



TECNOLÓGICO NACIONAL DE MÉXICO
INSTITUTO TECNOLÓGICO DE HERMOSILLO



DISEÑO DE UN MODELO DE GESTIÓN DE DATOS PARA ANALÍTICA INTELIGENTE EN LA INDUSTRIA

Tesis que presenta

Eduardo Antonio Hinojosa Palafox

como requisito para obtener el grado de

Doctorado en Ciencias de la Ingeniería

Director de tesis: **Dr. Oscar Mario Rodriguez Elias**

Hermosillo, Sonora, México

Septiembre de 2022

REFERENTES RESPONSABLES

Título: Diseño de un modelo de gestión de datos para analítica inteligente en la industria

Autor: Eduardo Antonio Hinojosa Palafox

Director de tesis: Dr. Oscar Mario Rodríguez Elías, ITH

Comité tutorial: Dr. Héctor Guerra Crespo, IITG
Dr. Madain Pérez Patricio, ITTG
Dr. José Antonio Hoyo Montaña, ITH
Dr. Jesús Horacio Pacheco Ramírez, UNISON
Dr. José Manuel Nieto Jalil, ITESM CSN

Instituto Tecnológico de Hermosillo
División de Estudios de Posgrado e Investigación

HERMOSILLO, SON., 3 DE AGOSTO DE 2022
SECCIÓN: DIV. EST. POS. E INV.
No. OFICIO: DEPI/155/22.
ASUNTO: AUTORIZACIÓN DE
IMPRESIÓN DE TESIS.

**C. EDUARDO ANTONIO HINOJOSA PALAFOX
PRESENTE**

De acuerdo con los Lineamientos para la Operación de los Estudios de Posgrado en el Tecnológico Nacional de México y las disposiciones en este Instituto, y habiendo cumplido con todas las indicaciones que el Comité Tutorial, compuesta por Dr. Oscar Mario Rodríguez Elías, Dr. Héctor Guerra Crespo, Dr. Madain Pérez Patricio, Dr. José Antonio Hoyo Montaña, Dr. Jesús Horacio Pacheco Ramírez y el Dr. José Manuel Nieto Jalil, realizó con respecto a su trabajo de tesis titulado "DISEÑO DE UN MODELO DE GESTION DE DATOS PARA ANALÍTICA INTELIGENTE EN LA INDUSTRIA", elaborado por Usted, como prueba escrita para obtener el Grado de Doctor en Ciencias de la Ingeniería, la División de Estudios de Posgrado e Investigación de este Instituto, concede la Autorización para que proceda a la impresión de la tesis.

Deseándole éxito en su vida profesional, quedo de usted.

ATENTAMENTE

*Excelencia en Educación Tecnológica-
En el Esfuerzo Común, la Grandeza de Todos*



DR. GERMÁN ALONSO RUÍZ DOMÍNGUEZ
JEFE DE LA DIVISIÓN DE ESTUDIOS DE
POSGRADO E INVESTIGACIÓN



GARD/momv*



Av. Tecnológico #13 Col. El Salvario C.P. 83770 Hermosillo, Sonora. Tel. (562) 2605500, ext 136
correo: depi_hermosillo@tecnm.mx | www.ihumt



Hermosillo, Sonora, 01/Agosto/2022.

**GERMÁN ALONSO RUIZ DOMÍNGUEZ
JEFE DE LA DIVISIÓN DE ESTUDIOS DE
POSGRADO E INVESTIGACIÓN
INSTITUTO TECNOLÓGICO DE HERMOSILLO
PRESENTE**

Por este conducto, los integrantes de Comité Tutorial del C. **Eduardo Antonio Hinojosa Palafox**, con número de control **D02330027**, del Doctorado en Ciencias de la Ingeniería, le informamos que hemos revisado el trabajo de tesis profesional titulado: **“Diseño de un modelo de gestión de datos para analítica inteligente en la industria”**; y hemos encontrado que se han realizado todas las correcciones y observaciones que se le indicaron, por lo que se acuerda aceptar el documento de tesis y le solicitamos la autorización de impresión definitiva.

Sin otro particular, aprovecho la ocasión para enviarle un cordial saludo.

DIRECTOR DE TESIS

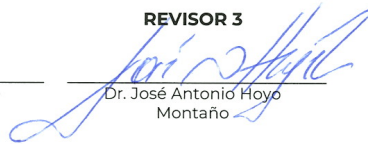
Dr. Oscar Mario Rodríguez Elías

REVISOR 1

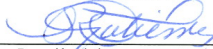
Dr. Héctor Guerra Crespo

REVISOR 2

Dr. Madari Pérez Patricio

REVISOR 3Dr. José Antonio Hoyo
Montaño**REVISOR 4**Dr. Jesús Horacio Pacheco
Ramírez**REVISOR 5**

Dr. José Manuel Nieto Jalil

SUPLENTE 1Dr. Guillermo Valencia
Palomo**SUPLENTE 2**Dr. Rosalía del Carmen Gutiérrez
Urquidez

C.p. Archivo



Av. Tecnológico #115 Col. El Sahuaro C.P. 83170 Hermosillo, Sonora. Tel. (662) 2606500, ext. 136
correo: depi_hermosillo@tecnm.mx | www.ich.mx





CARTA CESIÓN DE DERECHOS

En la ciudad de Hermosillo Sonora a el día 15 de agosto del año 2022 el que suscribe C. Eduardo Antonio Hinojosa Palafox, alumno del Doctorado en Ciencias de la Ingeniería adscrito a la División de Estudios de Posgrado e Investigación, manifiesta que es autor intelectual del presente trabajo de Tesis titulado “Diseño de un Modelo de Gestión de Datos para Analítica Inteligente en la Industria” bajo la dirección del Dr. Oscar Mario Rodríguez Elías y ceden los derechos del mismo al Tecnológico Nacional de México/Instituto Tecnológico de Hermosillo, para su difusión, con fines académicos y de investigación.

Los usuarios de la información no deben de reproducir el contenido textual, graficas, tablas o datos contenidos sin el permiso expreso del autor y del director del trabajo. Este puede ser obtenido a la dirección de correo electrónico siguiente: omrodriguez@hermosillo.tecnm.mx. Una vez otorgado el permiso se deberá expresar el agradecimiento correspondiente y citar la fuente del mismo.

ATENTAMENTE

Eduardo Antonio Hinojosa Palafox



Av. Tecnológico #115 Col. El Sahuaro C.P. 83170 Hermosillo, Sonora. Tel. (662) 2606500, ext 136
correo: posgrado@hermosillo.tecnm.mx tecnm.mx | www.ith.mx



Ricardo
2022 Flores
Año de
Magón
PRELUSOR DE LA REVOLUCIÓN MEXICANA

Resumen

La industria actual enfrenta el reto de integrar datos de equipos de la planta y datos externos, compartir servicios de datos, lograr la interconexión y la interoperabilidad entre el espacio físico y el espacio cibernético. A través de los sistemas de analítica industrial es posible identificar ideas, patrones o modelos útiles necesarios para la innovación sostenible. Tales sistemas son inherentemente complejos, dada la necesidad de alinear el Internet de las Cosas, el Cómputo en la Nube, el Big Data y el modelado de métodos basados en datos con la experiencia en el campo. Por lo anterior, la creación de plataformas tecnológicas para promover servicios de optimización enfrenta los desafíos de los sistemas ciberfísicos industriales que deben considerar un enfoque novedoso para el diseño de una arquitectura de referencia que integre la convergencia de tecnologías en la analítica de Big Data industrial y el aprendizaje máquina.

Por otro lado, los sistemas de plantas industriales están compuestos de numerosos componentes y operan en una variedad de condiciones, en tales circunstancias, la detección y el diagnóstico de fallas basados en datos enfrentan los desafíos de no contar con suficientes datos históricos, además de la falta de rapidez para detectar nuevas características. El enfoque del aprendizaje no supervisado puede ayudar a beneficiarse de estos métodos para la detección temprana de fallas, menos dependiente de conocimientos previos y experiencia diagnóstica al procesar Big Data, mientras se asegura minimizar las fallas y sus altos costos de reparación.

Este trabajo de investigación presenta un enfoque metodológico para el diseño de una arquitectura de referencia para analítica de Big Data industrial que provee servicios de optimización para la detección temprana de fallas en la industria 4.0 a través de los métodos basados en datos. Presenta una propuesta original para determinar los impulsores de la gestión de datos en los sistemas ciberfísicos industriales. Los escenarios para los atributos de Big Data industrial son un aporte que ayuda a establecer los requerimientos que debe cumplir una solución de analítica industrial, además de servir para comparar el diseño de una arquitectura para soluciones de analítica de Big Data industrial con la literatura revisada. La metodología de diseño arquitectónico de aplicaciones está basada en el enfoque de diseño basado en atributos (ADD, por sus siglas en inglés de Attribute-Driven Design) que consta de siete etapas y se enfoca específicamente en los atributos de calidad a través de la selección de estructuras arquitectónicas y su representación en vistas, también incluye análisis de arquitectura y documentación como parte integral del proceso de diseño.

Como aportación científica, se desarrolló una metodología basada en datos para apoyar la selección del mejor modelo no supervisado para la detección temprana de fallas. Para obtener el mejor modelo para predecir anomalías de un conjunto de métodos de aprendizaje. Cada algoritmo definido por el método de aprendizaje crea diferentes modelos del mismo tipo, pero con diferentes parámetros, dando lugar a conjuntos de modelos, dentro de estos conjuntos, el modelo con la menor varianza y sesgo se considera el más competente, luego este es comparado con otros algoritmos hasta obtener el modelo con el mejor desempeño global.

La arquitectura de referencia fue validada en diferentes escenarios de fallas en la industria, a través de aplicaciones reales tomadas de la literatura, para ilustrar diferentes casos de uso en contextos industriales que muestran cómo se puede aplicar la arquitectura. También se describió para cada escenario, una de las muchas

aplicaciones posibles de la propuesta de arquitectura a través del caso de uso del mundo real tomado de la literatura. Finalmente, se detallaron las interacciones entre los componentes de la arquitectura propuestos y el caso de uso revisado.

Además, se presenta una segunda validación de la arquitectura de referencia, a través de la implantación de una instancia de la misma para el escenario de fallas en la industria con el caso de uso que aplica el método desarrollado para encontrar el mejor modelo de detección temprana de fallas, con el fin de integrar las dos principales aportaciones de este trabajo de tesis.

Abstract

Today industry faces the challenge of integrating data from plant equipment and external data, sharing data services, achieving interconnectivity and interoperability between physical and cyberspace. Through industrial analytics systems, it is possible to identify useful insights, patterns, or models needed for sustainable innovation. Such systems are inherently complex, given the need to align the Internet of Things, Cloud computing, Big Data, and data-driven modeling methods with domain expertise. Given the above, the creation of technology platforms to promote optimization services faces the challenges of industrial cyber-physical systems that must consider a novel approach to the design of a reference architecture that integrates the convergence of technologies in industrial Big Data analytics and machine learning.

On the other hand, industrial plant systems are composed of numerous components and operate in a variety of conditions, in such circumstances data-driven fault detection and diagnosis face the challenges of not having sufficient historical data, in addition to the lack of speed in detecting new features. The unsupervised learning approach can help to benefit from these methods for early fault detection less dependent on prior knowledge and diagnostic experience by processing Big Data while ensuring that failures and their high repair costs are minimized.

This research presents a methodological approach for designing a reference architecture for industrial Big Data analytics that provides optimization services for early fault detection in Industry 4.0 through data-driven methods. It presents an original proposal to determine the drivers for data management in industrial cyber-physical systems. The scenarios for industrial Big Data attributes are a contribution

that helps to establish the requirements to fulfill an industrial analytics solution and to compare the design of an architecture for industrial Big Data analytics solutions with the literature reviewed. The application architecture design methodology is based on the Attribute Driven Design (ADD) approach which consists of seven stages. And focuses specifically on the quality attributes through the selection of architectural structures and their representation in views, also includes architecture analysis and documentation as an integral part of the design process.

As a scientific contribution, a data-driven methodology was developed for selecting the best-unsupervised model for early fault detection to obtain the best model for predicting anomalies from a set of learning methods. Each algorithm defined by the learning method creates different models of the same type but with distinct parameters giving models sets. Within this model set, the model with the lowest variance and bias is considered the most competent. Then this is compared with other algorithms until the model with the best overall performance is obtained.

The reference architecture was validated in different industrial failure scenarios through real applications taken from the literature to illustrate use cases in industrial contexts that show how the architecture can be applied. For each scenario, one application of the proposed architecture was also described through the real-world use case taken from the literature. Finally, the interactions between the proposed architecture components and the reviewed use case were detailed.

Agradecimientos

Deseo expresar mi aprecio y agradecimiento sincero y profundo a mi director Dr. Oscar Mario Rodríguez Elías por la oportunidad de aprender cómo ser investigador y por haber confiado en mí. Un agradecimiento especial a los integrantes de mi comité tutorial Dr. Héctor Guerra Crespo, Dr. Madain Pérez Patricio, Dr. José Antonio Hoyo Montaña, Dr. Jesús Horacio Pacheco Ramírez y al Dr. José Manuel Nieto Jalil Me, me gustaría agradecerles por alentar mi investigación y por permitirme crecer como científico investigador. Sus consejos tanto sobre la investigación como sobre mi carrera han sido invaluable, además por sus brillantes comentarios y sugerencias, muchas gracias.

Las palabras no pueden expresar lo agradecido que estoy con mi madre Alejandrina, mi padre Eduardo y mis dos hermanos Jesús y Gabriela por todos los sacrificios que han hecho en mi nombre.

Finalmente pero no menos importante, un enorme agradecimiento a Nancy quien con su infinito amor y paciencia me apoyo en esta travesía, junto con mis hijos Jesús y Dulce.

Muchas gracias a todos.

Eduardo A. Hinojosa-Palafox

Hermosillo, 2022

Índice General

Resumen	ii
Abstract	viii
Agradecimientos	x
Índice General	xi
Índice de Figuras	xvi
Índice de Tablas	xx
Acrónimos	xxii
Introducción	1
1.1 Introducción.....	2
1.2 Antecedentes.....	4
1.3 Planteamiento del problema	8
1.4 Motivación	12
1.5 Objetivos	13
1.5.1 Objetivo general	13
1.5.2 Objetivos específicos	13
1.6 Metodología de investigación	14
1.7 Alcances y limitaciones	15
1.8 Estructura de la tesis	16
Marco teórico	18

2.1	Gestión de datos en la Industria 4.0	19
2.1.1	Preguntas de investigación y objetivos.....	21
2.1.2	Estrategia de búsqueda	21
2.1.3	Criterios de selección.....	22
2.1.4	Evaluación de la calidad.....	23
2.1.5	Extracción de datos	25
2.1.6	Convergencia de tecnologías en la gestión de datos.....	25
2.1.7	El alcance de la gestión de datos en la Industria 4.0	27
2.1.8	Tendencias y desafíos de la Analítica de Big Data en la industria 4.0	29
2.2	Analítica de Big Data en la industria.....	30
2.2.1	Impulsores de la gestión de Big Data industrial	30
2.2.2	Restricciones en la gestión de datos industriales.....	32
2.2.3	Atributos de calidad del Big Data industrial	33
2.3	Modelo de analítica para detección de fallas en la industria.....	36
2.3.1	Modelo de aprendizaje para la detección de fallas	39
2.4	Almacenamiento y procesamiento distribuido de datos.....	47
2.4.1	Sistema de archivos distribuido Hadoop (HDFS)	47
2.4.2	Hadoop MapReduce	49
2.4.3	Herramientas de Apache Hadoop.....	50
2.4.4	Sandbox HDP	51
2.4.5	Cloudera DataFlow (CDF).....	54

2.5	Conclusiones	55
Metodología		57
3.1	Diseño de la arquitectura de referencia para analítica de Big Data industrial 58	
3.2	Modelo de aprendizaje no supervisado para la detección temprana de fallas en la industria.....	61
3.2.1	Selección de los detectores base con conjunto de datos validados	62
3.2.2	Modelo de aprendizaje no supervisado para la detección temprana de fallas	71
3.3	Conclusiones	76
Resultados		78
4.1	Arquitectura de referencia para analítica de Big Data en la industria.....	80
4.1.1	Capa de infraestructura	83
4.1.2	Capa de monitoreo	84
4.1.3	Capa de presentación	85
4.1.4	Despliegue de componentes.....	85
4.2	Validación de la arquitectura con escenarios para fallas en la industria ...	87
4.2.1	Evaluación de la arquitectura de referencia	88
4.2.2	Diagnóstico de fallas	91
4.2.3	Monitoreo en tiempo real	94
4.2.4	Pronóstico de condiciones de proceso inusuales	99

4.2.5	Utilidad de la arquitectura de referencia para la investigación y la práctica.....	102
4.3	Validación del modelo basado en datos.....	102
4.3.1	Descripción del caso práctico	103
4.3.2	Implementación del modelo basado en datos	106
4.3.3	Utilidad del modelo basado en datos para la detección de posibles fallas 111	
4.4	Validación de la arquitectura de referencia con Hadoop	113
4.4.1	Escenario de datos históricos para analítica de Big Data industrial .	113
4.4.2	Escenario de flujo de datos para analítica de Big Data industrial.....	119
4.4.3	Utilidad de la implementación de la arquitectura de referencia.....	129
	Discusión.....	132
5.1	Arquitecturas de referencia en contextos industriales	132
5.2	Diseño de arquitecturas de análisis de Big Data aplicadas en contextos iCPS 134	
5.3	Modelos no supervisados para detección temprana de fallas.....	136
5.3.1	Métodos no supervisados para obtener características de los datos y su aplicación en fallas industriales	137
5.3.2	Metodologías genéricas	138
5.3.3	Método único	139
5.3.4	Comparativa con el conjunto de datos del caso práctico	140
	Conclusiones, metas y futuras líneas de investigación.....	143

6.1	Conclusiones	143
6.2	Metas.....	149
6.2.1	Artículos de revista con factor de impacto, Q1	149
6.2.2	Congresos.....	149
6.2.3	Capítulos de libro	149
6.2.4	Proyectos	150
6.3	Futuras líneas de investigación.....	150
6.3.1	Diseño de arquitectura para analítica industrial	150
6.3.2	Modelos basados en datos para la detección temprana de fallas en la industria.....	151
	Referencias	152

Índice de Figuras

Figura 1. La perspectiva industrial del big data en CPS.....	6
Figura 2. Analítica para Big Data industrial (adaptado de [27]).....	7
Figura 3. Ciclo de vida del Big Data.....	8
Figura 4. Modelo de gestión de análisis de Big Data.....	10
Figura 5. Organización del Marco teórico.....	19
Figura 6. Proceso de selección de artículos académicos.....	20
Figura 7. Clasificación de tecnologías con enfoque de gestión de datos.....	25
Figura 8. Clasificación de artículos de Gestión de Datos en la Industria 4.0.....	27
Figura 9. Datos en los procesos de manufactura.....	30
Figura 10. Datos provenientes del ciclo de vida de manufactura.....	31
Figura 11. Definición de un escenario.....	34
Figura 12. Fuente de estímulo para los principales escenarios de atributos de calidad.	34
Figura 13. Definición de escenario.....	35
Figura 14. Modelo de aprendizaje.....	37
Figura 15. Escenarios para métodos de detección de anomalías.....	38
Figura 16. El espectro de datos normales a valores atípicos, adaptado de [60].	42
Figura 17. Tipos de anomalías.....	42

Figura 18. HDFS (fuente: [88]).....	48
Figura 19. YARN (fuente [88]).....	49
Figura 20. Instalación de HDP 3.0.1 en contenedores Docker.	50
Figura 21. Sandbox Hortonworks (fuente: https://www.cloudera.com/tutorials/sandbox-architecture.html).....	51
Figura 22. Organización de la metodología empleada en esta tesis.	58
Figura 23. El proceso de diseño basado en atributos.....	60
Figura 24. Metodología para la selección de los detectores base.	63
Figura 25. Curvas ROC, adaptado de [106].	67
Figura 26. ROC-AUC para el método OSCV con hiperparámetros, aplicado a diferentes conjuntos de datos.	70
Figura 27. Organización de la presentación de los resultados.....	79
Figura 28. Arquitectura Lambda (Fuente: http://lambda-architecture.net/).	81
Figura 29. Arquitectura de referencia para la gestión de datos para iCPS.	82
Figura 30. Despliegue de tecnologías.	86
Figura 31. Instancia arquitectónica de Big Data de analítica industrial para el caso de uso de diagnóstico de fallas.....	89
Figura 32. Iconos de notación de los diagramas de vista de proceso.	90
Figura 33. Instancia del proceso de diagnóstico de fallas.....	92
Figura 34. Instancia del proceso de diagnóstico de fallas (caso de uso reportado en [116]).....	93

Figura 35. Instancia del proceso de monitoreo en tiempo real para la detección de fallas.	95
Figura 36. Instancia del proceso de monitoreo en tiempo real para la detección de falla (Vista del caso de uso reportado en [119]).	97
Figura 37. Instancia para el pronóstico de condiciones inusuales.	99
Figura 38. Instancia de procesos inusuales (Vista para el caso de uso reportado en [121]).	101
Figura 39. Descripción de los conjuntos de datos, adaptado de [124].	104
Figura 40. Instancia para el caso de uso reportado en [18].	105
Figura 41. Sistema de bucle de retroalimentación (fuente [122]).	105
Figura 42. Selección del modelo no supervisado.	¡Error! Marcador no definido.
Figura 43. Serie temporal para las variables S1, S2 y S4, E1 y E2 en la planta 1. ...	108
Figura 44. Preprocesamiento de los datos de los archivos tipo “a”	109
Figura 45. Gráfico de flujo general (100 puntos de posibles fallas en sensores). ...	111
Figura 46. Escenario de modelos basados en datos.	112
Figura 47. Instancia para analítica de Big Data industrial con datos históricos.	113
Figura 48. Sistema de Archivos de Hadoop	114
Figura 49. Data Analytics Studio (DAS).....	115
Figura 50. Vista para la creación de consultas SQL en Hive para HDFS.	115
Figura 51. Consulta SQL de los datos de sensores en Hive.	116
Figura 52. Análisis interactivo de datos con Zeppelin.....	116
Figura 53. Contexto de Hive para soportar SQL en Spark.	117

Figura 54. Creación de un conjunto de datos distribuidos.	117
Figura 55. Poblar el esquema con datos de los promedios de los sensores.	118
Figura 56. Vista temporal <i>sensorPromedio</i>	118
Figura 57. Consulta SQL a <i>sensorPromedio</i>	118
Figura 58. Consulta SQL interactiva a <i>sensor_prom</i>	119
Figura 59. Gráfico interactivo para la consulta SQL a <i>sensor_prom</i>	119
Figura 60. Instancia para analítica de Big Data industrial para flujo de datos.	119
Figura 61. Estructura del flujo de datos.	120
Figura 62. Creación de temas en Kafka para este caso.	120
Figura 63. Transmisión de los datos de componentes CSV con Kafka Connect. ...	121
Figura 64. Flujo de datos y procesamiento por nifi usando procesamiento basado en registros.	122
Figura 65. Ingesta de los eventos de los sensores sin procesar.	123
Figura 66. Procedencia de los datos de Nifi.	124
Figura 67. Evento de procedencia.	124
Figura 68. Topología de SAM.	125
Figura 69. Componente Kafka.	125
Figura 70. Componente de agregación.	126
Figura 71. Reglas para los sensores de los componentes.	126
Figura 72. Enviar el flujo de datos a Druid.	127
Figura 73. Porciones de visualización.	128
Figura 74. Gráfico de línea de serie de tiempo.	129

Índice de Tablas

Tabla 1. Cadena de búsqueda por fuente consultada	22
Tabla 2. Criterios de inclusión y exclusión.....	23
Tabla 3. Formulario de criterios de calidad.....	24
Tabla 4. Formulario de extracción de datos.	24
Tabla 5. Propiedades y restricciones de datos y analítica.	33
Tabla 6. Atributos de calidad para la gestión de datos para Big Data industrial. ..	36
Tabla 7. Métodos de aprendizaje utilizados en el diagnóstico de fallas (Adaptada de [53]).	39
Tabla 8. Algoritmos más comunes para la detección de anomalías no supervisadas, adaptado de [67].....	46
Tabla 9. Conjuntos de datos disponibles para probar el desempeño de métodos no supervisados.	64
Tabla 10. Métodos para la detección de anomalías	65
Tabla 11. Parámetros usados para la selección de algoritmos base, adaptado de [108].	65
Tabla 12. Matriz de confusión, adaptado de [106]	68
Tabla 13. Características de hardware.	68

Tabla 14. Rendimiento ROC-AUC.	69
Tabla 15. Parámetros de los modelos.	76
Tabla 16. Requerimientos de los escenarios de diagnóstico de fallas.....	88
Tabla 17. Cuenta de niveles únicos para variables categóricas.	107
Tabla 18. Variables indicadoras.	108
Tabla 19. Modelo y parámetros seleccionados en las primeras cinco plantas.	109
Tabla 20. Posibles fallas en componentes y zonas.....	110
Tabla 21. Literatura revisada para aprendizaje no supervisado aplicado a fallas en la industria.	136
Tabla 22. Literatura relacionada con el caso de uso reportado en [132].	140

Acrónimos

ADD	<i>Attribute-driven design</i>	<i>Diseño impulsado por atributos</i>
ALMA	<i>Architecture-level modifiability analysis</i>	<i>Análisis de modificabilidad a nivel de arquitectura</i>
ANN	<i>Artificial neural network</i>	<i>Red neuronal artificial</i>
ATAM	<i>Architecture tradeoff analysis method</i>	<i>Método de análisis de compensación de arquitectura</i>
BN	<i>Bayesian network</i>	<i>Red bayesiana</i>
CAD	<i>Computer-aided design</i>	<i>Diseño asistido por ordenador</i>
CAE	<i>Computer aided engineering</i>	<i>Ingeniería asistida por ordenador</i>
CAM	<i>Computer-aided manufacturing</i>	<i>Fabricación asistida por ordenador</i>
CBAM	<i>Cost benefit analysis method</i>	<i>Método de análisis de beneficios de costo</i>
CRM	<i>Customer Relationship Management</i>	<i>Gestión de relaciones con los clientes</i>
ERP	<i>Enterprise resource planning</i>	<i>Planificación de recursos empresariales</i>
ETL	<i>Extract, transform and load</i>	<i>Extraer-Transformar-Cargar</i>
HDFS	<i>Hadoop distributed file system</i>	<i>Sistema de archivos distribuido de Hadoop</i>
HMM	<i>Hidden markov model</i>	<i>Modelo oculto de markov</i>

iCPS	<i>Industrial cyber-physical system</i>	<i>Sistema Ciberfísico Industrial</i>
IIoT	<i>Industrial internet of things</i>	<i>Internet industrial de las cosas</i>
KPI	<i>Key Performance Indicators</i>	<i>Indicador clave de rendimiento</i>
MES	<i>Manufacturing Execution System</i>	<i>Sistema de ejecución de fabricación</i>
MIS	<i>Manufacturing information system</i>	<i>Sistemas de información de fabricación</i>
MQTT	<i>MQ Telemetry Transport</i>	<i>Transporte de telemetría de cola de mensajes</i>
MRO	<i>Maintenance, repair, and overhaul</i>	<i>Mantenimiento, reparación y revisión</i>
OLAP	<i>Online analytical processing</i>	<i>Procesamiento analítico en línea</i>
PaaS	<i>Platform as a service</i>	<i>Plataforma como servicio</i>
RFID	<i>Radio Frequency Identification</i>	<i>Identificación por radiofrecuencia</i>
SAAM	<i>Software architecture analysis method</i>	<i>Método de análisis de arquitectura de software</i>
SCM	<i>Supply Chain Software</i>	<i>Gestión de la cadena de suministro</i>
SVM	<i>Support vector machine</i>	<i>Máquinas de vectores de soporte</i>

Capítulo 1.

Introducción

En este capítulo se realiza un breve acercamiento a la temática tratada en esta tesis. Adicionalmente se presentan los antecedentes de investigación, que incluyen las principales tecnologías habilitadoras relacionadas a la gestión de datos en la industria 4.0: el internet industrial de las cosas, el cómputo en la nube en la industria, el aprendizaje automático y la analítica de Big Data industrial. Adicionalmente, el capítulo describe el problema a investigar, las motivaciones del desarrollo de aplicaciones de analítica de Big Data industrial y los objetivos que guiaron el desarrollo de la investigación. Finalmente, el capítulo detalla las principales contribuciones realizadas en la investigación.

1.1 Introducción

Las nuevas tecnologías digitales en industrias como la agroalimentaria, la logística y la fabricación están permitiendo que los humanos, las máquinas, los productos y los recursos intercambien información entre ellos [1]. La industria está migrando de un enfoque tradicional a uno en el que una máquina no solo se limita a producir, sino que debe hacerlo de una manera inteligente y energéticamente eficiente, también debe ser capaz de proporcionar información sobre el proceso a varios rangos de la jerarquía de la organización [2].

Este nuevo enfoque conocido como Industria 4.0 marca un hito importante en el desarrollo industrial y expresa la idea de que se está en el comienzo de una cuarta revolución industrial [3]. Su base es que la conexión de máquinas, sistemas y activos en las organizaciones pueden crear redes inteligentes a lo largo de la cadena de valor para controlar los procesos de producción [4]. En este nuevo escenario, el foco no está solo en las nuevas tecnologías sino también en cómo se combinan desde la perspectiva de los datos.

El impacto que genera el explosivo desarrollo del internet de las cosas en la industria es ampliamente conocido [5], esto abre la posibilidad de que dispositivos como sensores y actuadores puedan conectarse e interactuar entre sí y con otros sistemas para la construcción de sistemas ciberfísicos industriales (iCPS, por las siglas en inglés de Industrial Cyber-Physical Systems) [6]. Eso permite tener más y mejor información sobre lo que sucede en un proceso industrial en tiempo real y así poder tomar acciones de manera más eficiente [7]. Los sistemas ciberfísicos pueden verse como la evolución de los sistemas de información y comunicación, derivados del crecimiento exponencial de la capacidad de cómputo, transmisión y almacenamiento de las computadoras, que posibilita la integración de estas tecnologías con los procesos físicos.

Esta oportunidad de interconexión trae consigo la posibilidad de que la generación constante de grandes volúmenes de datos sea utilizada en diversas aplicaciones de la industria, aunque también trae desafíos, como la necesidad de un modelo novedoso que tenga en cuenta cambios en la convergencia de tecnologías en sistemas ciberfísicos industriales [8], además de considerar la importancia de un enfoque de diseño centrado en el modelo de datos que permita crear una arquitectura de sistema que sirva de referencia para el desarrollo de la analítica industrial de Big Data [9].

Los iCPS integran los sistemas de información industrial con la tecnología operativa para, entre otras cosas, optimizar los procesos de producción a través del análisis de datos [10]. En ese sentido, es posible considerar el desarrollo de los modelos de aprendizaje aplicado a problemas en la industria, a la presencia de teorías y métodos relacionados con otros campos, por ejemplo, la estadística, el aprendizaje automático, el aprendizaje profundo, entre otros [11]; su aplicabilidad a la necesidad de alinear la tecnología, el modelado, el pronóstico, la optimización y la experiencia en el campo.

El diagnóstico y detección de fallas basado en grandes volúmenes de datos es un enfoque común que se puede aplicar en diferentes contextos industriales [12]. Este enfoque es adecuado cuando se tienen datos históricos con información de fallas, además de los recursos necesarios para hacer un análisis profundo de los datos y probar con diferentes estrategias para seleccionar la combinación de modelos más adecuada.

Sin embargo, existen en la industria dificultades para contar con estos recursos, por lo que es deseable hacer una gestión inteligente de fallas menos dependiente de conocimientos previos y experiencia diagnóstica al procesar macrodatos. En tales casos, el aprendizaje no supervisado sería una mejor opción para la construcción de

modelos ya que permite el diagnóstico inteligente de fallas menos dependiente de conocimientos previos y experiencia en el diagnóstico del proceso estudiado.

Este capítulo presenta una introducción a la tesis, resaltando la importancia de la convergencia tecnológica de la analítica industrial generado por los cambios disruptivos en la industria 4.0 en los diferentes dominios de aplicación. También se presentan los principales antecedentes y planteamiento del problema que dieron origen a la presente investigación, además se detallan los objetivos que guiaron su desarrollo. Finalmente se detalla la estructura de este documento.

1.2 Antecedentes

Para abrir este campo de investigación interdisciplinario, es importante obtener conocimiento de las tecnologías requeridas para habilitar la analítica industrial. Además, es importante reconocer los problemas y limitaciones actuales con respecto a la gestión de datos en la industria ya que a menudo conducen a temas inesperados en relación con las técnicas, herramientas, metodologías y recursos requeridos para su implementación.

El concepto de Industria 4.0 surge en Alemania como parte de los esfuerzos del sector manufacturero para ponerse a la vanguardia y ser competitivo en un entorno globalizado altamente competitivo. El año específico es incierto. Los autores en [13] consideran 2011 como el año del nacimiento del concepto de Industria 4.0 como una propuesta para el desarrollo de una nueva conceptualización de la política económica alemana basada en estrategias de alta tecnología. En [14] se menciona que la Industria 4.0 forma parte de un plan industrial llamado "La nueva estrategia de alta tecnología: innovaciones para Alemania" lanzado en 2006.

La Industria 4.0 se refiere a un cambio de paradigma que evidencia la naturaleza interconectada de la estandarización y tecnologías de software dentro de la industria, y hace un cambio importante llamado la cuarta revolución industrial. La

Industria 4.0 parece estar guiada por los avances en la informática, en la integración de las redes de datos y la inteligencia artificial en los procesos de producción. En este sentido, el software se convierte en un componente esencial en el desarrollo de productos, servicios, procesos de producción, administrativos, logísticos e incluso en el desarrollo de nuevos modelos de negocio, compra y venta, marketing, entre otras múltiples áreas [15] .

El Internet de las Cosas (IoT, por las siglas en inglés de Internet of Things) está siendo ampliamente incorporado en la industria, y su impacto la está transformando [16]. El Internet Industrial de las Cosas (IIoT, por las siglas en inglés de Industrial Internet of Things) consiste en crear redes de objetos físicos, entornos, vehículos y máquinas a través de dispositivos electrónicos integrados que permiten la recopilación e intercambio de datos [17]. El IIoT abre la posibilidad a los iCPS de converger con la tecnología de la información y relacionado a redes, conectividad, datos, ecosistemas y sistemas de información con tecnología operativa hablando de equipos físicos de la planta, maquinaria, sistemas de monitoreo y control. Es decir, la construcción de iCPS permite la integración de la información con los procesos industriales [18]. En este contexto, los iCPS conectan máquinas, sistemas, activos y organizaciones para crear redes inteligentes a lo largo de la cadena de valor, entre otras cosas, para optimizar los procesos de producción [19].

Toda esta interconexión de objetos que generan y procesan datos e información también requiere nuevos enfoques para gestionar este crecimiento exponencial de datos e información como estrategias que permitan aprovechar esta acumulación exponencial de datos [20]. En este nuevo escenario, como se muestra en la **Figura 1**, el enfoque no se centra sólo en las nuevas tecnologías, sino también en cómo se combinan, teniendo en cuenta los tres niveles de integración desde la perspectiva de los datos (datos locales, datos globales y datos en la nube) [21].

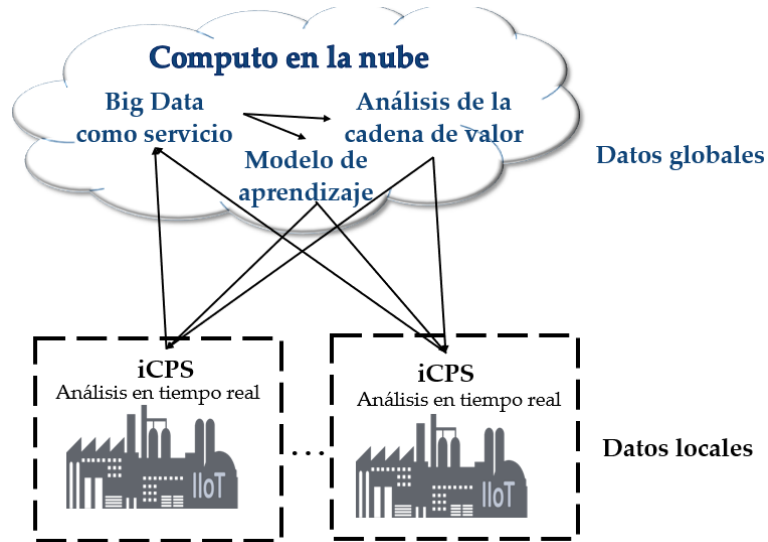


Figura 1. La perspectiva industrial del big data en CPS.

La computación en la nube de forma general puede entenderse como un enfoque en el uso de recursos informáticos (hardware y/o software), a los que se accede a voluntad mediante la contratación de servicios a terceros [22]. El enfoque principal de la computación en la nube industrial es la integración y las soluciones verticales en lugar de las horizontales, que es el foco del cómputo en la nube general, esto significa que las soluciones de nube industrial se centran en crear más valor dentro de los límites de la industria en lugar de ampliar sus alcances. Dado que los dispositivos están conectados a una red amplia, requieren un entorno que permita reunirse e interactuar entre sí ofreciendo y requiriendo servicios. El cómputo en la nube facilita el almacenamiento, procesamiento y gestión de Big Data en iCPS [23]. La integración con IIoT puede llevar el procesamiento de flujos de datos de detección al siguiente nivel para proporcionar servicios de detección omnipresente más allá de las capacidades de las cosas individuales [24].

Los modelos de aprendizaje automático se utilizan en Big Data para identificar patrones complejos para extraer información de grandes cantidades de datos de procesos industriales bajo observación [25]. El aprendizaje automático para Big Data es el servicio típico que las empresas tienden a externalizar en la nube, debido a su

naturaleza de uso intensivo de datos y la complejidad de los algoritmos [26]. Lo que es atractivo en la industria es confiar en la experiencia externa y la infraestructura para calcular los resultados analíticos y los modelos que los analistas de datos requieren para entender los procesos en observación.

Debido a que se requiere un análisis de datos sofisticado, es necesario un enfoque de diseño de modelos de datos para permitir el desarrollo de arquitecturas de sistemas de análisis para la industria [27].

Analítica es el procesamiento de Big Data para identificar ideas, patrones o modelos útiles; es la clave de la innovación sostenible en la Industria 4.0. En la **Figura 2**, se muestran los diferentes tipos de analítica de Big Data que es posible aplicar a la industria [28].

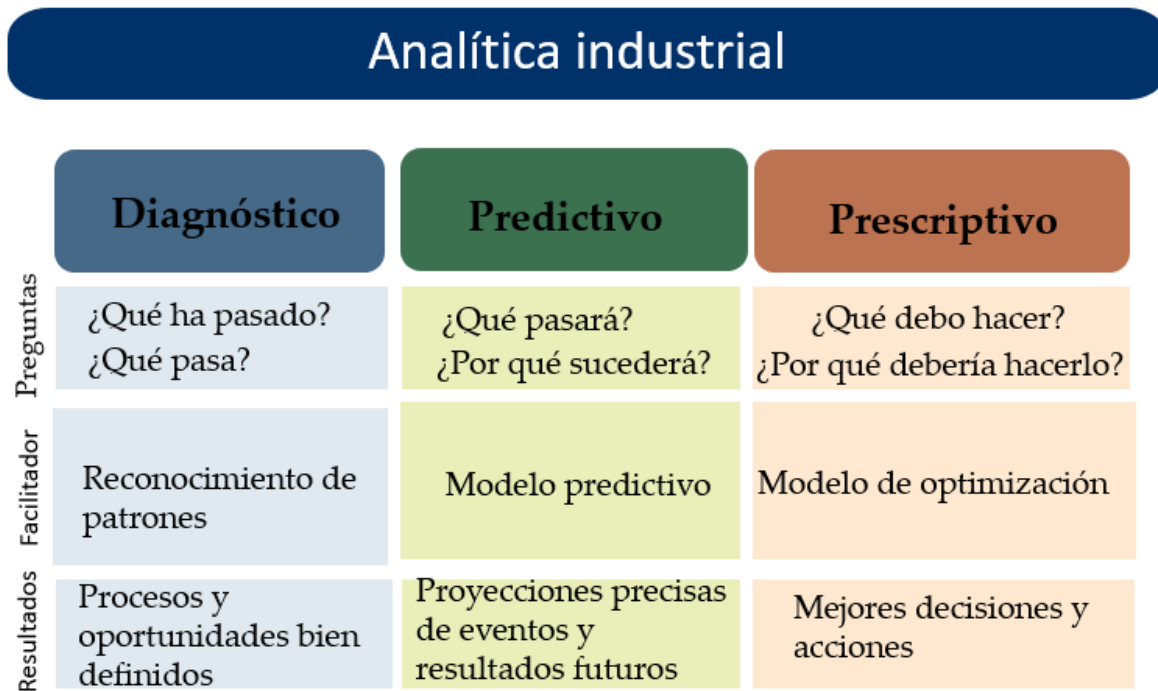


Figura 2. Analítica para Big Data industrial (adaptado de [29]).

Analítica de diagnóstico. Extrae información de procesos de datos industriales históricos y transaccionales para reconocer patrones en los datos mediante el empleo de reconocimiento de patrones.

Analítica predictiva. Utiliza la minería de Big Data mediante la aplicación de técnicas modernas de enfoques estadísticos, visualización de datos y enfoque de reconocimiento de patrones para reconocer las amenazas y predicciones en la industria.

Analítica prescriptiva. Se utiliza para generar mejores soluciones mediante modelos de optimización basados en aprendizaje automático que permiten la integración y el procesamiento de datos, para el monitoreo de datos y toma de decisiones.

1.3 Planteamiento del problema

En la Industria 4.0, los datos son generados por múltiples fuentes en diferentes contextos, como equipos de instalaciones, equipos de proceso, sistemas de fabricación y datos de entorno. Toda esta variedad de datos que llega a alta velocidad y grandes volúmenes se llama Big Data industrial (ver **Figura 3**).

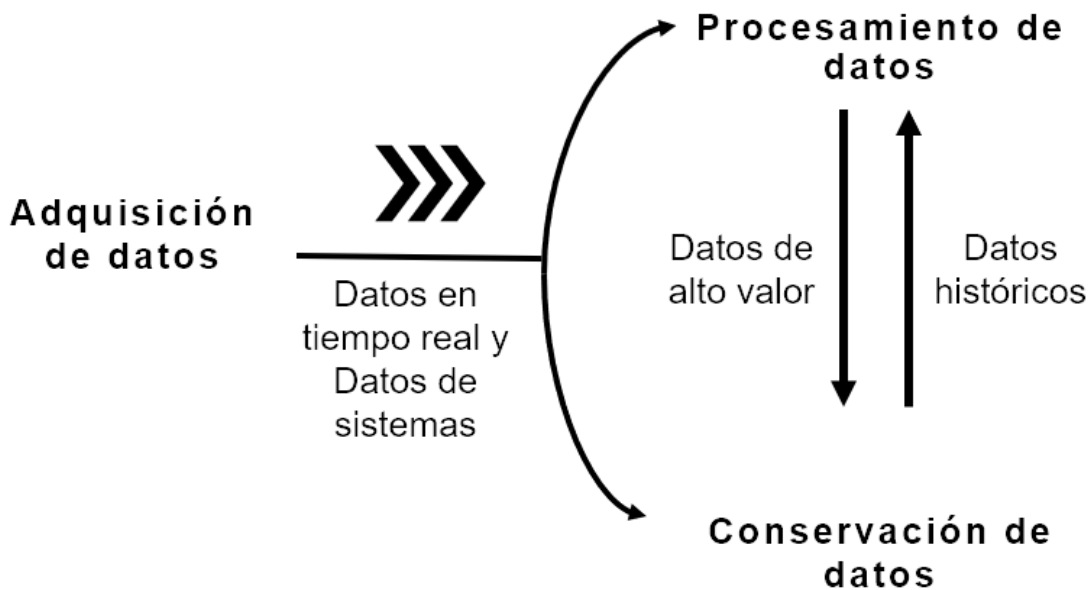


Figura 3. Ciclo de vida del Big Data.

