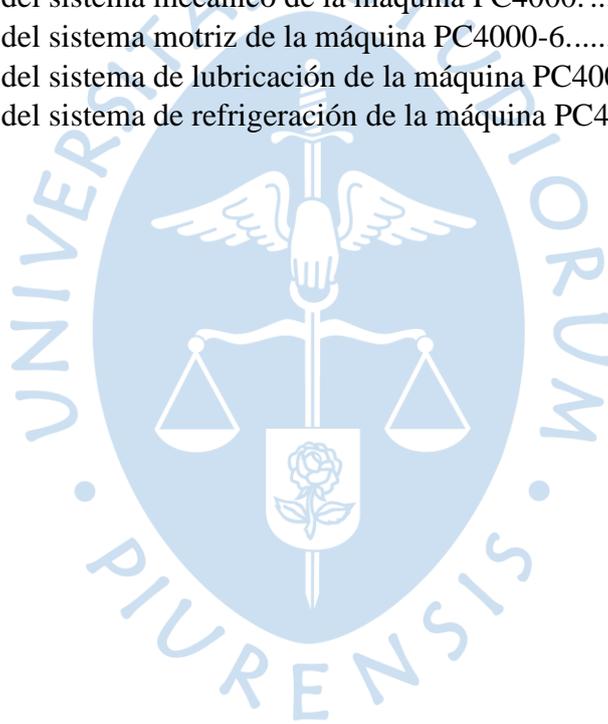
Introducción	1
Capítulo 1 Maquinaria pesada en minería	3
1.1. Justificación de la tesis	3
1.2. Objetivos de la investigación	4
1.2.1. Objetivo general.....	4
1.2.2. Objetivos específicos	4
1.3. Información general de maquinaria pesada en minería.....	4
1.4. La excavadora hidráulica	5
1.4.1. Características de la excavadora hidráulica	6
1.4.2. Componentes de la excavadora hidráulica.....	6
1.4.2.1. Chasis o <i>Carbody</i>	7
1.4.2.2. Corona de giro.....	7
1.4.2.3. Estructura superior	8
1.4.2.4. Equipo de trabajo	8
1.4.2.5. Tren de rodaje.....	9
Capítulo 2 Máquina PC4000-6	11
2.1. Características generales	11
2.2. Componentes principales de la máquina hidráulica PC4000-6.....	13
2.3. Clasificación de la máquina hidráulica bajo la norma ISO 14224:2016.....	13
2.3.1. Sistema Hidráulico.....	14
2.3.2. Sistema Eléctrico.	15
2.3.3. Sistema Mecánico	16
2.3.4. Sistema Motriz	16
2.3.5. Sistema de Lubricación.....	17
2.3.6. Sistema de Refrigeración.	18
Capítulo 3 Inteligencia artificial en el mantenimiento moderno	19
3.1. Definición de mantenimiento	19
3.1.1. Mantenimiento preventivo	20
3.1.2. Mantenimiento correctivo	20
3.1.3. Mantenimiento proactivo	21
3.2. Análisis de datos de fallas de la flota de máquinas PC4000-6.....	21
3.2.1. Diagrama de Pareto (Análisis ABC).....	22
3.3. Introducción a la Inteligencia artificial. <i>Machine learning</i> y <i>Deep learning</i>	23
3.4. Mantenimiento inteligente.....	26

3.5. Ventajas del mantenimiento inteligente	27
Capítulo 4 Sistema inteligente para recolectar y clasificar datos de fallas.....	29
4.1. Desarrollo del sistema inteligente	29
4.1.1. Desarrollo del aplicativo móvil.....	31
4.1.2. Desarrollo del sistema de clasificación de datos de fallas	34
4.2. Resultados	50
Conclusiones	53
Recomendaciones	55
Referencias bibliográficas.....	57
Apéndices	63
Apéndice A. Manual de usuario del aplicativo móvil PC4000-6	65
Apéndice B. Código completo del sistema de clasificación	66
Anexos	69
Anexo A. Logistic Regression	71
Anexo B. Naive Bayes.....	73
Anexo C. Support Vector Machine (SVM)	75
Anexo D. Decision Tree	77



Lista de tablas

Tabla 1. Dimensiones de la máquina hidráulica	12
Tabla 2. Componentes principales de la máquina hidráulica.....	13
Tabla 3. Clasificación del sistema hidráulico de la máquina PC4000-6.....	14
Tabla 4. Clasificación del sistema eléctrico de la máquina PC4000-6.	15
Tabla 5. Clasificación del sistema mecánico de la máquina PC4000.	16
Tabla 6. Clasificación del sistema motriz de la máquina PC4000-6.....	17
Tabla 7. Clasificación del sistema de lubricación de la máquina PC4000-6.	18
Tabla 8. Clasificación del sistema de refrigeración de la máquina PC4000-6.....	18





Lista de figuras

Figura 1. Máquina hidráulica PC4000-6.....	1
Figura 2. Etapa de carguío y acarreo.....	5
Figura 3. <i>Front Shovel</i> (izquierda) y <i>Backhoe</i> (derecha).....	5
Figura 4. Traslado de una <i>Front Shovel</i>	6
Figura 5. Componentes de una excavadora hidráulica.....	7
Figura 6. <i>Carbody</i>	7
Figura 7. Corona de giro.....	8
Figura 8. Estructura superior.....	8
Figura 9. Equipo de trabajo.....	9
Figura 10. Tren de rodaje.....	9
Figura 11. Dimensiones de la máquina hidráulica.....	12
Figura 12. Componentes principales de la máquina.....	13
Figura 13. Clasificación del mantenimiento.....	19
Figura 14. Personal de mantenimiento.....	20
Figura 15. Costo total correctivo por trabajos con y sin planificación.....	21
Figura 16. Diagrama de Pareto con respecto al tiempo (horas) de parada por falla.....	23
Figura 17. Inteligencia artificial en la actualidad.....	23
Figura 18. Aplicación de la IA.....	24
Figura 19. Aplicaciones del machine learning.....	25
Figura 20. Aplicaciones de deep learning.....	26
Figura 21. Descripción arquitectónica del conjunto de herramientas i-maintenance.....	26
Figura 22. Desarrollo de sistemas inteligentes.....	27
Figura 23. Operaciones remotas.....	28
Figura 24. Esquema general del Sistema automático de Clasificación de datos de fallas.....	30
Figura 25. Smartphones.....	31
Figura 26. Aplicativo móvil instalado en un teléfono móvil Android.....	32
Figura 27. Pantalla de inicio(izquierda) y pantalla de registro(derecha) del aplicativo móvil.....	32
Figura 28. Pantalla de registro del aplicativo móvil.....	33
Figura 29. Selección del código de la máquina PC4000-6.....	33
Figura 30. Almacenamiento de datos de fallas en la nube.....	34
Figura 31. Extracto de la base de datos PC4000-6.....	35
Figura 32. Sistemas de la máquina PC4000-6.....	36
Figura 33. Set de datos (Entrenamiento y prueba).....	36
Figura 34. Limpieza de datos.....	37
Figura 35. Ejemplo sencillo de vectorización.....	38
Figura 36. Representación del conjunto de datos y margen.....	39

Figura 37. Decision Tree	40
Figura 38. Análisis con diferentes algoritmos de Machine learning.....	42
Figura 39. Puntuación obtenida.....	44
Figura 40. Código de librerías importadas	45
Figura 41. Data de entrenamiento	45
Figura 42. Data a clasificar	46
Figura 43. DataFrame.....	46
Figura 44. Clasificación de la data usando Textblob	47
Figura 45. Código para agregar la clasificación.....	47
Figura 46. Resultado obtenido.	47
Figura 47. Tiempo de fallas en cada equipo.....	48
Figura 48. Tiempo de fallas en cada sistema	48
Figura 49. Fallas clasificadas	48
Figura 50. Código para crear el diagrama de Pareto	49
Figura 51. Gráfica de Pareto	49
Figura 52. Código para generar la gráfica dinámica.	49
Figura 53. Gráficas dinámicas de cada equipo.....	50
Figura 54. Resultados.....	51



Introducción

El presente proyecto de tesis tiene como objetivos desarrollar un sistema inteligente para recolectar datos de las fallas de una flota de máquinas PC4000-6, que se encuentra operando en una instalación minera del Perú; y automatizar la clasificación de estas fallas de acuerdo con sus respectivos sistemas (hidráulico, eléctrico, motriz, mecánico, lubricación y refrigeración), para mejorar la gestión de los datos de las fallas de mantenimiento.

El primer capítulo abarca los siguientes temas: justificación y objetivos (general y específicos) de la investigación, información general de maquinaria pesada en minería, y conceptos de una excavadora hidráulica.

El segundo capítulo describe las características más relevantes de la máquina PC4000-6 (ver figura 1) y, con la finalidad de gestionar la información de las fallas para una adecuada clasificación, se elabora una taxonomía en sistemas, subsistemas y componentes, siguiendo la norma ISO 14224:2016.

El tercer capítulo abarca los conceptos generales del mantenimiento, análisis de los datos de fallas de la flota de máquinas PC4000-6, introducción a la inteligencia artificial, *machine learning* y *deep learning*. Además, se explica el mantenimiento actual (mantenimiento inteligente) y sus ventajas.

Finalmente, en el cuarto capítulo se desarrolla el sistema inteligente para recolectar y clasificar los datos de fallas de la flota de máquina PC4000-6 en sus respectivos sistemas, con la finalidad de mejorar la gestión de datos de mantenimiento.



Figura 1. Máquina hidráulica PC4000-6

Fuente: <https://www.komatsulatinoamerica.com/>

Visita: 09/08/2019



Capítulo 1

Maquinaria pesada en minería

El presente capítulo abarca los siguientes temas: justificación y objetivos (objetivo general y objetivos específicos) de esta investigación, información general de maquinaria pesada en minería y conceptos de una excavadora hidráulica.

1.1. Justificación de la tesis

El Perú es un país minero, lo que se refleja en el porcentaje de sus exportaciones (aproximadamente, un 61% del total de exportaciones a los distintos mercados internacionales), y en su aporte al PBI (10 % del PBI nacional), según el Anuario Minero 2018 elaborado por el Ministerio de Energía y Minas.

Las actividades de carguío y acarreo de material en una operación minera son muy importantes, ya que cualquier falla en uno de los componentes (activos) de estas actividades puede generar un retraso en la producción del mineral. Con el fin de alcanzar las metas de producción anual, es necesario preservar de manera eficaz y eficiente tales activos y disponer de apropiados planes de mantenimiento preventivo que aseguren su disponibilidad. Por tanto, para cumplir con su objetivo, el planificador, responsable de elaborar los planes debe disponer de información confiable que le permita identificar, mediante herramientas de calidad, los componentes críticos para aplicar las estrategias adecuadas y optimizar la disponibilidad de los equipos.

El planificador requiere que el personal técnico o administrativo registre correctamente gran parte de esa información. De lo contrario, se generan dificultades para clasificar las fallas de los equipos y, en consecuencia, se aplican planes de mantenimiento poco eficientes.

Con la finalidad de solucionar estos problemas, en el proyecto de tesis se desarrolla un sistema de mantenimiento inteligente para automatizar el proceso de clasificación de datos de fallas de la flota de máquinas PC4000-6; lo que permitirá mejorar la gestión de datos de fallas de mantenimiento y crear planes de mantenimiento eficaces.

1.2. Objetivos de la investigación

1.2.1. Objetivo general

- Desarrollar un sistema de mantenimiento inteligente para automatizar el proceso de clasificación de datos de fallas en una flota de máquinas PC4000-6.

1.2.2. Objetivos específicos

- Conocer a profundidad las características y especificaciones técnicas de la máquina PC4000-6.
- Generar una óptima clasificación de los sistemas, subsistemas y componentes, teniendo en cuenta la norma ISO 14224:2016¹.
- Utilizar el diagrama de Pareto, como herramienta de calidad, a fin de realizar el análisis de los datos de la flota de la máquina PC4000-6 para conocer las fallas más recurrentes de estos activos.
- Aplicar conocimientos de *machine learning* para automatizar los procesos de recolección y clasificación de datos de fallas.

1.3. Información general de maquinaria pesada en minería

La maquinaria pesada es un conjunto de equipos automotores que se emplea para realizar diversas actividades como transporte de objetos grandes y pesados, demoliciones, excavaciones, entre otras. Desde que ingresaron al mercado, estas máquinas han mejorado la productividad de las tareas y han permitido realizar obras en un menor tiempo y con menos personal.

El Perú cuenta con importantes proyectos mineros que emplean diferentes tipos de maquinaria pesada. Algunas de las empresas mineras con operaciones en el Perú son Compañía Minera Antamina S.A, Compañía de Minas Buenaventura S.A.A, Sociedad Minera Cerro Verde S.A.A, Minera las Bambas S.A, Compañía Minera Antapaccay S.A, Minera Barrick S.A, Compañía Minera Miski Mayo S.R.L, entre otras.

El proceso de producción de una mina está compuesto por varias etapas, una de ellas es la etapa de carguío y acarreo (ver figura 2), en la que se emplean palas y camiones de gran tonelaje para la carga y el traslado del material. A nivel nacional los principales proveedores de maquinaria pesada para minería son Ferreyros S.A y Komatsu-Mitsui Maquinarias Perú S.A.

¹ ISO 14224:2016: Norma orientada a la recolección e intercambio de datos de confiabilidad y mantenimiento de equipos.



Figura 2. Etapa de carguío y acarreo

Fuente: <https://www.antamina.com/proceso-productivo/>
 Visita: 09/08/2019

Para que las máquinas puedan funcionar correctamente, es necesaria una adecuada gestión que determine las estrategias aplicadas por medio de la planificación, el control y los procesos de mejora continua del mantenimiento.

1.4. La excavadora hidráulica

La excavadora hidráulica es un tipo de maquinaria pesada autopropulsada que puede girar 360° sobre su propio eje, y que puede desplazarse sobre orugas o ruedas. Esta máquina se creó en Alemania en los años 50 con el objetivo de mover tierra para realizar zanjas. Gracias a su desarrollo a lo largo de los años, se desempeña con eficacia en la minería y en la construcción.

Existen dos tipos de excavadoras hidráulicas, frontales y retros, que difieren en la manera de excavar (ver figura 3).

La excavadora hidráulica frontal (*Front Shovel*): su configuración permite que el cucharón realice movimientos de abajo hacia arriba, por lo que son útiles en la minería para cargar material por encima del nivel del suelo.

La excavadora hidráulica retro (*Backhoe*): su configuración diferente permite que el cucharón realice movimientos de arriba hacia abajo, con lo que es posible llegar a cotas bajas. En las actividades de minería, esta máquina se utiliza para cargar material por debajo del nivel del suelo.

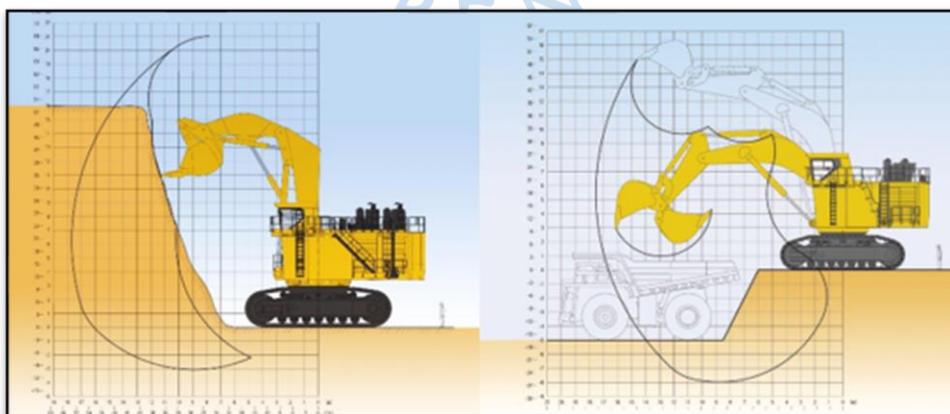


Figura 3. *Front Shovel* (izquierda) y *Backhoe* (derecha)

Fuente: <https://www.komatsulatinoamerica.com/>
 Visita: 15/08/2019

1.4.1. Características de la excavadora hidráulica

La característica principal de la excavadora hidráulica es que se puede trasladar por sus propios medios de un lado a otro en operaciones mineras. Sin embargo, hay que tener en cuenta que, si la distancia de traslado es larga, lo recomendable es utilizar herramientas, como un camión remolcador para sostener el equipo de trabajo, y patines de apoyo para evitar dañar el tren de rodaje y disminuir el tiempo de traslado (ver figura 4).



Figura 4. Traslado de una Front Shovel
Fuente: <https://www.sleipner.fi/>
Visita: 16/08/2019

Características de la excavadora hidráulica:

- Versatilidad, confiabilidad y alto rendimiento en operaciones de minería.
- Elevada fuerza de penetración y de excavación.
- Control preciso del brazo hidráulico al momento de realizar el carguío de camiones, que permite una buena distribución del material.
- Debido a su sistema de rotación que le permite girar sobre su propio eje, requiere poco espacio para operar.
- Por su durabilidad y eficiencia, representan una buena opción para proyectos de minería.
- Dependiendo del terreno a excavar se pueden incorporar piezas de desgaste más resistentes.
- Poseen motores eficientes de gran potencia.
- Diseño compacto resistente a fuerzas de flexión y torsión.
- Gran estabilidad a la hora de realizar sus actividades.

1.4.2. Componentes de la excavadora hidráulica

Los componentes de la excavadora hidráulica se agrupan, principalmente, en: chasis o *carbody*, corona de giro, estructura superior, equipo de trabajo, y tren de rodaje (ver figura 5).



Figura 5. Componentes de una excavadora hidráulica
 Fuente: <https://www.komatsulatinoamerica.com/>
 Visita: 18/08/2019
 Leyenda: elaboración propia

1.4.2.1. Chasis o *Carbody*

Soporta el peso de la excavadora hidráulica y permite transmitir las cargas de estructura superior al tren de rodaje. Su estructura tiene forma de “H” y aloja en su parte central la corona de giro (ver figura 6).

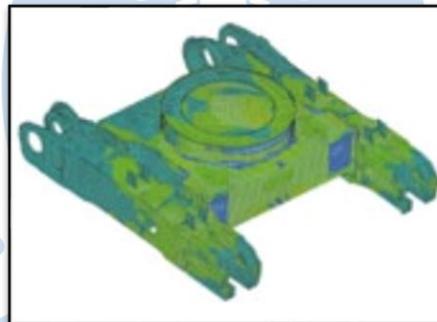


Figura 6. *Carbody*
 Fuente: <https://www.komatsulatinoamerica.com/>
 Visita: 18/08/2019

1.4.2.2. Corona de giro

Sirve de apoyo de la estructura superior sobre el *carbody*, por lo que es uno de los componentes que soporta mayor tensión. También permite la rotación de la estructura superior respecto al *carbody*, el cual permanece fijo sobre el suelo (ver figura 7).



Figura 7. Corona de giro
Fuente: <https://www.komatsulatioamerica.com/>
Visita: 18/08/2019

1.4.2.3. Estructura superior

Llamada también superestructura. Es el componente donde va fija la cabina, el motor Diésel, el depósito de combustible, el tanque hidráulico, el equipo de trabajo y todos los componentes para el funcionamiento de la máquina. Así mismo, absorbe los esfuerzos transmitidos por el equipo de trabajo y en la parte posterior lleva un contrapeso para mejorar la estabilidad de la máquina (ver figura 8).



Figura 8. Estructura superior
Fuente: <https://www.komatsulatioamerica.com/>
Visita: 18/08/2019

1.4.2.4. Equipo de trabajo

Está compuesto por el cucharón (*Bucket*), el brazo (*Stick*) y la pluma (*Boom*), articulados entre sí y accionados hidráulicamente. Los cilindros hidráulicos del cucharón (cilindros del *Bucket*) permiten obtener la fuerza de excavación; los cilindros hidráulicos del brazo (cilindros del *Stick*), alcanzar la fuerza de penetración requerida; y los cilindros de la pluma (cilindros del *Boom*), realizar el movimiento vertical (ver figura 9).

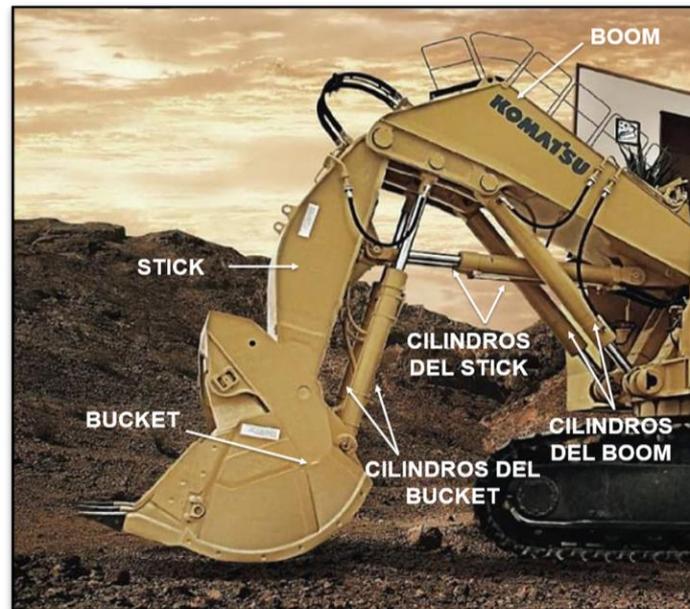


Figura 9. Equipo de trabajo

Fuente: <https://www.facebook.com/KomatsuPeru/>

Visita: 20/08/2019

Leyenda: elaboración propia

1.4.2.5. Tren de rodaje

Permite el desplazamiento de la máquina sobre cualquier tipo de terreno, gracias a su capacidad de adherencia y de agarre. Está compuesto por las orugas o cadenas, ruedas tensoras, rodillos inferiores, rodillos superiores y ruedas dentadas (ver figura 10).

Para elegir el tipo de oruga (estándar, de baja presión y reforzados) se debe tener en cuenta las condiciones de trabajo por realizar y las condiciones del terreno.

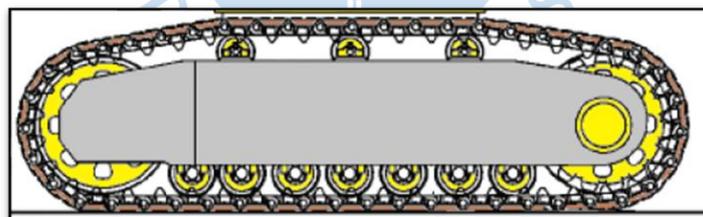


Figura 10. Tren de rodaje

Fuente: <https://www.komatsulatinoamerica.com/>

Visita: 21/08/2019.



Capítulo 2

Máquina PC4000-6

El presente capítulo describe las características más relevantes de la máquina PC4000-6 y, con la finalidad de gestionar la información de las fallas para la adecuada clasificación, se elabora una taxonomía de esta clasificándola en sistemas, subsistemas y componentes, siguiendo la norma ISO 14224:2016.

2.1. Características generales²

La máquina PC4000-6 es un equipo que se utiliza en las operaciones de minería y su función es el carguío de camiones de aproximadamente 190 t de carga útil. Tal como se indicó en el capítulo 1, existen dos versiones, *Backhoe* y *Front Shovel*, cuya diferencia es la disposición del equipo de trabajo.

Está equipada con un motor Diésel de 16 cilindros, 4 tiempos, refrigerado por agua; posee una potencia nominal de 1875 hp a 1800 rpm con control electrónico e inyección directa y, según la versión, tiene un sistema de emisiones TIER³ 1 o 2.

Tiene un subsistema de enfriamiento mediante radiadores de alta capacidad, apoyados por ventiladores que se activan hidráulicamente y proveen suficiente volumen de aire para el intercambio del calor, de esta manera, se garantiza la eficiencia del radiador.

Es autopropulsada por motores hidráulicos con una velocidad de desplazamiento máxima de 2.1 km/h, pesa aproximadamente 399 t y el volumen del cucharón estándar es de 22 m³.

Su sistema eléctrico es de 24 V y alimenta a todos los componentes electrónicos de la máquina hidráulica, tiene un alternador de 260 A, posee 14 focos de xenón de alta potencia y luces de servicio estándar en toda la plataforma.

² Página web de Komatsu Latinoamérica (www.komatsulatinamerica.com). Visita: 21/08/2019

³ Norma de la agencia de protección ambiental de los Estados Unidos (EPA) que regula la emisión de contaminantes producidos por los motores de combustión.

El tren de rodaje tiene 49 zapatas de acero fundido a cada lado, 3 rodillos superiores, 7 rodillos inferiores en ambos lados, 1 rueda motriz y 1 rueda guía con poleas tensoras. El sistema de giro posee 2 motores hidráulicos y la velocidad máxima de giro es de 4 rpm.

Para que el sistema hidráulico funcione correctamente, el tren de potencia consta de un motor Diésel y un sistema de toma de fuerza (PTO), el cual acciona 4 bombas de igual similitud que succionan aceite hidráulico de un sistema no presurizado.

Tiene un sistema central de lubricación automática de grasa, la cual es suministrada por un barril rellenable de 200 L. Además, tiene otro sistema idéntico que sirve para suministrar lubricante a los dientes de la corona de giro por medio del piñón de lubricación.

Esta máquina hidráulica es un equipo de grandes dimensiones, que se muestran en la tabla 1 y en la figura 11.

Tabla 1. Dimensiones de la máquina hidráulica

A	1.2 m	J	3.1 m
B	1.5 m	K	7.6 m
C_A	6.8 m	L	8.3 m
C_B	7.1 m	M_A	4.7 m
D	2.5 m	M_B	3.9 m
E	0.9 m	N	3.5 m
F	3.4 m	O	8.3 m
G	6.7 m	P	6.1 m
H	8.8 m	P_R	6.5 m
I	3.1 m		

Fuente: <https://www.komatsulatioamerica.com/>
Visita: 21/08/2019.

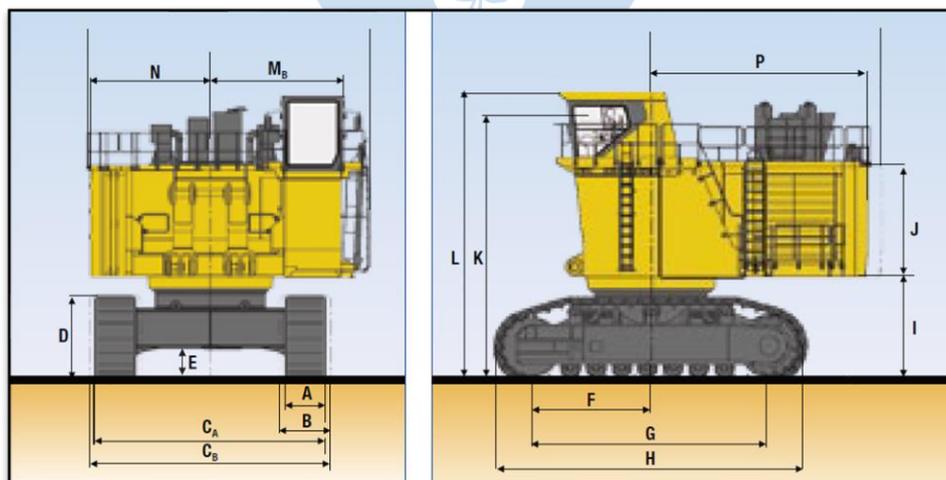


Figura 11. Dimensiones de la máquina hidráulica

Fuente: <https://www.komatsulatioamerica.com/>
Visita: 21/08/2019

2.2. Componentes principales de la máquina hidráulica PC4000-6

En la tabla 2 se presenta el nombre de los componentes principales con su respectiva identificación, y en la figura 12 se presenta la ubicación de cada componente.

Tabla 2. Componentes principales de la máquina hidráulica

1	<i>Boom.</i>	9	Tablero de distribución eléctrica.
2	Cilindros del <i>Boom.</i>	10	Corona de giro.
3	<i>Stick.</i>	11	Tanque de combustible.
4	Cilindros del <i>Stick.</i>	12	<i>Carbody.</i>
5	<i>Bucket backwall.</i>	13	Escalera hidráulica.
6	Cilindros del <i>Bucket.</i>	14	Contrapeso.
7	<i>Clam</i> ⁴ .	15	Motor diésel.
8	Cilindros del <i>Bucket clam.</i>	16	Cabina.

Fuente: <https://www.komatsulatinoamerica.com/>

Visita: 22/08/2019.

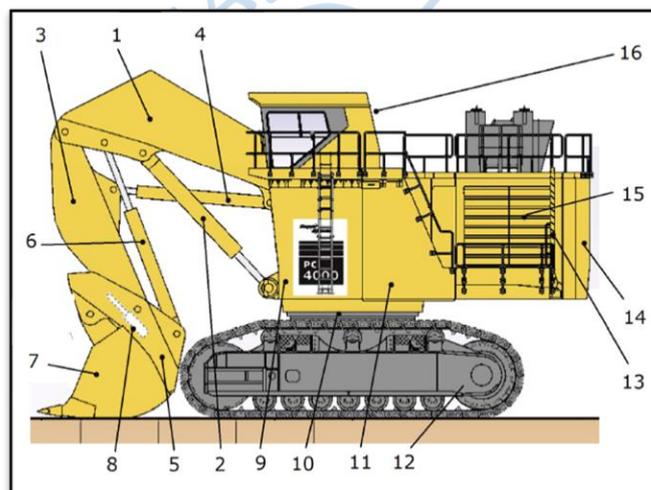


Figura 12. Componentes principales de la máquina

Fuente: <https://www.komatsulatinoamerica.com/>

Visita: 22/08/2019.

2.3. Clasificación de la máquina hidráulica bajo la norma ISO 14224:2016

La Norma ISO 14224:2016 brinda una guía para la especificación, recolección y aseguramiento de la calidad de los datos de confiabilidad y mantenimiento, en un formato estándar para las áreas de perforación, producción, refinación transporte de petróleo y gas natural. Sin embargo, sus criterios pueden extenderse perfectamente a otros sectores económicos. Ajustarse a esta norma permite a las empresas compararse con otras de similar actividad.

Además, ayuda a establecer una taxonomía de la planta industrial desde su actividad principal, pasando por sistemas, subsistemas, hasta el *ítem* mantenible (componente que, cuando falla, no puede ser reparado sino cambiado). Define también los modos de falla para un

⁴ Este elemento se encuentre presente únicamente en la *Front Shovel*.

significativo número de equipos. La profundidad del detalle del análisis debe ser establecida en función de las características de la organización.

Se aplica la guía de la Norma ISO 14224:2016 a la máquina PC4000-6, es decir, se realiza su taxonomía como si fuera una planta, jerarquizándola en los siguientes sistemas: hidráulico, eléctrico, mecánico, motriz, lubricación y refrigeración.

2.3.1. Sistema Hidráulico.

Es el encargado de proveer a la máquina PC4000-6 la potencia necesaria para desplazarse, rotar, accionar el equipo de trabajo, entre otras funciones que permitan su correcto funcionamiento.

Tabla 3. Clasificación del sistema hidráulico de la máquina PC4000-6.

Máquina PC4000-6	Sistema hidráulico	Subsistema de bombas hidráulicas	<ul style="list-style-type: none"> • Bombas principales (total = 04). • Bombas secundarias (total = 04).
		Subsistema de motores hidráulicos	<ul style="list-style-type: none"> • Motor de giro izquierdo. • Motor de giro derecho. • Motor de traslación izquierdo. • Motor de traslación derecho. • Motor inferior del ventilador del enfriador de aceite hidráulico. • Motor superior del ventilador del enfriador de aceite hidráulico. • Motor del ventilador del radiador.
		Subsistema de cilindros hidráulicos	<ul style="list-style-type: none"> • Cilindros hidráulicos del <i>Boom</i>. • Cilindros hidráulicos del <i>Stick</i>. • Cilindros hidráulicos del <i>Bucket</i>. • Cilindros hidráulicos del <i>Bucket clam</i>. • Cilindro hidráulico de la escalera principal. • Cilindro hidráulico del brazo de servicio.
		Subsistema de tuberías y mangueras	<ul style="list-style-type: none"> • Tuberías metálicas. • Mangueras de alta presión (1er arco y 2do arco). • Mangueras de baja presión (Pilotaje y retorno).
		Subsistema de tanque hidráulico	<ul style="list-style-type: none"> • Filtro respirador. • Acople de drenaje. • Filtros de retorno. • Filtros de drenaje. • Tubo colector de retorno (Tubo J) • Conector de drenaje.
		Subsistema de válvulas	<ul style="list-style-type: none"> • Válvulas de control de presión (MRV / SRV). • Válvulas de control direccional. • Válvulas de control de flujo (Throttle). • Válvulas anticavitacionales (ACV).
		Subsistema de bloques de control	<ul style="list-style-type: none"> • Portador de los bloques de control. • Bloques de control principal (total =4). • Filtros de línea de alta presión.
		Subsistema de enfriamiento del aceite hidráulico	<ul style="list-style-type: none"> • Enfriador del aceite hidráulico superior exterior. • Enfriador de aceite hidráulico superior interior. • Enfriador de aceite hidráulico inferior exterior. • Enfriador de aceite hidráulico inferior interior.
		Otros	<ul style="list-style-type: none"> • Junta Giratoria o <i>Rotary Joint</i>. • <i>O-rings</i>. • Bridas. • Sellos.

Fuente: Elaboración propia.

2.3.2. Sistema Eléctrico.

Es el encargado de permitir el arranque y el control de todas las funciones de la máquina PC4000-6, por medio de señales eléctricas.

Asimismo, agrupa diversos componentes eléctricos y electrónicos como sensores, cables, actuadores, controladores, interruptores, entre otros.

Tabla 4. Clasificación del sistema eléctrico de la máquina PC4000-6.

Máquina PC4000-6	Sistema eléctrico	Subsistema de arranque	<ul style="list-style-type: none"> • 04 baterías principales de 24 V. • Llave de arranque. • Alternador. • Arrancador.
		Subsistema de cableado	<ul style="list-style-type: none"> • Cableado principal (Arranque) • Cableado secundario.
		Subsistema de control y monitoreo (Cabina X2)	<ul style="list-style-type: none"> • 02 baterías secundarias (cabina X2). • Controlador electrónico de límite de carga (RC4). • Controlador electrónico VHMS (Monitoreo). • Interruptores de control. • Sensores electrónicos de regulación. • Sensores electrónicos de control. • Sensores electrónicos de monitoreo. • Transductores electrónicos de regulación. • Transductores electrónicos de control. • Transductores electrónicos de monitoreo.
		Subsistema de actuadores eléctricos	<ul style="list-style-type: none"> • Solenoide en válvula de control de presión. • Solenoide en válvula de control de direccional. • Solenoide en válvula de control de flujo.
		Subsistema de protección	<ul style="list-style-type: none"> • Interruptores disyuntores de protección. • Fusibles. • Contactores y relés.
		Otros	<ul style="list-style-type: none"> • Pulsadores. • Conectores. • Luminarias. • Monitor VHMS. • Amplificadores. • Conversores.

Fuente: Elaboración propia.

2.3.3. Sistema Mecánico

Es el encargado de transmitir o transformar movimientos. Está formado por diversos componentes que en conjunto permiten realizar una función determinada de la máquina PC4000-6.

Tabla 5. Clasificación del sistema mecánico de la máquina PC4000.

Máquina PC4000-6	Sistema mecánico	Subsistema del equipo de trabajo	<ul style="list-style-type: none"> • Estructura del <i>Stick</i>. • Estructura del <i>Boom</i>. • Estructura del <i>Bucket</i>. • Pines y pasadores. • Bocinas.
		Subsistema de Tren de rodaje	<ul style="list-style-type: none"> • Ruedas dentadas (Rueda motriz o <i>Sprocket</i>). • Ruedas tensoras (Rueda guía). • Orugas (cadenas o zapatas). • Pasadores de conexión. • Rodillos superiores. • Rodillos Inferiores.
		Subsistema de Giro	<ul style="list-style-type: none"> • Corona de giro o tornamesa. • Pernos y tuercas de sujeción.
		Subsistema de estructuras	<ul style="list-style-type: none"> • <i>Carbody</i>. • Cabina del operador. • Cabina X2. • Carcasa de la estructura superior (Superestructura). • Escalera principal (acceso cabina). • Escalera de emergencia (acceso cabina). • Tanque de aceite hidráulico. • Tanque de combustible. • Tornillos y tuercas de sujeción. • Rodamientos.
		Otros	<ul style="list-style-type: none"> • Mallas. • Abrazaderas. • Tapas. • Carcasas. • Caja de engranajes (PTO).

Fuente: Elaboración propia.

2.3.4. Sistema Motriz

Es el responsable de proveer a la máquina PC4000-6 de energía mecánica, para ello utiliza un motor diésel que, por medio de un proceso de combustión, transforma la energía del combustible en energía mecánica.

Tabla 6. Clasificación del sistema motriz de la máquina PC4000-6.

Máquina PC4000-6	Sistema motriz	Subsistema de combustible	<ul style="list-style-type: none"> • Bomba de combustible. • Filtros de combustible. • Tuberías de combustible. • Inyectores. • Válvulas.
		Subsistema de lubricación (Centinel)	<ul style="list-style-type: none"> • Aceite de lubricación. • Bombas de aceite. • Carter de aceite de retorno. • Tuberías de aceite principal. • Tuberías de aceite secundarias. • Filtros. • Válvulas.
		Subsistema de enfriamiento	<ul style="list-style-type: none"> • Líquido refrigerante. • Ventilador. • Radiadores principales. • Radiadores secundarios. • 01 bomba de agua. • 01 bomba refrigerante. • Válvulas. • Filtros.
		Subsistema de admisión y escape	<ul style="list-style-type: none"> • 04 turbocompresores. • 02 cajas de filtros primarios. • 02 cajas de filtros secundarios. • Múltiple de admisión. • Múltiple de escape. • Mangueras. • Abrazaderas. • Tuberías metálicas. • Cámara de combustión. • Tubo de escape. • Válvulas de admisión. • Válvulas de escape.
		Subsistema potencia	<ul style="list-style-type: none"> • Cigüeñal. • 16 cilindros (8 derechos-8 izquierdos). • 2 ejes de levas. • Caja de distribución. • Volante de inercia. • Dámper de motor.
		Subsistema de control del motor	<ul style="list-style-type: none"> • Módulo de control electrónico del motor (ECM o MTC). • Sensores.
		Otros	<ul style="list-style-type: none"> • Mirillas nivel líquido (refrigerante, lubricación). • Bloque del motor. • Tanque de reserva.