Sin embargo, los datos sin procesar son inútiles, por lo que, para obtener la información de los datos, en primer lugar, es necesario limpiar, unificar, consolidar y normalizar los datos antes de ser procesados debido al ruido, el multiformato, las diferencias de escala, fuentes heterogéneas, entre otros aspectos a considerar en los datos. A continuación, los datos con alto valor se conservan como datos históricos y procesan para el intercambio y el uso compartido en todos los niveles. Por lo general, a través de los servicios en la nube como los servicios de predicción a través de la minería de datos y el aprendizaje automático [30].

Diferentes enfoques se pueden encontrar en el diseño arquitectónico para abordar tecnologías específicas o áreas particulares, pero uno que integra una solución cuyo eje central es la gestión de Big Data a lo largo de su ciclo de vida en el contexto iCPS todavía está en desarrollo [19, 28, 29]. Las soluciones de Big Data para iCPS necesitan integrar tecnologías en un ecosistema consistente en un entorno industrial, pero tales soluciones son complejas. Una guía de arquitectura de referencia para el análisis de Big Data podría facilitar el desarrollo, la implementación y el funcionamiento de las soluciones iCPS de Big Data en la industria [33]. Tomando en cuenta lo anterior, la presente tesis se encuadra en una investigación enfocada en proporcionar conocimientos que faciliten la adopción exitosa de las metodologías, técnicas y herramientas de la analítica de Big Data en la industria.

Se requiere una arquitectura que considere el nuevo paradigma que integra las tecnologías de Big Data en iCPS que permite la creación de arquitecturas a partir de modelos de datos que admitan el análisis de Big Data, como se muestra en la **Figura 4**, para ayudar a abordar los desafíos industriales, considerando que la tecnología de Big Data cambia rápidamente.

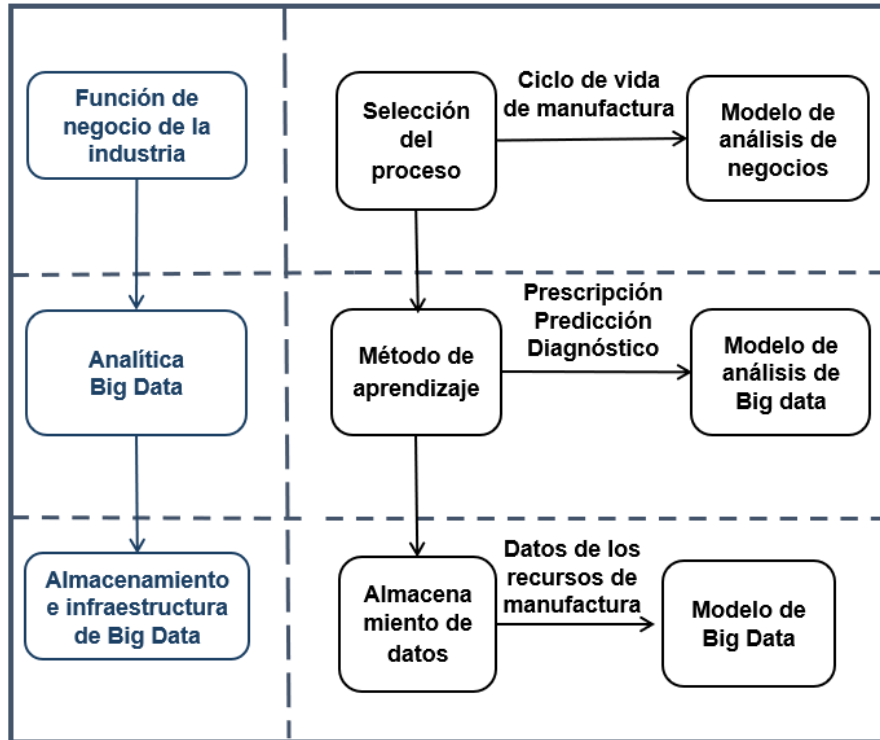


Figura 4. Modelo de gestión de análisis de Big Data.

El análisis de este modelo de gestión de analítica de Big Data presenta ciertas interrogantes a superar para generar valor. Estas interrogantes son las principales motivaciones para el desarrollo de esta tesis y se muestran a continuación:

La primera interrogante es ¿cómo abordar los desafíos de los procesos industriales para lograr un modelo de Big Data? relacionada a esta, la segunda interrogante es ¿cómo se lleva la fase de diseño en los sistemas de Big Data industriales? El diseño arquitectónico del sistema de análisis industrial es mucho más crítico que para los sistemas de datos heredados, así que la tercera interrogante se centra en ¿cómo ampliar la arquitectura de diseño tradicional para el diseño del sistema de análisis en iCPS? Una cuarta pregunta considera las anteriores: ¿cómo se pueden integrar el modelo de Big Data y el diseño de arquitectura para el marco del sistema de análisis en el contexto de iCPS?

Además, el diagnóstico y detección de fallas basado en grandes volúmenes de datos es un enfoque común que se puede aplicar en diferentes contextos industriales

[12]. Este enfoque es adecuado cuando se tiene datos históricos con información de fallas, además de los recursos necesarios para hacer un análisis profundo de los datos y probar con diferentes estrategias para seleccionar la combinación de modelos más adecuada.

Sin embargo, existen en la industria dificultades para contar con estos recursos, por lo que es deseable hacer una gestión inteligente de fallas menos dependiente de conocimientos previos y experiencia diagnóstica al procesar macrodatos. En tales casos, el aprendizaje no supervisado sería una mejor opción para la construcción de modelos ya que permite el diagnóstico inteligente de fallas menos dependiente de conocimientos previos y experiencia en el diagnóstico del proceso estudiado [34]. Es decir, los desafíos del aprendizaje supervisado, como la necesidad de datos históricos y la incapacidad de clasificar nuevas fallas con precisión, pueden ser superados con una nueva metodología que utilice aprendizaje no supervisado para una implementación rápida de la actividad del monitoreo de la actividad industrial que incluya la predicción de fallas y la detección de clases de fallas para fallas conocidas y desconocidas [35].

Con base en lo anterior, esta tesis doctoral demuestra un modelo de gestión de datos que provee servicios de analítica en la industria a través de:

- Una arquitectura de referencia flexible y adaptable de analítica de Big Data industrial basada en la gestión de modelos de datos para extraer el conocimiento de los datos generados por los sistemas ciberfísicos industriales.
- Una metodología para la detección de posibles fallas basada en un algoritmo para la selección del modelo para predecir posibles anomalías en los datos.

1.4 Motivación

Entre los diversos tipos de servicios que se han derivado del cómputo en la nube, se encuentra la optimización como servicio [36], un modelo de negocios que empieza a ser impulsado en la región, en particular en la industria minera, y que tiene como propósito facilitar herramientas avanzadas de análisis de datos, así como conocimiento experto, para proveer soluciones a las empresas que les permitan optimizar sus procesos. Esto se logra mediante la integración del internet de las cosas en los dispositivos que controlan el proceso (sensores, actuadores, y controladores), con el fin de recabar datos de forma constante, que deben ser almacenados y posteriormente procesados mediante algoritmos de aprendizaje máquina, que por la cantidad de datos que se recaban, requieren de las técnicas e infraestructura para gestión y procesamiento de Big Data. La implementación de estos nuevos modelos de servicios en la industria trae retos como los siguientes:

La necesidad de una infraestructura tecnológica para la captura, almacenamiento, procesamiento y aprovechamiento de los datos, que puede resultar costosa para algunos sectores [18], como es el caso de la industria regional, debido a que se requiere infraestructura física especial, así como pago de licencias de software costosas. En el caso de las licencias de software existe alternativas como el aprovechamiento del software libre [37], pero que requiere de un análisis profundo para identificar el tipo de software aplicable, así como lograr su integración, y desarrollo de módulos con funcionalidades que el software libre aun no soporte para situaciones particulares. Por su parte, en el caso de la infraestructura física, existe la posibilidad de la contratación de infraestructura como servicios, uno de los esquemas de servicios del cómputo en la nube.

No obstante, este enfoque involucra aspectos importantes a considerar, los cuales se describen a continuación:

- Los proyectos de Big Data industrial, en general, adolecen de un largo tiempo de desarrollo y ejecución de proyectos, lo que hace que muchos proyectos interesantes no sean económicamente atractivos.
- La gestión de datos en el contexto de la industria 4.0 es un tema complejo que requiere una propuesta que considere la gestión de datos como un componente nuclear en el diseño de una arquitectura de referencia para la analítica industrial.
- Esta propuesta debe considerar la convergencia de IIoT, con el cómputo en la nube en la industria y el modelado basado en datos para el diseño de una arquitectura de referencia que sea la base para el desarrollo de soluciones de analítica industrial en el contexto de los iCPS.

1.5 Objetivos

Con base en lo anterior, los objetivos de esta investigación son:

1.5.1 *Objetivo general*

Diseñar un enfoque metodológico que permita crear plataformas tecnológicas para proveer servicios de optimización a la Industria 4.0 a través del análisis de Big Data.

1.5.2 *Objetivos específicos*

1. Identificar el problema de la gestión de datos para analítica de Big Data y los retos que afrontan actualmente los sistemas ciberfísicos industriales, así como su relevancia en la industria con el fin de aportar mejoras en el estado del arte.
2. Caracterizar las metodologías, estrategias y las tecnologías libres relacionadas con la gestión de datos para analítica de Big Data en la industria 4.0 por medio de las referencias a la literatura de soporte y el estudio de

conceptos relacionados con la temática para sentar las bases en el desarrollo de la solución al problema planteado.

3. Caracterizar la gestión de datos para analítica de Big Data en la Industria 4.0.
4. Diseñar una arquitectura de referencia para el desarrollo de plataformas tecnológicas con apoyo del modelo de gestión de datos desarrollado, siguiendo los requerimientos de analítica para iCPS. La modularidad de la arquitectura permitirá la adaptación a los requerimientos funcionales y no funcionales de los diferentes entornos de iCPS.
5. Validar la arquitectura propuesta en diferentes casos de uso en contextos industriales que muestran cómo se puede aplicar la arquitectura, a través de instancias de la arquitectura con el fin de proveer información útil para el seguimiento de fallas industriales.
6. Caracterizar los métodos de aprendizaje para la detección de anomalías.
7. Diseñar un modelo basado en datos para la detección temprana de fallas aplicado a la industria que soporte el procesamiento de datos por lotes y en flujo.
8. Validar el modelo basado en datos a través de un escenario para la detección de fallas en la industria.
9. Desarrollar un prototipo de plataforma tecnológica con base en la arquitectura de referencia propuesto y el modelo basado en datos desarrollado.

1.6 Metodología de investigación

Esta tesis aborda la gestión de datos para analítica industrial más allá del estado del arte en el contexto específico de los sistemas ciberfísicos industriales. La metodología de investigación usada en la elaboración de este tesis doctoral consiste en cuatro fases, como sigue:

En una primera fase, se ha revisado la literatura existente para conocer los enfoques utilizados para la gestión de datos para analítica industrial, concluyéndose que no existe un enfoque de analítica industrial que satisfaga las necesidades específicas de la Industria 4.0, estableciendo así la evaluación del problema.

En línea con los requerimientos de esta investigación, en una segunda fase se ha propuesto una arquitectura que pueda ser utilizada como referencia para el desarrollo de aplicaciones de analítica de Big Data industrial que se puede adaptar a diferentes escenarios de aplicación de acuerdo con los requerimientos de la gestión de datos.

En una tercera fase, se ha propuesto una metodología para apoyar la selección del modelo basado en datos para la detección temprana de posibles fallas.

En una cuarta y última fase se ha validado: (1) La arquitectura de referencia propuesta en términos de escenarios para fallas industriales incluyendo los escenarios de monitoreo en tiempo real y el pronóstico de condiciones inusuales (2) La selección del modelo basado en datos para la detección temprana de posibles fallas ha sido validada con datos de mediciones de sensores y señales de referencia de control para diferentes componentes de control en una planta industrial (3) Finalmente la integración de la arquitectura de referencia y el modelo de gestión de datos ha sido validado con la implementación de tecnologías basadas en código abierto que soportan una plataforma tecnológica aplicable a cada escenario de gestión de Big Data en la industria.

1.7 Alcances y limitaciones

Debido a la situación de la pandemia, no se pudo concretar la implementación del escenario de aplicación como se había planificado, sin embargo, se logró identificar la existencia de escenarios concretos dentro de la industria local en el que se pueden aplicar las propuestas derivadas del proyecto. Así mismo, se dejó el

entorno preparado para colaborar con una empresa local en cuanto la situación derivada de la pandemia lo permita.

Se probó el algoritmo propuesto en un escenario real para el monitoreo y control de un proceso industrial, no obstante, en este último caso, los datos se obtuvieron de una base de datos, en vez de obtenerlos directamente de un proceso industrial en operación, lo cual no fue posible lograr debido a la pandemia, la empresa con la que se estaba trabajando debió cerrar operaciones por un período prolongado, y posterior a eso, debió cambiar prioridades por las afectaciones económicas y de operación que la pandemia provocó.

1.8 Estructura de la tesis

El manuscrito de la tesis está estructurado en seis capítulos de la siguiente forma:

El Capítulo 2 presenta la perspectiva teórica que da fundamento a la investigación original, con el fin de entender los enfoques utilizados para el modelo de gestión de Big Data para analítica industrial.

El Capítulo 3 describe la metodología utilizada en el diseño de la arquitectura para analítica de Big Data industrial y la metodología empleada para seleccionar el modelo de aprendizaje no supervisado para detección temprana de fallas.

El Capítulo 4 presenta los elementos que constituyen la arquitectura de referencia para analítica de Big Data Industrial y su validación a través de tres escenarios de fallas industriales. También, se valida el método de aprendizaje basado en datos para la selección del mejor modelo de datos para la detección temprana de fallas, con un caso práctico que incluye un conjunto de datos de mediciones de sensores y señales de referencia de control para cada uno de varios componentes de control de una planta industrial y mediciones de energía eléctrica de diferentes zonas de la planta.

En el Capítulo 5 se presenta una discusión de los resultados obtenidos en relación con la literatura relacionada.

Finalmente, en el Capítulo 6 se presentan las conclusiones derivadas de la investigación realizada, las metas alcanzadas y las futuras líneas de trabajo de investigación que se derivan del presente proyecto.

Capítulo 2.

Marco teórico

Esta sección presenta la perspectiva teórica que da fundamento a la investigación original, con el fin de entender los enfoques utilizados para el modelo de gestión de Big Data para analítica industrial. Como una aportación original se elaboró y describió el ciclo de vida de la gestión datos en la industria, su estado del arte y su convergencia en los iCPS. También, como una aportación se describen los impulsores de la gestión de datos en la industria, las restricciones y atributos de la analítica de Big Data industrial. Además, se describen los conceptos relacionados al aprendizaje automático para fallas en la industria, su clasificación y la descripción de los algoritmos más comunes para la detección de anomalías no supervisadas. Finalmente, en este capítulo se incluye la descripción del ecosistema que permite el almacenamiento y procesamiento distribuido de datos y el desarrollo de plataformas tecnológicas para el análisis de Big Data industrial.

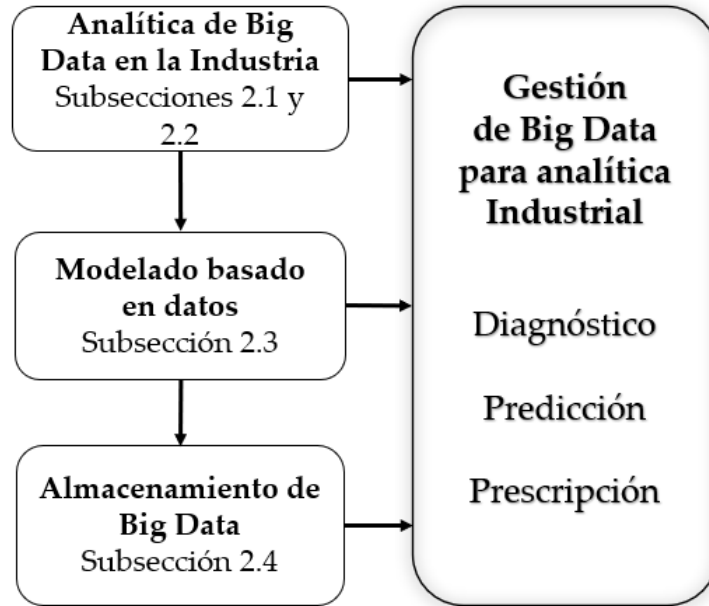


Figura 5. Organización del Marco teórico.

En la **Figura 5**, se presenta la organización de este capítulo. Los elementos de la industria que corresponden a la analítica de Big Data son explicados en las subsecciones 2.1 y 2.2. En la subsección 2.1 se identifican los retos y oportunidades que afrontan actualmente la analítica de Big Data para sistemas ciberfísicos industriales, así como su relevancia. En la subsección 2.2 se caracteriza el ciclo de vida de los datos de manufactura.

El modelado basado en datos es presentado en la subsección 2.3. Aquí se caracterizan los métodos de aprendizaje para la detección de posibles fallas en iCPS.

En la sección 2.4 se presenta el almacenamiento de Big Data el cuál trata las necesidades de recopilar grandes volúmenes de datos del ciclo de vida de manufactura y las relaciones que guardan con el almacenamiento y el procesamiento distribuido de datos.

2.1 Gestión de datos en la Industria 4.0

Esta subsección presenta una revisión sistemática de literatura (SLR, por las siglas en inglés de Systematic Literature Review) para mostrar las intersecciones en

las tendencias actuales que se utilizaron para conocer las posibilidades en el desarrollo potencial y tendencias futuras.

El protocolo de revisión se elaboró con base en la guía para una revisión sistemática de la literatura como se propone en [38], y describe las preguntas de investigación (y objetivos), criterios de inclusión/exclusión, bases de datos y motores de búsqueda, términos de búsqueda, extracción de contenidos y datos relevantes, evaluación de la calidad de estos resultados, y recopilación de los resultados más destacados para el análisis. La Figura 6 muestra el proceso de búsqueda y selección, que incluye múltiples pasos y se llevó a cabo de acuerdo con el protocolo descrito anteriormente.

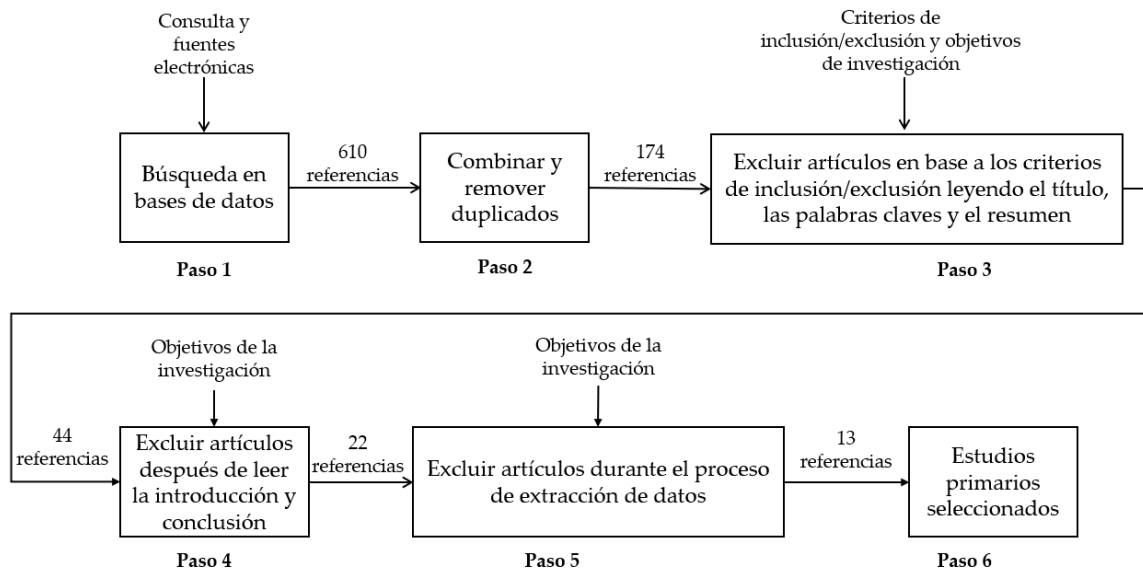


Figura 6. Proceso de selección de artículos académicos.

La revisión sistemática de literatura permite descubrir intersecciones en las tendencias actuales que se pueden utilizar para conocer las posibilidades en el desarrollo potencial y tendencia futura. A continuación se presentan los resultados abordando primeramente el ciclo de vida de los datos en la industria, para

posteriormente revisar la gestión de los datos industriales y concluir con el estado actual de la analítica de Big Data industrial.

2.1.1 Preguntas de investigación y objetivos

El propósito de esta revisión de literatura es conocer qué impulsa la gestión de datos para analítica en los sistemas ciberfísicos industriales, el estado actual, las tendencias y desafíos. La investigación tiene como objetivo responder a las siguientes preguntas de investigación:

PI1: ¿Qué impulsa el IIoT enfocado a la analítica industrial?

PI2: ¿Qué impulsa el cómputo industrial en la nube enfocado a la analítica industrial?

PI3: ¿Qué impulsa el Big Data industrial y su enfoque en analítica?

PI4: ¿Qué estrategias abordan los modelos basados en datos aplicados a la analítica industrial?

PI5: ¿Cómo se integran la convergencia tecnológica en un ecosistema consistente para el desarrollo de soluciones de analítica industrial?

2.1.2 Estrategia de búsqueda

Para obtener una descripción general completa de los métodos basados en datos que respaldan una implementación rápida de sistemas de gestión temprana de fallas para entornos industriales, se buscaron artículos científicos en diferentes fuentes electrónicas accesibles en la Web. El tipo de documento se limitó a publicaciones de conferencias, revistas de investigación y artículos en formato digital.

Para encontrar palabras clave y sus sinónimos más apropiados para ser incluidos en la cadena de búsqueda de la SLR, y probar su efectividad se realizó una búsqueda preliminar basada en una muestra de literatura. La combinación de palabras clave seleccionadas para la cadena de búsqueda, fue aplicada a diferentes librerías electrónicas, como se muestra en la **Tabla 1**.

Tabla 1. Cadena de búsqueda por fuente consultada

Fuente	Cadena de búsqueda
ACM Digital Library	recordAbstract:("Data model " AND ("Big Data" OR "machine learning" OR "Cloud Computing" OR "Internet of Things"))
Google Scholar	"Data management model " and ("Big Data" or "Cloud Computing" or "Internet of Things"
IEEE Digital Library	((("Data model ") AND ("Big Data" OR "machine learning" OR "Cloud Computing" OR "Internet of Things")) Filters Applied: Journals & Magazines
ISI Web of Science	(TS= (("Data model ") AND ("Big Data" OR "machine learning" OR "Cloud Computing" OR "Internet of Things"))) AND Tipos de documento: (Article) Refinado por: Tipos de documento: (ARTICLE)
Science@Direct	Período de tiempo: Todos los años. Índices: SCI-EXPANDED, ESCI. ("Data model ") AND ("Big Data" OR "machine learning" OR "Cloud Computing" OR "Internet of Things") Nota: En este buscador las comillas indican proximidad entre las palabras, lo que significa que no busca por frases.
CONRICyT	((("Data management model ") AND ("Big Data" OR "machine learning" OR "Cloud Computing" OR "Internet of Things"))

2.1.3 Criterios de selección

Esta búsqueda dio como resultado una gran cantidad de artículos, en una variedad de contextos más amplios, como pautas generales, problemas de gestión de datos fuera de contextos industriales o basados en modelo de proceso o en modelo matemáticos. El objetivo de la definición de los criterios de selección fue encontrar toda la literatura publicada relevante a este estudio. En la **Tabla 2**, se muestran los criterios aplicados de inclusión para cada búsqueda que se realizó en las fuentes de la consulta.

Con base en la lectura del resumen y conclusiones se evaluó si el artículo era relevante para el estudio. En caso afirmativo, se hizo una evaluación adicional para hacer una lectura completa al artículo.

Tabla 2. Criterios de inclusión y exclusión.

Criterio de inclusión	Criterio de exclusión
Muestra claramente una contribución a la analítica industrial basado en datos de la industria	No está relacionado con la pregunta de investigación
Presenta los retos o tendencias en técnicas aplicadas a la gestión de datos en la industria, o de la analítica en iCPS	Sólo se refiere a técnicas de gestión de datos, o analítica en iCPS como referencia
Se centra en modelos basados en datos, en la nube industrial, Big Data industrial o Internet Industrial de las Cosas	No está completamente disponible
Incluye un caso real o un estudio de caso	Es una investigación en curso, o las conclusiones no están disponibles en el documento

2.1.4 Evaluación de la calidad

Para evaluar detalladamente cada documento se desarrolló un “instrumento de evaluación de la calidad”, de acuerdo con los lineamientos de [38], con listas de verificación de factores que deben evaluarse para cada estudio. Una lista de verificación de evaluación de la calidad consta de tres partes: una lista de preguntas, una lista de respuestas predefinidas y una puntuación de corte. La puntuación máxima es de 100 y la evaluación mínima para aceptar un artículo fue de 90. Cuarenta y cuatro artículos cumplieron con la evaluación de calidad mínima. Esta evaluación proporciona criterios de inclusión/exclusión aún más detallados, como medio de ponderar la importancia de los estudios individuales cuando se están sintetizando los resultados.

Con este instrumento se evaluó la calidad para verificar los aspectos relevantes en la SLR de acuerdo con los objetivos de la investigación a través de una lista de cotejo para cada documento que se muestra en la **Tabla 3**.

Tabla 3. Formulario de criterios de calidad.

Pregunta	Respuesta		
¿El tema cubierto en la investigación es relevante con el objetivo de la revisión?	Alto 20	Medio 15	Bajo 10
¿Se especifican claramente los objetivos de la investigación?	Alto 20	Medio 15	Bajo 10
¿El modelo basado en datos aporta novedad en la analítica industrial?	Alto 20	Medio 15	Bajo 10
¿Existe una descripción de las características de la tecnología convergente en la analítica industrial o de su implementación en el trabajo de investigación?	Alto 20	Medio 15	Bajo 10
¿Se ha validado la aportación a una escala fiable (ya sea en la academia o en la industria)?	Alto 20	Medio 15	Bajo 10

Tabla 4. Formulario de extracción de datos.

Descripción	Valores
País	No Definido
Ámbito del modelo de datos	Big Data, Combinado, Cómputo en la nube, Industria 4.0, Internet de las cosas, Ninguno
Contribución	Arquitectura, Framework, Herramienta, Metodología, Modelo, Plataforma, Procesos, Teoría
Big Data	Analítica, Infraestructura de cómputo, Datos, No Aplica, Seguridad y Privacidad, Infraestructura de Almacenamiento, Visualización
Internet de las Cosas	Arquitectura, Desafíos Generales, Hardware, No aplica, Retos de seguridad y privacidad, Infraestructura inteligente, Aplicaciones sociales, Software, Cadenas de suministro/logística
Industria 4.0	Sistemas Ciberfísicos (CPS), Interoperabilidad, Machine to Machine (M2M), Fábrica Inteligente (Producto/Servicio), No aplica
Síntesis	No Definido
Imagen 1	Dirección web almacenamiento en la nube
Imagen 2	Dirección web almacenamiento en la nube
Imagen 3	Dirección web almacenamiento en la nube
Imagen 4	Dirección web almacenamiento en la nube
Paradigma	No Definido

2.1.5 Extracción de datos

Una vez completado el cribado de artículos, es necesario extraer la información relevante de cada artículo. Los formularios de extracción de datos son útiles para extraer los datos de los estudios seleccionados, para agrupar los resultados relevantes y analizarlos posteriormente. De acuerdo con la metodología de la revisión sistemática de literatura, en esta etapa los elementos específicos que deben recogerse varían para cada trabajo y es necesario guiarse por las preguntas y objetivos de investigación, para ese propósito se elaboró el formulario de extracción de datos que se muestra en la **Tabla 4**. Adicionalmente, después de este nivel de análisis detallado, se excluyeron algunos estudios considerados como irrelevantes.

2.1.6 Convergencia de tecnologías en la gestión de datos

Para comprender más claramente el ciclo de vida de los datos de la Industria 4.0, basado en la literatura revisada, se elaboró la clasificación que se muestra en la **Figura 7**.

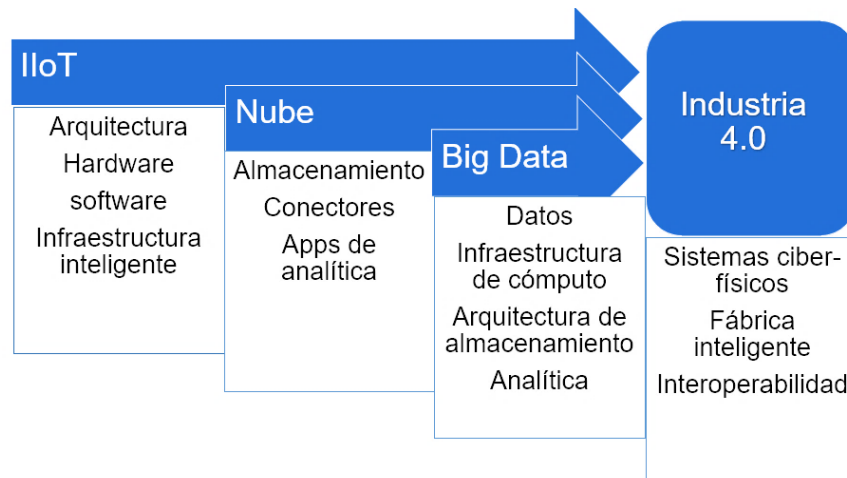


Figura 7. Clasificación de tecnologías con enfoque de gestión de datos.

Se clasificó la categoría de IoT de acuerdo con [39], y se dividió en la subcategoría de arquitectura que, a su vez, puede ser hardware/redes, software y procesos. La subcategoría de hardware incluye identificación por radio frecuencia (RFID, por las

siglas en inglés de Radio Frequency Identification), comunicación de campo cercano (NFC, por las siglas en inglés de Near-Field Communication) y tecnologías de red de sensores; la subcategoría de software incluye middleware (lógica de intercambio de información) y búsqueda/navegación de dispositivos. Por último, la infraestructura inteligente integra objetos inteligentes en una infraestructura física que puede proporcionar flexibilidad, confiabilidad y eficiencia en la infraestructura operativa.

Se desarrolló una clasificación con respecto a la nube, basada en [40]. En el almacenamiento, la nube se utiliza para almacenar Big Data desde el IoT, en la categoría del conector, los servicios en la nube se utilizan para integrar diferentes fuentes de datos. Por último, las aplicaciones para analítica utilizan el análisis de Big Data como un servicio en la nube (Analytics as a Service, AaaS).

La categoría de Big Data se desarrolló con base en [22]. En la subcategoría de Big Data, hay artículos centrados en el proceso extracción, transformación y carga (ETL, por las siglas en inglés de Extract, transform and load) de IoT (sensores, RFID). En la subcategoría de computación se encuentra el procesamiento por lotes, el procesamiento de transmisión de datos y el procesamiento de datos en tiempo real. En la subcategoría de arquitectura de almacenamiento, los estudios pueden abordar propuestas para almacenar datos en bases de datos estructuradas (SQL) o no estructuradas (NoSQL) y almacenes de Big Data. La subcategoría de analítica incluye los componentes que permiten algoritmos de aprendizaje automático, minería de datos y visualización avanzada de datos.

Se clasificó la Industria 4.0 de acuerdo a [41]. Los sistemas ciberfísicos integran computación, redes y procesos físicos. La subcategoría de fábrica inteligente busca flexibilizar los procesos de fabricación, y subcategoría de interoperabilidad es la intercomunicación transparente entre sistemas, personas e información en sistemas

ciberfísicos que permiten intercambiar información entre máquinas, procesos, interfaces y personas.

2.1.7 El alcance de la gestión de datos en la Industria 4.0

Para obtener información sobre el estado actual de la gestión y organización de los datos sobre la convergencia de la Industria 4.0, se analizaron las propuestas que existen para la Industria 4.0 y en particular, tecnologías emergentes que componen el IoT, el Big Data y la nube, así como sus posibles combinaciones (ver **Figura 8**).

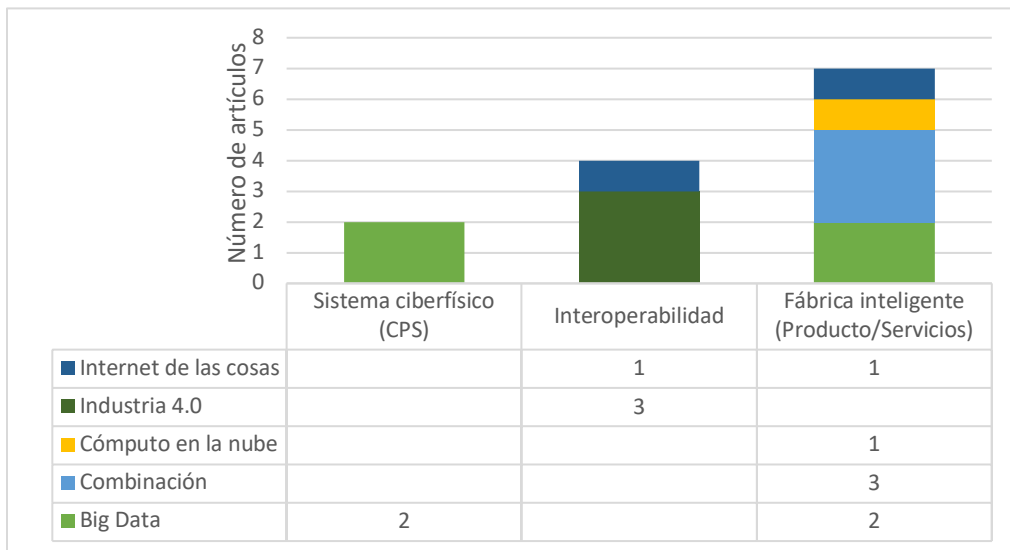


Figura 8. Clasificación de artículos de Gestión de Datos en la Industria 4.0.

Con el desarrollo de tecnologías de IoT en crecimiento, han surgido diferentes enfoques de gestión, por lo que en los últimos años se han propuesto diferentes arquitecturas para nubes de sensores.

Para [42], la arquitectura de los sensores en la nube se debe implementar en los datos: seguridad, velocidad y fiabilidad al proporcionar servicios adecuados de procesamiento y gestión de datos. Para [43], un modelo basado en software para administrar datos en el IoT se basa en una arquitectura para controlar el internet de

las cosas en tres capas: la capa de clúster de sensores, la capa de middleware que administra los datos y la capa de aplicación que considera los datos como servicios.

Con respecto a la nube y la Industria 4.0, los autores en [44] proponen un servicio como plataforma (SaaP, por las siglas en inglés de Service as a Platform) llamado Hana para capturar información de fabricación proveniente de IIoT con tecnologías para análisis en tiempo real que se pueden integrar con sistemas de planificación de recursos empresariales (ERP, por las siglas en inglés de Enterprise Resource Planning).

Actualmente, existe una tendencia a utilizar Big Data en la Industria 4.0, en este sentido en [45] se describe un motor de Big Data definido para modelos de análisis industrial de Big Data, que procesa los datos en paralelo para mejorar el rendimiento. En [21] los autores presentan el modelo multidimensional para el análisis de datos en un almacén de Big Data implementado en Hive, un proyecto Apache de código abierto, usado como herramienta para crear almacenes para conjuntos de datos estructurados y apoyo a la toma de decisiones. En [46] los autores publican una arquitectura para análisis de Big Data desarrollada para la Industria 4.0 que soporta la recopilación, integración, almacenamiento y distribución de datos diseñados para tener en cuenta el volumen, la variedad y la velocidad de los datos que pueden provenir de diferentes necesidades de procesamiento, diferentes usuarios finales y sus funciones en el proceso de toma de decisiones. Para [47], el desarrollo de un sistema de Big Data es diferente del desarrollo de un sistema tradicional con un conjunto pequeño de datos, y proponen un primer intento de una metodología que combine el diseño arquitectónico con el modelado de datos.

Una combinación se refiere a cuando el IoT, el Big Data y la nube están integrados en cualquiera de sus posibles combinaciones para responder a un problema. En este sentido, en [21] proponen una arquitectura de datos basada en cinco capas para integrar sensores, actuadores, redes, computo en la nube y tecnologías IoT para la

generación de aplicaciones para la Industria 4.0. Para mantener la persistencia entre capas, se considera una capa de respuesta que administra los datos. En [48], el marco COIB (por las siglas en inglés de Cognitive Oriented IoT Big-data Framework) propone integrar Big Data con IoT para implementar una arquitectura para la gestión de datos.

Para la Industria 4.0, los autores en [49] proponen una arquitectura para todo el ciclo de vida de la industria del aluminio 4.0. Esta propuesta teórica se integra en una arquitectura de seis capas, el sistema de sensores físicos (IoT), la plataforma de gestión de datos (nube industrial) y un modelo de análisis de Big Data que permite la toma de decisiones, a través de la supervisión de aplicaciones en tiempo real.

2.1.8 Tendencias y desafíos de la Analítica de Big Data en la industria 4.0

El análisis de los artículos seleccionados indica que la analítica de Big Data recibe actualmente el mayor interés de investigación en la Industria 4.0. El proceso y la planeación en la fabricación con diversas aplicaciones interdepartamentales, mantenimiento y diagnóstico presentan el desafío de la precisión de la predicción, que es una cualidad deseable en la toma de decisiones. Es posible atribuir la importancia del análisis predictivo a la presencia de teorías y métodos relacionados con la predicción de otros campos (por ejemplo, estadísticas) y la aplicabilidad del análisis predictivo a problemas del mundo real. Por otro lado, la falta de implementaciones de análisis prescriptivos es evidente a partir de los resultados. Por lo tanto, es posible asociar esto al desafío de desarrollar aplicaciones de análisis prescriptivo; son inherentemente complejas en comparación con el análisis descriptivo y predictivo, dada la necesidad de alinear la tecnología, el modelado, el pronóstico, la optimización y la experiencia en el campo. Por lo tanto, dado que el área de Big Data en la industria todavía está en sus inicios, no es de extrañar que sólo haya unas pocas aplicaciones de análisis prescriptivo que se han desarrollado.

2.2 Analítica de Big Data en la industria

En esta subsección se caracteriza el ciclo de vida del Big Data de manufactura considerando los impulsores de cambio para la gestión de datos en la industria 4.0: El internet industrial de las cosas, el computo en la nube para la industria y el Big Data industrial aplicado en el contexto de los sistemas ciberfísicos industriales.

La industria basada en Big Data necesita adquirir datos de fabricación a gran escala en el ciclo de vida del producto (Ver **Figura 9**), integrar datos de equipos de la planta y datos externos compartir servicios de datos con los usuarios y, finalmente, lograr la interconexión y la interoperabilidad entre el espacio físico y el espacio cibernético.

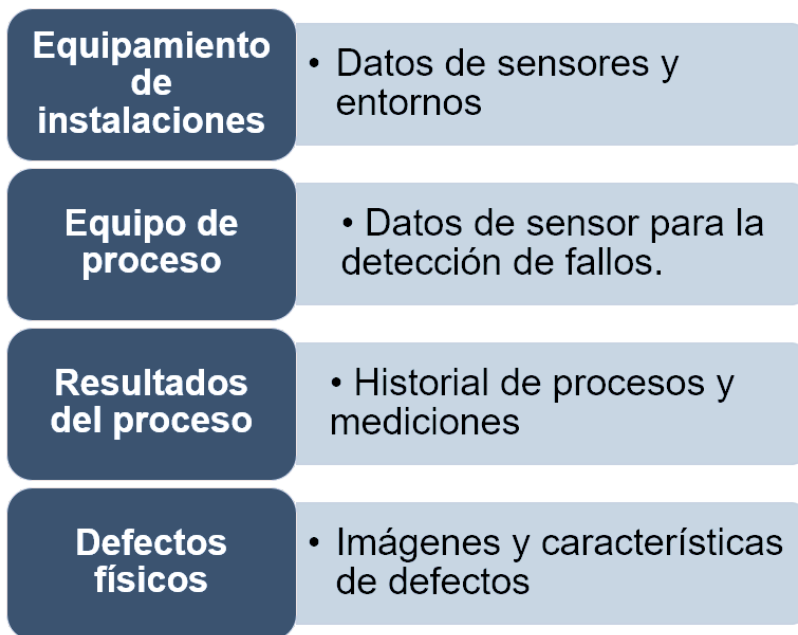


Figura 9. Datos en los procesos de manufactura

2.2.1 Impulsores de la gestión de Big Data industrial

Los datos son un activo invaluable en iCPS [11], pues permiten la manufactura inteligente. Su importancia estratégica es obtener el valor para la toma de decisiones a través del procesamiento de Big Data.

La **Figura 10** muestra que en iCPS, el Big Data surge de la acumulación de datos generados en el ciclo de vida de manufactura [50], planeación, producción y el mantenimiento. Antes de que comience el proceso de producción, el plan de producción inteligente se realiza teniendo en cuenta los datos de recursos del proceso de producción y con fundamento en la relación de los datos globales, el plan de producción global y optimizado puede generarse rápidamente, mejorando la velocidad y precisión de la planeación.

En la fabricación, los datos en tiempo real facilitan el monitoreo del proceso de producción, de modo que los fabricantes puedan mantenerse actualizados sobre las desviaciones de producción para generar planes de control operativo óptimos [51]. El mantenimiento preventivo activo, a través del almacenamiento y análisis de Big Data del IIoT facilita el diagnóstico de fallas y la optimización del proceso de operación.

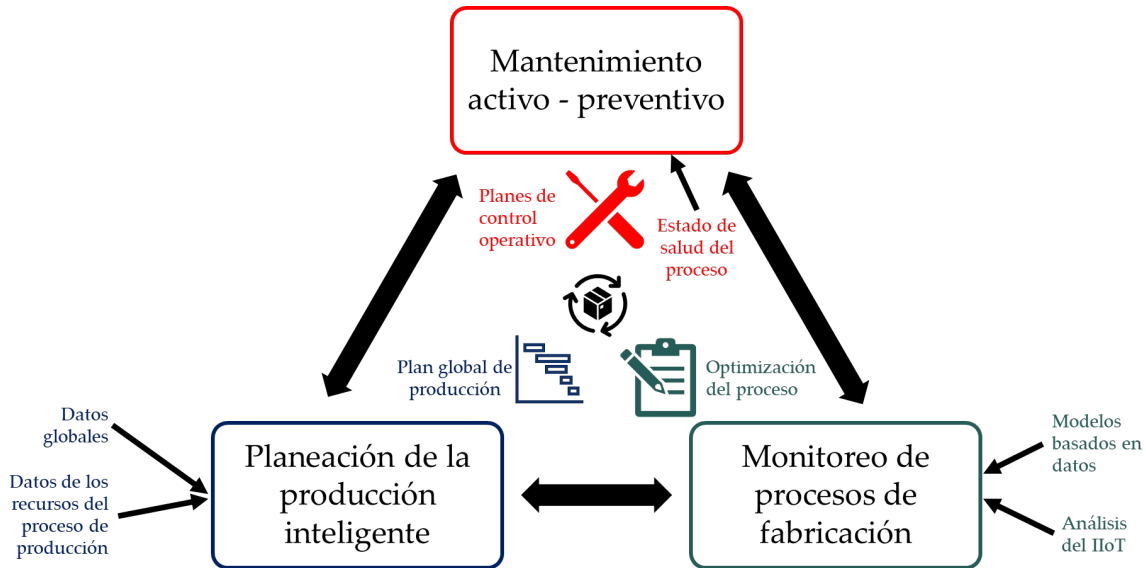


Figura 10. Datos provenientes del ciclo de vida de manufactura.

A continuación, se describe de forma simplificada el origen de los datos en iCPS para analítica en la industria, que es útil para comprender los impulsores de la gestión de Big Data industrial, incluyendo datos en tiempo real para procesos

industriales y datos para sistemas de información de fabricación. En [11] se puede encontrar una clasificación completa de los tipos de datos para la fabricación inteligente.

- Datos de recursos del procesamiento de producción, incluidos a) datos recopilados de iCPS por el IIoT; b) los datos de material y producto recopilados de sí mismos y de los sistemas de servicio; c) datos ambientales.
- Datos de gestión de los sistemas de información de fabricación (Sistema de ejecución de manufactura (MES, por las siglas en inglés de Manufacturing Execution System), Planificación de recursos empresariales (ERP, por las siglas en inglés de Enterprise Resource Planning), Gestión de relaciones con el cliente (CRM, por las siglas en inglés de Customer Relationship Management), Gestión de la cadena de suministro (SCM, por las siglas en inglés de Supply Chain Software), Método de diagrama de precedencia (PDM, por las siglas en inglés de Precedence Diagramming Method), Sistemas asistidos por computadora (CAS, por las siglas en inglés de Computer Aided Systems), Diseño asistido por computadora (CAD, por las siglas en inglés de Computer-Aided Design), Ingeniería asistida por computadora (CAE, por las siglas en inglés de Computer-Aided Engineering), y la fabricación asistida por computadora (CAM, por las siglas en inglés de computer aided manufacturing).

2.2.2 Restricciones en la gestión de datos industriales

Las restricciones son limitaciones para el proceso de diseño, son un subconjunto de requisitos que dan forma a la arquitectura. La **Tabla 5** agrupan las diferentes características de los datos y análisis respecto a la restricción de diseño.

Tabla 5. Propiedades y restricciones de datos y analítica.

Propiedad	Restricción
Datos	
Muestreo	Los datos proporcionados se muestrean equidistantemente.
Volumen	Big Data (365 x 24 x 60 x sensores)
Veracidad	Los datos contienen valores atípicos, fallos del sensor y valores perdidos.
Variedad	Los datos incluyen valores atípicos, errores del sensor y valores que faltan.
Redundancia	Los datos incluyen sensores redundantes muy correlacionados.
Analítica	
Visual	Las aplicaciones informáticas deben visualizar Big data, sus propiedades, predicciones y análisis de anomalías.
Escalable	El procesamiento de datos debe ser escalable y elástico en respuesta al aumento de los datos.
Flexible	Soporte para limpieza de datos.
Extensible	Admite Extracción-Transformación-Carga (ETL) y métodos de aprendizaje.

2.2.3 Atributos de calidad del Big Data industrial

Los atributos de calidad en este contexto están relacionados con el diseño de productos de software, con los requisitos funcionales que debe satisfacer el diseño de la arquitectura de software. Por otro lado, los atributos de la calidad de los datos, aunque son una cuestión importante en la calidad de la fabricación, están fuera del alcance de este trabajo y se tratan en otros trabajos, como [52].

El esquema propuesto por [53], presentado en la **Figura 11**, describe un atributo de calidad. El estímulo describe un evento que llega al sistema y representa una condición que requiere una respuesta. La fuente del estímulo puede afectar la forma en que el sistema trata el estímulo. La respuesta es la actividad que se realiza en respuesta a la llegada de un estímulo. La medida de la respuesta permite determinar si se cumplió el requisito. El artefacto es la parte del sistema que aplica el requisito. El entorno es el conjunto de circunstancias bajo las que se realiza el estímulo.

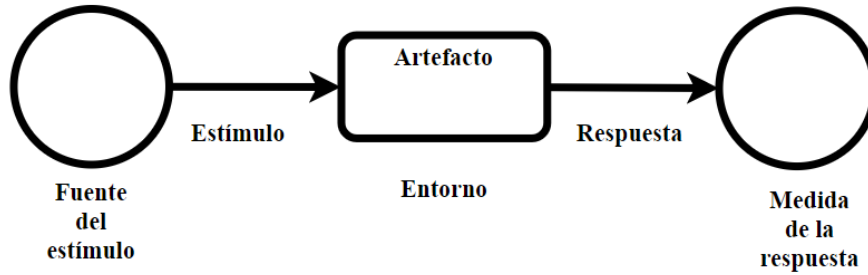


Figura 11. Definición de un escenario.

La técnica de escenarios se centra en identificar el estímulo y cómo el sistema debe responder a él. También se relaciona con los atributos de calidad y busca resaltar las consecuencias de las decisiones arquitectónicas encapsuladas en el diseño. Se consideraron seis escenarios principales para identificar los atributos de calidad que ilustran las características de interés para la gestión de datos en la Industria 4.0. Estos escenarios se propusieron por primera vez en [54] y se definieron siguiendo la técnica de escenarios para la identificación y descripción de requisitos arquitectónicos de calidad [55].

La **Figura 12** describe la fuente de estímulo basada en los datos del ciclo de vida de fabricación que es la misma para los seis escenarios.

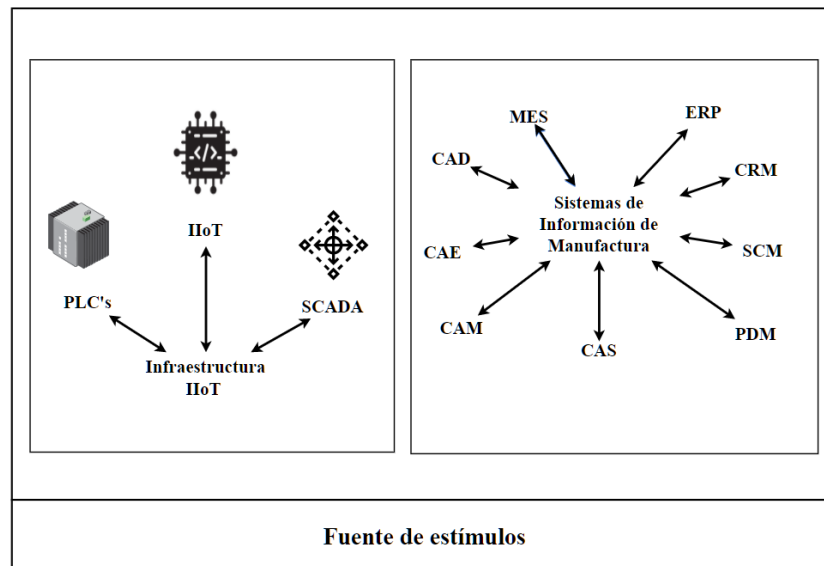


Figura 12. Fuente de estímulo para los principales escenarios de atributos de calidad.

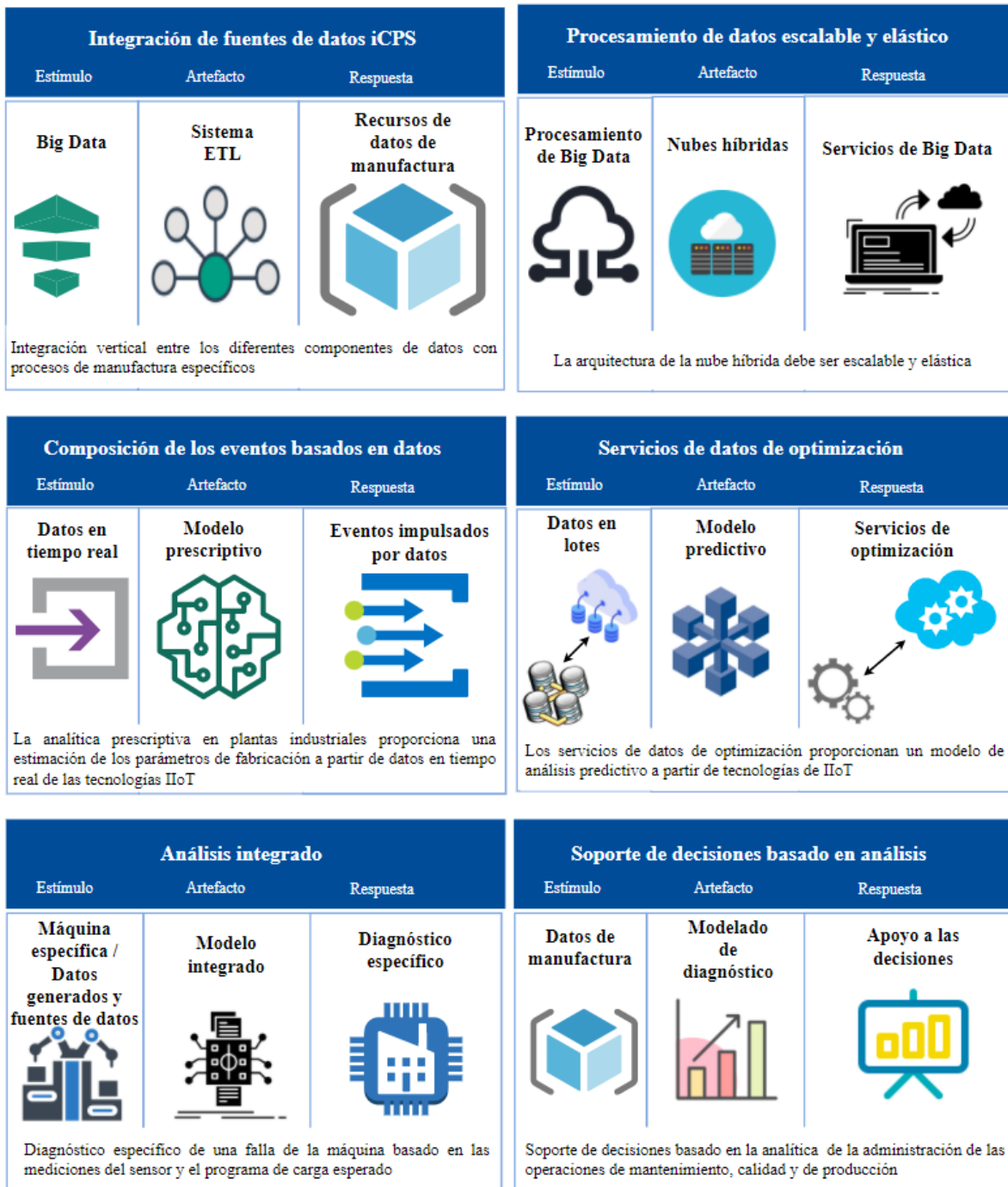








Figura 13. Definición de escenario.

La Figura 13 muestra la definición de escenario para cada atributo industrial de Big Data.

Con base en lo anterior, los atributos de calidad se definen en la **Tabla 6**.

Tabla 6. Atributos de calidad para la gestión de datos para Big Data industrial.

Atributos de calidad	Descripción
Integración de fuentes de datos iCPS	 <p>La amplia variedad de sistemas ciberfísicos industriales (iCPS) implementados en una fábrica inteligente generan enormes cantidades de datos. Sin embargo, dado que estos datos provienen de fuentes heterogéneas (PLC, SCADA, ERP), se requiere un sistema ETL para la combinación e integración y almacenamiento posterior en Big Data a gran escala.</p>
Procesamiento de datos escalable y elástico	 <p>Para garantizar el procesamiento de Big Data procedente de tecnologías IIoT (sensores inteligentes, RFID), la arquitectura de nube híbrida tiene que ser escalable y elástica.</p>
Composición de los eventos basados en datos	 <p>Proporciona una estimación de análisis prescriptivo a partir de datos en tiempo real de tecnologías IIoT (sensores inteligentes, RFID) a partir de los parámetros de fabricación esperados en un tiempo de respuesta fiable.</p>
Servicios de datos de optimización	 <p>Proporciona un modelo de análisis predictivo a partir de tecnologías IIoT (sensores inteligentes, RFID) en el procesamiento de Big data en nubes híbridas en un tiempo de respuesta fiable.</p>
Análisis integrado	 <p>Proporcionar algoritmos específicos de análisis de datos adaptados al hardware integrado que genere información cercana al proceso/máquina específica basada en datos generados propios y fuentes de datos en reposo en un tiempo de respuesta confiable.</p>
Soporte de decisiones basado en análisis	 <p>Integración de los datos de fabricación procedentes de las tecnologías IIoT y los Sistemas de Información de Fabricación (MIS) en la toma de decisiones empresariales a través de análisis prescriptivos avanzados, para realizar un análisis paramétrico de los indicadores clave de rendimiento (KPI) del negocio y estimar el error/riesgo o las predicciones de estos KPI.</p>

2.3 Modelo de analítica para detección de fallas en la industria

Esta subsección caracteriza los métodos de aprendizaje que utilizan el Big Data del ciclo de vida de manufactura para la detección de anomalías, que puede resultar en la detección de fallas en los procesos de los iCPS.

Una tarea importante de los métodos de aprendizaje es construir buenos modelos a partir de conjuntos de datos. Un "conjunto de datos" generalmente consta de vectores de características, donde cada vector de una característica es una descripción de un objeto mediante un conjunto de características [56], ver **Figura 14**.

El número de características de un conjunto de datos se denomina dimensión o dimensionalidad.

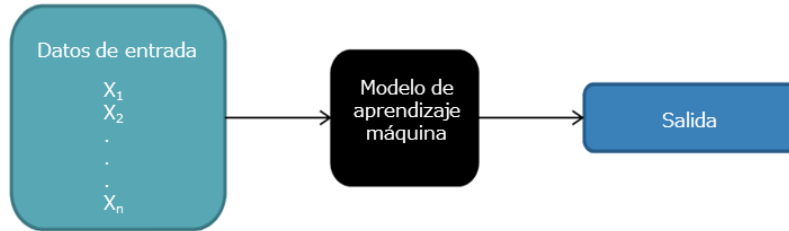


Figura 14. Modelo de aprendizaje.

Las características también se denominan atributos, un vector de una característica también se denomina instancia y, a veces, un conjunto de datos se denomina muestra. Un "modelo" suele ser un modelo predictivo o un modelo de la estructura de los datos que se quiere construir o descubrir a partir del conjunto de datos, como un árbol de decisiones, una red neuronal, una máquina vectorial de soporte, etc.

El proceso de generación de modelos a partir de datos se denomina aprendizaje o entrenamiento, que se logra mediante un algoritmo de aprendizaje. El modelo aprendido se puede llamar una hipótesis. Hay diferentes entornos de aprendizaje, entre los que los más comunes son el aprendizaje supervisado y el aprendizaje no supervisado. En el aprendizaje supervisado, el objetivo es predecir el valor de una entidad de destino en instancias no vistas, y el modelo aprendido también se denomina predictor. Básicamente, si un modelo es "bueno" depende de si puede cumplir con los requisitos del usuario. Diferentes usuarios pueden tener diferentes expectativas de los resultados de aprendizaje, y es difícil conocer la "expectativa correcta" antes de que se haya abordado la tarea en cuestión. Una estrategia popular es evaluar y estimar el rendimiento de los modelos y, a continuación, permitir que

el usuario decida si un modelo es aceptable o elegir el mejor modelo disponible de un conjunto de candidatos.

Dado que el objetivo fundamental del aprendizaje es la generalización, es decir, ser capaz de generalizar el "conocimiento" aprendido de los datos de entrenamiento a instancias de datos nuevas, un buen modelo de aprendizaje debe generalizar bien, es decir, tener un pequeño error de generalización, también llamado error de predicción. En la **Figura 15** se muestran los escenarios en funciones de la disponibilidad de la etiqueta de verdad. En el aprendizaje no supervisado es inviable estimar el error de generalización directamente, ya que eso requiere conocer la información de la etiqueta de la verdad que se desconoce para los casos nuevos. En el aprendizaje supervisado, un proceso empírico típico es permitir que el predictor haga predicciones en los datos de prueba de los cuales se conocen las etiquetas de la verdad, y tomar el error de prueba como una estimación del error de generalización.

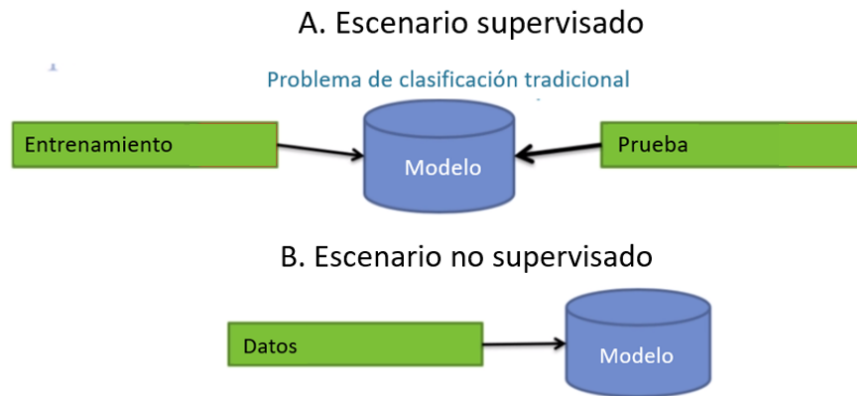


Figura 15. Escenarios para métodos de detección de anomalías.

El proceso de aplicar un modelo aprendido a los datos nuevos se denomina prueba. Antes de las pruebas, a menudo es necesario configurar un modelo aprendido, por ejemplo, ajustar los parámetros, y este proceso también implica el uso de datos con etiquetas de verdad conocidas para evaluar el rendimiento del

aprendizaje; esto se llama validación y los datos son datos de validación. Por lo general, los datos de prueba no deben superponerse con los datos de entrenamiento y validación; de lo contrario, el rendimiento estimado puede ser demasiado optimista.

2.3.1 Modelo de aprendizaje para la detección de fallas

Los algoritmos de aprendizaje automático se utilizan a menudo para mejorar el diagnóstico de fallas del sistema de fabricación. A continuación se presenta una revisión de las aplicaciones recientes de diagnóstico de fallas que se basan en varios algoritmos de aprendizaje automático destacados: Los métodos de aprendizaje [57][57], como las redes neuronales (RN), las redes bayesianas (RB), la máquina de vectores de soporte (MVS) y las técnicas ocultas del modelo de Márkov (MM) se han utilizado para mejorar el diagnóstico de fallas en la industria. Estos métodos de aprendizaje se han aplicado en diferentes áreas, desde la máquina de control numérico por computadora (CNC) en fallas de rugosidad de la superficie, hasta el proceso de grabado de obleas en la fabricación de semiconductores.

En la **Tabla 7** se resumen las técnicas de aprendizaje más comunes a partir de una investigación que abarca del 2007 al 2017, adaptado de [58], el entrenamiento del algoritmo de aprendizaje en la mayoría de los artículos revisados está soportado en experimentos diseñados para simular diferentes condiciones de procesamiento defectuosos y normales. se resumen las técnicas de aprendizaje más comunes a partir de una investigación que abarca del 2007 al 2017, adaptado de [58], el entrenamiento del algoritmo de aprendizaje en la mayoría de los artículos revisados está soportado en experimentos diseñados para simular diferentes condiciones de procesamiento defectuosos y normales.

Tabla 7. Métodos de aprendizaje utilizados en el diagnóstico de fallas (Adaptada de [58]).

Técnica	Ventajas	Desventajas
<p>BN. Las redes bayesianas son una técnica de aprendizaje automático utilizada comúnmente para la detección de fallas. Es un gráfico acíclico dirigido cuyos nodos representan variables aleatorias y sus dependencias condicionales se representan mediante arcos dirigidos que unen los nodos.</p>	<p>Intuitivamente fácil de entender. Bueno para modelar la incertidumbre. Puede utilizarse para modelar niveles jerárquicos de múltiples causas y efectos. Se puede razonar en ambas direcciones (predicción y diagnóstico).</p>	<p>La estructura de árbol hace que sea relativamente menos fácil de inicializar. Construir la estructura del árbol puede ser un desafío.</p>
<p>ANN. La red neuronal artificial es un algoritmo de aprendizaje automático no paramétrico inspirado en el funcionamiento del sistema nervioso central humano.</p>	<p>Puede modelar problemas complejos no lineales con alto grado de precisión. Relativamente más fácil de inicializar, no es necesario especificar la estructura de la red como en el caso de BN.</p>	<p>El modelo no es fácil de interpretar y no puede lidiar con la incertidumbre en los insumos. La convergencia computacionalmente intensiva suele ser lenta durante el entrenamiento. Propenso a sobrealimentación.</p>
<p>SVM. Las máquinas de vectores de soporte utilizan diferentes funciones del núcleo como la función de base radial (RBF) o el núcleo polinómico para encontrar un hiperplano que mejor separa los datos en sus clases, y tiene un buen rendimiento de clasificación cuando se utiliza con pequeños conjuntos de entrenamiento.</p>	<p>Excelente en el modelado de relaciones lineales y no lineales. El tiempo de cálculo es relativamente rápido en comparación con ANN. Tiende a generalizar bien incluso con una cantidad limitada de datos de entrenamiento.</p>	<p>La selección de los parámetros de la función del núcleo es un reto. No es fácil incorporar el conocimiento del dominio. Difícil de entender la función aprendida.</p>
<p>HMM. El modelo oculto de Markov es una extensión del modelo de cadena de Markov que se utiliza para estimar las distribuciones de probabilidad de las transiciones de estado y las salidas de medición en un proceso dinámico, dados los estados no observables del proceso.</p>	<p>Excelente en procesos de modelado con estados inobservables.</p>	<p>El proceso de entrenamiento suele ser computacionalmente intensivo.</p>

Recientemente, la detección de anomalías aplicada a iCPS ha permitido una nueva forma de optimizar los sistemas, procesos y máquinas al detectar anomalías desconocidas, ayudando a los analistas y operadores industriales a resolver posibles problemas [31,59–61]: condiciones de proceso inusuales, características atípicas del producto, equipos de instalaciones, equipos de proceso, defectos físicos.

Es por esto que los modelos para el análisis de anomalías han sido ampliamente estudiados por los científicos de datos, el aprendizaje automático y la estadística [62]

para su aplicación en la industria. A pesar de la enorme cantidad de datos disponibles en la industria, los eventos particulares de interés siguen siendo muy raros [63].

Estos eventos raros, a menudo llamados anomalías, se definen como eventos que ocurren con muy poca frecuencia (su frecuencia varía del 5% a menos del 0,01% dependiendo de la aplicación) [64]. A menudo es un juicio subjetivo, en cuanto a lo que constituye una desviación "suficiente" para que un punto se considere una anomalía. En aplicaciones reales, los datos pueden estar incrustados en una cantidad significativa de ruido, y dicho ruido puede no ser de ningún interés para el analista. Por lo general, son las desviaciones significativamente interesantes que son de interés.

Para ilustrar este punto, considérense los ejemplos que se muestran en la **Figura 16** (a) y (b). Es evidente que los patrones principales (o clústeres) en los datos son idénticos en ambos casos, aunque hay diferencias significativas fuera de estos grupos principales. En el caso de la **Figura 16** (a), un único punto de datos (marcado por 'A') parece ser muy diferente de los datos restantes, y por lo tanto es muy probablemente una anomalía. La situación en la **Figura 16** (b) es mucho más subjetiva. Mientras que el punto de datos correspondiente 'A' en la **Figura 16** (b) también está en una región dispersa de los datos, es mucho más difícil afirmar con confianza que representa una verdadera desviación del conjunto de datos restante. Es muy probable que este punto de datos represente ruido distribuido aleatoriamente en los datos. Esto se debe a que el punto 'A' parece encajar con un patrón representado por otros puntos distribuidos aleatoriamente. Por lo tanto, el término "atípico" se refiere a un punto de datos que podría considerarse ruido, mientras que una "anomalía" se refiere a un tipo especial de valor atípico que es de interés para un analista de datos.

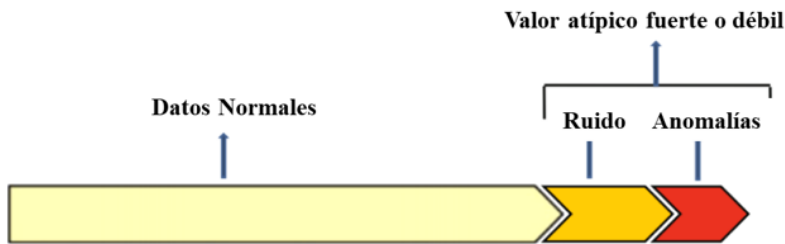
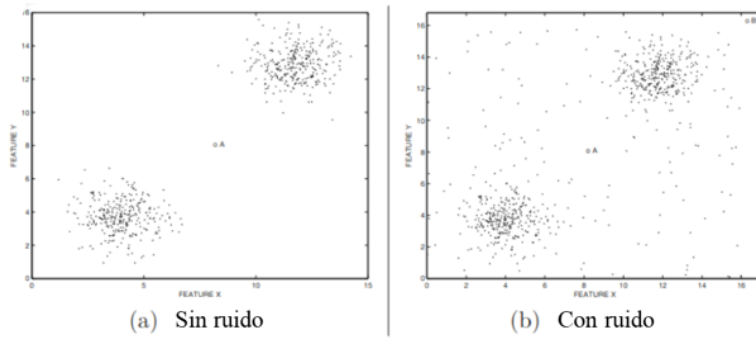


Figura 16. El espectro de datos normales a valores atípicos, adaptado de [65].

Sin embargo, hay una variedad de casos en la práctica en los que esta suposición básica es ambigua. La **Figura 17** ilustra algunos de estos casos utilizando un simple conjunto de datos bidimensional.

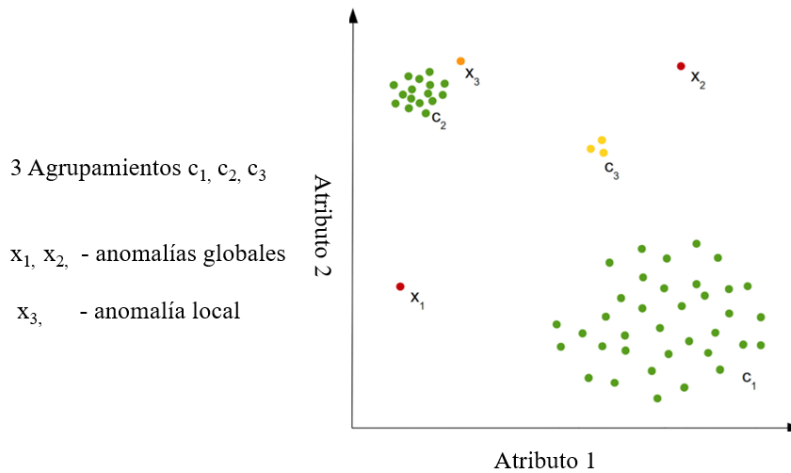


Figura 17. Tipos de anomalías.

Dos anomalías se pueden identificar fácilmente a simple vista: x_1 y x_2 son muy diferentes de las áreas densas con respecto a sus atributos y por lo tanto se

denominan anomalías globales. Al mirar el conjunto de datos globalmente, x_3 se puede ver como un registro normal, ya que no está demasiado lejos del clúster c_2 . Sin embargo, cuando nos centramos sólo en el clúster c_2 y lo comparamos con x_3 mientras descuidamos todas las otras instancias, puede ser visto como una anomalía. Por lo tanto, x_3 se llama una anomalía local, ya que sólo es anómalo cuando se compara con su vecindario cercano. Depende de la aplicación, independientemente de si las anomalías locales son de interés o no.

Otra pregunta interesante es si las instancias del clúster c_3 deben considerarse como tres anomalías o como un clúster (pequeño) regular. Estos fenómenos se denominan micro clúster y los algoritmos de detección de anomalías deben asignar puntuaciones a sus miembros más grandes que las instancias normales, pero valores más pequeños que las anomalías obvias. Este sencillo ejemplo ya ilustra que las anomalías no siempre son obvias y una puntuación es mucho más útil que una asignación de etiquetas binarias.

A diferencia de la conocida configuración de clasificación, donde los datos de entrenamiento se utilizan para entrenar a un clasificador y los datos de prueba miden el rendimiento después, hay múltiples configuraciones posibles cuando se habla de detección de anomalías. Básicamente, la configuración de detección de anomalías que se utilizará depende de las etiquetas disponibles en el conjunto de datos y podemos distinguir entre dos tipos principales: los métodos de aprendizaje supervisados y los no supervisados.

Los métodos de aprendizaje supervisados generalmente crean un modelo de predicción para eventos anómalos basados en datos etiquetados (el conjunto de entrenamiento) y lo usan para clasificar cada caso [66]. La principal debilidad de las técnicas de extracción de datos supervisadas es la necesidad de tener datos etiquetados, que pueden llevar a la incapacidad para detectar nuevos tipos de eventos anómalos.

Por otro lado, los métodos de aprendizaje no supervisados no necesitan datos etiquetados y detectan eventos como datos muy distintos de la mayoría de los datos basados en alguna medida [67]. Las técnicas de detección de anomalías no supervisadas se basan en suposiciones sobre valores atípicos frente al resto de los datos.

Los algoritmos de detección de valores atípicos pueden detectar nuevos tipos de eventos raros como desviaciones del comportamiento normal, pero, por otro lado, adolecen de una posible alta tasa de falsos positivos, principalmente porque los datos que no se habían procesado anteriormente (aunque normales) también se reconocen como errores/ anomalías y, por lo tanto, se señalan como interesantes. Es posible clasificar los métodos de detección de valores atípicos en cinco:

- *Métodos probabilísticos y estadísticos.* Los datos se modelan en la forma de una distribución de probabilidad cercana y el modelo aprende los parámetros de esta distribución. La eficacia de los modelos estadísticos depende en gran medida de los supuestos realizados para el conjunto de datos dado.
- *Métodos basados en proximidad.* La idea de estos métodos es modelar valores atípicos como puntos que están separados de los datos restantes de acuerdo con las funciones de similitud o distancia. Los métodos basados en proximidad se pueden aplicar de tres maneras, que son métodos de agrupación en clústeres, métodos basados en densidad y métodos de vecino más cercano. Los métodos de proximidad basan su eficacia en la medición de proximidad utilizada.
- *Métodos de agrupación.* En los métodos de agrupación, el primer paso es utilizar un algoritmo para determinar las regiones densas del conjunto de datos. En el segundo paso, se utiliza alguna medida del ajuste de los puntos de datos a los diferentes clústeres para calcular una puntuación atípica para el punto de datos. Una simple adaptación de un algoritmo de agrupación

para la detección de valores atípicos puede ser muy costosa en el procesamiento del tiempo y no puede extenderse adecuadamente en algunas aplicaciones de Big Data.

- *Enfoque multidimensional.* La alta dimensionalidad plantea enormes desafíos para la detección efectiva de valores atípicos: la distancia y la similitud entre dos puntos en un espacio de alta dimensión pueden no reflejar la relación real entre los puntos. En consecuencia, los métodos convencionales de detección de valor atípico que utilizan principalmente proximidad o densidad para identificar valores atípicos se deterioran a medida que aumenta la dimensionalidad.
- *Redes neuronales.* El aprendizaje profundo integra en un método el aprendizaje de características y la construcción de modelos mediante la selección de diferentes núcleos (“kernel”) o el ajuste de los parámetros a través de la optimización de extremo a extremo [68]. La arquitectura profunda de redes neuronales con muchas capas ocultas son esencialmente operaciones no lineales de varios niveles. Existen dos problemas con el uso de redes neuronales [69]. El primer problema es que las redes neuronales tardan en entrenarse. El segundo problema con las redes neuronales es que son sensibles al ruido. Dado que los valores atípicos se tratan como puntos normales durante la fase de entrenamiento, inevitablemente habrá errores en el modelo. El problema de tratar los valores atípicos como puntos normales se manifestará como sobreajuste. Este problema es particularmente significativo en el caso de las redes multicapa. El Auto Codificador (“Auto Encoder”, AE) es un algoritmo de aprendizaje no supervisado que extrae características de los datos de entrada sin necesidad de información de etiquetas [70]. Los autocodificadores son una opción natural para la detección de valores atípicos porque se utilizan comúnmente para la reducción de la

dimensionalidad de conjuntos de datos multidimensionales como alternativa a la factorización de matriz o PCA (por las siglas en inglés de Principal Component Analysis) [71].

Tabla 8. Algoritmos más comunes para la detección de anomalías no supervisadas, adaptado de [72].

Clasificación	Método	Descripción	Ref.
Modelo estadístico	PCA	Análisis de componentes principales	[73]
Modelo estadístico	MCD	Determinante mínimo de covarianza	[74]
Modelo estadístico	OCSVM	Máquinas vectoriales de soporte de una clase	[75]
Proximidad	LOF	Factor atípico local	[76]
Proximidad	COF	Factor atípico basado en conectividad	[77]
Proximidad	CBLOF	Factor de valores atípicos local basado en clústeres	[78]
Proximidad	LOCI	Detección rápida de valores atípicos mediante la integración de correlación local	[79]
Proximidad	HBOS	Puntuación de valor atípico basada en histograma	[80]
Proximidad	kNN	Vecinos k Próximos	[81]
Proximidad	AvgKNN	Promedio de kNN	[82]
Proximidad	MedKNN	Mediana kNN	[82]
Proximidad	SOD	Detección de valores atípicos sub espaciales	[83]
Probabilística	ABOD	Detección de valores atípicos basados en ángulos	[84]
Probabilística	FastABOD	Detección rápida de valores atípicos basados en ángulos mediante aproximación	[84]
Probabilística	SOS	Selección de atípico estocástica	[85]
Redes neuronales	AutoEncoder	Codificador automático totalmente conectado	[65]
Redes neuronales	SO_GAAL	Aprendizaje Activo Adversarial Generativo de un solo Objetivo	[86]
Redes neuronales	MO_GAAL	Aprendizaje Activo Adversarial Regenerativo De Múltiples Objetivos	[86]

Numerosos algoritmos han sido propuestos para la detección de valores atípicos no supervisados en los últimos años [67]. En [72] varios paquetes de detección de valores atípicos se mencionan: ELKI Data Mining [87], RapidMing [88] en Java y Valores Atípicos en R [89], los algoritmos de detección más populares se han implementado en PyOD, como se muestra en la **Tabla 8**.

2.4 Almacenamiento y procesamiento distribuido de datos

En esta subsección se analizan los componentes principales del ecosistema Apache Hadoop aplicados al desarrollo de la plataforma tecnológica y los conceptos relacionados a este marco de trabajo. Apache Hadoop es un marco de referencia creado para procesar eficientemente grandes volúmenes de datos, que permite la creación de clústeres con equipos de cómputo de consumo y facilita el análisis de datos masivos con el procesamiento de datos en paralelo [90]. Los clústeres de Hadoop son el lugar común donde se almacenan y procesan los datos operativos [91,92], A continuación se describen los tres elementos esenciales de Hadoop: Sistema de archivos distribuido Hadoop (HDFS, por las siglas en inglés de Hadoop Distributed File System), Hadoop MapReduce y Hadoop Común (Hadoop Common).

2.4.1 Sistema de archivos distribuido Hadoop (HDFS)

Apache Hadoop es una implementación de código abierto del marco MapReduce patentado de Google. HDFS es el componente del sistema de archivos de Hadoop [90], es un sistema de archivos distribuido, escalable, escrito en Java y está diseñado para almacenar conjuntos de datos muy grandes de manera confiable y para transmitir estos conjuntos de datos a aplicaciones de usuario con un ancho de banda elevado. Este sistema de archivos utiliza el método de “escribe una vez y lee muchas”, con bloques grandes de 64 MB, haciéndolo ideal para almacenamiento masivo de datos, por ejemplo, la implementación de HIVE como base de datos, otro ejemplo es Hbase.

HDFS fue creado para sacar provecho de equipos de bajo costo, y a la vez, ser tolerante a fallos, al estar los nodos distribuidos en equipos diferentes e incluso en localizaciones distantes.

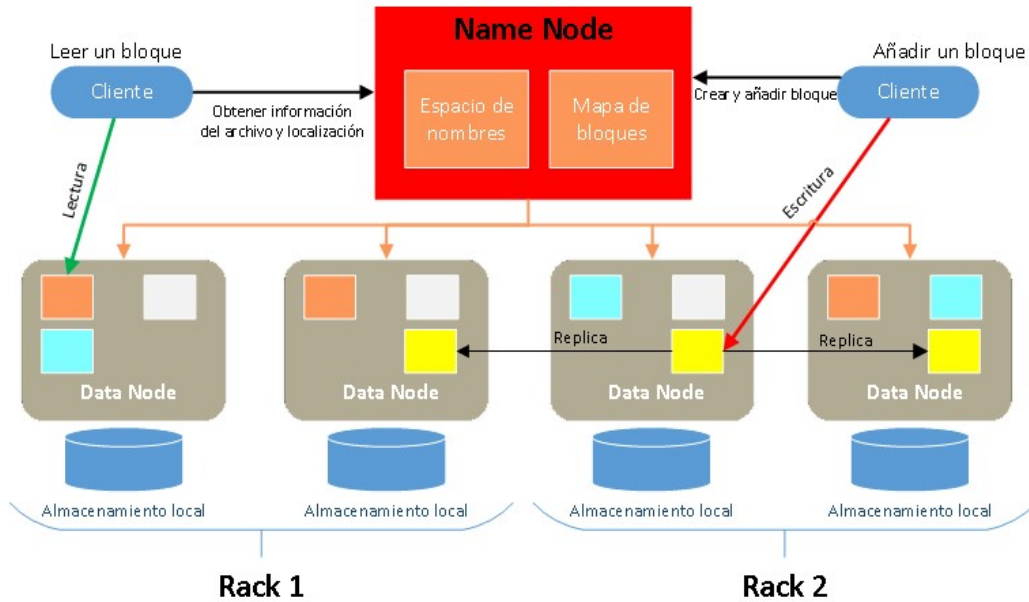


Figura 18. HDFS (fuente: [93]).

En la **Figura 18**, se muestran los dos tipos de nodos diferentes en un clúster HDFS, diferenciados por la función que desempeñan al ser usados. Los tipos de nodos HDFS son los siguientes:

- **Nodo de nombre (Name node)**. Mantiene la jerarquía del espacio de nombres y los metadatos del sistema de archivos, como las ubicaciones de los bloques. El espacio de nombres y los metadatos se almacenan en la RAM, pero periódicamente se descargan en el disco. El registro de modificaciones mantiene actualizada la imagen del disco.
- **Nodo de tareas (Task nodes)**. Este tipo de nodos son los que realizan el acceso a los datos, almacenan los bloques de información y los recuperan bajo demanda. Recibe comandos del nodo de nombres que le indican:
 - Replicar bloques a otros nodos
 - Quitar réplicas de bloques locales
 - Volver a registrarse o apagar
 - Enviar informe de bloqueo inmediato.

2.4.2 Hadoop MapReduce

Hadoop MapReduce [94] permite procesar grandes volúmenes de datos en paralelo, está basado en YARN (por las siglas en inglés de Yet Another Resource Negotiator) para la administración de la tecnología de clústeres y que permite la separación de HDFS, consiguiendo que Hadoop sea más adecuado para aplicaciones que necesiten procesamiento en paralelo. La arquitectura de MapReduce 2.0 se puede observar en la **Figura 19**.

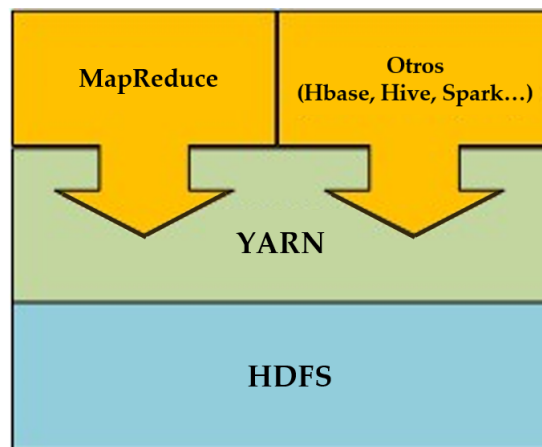


Figura 19. YARN (fuente [93]).

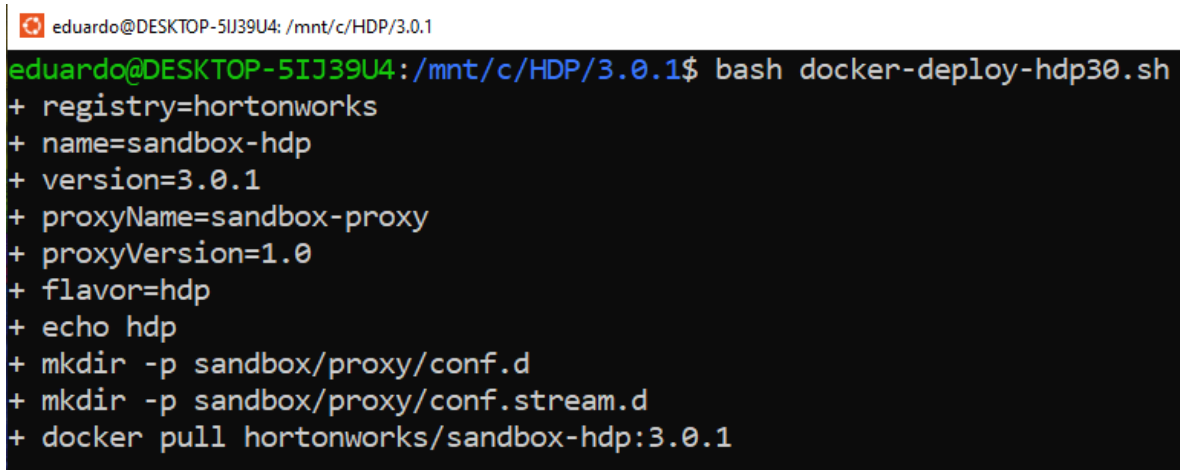
La tecnología MapReduce envía el proceso de cómputo al sitio donde se encuentran los datos a procesar, los datos se encuentran distribuidos en el clúster Hadoop. Al iniciar un proceso de MapReduce, las tareas son distribuidas entre los nodos del clúster y es el marco de referencia (framework) de Hadoop quien lleva a cabo el intercambio de datos entre nodos. La mayor parte de la carga computacional acontece en los nodos que contienen datos locales, para que el tráfico en la red sea mínimo.

Hadoop Común es un conjunto de librerías que soporta otros módulos que contienen los scripts para iniciar Hadoop y las utilerías comunes en las que se apoyan otros módulos

2.4.3 Herramientas de Apache Hadoop

Existen dos alternativas a seguir para implementar una plataforma tecnológica basada en el ecosistema de Apache Hadoop [29]. La primera alternativa utiliza servicios de cómputo en la nube a través de empresas que soportan los servicios de Hadoop y que se integran en forma global a sus propias aplicaciones, por ejemplo, Pivotal, MapR, Microsoft Azure, Amazon AWS, entre otras. La segunda alternativa integra el ecosistema y las herramientas de Apache Hadoop a través de un ambiente tipo arenero (sandbox) que gestiona la instalación y manejo, por ejemplo, Cloudera, MapR Technologies, IBM InfoSphere Insights, HD Microsoft Insight.

Para el desarrollo del prototipo basado en Apache Hadoop se ha seleccionado HDP (Hortons Data Platform) y Data Flow de Cloudera. Ambos están disponibles a través de contenedores Docker creados a partir de una imagen proporcionada por Cloudera, lo que permite enfocarse en el flujo de trabajo de desarrollo y prueba. En la **Figura 20**, se muestra el proceso de despliegue del sanbox-hdp y sandbox-proxy en la versión 3.0.1.



```

eduardo@DESKTOP-5IJ39U4: /mnt/c/HDP/3.0.1
eduardo@DESKTOP-5IJ39U4: /mnt/c/HDP/3.0.1$ bash docker-deploy-hdp30.sh
+ registry=hortonworks
+ name=sandbox-hdp
+ version=3.0.1
+ proxyName=sandbox-proxy
+ proxyVersion=1.0
+ flavor=hdp
+ echo hdp
+ mkdir -p sandbox/proxy/conf.d
+ mkdir -p sandbox/proxy/conf.stream.d
+ docker pull hortonworks/sandbox-hdp:3.0.1

```

Figura 20. Instalación de HDP 3.0.1 en contenedores Docker.