Fuente: Elaboración propia.

2.3.5. Sistema de Lubricación.

Es el encargado de proveer a la máquina PC4000-6 la grasa necesaria para realizar un proceso adecuado de lubricación que permita evitar el desgaste de los componentes mecánicos.

Tabla 7. Clasificación del sistema de lubricación de la máquina PC4000-6.

Máquina PC4000-6	Sistema de lubricación	Subsistema de Lubricación Central (CLS)	<ul style="list-style-type: none"> • Línea de relleno de grasa del adaptador del brazo de servicio al filtro de grasa. • Filtro de grasa. • Línea de relleno de grasa del filtro de grasa al contenedor de grasa. • Contenedor de grasa. • Bomba de grasa CLS. • Indicador de presión de grasa. • Línea de suministro de grasa a los inyectores para la lubricación de pasadores y rodamientos. • Inyectores. • Filtro respirador. • Mangueras de grasa liviana.
		Subsistema de lubricación del piñón de la corona de giro (SLS)	<ul style="list-style-type: none"> • Filtro de grasa. • Línea de rellenado de grasa desde el filtro de grasa al contenedor de grasa. • Contenedor de grasa. • Bomba de grasa SLS. • Indicador de presión de grasa. • Línea de suministro de grasa a los inyectores para la lubricación del piñón de la corona de giro. • Inyectores. • Indicador del nivel de grasa. • Filtro respirador. • Mangueras de grasa pesada.

Fuente: Elaboración propia.

2.3.6. Sistema de Refrigeración.

Es el responsable de mantener la temperatura adecuada en la máquina PC4000-6 para el correcto funcionamiento de esta.

Tabla 8. Clasificación del sistema de refrigeración de la máquina PC4000-6.

Máquina PC4000-6	Sistema de refrigeración	Subsistema principal	<ul style="list-style-type: none"> • Compresor. • Condensador. • Evaporador.
		Subsistema secundario	<ul style="list-style-type: none"> • Acumulador. • Correas motrices. • Depósito refrigerante. • Filtros. • Mirilla refrigerante. • Receptor secador. • Tuberías y mangueras de líquido refrigerante a baja presión. • Tuberías y mangueras de líquido refrigerante a alta presión. • Válvulas. • Ventilador – condensador.

Fuente: Elaboración propia.

Capítulo 3

Inteligencia artificial en el mantenimiento moderno

El presente capítulo abarca los conceptos generales del mantenimiento, el análisis de los datos de fallas de la flota de máquinas PC4000-6 y una introducción a la inteligencia artificial, machine learning y deep learning. Además, se explica el mantenimiento actual (mantenimiento inteligente) y sus ventajas.

3.1. Definición de mantenimiento

El mantenimiento se define como: “la combinación de todas las acciones técnicas y de gestión que tienen la intención de retener un *ítem*⁵, restaurarlo a un estado en que pueda realizar lo requerido” (ISO 14224:2016).

Se clasifica en tres categorías básicas: **mantenimiento preventivo**, antes que ocurra la falla en el *ítem*; **mantenimiento correctivo**, después que sucede la falla en el *ítem*; y el **mantenimiento proactivo**, para identificar la causa de la falla. En la figura 13 se aprecia la clasificación del mantenimiento.

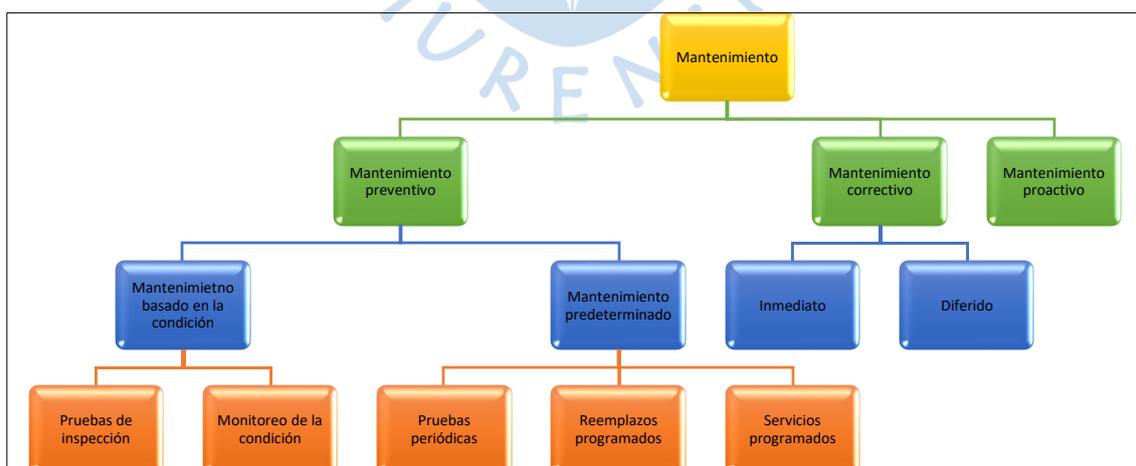


Figura 13. Clasificación del mantenimiento
Fuente: Norma ISO 14224.

⁵ Término general para indicar un equipo, obra o instalación

3.1.1. Mantenimiento preventivo

“Mantenimiento llevado a cabo para mitigar la degradación y reducir la probabilidad de falla” (ISO 14224:2016); con el fin de reducir los tiempos muertos por paradas no programadas. (ver figura 14).



Figura 14. Personal de mantenimiento
Fuente: <https://www.facebook.com/KomatsuPeru/>
Visita: 01/11/2019

Hay dos tipos de mantenimiento preventivo: basado en la condición y predeterminado.

El mantenimiento basado en la condición consiste en inspeccionar ciertos parámetros del *ítem* (máquina hidráulica PC4000-6), con el objetivo de detectar anticipadamente las posibles fallas sintomáticas⁶ y sugerir planes de acción (restaurativos y mejorativos), para prevenir la falla funcional⁷.

El mantenimiento predeterminado, abarca labores que han sido programadas (pruebas programadas, reemplazos y servicios programados) con base en las horas recomendadas por el fabricante, los especialistas, el personal técnico, entre otros.

3.1.2. Mantenimiento correctivo

“Mantenimiento llevado a cabo después de la detección de una falla hasta completar la restauración”. (ISO 14224:2016).

Incluye el tiempo empleado para el diagnóstico y la reparación, que se detalla en la siguiente fórmula:

$$T_{TC} = T_D + T_R$$

Dónde:

⁶ El componente deja de satisfacer completamente la función requerida como resultado de manifestarse uno u otro modo de falla. Manual de mantenimiento, Alejandro Pistarelli.

⁷ Condición u estado verificable en el componente que indica que una falla funcional está en proceso de ocurrencia. Manual de mantenimiento, Alejandro Pistarelli.

T_{TC} = Tiempo total correctivo.
 T_D = Tiempo de diagnóstico.
 T_R = Tiempo de reparación.

Asimismo, hay dos tipos de mantenimiento correctivo: inmediato y diferido.

El **mantenimiento inmediato** implica la ejecución de tareas no programadas generadas por un fallo imprevisto del *ítem* que requiere de una intervención rápida para restablecer la función del equipo.

El **mantenimiento diferido** implica una ejecución de tareas planificada.

En la figura 15, se determina que, después del tiempo de inflexión de “no planificación”, debe realizarse obligatoriamente una planificación del mantenimiento correctivo, pues esta genera un beneficio económico. Sin embargo, para tiempos menores no es obligatoria una planificación inicial.

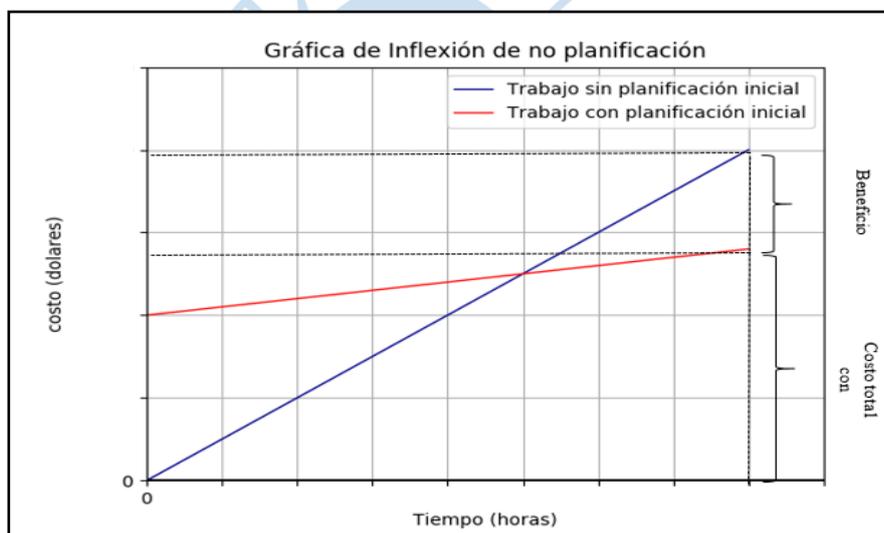


Figura 15. Costo total correctivo por trabajos con y sin planificación
Fuente: Manual de mantenimiento Ingeniería, Gestión y planificación, Alejandro Pistarelli.

3.1.3. Mantenimiento proactivo

Permite identificar y corregir las causas que originan las fallas del *ítem*. Esta técnica genera beneficios; ya que, encontrar soluciones a fallas recurrentes o crónicas permite un incremento del tiempo medio transcurrido entre fallas (MTBF).

3.2. Análisis de datos de fallas de la flota de máquinas PC4000-6

La flota de máquinas PC4000-6 representa los activos más importantes en la etapa de carguío y acarreo, pues una máquina alimenta alrededor de 6 volquetes; por lo que, si falla, se dejaría de extraer aproximadamente 2280 t/h de roca fosfórica.

La unidad de medida empleada para el análisis es el “**tiempo de parada⁸ (horas)**”; causado por las fallas presentadas durante un año en la flota de máquinas PC4000-6 que se encuentra funcionando en una mina del Perú.

3.2.1. Diagrama de Pareto (Análisis ABC)

“El nombre de Pareto se denomina así en honor a Vilfredo Pareto (1848-1923) quien fue un economista italiano, que descubrió a través de un estudio sobre la distribución de la riqueza que la minoría de la población poseía la mayor parte de la riqueza, a su vez la mayoría de la población poseía la menor parte de la riqueza”. (Sales, 2002)

El Dr. Joseph Juran aplicó este concepto a la calidad y obtuvo lo que se conoce como la ley 80-20, debido a que el 80% de los problemas es provocado por el 20% de la población. Esta herramienta de calidad aplicada a la flota de máquinas PC4000-6 permite determinar cuáles son los sistemas más críticos en los que se deben aplicar las estrategias y las acciones de mejoramiento.

Para su construcción se deben seguir los siguientes pasos:

- 1) Encontrar las horas en las que ha fallado cada sistema (hidráulico, eléctrico, mecánico, motriz, lubricación y refrigeración) de la flota de máquinas PC4000-6.
- 2) Sumar el total de las horas que han fallado los sistemas.
- 3) Colocar la escala de frecuencia desde “0” hasta el “total” en el eje vertical izquierdo.
- 4) Dibujar en el eje horizontal el diagrama de barras.
- 5) Colocar la escala del porcentaje acumulativo desde “0%” hasta “100%” en el eje vertical derecho.
- 6) Calcular qué porcentaje del total representa cada sistema.
- 7) Calcular el porcentaje acumulativo de cada sistema.
- 8) Dibujar la curva de Pareto.
- 9) Colocar títulos a los ejes.

⁸ Intervalo de tiempo durante el cual un ítem está en estado de parada (IEC 60050-192;2015)

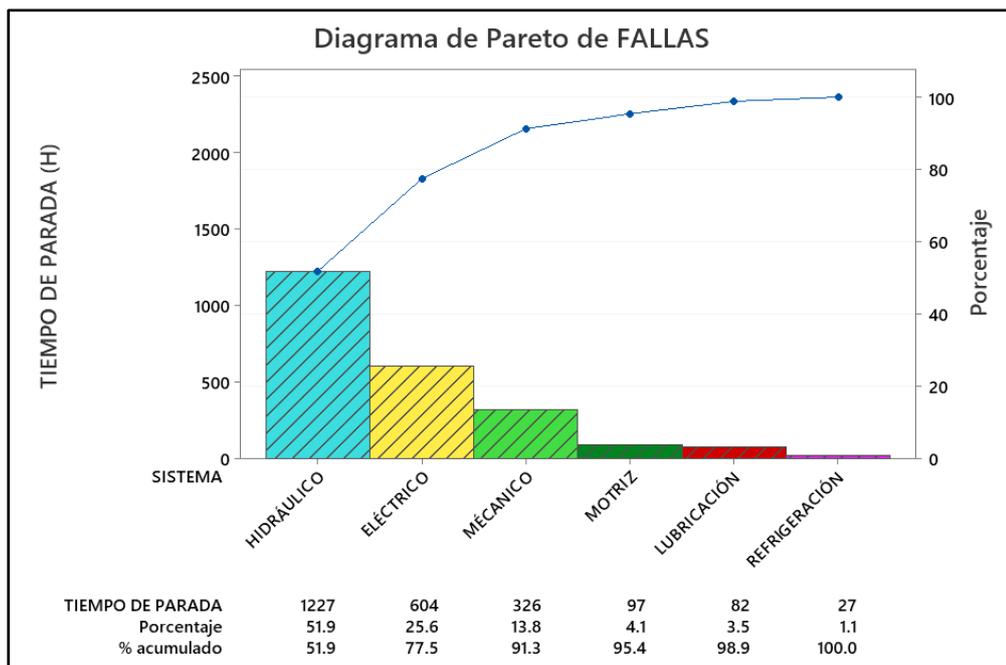


Figura 16. Diagrama de Pareto con respecto al tiempo (horas) de parada por falla
 Fuente: Elaboración Propia

En la gráfica de Pareto, generada a través del software estadístico “Minitab” (ver figura 16), se aprecia que el **80%** de los problemas pertenecen a los sistemas **hidráulico** y **eléctrico**. Por tanto, se necesita prestar atención a estos sistemas para mejorar la disponibilidad de la flota de máquinas PC4000-6.

3.3. Introducción a la Inteligencia artificial. *Machine learning* y *Deep learning*

Los términos *machine learning* y *deep learning* están relacionados entre sí y se usan con mucha frecuencia en la actualidad (ver figura 17).

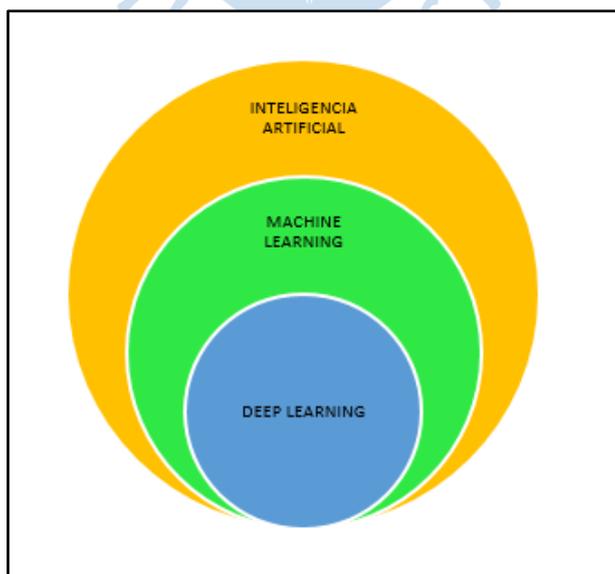


Figura 17. Inteligencia artificial en la actualidad.
 Fuente: Elaboración Propia

Inteligencia Artificial

Existen diversas definiciones de inteligencia artificial (IA), pero, en esencia, se puede definir como la capacidad que tiene una máquina para interactuar con el mundo que le rodea y lograr objetivos complejos. En tal sentido, la máquina intenta adquirir capacidades para ver, escuchar, comprender y aprender de la manera en que lo hacen los humanos.

La IA abarca un campo muy amplio de herramientas que en conjunto van delineando su definición. En los últimos años, la IA está en auge debido al incremento de la capacidad de procesamiento de las computadoras; por lo que se aprecia máquinas que:

- Interactúan con los humanos a través del lenguaje natural (ver figura 18),
- Aprenden de sus resultados,
- Se adaptan a nuevos ambientes y
- Resuelven problemas de manera rápida y eficiente.

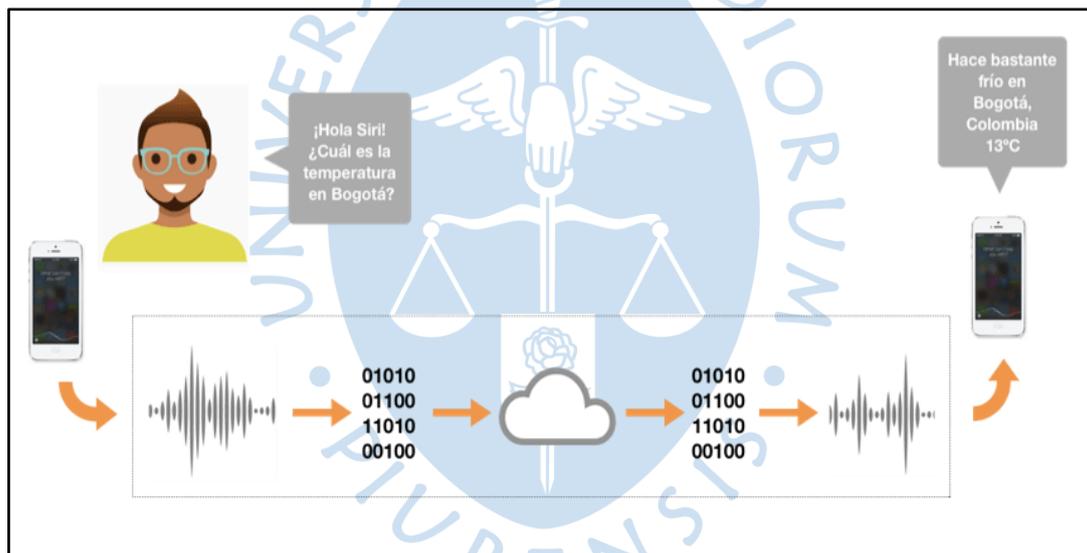


Figura 18. Aplicación de la IA
Fuente: <https://ligdigonzalez.com/>
Visita: 08/12/2019

Machine Learning

El *machine learning* (aprendizaje automático) es una rama de la IA y se define como la capacidad que tienen las máquinas para aprender sin ser explícitamente programadas.

También se puede definir como la práctica de usar algoritmos para parsear datos (analizar una secuencia de símbolos a fin de determinar su estructura gramatical), aprender de ellos (por ejemplo: hablar, reconocer imágenes, entre otras), y luego ser capaces de hacer una predicción o sugerencia sobre algo que permita mejorar tareas con la experiencia.

Los algoritmos que se usan en el *machine learning* llegan a conseguir que las máquinas no solo sean inteligentes, sino que también tengan un cierto grado de autonomía. Se basa fundamentalmente en la estadística para reconocer patrones entre un conjunto de datos y, de esa manera, hacer predicciones. Por ejemplo, como se muestra en la figura 19, el algoritmo de *machine learning* reconoce patrones en un conjunto de correos previamente clasificados en categorías (*spam* y no *spam*) y, con base en ello, cuando se le suministra un nuevo lote de correos sin clasificar, el algoritmo es capaz de separarlos automáticamente por categorías, en este caso, en *spam* y no *spam*.

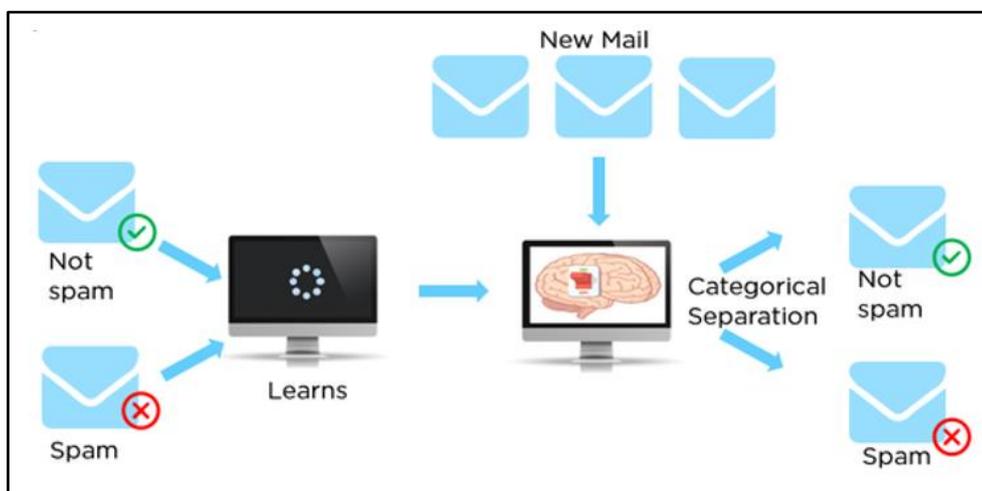


Figura 19. Aplicaciones del *machine learning*.
 Fuente: <https://mc.ai/what-is-machine-learning-14/>
 Visita: 08/12/2019

Deep Learning

Llamado también “aprendizaje profundo”, es un subconjunto de algoritmos de *machine learning* capaz de imitar la conectividad del ser humano para clasificar conjuntos de datos y encontrar correlaciones entre ellos.

Hace uso de redes neuronales multicapa (ver figura 20) con la finalidad de conseguir que la máquina sea capaz de aplicar su conocimiento a otro conjunto de datos para realizar predicciones; mientras más datos se tenga, mejores resultados se obtendrán.

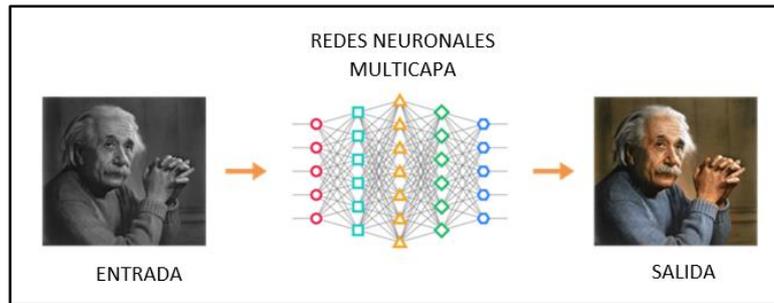


Figura 20. Aplicaciones de deep learning.
Fuente: <https://ligdigonzalez.com/>
Visita: 08/12/2019

3.4. Mantenimiento inteligente

Es el conjunto de tecnologías de la industria 4.0 que se integran para mejorar la disponibilidad de la planta. Estas tecnologías son internet de las cosas, *big data*, IA, 5G, ciberseguridad, *cloud computing*, entre otras. (ver figura 21).

- **Industria 4.0:** Proceso de digitalización de la empresa.
- **Internet de las cosas:** Posibilita conectar diversos objetos mediante internet.
- **Big data:** Conjunto de datos a gran escala.
- **Ciberseguridad:** Herramienta que permite defender a los aparatos electrónicos de ataques maliciosos.
- **Cloud computing:** Prestación de servicios informáticos mediante internet.

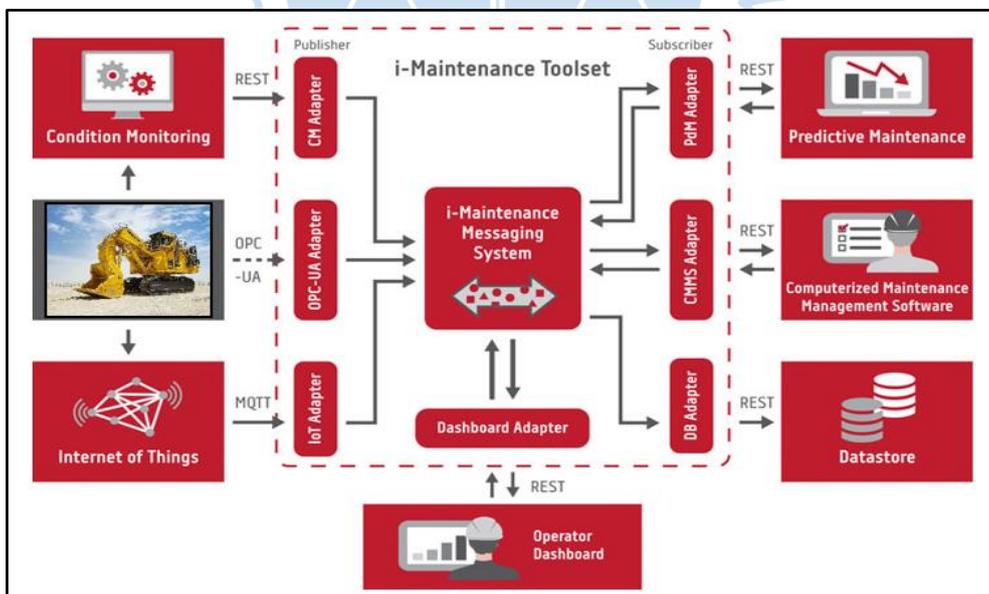


Figura 21. Descripción arquitectónica del conjunto de herramientas i-maintenance.
Fuente : <https://ercim-news.ercim.eu/en115/special/2095-i-maintenance-a-digital-twin-for-smart-maintenance>
Visita: 08/12/2019

En la actualidad, existen dispositivos electrónicos que poseen gran capacidad de almacenamiento y alta potencia de procesamiento, lo que permite utilizar algoritmos predictivos de *machine learning* para analizar los datos provenientes de sensores estratégicamente ubicados en las máquinas. La implementación del mantenimiento inteligente permite que el personal pueda realizar las acciones necesarias en el momento adecuado, con el fin de preservar la función de la máquina. Esto es importante debido a que se mejora la disponibilidad de la máquina y se utilizan de manera eficiente los recursos destinados al mantenimiento de esta.

Parte del mantenimiento inteligente son los sistemas informáticos inteligentes (ver figura 22) que permitan gestionar apropiadamente los datos de mantenimiento y dar respuestas adecuadas, con la finalidad de que el personal de mantenimiento tome decisiones eficaces. En tal sentido es importante que para la construcción de sistemas inteligentes de clasificación se realice un estudio de la planta y se utilicen adecuadamente los algoritmos predictivos de *machine learning*.



Figura 22. Desarrollo de sistemas inteligentes
Fuente: Dataquest
Visita: 08/12/2019

3.5. Ventajas del mantenimiento inteligente

El mantenimiento inteligente ofrece diversas ventajas entre las que destacan: menor costo de mantenimiento, mayor eficiencia en la producción, mayor seguridad en las operaciones y menor riesgo del medio ambiente.

- **Menor Costo de Mantenimiento**

Se procesa la información proveniente de los sensores de la máquina y, por medio del uso de algoritmos predictivos, se realizan acciones tempranas que permitan reducir significativamente las actividades correctivas que ocasionan mayores costos de mantenimiento.

- **Mayor eficiencia en la producción**

Permite que el personal conozca el estado de la máquina en tiempo real para corregir las desviaciones y así evitar paradas no programadas que afecten el plan de producción.

- **Mayor seguridad**

Permite realizar mediciones de manera remota, logrando disminuir la interacción hombre-máquina y, a su vez, los riesgos de accidentes. Asimismo, permite obtener grandes volúmenes de información que pueden ser analizados por el fabricante para desarrollar máquinas más seguras (ver figura 23).



Figura 23. Operaciones remotas

Fuente: <https://www.youtube.com/watch?v=19nmFSQpHEk>

Visita: 05/12/2019

- **Menor riesgo al medio ambiente**

Los diversos sensores que están instalados en la máquina realizan un monitoreo continuo, lo que permite detectar en tiempo real diversas averías o fallas que puedan afectar al medio ambiente; y así, de ser el caso, solucionar el problema en el menor tiempo posible.

Capítulo 4

Sistema inteligente para recolectar y clasificar datos de fallas

En este capítulo se desarrolla el sistema inteligente para recolectar y clasificar datos de fallas de la flota de máquina PC4000-6 en sus respectivos sistemas, con la finalidad de mejorar la gestión de datos de mantenimiento.

4.1. Desarrollo del sistema inteligente

La empresa minera que apoyó este trabajo dispone de un procedimiento para el registro de fallas que consiste en:

- a) A través de la radio, el operador de la máquina transmite la falla al controlador de flota, quien se encuentra en la sala *Dispatch*.
- b) El controlador de flota registra la falla y le da una etiqueta⁹ de acuerdo con su criterio, que en la mayoría de los casos no es la correcta.
- c) El planner recibe diariamente el registro de fallas y tiene que corregir manualmente dichas etiquetas para poder realizar un adecuado análisis de los datos de fallas.

Esta forma de realizar las tareas antedichas implica para el planner un gran consumo de tiempo (2 de las 8 horas de su jornada laboral), que podría aprovechar para cumplir sus verdaderas funciones, como mejorar la productividad de la mano de obra y reducir al máximo el Tiempo Medio para Reparar (MTTF).

Ante esta situación se propone una herramienta que permite reducir la intervención humana mediante la automatización del proceso de clasificación de las fallas; sobre todo, cuando se trata de un gran volumen de fallas. Para ello, se utiliza un algoritmo que garantiza una acción eficaz del control de estas. (ver figura 24).

⁹ Las etiquetas están conformadas por los Sistemas: Hidráulico, eléctrico, motriz, lubricación y refrigeración

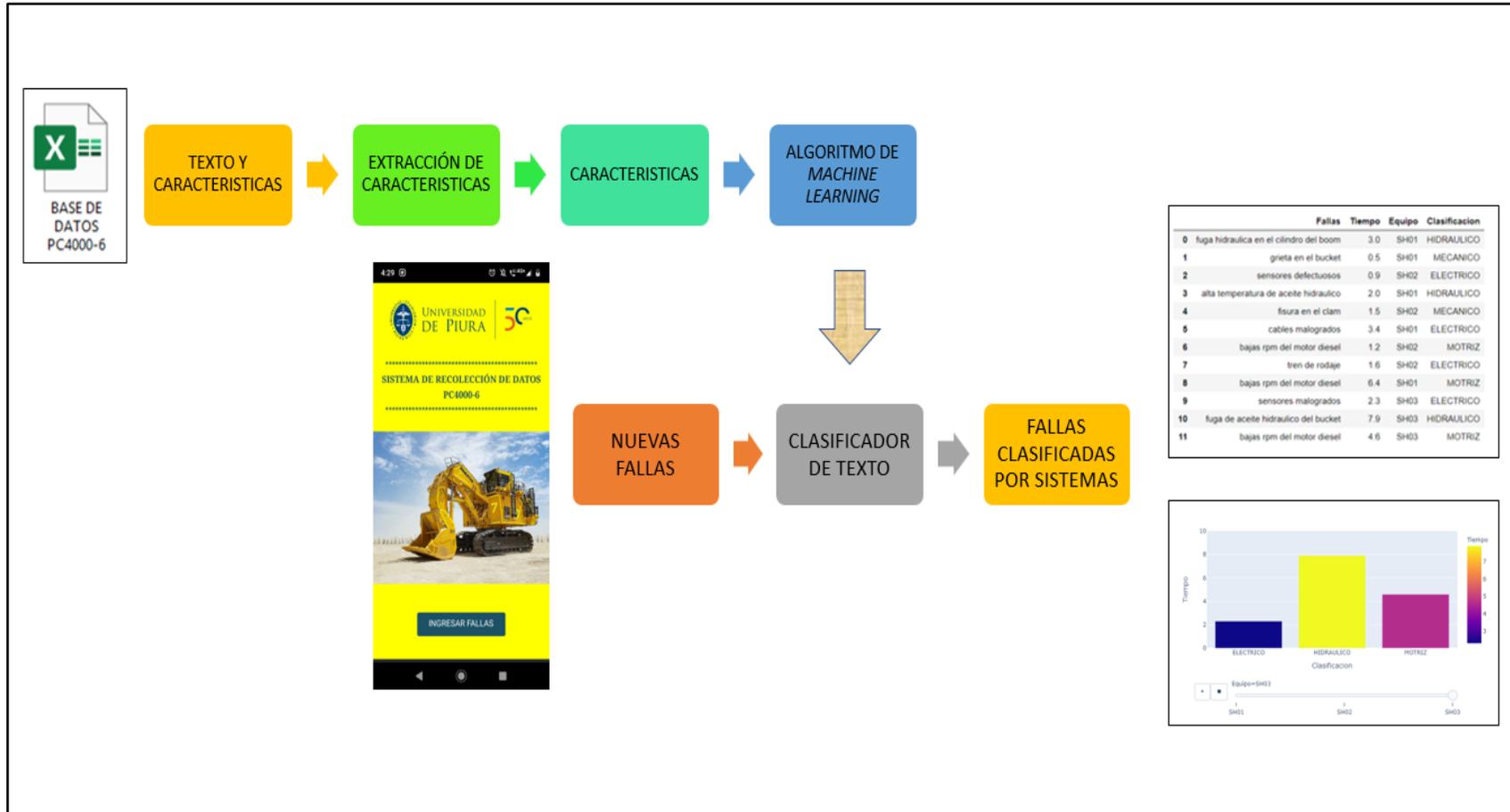


Figura 24. Esquema general del Sistema automático de Clasificación de datos de fallas
Fuente: elaboración propia

El desarrollo de la herramienta presenta dos partes:

- **Primera parte:** crear un aplicativo móvil que permita registrar los datos de fallas y almacenarlas en la nube.
- **Segunda parte:** desarrollar un sistema de clasificación que utilice un algoritmo de *machine learning* para clasificar los datos de fallas en los respectivos sistemas: mecánico, eléctrico, hidráulico, motriz, lubricación y refrigeración.

4.1.1. Desarrollo del aplicativo móvil

En la actualidad los *smartphones* se han convertido en una herramienta importante que ha permitido redefinir la forma en que el ser humano se comunica, se informa, estudia, entre otras acciones.

En el campo de la industria estos teléfonos inteligentes han permitido el desarrollo de aplicaciones que permiten controlar actividades de manera remota, como velocidad de los motores, temperatura de hornos industriales, presión de los tanques de almacenamiento de gas natural, entre otras.

Asimismo, los *smartphones* permiten el desarrollo de aplicaciones para almacenar información en bases de datos de manera local o en la nube. (ver figura 25).



Figura 25. Smartphones

Fuente: <https://firebase.google.com>

Visita: 08/12/2019

Los atributos de los *smartphones* han sido utilizados en este trabajo para la recopilación de información mediante el desarrollo de un aplicativo móvil (PC4000), el cual permite registrar la información sobre las fallas presentes en la flota de máquinas hidráulicas PC4000-6 y compartir dicha información con la nube en tiempo real.

El aplicativo móvil se ha desarrollado en la plataforma *MIT APP INVENTOR*, que permite hacer un prototipo rápido (ver figura 26).

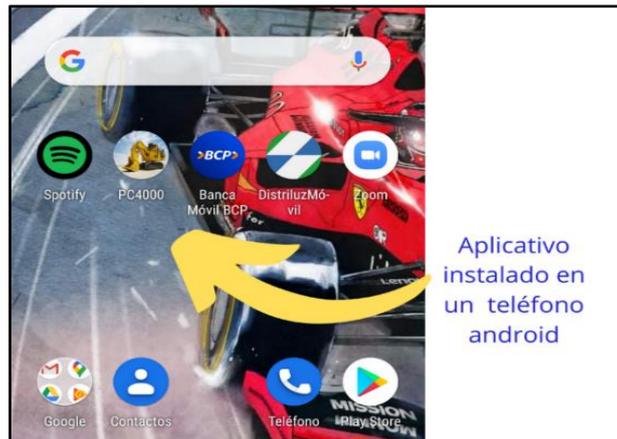


Figura 26. Aplicativo móvil instalado en un teléfono móvil Android.
Fuente: elaboración propia.

El aplicativo móvil permite al personal de mantenimiento registrar los datos de fallas de la flota de máquinas hidráulicas PC4000-6. La información es guardada en una hoja de cálculo que se almacena en Google Drive, de esta manera, es posible disponer del archivo en la nube.

En la figura 27 se muestra la pantalla de inicio del aplicativo móvil en la que se encuentra el botón “INGRESAR FALLA”; que permite abrir la pantalla de registro para ingresar los datos, el tiempo de duración y el equipo en el que se presentó la falla. Además, esta misma pantalla hay 2 botones: uno permite subir el registro de la falla a la nube y el otro, limpiar la pantalla en caso se requiera borrar todo lo escrito.



Figura 27. Pantalla de inicio(izquierda) y pantalla de registro(derecha) del aplicativo móvil.
Fuente: elaboración propia.

En la figura 28 se muestra un campo de texto en el cual se registran los datos de la falla y otro de tipo numérico que permite ingresar la duración de la falla (en horas).



Figura 28. Pantalla de registro del aplicativo móvil.
Fuente: Elaboración propia.

En la figura 29 se observa que, para el ingreso del código del equipo que presenta la falla, hay una lista en la que se encuentra el código de todos los equipos que conforman la flota de máquinas PC4000-6.

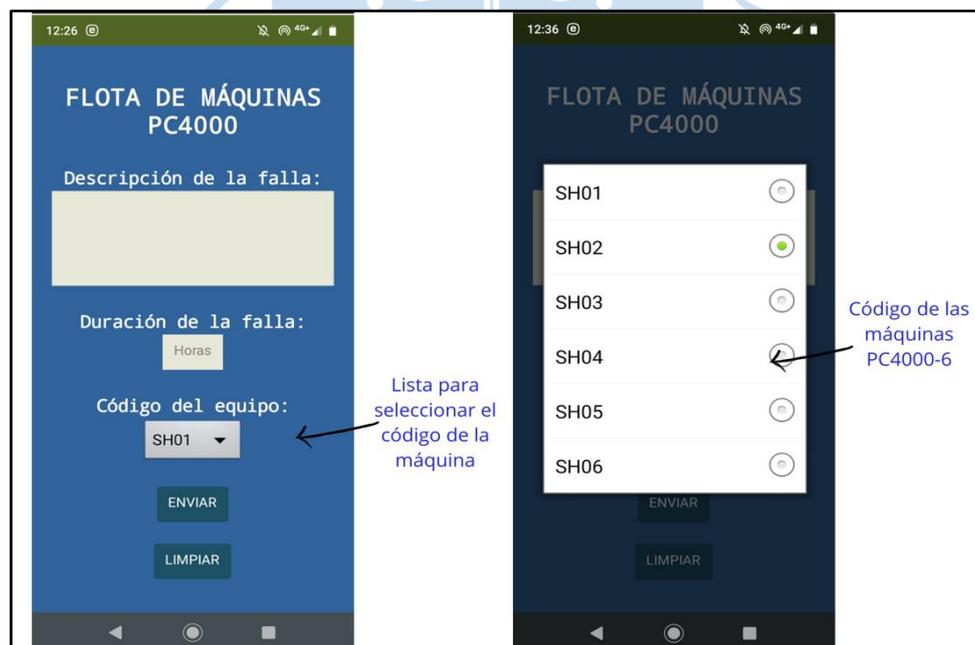


Figura 29. Selección del código de la máquina PC4000-6.
Fuente: elaboración propia.