La arquitectura de Hortonworks se puede apreciar en la **Figura 21**, tiene cuatro bloques principales, los cuales se componen de aplicaciones de Apache Hadoop.

Estos bloques son: integración y gobierno, acceso a los datos, seguridad y operaciones.

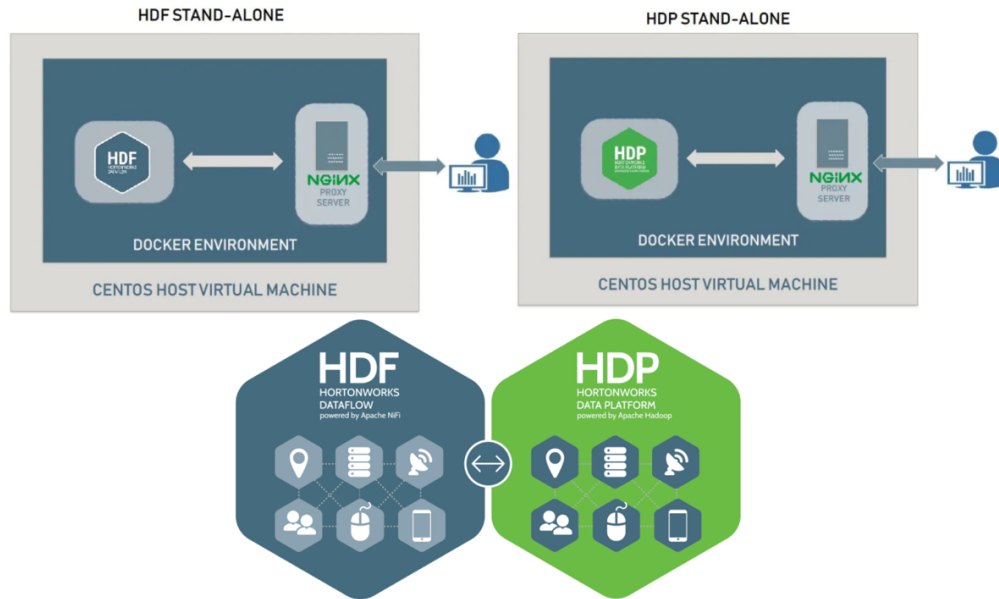


Figura 21. Sandbox Hortonworks (fuente: <https://www.cloudera.com/tutorials/sandbox-architecture.html>).

HDP se utiliza para mantener grandes volúmenes de datos en la nube o localmente de forma segura, en tiempo real, con aplicaciones para desarrollar aplicaciones ágiles para aprendizaje máquina y aprendizaje profundo.

CDF se utiliza para procesar grandes flujos de datos en tiempo real, con aplicaciones para analizar el flujo entrante de datos, para conocer el estado que guardan los datos en una ventana de tiempo determinada.

2.4.4 Sandbox HDP

La plataforma de datos de Hortonworks (HDP, por las siglas en inglés de Hortonworks Data Platform) permite implementar Apache Hadoop para poder escalar a grandes volúmenes de datos y crear soluciones de Big Data para integrar almacenes de datos y conectarlos con otros sistemas de datos.

Existen numerosos proyectos de Apache Software Foundation que se requieren para implementar una solución funcional en una empresa, sin embargo, su uso depende del caso a desarrollar. Abajo se describen los proyectos asociados de Apache incluidos en HDP.

Apache Ambari

El objetivo de Apache Ambari [95] es simplificar la administración de Hadoop mediante el desarrollo de software para aprovisionar, administrar y monitorear clústeres de Apache Hadoop. Ambari proporciona una interfaz de usuario web de administración de Hadoop intuitiva y fácil de usar respaldada por sus API RESTful.

Apache Hive

Apache Hive [96] está construido sobre el marco de trabajo de MapReduce, Hive es un almacén de datos que permite un resumen de datos sencillo y consultas ad-hoc a través de una interfaz similar a SQL para grandes conjuntos de datos almacenados en HDFS.

Apache Pig

Apache Pig [97] es una plataforma para procesar y analizar grandes conjuntos de datos. Pig consiste en un lenguaje de alto nivel (Pig Latin) para expresar programas de análisis de datos emparejados con el marco MapReduce para procesar estos programas.

MapReduce

MapReduce [91] es un marco de trabajo para escribir aplicaciones que procesan grandes cantidades de datos estructurados y no estructurados en paralelo en un grupo de miles de máquinas, de una manera confiable y tolerante a fallas.

Apache Spark

Apache Spark [98] es ideal para el procesamiento de datos en memoria. Permite a los científicos de datos implementar algoritmos rápidos e iterativos para análisis avanzados, como agrupamiento y clasificación de conjuntos de datos.

Apache Hbase

Apache Hbase [99] un sistema de almacenamiento de datos NoSQL orientado a columnas que proporciona acceso aleatorio de lectura / escritura en tiempo real a big data para aplicaciones de usuario.

Apache Tez

Apache Tez [100] generaliza el paradigma MapReduce a un marco más potente para ejecutar un gráfico acíclico dirigido DAG (por las siglas en inglés de Directed Acyclic Graph) complejo de tareas para el procesamiento de big data casi en tiempo real.

Apache Kafka

Apache Kafka [101] es un sistema de mensajería de publicación y suscripción rápido y escalable que se usa a menudo en lugar de los intermediarios de mensajes tradicionales debido a su mayor rendimiento, replicación y tolerancia a fallas.

Apache HCatalog

Apache HCatalog [102] un servicio de administración de tablas y metadatos que proporciona una forma centralizada para que los sistemas de procesamiento de datos comprendan la estructura y ubicación de los datos almacenados en Apache Hadoop.

Apache Solr

Apache Solr [103] es la plataforma de código abierto para búsquedas de datos almacenados en Hadoop. Solr permite una potente búsqueda de texto completo y

una indexación casi en tiempo real en muchos de los sitios de Internet más grandes del mundo.

2.4.5 Cloudera DataFlow (CDF)

La arquitectura Sandbox independiente de HDF viene con las siguientes herramientas de Big Data: Zookeeper, Storm, Kafka, NiFi, NiFi Registry, Schema Registry, Streams Messaging Manager (SMM) y Stream Analytics Manager (SAM).

Apache NiFi

Apache NiFi [104] es una plataforma de logística de datos integrados para la automatización del movimiento de datos entre sistemas diversos. Ofrece control en tiempo real y facilita el movimiento de datos entre cualquier fuente y destino.

Apache ZooKeeper

Apache ZooKeeper [105] proporciona servicios operativos para un clúster de Hadoop. ZooKeeper proporciona un servicio de configuración distribuida, un servicio de sincronización y un registro de nombres para sistemas distribuidos. Las aplicaciones distribuidas usan Zookeeper para almacenar y mediar actualizaciones a información de configuración importante.

Apache Storm

Apache Storm [106] es un sistema para procesar datos de transmisión en tiempo real. Agrega capacidades confiables de procesamiento de datos en tiempo real a Hadoop, para escenarios que requieren análisis en tiempo real, aprendizaje automático y monitoreo continuo de operaciones.

Storm es un sistema de cómputo distribuido en tiempo real para procesar grandes volúmenes de datos de alta velocidad. Storm es extremadamente rápido, con la capacidad de procesar más de un millón de registros por segundo por nodo en un clúster de tamaño modesto.

SAM (Streaming Analytics Manager)

SAM [107] facilita crear aplicaciones de análisis de transmisión para la correlación de eventos, el enriquecimiento del contexto, la coincidencia de patrones complejos y las agregaciones analíticas.

Registro de esquemas (Schema Registry)

El registro de esquemas [108] proporciona un repositorio compartido de esquemas que permite que las aplicaciones interactúen de forma flexible entre sí. El principio de diseño del Registro de esquemas es proporcionar una manera de abordar los desafíos de administrar y compartir esquemas entre componentes y de tal manera que los esquemas estén diseñados para respaldar la evolución de tal manera que un consumidor y productor puedan entender diferentes versiones de esos esquemas, pero aun así leer toda la información compartida entre ambas versiones e ignorar con seguridad el resto.

2.5 Conclusiones

Las subsecciones previas detallan las tecnologías que convergen en el Big Data industrial y describen el ciclo de vida de la gestión de datos en la industria. De la misma manera, se presenta una descripción de los impulsores de la gestión de datos relacionados a la analítica industrial y sus características generales. La integración de estos conceptos tecnológicos requiere de nuevos requerimientos de la gestión de datos industrial o atributos de la calidad del Big Data Industrial que debe satisfacer el diseño de una arquitectura de referencia para analítica industrial basada en Big Data. Estos son los requerimientos que cubre el diseño de la mencionada arquitectura de referencia: (1) Integración de fuentes de datos iCPS, (2) Procesamiento de datos escalable y elástico, (3) Composición de los eventos basados en datos, (4) Servicios de datos de optimización, (5) Análisis integrado, (6) Soporte de decisiones basado en análisis.

Por otro lado, los modelos basados en datos generados por los métodos de aprendizaje automático han sido empleados como una tecnología habilitadora para la analítica inteligente en la industria. Si bien el uso de estos modelos ha sido principalmente en aplicaciones basadas en grandes volúmenes de datos mediante un análisis profundo de los datos y diferentes estrategias para la selección del modelo más adecuado, existen limitaciones para el desarrollo de aplicaciones para fallas industriales que respondan a las dificultades para contar con estos recursos.

Los modelos basados en datos no supervisados para la detección de anomalías aplicados a las fallas industriales presentan un gran potencial para cubrir las limitaciones de los modelos basados en datos supervisados.

Por otro lado, el almacenamiento y procesamiento distribuido de datos ha sido empleado como la principal tecnología para el desarrollo de plataformas de analítica de Big Data. Sin embargo, estas tecnologías aún se encuentran en desarrollo, y sin lugar a duda serán un importante aliado para el manejo del Big Data en los sistemas ciberfísicos industriales en un futuro cercano.

Capítulo 3

Metodología

El tercer capítulo presenta la metodología utilizada en el diseño de la arquitectura para analítica de Big Data industrial y la metodología empleada para seleccionar el modelo de aprendizaje no supervisado para detección temprana de fallas. El método de diseño de la arquitectura seleccionado está basado en el enfoque de diseño basado en atributos (ADD) que consta de siete etapas y se enfoca específicamente en los atributos de calidad a través de la selección de estructuras arquitectónicas y su representación en vistas. También incluye el análisis de la arquitectura y documentación como parte integral del proceso de diseño. El método para la selección del modelo de aprendizaje consta de dos etapas: (1) la selección de los detectores base, (2) selección del mejor modelo de aprendizaje no supervisado.

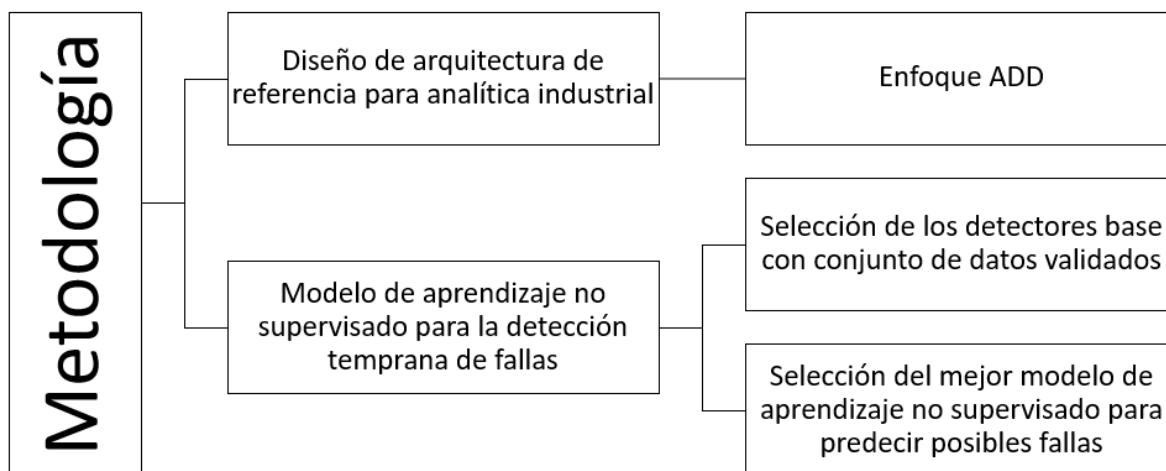


Figura 22. Organización de la metodología empleada en esta tesis.

En la **Figura 22** se presenta la organización de este capítulo. En la sección 3.1 se presenta una metodología basada en el enfoque ADD con la que se diseña una arquitectura de referencia que satisface los requerimientos presentados en la sección 2.2 y que considere la convergencia del IIoT, el cómputo en la nube y el modelado basado en datos que sirva para facilitar el desarrollo de aplicaciones de analítica industrial en el contexto de los iCPS.

En la sección 3.2 se presenta una metodología propuesta para determinar el mejor modelo no supervisado para la detección temprana de fallas.

3.1 Diseño de la arquitectura de referencia para analítica de Big Data industrial

En el área del diseño de arquitecturas basadas en software, la propuesta de arquitecturas de referencia en diferentes contextos es uno de los enfoques de varios profesionales e investigadores [19]. La determinación de estilos o patrones arquitectónicos, o bien, arquitecturas de referencia, facilitan el diseño de soluciones a problemas que tienen aspectos o características comunes.

Sin embargo, en la revisión del estado del arte, se encontró que las arquitecturas para Big Data en la industria se centran en aspectos relacionados con la gestión de datos aplicados a situaciones específicas. Es decir, en los entornos iCPS, la

integración del IIoT con la computación en la nube industrial, incluidos los desafíos de velocidad, volumen, variedad y veracidad, desafían el diseño de sistemas de análisis de Big Data. Por lo que, describir un nuevo proceso de diseño que refleje el cambio necesario para el desarrollo de sistemas de análisis de Big Data que se adapten al cambio de paradigma de la Industria 4.0, es un aporte al estado del arte.

Existen varios métodos de desarrollo de arquitectura de sistemas de software [109], la mayoría de ellos cubren todo el ciclo de vida de la arquitectura y proporcionan pocos detalles sobre cómo realizar la actividad de diseño. El enfoque de diseño basado en atributos (ADD) es el primer método de diseño que se enfoca específicamente en los atributos de calidad a través de la selección de estructuras arquitectónicas y su representación en vistas, también incluye análisis de arquitectura y documentación como parte integral del proceso de diseño [110]. Las actividades de diseño en ADD pueden incluir refinar los bocetos que se crearon durante las primeras iteraciones de diseño para producir una arquitectura más detallada. ADD comienza con requisitos arquitectónicamente significativos (impulsores y restricciones), y los conecta sistemáticamente con las decisiones de diseño y luego las une a las opciones de implementación disponibles a través de los marcos de referencia (frameworks) . ADD también puede usar arquitecturas de referencia con un catálogo de tecnologías que califique sus atributos de calidad, que incluye tácticas, patrones, entorno de trabajo y tecnologías (herramientas).

El proceso ADD consta de siete etapas (ver **Figura 23**), comenzando con las identificaciones de entrada para el diseño de la arquitectura. Estas entradas son el propósito del diseño, los principales requisitos funcionales, los principales escenarios de atributos de calidad, restricciones y preocupaciones arquitectónicas. Con la revisión de las entradas mencionadas, el primer paso es confirmar que existe suficiente información sobre los requisitos a través del ciclo de vida de Big Data industrial: adquisición de datos, preservación de datos, procesamiento de datos.

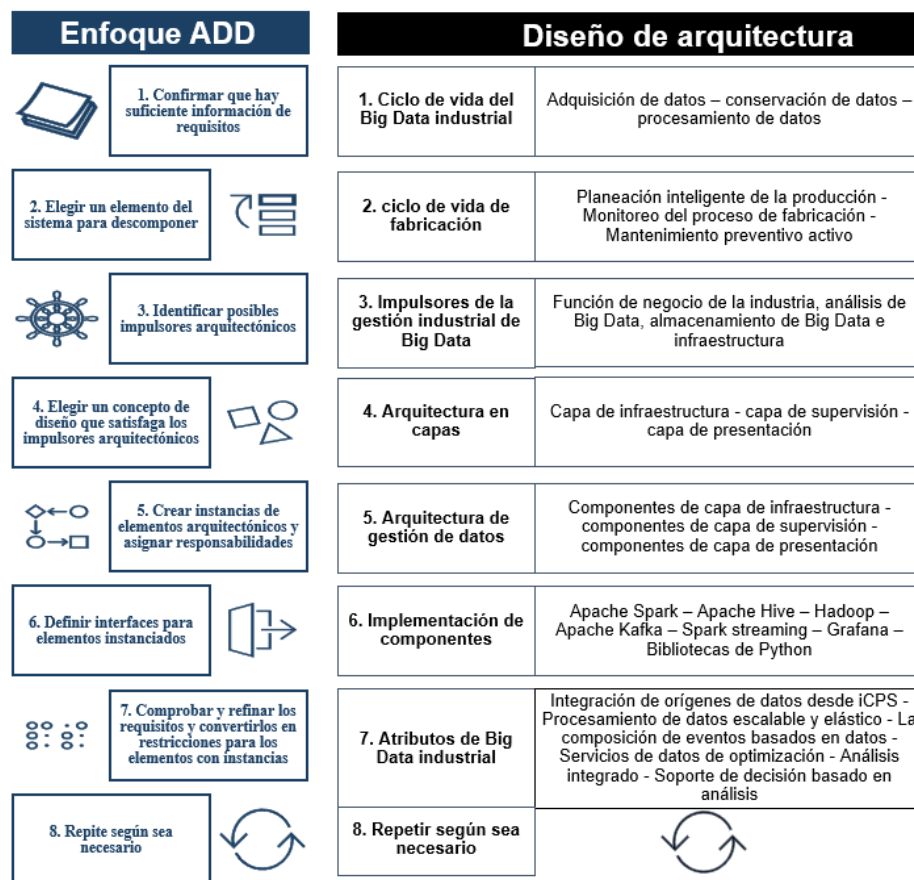


Figura 23. El proceso de diseño basado en atributos.

El segundo paso es elegir un elemento del sistema para dividirlo a lo largo del ciclo de vida de la fabricación (planificación inteligente de la producción, supervisión del proceso de fabricación, mantenimiento preventivo activo).

En el tercer paso, se identifican los posibles impulsores de la gestión de Big Data industrial, clasificando los requisitos en función de su impacto en la arquitectura (función de negocio de la industria, análisis de Big Data, almacenamiento de Big Data e infraestructura).

El cuarto paso implica elegir un concepto de diseño que satisfaga los impulsores arquitectónicos. Se adopta una arquitectura en capas para cumplir con los requisitos de los atributos de análisis de Big Data industriales (capa de infraestructura, capa de monitoreo, capa de presentación).

En el paso cinco, se crean instancias de elementos de la arquitectura y se asignan responsabilidades (componentes de la capa de infraestructura, componentes de la capa de supervisión, componentes de la capa de presentación).

En el paso seis, las interfaces se definen para los elementos instanciados a través de la implementación del componente a través de las herramientas adecuadas al Big Data industrial (Apache Spark, Apache Hive, Hadoop, Apache Kafka, Spark Streaming, Grafana y bibliotecas de Python).

El paso siete incluye verificar y refinar los requisitos y convertirlos en restricciones para los elementos de arquitectura instanciados (integración de fuentes de datos desde iCPS, procesamiento de datos escalable y elástico, composición de eventos basados en datos, servicios de optimización de datos, análisis integrado, soporte de decisiones basado en análisis). Finalmente, se repite según sea necesario.

3.2 Modelo de aprendizaje no supervisado para la detección temprana de fallas en la industria

En un entorno de producción, para minimizar el costo de mantenimiento, a veces es necesario crear un modelo con datos históricos mínimos o nulos. En tales casos, el aprendizaje no supervisado sería una mejor opción para la construcción de modelos ya que permite el diagnóstico inteligente de fallas menos dependiente de conocimientos previos y experiencia en el diagnóstico del proceso estudiado. Es decir, los desafíos del aprendizaje supervisado, como la necesidad de datos históricos y la incapacidad de clasificar nuevas fallas con precisión, podrían ser superados con una nueva metodología que utiliza aprendizaje no supervisado para una implementación rápida de la actividad de mantenimiento que incluye la predicción de fallas y la detección de clases de fallas para fallas conocidas y desconocidas.

Este enfoque podría ser importante en situaciones en las que no se cuenta con un conocimiento previo, como por ejemplo de las máquinas de un proceso a analizar,

en tales casos el análisis de datos con métodos no supervisados podría ayudar en la comprensión del estado de la máquina. Bajo este enfoque sería deseable hacer un diagnóstico inteligente de fallas menos dependiente de conocimientos previos del proceso y experiencia diagnóstica al procesar macrodatos.

3.2.1 Selección de los detectores base con conjunto de datos validados

Comparar el rendimiento de los algoritmos de detección de anomalías no supervisados no es tan sencillo como lo es en el caso clásico de clasificación supervisada. En contraste con simplemente comparar un valor de precisión o precisión/recuperación, el orden de las anomalías debe tomarse en cuenta.

Aunque la detección de anomalías no supervisada no utiliza ninguna información de etiqueta en la práctica, son necesarias en una etapa inicial para la evaluación y comparación. Es una práctica común que se modifique un conjunto de datos de clasificación pública disponible para determinar el rendimiento del método de detección de anomalías en relación con otros posibles.

En el caso de los métodos de clasificación, una instancia clasificada erróneamente es un error, lo que podría llegar a considerarse un pobre resultado del predictor, sin embargo, esto es diferente en la detección de anomalías no supervisadas. Por ejemplo, si se considera un conjunto de datos extenso y este contiene diez anomalías, pero se clasifican entre 15 valores atípicos, se podría considerar un buen resultado, tomando en cuenta que está lejos de ser perfecto.

Considerando lo anterior, una estrategia de evaluación común para los algoritmos de detección de anomalías no supervisados es clasificar los resultados de acuerdo con la puntuación de anomalía y luego aplicar iterativamente un umbral del primero hasta el último rango.

Con el fin de facilitar esta tarea, y como una aportación original a la solución del problema se presenta la **Figura 24**, con la finalidad de determinar los algoritmos base

que puedan ser utilizados en el desarrollo de un algoritmo que permita evaluar algoritmos no supervisado y determinar cuál desarrolla el mejor modelo de predicción de anomalías en conjuntos de datos, en los cuales no se cuenta con etiqueta verdad. Para esto, se desarrolló un algoritmo que evalúa los métodos no supervisados existentes en conjuntos de datos modificados para incluir la etiqueta de verdad, para comparar su rendimiento en la detección de anomalías.

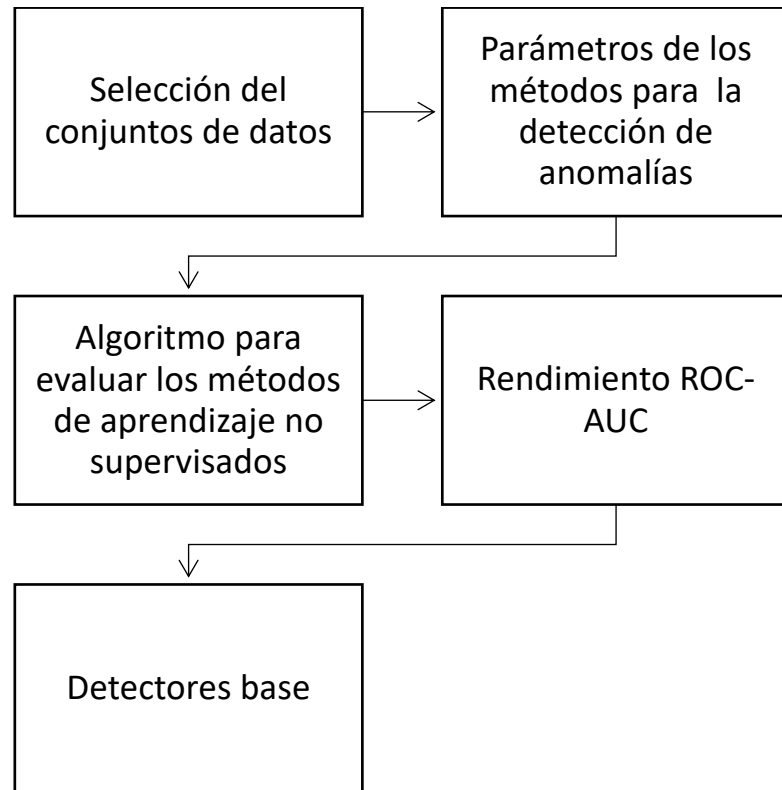


Figura 24. Metodología para la selección de los detectores base.

Selección del conjunto de datos

Los repositorios proporcionan acceso a una gran colección de conjuntos de datos para probar los métodos de detección de valores atípicos. Todos los conjuntos de datos que se muestran en la **Tabla 9** son reales y pueden ser encontrados en el repositorio ODDS [111]. Con esta selección de conjuntos de datos, se cubre un amplio espectro de dominios de aplicación, incluidas aplicaciones médicas, reconocimiento de imágenes y voz, así como el análisis de sistemas complejos.

Además, los conjuntos de datos cubren una amplia gama de propiedades con respecto al tamaño del conjunto de datos, el porcentaje atípico y la dimensionalidad.

Tabla 9. Conjuntos de datos disponibles para probar el desempeño de métodos no supervisados.

Datos	Descripción
arrhythmia	Distinguir entre la presencia y la ausencia de arritmia cardíaca
cardio	El conjunto de datos consiste en mediciones de la frecuencia cardíaca fetal (FHR) y las características de la contracción uterina (UC) en cardiogramas clasificados por obstetra experto
letter	Base de datos de entidades de imagen de caracteres; trata de identificar la carta
lympho	Este dominio de la linfografía se obtuvo del Centro Médico Universitario, Instituto de Oncología, Liubliana, Yugoslavia. (Acceso restringido)
mnist	Dígitos escritos a mano
musk	El objetivo es aprender a predecir si las nuevas moléculas serán almizcles o no almizcles
optdigits	Reconocimiento óptico del conjunto de datos de dígitos escritos a mano
pendigits	Reconocimiento basado en lápiz del conjunto de datos de dígitos escritos a mano
pima	Conjunto de datos sobre la diabetes de los indios Pima
satellite	Valores multiespectrales en vecindades de píxeles 3x3 en una imagen de satélite, y la clasificación asociada con el píxel central en cada vecindad
satimage-2	Valores multiespectrales en vecindades de píxeles 3x3 en una imagen de satélite, y la clasificación asociada con el píxel central en cada vecindad
shuttle	El conjunto de datos de transporte contiene 9 atributos, todos los cuales son numéricos. Aproximadamente el 80% de los datos pertenecen a la clase 1
vertebral	Conjunto de datos que contiene valores para seis características biomecánicas utilizadas para clasificar a pacientes ortopédicos en 3 clases (normal, hernia de disco o espondilolistesis) o 2 clases (normales o anormales)
vowels	Conjunto de datos de las vocales japonesas
wbc	Conjunto de datos de Cáncer de mama de Wisconsin (Diagnóstico)

Hiperparámetros de los métodos para la detección de anomalías

Como se mencionó previamente, comparar el rendimiento de los algoritmos de detección de anomalías no supervisados no es tan sencillo como lo es en el caso clásico de clasificación supervisada, ya que la selección de un algoritmo para la detección de anomalías adecuado para un caso particular es un proceso complejo.

La **Tabla 10** proporciona una descripción de los métodos usados como detectores de anomalías. Todos los modelos y parámetros se basan en la caja de herramientas

de detección de valores atípicos de Python (PyOD), los parámetros fueron adaptados de [112].

Tabla 10. Métodos para la detección de anomalías

Detector	Descripción
ABOD	Detección de valores extremos basados en ángulos
CBLOF	Factor de valor atípico local basado en agrupación
FB	Característica de embolsado
HBOS	Puntuación de valores atípicos basados en histogramas
kNN	k Vecinos más cercanos
LOF	Factor local atípico
MCD	Determinante de covarianza mínima
OCSVM	Máquinas de vectores de soporte de una clase
PCA	Análisis de componentes principales.

Además de los parámetros estadísticos presentados en la **Tabla 11**, se utilizaron los parámetros de referencia para calcular los modelos de los algoritmos FB, MCD, PCA.

Tabla 11. Parámetros usados para la selección de algoritmos base, adaptado de [112].

Método	Parámetro 1	Parámetro 2	Total
ABOD	n_vecinos: [3,5,10,15,20,25,50,60,70,80,90,100]	N/A	12
HBOS	n_contenedores: [10,20,30,40,50,75,100,150,200]	Tolerancia: [0.1,0.2,0.3,0.4,0.5]	40
KNN	n_vecinos: [1,5,10,15,20,25,50,60,70,80,90,100]	Método: ["largest", "mean", "median"]	36
LOF	n_vecinos: [1,5,10,15,20,25,50,60,70,80,90,100]	Método: ["manhatan", "euclidian", "minkowki"]	36
CBLOF	n_vecinos: [1,5,10,15,20,25,50,60,70,80,90,100]	N/A	36
OCSVM	nu (train error tol: [0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9])	Núcleo: ["Linear", "poly", "rbf", "sigmoid"]	36

ABOD: Detección de valores extremos basados en ángulos, HBOS: Puntuación de valores atípicos basados en histogramas, kNN: k Vecinos más cercanos, LOF: Factor local atípico, CBLOF: factor de valor atípico local basado en agrupación, OCSVM: Máquinas de vectores de soporte de una clase.

A continuación, se describen los parámetros utilizados:

- `n_vecinos`: Número de vecinos a utilizar de forma predeterminada para las consultas de `k` vecinos.
- `n_contenedores`: Número de contenedores que formará el histograma.
- Tolerancia: El parámetro para decidir la flexibilidad al tratar las muestras que caen fuera de los contenedores
- Método `k` Vecinos más cercanos: 'largest': usa la distancia al vecino `k`th como la puntuación atípica – 'mean': usa el promedio de todos los vecinos `k` como la puntuación atípica – 'median': usa la mediana de la distancia a `k` vecinos como la puntuación atípica.
- Método Factor local atípico y Método de factor de valor atípico local basado en agrupación: métrica utilizada para el cálculo de distancia.
- Núcleo: Especifica el tipo de núcleo que se utilizará en el algoritmo.

Algoritmo para evaluar los métodos de aprendizaje no supervisados

Algoritmo: Selección de los detectores base

Entradas: la lista de detectores D , conjunto de datos DS , anomalías y

Salidas: Puntuación ROC $score_roc$, Puntuación tiempo $score_time$, DF tabla de conjunto de datos

```

1: Inicializa la lista de detectores de anomalías  $D$ 
2:
3: for dataset in  $DS$  do
4:   Construye  $X_{train}$ ,  $X_{test}$ ,  $y_{train}$ ,  $y_{test}$  con 60% de los datos para entrenamiento y 40% para pruebas
5:   Estandariza los datos para procesamiento  $X_{train\_norm}$ ,  $X_{test\_norm}$ 
6:   for cada detector  $D_c$  in  $D$  do
7:     Obtén la predicción de anomalía
            $D_c$ .ajusta ( $X_{train\_norm}$ )
           puntaje_prueba =  $D_c$ . función_decisión ( $X_{test\_norm}$ )
8:     Obtén los puntajes  $t$ , ROC
            $t$  = duración( $t_1-t_0$ )
           ROC = puntaje_roc ( $y_{test}$ , puntaje_prueba)
9:   endfor
10:  Selecciona el mejor puntaje del dataset
           Min(tiempo)
           Max (ROC)
11:  A cada tabla  $DF$ 
            $DF_t$ .Agrega (Min  $t$ )
            $DF_{ROC}$ .Agrega (Max ROC)
12: endfor

```

El algoritmo para la selección de los algoritmos base, permite comparar el desempeño de los diferentes predictores con distintas opciones de hiper parámetros, al evaluar el conjunto de modelos M, con base en el puntaje obtenido por clasificar los resultados de acuerdo con la puntuación de anomalía y luego aplicar iterativamente un umbral del primero hasta el último rango.

Esto da como resultado valores de tupla N (tasa positiva real y tasa de falsos positivos), que forman una característica de un solo operador receptor (ROC, por las siglas en inglés de Receiver Operating Characteristic). A continuación, el área debajo de la curva (AUC, por las siglas en inglés de Area Under The Curve), la integral del ROC puede ser utilizado como una medida de rendimiento de detección.

En la **Figura 25** se muestra el AUC, y el ROC que es una representación gráfica de la sensibilidad frente a la especificidad, que puede ser utilizada como una medida de rendimiento de detección. El AUC es la probabilidad de que un algoritmo de detección de anomalías asigne a una instancia normal elegida aleatoriamente una puntuación inferior a una instancia anómala elegida aleatoriamente.

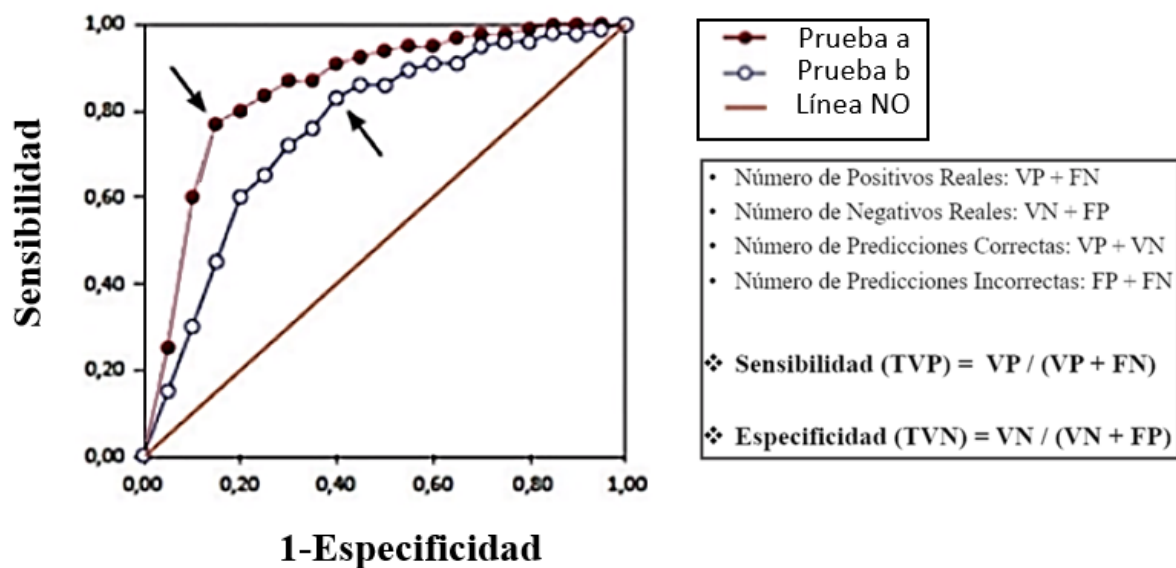


Figura 25. Curvas ROC, adaptado de [113].

En la **Tabla 12**, se presentan el resultado en valores de tupla N (tasa positiva real y tasa de falsos positivos), que forman una característica de un solo ROC.

Tabla 12. Matriz de confusión, adaptado de [113]

Actual / predicción	Es Positivo	Es Negativo
Ha dado positiva	Verdadero Positivo (VP) Predicción correcta del positivo Era positivo y la predicción ha sido positiva.	Falso Positivo (FP) Predicción incorrecta del positivo Era Negativo y la predicción ha sido positiva.
Ha dado Negativa	Falso Negativo (FN) Predicción incorrecta del negativo Era positivo y la predicción ha sido negativa.	Verdadero Negativo (VN) Predicción correcta del negativo Era negativo y la predicción ha sido negativa.

Rendimiento ROC-AUC

Para determinar los algoritmos base, se desarrolló el siguiente experimento. Para cada conjunto de datos, primero se divide en 60% para el entrenamiento y 40% para la validación. Todos los experimentos se repiten 10 veces de forma independiente con divisiones aleatorias.

Tabla 13. Características de hardware.

Especificación	Valor
Plataforma	PC
SO	Microsoft Windows 10 home
CPU	Intel i5-7400 @ 3.00GHz
RAM	8GB
Software	Jupyter 5.7.8
Python	Python 3.6.2
Core	Se utilizó uno sólo de sus cuatro núcleos (sin paralelización) debido a la gestión del sistema

El rendimiento se evalúa tomando el área bajo curva de la característica de funcionamiento del receptor (ROC-AUC) [113]. [113]. En la **Tabla 13** se proporciona la especificación de hardware y software usados en el experimento.

La **Tabla 14** resume las puntuaciones ROC-AUC de los 15 conjuntos de datos. Se obtiene la ejecución de cada uno de los algoritmos en diferentes configuraciones y se compara el ROC de los diferentes detectores. Esto proporciona el mejor rendimiento posible del algoritmo para un conjunto de datos específico.

Cada columna muestra el mejor desempeño del algoritmo para cada uno de los quince conjuntos de datos. Se muestran diferentes tonalidades de verde, rojo y amarillo en las celdas de cada columna indicando respectivamente: el mejor desempeño, desempeño aceptable y bajo desempeño.

Tabla 14. Rendimiento ROC-AUC.

Conjunto de datos	# Mue	# Dim	% Anom	ABOD	CBLOF	FB	HBOS	KNN	LOF	MCD	OCSVM	PCA
arrhythmia	452	274	14.6	0.77	0.78	0.78	0.85	0.78	0.78	0.82	0.8	0.8
cardio	1831	21	9.61	0.59	0.97	0.64	0.84	0.73	0.59	0.81	0.95	0.96
letter	1600	32	6.25	0.83	0.49	0.85	0.6	0.84	0.84	0.8	0.62	0.55
lympho	148	18	4.05	0.91	0.98	0.97	1	0.97	0.97	0.97	0.97	0.99
mnist	7603	100	9.21	0.78	0.89	0.69	0.58	0.85	0.69	0.89	0.86	0.86
musk	3062	166	3.17	0.17	1	0.44	1	0.83	0.44	1	1	1
optdigits	5216	64	2.88	0.48	0.52	0.5	0.89	0.4	0.51	0.37	0.49	0.51
pendigits	6870	16	2.27	0.68	0.97	0.51	0.92	0.75	0.51	0.84	0.92	0.94
pima	768	8	34.9	0.67	0.74	0.61	0.71	0.69	0.62	0.67	0.61	0.63
satellite	6435	36	31.64	0.58	0.53	0.57	0.77	0.69	0.57	0.81	0.68	0.62
satimage-2	5803	36	1.22	0.84	0.95	0.48	0.97	0.96	0.47	1	0.99	0.96
shuttle	49097	9	7.15	0.62	0.63	0.54	0.99	0.64	0.53	0.99	0.99	0.99
vertebral	240	6	12.5	0.37	0.31	0.37	0.3	0.39	0.37	0.38	0.43	0.39
vowels	1456	12	3.43	0.94	0.59	0.94	0.67	0.96	0.94	0.72	0.77	0.61
Wbc	378	30	5.56	0.81	0.89	0.9	0.9	0.89	0.89	0.82	0.9	0.86

ABOD: Detección de valores extremos basados en ángulos, MCD: Determinante de covarianza mínima, CBLOF: factor de valor atípico local basado en agrupación, FB: Característica de embolsado, HBOS: Puntuación de valores atípicos basados en histogramas, kNN: k Vecinos más cercanos, LOF: Factor local atípico, MCD: Determinante de covarianza mínima, OCSVM: Máquinas de vectores de soporte de una clase, PCA: El análisis de componentes principales.

Para ejemplificar lo anterior, considérese la **Figura 26** que muestra el ROC-AUC para el método OSCV aplicado a los diferentes conjuntos de datos, donde es posible observar las diferencias en el desempeño de acuerdo con los diferentes parámetros en el mismo conjunto de datos.

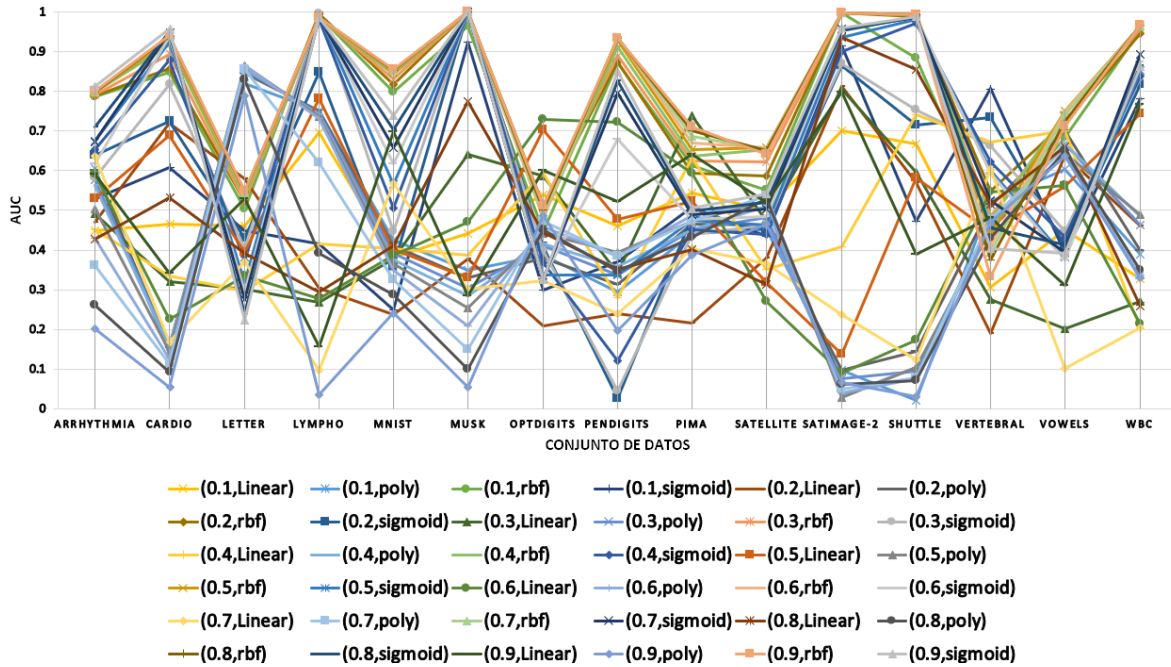


Figura 26. ROC-AUC para el método OSCV con hiperparámetros, aplicado a diferentes conjuntos de datos.

Más aun, se puede ver cómo el algoritmo con un conjunto de parámetros, por ejemplo (0.1, Linear), muestra desempeños diferentes en las puntuaciones del ROC-AUC a través de los conjuntos de datos.

El ranking de la lista detectores base de anomalías queda de la siguiente forma: En primer lugar, lo obtuvo el detector HBOS, y en segundo lugar tanto CBLOF y OCSVM, mientras que ABOD, KNN, y LOF y mostraron el peor desempeño.

Por lo anterior, los métodos HBOS, CBLOF y OCSVM son seleccionados como algoritmos base, y son utilizados en la implementación del algoritmo para la selección del mejor modelo no supervisado para predecir anomalías que se muestra a continuación.

3.2.2 Modelo de aprendizaje no supervisado para la detección temprana de fallas

El proceso de modelado en algunos problemas como la detección de anomalía suele ser un proceso inherentemente subjetivo, donde la función objetiva o el modelo definido para un problema determinado depende de la comprensión del comportamiento de los datos [114].

Por ejemplo, el algoritmo del vecino más cercano para anomalías podría proporcionar resultados muy diferentes al algoritmo de una máquina vectorial de soporte de una clase, debido a las diferencias subyacentes en las asunciones que estos modelos tienen. Por otro lado, el modelo seleccionado puede ser extremadamente sensible a la elección de parámetros utilizados en la detección de anomalías.