En la figura 30 se aprecia que una vez enviada la descripción de la falla, este se almacena en una hoja de cálculo en la nube. La descripción de la falla puede ser descargada por el planner de mantenimiento para analizarla usando el sistema de clasificación de datos de fallas.

The image shows a mobile application interface on the left and an Excel spreadsheet on the right. The app is titled 'FLOTAS DE MÁQUINAS PC4000' and has a form for reporting a fault. The form includes a text field for 'Descripción de la falla:' containing 'alta temperatura de aceite hidráulico', a numeric field for 'Duración de la falla:' with the value '1.2', and a dropdown menu for 'Código del equipo:' set to 'SH04'. There are 'ENVIAR' and 'LIMPIAR' buttons at the bottom. The spreadsheet, titled 'FALLAS', has columns for 'FECHA', 'DESCRIPCION DE LA FALLA', 'TIEMPO (H)', and 'EQUIPO'. It contains 18 rows of data, with a yellow arrow pointing from the 'alta temperatura de aceite hidráulico' entry in the app to row 18 in the spreadsheet.

	A	B	C	D
	FECHA	DESCRIPCION DE LA FALLA	TIEMPO (H)	EQUIPO
4	1/12/2019 4:50:47	bajas rpm del motor	5	SH02
5	1/12/2019 4:51:14	bajas rpm del motor	5	SH02
6	1/12/2019 4:52:45	cables rotos	1.5	SH06
7	1/12/2019 4:53:27	fisuras en el boom	3	SH02
8	1/12/2019 4:53:49	fuga hidráulica en el clam	8	SH01
	1/12/2019 5:53:50	fuga en el cilindro hidráulico del Clam	5	SH04
10	1/12/2019 5:54:06	sensores defectuosos	8	SH01
11	1/12/2019 6:17:18	bajas rpm del motor	6	SH01
12	1/12/2019 6:18:31	fuga hidraulica	6	SH01
13	1/12/2019 6:18:42	fisura en la abrazadera de la manguera	4	SH01
14	2/12/2019 14:03:10	rotura de tubería	2.4	SH01
15	2/12/2019 14:03:42	Mangueras defectuosas	5	SH01
16	2/12/2019 14:04:53	fisuras en el clam	4	SH03
17	2/12/2019 17:29:16	fuga hidráulica en el clam	5	SH04
18	2/12/2019 22:52:09	alta temperatura de aceite hidráulico	1.2	SH04

Figura 30. Almacenamiento de datos de fallas en la nube.

Fuente: elaboración propia.

En el **apéndice A** se muestra un manual de usuario, a modo de resumen, con la finalidad de que el usuario se familiarice rápidamente con el funcionamiento del aplicativo.

4.1.2. Desarrollo del sistema de clasificación de datos de fallas

Los pasos para desarrollar el sistema de clasificación de datos de fallas son recopilar datos de fallas, realizar preprocesamiento de datos de fallas, analizar qué algoritmo de *machine learning* se utilizará, implementar modelo, realizar pruebas con nuevos datos de fallas y guardar el modelo entrenado.

✓ **Recopilar datos**

El primer paso es contar con una base de datos para poder entrenar el modelo. Se debe tener en cuenta que mientras más datos se tenga, mejor será el desempeño del modelo. Para este caso se utilizará la data histórica de una flota de máquinas PC4000-6 que se encuentran operando en una mina en el territorio nacional; esta data se encuentra registrada en formato XLS (ver figura 31).



BASE DE DATOS PC4000-6

Ké	Pal	Hora	Tiempo de paradi	Falla	Sistema	Sub-Sistema	Causa Real Falla
1	SH03	17:24:00	01:15:22	FUGA DE ACEITE HIDRAULICO	HIDRÁULICO	Sistema de giro	Fuga por manguera de pilotaje de motor de giro
2	SH05	08:25:19	01:05:00	BLOQUEO DE EQUIPO DE TRABAJO	ELÉCTRICO	Cabina	Joystick LH defectuoso, se encontró 9V en su pin 2 de pruel cuando debería ser 0V.
3	SH05	16:49:07	01:14:35	BLOQUEO DE EQUIPO DE TRABAJO	ELÉCTRICO	Cabina	Joystick LH defectuoso, se encontró 9V en su pin 2 de pruel cuando debería ser 0V.
4	SH05	19:41:00	06:33:23	BLOQUEO DE EQUIPO DE TRABAJO	ELÉCTRICO	Cabina	Joystick LH defectuoso, se encontró 9V en su pin 2 de pruel cuando debería ser 0V.
5	SH06	14:28:00	02:11:32	EQUIPO DE TRABAJO LENTO	ELÉCTRICO	Gabinete X2	Funcionamiento defectuoso de amplificadores
6	SH03	06:35:00	00:14:12	FUGA DE ACEITE HIDRAULICO	HIDRÁULICO	Bloques de control principales	Fuga por desajuste de manguera de pilotaje de bloque pr
7	SH03	08:33:41	26:32:58	FUGA DE REFRIGERANTE	MOTOR DIESEL	Motor de ventilador de radiador	Falla de rodamientos y seguro de eje por falta de lubricación ventilador de radiador
8	SH04	23:02:25	00:22:00	FAROS INOPERATIVOS	ELÉCTRICO	Faros (Luces de Trabajo)	Falla de faro de trabajo superior izquierdo
9	SH05	03:19:29	00:37:52	BLOQUEO DE EQUIPO DE TRABAJO	ELÉCTRICO	Gabinete X2	Cable suelto de voltaje de alimentación de módulo de capacit
10	SH06	04:20:48	00:21:10	BLOQUEO DE ARRANQUE	ELÉCTRICO	Baterías	Bajo voltaje de baterías de Gabinete X2
11	SH02	17:48:56	01:03:50	FUGA DE ACEITE HIDRAULICO	HIDRÁULICO	Equipo de trabajo	Fuga de aceite por O-ring de manguera de cilindro de cla vástago
12	SH05	08:11:23	11:42:02	ALTA TEMPERATURA DE REFRIGERANTE	MECÁNICO	Bombas	Falla del eje estriado de bomba secundaria de accionamiento de radiador debido a mala lubricación
13	SH02	06:19:12	03:12:21	FUGA DE ACEITE HIDRAULICO	HIDRÁULICO		
14	SH03	14:09:28	00:12:49	ALTA TEMPERATURA DE ACEITE HIDRÁULICO	ELÉCTRICO	Sensores/ Conectores /Varios	Funcionamiento defectuoso de sensor de temperatura de acc 40B004
15	SH03	20:07:48	00:59:16	EQUIPO DE TRABAJO LENTO	ELÉCTRICO	Gabinete X2	Funcionamiento defectuoso de fusible que alimenta controla de bombas accionadas

Figura 31. Extracto de la base de datos PC4000-6.
Fuente: elaboración propia

✓ **Realizar preprocesamiento de datos de fallas**

Los datos de fallas de la flota de máquinas PC4000-6 se reclasificaron en sus respectivos sistemas (ver figura 32), teniendo en cuenta la norma ISO 14224:2016.



Figura 32. Sistemas de la máquina PC4000-6
Fuente: Elaboración propia.

El set de datos de fallas se tiene que dividir en dos partes: datos de entrenamiento y datos de prueba (ver figura 33). Se debe considerar que los datos de prueba no se pueden utilizar cuando se está realizando el entrenamiento del modelo, debido a que puede ocasionar el sobreajuste de este.

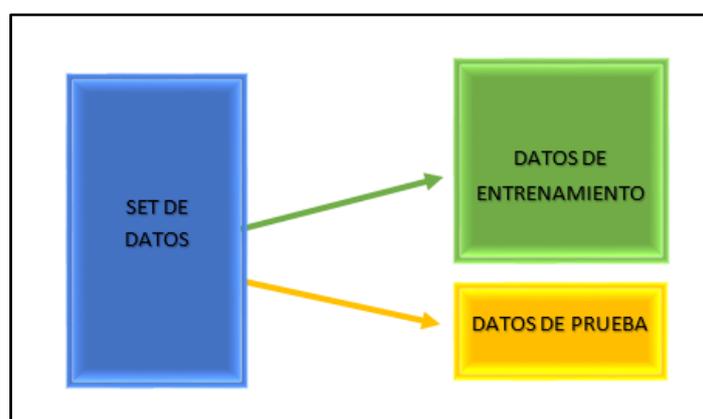


Figura 33. Set de datos (Entrenamiento y prueba)
Fuente: elaboración propia.

Tokenización

Con los datos de entrenamiento se lleva a cabo el “proceso de tokenización”, el cual consiste en dividir el texto en un conjunto de palabras, por ejemplo, en la figura 34 se puede apreciar: “Fisura en la cabina del operador”, como un dato de entrada en forma de *string* (cadena de texto). Al realizar el “proceso de tokenización”, este dato de entrada queda dividido en 6 palabras o tokens por separado: **[token 1: fisura, token 2: en, token 3: la, token 4: cabina, token 5: del, token 6: operador]**.

Luego se realiza una limpieza de los datos para eliminar aquellas palabras que no dan valor agregado a la clasificación del algoritmo, como artículos, signos de puntuación, entre otros. Respecto al ejemplo anterior, se realiza el proceso de limpieza en los datos, por lo que la cantidad de tokens o palabras que en un inicio eran 6 se reducen a solo 3: **[Token 1: fisura, Token 2: cabina, Token 3: operador]**.

<p>TEXTO: Fisura en la cabina del operador</p> <p>TEXTO TOKENIZADO: ['Fisura', 'en', 'la', 'cabina', 'del', 'operador']</p> <p>LIMPIEZA DEL TEXTO: [Fisura, cabina, operador]</p>

Figura 34. Limpieza de datos
Fuente: Elaboración propia.

Vectorización

Consiste en convertir una palabra o conjunto de palabras en un vector numérico. Este proceso es necesario para algoritmos de lenguaje natural debido a que es el idioma que maneja el software de una computadora.

A continuación, se dará un ejemplo sencillo de vectorización. Se ha supuesto que se realizó la tokenización y limpieza de todas las fallas presentes en el equipo, en consecuencia, se obtuvieron las siguientes palabras o tokens: **[token 1: fisura, token 2: sensor, token 3: defectuoso, token 4: cabina, token 5 : operador, token 6: hidráulica, token 7: fuga, token 8: boom, token 9: bucket, token 10: cilindro]**, Al ingresar una nueva falla: “fuga en el cilindro del boom”, se obtiene un vector de la siguiente forma:

$$[0,0,0,0,0,0,1,1,0,1],$$

Como puede observarse se obtiene un vector de 10 columnas que representa el número de tokens presentes; a su vez se observa que en la primera columna del anterior vector aparece el número “0”, lo cual indica que la palabra “**fisura**” no está presente en la nueva falla ingresada, y así sucesivamente para las siguientes columnas. En el caso de la columna 7, 8 y

10 aparece el número “1”, lo que indica que las palabras “Fuga”, “Boom” y “Cilindro” están presentes una vez en la falla “fuga en el cilindro del boom” (ver figura 35).

DICcionario DE PALABRAS	
TOKEN 1:	FISURA
TOKEN 2:	SENSOR
TOKEN 3:	DEFECTUOSO
TOKEN 4:	CABINA
TOKEN 5:	OPERADOR
TOKEN 6:	HIDRAULICA
TOKEN 7:	FUGA
TOKEN 8:	BOOM
TOKEN 9:	BUCKET
TOKEN 10:	CILINDRO

FUGA EN EL CILINDRO DEL BOOM
[0, 0, 0, 0, 0, 0, 1, 1, 0, 1]

Figura 35. Ejemplo sencillo de vectorización
Fuente: Elaboración propia.

✓ **Analizar qué algoritmo de *machine learning* se utilizará**

“Un algoritmo es un conjunto ordenado y finito de operaciones que permite hallar la solución de un problema” (Real Academia Española, 2019)

Es importante saber elegir el algoritmo de *machine learning* adecuado, es decir, el que mejor se adapte a nuestros datos para poder realizar adecuadamente las predicciones.

En el desarrollo del sistema de clasificación se realizaron pruebas utilizando diferentes algoritmos de *machine learning*, los cuales se detallan a continuación:

Logistic Regression

La regresión logística (*Logistic Regression*, en inglés) es un tipo de análisis utilizado para predecir el resultado de una variable. Se sistematiza por medio de un algoritmo usado para clasificación binaria y multivariada, basada en la función estocástica o función “Logística”, la cual proporciona la probabilidad P_i de pertenecer a la clase i .

$$P_i = \frac{e^{z_i}}{\sum_{j=1}^k e^{z_j}}$$

$$z_i = \alpha_{i,1} * x_1 + \alpha_{i,2} * x_2 + \alpha_{i,3} * x_3 + \dots + \alpha_{i,n} * x_n + \beta_i$$

Donde:

- K = número de clases
- $x_1, x_2, x_3, \dots, x_n$ = Valores de entrada del algoritmo

- $\alpha_{i,1}$, $\alpha_{i,2}$, $\alpha_{i,2}$ y β_i = Parámetros de las clases
- P_i = Probabilidad de pertenecer a la clase i

Support Vector Machine (SVM)

Las máquinas de vector soporte (*Support Vector Machine*, en inglés) son un conjunto de algoritmos de aprendizaje supervisado de *machine learning*, empleados mayormente en problemas de clasificación o regresión. Se basan en la idea de minimización de riesgo estructural (SRM).

Un algoritmo basado en SVM ubica los datos de entrada en un espacio de coordenadas como se muestra en la figura 37, y encuentra el mejor hiperplano de separación entre las 2 clases; para lograr esto, la línea o hiperplano trazado debe ser tal que maximice el margen “ m ” con los puntos que estén más cerca de este, por eso se les conoce también como clasificadores de margen máximo.

En la figura 36 los puntos “azules” y los puntos “rojos” representan las fallas eléctricas e hidráulicas, respectivamente. Lo que hará este algoritmo será encontrar el mejor hiperplano que separe las dos clases. En caso se ingrese un nuevo dato (falla) y este se ubique a la izquierda de la línea o hiperplano, tendría como salida “falla eléctrica”, caso contrario el sistema daría como salida “falla hidráulica”.

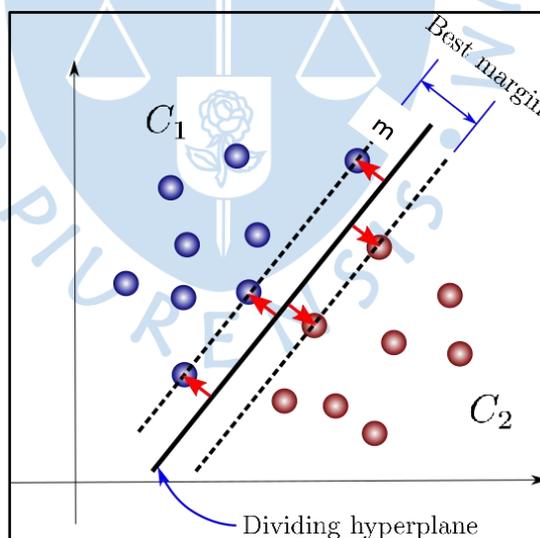


Figura 36. Representación del conjunto de datos y margen
 Fuente: <https://towardsdatascience.com/support-vector-machines-for-classification-fc7c1565e3>
 Visita: 10/12/2019

Decision Tree

El *machine learning* basado en el árbol de decisión es una metodología que permite el aprendizaje (toma de decisiones), creando un modelo que predice el valor de una variable en

función de diversas variables de entrada. La ventaja que ofrece esta metodología es su fácil visualización e interpretación como puede apreciarse en la figura 37. En este caso, parte de una condición inicial y van saliendo de ella ramificaciones hasta que se llega a la respuesta final.

Se ha supuesto para fines didácticos que en la figura 38, X_1 y X_2 son 2 características (entradas) y que t_1 , t_2 , t_3 y t_4 tomen los siguientes valores numéricos: 2, 5, 10 y 20, respectivamente. El algoritmo comparará la condición 1 ($X_1 < 2$) si se cumple, se analizará la siguiente ($X_2 < 10$). Si satisface la nueva condición, el algoritmo clasificará la falla como sistema “ELECTRICO”; caso contrario será clasificada como sistema “MECANICO”. En caso la condición 1 no sea correcta, se evaluará la siguiente ($X_2 < 5$); si es así el sistema analizará otra condición ($X_1 < 20$); en el caso de que esta se cumpla, la respuesta será sistema “HIDRAULICO”; de lo contrario será sistema “MOTRIZ”. Si la condición $X_2 < 5$ no se cumple, el algoritmo dará como respuesta sistema “LUBRICACION”.

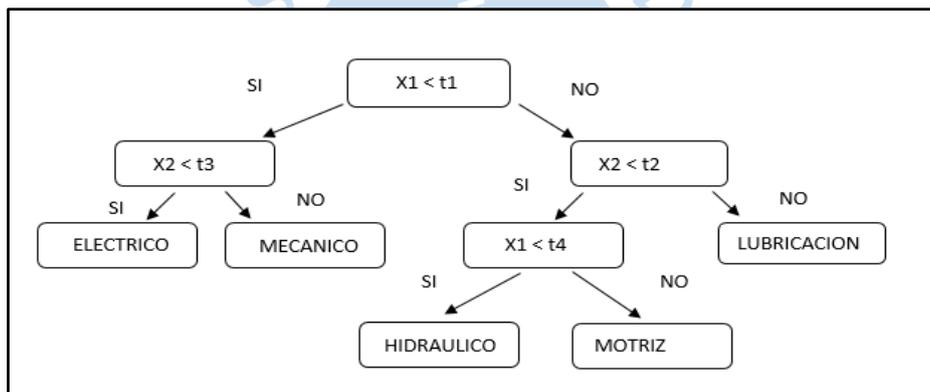


Figura 37. *Decision Tree*
Fuente: Elaboración propia
Visita: 15/12/2019

Naive Bayes

Los modelos de Naive Bayes son un conjunto de algoritmos de clasificación (clasificadores probabilísticos). Están basados en el teorema de Naive Bayes, el cuál calcula la probabilidad de que ocurra un suceso “c” habiendo ocurrido ya un suceso “x”, tal y como se muestra en la siguiente expresión.

$$P(C/X) = \frac{(P(X/C) * P(C))}{P(X)}$$

Donde:

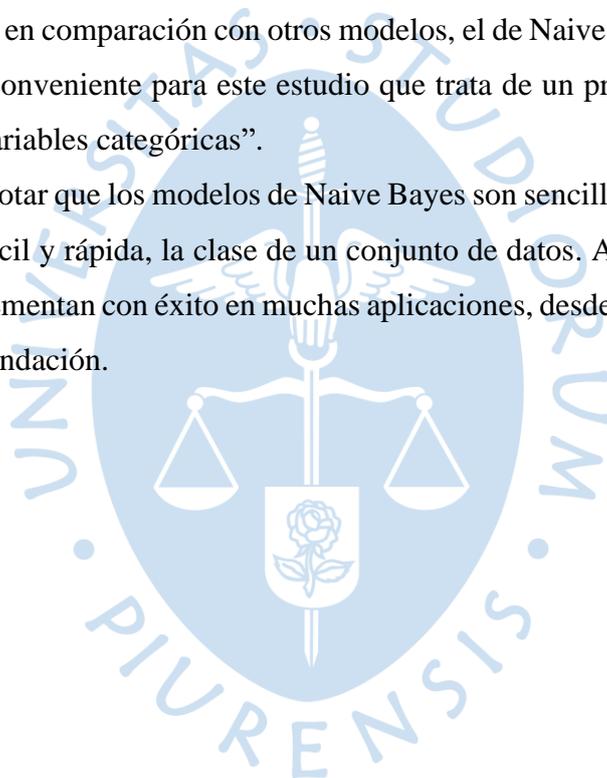
- **P(c):** Probabilidad de que la hipótesis “c” sea cierta (independientemente de los datos).
- **P(x):** probabilidad de los datos (independientemente de la hipótesis).