Todos estos problemas hacen que la evaluación del desempeño de los algoritmos de detección de valores atípicos no supervisados sea más difícil de llevar a cabo que los algoritmos supervisados, y en ausencia de la etiqueta de verdad, existe incertidumbre sobre la verdadera eficacia en la selección de un algoritmo [69][69].

A diferencia de la clasificación, la mayoría de los algoritmos de detección de valores atípicos generan puntuaciones de valor "atípico" para los puntos de datos. Se puede considerar el problema de detección de valores atípicos como una tarea de clasificación binaria que tiene una clase mayoritaria (valores típicos) y una clase minoritaria (valores atípicos) convirtiendo las puntuaciones de atípico en etiquetas de clase [115].

Los puntos con puntuaciones por encima de un umbral se consideran valores atípicos con la etiqueta 1 (etiqueta 0 para los valores inferiores al umbral). Después de convertir el problema de detección de valores atípicos no supervisado en una tarea de clasificación con sólo etiquetas no observadas, se puede explicar la compensación de variación de sesgo para la detección de valores atípicos utilizando ideas de clasificación.

Específicamente, el error esperado de detección de valores atípicos se puede dividir en dos componentes principales: error reducible y error irreducible (es decir, error debido al ruido). El error reducible se puede minimizar para maximizar la precisión del detector. Además, el error reducible puede descomponerse en (1) error debido al sesgo cuadrado y (2) error debido a la varianza.

A pesar de la naturaleza no supervisada del análisis de conjuntos de valores atípicos, en [114] los autores muestran que los fundamentos teóricos del análisis y la clasificación de valores atípicos son sorprendentemente similares. Para una mayor discusión de la teoría de los conjuntos de clasificación se puede consultar [116].

Considérese una instancia de datos denotada por \bar{X}_i , para la que la puntuación de valores atípicos se modela utilizando los datos de entrenamiento \mathcal{D} . Asíumase que todos los puntos de entrenamiento de datos son generados por una misma distribución base. La puntuación ideal se obtiene por una función desconocida $f(\bar{X}_i)$ y se asume que las puntuaciones generadas por esta función ideal también satisfacen la media cero y la suposición de varianza unitaria sobre todos los puntos posibles generados por la distribución de datos base [114]:

$$y_i = f(\bar{X}_i) \quad (1)$$

Dado que se desconoce el modelo verdadero $f(\cdot)$, la puntuación atípica de un punto de prueba \bar{X}_i sólo se puede estimar con el uso de un modelo de detección de valores atípicos $g(\bar{X}_i, \mathcal{D})$ utilizando el conjunto de datos base \mathcal{D} . Por ejemplo, en los detectores de valores atípicos de los vecinos más cercanos, la función $g(\bar{X}_i, \mathcal{D})$ se define de la siguiente manera:

$$g(\bar{X}_i, \mathcal{D}) = \alpha KNN - distance(\bar{X}_i, \mathcal{D}) + \beta \quad (2)$$

Donde se tiene que, α y β son constantes que son necesarias para estandarizar las puntuaciones de la media a cero y la varianza a la unidad, con el fin de respetar la restricción en la interpretación absoluta de las puntuaciones atípicas. Por otro lado, la función $g(\bar{X}_i, \mathcal{D})$ no modela correctamente la verdadera función $f(\bar{X}_i)$, por lo que se generan errores. Esto se conoce como sesgo de modelo.

Una segunda fuente de error es la varianza. La varianza es causada por el hecho de que la puntuación de valores atípicos depende directamente de la creación de instancias específica del conjunto de datos \mathcal{D} . Ahora, sea \mathcal{D} los datos de entrenamiento, y $\bar{X}_1 \dots \bar{X}_n$ un conjunto de n puntos de prueba, cuyas puntuaciones de valores atípicos (hipotéticamente ideales, pero no observados) son $y_1 \dots y_n$, utilizamos un algoritmo de detección de valores atípicos no supervisado que utiliza la función $g(\cdot)$ para estimar estas puntuaciones. Por lo tanto, las puntuaciones resultantes de $\bar{X}_1 \dots \bar{X}_n$ utilizando los datos de entrenamiento \mathcal{D} son $g(\bar{X}_1, \mathcal{D}) \dots g(\bar{X}_n, \mathcal{D})$, respectivamente.

El error medio cuadrado, o EMC, de los detectores de los puntos de prueba sobre una realización particular \mathcal{D} de los datos de entrenamiento se obtiene mediante el promedio de los errores al cuadrado en diferentes puntos de prueba:

$$E[EMC] = \frac{1}{n} \sum_{i=1}^n \{y_i - g(\bar{X}_i, \mathcal{D})\}^2 \quad (3)$$

El EMC esperado, sobre diferentes realizaciones de los datos de entrenamiento, generados usando algún proceso aleatorio, es el siguiente:

$$E[EMC] = \frac{1}{n} \sum_{i=1}^n E[\{y_i - g(\bar{X}_i, \mathcal{D})\}^2] \quad (4)$$

El término en el corchete en el lado derecho de (la Ecuación 4) se puede volver a escribir de la siguiente manera:

$$E[EMC] = \frac{1}{n} \sum_{i=1}^n E[\{y_i - f(\bar{X}_i) + f(\bar{X}_i) - g(\bar{X}_i, \mathcal{D})\}^2] \quad (5)$$

Se puede mostrar lo siguiente:

$$E[EMC] = \frac{1}{n} \sum_{i=1}^n E[\{f(\bar{X}_i) - g(\bar{X}_i, \mathcal{D})\}^2] \quad (6)$$

Este lado derecho se puede descomponer aún más añadiendo y restando $E[g(\bar{X}_i, \mathcal{D})]$ dentro del término cuadrado:

$$E[EMC] = \frac{1}{n} \sum_{i=1}^n E[\{f(\bar{X}_i) - E[g(\bar{X}_i, \mathcal{D})]\}^2] + \frac{1}{n} \sum_{i=1}^n \{f(\bar{X}_i) - E[g(\bar{X}_i, \mathcal{D})]\} \{E[g(\bar{X}_i, \mathcal{D})] - E[g(\bar{X}_i, \mathcal{D})]\} + \frac{1}{n} \sum_{i=1}^n E[\{E[g(\bar{X}_i, \mathcal{D})] - g(\bar{X}_i, \mathcal{D})\}^2] \quad (7)$$

Se tiene:

$$\begin{aligned} E[EMC] &= \frac{1}{n} \sum_{i=1}^n E[\{f(\bar{X}_i) - E[g(\bar{X}_i, \mathcal{D})]\}^2] + \frac{1}{n} \sum_{i=1}^n E[\{E[g(\bar{X}_i, \mathcal{D})] - g(\bar{X}_i, \mathcal{D})\}^2] \\ &= \frac{1}{n} \sum_{i=1}^n \{f(\bar{X}_i) - E[g(\bar{X}_i, \mathcal{D})]\}^2 + \frac{1}{n} \sum_{i=1}^n E[\{E[g(\bar{X}_i, \mathcal{D})] - g(\bar{X}_i, \mathcal{D})\}^2] \end{aligned} \quad (8)$$

El primer término de la expresión antes mencionada es el sesgo (cuadrado), mientras que el segundo término es la varianza. Dicho simplemente, se obtiene lo siguiente:

$$E[EMC] = \text{Sesgo}^2 + \text{Varianza} \quad (9)$$

Algoritmo: Selección del modelo no supervisado para predecir anomalías

Para abordar el problema de encontrar el modelo de predicción no supervisado para la detección de anomalías y como una aportación original al desarrollo del problema, se propone un algoritmo para la selección del modelo para predecir posibles anomalías. El algoritmo propuesta crea diferentes modelos a través de diferentes parámetros dando lugar a conjuntos de modelos, dentro de este conjunto, el modelo con la menor varianza y sesgo se considera el más competente, luego este

es comparado con otros algoritmos hasta obtener el modelo con el mejor desempeño global.

Algoritmo: Modelo no supervisado para predecir anomalías

Entrada: Lista de detectores base B , Lista de parámetros P , conjunto de datos D

Salida: TS, RB (para cada algoritmo)

- 1: **Inicializa** la lista de detectores base B
- 2: **Permite** a la serie temporal TS , conjunto de parámetros P , y a M el conjunto de modelos creados por los detectores base, ser una columna del conjunto de datos D
- 3: **Inicializa** TS como una serie temporal en D .
- 4: **Construye** la serie de tiempo TS
- 5: **for** cada B **do**
- 6: **Inicializa** los parámetros P del algoritmo B
- 7: **for** cada P **do**
- 8: **Identifica** el rango min, max
- 9: **Construye** una lista i con valores aleatorios de $[min, max]$
- 11: **for** each i **do**
- 12: **Aplica** el algoritmo B a TS con el parámetro i para construir el modelo M
- 13: **Obtén** la predicción de $M(i)$
- 14: **Obtén** los puntajes estandarizados $S(i)$
- 15: **Obtén** el error medio cuadrado EMC ($S(i)$)
- 16: **end for**
- 17: **Selecciona** el **Mínimo** EMC (P)
- 18: **end for**
- 19: **Selecciona** el **Mínimo** EMC (*puntajes estandarizados* B)
- 20: **end for**
- 21: **Selecciona** el **Mínimo** MES (B)
- 22: **Muestra** el mejor modelo de B

Sea $D \in R^{n \times d}$ denota los datos con n puntos y d características, el algoritmo primero genera un grupo de detectores base $B = \{B_1, \dots, B_r\}$, que se inicializan con un rango de hiper parámetros P . Todos los detectores base se entrenan y a continuación, la inferencia se realiza en el mismo conjunto de datos D en el vector de modelos $M(D)$. Los resultados se integran en una matriz de puntajes de anomalías $M(D), [B_1(D), \dots, B_r(D)] \in R_{n \times R}$ donde $B_r(\cdot)$ denota el vector de puntuación del r^{th} detector base.

Cada detector de puntuación $B_r(D)$ se normaliza mediante la *normalización Z* [114]. El error medio cuadrado (*EMC*) mide la competencia de cada detector base. El detector B_r con la menor varianza y sesgo se considera el detector más competente.

El conjunto de modelos (M) se compone de algoritmos de detección base B : HBOS, CBLOF, OCSVM con distintas opciones de hiperparámetros P .

La **Tabla 15** proporciona una descripción de los parámetros de los modelos.

Tabla 15. Parámetros de los modelos.

Método	Parámetro 1	Parámetro 2	Total
HBOS	n_histograma: [10,20,30,40,50,75,100,150,200]	tolerancia: [0.1,0.2,0.3,0.4,0.5]	40
CBLOF	n_vecninos: [1,5,10,15,20,25,50,60,70,80,90,100]	método: ["manhatan", "euclidian", "minkowki"]	36
OCSVM	error de tol: [0.1,0.2,0.3,0.4,0.5,0.6,0.7,0.8,0.9]	Nucleo: ["Linear", "poly", "rbf", "sigmoid"]	36

HBOS: Puntuación de valores atípicos basados en histogramas, CBLOF: factor de valor atípico local basado en agrupación, OCSVM: Máquinas de vectores de soporte de una clase.

3.3 Conclusiones

En este capítulo se presentó la metodología para el diseño de una arquitectura de referencia que puede ser aplicada en diferentes escenarios para la gestión de datos para la analítica inteligente en la industria y la metodología para el modelo basado en datos para la detección temprana de fallas.

La metodología para el diseño de una arquitectura de referencia ha sido desarrollada considerando el enfoque de diseño basado en atributos (ADD) y consta de siete etapas: (1) Ciclo de vida del Big Data industrial, (2) Ciclo de vida de fabricación, (3) Impulsores de la gestión industrial de Big Data, (4) Arquitectura en capas, (5) Arquitectura de gestión de datos, (6) Implementación de componentes, (7) Atributos de Big Data industrial. El uso de este enfoque general permitió el planteamiento de requerimientos funcionales y no funcionales para el soporte del desarrollo de aplicaciones de analítica de Big Data industrial.

Además, esta metodología desarrolló una arquitectura de referencia con un diseño integral que le permite adaptarse a los requerimientos particulares de los diferentes escenarios de los sistemas ciberfísicos en la Industria 4.0.

También en este capítulos se desarrolló un método para la selección del modelo basado en datos para la detección temprana de fallas. Para este propósito se desarrolló en primer lugar un algoritmo que evaluó métodos no supervisados para determinar el conjunto de detectores base, utilizando para ello un conjunto de datos modificados para incluir la etiqueta de verdad y diferentes opciones de parámetros para cada predictor de anomalías para evaluar su rendimiento en un conjunto de modelos utilizando la puntuación obtenida por el ROC-AUC.

Posteriormente se propuso un algoritmo para determinar el modelo con el mejor desempeño al predecir anomalías en un conjunto de datos que no cuenta con etiqueta de verdad. Cada predictor definido por el algoritmo crea diferentes modelos del mismo tipo, pero con diferentes parámetros dando lugar a conjuntos de modelos, dentro de este conjunto, el modelo con la menor varianza y sesgo se considera el más competente, luego este es comparado con otros algoritmos hasta obtener el modelo con el mejor desempeño global.

Capítulo 4

Resultados Experimentales

Este capítulo presenta los elementos que constituyen la arquitectura de referencia para analítica de Big Data Industrial y su validación a través de tres escenarios de fallas industriales: (1) Diagnóstico de fallas, (2) monitoreo en tiempo real y (3) pronóstico de condiciones inusuales. También, se valida el método basado en datos para la selección del mejor modelo de aprendizaje para la detección temprana de fallas, con un caso práctico que incluye un conjunto de datos de mediciones de sensores y señales de referencia de control para cada uno de varios componentes de control de una planta industrial y mediciones de energía eléctrica de diferentes zonas de la planta. Finalmente se presenta una segunda validación de la arquitectura de referencia a través de la implementación de una instancia de la arquitectura de referencia en un caso de uso que aplica un modelo basado en datos en un prototipo de plataforma para la detección temprana de fallas.

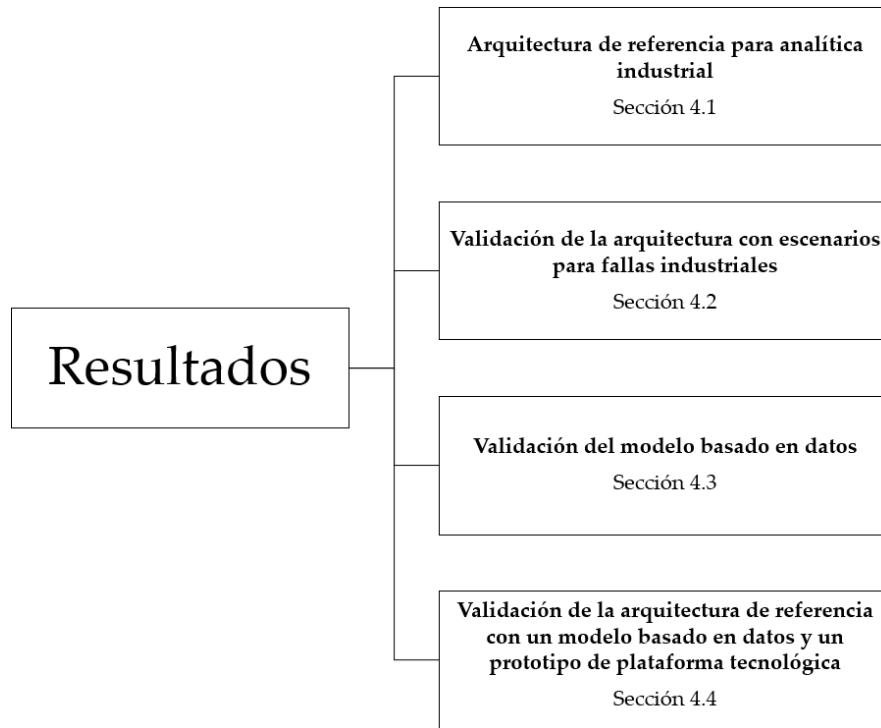


Figura 27. Organización de la presentación de los resultados.

En la **Figura 27** se presenta la organización de este capítulo. En la sección 4.1 se presenta la arquitectura de referencia desarrollada para apoyar el diseño de soluciones de analítica de Big Data industrial. Este aporte original facilita el diseño y la implementación de soluciones basadas en software con un enfoque en gestión de datos para analítica industrial.

En la sección 4.2 se valida la arquitectura de referencia presentada. Se muestran diferentes escenarios de fallas en la industria, a través de aplicaciones reales tomadas de la literatura, para ilustrar diferentes casos de uso en contextos industriales que muestran la aplicabilidad de la arquitectura. También, se revisa una aplicación para detallar las interacciones entre los componentes de la arquitectura propuestos y el caso de uso revisado.

En la sección 4.3 se valida el modelo basado en datos propuesto para obtener el mejor modelo para detectar en forma temprana posibles fallas. Este algoritmo fue evaluado en un conjunto de datos de mediciones de sensores y señales de referencia

de control para cada uno de varios componentes de control de la planta y mediciones de energía eléctrica de diferentes zonas de la planta.

En la sección 4.4 se presenta una segunda validación de la arquitectura de referencia y el modelo de datos. Esta validación tiene relación con el objetivo general en lo referente al desarrollo de un prototipo de plataforma tecnológica.

Se implementa una instancia de la arquitectura de referencia para el escenario de fallas en la industria y se aplica el método basado en datos desarrollado para encontrar el mejor modelo de detección temprana de fallas, con el fin mostrar la integración de las dos principales aportaciones de este trabajo de tesis.

4.1 Arquitectura de referencia para analítica de Big Data en la industria

Para satisfacer los requerimientos para el diseño de aplicaciones de analítica industria que se han presentado en la sección 2.2 (Analítica de Big Data en la industria) y manejar una amplia gama de cargas de trabajo y casos de uso, las cuales requieren baja latencia de lectura y escritura, se ha adoptado la arquitectura Lambda [117] para el desarrollo de la arquitectura de referencia presentada en esta sección.

En los sistemas de analítica industrial los datos en tiempo real se refieren a los datos que se presentan a medida que se adquieren, a través de tecnologías que brinden información actualizada de acuerdo con una ventana temporal en aplicaciones de monitoreo de IIoT. En este tipo de aplicaciones de analítica industrial, los datos en tiempo real no se guardan ni almacenan para su procesamiento, sino que son monitoreados mientras son transmitidos al usuario final tan pronto como se recopilan, para posteriormente almacenarse para propósitos futuros. Es importante tener en cuenta que los datos en tiempo real no significan que los datos lleguen al usuario final al instante.

La arquitectura Lambda se muestra en la **Figura 28**, este tipo de arquitectura de Big data resuelve el problema de calcular en gran medida las funciones en los datos en tiempo real desglosando el problema en tres capas:

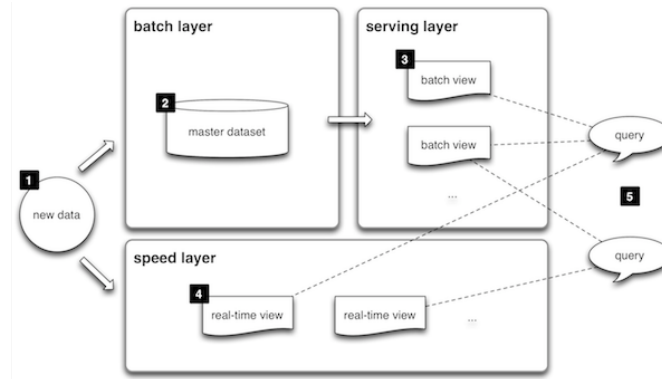


Figura 28. Arquitectura Lambda (Fuente: <http://lambda-architecture.net/>).

1. Una capa por lotes que administra un conjunto de datos maestros inmutable y solo calcula las funciones de consulta denominadas lotes de vistas,
2. Una capa de servicio que indexa las vistas por lotes para consultas ad hoc de baja latencia,
3. Y una capa de velocidad compensa la alta latencia de las actualizaciones a la capa de servicio y solo trata los datos recientes.

En la **Figura 29** se muestra la arquitectura de referencia para la administración de datos para iCPS que proporciona soporte para el análisis predictivo, la inferencia de los indicadores de rendimiento clave y el análisis en tiempo real. Este diseño arquitectónico se beneficia de los atributos de calidad del Big Data industrial presentados como propuesta original al estado del arte en la sección 2.2.3, a través de la identificación de sus elementos y el mapeo de las tecnologías de Big Data seleccionados.

1. El requisito de integración de múltiples fuentes se cumple en la capa de infraestructura con los siguientes componentes: Flujo de fuente de datos heterogéneo, flujo de datos de proceso y datos de recursos de fabricación.

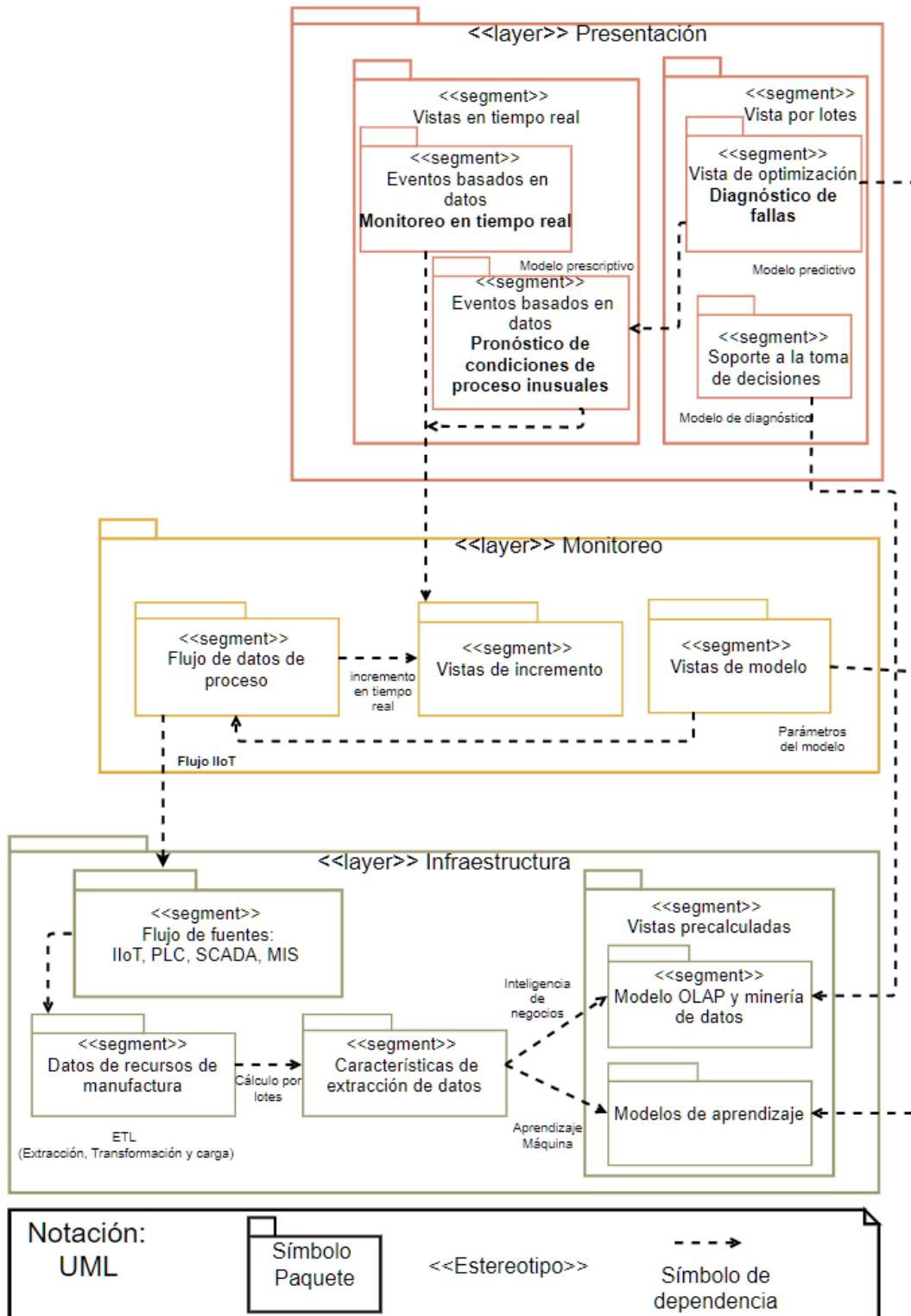


Figura 29. Arquitectura de referencia para la gestión de datos para iCPS.

2. El requisito de procesamiento de datos escalable y elástico se cumple en la capa de infraestructura con el componente de datos de recursos de fabricación.
3. La composición de eventos controlados por datos se cumple en la capa de monitoreo con los siguientes componentes: monitoreo en tiempo real, vistas incrementales y vista del modelo.
4. Los servicios de datos de optimización se realizan mediante el componente de vista incremental y el componente de vistas de modelo.
5. La analítica integrada y el soporte de decisiones basado en análisis se cumplen en las vistas previas al cálculo y en el componente de la vista por lotes.

4.1.1 Capa de infraestructura

La capa de infraestructura es la primera en explicarse. El componente de datos de recursos de fabricación almacena Big Data a partir de una entrada de datos de bajo volumen (por ejemplo, los datos provenientes de bases de datos transaccionales), o datos con alto volumen de entrada (por ejemplo, flujos de datos de sensores), como un conjunto histórico, sólo anexa datos sin procesar. A continuación, se obtiene una lista de características adecuadas para aplicar los algoritmos de aprendizaje de modelos respectivos mediante el componente de procesamiento de datos. Si hay cambios en los criterios, los datos se vuelven a procesar.

En el componente de modelos de aprendizaje, los parámetros que describen el modelo de datos dependen de los métodos de aprendizaje automático utilizados. La detección de anomalías incluiría agregados de sellos de tiempo de la operación de datos de fabricación. En el contexto del procesamiento de grandes conjuntos de

datos, es importante utilizar el procesamiento de datos distribuidos en función de los datos reales y el método de aprendizaje utilizado.

El componente modelo OLAP y minería de Big Data permite a través de los almacenes de Big Data el modelado de datos multidimensionales (MDM, por las siglas en inglés de Multi-Dimensional Data Modeling) para el análisis de datos de grandes volúmenes de datos estructurados para apoyar los procesos de toma de decisiones en un contexto de inteligencia empresarial.

4.1.2 *Capa de monitoreo*

La capa de monitoreo recibe un nuevo flujo de datos de IIoT, su función principal es analizar el flujo de datos entrante en tiempo real. Se ocupa del procesamiento de datos en tiempo real que suele depender del tiempo, por lo que es crucial reducir las latencias agregadas accediendo al modelo de aprendizaje guardado en la capa de presentación. El componente de flujo de proceso procesa datos de series temporales para obtener las características del nuevo flujo de datos. En el componente de vista de incremento, la medición continua a lo largo del tiempo representa una función importante en la identificación de valores atípicos de datos, se refiere al hecho de que los patrones no se supone que cambien abruptamente, excepto que ocurren procesos inusuales en los datos de trabajo, a continuación, transfiere la entrada del modelo obtenida al componente de monitoreo en tiempo real en la capa de presentación en el período de tiempo coincidente, con el resultado del modelo de aprendizaje y si el umbral es superado, y se detecta consecutivamente durante algún tiempo un evento basado en los datos, se presenta en la vista de tiempo real. Aunque pueden producirse errores del sensor u otras imprecisiones de datos para difundir un evento de anomalía. Por lo tanto, el evento de anomalía tiene lugar si sucede secuencialmente durante un tiempo específico.

4.1.3 *Capa de presentación*

La capa de presentación presenta una vista de la salida de los patrones de datos producidos por las funciones de la capa de infraestructura y de monitoreo a través de vistas en tiempo real y por lotes. Por lo tanto, el componente de monitoreo en tiempo real analiza la entrega de información actualizada continuamente para identificar anomalías en los datos agregados a lo largo del tiempo de las operaciones de fabricación desde el componente de vistas de incremento, para identificar problemas graves utilizando los parámetros del modelo de aprendizaje del modelo de detección de errores estimado en el componente de vista de modelo. Además, el componente de monitoreo de condiciones de proceso inusuales predice problemas de rendimiento del proceso analizando datos, seguimiento de patrones recurrentes, y detectando comportamientos anormales; utiliza vistas de optimización del modelo de análisis predictivo que se aplica a los eventos basados en datos para el monitoreo proactivo. En las vistas por lotes, el componente de diagnóstico de fallas presenta analítica predictiva para mejorar la detección de anomalías y el diagnóstico de los sistemas de fabricación, con las vistas de optimización de los equipos de proceso y los equipos de instalaciones, basados en métodos de minería de Big Data implementados en el componente de modelos de aprendizaje. El componente de soporte de decisiones basado en análisis utiliza vistas por lotes para admitir el procesamiento analítico en línea (OLAP), análisis para informar, observar y mostrar cuán grande o pequeño es el problema utilizando el almacén de Big Data de la capa de infraestructura.

4.1.4 *Despliegue de componentes*

Hoy en día, se están desarrollando diferentes tecnologías para el análisis de Big Data, a menudo superponiendo requisitos funcionales y atributos de gestión de datos de calidad. Para los arquitectos de Big Data de sistemas iCPS, la selección de

tecnología es una cuestión desafiante que requiere la atención de muchos detalles de implementación y limitaciones de compatibilidad. La **Figura 30** muestra tecnologías a las que se puede acceder con una licencia de código abierto.




Capas arquitectónicas	Atributos industriales de Big Data		Componentes arquitectónicos	Tecnologías de código abierto
Presentación		Servicios de optimización de datos Apoyo a la toma de decisiones basadas en análisis	<ul style="list-style-type: none"> Diagnóstico de fallas Monitoreo en tiempo real previsión de condiciones inusuales del proceso Apoyo a la toma de decisiones basada en análisis 	<ul style="list-style-type: none"> Herramientas de desarrollo Presentación de información
Monitoreo		Composición de eventos basados en datos	Flujo de datos de proceso	Procesamiento de flujo de datos
Infraestructura		Integración del origen de datos de iCPS Procesamiento de datos escalable y elástico	<ul style="list-style-type: none"> Flujo de fuente de datos heterogéneo Datos de recursos de fabricación Modelo OLAP, minería de datos Modelos de aprendizaje 	<ul style="list-style-type: none"> Servicios de mensajería, Integración de datos y transmisión de datos en directo Almacén de datos distribuidos Marco de procesamiento de datos distribuidos

Figura 30. Despliegue de tecnologías.

Una vez que los datos están disponibles a través del componente de extracción de características de datos se pueden utilizar para pre calcular vistas de aprendizaje automático o de inteligencia empresarial. Un almacén distribuido de Big Data a través de Hive habilitando el componente OLAP y Data Mining. Hive se desarrolló en HDFS para habilitar el almacenamiento y el procesamiento distribuidos para recopilar grandes conjuntos de datos. El componente del modelo de aprendizaje utiliza los datos almacenados en Hadoop para crear modelos de aprendizaje utilizando el marco de procesamiento de datos distribuido Apache Spark, un motor de análisis para Big Data que permite un mejor rendimiento utilizando la memoria de acceso aleatorio (RAM) del servidor, para procesar algoritmos de aprendizaje automático.

Para fines de monitoreo de datos, el componente de flujo de datos de proceso obtiene datos desde IIoT disponibles con Spark Streaming interactuando con

Apache Kafka, ofreciendo procesamiento de flujo de datos escalable, de alto rendimiento, tolerante a errores de procesamiento, para la composición de eventos basados en datos en vivo. A continuación, los datos se almacenan o insertan en el componente de eventos basado en datos para el análisis en tiempo real.

Para la visualización de datos, después de procesar datos para el uso analítico con herramientas de desarrollo, Grafana integra los datos para fines de presentación. El componente para la supervisión en tiempo real prepara datos para Grafana utilizando Spark Streaming para el procesamiento de eventos basados en datos. Además, el componente de apoyo a la toma de decisiones OLAP toma datos de Hive y produce salidas sobre Grafana. Finalmente, el componente de diagnóstico de fallas utiliza herramientas de desarrollo basadas en las bibliotecas Python para procesar algoritmos de aprendizaje automático a través de Big Data almacenado en Hadoop para preparar datos para Grafana.

4.2 Validación de la arquitectura con escenarios para fallas en la industria

Para validar la arquitectura de referencia propuesta para apoyar el diseño de soluciones de analítica de Big Data industrial, en esta sección se describen diferentes escenarios de fallas industriales tomados de la literatura, mostrando cómo se puede aplicar la arquitectura en el diseño de sistemas de analítica industrial que consideran el Big Data y el aprendizaje máquina.

Para ilustrar mejor la aplicabilidad en el mundo real de la arquitectura propuesta, se eligió un conjunto de casos reales de fallas industriales extraídos de la literatura de investigación como un medio para mostrar cómo los componentes de la arquitectura encajan con los elementos principales propuestos para brindar soluciones a situaciones reales.

4.2.1 Evaluación de la arquitectura de referencia

La evaluación arquitectónica basada en escenarios es un enfoque bien establecido para validar el diseño arquitectónico y analizar las decisiones que se han tomado para lograr el enfoque de diseño [118]. Los escenarios son enfoques completos e integrales que reúnen a las partes interesadas de un sistema y las guían a través de un proceso estructurado que explora las opciones de diseño arquitectónico y las implicaciones resultantes.

El propósito de los escenarios es mostrar la sensibilidad de la decisión arquitectónica y los puntos de compensación del diseño arquitectónico. También se describe las interacciones de los componentes de la arquitectura con la aplicación de un caso de uso revisado. Los escenarios brindan una visión del posible ajuste de las soluciones propuestas a los componentes de la arquitectura, considerando hipotéticamente las soluciones como si estuvieran implementadas siguiendo la arquitectura propuesta.

En la **Tabla 16** son presentados los escenarios de diagnóstico de fallas seleccionados para ilustrar cómo se aplica la arquitectura de referencia de analítica de Big Data para cumplir con los requisitos funcionales y requerimientos.

Tabla 16. Requerimientos de los escenarios de diagnóstico de fallas.

Requerimiento	Atributo de Big Data industrial	Escenario
Proporcionar analítica para iCPS con la capacidad de comparación, en donde los registros de rendimiento de la maquinaria se pueden comparar y clasificar entre máquinas.	Servicios de optimización de datos	Diagnóstico de fallas
Proporcionar un diagnóstico de fallas de la maquinaria basado en información sobre las similitudes de los registros de rendimiento de la máquina.		
Desplazamiento, desviación, valores atípicos en componentes subyacentes. El sensor virtual está funcionando y detecta cambios, desviaciones y valores atípicos.	Composición de eventos basados en datos	Monitoreo en tiempo real
Causas raíz entre parámetros. ¿Qué parámetros específicos contribuyen más al problema?		

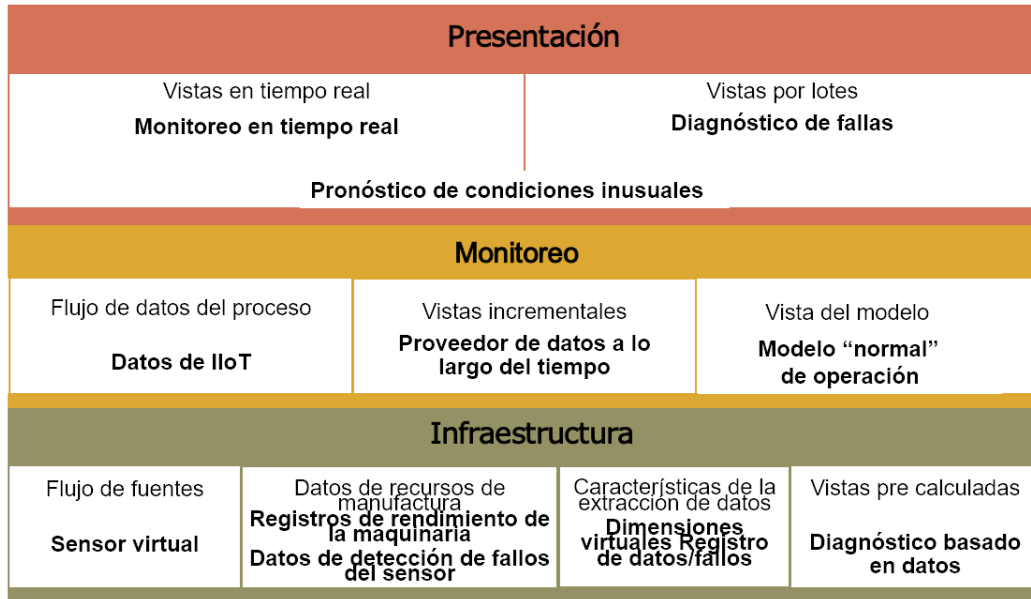


Figura 31. Instancia arquitectónica de Big Data de analítica industrial para el caso de uso de diagnóstico de fallas.

Para los escenarios de diagnóstico de fallas en plantas industriales anteriormente presentados, se muestran las funcionalidades y características de la implementación de los casos de uso en una instancia creada a partir de la arquitectónica de referencia, como se muestra en la **Figura 31**.

Estas funcionalidades y características están agrupadas en dominios funcionales. Esta instancia presenta tres dominios funcionales: diagnóstico de fallas, monitoreo en tiempo real y pronóstico de condiciones inusuales que están relacionado con los escenarios de aplicación. Cada dominio funcional agrupa un conjunto de funciones comunes. La implementación de los componentes de cada dominio funcional se relaciona con la instancia y la arquitectura de referencia de la siguiente forma:

1. Dominio funcional del escenario de diagnóstico de fallas: muestra los beneficios y cumple el requerimiento denominado servicios de datos de optimización. El componente de *vistas por lotes* realiza el diagnóstico de fallas con los datos que provee el componente de *recursos de manufactura* que

almacena los registros de rendimiento de la maquinaria y los datos de detección de fallas del sensor. El componente de *características de extracción de datos* prepara los registros de datos de fallos para el diagnóstico basado en datos.

2. Monitoreo en tiempo real en iCPS: muestra los beneficios y cumple con el requerimiento denominado composición de eventos controlados por datos. El componente *flujo de datos del proceso* provee los datos del IIoT al componente de *vistas incrementales* que provee los datos a lo largo del tiempo para su monitoreo a través del componente *vistas en tiempo real* en busca de posibles inconsistencias en base al componente *vistas del modelo* que utiliza el modelo de operación “normal”.
3. Pronóstico de condiciones de proceso inusuales: muestra los beneficios y cumple con los requerimientos de servicios de datos de optimización y eventos controlados por datos.

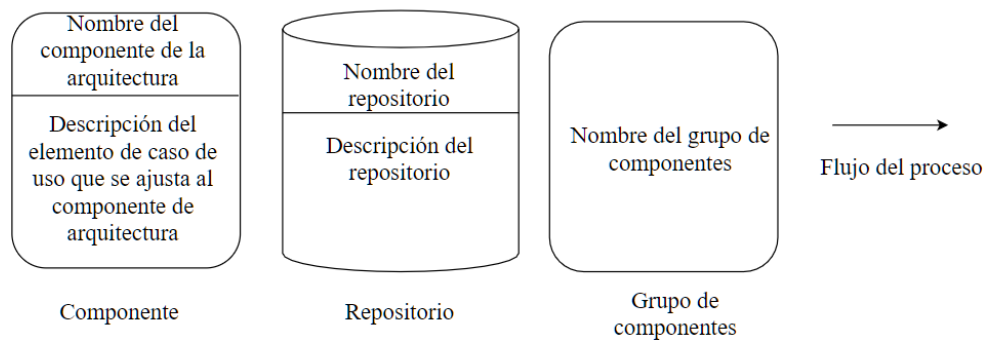


Figura 32. Iconos de notación de los diagramas de vista de proceso.

La **Figura 32** muestra la notación propuesta para representar componentes de arquitectura, grupos de componentes, repositorios y flujos de procesos. Estos iconos de notación se utilizan en los diferentes diagramas de la vista del proceso, para describir las interacciones entre los componentes de la arquitectura y los elementos que se ajustan a los componentes de la arquitectura con los elementos del caso de

uso que se presenta más adelante. El icono del componente describe los elementos que se ajustan a un componente de arquitectura para el caso de uso presentado. El icono del grupo de componentes representa varios componentes relacionados entre sí. El icono del repositorio representa el área de datos, modelos o vistas. Y finalmente, la relación de flujo de proceso describe la secuencia del diagrama de vista del proceso.

4.2.2 Diagnóstico de fallas

Un diagnóstico basado en datos es un enfoque común que se puede aplicar en diferentes contextos industriales para el diagnóstico de fallas. Esto se puede ilustrar con los siguientes ejemplos en escenarios reales: en [119] se presenta un enfoque de diagnóstico inteligente de fallas para un problema mecánico utilizando Big Data y aprendizaje de características no supervisado para proporcionar una predicción precisa en el caso de registros de cojinetes de motor. En [120] se presenta un ejemplo adicional de diagnóstico de fallas basado en datos, relacionado con la industria del petróleo, donde se observó que el mantenimiento regular no puede detectar de manera efectiva fallas en los compresores alternativos. En dicho trabajo, se aplicaron métodos de aprendizaje automático para analizar datos y diagnosticar fallas para predecir posibles fallas en los compresores.

La adecuación de la arquitectura de referencia al proceso de mecanizado con soporte Big Data y *machine learning* se describe siguiendo la vista de proceso. La vista de proceso describe las interacciones entre los componentes de la arquitectura para la analítica industrial y las actividades que realizan la gestión de Big Data durante su ciclo de vida. La **Figura 33** muestra la instancia de la arquitectura del proceso de detección de fallas con registros históricos.

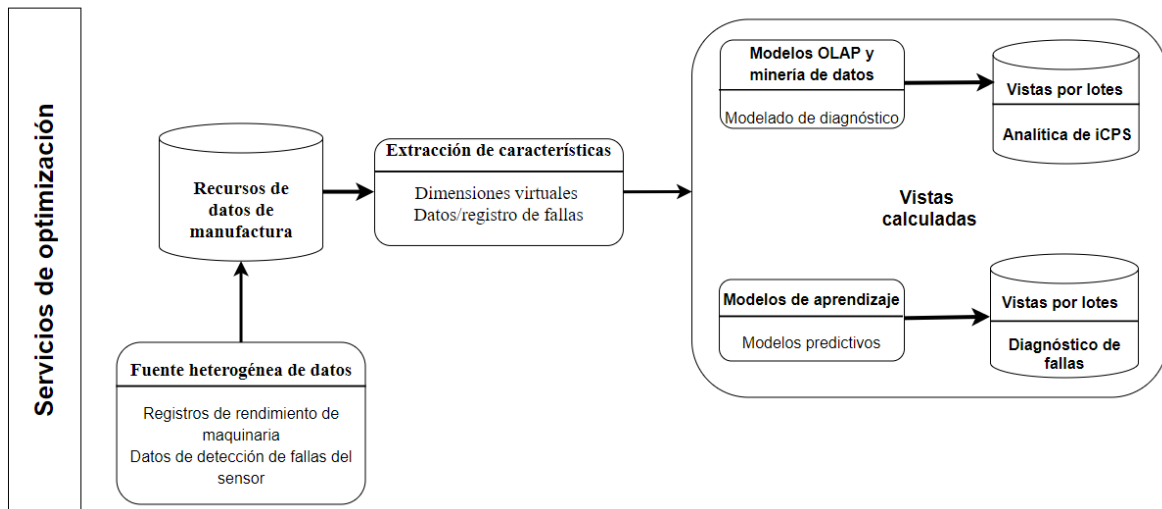


Figura 33. Instancia del proceso de diagnóstico de fallas.

El componente de la arquitectura *características de extracción* categoriza los datos mediante dimensiones virtuales que capturan los registros de los datos del proceso. Tienen las tareas de filtrar, limpiar y agregar los datos. El componente *vistas pre calculadas* realiza la extracción de información a través del modelado de diagnóstico o utilizando el modelado predictivo. El componente *modelos OLAP y minería de datos* permite el análisis de datos de los registros de rendimiento de la maquinaria y los datos de detección de fallas del sensor para obtener información sobre el estado de salud de la maquinaria presente y futuro. El componente *modelos de aprendizaje*, mediante el uso de algoritmos de aprendizaje automático permite predecir tendencias y detectar patrones.

Para ilustrar una de las muchas aplicaciones posibles de la arquitectura propuesta para el análisis de Big Data en la industria para escenarios de diagnóstico de fallas, a continuación, se analiza un caso real para el problema de automatización del proceso de mecanizado, donde se analiza un sistema de monitoreo efectivo del estado de la máquina. En [121] se presenta un enfoque de aprendizaje automático para condiciones defectuosas basado en datos del proceso de corte para controlar el proceso de mecanizado, alimentando datos optimizados al controlador de la

máquina. El sistema tiene como objetivos alargar la vida útil de la máquina herramienta y una calidad de procesamiento para garantizar una alta productividad. La **Figura 34** muestra las instancias de los componentes de la arquitectura que interactuarían durante un escenario de proceso de diagnóstico de fallas.

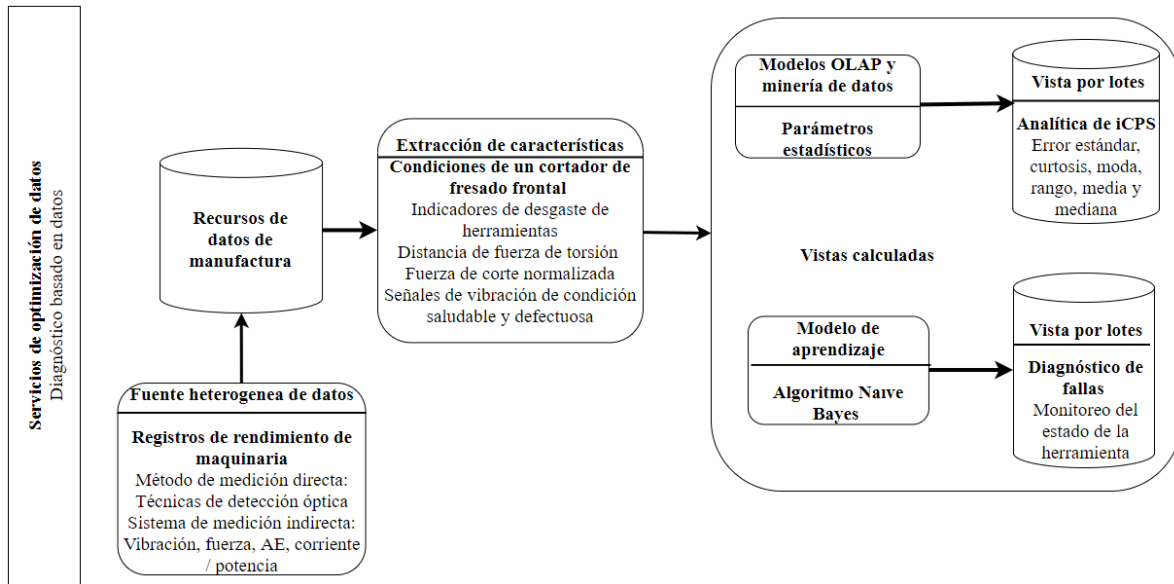


Figura 34. Instancia del proceso de diagnóstico de fallas (caso de uso reportado en [121]).

La correspondencia entre el escenario descrito en [121], y los componentes de la arquitectura es el siguiente: en el componente *datos de recursos de fabricación* de la arquitectura, se almacenan los datos históricos provenientes del componente *fuentes heterogéneas del flujo de datos*. Los datos almacenados incluyen registros de rendimiento de la maquinaria del método de medición. El proceso de extracción de información se realiza en el componente de la arquitectura *extracción de las características* mediante dimensiones virtuales que capturan las condiciones de la fresa frontal de maquinado y el monitoreo del registro de fallas de datos. El componente de *vistas calculadas* realiza la extracción de información a través del modelado de diagnóstico o utilizando el modelado predictivo. El componente *modelos OLAP y minería de datos* analiza los datos utilizando parámetros estadísticos

sobre los registros de rendimiento de la maquinaria y los datos de detección de fallas del sensor para obtener información sobre el pasado y el presente. Por otro lado, el uso del algoritmo Naïve Bayes en el componente del modelo de aprendizaje permite el monitoreo de la condición de la herramienta para el diagnóstico de fallas en el componente de *vistas por lotes*.

4.2.3 Monitoreo en tiempo real

La detección de anomalías en el análisis en tiempo real para iCPS ha permitido una nueva forma de optimizar los sistemas industriales apoyando a analistas y operadores para resolver posibles problemas [31]. En el escenario de la composición de eventos basados en datos, el monitoreo de los datos en tiempo real apoya al Mantenimiento preventivo activo, Reparación y Revisión (MRO, por las siglas en inglés Maintenance, Repair, and Overhaul) a evitar la pérdida de maquinaria y reducir los daños. La detección de anomalías en el análisis en tiempo real para iCPS ha permitido una nueva forma de optimizar los sistemas industriales apoyando a analistas y operadores para resolver posibles problemas [31]. En el escenario de la composición de eventos basados en datos, el monitoreo de los datos en tiempo real apoya al Mantenimiento preventivo activo, Reparación y Revisión (MRO) a evitar la pérdida de maquinaria y reducir los daños.

La **Figura 35** muestra una instancia del proceso de monitoreo de flujo de datos para la detección de fallas. El componente de *vistas incrementales* utiliza una ventana temporal para realizar operaciones sobre los datos. Esa operación está destinada a limpiar los datos, reducir la frecuencia de los datos y extraer variables. Cuando se completan las tareas, se seleccionan las variables de datos para enviarlas al subproceso de operaciones del modelo normal para su evaluación.

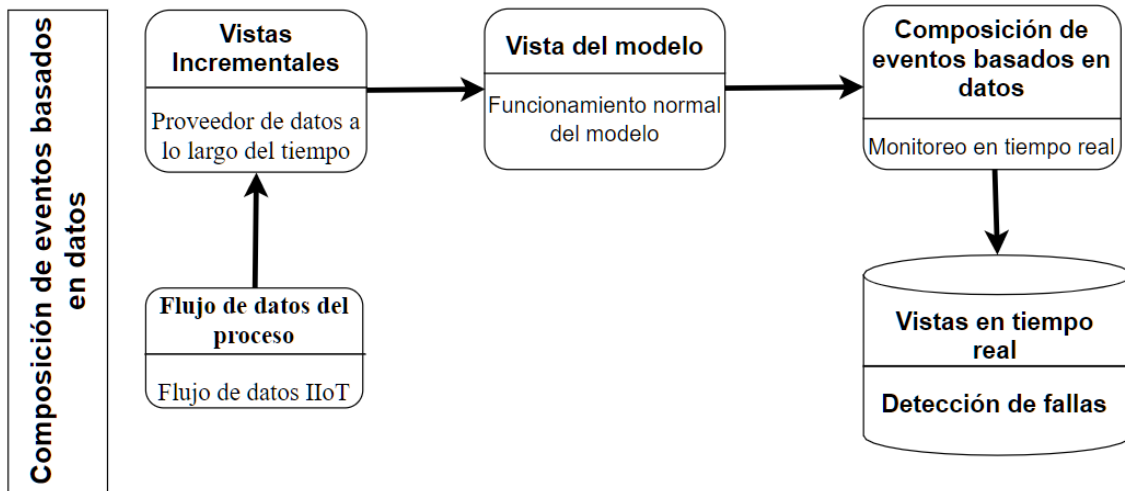


Figura 35. Instancia del proceso de monitoreo en tiempo real para la detección de fallas.

Para monitorear el *flujo de datos del proceso*, el componente *vistas del modelo* utiliza un modelo de operaciones normales que aplica un modelo de aprendizaje para dimensiones virtuales, que determina si los datos de proceso son "buenos" y están bajo control. El componente *composición de eventos basados en datos* detecta posibles fallas, y utiliza los modelos de predicción generados por la vista del modelo para detectar patrones o predecir estados en tiempo real. El monitoreo en tiempo real evalúa los datos enviados y devuelve una predicción. Los métodos de aprendizaje no supervisados no necesitan datos etiquetados y detectan eventos raros o valores atípicos como datos muy distintos de la mayoría de los datos basados en la identificación de nuevos tipos de anomalías como análisis de comportamiento de anomalías [57].

El monitoreo en tiempo real en una etapa temprana busca evitar la pérdida de maquinaria y reducir los daños. En [122] se propone una técnica basada en modelos que utiliza el aprendizaje automático para identificar fallas internas de las máquinas de inducción en tiempo real. La detección de fallas en el dispositivo podría tener solo manifestaciones menores, pero resultar en una menor eficiencia de operación. En ese caso, el monitoreo de la condición de la máquina para inferir advertencias

puede ser útil en reducir fallas internas de máquinas industriales. En [123] se propone un sistema de monitoreo de fallas en tiempo real para industrias que busca prevenir daños severos a las máquinas, basado en el método de aprendizaje “Random Forest”. El monitoreo en tiempo real en una etapa temprana busca evitar la pérdida de maquinaria y reducir los daños. En [122] se propone una técnica basada en modelos que utiliza el aprendizaje automático para identificar fallas internas de las máquinas de inducción en tiempo real. La detección de fallas en el dispositivo podría tener solo manifestaciones menores pero resultar en una menor eficiencia de operación. En ese caso, el monitoreo de la condición de la máquina para inferir advertencias puede ser útil en reducir fallas internas de máquinas industriales. En [123] se propone un sistema de monitoreo de fallas en tiempo real para industrias que busca prevenir daños severos a las máquinas, basado en el método de aprendizaje “Random Forest”.

Para ilustrar una de las muchas aplicaciones posibles de la arquitectura propuesta para el análisis de Big Data industrial en escenarios de monitoreo de equipos en tiempo real, a continuación, se analiza el siguiente caso real tomado de la literatura. En [124], se presenta un sistema de aprendizaje autónomo que propone un método para la detección de fallas en tiempo real en procesos industriales, basados en TEDA (por sus siglas en inglés Typicality and Eccentricity Data Analytics). análisis de datos de tipicidad y excentricidad). Las principales ventajas del enfoque TEDA son que no requiere un conocimiento a priori sobre la transmisión de datos, siendo esto de particular importancia en aplicaciones del mundo real; Además, TEDA es muy rápido en línea, lo que permite su uso para la detección de fallas en la industria.

Para validar el escenario para el monitoreo de fallas en tiempo real usando TEDA para problemas industriales, los autores usaron datos reales de plantas industriales. El conjunto de datos abierto DAMADICS (por sus siglas Desarrollo y aplicación de

métodos para el diagnóstico de actuadores en inglés Development and Application of Methods for Actuator Diagnosis in Industrial Control Systems (sistemas de control industrial) se utilizó junto con una planta piloto de laboratorio. Los datos de DAMADICS provienen de un actuador del proceso de evaporación del agua, que consta de los siguientes componentes: válvula de control, servomotor neumático y posicionador. La planta de datos proviene de varios sensores y actuadores controlados por un controlador lógico programable (PLC) que automatiza el flujo entre dos tanques, dos válvulas de control neumáticas y una bomba centrífuga.

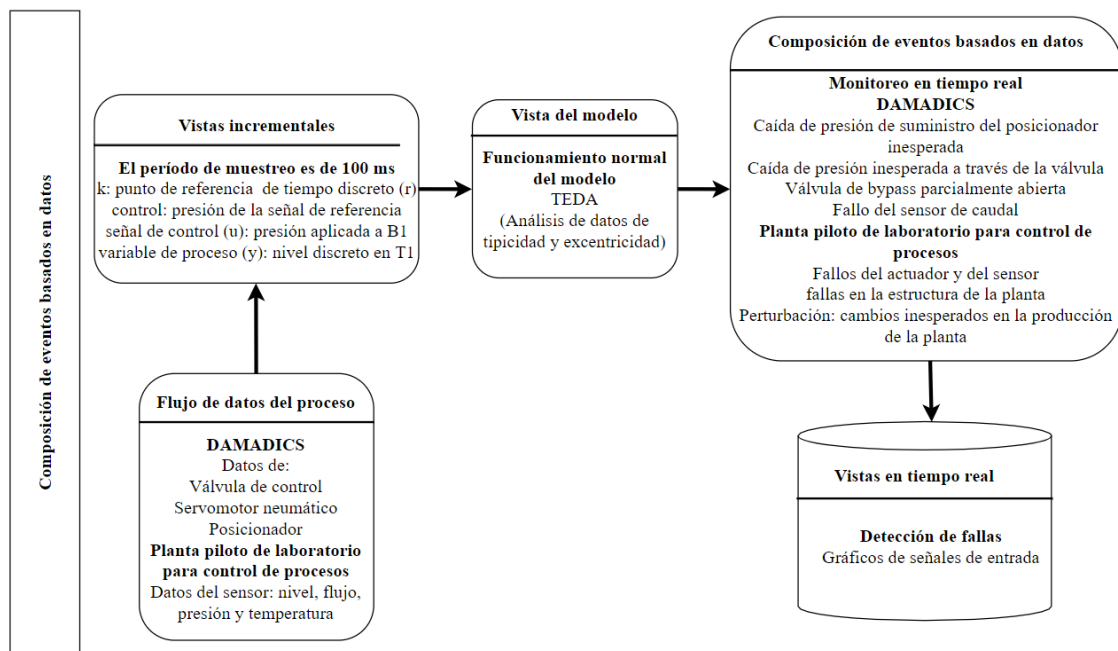


Figura 36. Instancia del proceso de monitoreo en tiempo real para la detección de falla (Vista del caso de uso reportado en [124]).

La vista del proceso de la **Figura 36** describe las interacciones entre los componentes de la arquitectura que realizan la gestión del flujo de datos para el análisis industrial y el caso de uso para las actividades de los equipos de monitoreo en tiempo real.

Los datos se procesan en tiempo real desde DAMADICS y la planta piloto del laboratorio para el control del proceso a través del componente de arquitectura del *flujo de datos del proceso*. En el componente de *vistas incrementales*, se procesan los datos temporalmente utilizando una ventana de tiempo para realizar operaciones en los datos.

Los datos de la planta piloto se recogen en un período de muestreo de 100 ms. Esa operación está destinada a limpiar los datos, reducir la frecuencia de los datos y extraer variables. Cada flujo de datos tiene variables monitoreadas que se muestran en el componente de *vistas incrementales*. Cuando se completan las tareas, se seleccionan las variables de datos para enviarlas al subproceso de operaciones del modelo normal para su evaluación.

En el componente *vistas de modelo*, las operaciones normales del modelo se utilizan para aplicar un modelo de aprendizaje multivariante para las dimensiones virtuales subyacentes que capturan los datos del proceso "bueno" y en control. El algoritmo TEDA no necesita ningún conocimiento previo de los procesos de datos para aprender a detectar anomalías. Los métodos de aprendizaje no supervisados no necesitan datos etiquetados y son capaces de detectar eventos raros o valores atípicos como datos muy distintos de la mayoría de los datos, basados en la identificación de nuevos tipos de anomalías. El proceso de detección y predicción utiliza los modelos predictivos generados durante el proceso de extracción de información para identificar patrones o predecir estados en tiempo real. El componente de la arquitectura *composición de eventos basados en datos* monitorea en tiempo real los datos enviados y devuelve una predicción para un evento en DAMADICS o en la planta piloto del laboratorio para el control del proceso. Este subproceso utiliza el modelo generado por el proceso de extracción de información para evaluar los datos en tiempo real. El componente de arquitectura de las vistas en tiempo real muestra gráficos de señales de entrada para la detección de fallas.

4.2.4 Pronóstico de condiciones de proceso inusuales

El escenario para pronosticar condiciones de proceso inusuales combina servicios de datos de optimización y requerimientos basados en datos para una evaluación de salud adaptativa. La **Figura 37** muestra una instancia para pronosticar procesos inusuales.

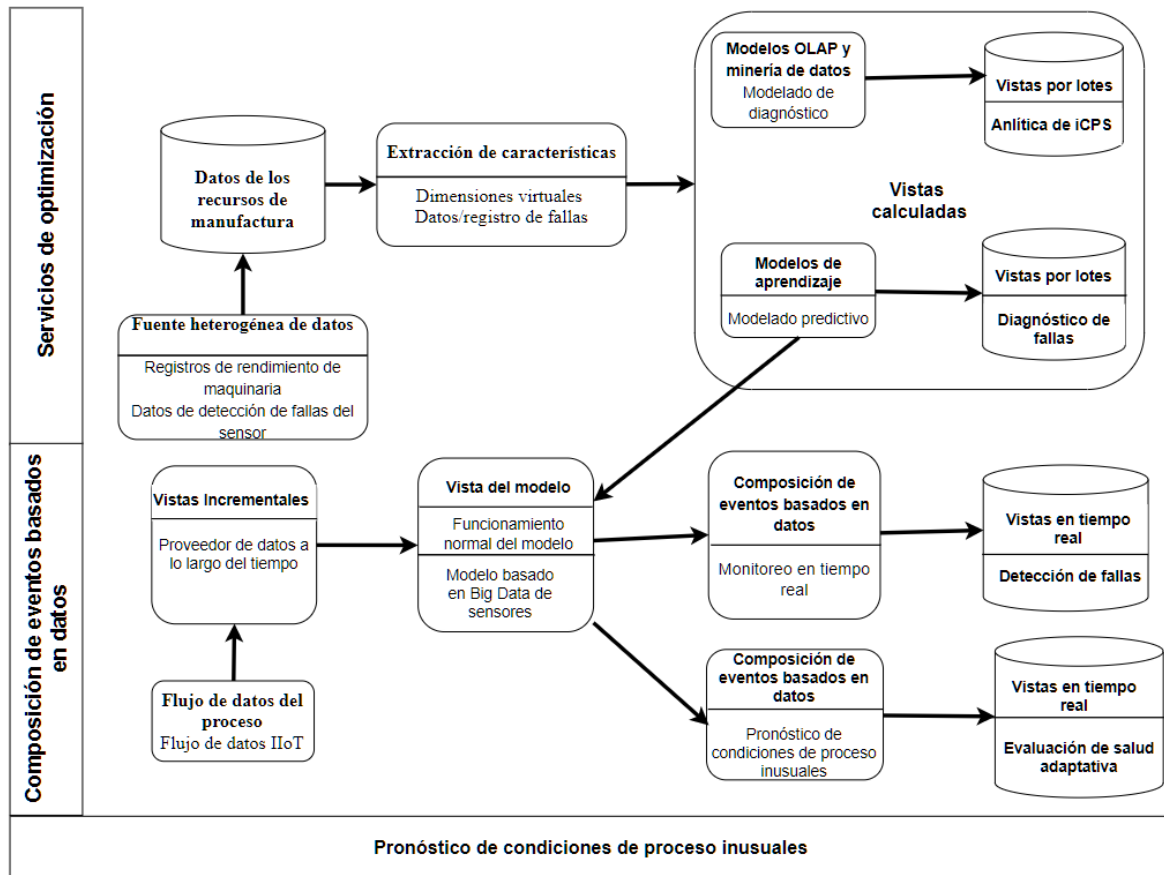


Figura 37. Instancia para el pronóstico de condiciones inusuales.

La premisa fundamental es desarrollar un modelo predictivo basado en el registro de datos del sensor y aplicar el modelo a las medidas de los datos del flujo entrante de los sensores en tiempo real. El desafío del escenario para el diseño de aprendizaje de modelos es doble. Primero, desarrollar un modelo de aprendizaje basado en sensores y Big Data. El segundo es aplicar el modelo analítico a las medidas de flujo de los sensores.

El desafío de Big Data es enfrentar cargas intensivas de datos en tiempo real en el momento en que se calcula un modelo de aprendizaje automático. La detección de anomalías en iCPS requiere tanto el análisis de cantidades masivas de datos históricos como un procesamiento rápido basado en resultados intermedios (modelo de detección de anomalías) [34].

El modelo de aprendizaje para pronosticar condiciones de proceso inusuales aplica conjuntos de algoritmos para extraer características de datos de Big Data mediante conjuntos de datos más pequeños al respaldar el uso de la capa de monitoreo que procesa los métodos de aprendizaje automático en lotes.

Se podría usar una evaluación de salud adaptativa para diagnosticar la confiabilidad del sistema y pronosticar el estado de la máquina, o un sistema basado en información de monitoreo de salud. En [125] se propone una metodología genérica basada en métodos de aprendizaje automático para correlacionar fuertemente las fallas detectadas en los datos históricos de los registros del proceso con el flujo de datos entrante, de acuerdo con un horizonte de predicción. Los datos provienen del dominio del aluminio y representan el flujo de las diferentes fases y máquinas del Proceso de electrólisis del aluminio, y representan el flujo de las diferentes fases y máquinas para preparar la pasta y formar el ánodo (bloques de carbono que se utilizan para el proceso de reducción de aluminio). Hay dos categorías de datos: las señales de entrada se analizan para el horizonte de pronóstico, luego los datos de entrada se correlacionan con los datos del sistema interno para el diagnóstico de fallas. El enfoque general consta de dos partes: procesar y aplicar el modelo al proceso en curso para detectar valores atípicos en componentes subyacentes y dimensiones virtuales.

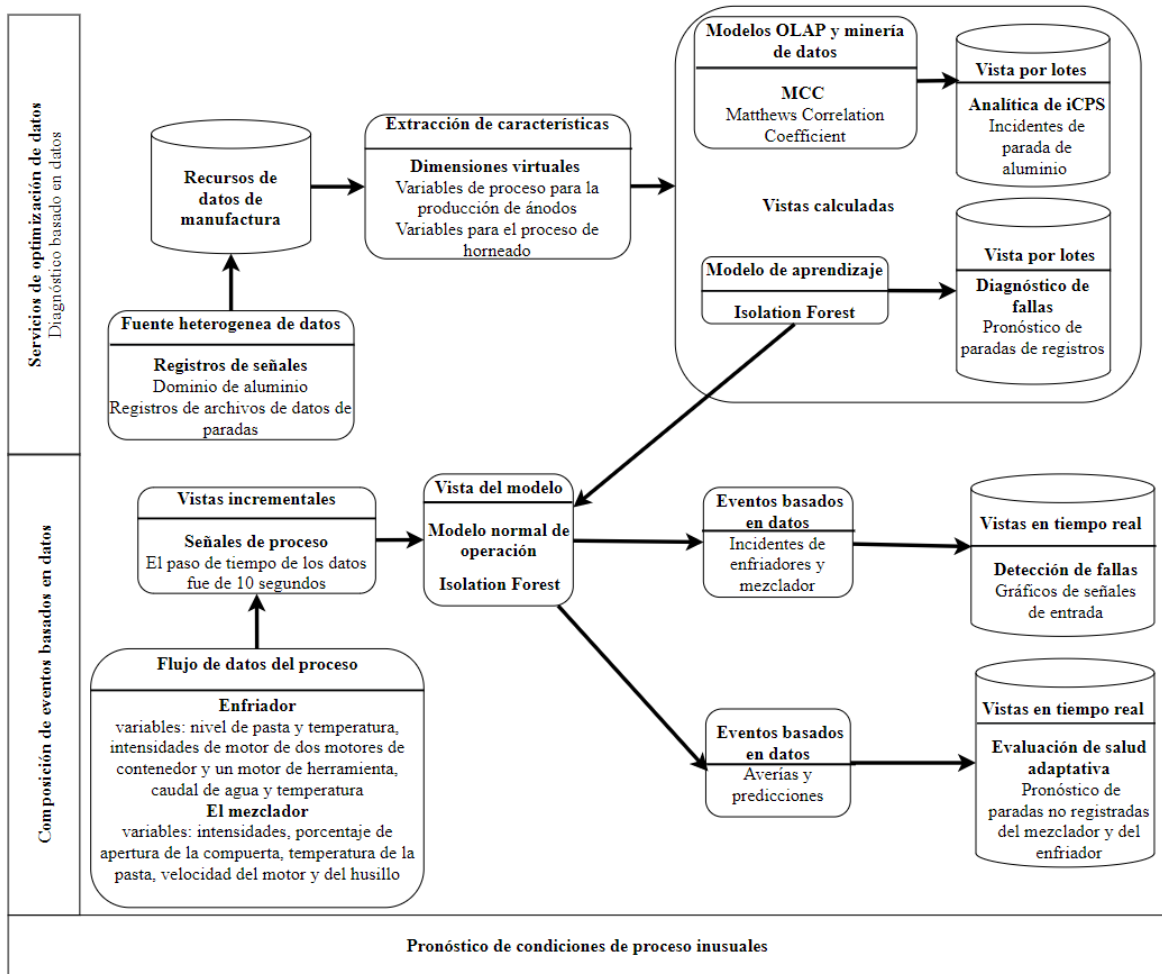


Figura 38. Instancia de procesos inusuales (Vista para el caso de uso reportado en [125]).

La Figura 38 muestra una instancia para pronosticar procesos inusuales, a través del diagnóstico basado en datos. El componente de arquitectura *recursos de datos de manufactura* almacena los archivos de datos de parada. El componente de arquitectura para la extracción de características de datos incluye variables de proceso para la producción de ánodos y variables para el proceso de horneado.

El uso del modelo de aprendizaje “Isolation Forest” en el componente *modelo de aprendizaje* permite pronosticar las paradas registradas para el diagnóstico de fallas en el componente de *vista de lotes*. Este modelo también se puede aplicar al componente de *vista del modelo* para monitorear un evento de ruptura en las señales de proceso provenientes del componente de *flujo de datos de proceso*. La fuente de

flujo de datos incluye variables de los dispositivos enfriadores y mezcladores. Eso permite una evaluación de salud adaptativa en el componente de arquitectura de *vistas en tiempo real* para el pronóstico de salud del enfriador, mezclador y registros de paradas.

Finalmente, también es posible que el componente *monitoreo de procesos en tiempo real* utilice operaciones de modelo normales para aplicar un modelo de aprendizaje que capture los datos de proceso "bueno" y en control para la detección de fallas.

4.2.5 Utilidad de la arquitectura de referencia para la investigación y la práctica

En esta sección se han presentado diferentes escenarios para una instancia de la arquitectura de referencia para analítica industrial con la finalidad de resolver las limitaciones en el desarrollo de soluciones a problemas que tienen características comunes, mostrándose la utilidad para aspectos prácticos y de investigación que aporta la arquitectura de referencia para analítica de Big Data en la Industria. Para conocer las ventajas que ésta tiene al estructurar soluciones de analítica industrial para el estado de salud de la maquinaria y procesos en la industria se presenta una discusión en las secciones 5.1 y 5.2 que aborda las implicaciones o beneficios para la investigación y la práctica.

4.3 Validación del modelo basado en datos

Esta sección presenta una validación del modelo basado en datos explicado en la sección 3.2, a través de un caso de uso que muestre un escenario para la detección de fallas, aplicado al flujo entrante de datos nuevos de los sensores.

El diagnóstico de fallas tiene un papel crítico en los sistemas de plantas industriales. Un sistema de diagnóstico de fallas robusto y preciso ayuda a prevenir accidentes fatales, ahorra costos y aumenta la eficiencia de la producción [126].[126]. El desarrollo de un sistema de diagnóstico de fallas de alto rendimiento para un sistema en particular requiere principalmente dos tipos de información: (1) una

comprensión profunda del sistema objetivo o (2) datos de monitoreo de condición / registro de fallas. Un amplio nivel de conocimiento sobre fallas del sistema (es decir, mecanismos, causas fundamentales) puede facilitar el diagnóstico efectivo de fallas para los sistemas de plantas industriales. Por otro lado, una cantidad significativa de monitoreo de las fallas a través de los datos de registro, si están disponibles, pueden proporcionar información excelente para el diagnóstico basado en datos (por ejemplo, analítica de Big Data). Desafortunadamente, tener un conocimiento profundo del sistema a optimizar es casi imposible en sistemas reales en plantas industriales, debido a que tales sistemas están compuestos de numerosos componentes y operan en una variedad de condiciones. Además, la mayoría de los datos disponibles contienen registros de fallas incompletos o faltantes debido a factores humanos o sistemas de monitoreo que proporcionan datos deficientes (por ejemplo, formato obsoleto).

4.3.1 Descripción del caso práctico

A continuación, se presenta un escenario de aplicación para el método propuesto para la selección del modelo no supervisado para la detección de anomalías.

El problema consiste en detectar fallas con datos reales, en la que los registros de fallas a menudo faltan o son incompletos. Este problema ha sido ampliado para considerar además aplicarlo al flujo entrante de datos nuevos de los sensores, al momento que se pueda detectar posibles fallas.

La sociedad de pronósticos y gestión de la salud (PHM, por sus siglas en inglés Prognostics and Health Management Society) abordó el tema del diagnóstico de fallas de planta industrial con sistemas con datos de registro de fallas incompletos en la Competencia de Desafío de Datos PHM 2015 [127]. Como se aprecia en la **Figura 39**, los datos representan: a) series temporales de mediciones de sensores y señales de referencia de control para cada uno de varios componentes de control de

la planta (por ejemplo, 6 componentes); (b) datos de series temporales que representen mediciones adicionales de un número fijo de zonas de la planta durante el mismo período de tiempo (por ejemplo, 3 zonas), donde una zona puede abarcar uno o más componentes de la planta; Cada planta se especifica a través de su número de componentes y el número de zonas. La frecuencia de las mediciones es de aproximadamente una muestra cada 15 minutos, y los datos de la serie temporal abarcan un período de aproximadamente tres a cuatro años.

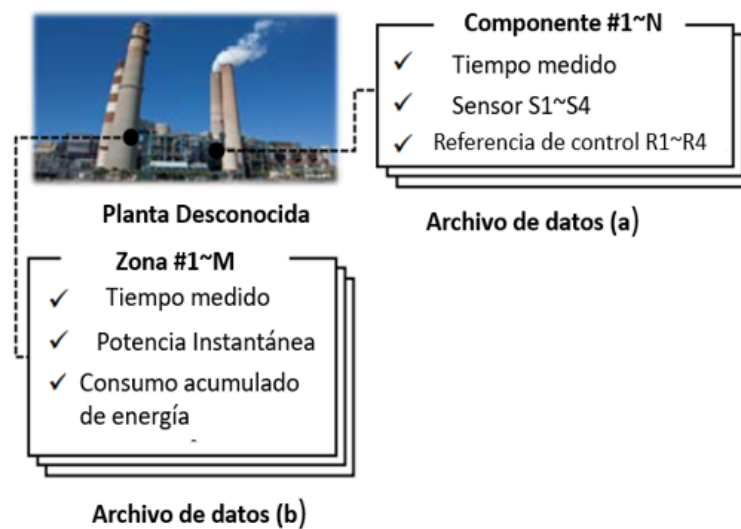


Figura 39. Descripción de los conjuntos de datos, adaptado de [128]

La **Figura 40**, muestra una instancia de la arquitectura de referencia para el escenario. La tarea consiste en predecir posibles fallas. Por ejemplo, el conjunto de datos para la Planta #1 se proporciona mediante una colección de dos archivos [.csv]: plant-1a.csv y plant-1b.csv.

Cada uno de los archivos (a), (b) contiene información como se ha descrito anteriormente. Más precisamente las columnas de cada uno de los archivos (a), y (b) [.csv] son:

Mediciones de la planta por componente: Número de componente "m", tiempo "t", sensores "S1"- "S4", y referencias de control "R1"- "R4".

Mediciones adicionales de la planta por zona en la planta: número de zona "n", hora "t", sensores "E1" y "E2".

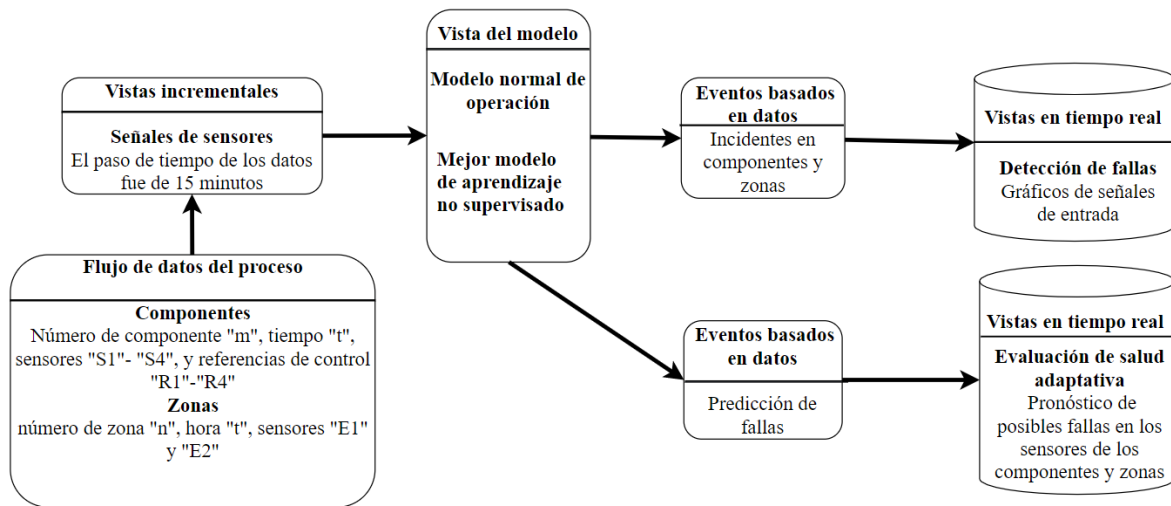


Figura 40. Instancia para el caso de uso reportado en [18].

Además, se proporciona la siguiente información del modelo de planta física:

Cada componente de la planta está controlado por un sistema de lazo de retroalimentación como se representa en la **Figura 41**; los componentes de la planta están desarticulados.

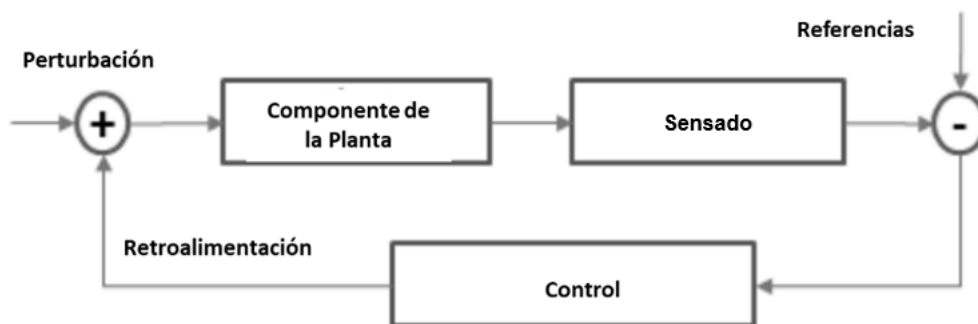


Figura 41. Sistema de lazo de retroalimentación (fuente [127]).

Cada zona mide la energía acumulada (E1) y la energía instantánea (E2) en secciones desarticuladas de la planta que cubren uno o más componentes.

4.3.2 Implementación del modelo basado en datos

La Figura 42, proporciona una visión general del proceso implementado.

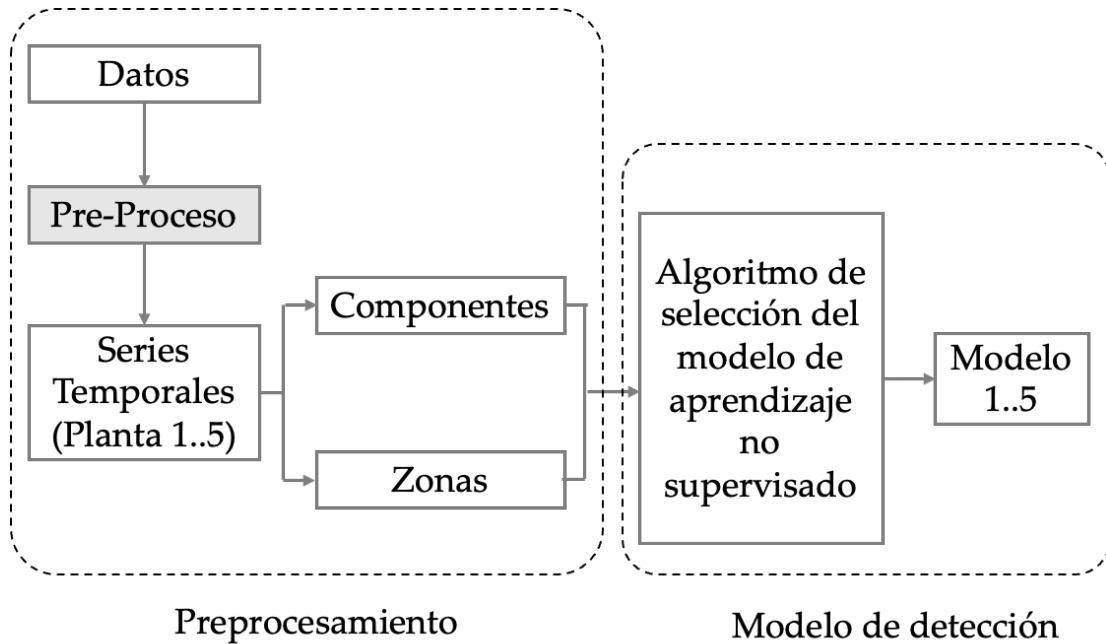


Figura 42. Selección del modelo no supervisado.

En la primera parte se lleva a cabo el preprocesamiento de los datos, que incluye la corrección de fechas, eliminación de datos faltantes, y el procesamiento de variables indicadoras.

En la segunda parte se aplica el algoritmo propuesto en la sección 3.2.2 (página 71) para la selección del modelo para predecir posibles anomalías. Este algoritmo selecciona el modelo con la menor varianza y sesgo de los modelos posibles, considerando los detectores base CBLOF, HBOS, OCSVM y el conjunto de parámetros descritos en la **Tabla 15**, en la página 76. El mejor modelo es aplicado al

conjunto de datos de sensores y energía para cada planta, por lo que en cada planta se crea un modelo independiente para predecir un evento inusual.

Preprocesamiento de datos

El análisis comienza estudiando primero los datos para obtener cualquier información que sea de utilidad para el modelo de detección de anomalías, para esto es importante comprender qué tipo de variables existen.

Tabla 17. Cuenta de niveles únicos para variables categóricas.

Planta	Nm	Nn	S3	R1	R2	R3	R4
1	6	3	12	38	6	8	3
2	13	2	11	26	6	6	3
3	10	2	12	30	7	8	3
4	8	4	12	34	7	7	3
5	3	2	12	12	7	6	3

Nm: Número de Componentes; Nn: Número de Zonas; S3 (Sensor 3); (R1~R4) Referencias de Control.

El primer paso es extraer características útiles de los datos sin procesar para facilitar la detección, por esto, en la Tabla 17 se muestra el conteo de los niveles únicos de las variables categóricas de las primeras cinco plantas, como se observa la mayoría de las plantas tienen un número irregular de componentes y zonas, por lo que puede haber cientos de reglas que definen fallas de estos sistemas, basadas en la combinación de múltiples señales los componentes y zonas. En este caso, es imposible identificar reglas generales para el diagnóstico de la mayoría de los errores.

En la **Figura 42** se muestran las características de las series temporales en sensores 1, 2 y 4, y en la energía eléctrica 1 y 2 durante un poco más de dos años para la planta 1.

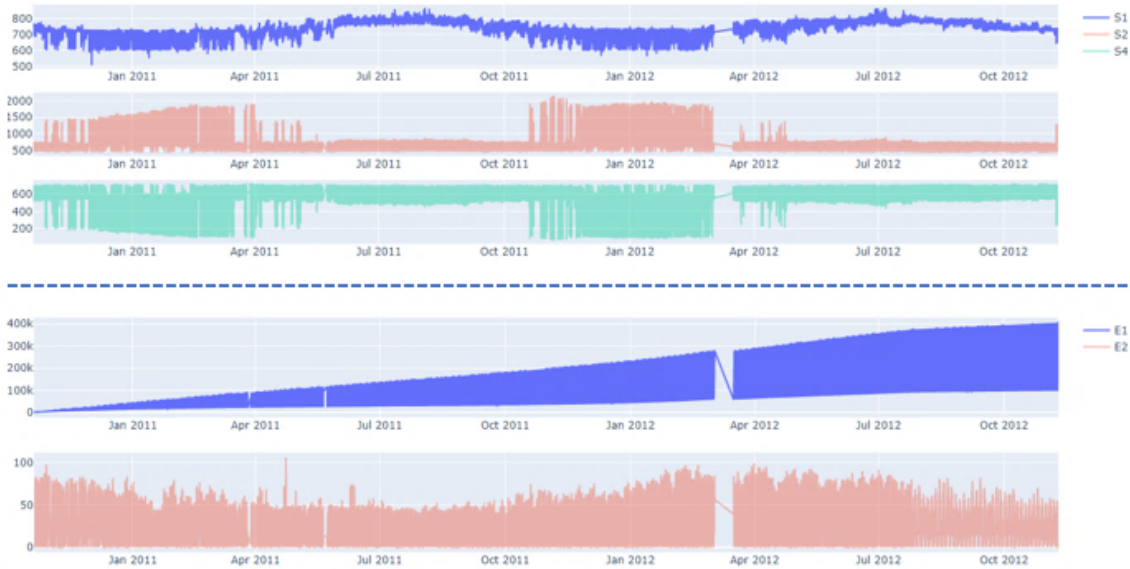


Figura 42. Serie temporal para las variables S1, S2 y S4, E1 y E2 en la planta 1.

En la **Tabla 18** se muestran las variables categóricas que se incluyeron en el procesamiento de datos. Para tal propósito se crearon variables indicadoras de tal forma que de la variable categórica Nn se obtuvieron 6 variables indicadoras (demand 1~5 y Meter), de la variable Nm se obtuvieron 14 variables indicadoras (HVAC 1~14) y las variables indicadoras “Off”, “Occupied” y “set back” se derivaron de la variable categórica R4.

Tabla 18. Variables indicadoras.

Nn	Nm	R4
Demand 1~5	HVAC 1~14	OFF
METER		Occupied, Set back

En la **Figura 43** se muestra el resultado de haber procesado los archivos de las plantas 1~30 para corregir la fecha, los datos faltantes y crear las variables indicadoras.

```

time S1 S2 S3 S4 R1 R2 R3 planta \
0 2009-08-18 18:00:00 711 630 69 600 689 20 40 plant1
1 2009-08-18 18:00:00 725 460 101 705 689 20 40 plant1
2 2009-08-18 18:00:00 711 505 69 678 689 20 40 plant1
3 2009-08-18 18:00:00 705 630 69 600 689 20 40 plant1
4 2009-08-18 18:00:00 734 516 101 671 689 20 40 plant1
...
17192923 2012-11-09 22:00:00 734 511 68 674 700 40 40 plant30
17192924 2012-11-09 22:00:00 729 648 69 589 700 40 40 plant30
17192925 2012-11-09 22:00:00 736 734 0 534 700 40 40 plant30
17192926 2012-11-09 22:00:00 736 592 68 624 700 40 40 plant30
17192927 2012-11-09 22:00:00 736 714 5 547 700 40 40 plant30

m_HVAC1 ... m_HVAC3 m_HVAC4 m_HVAC5 m_HVAC6 m_HVAC7 m_HVAC8 \
0 1 ... 0 0 0 0 0
1 0 ... 0 0 1 0 0
2 0 ... 0 0 0 1 0
3 1 ... 0 0 0 0 0
4 0 ... 0 0 0 0 0
...
17192923 0 ... 1 0 0 0 0
17192924 0 ... 0 0 0 0 0
17192925 0 ... 0 1 0 0 0
17192926 0 ... 0 0 1 0 0
17192927 0 ... 0 0 0 1 0

m_HVAC9 R4_OFF R4_Occupied R4_Setback
0 0 0 1 0
1 0 0 1 0
2 0 0 1 0
3 0 0 1 0
4 0 0 1 0
...
17192923 0 0 1 0
17192924 0 0 1 0
17192925 0 0 1 0
17192926 0 0 1 0
17192927 0 0 1 0

[17192928 rows x 26 columns]

```

Figura 43. Preprocesamiento de los datos de los archivos tipo “a”.