- $P(c|x)$: Probabilidad de la hipótesis “c” dados los datos “x”.
- $P(x|c)$: Probabilidad de los datos “x” dado que la hipótesis “c” era cierta.

Las pruebas se realizaron utilizando “*Scikit-learn*”, biblioteca de código abierto especializada en *machine learning* para el lenguaje de programación “*Python*” que incluye los modelos antes vistos. El código empleado, para cada uno de los modelos, se encuentra en el **anexo A** (*Logistic Regression*), **anexo B** (*Naive Bayes*), **anexo C** (*Support Vector Machine*) y **anexo D** (*Decision Tree*).

De los resultados obtenidos (ver figura 38), se evidencian diferentes predicciones (***Logistic regression*, 70%; Naive Bayes, 85%; Support Vector Machine, 75%; y Decision Tree, 60%**). Por tanto, en comparación con otros modelos, el de Naive Bayes presenta un mejor desempeño y es más conveniente para este estudio que trata de un problema de clasificación, cuyas entradas son “variables categóricas”.

Es importante acotar que los modelos de Naive Bayes son sencillos y potentes, y permiten predecir, de manera fácil y rápida, la clase de un conjunto de datos. Además, en la actualidad, estos modelos se implementan con éxito en muchas aplicaciones, desde el análisis de texto hasta los motores de recomendación.



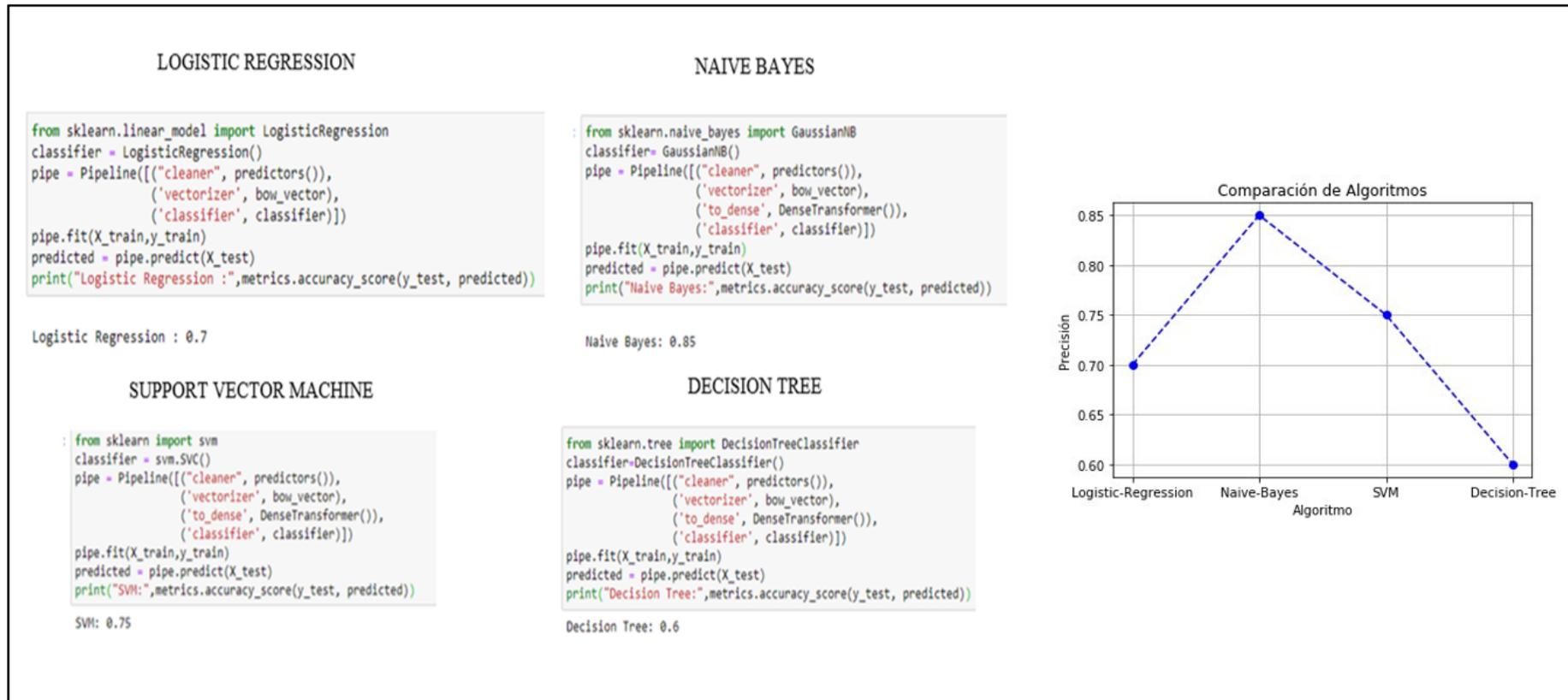


Figura 38. Análisis con diferentes algoritmos de *Machine learning*.
Fuente: Elaboración propia

✓ **Implementar modelo y realizar pruebas con nuevos datos**

Naive Bayes o Bayes ingenuo

Como se mencionó anteriormente, Naive Bayes es el algoritmo de *machine learning* que se utiliza para el desarrollo del sistema de clasificación de fallas.

Por ejemplo, se necesita saber a qué sistema pertenece la falla denominada “**fuga hidráulica en el cilindro del boom izquierdo**”. Una vez que se ingresa, y en base a la data de entrenamiento, el algoritmo empieza a calcular la probabilidad para cada uno de los sistemas en que se ha dividido la máquina. Así:

Para el sistema eléctrico

$P(\text{sistema eléctrico} | \text{fuga hidráulica en el cilindro del boom}) = P(\text{sistema eléctrico}) * P(\text{fuga} | \text{sistema eléctrico}) * P(\text{hidráulica} | \text{sistema eléctrico}) * P(\text{cilindro} | \text{sistema eléctrico}) * P(\text{boom} | \text{sistema eléctrico})$

Para el sistema mecánico

$P(\text{sistema mecánico} | \text{fuga hidráulica en el cilindro del boom}) = P(\text{sistema mecánico}) * P(\text{fuga} | \text{sistema mecánico}) * P(\text{hidráulica} | \text{sistema mecánico}) * P(\text{cilindro} | \text{sistema mecánico}) * P(\text{boom} | \text{sistema mecánico})$

Para el sistema de refrigeración

$P(\text{sistema de refrigeración} | \text{fuga hidráulica en el cilindro del boom}) = P(\text{sistema de refrigeración}) * P(\text{fuga} | \text{sistema de refrigeración}) * P(\text{hidráulica} | \text{sistema de refrigeración}) * P(\text{cilindro} | \text{sistema de refrigeración}) * P(\text{boom} | \text{sistema de refrigeración})$

Para el sistema motriz

$P(\text{sistema motriz} | \text{fuga hidráulica en el cilindro del boom}) = P(\text{sistema motriz}) * P(\text{fuga} | \text{sistema motriz}) * P(\text{hidráulica} | \text{sistema motriz}) * P(\text{cilindro} | \text{sistema motriz}) * P(\text{boom} | \text{sistema motriz})$

Para el sistema hidráulico

$P(\text{sistema hidráulico} | \text{fuga hidráulica en el cilindro del boom}) = P(\text{Sistema hidráulico}) * P(\text{Fuga} | \text{Sistema hidráulico}) * P(\text{hidráulica} | \text{Sistema hidráulico}) * P(\text{cilindro} | \text{Sistema hidráulico}) * P(\text{boom} | \text{Sistema hidráulico})$

Para el sistema de lubricación

$P(\text{sistema lubricación} | \text{fuga hidráulica en el cilindro del boom}) = P(\text{sistema lubricación}) * P(\text{Fuga} | \text{sistema lubricación}) * P(\text{hidráulica} | \text{sistema lubricación}) * P(\text{cilindro} | \text{sistema lubricación}) * P(\text{boom} | \text{sistema lubricación})$

La respuesta viene determinada por el valor máximo de puntuación (ver figura 39); para el caso del ejemplo la respuesta sería:

SISTEMA HIDRÁULICO

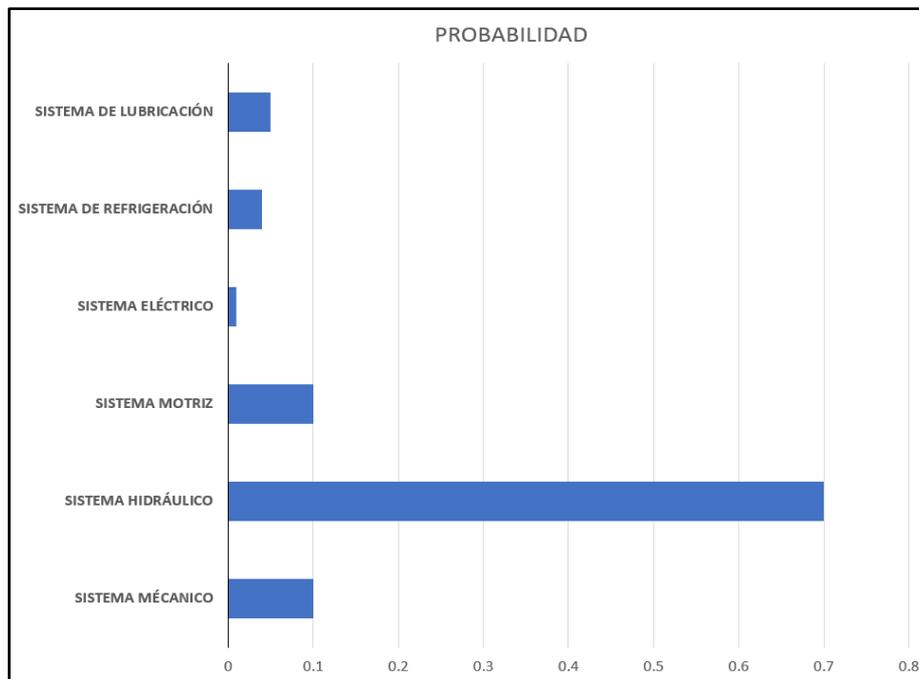


Figura 39. Puntuación obtenida
Fuente: Elaboración propia

Programación en Python

Python es un lenguaje de programación de alto nivel (estructura sintáctica y semántica legible), de multiplataforma (interpretado en Linux, Windows, Mac Os, entre otros) y de tipado dinámico (autoasigna el tipo de variables según el valor declarado). Fue creado por el programador Holandés Guido Van Rossum a principios de 1990, y su nombre se debe al grupo de humoristas británicos *Monty Python*. Inicialmente fue desarrollado para el sistema operativo *Amoeba*.

Presenta múltiples ventajas, como simplicidad de sintaxis, licencia de código abierto y diversas librerías (*Numpy*, *Open CV*, *Matplotlib*, *Tensor Flow*, entre otros). Este lenguaje se emplea en el desarrollo del sistema de clasificación; así mismo, se usa el algoritmo de Naive Bayes a través de *Textblob*, “Biblioteca de *Python* que permite realizar tareas de procesamiento del lenguaje natural (PNL) como por ejemplo análisis de sentimiento, extracción de frases nominales, etc.” (Steven, 2019).

El código del sistema de clasificación de datos de fallas de la flota de máquinas PC400-6, se explica mediante una serie de diez pasos que, a continuación, se detallan en forma ordenada:

El **primer paso** consiste en importar las librerías que se emplearán para el sistema (ver figura 40).

```
#Librerías
import pandas as pd
import numpy as np
import csv
import matplotlib.pyplot as plt
from matplotlib.ticker import PercentFormatter
from textblob.classifiers import NaiveBayesClassifier
import chart_studio
import chart_studio.plotly as py
from plotly.graph_objs import *
import plotly.express as px
```

Figura 40. Código de librerías importadas
Fuente: Elaboración propia.

- **Pandas:** Se emplea para la manipulación y el análisis de datos.
- **Numpy:** Se utiliza para realizar cálculos eficientes con matrices.
- **CSV:** Permite leer y escribir datos tabulares en formato CSV (valores separados por comas).
- **Matplotlib:** Permite visualizar datos de manera gráfica.
- **Chart Studio:** Sirve para crear gráficos interactivos.
- **Textblob Classifiers:** Simplifica la creación de clasificadores personalizados.

El **segundo paso** consiste en cargar los datos de entrenamiento preprocesados, que se han obtenido a partir de la data histórica de la flota de máquinas PC4000-6 con el fin de obtener buenas predicciones del modelo (ver figura 41).

```
#datos de entrenamiento
with open('pareto.csv','r') as fp:
    cl=NaiveBayesClassifier(fp,format="csv")
```

Figura 41. Data de entrenamiento
Fuente: elaboración propia.

El **tercer paso** consiste en cargar la data que se requiere clasificar (ver figura 42). Debido a que no hay encabezado, es necesario crearlo, dándole a cada columna un nombre. La primera es **Fallas**, que se usa para clasificar las fallas por sistemas (hidráulico, eléctrico, motriz, mecánico, lubricación y refrigeración); la segunda se llama **Tiempo**, usada para elaborar el diagrama de Pareto; y la tercera es **Equipo**, que permite conocer en qué equipo se ha presentado

la falla. Así mismo, se convierten los datos en un *DataFrame*¹⁰ para facilitar el análisis (ver figura 43).

```
#fallas a clasificar
infile = 'pareto1.csv'
datos= pd.read_csv(infile,header=None)
encabezado=["Fallas","Tiempo","Equipo"]
datos.columns= encabezado
df= pd.DataFrame(datos)
print("DataFrame a clasificar")
print("*****")
df
```

Figura 42. Data a clasificar
Fuente: elaboración propia.

	Fallas	Tiempo	Equipo
0	fuga hidraulica en el cilindro del boom	3.0	SH01
1	grieta en el bucket	0.5	SH01
2	sensores defectuosos	0.9	SH02
3	alta temperatura de aceite hidraulico	2.0	SH01
4	fisura en el clam	1.5	SH02
5	cables malogrados	3.4	SH01
6	bajas rpm del motor diesel	1.2	SH02
7	tren de rodaje	1.6	SH02
8	bajas rpm del motor diesel	6.4	SH01
9	sensores malogrados	2.3	SH03
10	fuga de aceite hidraulico del bucket	7.9	SH03
11	bajas rpm del motor diesel	4.6	SH03

Figura 43. DataFrame
Fuente: elaboración propia.

El **cuarto paso** consiste en clasificar los datos de fallas en sus respectivos sistemas (hidráulico, eléctrico, motriz, mecánico, lubricación y refrigeración) mediante *Textblob Classifiers*, el cual hace uso del algoritmo de clasificación “Naive Bayes”. El resultado se almacena en una columna denominada **Clasificación** (ver figura 44).

¹⁰ *DataFrame*: Son estructuras de datos bidimensionales.

```
#clasificando las fallas
with open(infile, 'r') as csvfile:
    filas = csv.reader(csvfile)
    datos_clasificados = []
    for fila in filas:
        sentence = fila[0]
        datos_clasificados.append(cl.classify(sentence))
datos=pd.DataFrame(datos_clasificados)
encabezado1=["Clasificación"]
datos.columns= encabezado1
```

Figura 44. Clasificación de la data usando Textblob
Fuente: elaboración propia.

El **quinto paso** consiste en agregar la columna creada al *DataFrame* de los datos de fallas que se están clasificando (ver figura 45), y mostrar el resultado obtenido (ver figura 46).

```
#fallas clasificadas
df=df.assign(Clasificacion=datos)
print("DataFrame modificado")
print("*****")
df
```

Figura 45. Código para agregar la clasificación
Fuente: elaboración propia.

	Fallas	Tiempo	Equipo	Clasificacion
0	fuga hidraulica en el cilindro del boom	3.0	SH01	HIDRAULICO
1	grieta en el bucket	0.5	SH01	MECANICO
2	sensores defectuosos	0.9	SH02	ELECTRICO
3	alta temperatura de aceite hidraulico	2.0	SH01	HIDRAULICO
4	fisura en el clam	1.5	SH02	MECANICO
5	cables malogrados	3.4	SH01	ELECTRICO
6	bajas rpm del motor diesel	1.2	SH02	MOTRIZ
7	tren de rodaje	1.6	SH02	ELECTRICO
8	bajas rpm del motor diesel	6.4	SH01	MOTRIZ
9	sensores malogrados	2.3	SH03	ELECTRICO
10	fuga de aceite hidraulico del bucket	7.9	SH03	HIDRAULICO
11	bajas rpm del motor diesel	4.6	SH03	MOTRIZ

Figura 46. Resultado obtenido.
Fuente: elaboración propia.

El **sexto paso** consiste en agrupar los datos por equipos para facilitar la construcción de la gráfica dinámica, la cual permitirá visualizar, de una manera interactiva, la cantidad de horas que los sistemas de cada máquina (hidráulico, eléctrico, motriz, mecánico, lubricación y refrigeración) han fallado (ver figura 47).

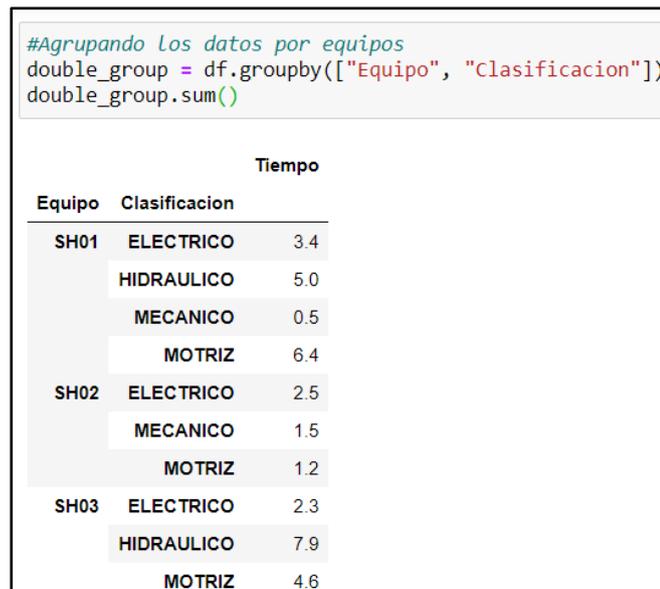


Figura 47. Tiempo de fallas en cada equipo
Fuente: elaboración propia.

El **séptimo paso** consiste en agrupar los datos en los sistemas señalados anteriormente para facilitar la construcción del diagrama de Pareto (ver figura 48),

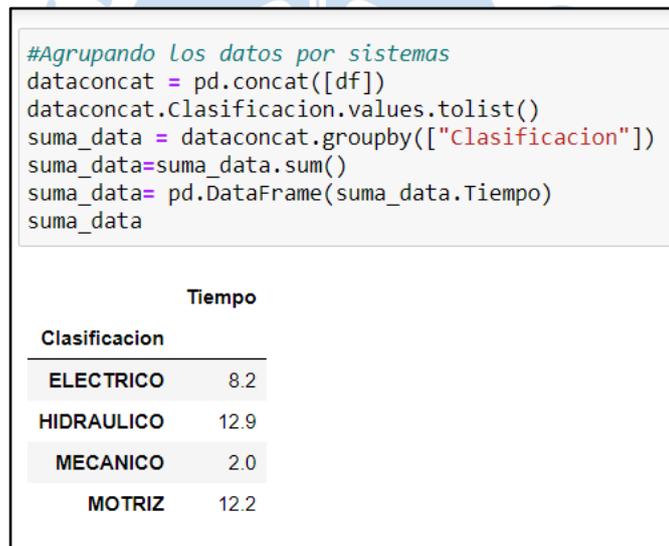


Figura 48. Tiempo de fallas en cada sistema
Fuente: elaboración propia.

El **octavo paso** es exportar y guardar los datos de fallas clasificadas en un archivo excel (ver figura 49).

```
#exportando a excel
df.to_excel("pareto_clasificado.xls")
```

Figura 49. Fallas clasificadas
Fuente: elaboración propia.

El **noveno paso** consiste en generar una gráfica de Pareto de toda la flota de máquinas PC400-6, para lo cual se emplea el código que se muestra en la figura 50. La gráfica de Pareto se puede observar en la figura 51.

```
df.index= df.Clasificacion.values.tolist()
suma_data = suma_data.sort_values(by='Tiempo',ascending=False)
suma_data["cumpercentage"] = suma_data["Tiempo"].cumsum()/suma_data["Tiempo"].sum()

fig, ax = plt.subplots()
ax.bar(suma_data.index, suma_data["Tiempo"], color=("orangered", "lightblue", "darkorange", "green"))
ax2 = ax.twinx()
ax2.plot(suma_data.index, suma_data["cumpercentage"], color="C1", marker="D", ms=10)
ax2.yaxis.set_major_formatter(PercentFormatter())
ax.tick_params(axis="y", colors="C0")
ax2.tick_params(axis="y", colors="C1")
plt.show()
```

Figura 50. Código para crear el diagrama de Pareto
Fuente: elaboración propia.

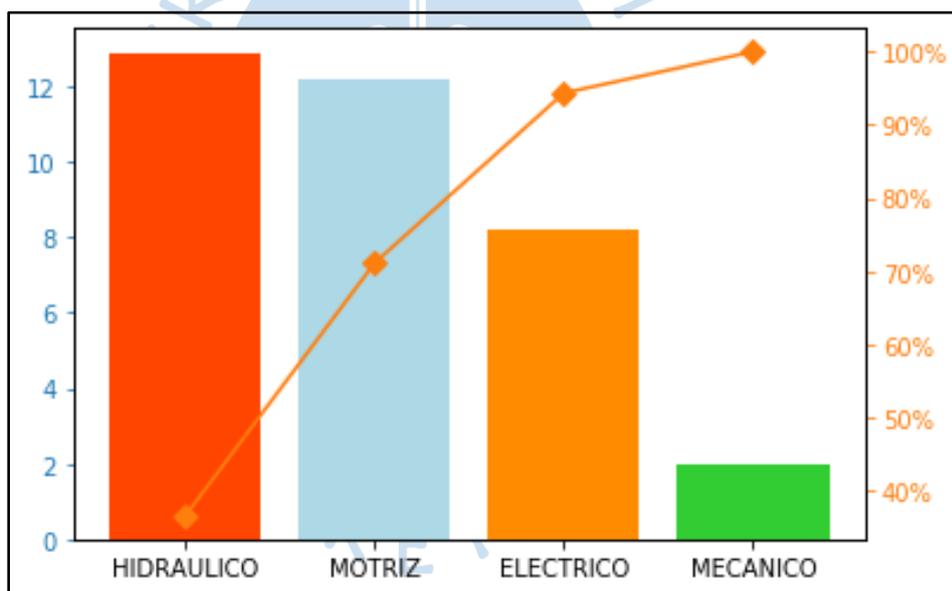


Figura 51. Gráfica de Pareto
Fuente: elaboración propia.

El **décimo paso**, consiste en generar una gráfica dinámica que muestre la información individual de cada máquina PC4000-6. Para ello se usa el código que se muestra en la figura 52 y el resultado obtenido se muestra en la figura 53.

```
# In[12]:
fig = px.bar(df, x="Clasificacion", y="Tiempo",
             color="Tiempo", animation_frame="Equipo",
             range_y=[0,10])
fig.show()
```

Figura 52. Código para generar la gráfica dinámica.
Fuente: elaboración propia.

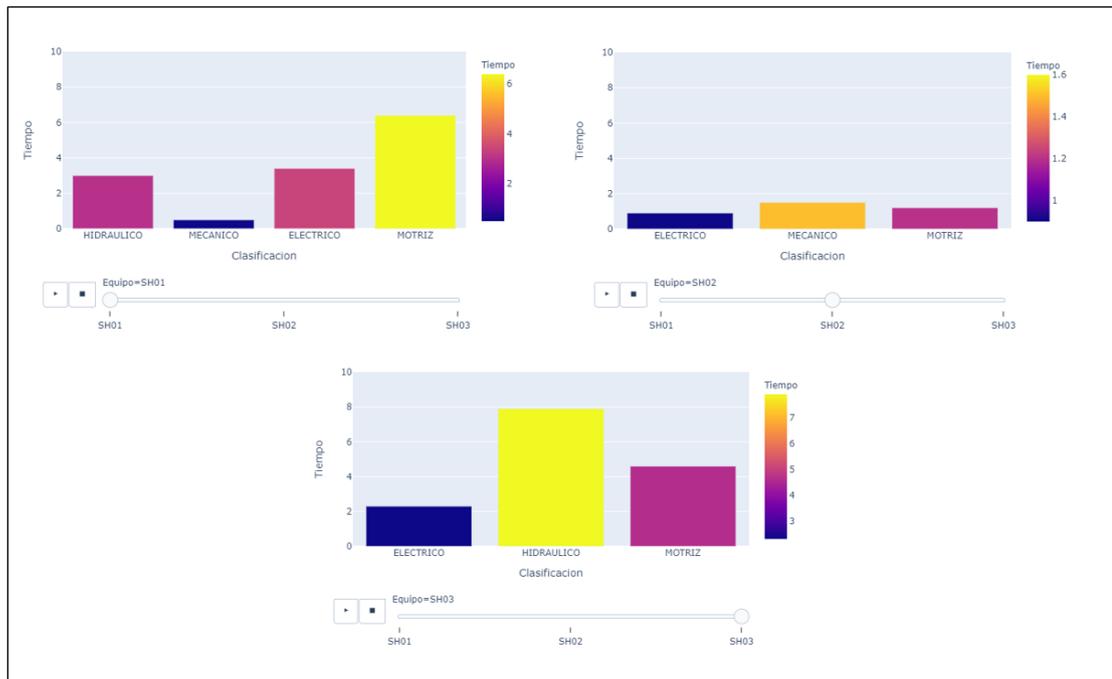


Figura 53. Gráficas dinámicas de cada equipo.
Fuente: elaboración propia.

Asimismo, es necesario capacitar adecuadamente a los operadores, con la finalidad de que la clasificación realizada por el algoritmo sea lo más confiable posible. Esta capacitación consistirá en el uso de las nuevas tecnologías, como almacenamiento en la nube, técnicas de *machine learning*, seguridad de los datos en la nube, entre otras.

En el **apéndice B** se muestra el código completo del sistema de clasificación para que sirva de guía en el desarrollo de sistemas inteligentes de clasificación.

4.2. Resultados

Se logró clasificar un conjunto de datos de texto (fallas) en los respectivos sistemas de la máquina, usando el algoritmo de clasificación de texto “Naive Bayes” (ver figura 54).

Las prácticas basadas en el uso de algoritmos de *machine learning* generan un importante ahorro de tiempo porque permiten que el planner de mantenimiento se centre en actividades concretas, como realizar una planificación eficaz y un adecuado control de la gestión de mantenimiento de la flota de máquinas PC4000-6.

Para que el personal de mantenimiento pueda tener acceso a los datos de fallas en tiempo real, es importante guardar los archivos de una manera sencilla y rápida en la nube, mediante el uso de dispositivos electrónicos (computadoras portátiles, *tablets* y *smartphones*) como herramientas de apoyo. En este proyecto de tesis se optó por guardar los archivos en “Google Drive”, debido a que proporciona almacenamiento gratuito (ver figura 30).

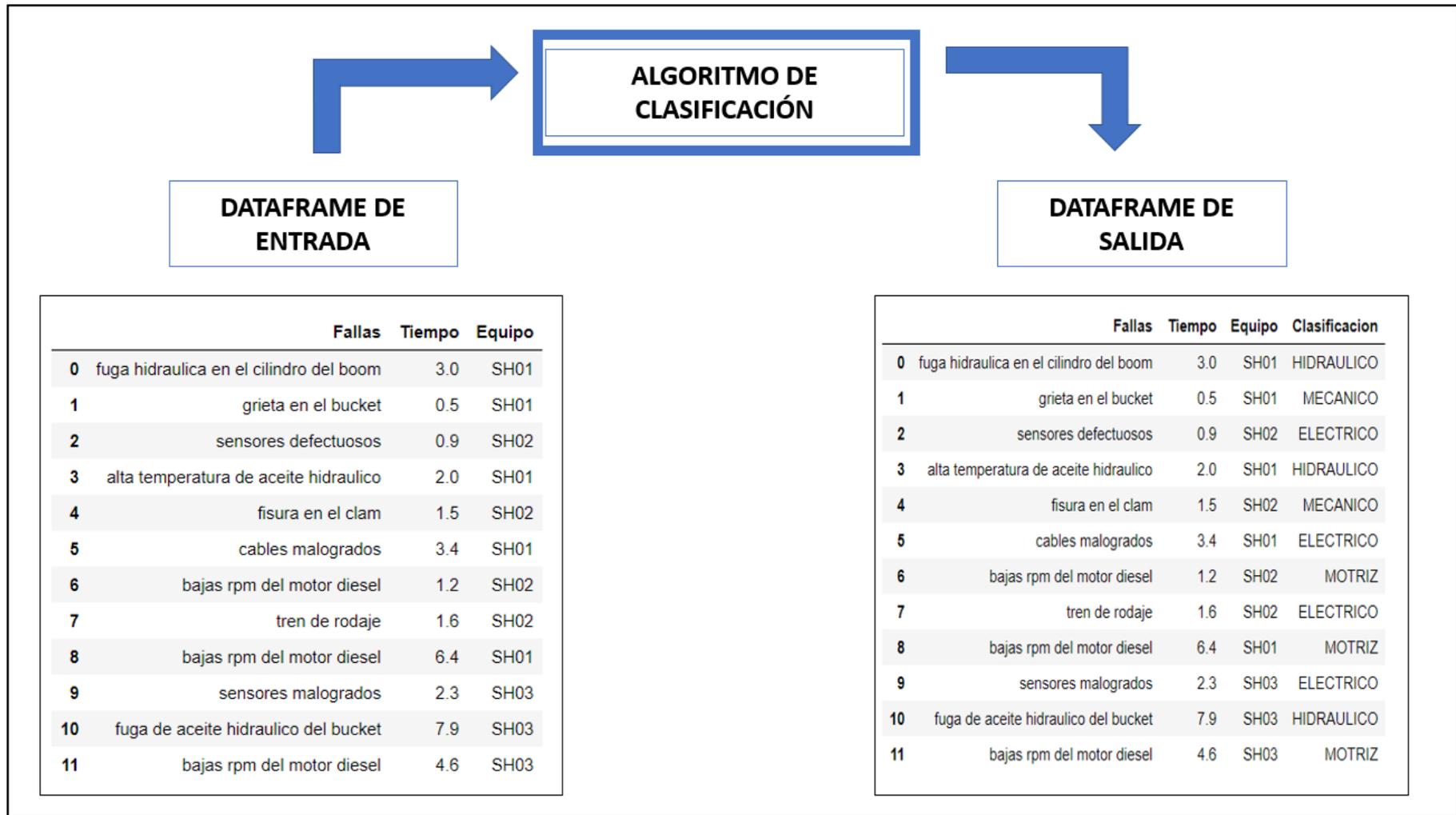


Figura 54. Resultados
Fuente: elaboración propia.



Conclusiones

En el presente trabajo se desarrolló un sistema inteligente de clasificación de datos (en este caso, de fallas) de una **flota de máquinas PC4000-6 Komatsu**, el cual se explicó a detalle en el capítulo 4. Con este sistema se logró clasificar en breve tiempo (**unos minutos**) las fallas con una precisión del **85%**; lo que representa, para el personal que se dedica a clasificar los datos de fallas de manera manual (**2 de las 8 horas de su jornada laboral**), un importante ahorro de tiempo, que puede ser empleado para determinar de una manera más rápida las estrategias adecuadas para la planificación, el control y los procesos de mejora continua del mantenimiento.

Se realizó el estudio de las características y especificaciones técnicas de la máquina PC4000-6, lo que permitió entender el funcionamiento de esta, y facilitar el análisis detallado de la data histórica de fallas de la flota de máquinas.

Con el análisis de los datos de las fallas de la flota de máquinas PC4000-6 utilizando el diagrama de Pareto, se concluye que el **80%** de los problemas pertenecen a los sistemas **hidráulico** y **eléctrico**. Por consiguiente, se necesita prestar mayor atención a estos sistemas para mejorar la disponibilidad de la flota de máquinas PC4000-6.

Se logró elaborar la taxonomía de la máquina PC4000-6 clasificándola en sistemas, subsistemas y componentes bajo la Norma ISO 14224:2016, lo que permitió reclasificar los datos de fallas bajo un formato estándar y, en consecuencia, asegurar la calidad de estos.

Comparando diversos algoritmos de *machine learning* (figura 38), ha sido posible realizar un estudio con el que se obtuvieron las siguientes predicciones: *Logistic Regression*, **70%**; *Naive Bayes*, **85%**; *Support Vector Machine*, **75%**; y *Decision Tree*, **60%**. En consecuencia, se concluye que, para una pequeña cantidad de datos, el modelo de **Naive Bayes** presenta un mejor desempeño en comparación a los otros modelos.



Recomendaciones

El éxito de este clasificador requiere de una adecuada data de entrenamiento, por lo cual se necesita del compromiso de las personas con mayor experiencia en el equipo, como técnicos de más experiencia, especialistas de cada área, manuales del fabricante, entre otros.

La aplicación desarrollada en este proyecto ha sido un prototipo creado en *APP Inventor* debido a la facilidad de programación, no obstante, para mejorar la experiencia se recomienda desarrollar el aplicativo en *Android Studio*.

Con la finalidad de aumentar la utilidad de este sistema, se recomienda brindar más herramientas de ayuda para el personal de mantenimiento como *RCM (Reliability Centred Maintenance)*, *ACR (Análisis de Causa Raíz)*, *Diagrama Jack-Knife*, entre otros.

El sistema de clasificación se construyó usando un algoritmo de *machine learning*. Si se desea mejorar las predicciones, se recomienda utilizar algoritmos de *deep learning* como las redes neuronales profundas.

Se recomienda también implementar en el sistema de clasificación una opción que permita ver qué elementos son los que podrían presentar mayores fallas, con base en alguna relación entre sus parámetros característicos como velocidad, temperatura, presión y caudal; de manera que, en conjunto con el área de logística, se puedan gestionar adecuadamente los materiales y repuestos.



Referencias bibliográficas

- Allison, J. (2019). *How to Send Data to a Google Sheet with MIT App Inventor*. Retrieved from <https://www.youtube.com/watch?v=SUFxL1awjO8>
- Antamina. (2019). *Proceso de producción*. Retrieved from <https://www.antamina.com/proceso-productivo/>
- Armulia Sánchez, V. D. (2012). *Mantenimiento predictivo de palas hidráulicas Komatsu PC4000-6 en la minera Barrick-Lagun Norte. Informe de suficiencia profesional para obtener el título de ingeniero mecánico electricista*. Lima: Universidad Nacional de Ingeniería.
- Bahit, E. (2012). *Python para principiantes*. Buenos Aires.
- Carrasco, O. (2019). *Towards data science*. Retrieved from <https://towardsdatascience.com/support-vector-machines-for-classification-fc7c1565e3>
- CCTMexico. (2019). *Agregar una columna a un DataFrame*. Retrieved from <https://www.youtube.com/watch?v=Es7SCnzlXsE>
- Chadwick, J. (1998). *Hydraulic Mining Excavators*. International mining.
- Commission, International Electrotechnical. (2015). *IEC 60050-192*. Londres: International Standar.
- Dataquest. (2019). *Tutorial: Clasificación de texto en Python usando SPACY*. Retrieved from <https://www.dataquest.io/blog/tutorial-text-classification-in-python-using-spacy/>
- Gica Ingenieros. (2017). *Mantenimiento de maquinaria pesada*. Trujillo.
- Gica Ingenieros. (2018). *Planificación y control de gestión de mantenimiento*. Trujillo.

- Gonzales Fernández F. J. (2015). *Teoría y práctica del mantenimiento industrial avanzado*. España: Fundación Confemetal.
- Gonzales, L. (2018). *Diferencia entre Inteligencia Artificial – Machine Learning – Deep Learning*. Retrieved from <https://ligdigonzalez.com/diferencia-entre-inteligencia-artificial-machine-learning-deep-learning/>
- González, L. (2019). *Importar y exportar datos con python*. Retrieved from <https://www.youtube.com/watch?v=uaiYBm-ayio&t=6s>
- González, L. (2019). *Machine Learning*. Retrieved from <http://ligdigonzalez.com/blog/>
- Google. (2019). *Authenticate your users*. Retrieved from <https://developers.google.com/drive/api/v2/about-auth>
- Google. (2019). *Download files*. Retrieved from <https://developers.google.com/drive/api/v3/manage-downloads>
- Google. (2019). *Method: people.get*. Retrieved from <https://developers.google.com/people/api/rest/v1/people/get?authuser=1>
- Google. (2019). *Store application-specific data*. Retrieved from <https://developers.google.com/drive/api/v2/appdata>
- Google. (2019). *Upload files*. Retrieved from <https://developers.google.com/drive/api/v2/manage-uploads>
- Harry, Z. (2004). *The Optimality of Naive Bayes*. Canada: University of New Brunswick.
- Holmberg K., Adgar A., Arnaiz A., Jantunen E., Mascolo J. y Mekid S. (2010). *E-maintenance*. London: Springer.
- Illarbe Izquierdo, L. y Vegas Chiyón, S. (2008). *Herramientas para la gestión de la calidad*. Piura: Universidad de Piura.
- Instituto de Ingeniería del Conocimiento. (2019). *Procesamiento del lenguaje Natural*. Retrieved from <https://www.iic.uam.es/soluciones/inteligencia-de-cliente/procesamiento-lenguaje-natural/>
- Komatsu. (2009). *Manual de Operaciones PC4000-6*. Alemania.
- Komatsu. (2009). *Manual de taller PC4000-6D*. Alemania.