Modelo de detección

En cada uno de los conjuntos de datos de las cinco plantas se aplicó el algoritmo de selección del modelo de aprendizaje no supervisado para predecir posibles fallas, en una muestra del conjunto de datos y estimar el modelo y los parámetros que mejor predicen posibles fallas en los componentes y posibles fallas de electricidad en las zonas de la planta. En consecuencia, en cada planta se seleccionó el mejor modelo y sus parámetros, para predecir posibles fallas en el conjunto de datos.

Tabla 19. Modelo y parámetros seleccionados en las primeras cinco plantas.

Planta	Posibles fallas en los componentes				Posibles fallas en las zonas			
	Muestra	Detector	Pará. 1	Pará. 2	Muestra	Detector	Para. 1	Pará. 2
1	67,253	HBOS	20	0.4	3394	CBLOF	50	euclidian
2	6,356	OCSVM	5	sigmoid	3636	CBLOF	40	euclidian
3	10,495	HBOS	30	0.3	2103	OCSVM	0.5	Lineal
4	8,243	CBLOF	0.1	euclidian	4122	CBLOF	60	manhatan
5	2,836	CBLOF	0.2	manhatan	1933	HBOS	75	0.2

HBOS: Puntuación de valores atípicos basados en histogramas, CBLOF: factor de valor atípico local basado en agrupación, OCSVM: Máquinas de vectores de soporte de una clase.

La **Tabla 19** proporciona una descripción de los modelos seleccionados para las cinco plantas. Para explicar esta tabla consideremos el primer renglón que muestra la planta 1. Aquí se considera una muestra de 67,253 registros considerados para seleccionar el modelo del conjunto de datos de los componentes. En este caso el modelo está construido con el detector HBOS con los valores de los parámetros $n_{\text{histograma}} = 20$ y $\text{tolerancia} = 0.4$. También, en la planta 1, para el conjunto de datos de las zonas, se considera una muestra de 3394 con el detector CBLOF y los valores de los dos parámetros $n_{\text{vecinos}} = 50$ y $\text{método} = \text{“euclidian”}$.

La descripción completa de los parámetros de los modelos no supervisados se puede encontrar en la **Tabla 11** (página 65).

Tabla 20. Posibles fallas en componentes y zonas.

Planta	Posibles fallas en componentes				Posibles fallas en zonas			
	Tamaño	Anomalía	% Anomalía	Tiempo(s)	Tamaño	Anomalía	% Anomalía	Tiempo(s)
1	672530	65291	9.70	45207	339494	20161	5.93	4091
2	635660	45518	7.10	46183	363658	28924	7.95	4869
3	1049547	163469	15.57	68534	210313	12619	6.00	1368
4	824370	72817	8.83	39864	412247	31686	7.68	5899
5	283661	17980	6.33	8566	193343	11601	6.00	1179

La **Tabla 20** muestra los resultados obtenidos al aplicar el mejor modelo obtenido. En la planta 1 para detectar posibles fallas en los componentes se aplicó el modelo HBOS con los parámetros $n_{\text{contenedores}} = 20$ $\text{tolerancia} = 0.4$. La columna tamaño indica el número de observaciones procesadas, en este caso 672530. La columna anomalía es el conjunto de posibles fallas, que corresponde a 65291. La columna porcentaje de anomalía indica la relación entre las anomalías detectadas en el conjunto de datos que es de 9.7. El tiempo indica el número de segundos de procesamiento con las características de equipo y software que se mostraron en la

Tabla 13 (página 68). Aplica lo anterior a la detección de posibles fallas en el conjunto de datos de las zonas para cada una de las cinco plantas.

Para poder visualizar las anomalías encontradas basadas en el algoritmo de selección de modelo, se procede a ajustar las 24 dimensiones del archivo “a” mediante el método del análisis de componentes principales (PCA, por sus siglas en inglés de Principal Component Analysis) para reducir el número a dos dimensiones. En la **Figura 44** se visualizan los resultados a través de una gráfica 2D que proporciona una imagen clara de los puntos de anomalías, las anomalías se resaltan como bordes rojos y los puntos normales se indican con puntos verdes en el trazado. Para poder apreciar de manera clara la gráfica, solamente se presenta una muestra de los primeros 100 puntos del conjunto de datos del archivo que contiene posibles fallas en sensores.

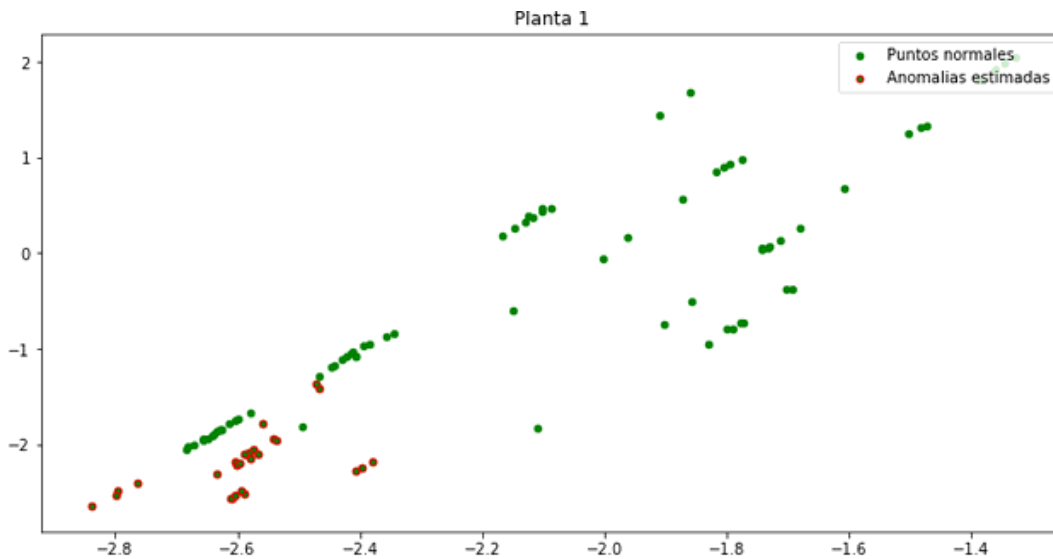


Figura 44. Gráfico de flujo general (100 puntos de posibles fallas en sensores).

4.3.3 Utilidad del modelo basado en datos para la detección de posibles fallas

En esta sección se ha presentado un caso de uso que muestra un escenario para la detección de fallas en un contexto en el que no se conoce con suficiente profundidad el sistema a optimizar, además que los datos provienen de varios

componentes y zonas operando en una variedad de condiciones, mostrando la utilidad para aspectos prácticos y de investigación que aporta el algoritmo propuesto para la selección del modelo de aprendizaje no supervisado para la detección de posibles fallas en etapas iniciales. Para conocer las ventajas que este tiene para la detección de fallas basados en datos en la industria se presenta una discusión en la sección 5.3 que aborda los escenarios de aplicación de los métodos de aprendizaje no supervisado para fallas en la industria abordando las implicaciones o beneficios para la investigación y la práctica.

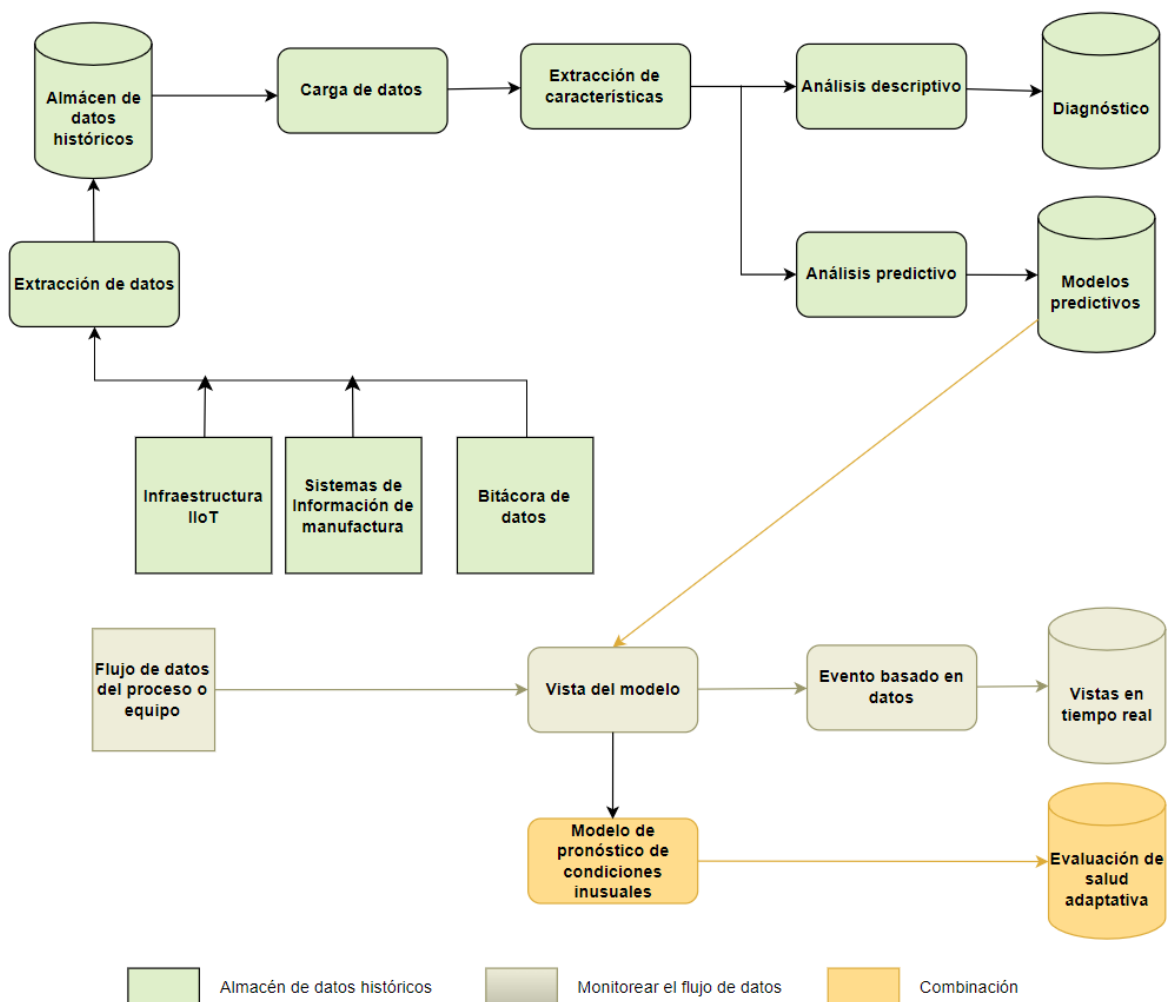


Figura 45. Escenario de modelos basados en datos

4.4 Validación de la arquitectura de referencia con Hadoop

Esta subsección presenta una segunda validación de la arquitectura de referencia y el modelo de gestión de datos, a través de escenarios de Big Data industrial y las tecnologías específicas basadas en código abierto que soportan una plataforma tecnológica aplicable a cada caso.

Cómo se observa en la **Figura 45**, las diferentes fuentes de datos pueden dar origen a tres situaciones en el modelado de Big Data y que han sido considerados en el diseño de la arquitectura de referencia: (1) Almacenar los datos en un depósito histórico para su posterior aprovechamiento, (2) Monitorear el flujo de datos y (3) la combinación de datos históricos con el flujo de datos en tiempo real.

4.4.1 Escenario de datos históricos para analítica de Big Data industrial

En la **Figura 46**, se muestra una instancia de la arquitectura de referencia propuesta para la analítica de Big Data con datos históricos.

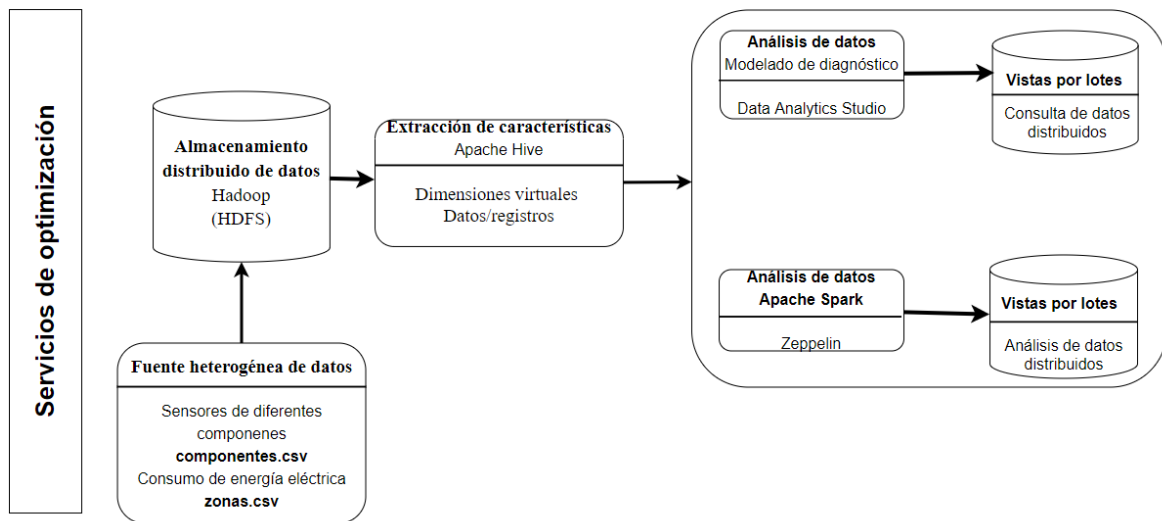


Figura 46. Instancia para analítica de Big Data industrial con datos históricos.

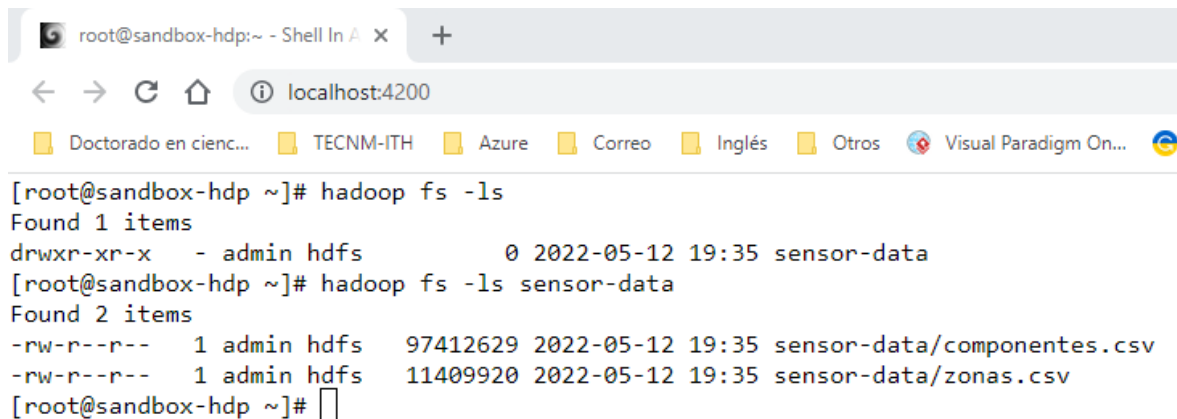
El almacenamiento de Big Data industrial impulsa la necesidad de dividir los datos en distintas computadoras para evitar que una sola máquina se sature. El tipo de sistema de archivos que gestiona el almacenamiento de datos a través de una red

de máquinas se llama Sistemas de Archivos Distribuidos. Apache Hadoop está diseñado para almacenar archivos de gran tamaño con acceso a flujo de datos y que se ejecutan en clústeres de hardware básico.

Cargar los datos de sensores en HDFS

Se creó la carpeta sensor-data para almacenar en HDFS los archivos de datos de una planta industrial: (a) de los sensores de diferentes componentes y sus referencias de control y (b) del consumo de energía eléctrica instantáneo y acumulado.

En la **Figura 47**, se muestra el sistema de archivos de Hadoop para el escenario de datos históricos de analítica de Big data con datos de sensores industriales.



```

root@sandbox-hdp:~ - Shell In A x +
localhost:4200
Doctorado en cienc... TECNM-ITH Azure Correo Inglés Otros Visual Paradigm On...
[root@sandbox-hdp ~]# hadoop fs -ls
Found 1 items
drwxr-xr-x - admin hdfs 0 2022-05-12 19:35 sensor-data
[root@sandbox-hdp ~]# hadoop fs -ls sensor-data
Found 2 items
-rw-r--r-- 1 admin hdfs 97412629 2022-05-12 19:35 sensor-data/componentes.csv
-rw-r--r-- 1 admin hdfs 11409920 2022-05-12 19:35 sensor-data/zonas.csv
[root@sandbox-hdp ~]#

```

Figura 47. Sistema de Archivos de Hadoop

Extracción de características y análisis de datos

Apache Hive facilita el análisis de Big Data industrial almacenado en HDFS con consultas tipo SQL y provee herramientas para la extracción, transformación y carga de datos (ETL, por sus siglas en ingles de Extract, Transform and Load). En la herramienta Data Analytics Studio (DAS) proporciona una interface interactiva para Hive. Los archivos de datos de sensores y mediciones electicas son convertidos en tablas ORC que es un formato optimizado para Big Data y almacenadas en HDFS.

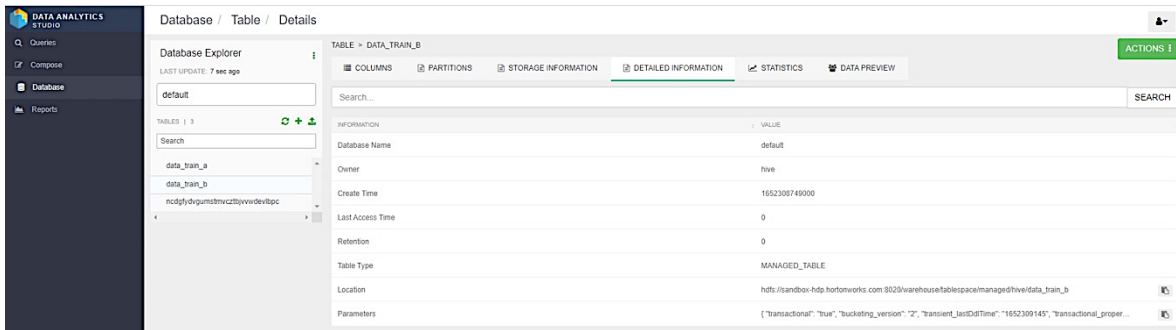


Figura 48. Data Analytics Studio (DAS)

En la **Figura 49**, se presenta una consulta a la tabla de sensores con una instrucción SQL que calcula el promedio de las observaciones de los cuatro sensores agrupados por fecha y hora.

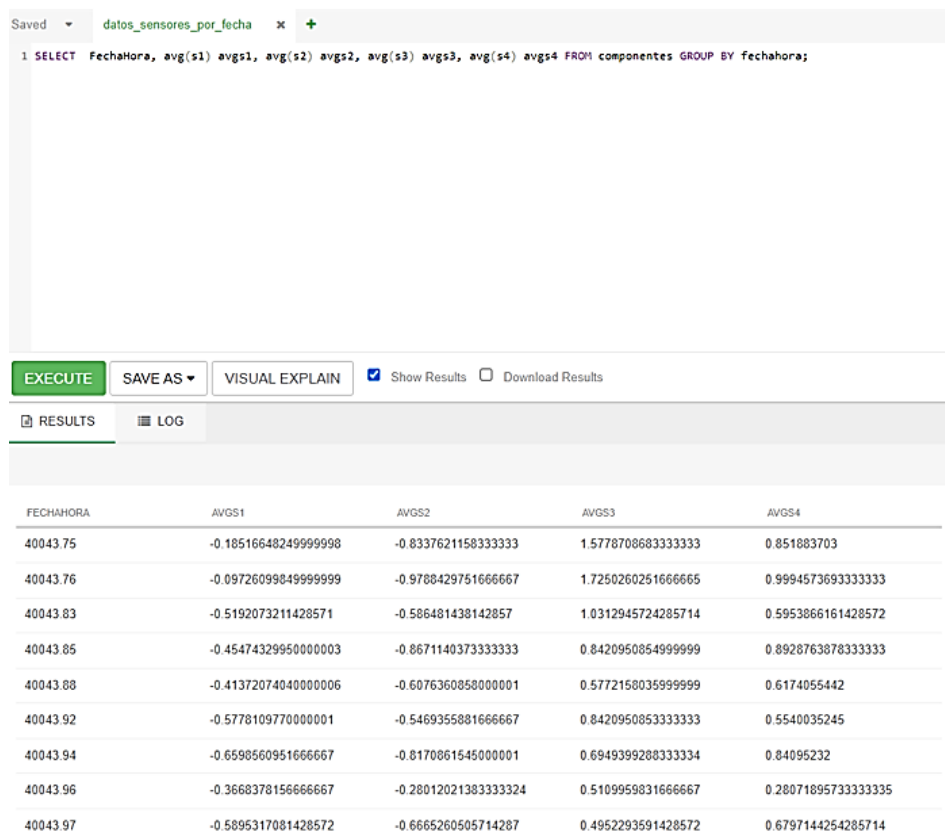


Figura 49. Vista para la creación de consultas SQL en Hive para HDFS.

Esta vista de datos es almacenada en forma permanente como una tabla para su posterior uso como se puede observar en la **Figura 50**.

```

Consulta SQL: promedio de los datos de sensores por fecha y hora

CREATE TABLE sensor_prom
STORED AS ORC
AS
SELECT fechaHora, avg(s1) avgs1, avg(s2) avgs2, avg(s3) avgs3, avg(s4)
avgs4
FROM componentes
GROUP BY fechaHora;

```

Figura 50. Consulta SQL de los datos de sensores en Hive.

Posteriormente los datos son exportados de la tabla ORC a un archivo tipo CSV para el análisis de datos con la herramienta Zeppelin.

Análisis de datos con Apache Spark

Zeppelin es un cuaderno de trabajo basado en la web que permite análisis de datos en forma interactiva (Ver **Figura 51**). Zeppelin aprovecha las características de Apache Spark de extender el modelo de MapReduce, ya que Spark fue diseñado para ser rápida además de una plataforma de cómputo de uso general. Spark tiene integración con Hive, lo que le brinda soporte para los archivos ORC.

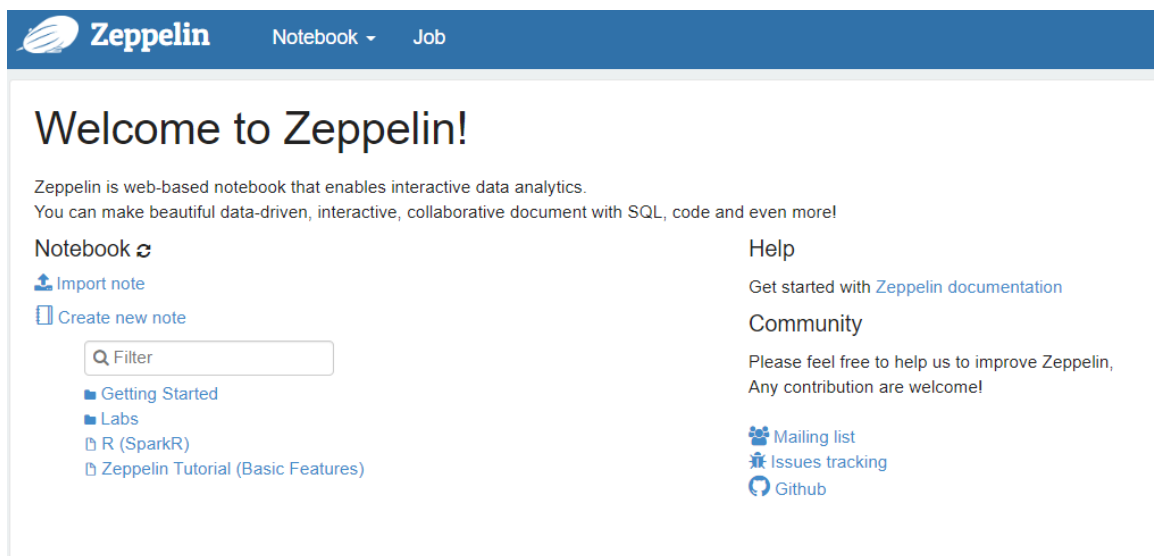


Figura 51. Análisis interactivo de datos con Zeppelin.

Usar datos en Hive

El contexto de Hive es una instancia del motor de ejecución SparkSQL que se integra con los datos almacenados en Hive y da soporte SQL en Spark. En la **Figura 52** se muestra la creación de un contexto de Hive a través de la variable `hiveContext`.

```

%spark2
val hiveContext = new org.apache.spark.sql.SparkSession.Builder().getOrCreate()
hiveContext: org.apache.spark.sql.SparkSession = org.apache.spark.sql.SparkSession@26eacd98

Took 4 min 15 sec. Last updated by anonymous at May 11 2022, 6:35:48 PM.
    
```

Figura 52. Contexto de Hive para soportar SQL en Spark.

Procesar datos distribuidos

La abstracción del núcleo principal de Spark se denomina conjunto de datos distribuido resistente o RDD (Por sus siglas en inglés Resilient Distributed Dataset).. En otras palabras, RDD es una colección inmutable de objetos que se divide y distribuye en varios nodos físicos de un clúster de YARN y que se puede operar en paralelo. En la **Figura 53**, se crea un RDD del conjunto de datos almacenados en HDFS desde el archivo CSV que contiene los datos de los promedios de los sensores por fecha y hora, con el esquema que definimos en la tabla.

```

%spark2
/**
 * La libreria SQL Types nos permite definir los tipos de nuestro esquema de la base de datos creada
 * en HIVE y almacenada en HDFS
 */
import org.apache.spark.sql.types._

val sensorPromSchema = new StructType().add("fechaHora",DoubleType,true).add("avgs1",DoubleType,true)
    .add("aves2".DoubleTvpe.true).add("aves3".DoubleTvpe.true).add("aves4".DoubleTvpe.true)

import org.apache.spark.sql.types._
sensorPromSchema: org.apache.spark.sql.types.StructType = StructType(StructField(fechaHora,DoubleType,true), StructField(avgs1,DoubleType,true), StructField(avgs2,DoubleType,true), StructField(avgs3,DoubleType,true), StructField(avgs4,DoubleType,true))

Took 1 sec. Last updated by anonymous at May 11 2022, 6:46:09 PM. (outdated)
    
```

Figura 53. Creación de un conjunto de datos distribuidos.

Ahora es posible poblar el esquema *sensorPromSchema* con los datos del archivo CSV que está almacenado en HDFS.

```
%spark2
val sensorPromDataFrame = spark.read.format("csv").option("header", "true").schema(sensorPromSchema)
    load("hdfs://tmp/data/sensor_prom.csv")

sensorPromDataFrame: org.apache.spark.sql.DataFrame = [fechaHora: double, avgs1: double ... 3 more fields]

Took 4 sec. Last updated by anonymous at May 11 2022, 7:30:08 PM.
```

Figura 54. Poblar el esquema con datos de los promedios de los sensores.

Como se observa en la Figura 55, se crea una vista temporal *sensorPromedio*.

```
%spark2
sensorPromDataFrame.createOrReplaceTempView("sensorPromedio")

Took 0 sec. Last updated by anonymous at May 11 2022, 7:30:18 PM.
```

Figura 55. Vista temporal *sensorPromedio*.

Con lo anterior se usó la sesión de Spark SQL para hacer la consulta a *sensorPromedio*, como se observa en la Figura 56.

```
%spark2
/**
 * Inicializa los promedios de los sensores y los registra como un RDD
 */
val sensor_prom = hiveContext.sql("SELECT * FROM sensorPromedio LIMIT 15")
sensor_prom.createOrReplaceTempView("sensor_prom")
hiveContext.sql("SELECT * FROM sensor_prom LIMIT 15").show
```

fechaHora	avgs1	avgs2	avgs3	avgs4
40581.15	-1.9667176215	-1.0238680698333333	-0.6064634870000001	1.0486485911666668
40581.17	-2.0956456648333335	-1.0505496073333334	-0.6064634870000001	1.0787098935
40581.19	-2.046251154857143	-1.1082007862857142	-0.604492659	1.1364900592857141
40581.2	-2.2245737076666665	-1.082233933	-0.6110620856666666	1.108771196
40581.21	-2.265596267	-1.0905719133333331	-0.6110620856666666	1.116969733
40581.26	-2.4414072345	0.03338785100000008	-0.6110620856666668	0.021098618666666586
40581.28	-1.4861676436666666	-0.1934052171666667	-0.5926676911666667	0.19600074150000002
40581.3	-1.2827292380000002	0.9815353417142857	-0.5966093471428572	-0.6859275972857143
40581.31	-1.5447712993333333	1.2824173216666666	-0.4639069291666666	-1.0173827368333332
40581.33	-1.353053625142857	0.3211672904285714	-0.5020096035714287	-0.25960367257142863
40581.34	-0.8591085253333333	1.0689650224999998	-0.4823013236666667	-0.9681915146666666
40581.35	-0.7794075534	-0.16739071799999997	-0.5871493728000001	0.1714051306
40581.41	-1.263473751	0.8371691659999999	-0.4868999223333333	-0.8014879285

Figura 56. Consulta SQL a *sensorPromedio*.

```
%jdbc(Hive)
Select * from sensor_prom
```

Figura 57. Consulta SQL interactiva a *sensor_prom*.

En la **Figura 57**, se muestra la instrucción requerida para interactuar a través de cada pestaña que aparecen en la consulta, y ver desplegados un tipo diferente de gráfica dependiendo la configuración de los datos que se deseen como se aprecia en la **Figura 58**.

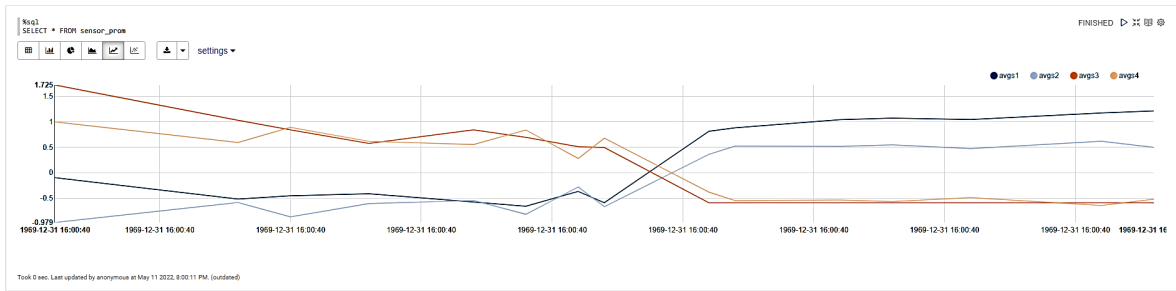


Figura 58. Gráfico interactivo para la consulta SQL a *sensor_prom*.

4.4.2 Escenario de flujo de datos para analítica de Big Data industrial

En la **Figura 59**, se muestra una instancia de la arquitectura de referencia propuesta para la analítica de Big data para flujo de datos proporcionados por el IIoT provenientes del iCPS.

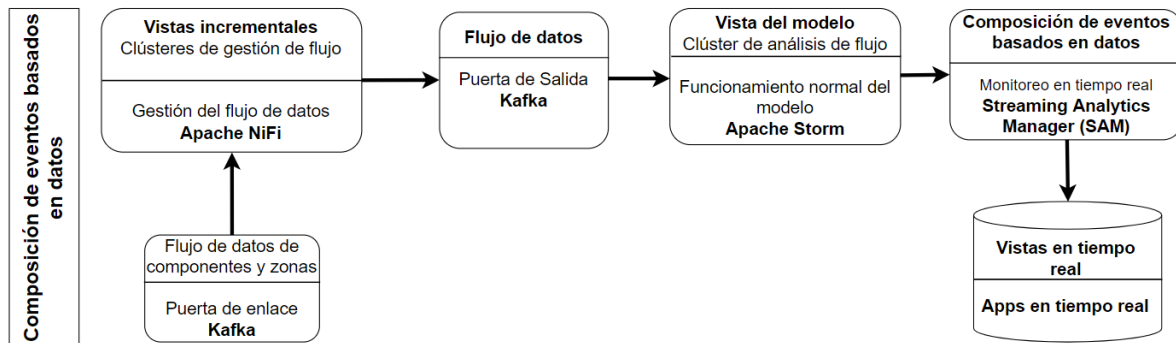


Figura 59. Instancia para analítica de Big Data industrial para flujo de datos.

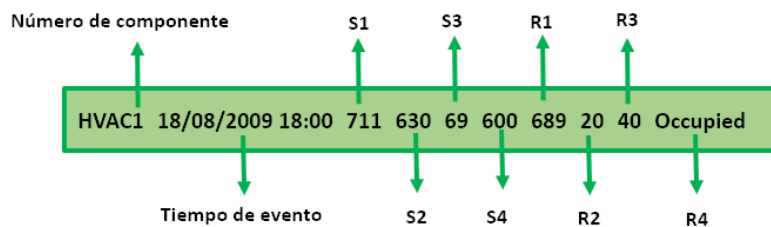
El flujo de datos recopilados por el IIoT genera la necesidad de procesar los datos a través de un flujo que pueda ser controlado y realizar cálculos sencillos a procesar

eventos complejos para presentar eventos en operación actualizados y poder tomar acciones.

Flujo de datos de componentes y zonas

En la **Figura 60** se muestra cómo están compuestos el flujo de datos de los sensores de los componentes (plant-1.csv) y flujo de datos de las zonas (plant-1b).

Flujo de datos de plant-1a.csv



Flujo de datos de plant-1b.csv



Figura 60. Estructura del flujo de datos.

Kafka mantiene la ingesta de mensajes en categorías llamadas temas. En la **Figura 61** se muestra la creación de estos temas para este caso.

```
[root@sandbox-hdf bin]# ./kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 3 --topic componentes;
Created topic "componentes".
[root@sandbox-hdf bin]# ./kafka-topics.sh --create --zookeeper localhost:2181 --replication-factor 1 --partitions 3 --topic zonas;
Created topic "zonas".
[root@sandbox-hdf bin]#
```

Figura 61. Creación de temas en Kafka para este caso.

El flujo de eventos datos de un enlace de IIoT se envía en archivos CSV. Para este escenario los datos son transmitidos desde un archivo a través de Kafka Connect desde un archivo local en la máquina anfitrión, como se muestra en la **Figura 62**.


```

curl -i -X PUT -H "Accept:application/json" \
  -H "Content-Type:application/json"
http://sandbox.hortonworks.com:8083/connectors/source-csv-spooldir-
00/config \
  -d '{
    "connector.class":
"com.github.jcustenborder.kafka.connect.spooldir.SpoolDirCsvSourceConnect
or",
    "topic": "componentes",
    "input.path": "/data/unprocessed",
    "finished.path": "/data/processed",
    "error.path": "/data/error",
    "input.file.pattern": "componentes.csv",
    "schema.generation.enabled":"true",
    "csv.first.row.as.header":"true"
  }'

```

Figura 62. Transmisión de los datos de componentes CSV con Kafka Connect.

Vistas incrementales y puerta de salida

En este caso NiFi actúa como el productor que ingiere datos del IIoT, y permite controlar la ingesta de datos identificando el evento de datos transmitido por Kafka, aprovechando la función Kafka Header en Kafka 1.0 en NiFi.

En la **Figura 63**, se describe cómo los datos de los sensores contenidos en CVS son enviados por la puerta de enlace a través de la aplicación de Kafka, en un tema de Kafka que contiene el encabezado (Kafka header) y los datos del archivo CVS (kafka payload).

Nifi permite la entrega controlada de datos mediante un búfer y la persistencia de los datos a tasas muy altas de transacciones.

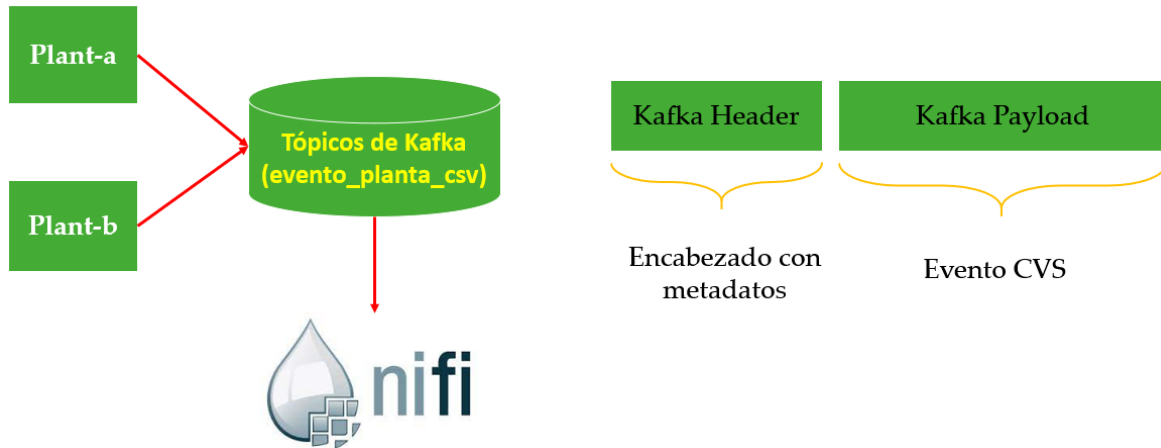


Figura 63. Flujo de datos y procesamiento por nifi usando procesamiento basado en registros.

En la **Figura 64**, se muestra la plantilla de Nifi creada para este escenario. El enlace de IIoT envía eventos CSV con el nombre de esquema en el mensaje de la cabecera de Kafka (Kafka Header).

A continuación, se describen los siguientes procesadores usados en la plantilla:

1. GetPlantData genera los datos de los componentes y zonas como un flujo de datos.
2. Atributos lee los datos en CVS
3. ConvertRecord transforma en datos con el esquema de AVRO para poder ser analizados por SAM (Streaming Analytics Manager). AVRO es utilizado en Apache Hadoop para el intercambio de Big Data entre aplicaciones.
4. PublishKafka_1_0 envía el contenido del archivo de flujo de datos como un mensaje a un tema de Kafka.

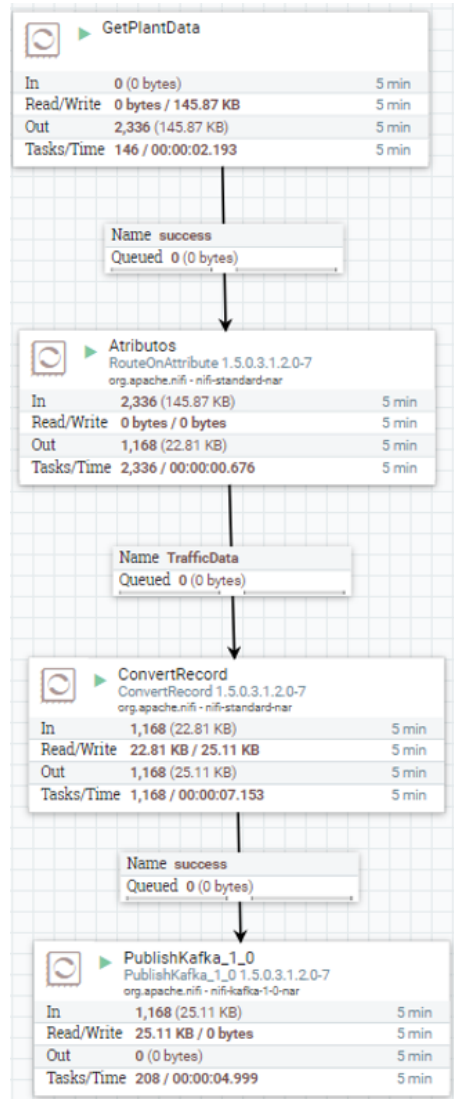


Figura 64. Ingesta de los eventos de los sensores sin procesar.

En la Figura 65, se muestran las acciones que los procesadores de Nifi están haciendo en los datos a través de la tabla procedencia de los datos de Nifi. En la medida que los datos se procesan a través del sistema se transforman, enrutan, y distribuyen a través de los procesadores. “DROP” indica la conclusión de la vida de un objeto por algún motivo que no sea la caducidad del objeto. “ROUTE” Indica que un flujo se enrutó a una relación específica y brinda información sobre por qué el flujo se enrutó a esta relación. “CREATE” Indica que un flujo se generó a partir de datos que no se recibieron del proceso de ingestión de datos.

Displaying 1,000 of 1,000
 Oldest event available: 05/14/2022 23:57:30 UTC

Filter by component name ▼

	Date/Time	Type	FlowFile Uuid
❏	05/15/2022 00:03:53.236 UTC	DROP	3ac5cc41-080b-4693-877a-3f456b77bc0a
❏	05/15/2022 00:03:53.236 UTC	ROUTE	3ac5cc41-080b-4693-877a-3f456b77bc0a
❏	05/15/2022 00:03:53.236 UTC	CREATE	55cc52d3-dbac-42f4-9eae-1eab9894fa83
❏	05/15/2022 00:03:53.236 UTC	ROUTE	55cc52d3-dbac-42f4-9eae-1eab9894fa83
❏	05/15/2022 00:03:53.236 UTC	ROUTE	f82af952-3d63-4e3d-bea6-cd6ed7f7a6b6
❏	05/15/2022 00:03:53.236 UTC	CREATE	22fb5adb-af73-4d0f-b4d7-f26d2f1322b4
❏	05/15/2022 00:03:53.236 UTC	ROUTE	22fb5adb-af73-4d0f-b4d7-f26d2f1322b4

Figura 65. Procedencia de los datos de Nifi.

En la **Figura 66**, un evento ilustra qué tipo de acción tomó el procesador contra los datos. Es posible verificar la procedencia de los datos en cada procesador para obtener una visión más detallada de los pasos que realiza NiFi para procesar y transformar los datos.

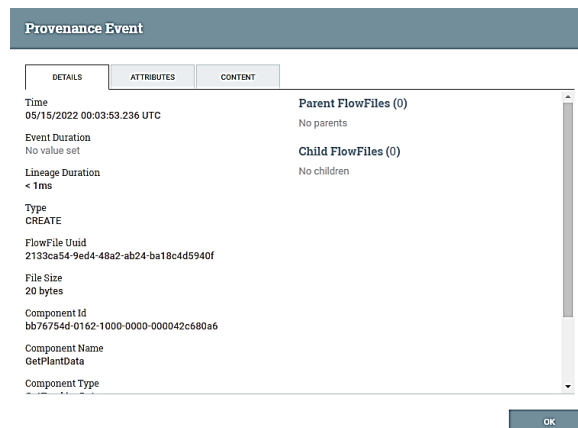


Figura 66. Evento de procedencia.

Vistas del modelo

En el modelo de procesamiento de flujo, los datos se envían directamente a un motor de análisis de datos para que se calculen uno por uno y los resultados se produzcan en tiempo real. Se ha creado una fuente de flujo de datos usando Kafka y NiFi, a continuación, se presentan los resultados de la aplicación de la herramienta

de procesamiento de flujo de datos en tiempo real SAM (Streaming Analytics Manager) que integra Apache Storm como motor de análisis de flujo de datos.

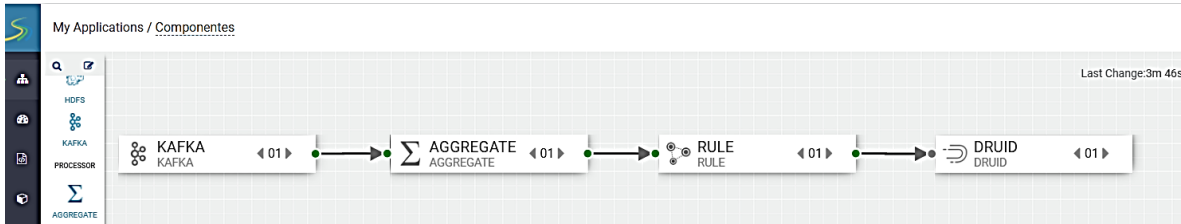


Figura 67. Topología de SAM.