- Komatsu. (2019). *Carbody*. Retrieved from <https://www.komatsulatioamerica.com/>
- Komatsu. (2019). *Componentes de una excavadora hidráulica*. Retrieved from <https://www.komatsulatioamerica.com/>
- Komatsu. (2019). *Corona de giro*. Retrieved from <https://www.komatsulatioamerica.com/>
- Komatsu. (2019). *Equipo de trabajo*. Retrieved from <https://www.facebook.com/KomatsuPeru/>
- Komatsu. (2019). *Escavadora hidráulica PC4000-6*. Retrieved from <https://www.komatsulatioamerica.com/peru/productos/pc4000-6-bh-d/>
- Komatsu. (2019). *Estructura superior*. Retrieved from <https://www.komatsulatioamerica.com/>
- Komatsu. (2019). *Komatsu Latinoamérica*. Retrieved from <https://www.komatsulatioamerica.com/>
- Komatsu. (2019). *Pala Hidráulica PC4000*. Retrieved from <https://www.komatsulatioamerica.com/>
- Komatsu. (2019). *Personal de mantenimiento*. Retrieved from <https://www.facebook.com/KomatsuPeru/>
- Komatsu. (2019). *Tren de rodaje*. Retrieved from <https://www.komatsulatioamerica.com/>
- Komatsu Construction. (2018). *Komatsu Technology Vision*. Retrieved from <https://www.youtube.com/watch?v=l9nmFSQpHEk>
- Komatsu Perú. (2011). *Curso completo de la máquina PC4000-6*. Lima: Perú.
- Merma Rodríguez, J. (2018). *Implementación del mantenimiento Centrado en la confiabilidad (RCM2) en las palas hidráulicas PC4000-6 Komatsu para el incremento de la disponibilidad. Informe de suficiencia profesional para obtener el título de ingeniero mecánico electricista*. Arequipa: Universidad Nacional de San Agustín de Arequipa.
- Michie, D., Spiegelhalter, D. J., & Taylor, C. C. (1994). *Machine learning, neural and statistical classification*. New York: Ellis Horwood.

- Ministerio de Energía y Minas. (2019). *Anuario Minero 2018*. Retrieved from [https://www.minem.gob.pe/minem/archivos/file/Mineria/PUBLICACIONES/ANUAR_IOS/2018/AM2018\(VF\).pdf](https://www.minem.gob.pe/minem/archivos/file/Mineria/PUBLICACIONES/ANUAR_IOS/2018/AM2018(VF).pdf)
- Ng, Andrew. (2019). *Aprendizaje automático*. Retrieved from Universidad de Stanford & Coursera: <https://es.coursera.org/learn/machine-learning>
- Organización Internacional de Estandarización. (2016). *Industrias de petróleo, petroquímica y gas natural- Recolección e intercambio de datos de confiabilidad y mantenimiento de equipos (ISO 14224:2016)*. Londres: BSI standards Limited.
- Pedregosa, F. (2019). *Scikit-learn: Machine Learning in Python*. Retrieved from https://scikit-learn.org/stable/supervised_learning.html#supervised-learning
- Pistarelli, A. (2010). *Manual de mantenimiento: ingeniería, gestión y organización*. Argentina: Buenos Aires.
- Plotly. (2019). *Gráficas animadas*. Retrieved from <https://plot.ly/python/animations/>
- Programador MP. (2018). *Historia y características principales de Python*. Retrieved from https://www.youtube.com/watch?v=E_e1252EO1Y&t=188s
- Real Academia Española. (2019). *Definición de algoritmo*. Retrieved from <https://dle.rae.es/algoritmo>
- Rojas, C. B. (2019). *Ley de Pareto aplicada a la fiabilidad*. *Revista de ingeniería mecánica del Instituto Superior Politécnico José Antonio Echevarría de Cuba*. Retrieved from <https://www.redalyc.org/pdf/2251/225118188010.pdf>
- Salas, J. G. (2019). *Curso completo de Machine Learning: Data Science en Python*. Retrieved from <https://www.udemy.com/course/machinelearningpython/>
- Sales, M. (2002). *Diagrama de Pareto*. Retrieved from <https://www.gestiopolis.com/diagrama-de-pareto/>
- Sleipner. (2019). *Traslado de una Front Shovel*. Retrieved from <https://www.sleipner.fi/>
- Stackoverflow. (2019). *How to make Pareto Chart in python?* Retrieved from <https://stackoverflow.com/questions/53577630/how-to-make-pareto-chart-in-python>

Stackoverflow. (2019). *Looking for example using MediaFileUpload*. Retrieved from <https://stackoverflow.com/questions/11472401/looking-for-example-using-mediafileupload>

Stackoverflow. (2019). *Textblob analysis on a csv file*. Retrieved from <https://stackoverflow.com/questions/35559199/textblob-sentiment-analysis-on-a-csv-file>

Steven Bird, E. K. (2009). *Natural Language Processing with Python*. United States of America.: O'Reilly.

Steven, L. (2019). *Textblob*. Retrieved from <https://textblob.readthedocs.io/en/dev/classifiers.html>

Suthaharan, S. (2016). *Machine Learning Models and Algorithms for big data*. Boston: Springer US.

Technology, Massachusetts Institute. (2019). *MIT inventor*. Retrieved from <https://appinventor.mit.edu/about-us>

Villaronga Maicas, M. (1997). *Curso de introducción a la oleohidráulica* . Barcelona: Vickers.

Winston, P. (2014). *MIT OpenCourseWare*. Retrieved from https://www.youtube.com/watch?v=_PwhiWxHK8o



Apéndices





Apéndice A. Manual de usuario del aplicativo móvil PC4000-6

RECOLECCION DE DATOS DE FALLAS

"PC4000"

MANUAL DE USUARIO

1.) INTERFAZ DE INICIO



En primer lugar se debe pulsar en el icono de la aplicación creada, una vez realizada esta acción se le mostrará un interfaz y se debe pulsar en el botón "INGRESAR FALLAS".

2.) INTERFAZ DE REGISTRO DE FALLAS



Se mostrará una nueva interfaz la cual consta de dos campos de texto, una lista desplegable y dos botones: "Guardar" y "Limpiar".

3.) REGISTRAR FALLA



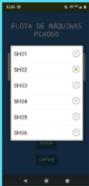
Para registrar la descripción de fallas, debe presionar en el recuadro que se encuentra debajo del texto "Descripción de la falla".

4.) REGISTRAR TIEMPO DE PARADA



Para registrar el tiempo de parada debe presionarse en el recuadro que se encuentra debajo del texto "Duración de la falla".

5.) SELECCIONAR MÁQUINA



Para registrar la descripción de fallas, debe presionar en el recuadro que se encuentra debajo del texto "Código del equipo" en el cual aparecerá una lista desplegable para seleccionar el equipo en el cuál se presentó la falla.

6.) GUARDAR FALLA Y LIMPIAR PANTALLA



PEL botón Guardar se empleará para registrar todos los datos de la falla y enviarlos a Google Drive, a su vez el botón Limpiar se empleará para limpiar todos los datos que han sido digitados, con la finalidad de reingresar un nuevo reporte de fallas.

Apéndice B. Código completo del sistema de clasificación

```

# In[1]:
#librerias
import pandas as pd
import numpy as np
import csv
import matplotlib.pyplot as plt
from matplotlib.ticker import PercentFormatter
from textblob.classifiers import NaiveBayesClassifier
import chart_studio
import chart_studio.plotly as py
from plotly.graph_objs import *
import plotly.express as px

# In[2]:
#datos de entrenamiento
with open('pareto.csv','r') as fp:
    cl=NaiveBayesClassifier(fp,format="csv")

# In[3]:
#fallas a clasificar
infile = 'pareto1.csv'
datos= pd.read_csv(infile,header=None)
encabezado=["Fallas", "Tiempo", "Equipo"]
datos.columns= encabezado
df= pd.DataFrame(datos)
print("DataFrame a clasificar")
print("*****")
df

# In[4]:
#clasificando las fallas
with open(infile, 'r') as csvfile:
    filas = csv.reader(csvfile)
    datos_clasificados =[]
    for fila in filas:
        sentence = fila[0]
        datos_clasificados.append(cl.classify(sentence))
datos=pd.DataFrame(datos_clasificados)
encabezado1=["Clasificación"]
datos.columns= encabezado1

# In[5]:
#fallas clasificadas
df=df.assign(Clasificacion=datos)

```

```

print("DataFrame modificado")
print("*****")
df

# In[6]:
#Agrupando los datos por equipos
double_group = df.groupby(["Equipo", "Clasificacion"])
double_group.sum()

# In[7]:
#Agrupando los datos por sistemas
dataconcat = pd.concat([df])
dataconcat.Clasificacion.values.tolist()
suma_data = dataconcat.groupby(["Clasificacion"])
suma_data=suma_data.sum()
suma_data= pd.DataFrame(suma_data.Tiempo)
suma_data

# In[8]:
#exportando a excel
df.to_excel("pareto_clasificado.xls")

## Gráfico de Pareto 1
# In[9]:
df.index= df.Clasificacion.values.tolist()
suma_data = suma_data.sort_values(by='Tiempo',ascending=False)
suma_data["cumpercentage"] =
suma_data["Tiempo"].cumsum()/suma_data["Tiempo"].sum()*100
fig, ax = plt.subplots()
ax.bar(suma_data.index, suma_data["Tiempo"],
color=("orangered", "lightblue", "darkorange", "limegreen", "lime"), align = 'center')
ax2 = ax.twinx()
ax2.plot(suma_data.index, suma_data["cumpercentage"], color="C1", marker="D",
ms=7)
ax2.yaxis.set_major_formatter(PercentFormatter())
ax.tick_params(axis="y", colors="C0")
ax2.tick_params(axis="y", colors="C1")
plt.show()

## Gráfico Pareto 2
# In[12]:
fig = px.bar(df, x="Clasificacion", y="Tiempo",
color="Tiempo",animation_frame="Equipo", animation_group = "Clasificacion",
range_y=[0,10])
fig.show()

```



Anexos





Anexo A. Logistic Regression

```
import pandas as pd
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectorizer
from sklearn.base import TransformerMixin
from sklearn.pipeline import Pipeline
import string
from spacy.lang.es.stop_words import STOP_WORDS
from spacy.lang.es import Spanish
import spacy
from sklearn.model_selection import train_test_split
from sklearn import metric
#*****
*****
infile = 'revisar2.csv'
df_amazon= pd.read_csv(infile)
punctuations = string.punctuation
stop_words = spacy.lang.es.stop_words.STOP_WORDS
parser = Spanish()
def spacy_tokenizer(sentence):
    mytokens = parser(sentence)
    mytokens = [ word.lemma_.lower().strip() if word.lemma_ != "-PRON-" else
word.lower_ for word in mytokens ]
    mytokens = [ word for word in mytokens if word not in stop_words and word not in
punctuations ]
    return mytokens
class predictors(TransformerMixin):
    def transform(self, X, **transform_params):
        return [clean_text(text) for text in X]
    def fit(self, X, y=None, **fit_params):
        return self
    def get_params(self, deep=True):
        return {}
def clean_text(text):
    return text.strip().lower()
class DenseTransformer(TransformerMixin):
    def fit(self, X, y=None, **fit_params):
        return self
    def transform(self, X, y=None, **fit_params):
        return X.todense()
bow_vector = CountVectorizer(tokenizer = spacy_tokenizer, ngram_range=(1,1))
tfidf_vector = TfidfVectorizer(tokenizer = spacy_tokenizer)
X = df_amazon['falla'] # the features we want to analyze
ylabels = df_amazon['sistema'] # the labels, or answers, we want to test against
X_train, X_test, y_train, y_test = train_test_split(X, ylabels, test_size=0.2)
```

```
from sklearn.linear_model import LogisticRegression
classifier = LogisticRegression()
pipe = Pipeline([("cleaner", predictors()),
                 ('vectorizer', bow_vector),
                 ('classifier', classifier)])
pipe.fit(X_train,y_train)
predicted = pipe.predict(X_test)
print("Logistic Regression :",metrics.accuracy_score(y_test, predicted))
```



Anexo B. Naive Bayes

```

import pandas as pd
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectorizer
from sklearn.base import TransformerMixin
from sklearn.pipeline import Pipeline
import string
from spacy.lang.es.stop_words import STOP_WORDS
from spacy.lang.es import Spanish
import spacy
from sklearn.model_selection import train_test_split
from sklearn import metric
#*****
*****
infile = 'revisar2.csv'
df_amazon= pd.read_csv(infile)
punctuations = string.punctuation
stop_words = spacy.lang.es.stop_words.STOP_WORDS
parser = Spanish()
def spacy_tokenizer(sentence):
    mytokens = parser(sentence)
    mytokens = [ word.lemma_.lower().strip() if word.lemma_ != "-PRON-" else
word.lower_ for word in mytokens ]
    mytokens = [ word for word in mytokens if word not in stop_words and word not in
punctuations ]
    return mytokens
class predictors(TransformerMixin):
    def transform(self, X, **transform_params):
        return [clean_text(text) for text in X]
    def fit(self, X, y=None, **fit_params):
        return self
    def get_params(self, deep=True):
        return {}
def clean_text(text):
    return text.strip().lower()
class DenseTransformer(TransformerMixin):
    def fit(self, X, y=None, **fit_params):
        return self
    def transform(self, X, y=None, **fit_params):
        return X.todense()
bow_vector = CountVectorizer(tokenizer = spacy_tokenizer, ngram_range=(1,1))
tfidf_vector = TfidfVectorizer(tokenizer = spacy_tokenizer)
X = df_amazon['falla'] # the features we want to analyze
ylabels = df_amazon['sistema'] # the labels, or answers, we want to test against
X_train, X_test, y_train, y_test = train_test_split(X, ylabels, test_size=0.2)

```

```
from sklearn.naive_bayes import GaussianNB
classifier= GaussianNB()
pipe = Pipeline([("cleaner", predictors()),
                 ('vectorizer', bow_vector),
                 ('to_dense', DenseTransformer()),
                 ('classifier', classifier)])
pipe.fit(X_train,y_train)
predicted = pipe.predict(X_test)
print("Naive Bayes:",metrics.accuracy_score(y_test, predicted))
```



Anexo C. Support Vector Machine (SVM)

```

import pandas as pd
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer, TfidfVectorizer
from sklearn.base import TransformerMixin
from sklearn.pipeline import Pipeline
import string
from spacy.lang.es.stop_words import STOP_WORDS
from spacy.lang.es import Spanish
import spacy
from sklearn.model_selection import train_test_split
from sklearn import metric
#*****
*****
infile = 'revisar2.csv'
df_amazon= pd.read_csv(infile)
punctuations = string.punctuation
stop_words = spacy.lang.es.stop_words.STOP_WORDS
parser = Spanish()
def spacy_tokenizer(sentence):
    mytokens = parser(sentence)
    mytokens = [ word.lemma_.lower().strip() if word.lemma_ != "-PRON-" else
word.lower_ for word in mytokens ]
    mytokens = [ word for word in mytokens if word not in stop_words and word not in
punctuations ]
    return mytokens
class predictors(TransformerMixin):
    def transform(self, X, **transform_params):
        return [clean_text(text) for text in X]
    def fit(self, X, y=None, **fit_params):
        return self
    def get_params(self, deep=True):
        return {}
def clean_text(text):
    return text.strip().lower()
class DenseTransformer(TransformerMixin):
    def fit(self, X, y=None, **fit_params):
        return self
    def transform(self, X, y=None, **fit_params):
        return X.todense()
bow_vector = CountVectorizer(tokenizer = spacy_tokenizer, ngram_range=(1,1))
tfidf_vector = TfidfVectorizer(tokenizer = spacy_tokenizer)
X = df_amazon['falla'] # the features we want to analyze
ylabels = df_amazon['sistema'] # the labels, or answers, we want to test against
X_train, X_test, y_train, y_test = train_test_split(X, ylabels, test_size=0.2)

```

```
from sklearn import svm
classifier = svm.SVC()
pipe = Pipeline([("cleaner", predictors()),
                 ('vectorizer', bow_vector),
                 ('to_dense', DenseTransformer()),
                 ('classifier', classifier)])
pipe.fit(X_train,y_train)
predicted = pipe.predict(X_test)
print("SVM:",metrics.accuracy_score(y_test, predicted))
```



Anexo D. Decision Tree

```

import pandas as pd
import matplotlib.pyplot as plt
from sklearn.feature_extraction.text import CountVectorizer,TfidfVectorizer
from sklearn.base import TransformerMixin
from sklearn.pipeline import Pipeline
import string
from spacy.lang.es.stop_words import STOP_WORDS
from spacy.lang.es import Spanish
import spacy
from sklearn.model_selection import train_test_split
from sklearn import metric
#*****
*****
infile = 'revisar2.csv'
df_amazon= pd.read_csv(infile)
punctuations = string.punctuation
stop_words = spacy.lang.es.stop_words.STOP_WORDS
parser = Spanish()
def spacy_tokenizer(sentence):
    mytokens = parser(sentence)
    mytokens = [ word.lemma_.lower().strip() if word.lemma_ != "-PRON-" else
word.lower_ for word in mytokens ]
    mytokens = [ word for word in mytokens if word not in stop_words and word not in
punctuations ]
    return mytokens
class predictors(TransformerMixin):
    def transform(self, X, **transform_params):
        return [clean_text(text) for text in X]
    def fit(self, X, y=None, **fit_params):
        return self
    def get_params(self, deep=True):
        return {}
def clean_text(text):
    return text.strip().lower()
class DenseTransformer(TransformerMixin):
    def fit(self, X, y=None, **fit_params):
        return self
    def transform(self, X, y=None, **fit_params):
        return X.todense()
bow_vector = CountVectorizer(tokenizer = spacy_tokenizer, ngram_range=(1,1))
tfidf_vector = TfidfVectorizer(tokenizer = spacy_tokenizer)
X = df_amazon['falla'] # the features we want to analyze
ylabels = df_amazon['sistema'] # the labels, or answers, we want to test against
X_train, X_test, y_train, y_test = train_test_split(X, ylabels, test_size=0.2)

```

```
from sklearn.tree import DecisionTreeClassifier
classifier=DecisionTreeClassifier()
pipe = Pipeline([("cleaner", predictors()),
                 ('vectorizer', bow_vector),
                 ('to_dense', DenseTransformer()),
                 ('classifier', classifier)])
pipe.fit(X_train,y_train)
predicted = pipe.predict(X_test)
print("Decision Tree:",metrics.accuracy_score(y_test, predicted))
```