En la **Figura 67**, se muestra la topología de componentes:

1. Cada aplicación de transmisión debe comenzar con una fuente como es mostrado en la **Figura 68**. El componente de construcción de Kafka conecta con Nifi y el flujo de datos con el esquema de AVRO.

Figura 68. Componente Kafka.

- La **Figura 69** muestra cómo usar funciones agregadas. Se requiere crear una ventana temporal para dividir los datos cada tres minutos de operación y obtener el promedio de los datos de los sensores en ese intervalo de tiempo.

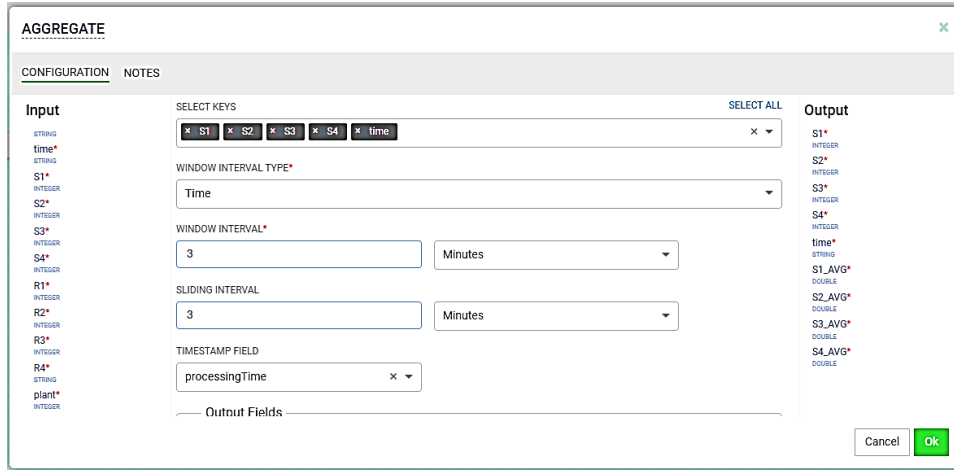


Figura 69. Componente de agregación.

- Para filtrar eventos que son infracciones o violaciones en el flujo de datos SAM usa reglas las cuales son trasladadas en consultas SQL que operan en el flujo de datos. En la **Figura 70**, se muestra la regla con nombre “rango del sensor” que determina una violación para la entrada S1_AVG cuando el promedio del sensor 1 sobrepasa el valor de 2.

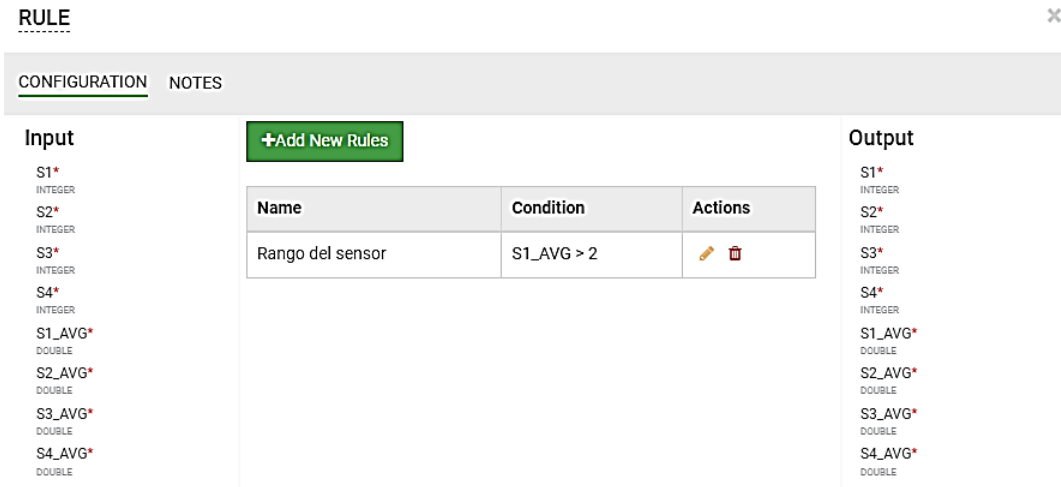


Figura 70. Reglas para los sensores de los componentes.

- En la **Figura 71** se muestra la regla de transformación que permiten enviar los datos a DRUID. Esto permite transmitir las alertas a un tablero y que puedan ser mostradas para su análisis se transmiten los datos a DRUID para crear cubos OLAP para posteriormente visualizar los datos en un tablero en Superset.



Figura 71. Enviar el flujo de datos a Druid.

Vistas en tiempo real

Para diseñar, mantener y permitir las visualizaciones de datos recibidos de los componentes y zonas, además de comunicar los descubrimientos a los interesados, se ha utilizado la plataforma de exploración y visualización de datos Apache Superset.

En la **Figura 72** se muestra la creación de una visualización de los sensores, para recompilar con la configuración requerida para mostrar el promedio de los sensores S1, S2, S3 y S4 filtrados por el componente m_HVAC1. La pestaña “Datasource & Chart Type” se utiliza para seleccionar el origen de los datos y seleccionar el tipo de gráfico. La pestaña “Time” se utiliza para determinar el valor agregado, es decir, Los valores se agregan utilizando intervalos de tiempo basados en el rango de tiempo

de los datos que se trazan. La pestaña “Query” se utiliza para seleccionar consulta de los datos a mostrar en la gráfica, incluyendo las métricas, si están los datos agrupados por una variable, el límite de la serie temporal y permite ordenar la datos por una variable. La pestaña “Chart Options” permite determinar la leyenda, la simbología y la apariencia.

La **Figura 73** muestra la gráfica de línea de series de tiempo para un periodo de dos días de monitoreo. El eje x muestra los campos de fecha/hora visualizados en el gráfico, el eje y muestra los campos numéricos del promedio de los sensores.

The image shows a configuration interface for a time series visualization. It is organized into several sections:

- Datasource & Chart Type:**
 - Datasource: sensor3
 - Visualization Type: Time Series - Line Chart
- Time:**
 - Time Column: time
 - Time Grain: Time Column
 - Since: 2009-08-18
 - Until: now
- Query:**
 - Metrics: avg_S1, avg_S2, avg_S3, avg_S4
 - Group by: Select 14
 - Series limit: Select 7
 - Sort By: Select 55
 - Sort Descending
- Chart Options:**
 - X Axis
 - Y Axis
 - Advanced Analytics

Figura 72. Porciones de visualización.

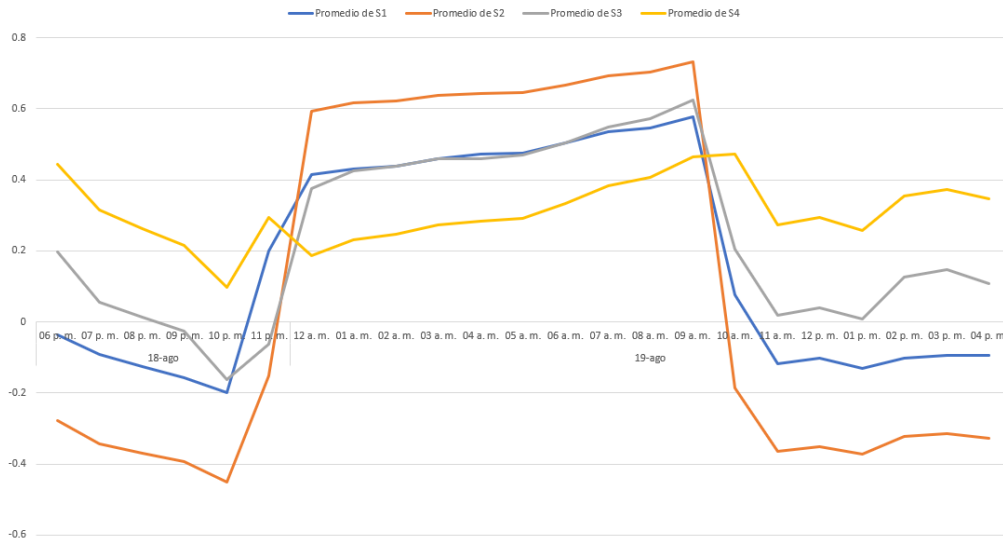


Figura 73. Gráfico de línea de series de tiempo

4.4.3 Utilidad de la implementación de la arquitectura de referencia

El uso de analítica de Big Data en la industria es una de las aplicaciones del IIoT con más interés en el ámbito de los iCPS. La principal motivación es la mejora de los servicios los procesos en la industria con el uso de las tecnologías emergentes.

Actualmente la industria no está preparada para afrontar la generación de grandes volúmenes de datos por los dispositivos de IIoT desplegados a gran escala. Por ello en esta sección se ha presentado el uso de dos instancias de la arquitectura de referencia para analítica de Big Data industrial con el fin de resolver las limitaciones actuales de los sistemas de iCPS.

En el escenario de datos históricos para analítica de Big Data se ha validado la gestión de los datos generados por los componentes y zonas almacenadas en HDFS. La gestión de los datos masivos de IIoT permite extraer información útil que permite conocer el estado de los sensores de los componentes y el consumo eléctrico de las zonas. Este sistema de analítica de Big Data industrial incorpora dos aplicaciones basadas en almacenamiento HDFS. La primera aplicación, Data Analytics Studio

(DAS), extrae información basada en consultas SQL que permite generar vistas y nuevas tablas. De esta manera es posible extraer información almacenada en HDFS de forma estructurada.

Por otro lado, la segunda aplicación permite el análisis con Spark mediante un cuaderno de trabajo Zeppelin basado en la web para el análisis de datos en forma interactiva. De esta forma es posible realizar consultas a la información almacenada en HDFS en forma distribuida. Lo que habilita controlar de los datos para la toma de decisiones.

Además, las plataformas de código abierto utilizadas en la instanciación de la arquitectura de referencia han probado ser adecuadas para el manejo de grandes volúmenes de datos históricos generados por el IIoT. El uso de la API de SparkSQL de Apache Spark para el procesamiento de los datos facilitó una integración más fácil e intuitiva. La abstracción RDD Apache Spark permitió la extracción de un análisis descriptivo de forma rápida.

El escenario de flujo de datos para analítica de Big Data industrial permite habilitar un entorno inteligente con el fin de que los datos producidos por los sensores puedan ser utilizados en eventos actualizados de manera activa. La monitorización de los procesos industriales es una de las aplicaciones de iCPS con mayor investigación en los últimos años. Se desea tener un control más activo que permita un estado de salud adaptativa de los procesos y máquinas industriales. En este sentido se ha presentado una instancia para la composición de eventos basados en datos.

Este sistema incorpora los servicios de Apache Kafka y Apache Nifi para desarrollar un mecanismo de integración en la ingesta de datos que permita un flujo controlado de eventos que asegure la persistencia y el control mediante un búfer de los datos a tasas muy altas de transacciones. Por otro lado, el uso de SAM para crear una aplicación de análisis del flujo de datos desplegada en Storm como motor de

análisis de flujo de datos permite el análisis de manera más rápida y sencilla. Los resultados de este procesamiento son mostrados en Apache Superset que permite la exploración y visualización de los datos.

Con lo anterior, se puede establecer que la arquitectura de referencia ha definido un marco de trabajo que agiliza y facilita el diseño de soluciones para analítica industrial en diferentes escenarios al ser una guía metodológica para integrar fases o etapas con herramientas de software libre que soportan el procesamiento de Big Data de IIoT por lotes y el procesamiento de flujo de datos del IIoT en tiempo real. Por lo que estas dos implementaciones muestran que la arquitectura de referencia puede ser utilizada para guiar el diseño de una solución para analítica industrial.

Capítulo 5

Discusión

Este capítulo presenta una discusión de los resultados obtenidos en relación con la literatura relacionada. Los trabajos de investigación relacionados que se encuentran en el ámbito de esta investigación se clasifican de la siguiente forma: (1) Arquitecturas de referencia en contextos industriales, (2) Adopción de arquitecturas de referencia en el ámbito de la analítica industrial y (3) los métodos y modelos no supervisados para la detección temprana de fallas en la industria.

5.1 Arquitecturas de referencia en contextos industriales

La adopción de arquitecturas de referencia acelera el diseño de soluciones de sistemas para problemas comunes de procesos industriales. Sin embargo, las propuestas dentro de las áreas de Big Data se centran en sistemas de información o software en el ámbito de los procesos de negocio, siendo escasas las propuestas explícitamente para la Industria 4.0.

A partir de la literatura relacionada con arquitecturas de referencia para soluciones industriales de Big Data, se ha observado que no brindan soporte sistemático a la analítica de iCPS para la generación de posibles alternativas de decisiones arquitectónicas de acuerdo con los requerimientos de la Industria 4.0.

En [129], se propone una metodología para el diseño de Big Data que puede beneficiarse con la arquitectura de referencia para analítica industrial propuesta en esta tesis , considerando los componentes de cada capa como métodos de diseño y enriqueciendo el catálogo de tecnologías para la analítica de Big Data en contextos industriales. En [129], se propone una metodología para el diseño de Big Data que puede beneficiarse con la arquitectura de referencia para analítica industrial propuesta en esta tesis , considerando los componentes de cada capa como métodos de diseño y enriqueciendo el catálogo de tecnologías para la analítica de Big Data en contextos industriales.

Los autores en [27] consideran la necesidad de rediseñar sus sistemas de información heredados de la industria manufacturera para que se integren con las plataformas de análisis de datos actuales, para ello se requiere un enfoque que evalúe en forma cuidadosa y exhaustiva los objetivos para la adopción de analítica de datos. En esta tesis (ver subsección 2.2.3), se han presentado los requisitos funcionales y atributos de calidad requeridos para permitir la toma de decisiones en relación con el diseño de un sistema de analítica industrial y que han sido incorporados a la arquitectura de referencia propuesta para el diseño integral de sistemas de analítica industrial.

En comparación con el modelo 1-3-5 en [130], la arquitectura de referencia propuesta en esta tesis se puede utilizar como guía para identificar soluciones arquitectónicas alternativas que consideren nuevas técnicas analíticas de modelado de datos que incluyen modelado de diagnóstico, modelado predictivo y modelado prescriptivo.

5.2 Diseño de arquitecturas de análisis de Big Data aplicadas en contextos iCPS

La razón principal de la complejidad del desarrollo de análisis de Big Data para iCPS es la dificultad en la integración de tecnologías en un ecosistema consistente en un entorno industrial. Además, la analítica prescriptiva es intrínsecamente compleja debido a la necesidad de alinear la experiencia con el diseño de modelos, la optimización de procesos y la previsión de fabricación. En la revisión de literatura presentada en la sección 2.1 (página 19), se observa que existe una falta de desarrollo de aplicaciones de analítica prescriptiva. En la **Tabla 21** se presenta una comparativa de la arquitectura de referencia propuesta en relación con arquitecturas para iCPS reportadas en la literatura.

Tabla 21. Comparativa de la arquitectura propuesta con arquitecturas reportadas en la literatura.

Característica	Arquitectura					Propuesta
	[131]	[7]	[44]	[43]	[128]	
Integración de múltiples fuentes	•					•
Procesamiento de flujo de datos		•	•			•
Procesamiento por lotes			•			•
Procesamiento de datos escalable y elástico	•				•	•
Composición de eventos basados en datos				•		•
Soporte de decisiones basado en análisis		•		•		•

Por ejemplo, en [131], la propuesta de los autores genera arquitecturas para la integración de datos industriales en base a modelos. En [7], se define un marco conceptual para la adopción de CPS que soporten la arquitectura de sistemas CPS de manufactura para la Industria 4.0. Se enfocan particularmente en definir principios rectores para la implementación de CPS en la industria a través de una

metodología. A diferencia de esas propuestas, en esta tesis se han introducido atributos de Big Data industriales e impulsores de la gestión de datos que especifican requisitos arquitectónicos que guían el diseño de la arquitectura para analítica en iCPS, que además considera la analítica prescriptiva.

En [49], los autores presentan una arquitectura de Industria 4.0 como una descripción general de los componentes que incluyen analítica. La principal diferencia entre el enfoque de esta tesis radica en limitar el enfoque en los sistemas de analítica basados en el modelo de datos y diseñar arquitecturas de soluciones para integrarlos con las plataformas para analítica.

De igual forma, los autores en [48] proponen integrar las plataforma IIoT con el análisis de Big Data, en ese sentido, una característica clave es habilitar IIoT para Big Data. Si bien este enfoque reconoce las limitaciones en las plataformas IIoT que permiten escenarios de Big Data y mantiene descripciones de análisis de procesos, no representa los escenarios de gestión de datos de los sistemas de información de manufactura. Tampoco aborda el procesamiento en línea para analítica que permite el soporte a diferentes perspectivas en la toma de decisiones.

En [132], se espera que el análisis prescriptivo para Big Data pueda utilizarse en sistemas de información de manufactura (MIS) para análisis de datos avanzados. Sin embargo, dicho trabajo presenta una referencia arquitectónica que guía la toma de decisiones en el diseño de la arquitectura para analítica de Big Data. La arquitectura de gestión de datos propuesta en esta tesis presenta fuertes indicios de poder manejar una amplia gama de escenarios industriales para análisis prescriptivo, incluido el diagnóstico de fallas, el monitoreo en tiempo real y el pronóstico de condiciones inusuales del proceso.

5.3 Modelos no supervisados para detección temprana de fallas

Un problema frecuente en la industria es la dificultad de contar con grandes volúmenes de datos históricos con información de fallas. Además, resulta una limitante contar con los recursos necesarios para hacer un análisis profundo de los datos y probar con diferentes estrategias para seleccionar la combinación de modelos más adecuada. Existen en la industria dificultades para contar con estos recursos, por lo que es deseable hacer una gestión inteligente de fallas menos dependiente de conocimientos previos y experiencia diagnóstica al procesar macrodatos. A partir de la literatura relacionada para la detección de fallas basadas en datos en la industria, se ha observado que no aportan un soporte para la selección del mejor modelo de aprendizaje no supervisado.

Tabla 22. Literatura revisada para aprendizaje no supervisado aplicado a fallas en la industria.

Ref.	Metodología	Método	Escenario
[119]	Se presenta un enfoque de diagnóstico inteligente de fallas para un problema mecánico que utiliza Big Data y aprendizaje no supervisado para proporcionar una predicción precisa en el caso de registros de cojinetes de motor.	Primera etapa de aprendizaje: Red neuronal de dos capas no supervisada. Segunda etapa: clasificación por regresión	Diagnóstico de fallas
[120]	Se aplicaron métodos de aprendizaje no supervisado para analizar datos y diagnosticar fallas y para predecir fallas potenciales en los compresores.	Parte de aprendizaje: Aprendizaje del diccionario Parte de análisis: Clasificación SVM.	Diagnóstico de fallas
[133]	Metodología que utiliza aprendizaje no supervisado para una implementación rápida de la actividad de mantenimiento predictivo que incluye la predicción de fallas y la detección de clases de fallas para fallas conocidas y desconocidas.	Gaussian Mixture Model y K-Means	Diagnóstico de fallas
[124]	Un método de sistema de aprendizaje autónomo para la detección de fallas en tiempo real en procesos industriales basado en un enfoque evolutivo.	TEDA (Typicality and Eccentricity Data Analytics)	Monitoreo de flujo de datos
[125, 134, 135]	Metodología genérica basada en métodos de aprendizaje no supervisado para correlacionar fuertemente fallas detectadas en los datos históricos del registro del proceso con el flujo de datos entrante, de acuerdo con alcance de predicción.	Decision Tree, Random Forest, Gaussian/Bernoulli Naive Bayes, Multilayer Perceptron	Procesos inusuales

En la **Tabla 22** se resume la metodología, el algoritmo implementado y el escenario de aplicación de los métodos de aprendizaje no supervisado para fallas en la industria. A continuación, se agrupan de acuerdo con la metodología empleada y se compara con la metodología usada en esta tesis para seleccionar el mejor modelo no supervisado para la detección de fallas en etapas iniciales: (sección 5.3.1) Métodos no supervisados para obtener características de los datos y su aplicación en fallas industriales, (sección 5.3.2) Metodologías genéricas, (sección 5.3.2) Método único.

5.3.1 Métodos no supervisados para obtener características de los datos y su aplicación en fallas industriales

El método inteligente para la detección de fallas usado en [119] utiliza métodos no supervisados para aprender las características de los datos iniciales, a diferencia de los métodos tradicionales en donde las características son extraídas manualmente con base en la experiencia del proceso en particular. En una primera etapa los autores aplican una red neuronal de dos capas no supervisada. En la segunda etapa se aplica un método de clasificación para conocer la condición de salud basada en las características aprendidas.

Un enfoque similar de dos etapas es propuesto en [120], donde los autores consideran una primera etapa llamada de aprendizaje, en la que se analizan los datos para desarrollar un modelo para la predicción del estado futuro de trabajo de los compresores, la segunda etapa los autores la consideran como etapa de análisis, y utiliza el modelo generado para predecir el estado de los compresores para identificar posibles fallas.

En [133] se presenta una propuesta que pretende superar los desafíos del aprendizaje supervisado al no contar con suficientes datos históricos, y la falta de rapidez para detectar nuevas características. Este enfoque requiere un mínimo de datos históricos con el método Gaussian Mixture para separar condiciones normales de operación y clases sin etiquetas para detectar posibles fallas.

La estrategia de usar dos etapas, la primera para extraer las características de los datos de operación normal de una máquina o proceso con un método no supervisado requiere al menos de un mínimo de datos históricos. Con base en esto, es posible aplicar un método de clasificación supervisada para predecir cualquier valor que no esté dentro de la clasificación propuesta por el método de aprendizaje no supervisado. En tanto que el método del mejor modelo no supervisado es capaz de detectar posibles anomalías en los datos directamente de los datos originales en una sola etapa.

5.3.2 Metodologías genéricas

En [125,134,135] la premisa fundamental es desarrollar un modelo predictivo basado en el registro de datos del sensor y aplicar el modelo para medir en tiempo real los datos del flujo entrante de los sensores. Los autores prefieren el enfoque de aprendizaje no supervisado, porque no implica la especificación (arbitraria) de un marco de tiempo de pronóstico para la definición de las etiquetas de verdad (clase objetivo) excepto para propósitos de evaluación principalmente, así como debido a las muchas razones que causan fallas, lo que conduce a un comportamiento diferente de cada señal ante diferentes incidentes, incluso del mismo tipo. Los métodos (no supervisados) empleados fueron el Isolation Forest y Elliptic Envelope. Las razones de los autores para esta selección son las siguientes: eficiencia en términos de tiempo y de cálculo de memoria dado el gran volumen de datos.

Esta estrategia es diferente porque los autores seleccionan el modelo manualmente, en tanto que el método para la selección del mejor modelo propuesto en esta tesis establece una metodología para determinar el mejor modelo no supervisado para un conjunto de datos dado.

5.3.3 Método único

En [124] los autores presentan un algoritmo totalmente autónomo, aplicable al problema de la detección de fallas en los procesos industriales conocido como TEDA. Este algoritmo no supervisado analiza la densidad de cada muestra de datos leídos, que se calcula en función de la distancia entre esa muestra y todas las demás leídas hasta el momento, sin requerir de un conocimiento previo sobre los datos. Los autores presentan un caso de uso en el escenario de monitoreo en tiempo real. Generalmente, los datos en un proceso industrial se obtienen de manera continua, en tiempo real y, por lo tanto, los métodos de detección de valores atípicos deben poder manejar los datos en forma de flujos de datos. Por tanto, cada muestra analizada tiene un aspecto temporal y solo está disponible en el instante de la adquisición. En este contexto, se detecta un valor atípico a partir de la observación de una secuencia de muestras de datos analizadas a lo largo del tiempo.

Básicamente este algoritmo es una propuesta que cae entre los métodos para detectar anomalías no supervisadas. Entre las limitaciones de TEDA los autores mencionan las siguientes: el enfoque propuesto podría no ser totalmente adecuado para fallas incipientes / graduales / suaves, que podrían no distinguirse fácilmente. Además, podría ser sensible a la oscilación natural de la señal, especialmente si el "concepto de normalidad" no está bien establecido en relación con el comportamiento significativamente largo.

La diferencia fundamental es que el método propuesto en este trabajo para la selección del algoritmo parte de un conjunto de algoritmos y con base en la metodología establecida permite determinar el mejor modelo para un conjunto de datos.

5.3.4 Comparativa con el conjunto de datos del caso práctico

A continuación se compara la aplicación del método para la selección del mejor modelo en el conjunto de datos presentados tomado de [136], en relación con otros autores que hayan utilizado el mismo conjunto de datos.

En la **Tabla 23** se presenta un resumen de la metodología y método utilizado en relación con [136].

Tabla 23. Literatura relacionada con el caso de uso reportado en [136].

Ref.	Metodología	Método
[126]	Probar varios modelos utilizando los datos de entrenamiento de validación cruzada y, a continuación, evaluar su rendimiento en función de su capacidad para pronosticar errores en los datos de prueba de validación cruzada.	Se trataron varios algoritmos de aprendizaje supervisado: vecinos más cercanos al K (KNN), bahías ingenuas, máquina de refuerzo de gradiente (GBM), bosque aleatorio, regresión logística penalizada, etc. En el algoritmo final, los autores usaron máquina de refuerzo de gradiente, bosque aleatorio y regresión logística penalizada.
[128]	Extraer las características relevantes en función de la interpretación física de los datos. A continuación, proponer un clasificador basado en el análisis discriminante de Fisher (FDA, en inglés) para incorporar datos incompletos.	Los autores proponen una técnica de recuperación de registro de errores para el diagnóstico de errores.
[137]	El primer paso es extraer características útiles de los datos sin procesar para facilitar la detección. El segundo paso es construir el modelo del clasificador de árbol de decisión. Tercer paso es desarrollar el algoritmo.	Los autores adoptaron el clasificador de árbol de decisión para predecir el tipo de falla. Más específicamente, fue usado el Bosque aleatorio y el Árbol de decisión de aumento de gradiente como clasificadores.
Propio	En la primera parte se lleva a cabo el preprocesamiento de los datos. En la segunda parte se aplica el algoritmo propuesto en la sección 3.3.2 (página 71)	El algoritmo selecciona el modelo con la menor varianza y sesgo de los modelos posibles, considerando los detectores base CBLOF, HBOS, OCSVM y el conjunto de parámetros descritos en la Tabla 15, en la página 76.

En [126], la estrategia de los autores fue combinar diferentes métodos de aprendizaje supervisado en la detección de fallas. Los autores han tenido acceso a

datos completos de eventos de falla y que fueron utilizados para entrenar el modelo. En [128] los autores realizaron en una primera etapa una interpretación física de los datos dados para seleccionar las características adecuadas para un clasificador de fallas. En una segunda etapa, se empleó el análisis discriminante de Fisher (FDA, por sus siglas en inglés de Fisher Discriminant Analysis) para minimizar el efecto de los valores atípicos en los conjuntos de datos incompletos. Finalmente, se recuperaron el tipo de los registros de fallas faltantes y la duración de las fallas correspondientes.

Este enfoque es adecuado cuando se tiene datos históricos con información de fallas, además de los recursos necesarios para hacer un análisis profundo de los datos y probar con diferentes estrategias para seleccionar la combinación de modelos más adecuada. Sin embargo, como ya se ha mencionado, existe en la industria dificultades para contar con estos recursos, por lo que es deseable hacer un diagnóstico inteligente de fallas menos dependiente de conocimientos previos y experiencia diagnóstica al procesar macrodatos. Por lo que este problema fue abordado con el método propuesto en esta tesis para la selección del mejor modelo al no disponer de los datos de fallas requeridos para entrenar el problema. Es decir, los datos públicos disponibles en este momento de [127], carecen de la información de fallas necesarias para utilizar métodos supervisados ya que estos requieren datos etiquetados con fallas para entrenar los modelos.

En [137] los autores proponen una clasificación mediante tres pasos clave que incluyen: 1) limpieza de datos y alineación del tiempo del evento; 2) extracción de características; 3) aplicación de los clasificadores de árboles de decisión por conjuntos. Este problema es abordado en forma similar a lo expuesto previamente en los métodos no supervisados para obtener características y requiere al menos un mínimo de datos históricos para poder extraer las características de operación normal del proceso, en tanto el método del mejor modelo no supervisado es capaz

de detectar posibles anomalías en los datos directamente de los datos originales en una sola etapa. Esta estrategia no puede ser comparada con relación a su eficacia porque los autores en [137] utilizan la matriz de confusión para comparar los resultados obtenidos por su modelo y los datos de fallas conocidas, lo cual un método no supervisado para detectar anomalías no puede proporcionar.

Capítulo 6

Conclusiones, metas y futuras líneas de investigación

Este capítulo presenta las conclusiones derivadas de la investigación realizada, las metas alcanzadas y las futuras líneas de investigación.

6.1 Conclusiones

En esta tesis se han identificado y caracterizado los principales problemas, metodologías, estrategias y tecnologías libres presentes en la integración de la convergencia de tecnologías de analítica de Big Data en la Industria 4.0. Posteriormente, una arquitectura de referencia ha sido presentada basada en la gestión de modelos de datos. Se adoptó un enfoque ADD para recopilar atributos industriales de Big Data (requisitos funcionales) y los impulsores industriales de la gestión del Big Data (atributos de calidad de datos) de los sistemas de planeación,

producción y mantenimiento preventivo (diferentes partes interesadas) para la especificación de la arquitectura, que se pueda adaptar fácilmente a los dominios de aplicación de analítica industrial de Big Data.

Para validar la arquitectura de referencia propuesta se describieron diferentes escenarios de fallas en la industria, a través de aplicaciones reales tomadas de la literatura, para ilustrar diferentes casos de uso en contextos industriales que mostraron cómo se puede aplicar la arquitectura. También se describió una de las muchas aplicaciones posibles para escenarios con apoyo de un caso real reportado en la literatura. Se detallaron las interacciones entre los componentes de la arquitectura propuestos y los elementos de las soluciones de los casos de uso.

Como aportación científica se desarrolló una metodología para apoyar la selección del modelo no supervisado para la detección de anomalías. Con este propósito se creó un algoritmo que permite entrenar a varios detectores base con diferentes parámetros para determinar el mejor modelo predictor de fallas cuando se desconoce la etiqueta de verdad en los datos. Para validar dicho algoritmo se aplicó a un conjunto de datos reales para predecir fallas en series temporales de mediciones de sensores y señales de referencia de control para cada uno de varios componentes de control de una planta industrial, y datos de series temporales que representen mediciones de energía eléctrica de un número fijo de zonas de la planta durante el mismo período de tiempo, para predecir el conjunto de posibles fallas para cada planta.

Además, se validó la arquitectura de referencia y el modelo basado en datos a través de un prototipo de plataforma tecnológica para el escenario de fallas en la industria con el caso de uso que aplica el método desarrollado para encontrar el mejor modelo de detección temprana de fallas, con el fin de integrar las dos principales aportaciones de este trabajo de tesis.

Con lo anterior, se puede establecer que la principal aportación de esta investigación ha sido la definición de un marco de trabajo que sirve para agilizar y facilitar el diseño de soluciones de Big Data industrial, al proveer herramientas que ayudan en dos de las etapas que más tiempo y recursos consumen: 1) el diseño de una arquitectura que permita integrar fases y herramientas para brindar soluciones a escenarios de uso concretos, así como la selección de algoritmos de aprendizaje adecuados para el análisis de los datos en cada escenario o caso concreto, lo cual en conjunto sienta las bases para un enfoque metodológico que guíe el diseño de sistemas de gestión de datos para analítica inteligente en la industria, lo cual es el enfoque de esta tesis.

La caracterización de la gestión de datos desarrollado a partir de la revisión sistemática de literatura en relación con el modelado de datos para proveer servicios de optimización a través de Big Data en sistemas ciberfísicos en la Industria 4.0, satisface las siguientes consideraciones:

- La gestión de datos satisface el ciclo de vida de Big Data en la industria 4.0.
- Considera la convergencia tecnológica del Big Data industrial en el contexto de los sistemas ciberfísicos industriales: El Internet Industrial de las Cosas, el Computo en la Nube para la industria y modelado basado en datos.
- Los requerimientos de calidad o los atributos de calidad necesarios para el diseño de analítica de Big Data industrial. Estos atributos de calidad son los siguientes: (1) Integración de fuentes de datos iCPS, (2) Soporte de decisiones basado en análisis, (3) Procesamiento de datos escalable y elástico, (4) Composición de los eventos basados en datos, (5) Servicios de datos de optimización, (6) Análisis integrado, (7) Soporte de decisiones basado en análisis.

De los resultados obtenidos en el diseño de la arquitectura de referencia para analítica de Big Data en la Industria 4.0 se concluye lo siguiente:

- La arquitectura de referencia se pueda adaptar a diferentes escenarios de aplicación basados en los requerimientos o atributos de calidad de la gestión de datos en la industria. Los escenarios para los atributos de Big Data industrial son un aporte que ayuda a establecer los requerimientos que debe cumplir una solución de analítica industrial.
- La arquitectura de referencia soporta el procesamiento de datos por lotes y el procesamiento de flujo de datos en tiempo real.
- El diseño de la arquitectura de referencia considera el modelado de datos, la inferencia de los indicadores clave de rendimiento, las funciones predictivas y los eventos basados en datos para el análisis en tiempo real.
- La arquitectura de referencia propuesta facilita el desarrollo de aplicaciones de Big Data industrial en una amplia gama de escenarios industriales para análisis prescriptivo, incluido el diagnóstico de fallas, el monitoreo en tiempo real y el pronóstico de condiciones inusuales del proceso.

De los resultados obtenidos en la validación de la arquitectura de referencia a través de escenarios para fallas industriales, la arquitectura de referencia fue validada en diferentes escenarios tomados de la literatura:

- En el diseño de aplicaciones que proporcionen analítica para iCPS a través de los registros de rendimiento de la maquinaria, mediante servicios de optimización y diagnóstico de fallas.
- En el diseño de aplicaciones que proporcionen analítica para iCPS a través de detectar eventos basados en datos en el flujo de datos provenientes del IIoT para el monitoreo del estado de salud de la maquinaria/proceso.

- En el diseño de aplicaciones de analítica de Big Data que detecten condiciones inusuales en la maquinaria/proceso a través de combinar los servicios de optimización y el monitoreo en tiempo real.

De la caracterización de los métodos de aprendizaje para la detección de anomalías se concluye lo siguiente:

- Se identificaron y documentaron los algoritmos más comunes de aprendizaje máquina supervisados y no supervisados para la detección de anomalías.
- Se obtuvo un conjunto base de métodos no supervisados para la detección de anomalías: HBOS, CBLOF y OCSVM. Para lograr lo anterior, se evaluaron los algoritmos de aprendizaje máquina no supervisados para la detección de anomalías al comparar el desempeño de los diferentes predictores con distintas opciones de hiper parámetros, en quince conjuntos de datos reales modificados para incluir la etiqueta de verdad.

De los resultados obtenidos en la validación del modelo basado en datos para la detección de posibles fallas se concluye lo siguiente:

- El algoritmo para la selección del mejor modelo no supervisado fue evaluado en un conjunto de datos de mediciones de sensores y señales de referencia de control para cada uno de varios componentes de control de la planta y mediciones de energía eléctrica de diferentes zonas de la planta.
- En cada planta se seleccionó el mejor modelo y sus parámetros del conjunto de modelos M (HBOS, CBLOF, OCSVM), para predecir posibles fallas en el conjunto de datos.

De los resultados obtenidos en el diseño del prototipo de plataforma tecnológica, se concluye lo siguiente:

- En el escenario de datos históricos para analítica de Big Data se ha validado la gestión de los datos generados por los componentes y zonas almacenadas en HDFS. Este sistema de analítica de Big Data industrial incorpora dos servicios basados en almacenamiento HDFS. El primer servicio, Data Analytics Studio (DAS), extrae información basada en consultas SQL que permite generar vistas y nuevas tablas. El segundo servicio permite el análisis con Spark mediante un cuaderno de trabajo Zeppelin basado en la web para el análisis de datos en forma interactiva. De esta forma es posible realizar consultas a la información almacenada en HDFS en forma distribuida.
- El escenario de flujo de datos para analítica de Big Data industrial permite habilitar un entorno inteligente con el fin de que los datos producidos por los sensores puedan ser utilizados en eventos actualizados de manera activa. Este sistema incorpora los servicios de Apache Kafka y Apache Nifi para desarrollar un mecanismo de integración en la ingesta de datos que permita un flujo controlado de eventos que asegure la persistencia y el control mediante un búfer de los datos a tasas muy altas de transacciones. Por otro lado, el uso de SAM para crear una aplicación de análisis del flujo de datos desplegada en Storm como motor de análisis de flujo de datos permite el análisis de manera más rápida y sencilla. Los resultados de este procesamiento son mostrados en Apache Superset que permite la exploración y visualización de los datos.
- Además, las plataformas de código abierto utilizadas en la instanciación de la arquitectura de referencia han probado ser adecuadas para el manejo de grandes volúmenes de datos generados por el IIoT.

6.2 Metas

Las metas alcanzadas son las siguientes:

6.2.1 Artículos de revista con factor de impacto, Q1

- **Hinojosa-Palafox, E.A.**; Rodríguez-Elías, O.M.; Hoyo-Montaño, J.A.; Pacheco-Ramírez, J.H.; Nieto-Jalil, J.M. An Analytics Environment Architecture for Industrial Cyber-Physical Systems Big Data Solutions. *Sensors* 2021, 21, 4282. <https://doi.org/10.3390/s21134282>.

6.2.2 Congresos

- **E.A. Hinojosa-Palafox**, O.M. Rodriguez-Elias, J.A. Hoyo-Montano, J.H. Pacheco-Ramirez, Towards an Architectural Design Framework for Data Management in Industry 4.0, in: Proc. - 2019 7th Int. Conf. Softw. Eng. Res. Innov. CONISOFT 2019, IEEE, 2019: pp. 191–200. <https://doi.org/10.1109/CONISOFT.2019.00035>.
- **Hinojosa-Palafox, E.A.**; Rodríguez-Elías, O.M.; Hoyo-Montaño, J.A.; Pacheco-Ramírez, J.H. Trends and Challenges of Data Management in Industry 4.0. In *LISS2019*; Zhang, J., Dresner, M., Zhang, R., Hua, G., Shang X., Ed.; Springer: Singapore, 2020.

6.2.3 Capítulos de libro

- **Hinojosa-Palafox E.A.**, Rodríguez-Elías O.M., Hoyo-Montaño J.A., Pacheco-Ramírez J.H. (2020) Trends and Challenges of Data Management in Industry 4.0. In: Zhang J., Dresner M., Zhang R., Hua G., Shang X. (eds) *LISS2019*. Springer, Singapore. https://doi.org/10.1007/978-981-15-5682-1_16

6.2.4 *Proyectos*

- Diseño y validación de un algoritmo para la selección de algoritmos no supervisados para la detección de anomalías, registrado ante el Tecnológico Nacional de México con clave 9366.20-P.

Desarrollo de una arquitectura para el diseño de soluciones informáticas para analítica de Big Data en la industria, registrado ante el Tecnológico Nacional de México con clave 10980.21-P.

6.3 **Futuras líneas de investigación**

Como parte del trabajo futuro, es posible considerar diferentes líneas de investigación que amplíen y mejoren la investigación presentada en esta tesis de doctorado, en diferentes dominios de la Industria 4.0.

A continuación, se muestran futuras líneas de investigación posibles a considerar.

6.3.1 *Diseño de arquitectura para analítica industrial*

Los siguientes aspectos de investigación significativos en el diseño de la arquitectura para el análisis de Big Data en iCPS se encontraron en la revisión de la literatura y no se abordaron en esta tesis, por lo que deben ser considerados para futuras investigaciones:

- Control descentralizado en iCPS

El enfoque presentado en este trabajo se enmarca en el paradigma de control centralizado que integra iCPS, inteligencia artificial y analítica de Big Data, y modelos basados en datos que permiten la analítica del ciclo de vida de los datos de manufactura.

Una tendencia emergente es un control descentralizado que utiliza modelos adaptativos que admiten una rápida alteración del modelo que no requiere actualizaciones manuales. Este nuevo enfoque de los problemas surge al considerar

el siguiente problema: los procesos cambian con el tiempo en sistemas complejos como el flujo de materiales de las piezas de los productos y las operaciones logísticas asociadas. La cuestión que se plantea es ¿cómo puede integrarse la analítica de Big Data en el enfoque de control descentralizado de iCPS?

- Preocupación por la privacidad de la analítica de iCPS

Una preocupación subyacente en la seguridad de los datos es la privacidad de los datos industriales relacionada con el manejo de datos sensibles sin tomar medidas de seguridad, como la consideración del procesamiento analítico de Big Data oculto. Un servidor externo se ha convertido en un potencial puente de seguridad para acceder a información confidencial desde él. Por lo tanto, las preocupaciones de privacidad planteadas en el análisis de Big Data para conceder el control de los datos a la nube disminuyen la confidencialidad, ya que los datos son probablemente almacenados, procesados y analizados en varios centros de la nube, lo que lleva a las preocupaciones de seguridad en distintas ubicaciones de los datos. Entonces, la pregunta que surge es, ¿cómo maneja la arquitectura de análisis de Big Data los datos sensibles en contextos de iCPS?

6.3.2 Modelos basados en datos para la detección temprana de fallas en la industria

Actualmente existe un notable interés en desarrollar soluciones que consideren los beneficios de los modelos basados en datos. La detección y diagnóstico de fallas es un importante tema que puede ser abordado con distintos enfoques y que plantea algunos retos y desafíos, particularmente en el flujo de datos en tiempo real, lo que hace necesario considerar nuevos enfoques que permitan la salud adaptativa de las máquinas/procesos y que ayude a la solución de problemas que pasan desapercibidos y que potencialmente son costosos o ponen en riesgo a las personas.

Referencias

1. Skilton, M.; Hovsepian, F. *Mastering the 4th Industrial Revolution*; Springer International Publishing AG: Cham, Switzerland, 2016; ISBN 9788740318838.
2. Drath, R.; Horch, A. Industrie 4.0: Hit or Hype? [Industry Forum]. *IEEE Industrial Electronics Magazine* **2014**, *8*, 56–58, doi:10.1109/MIE.2014.2312079.
3. Zezulka, F.; Marcon, P.; Vesely, I.; Sajdl, O. Industry 4.0 – An Introduction in the Phenomenon. *IFAC-PapersOnLine* **2016**, *49*, 8–12, doi:10.1016/j.ifacol.2016.12.002.
4. Ochs, Th.; Riemann, U. Smart Manufacturing in the Internet of Things Era. In *Internet of Things and Big Data Analytics Toward Next-Generation Intelligence*; Cham, Ed.; Springer, 2017; pp. 199–217.
5. Chauhan, B.; Bhatt, C. Bigdata Analytics in Industrial IoT. In *Internet of Things and Big Data Analytics Toward Next-Generation Intelligence*; Janusz Kacprzyk, Polish Academy of Sciences, Warsaw, P., Ed.; Springer International Publishing AG: Cham, Switzerland, 2018; pp. 381–406.
6. Jazdi, N. Cyber Physical Systems in the Context of Industry 4.0. In Proceedings of the 2014 IEEE international conference on automation, quality and testing, robotics; IEEE, 2014; pp. 1–4.
7. Lee, J.; Ardakani, H.D.; Yang, S.; Bagheri, B. Industrial Big Data Analytics and Cyber-Physical Systems for Future Maintenance & Service Innovation. In Proceedings of the Procedia CIRP; Elsevier B.V., 2015; Vol. 38, pp. 3–7.

8. Lee, J.; Bagheri, B.; Kao, H.-A. Recent Advances and Trends of Cyber-Physical Systems and Big Data Analytics in Industrial Informatics. In Proceedings of the Int. Conference on Industrial Informatics (INDIN); 2014; pp. 1–6.
9. Lee, J.; Bagheri, B.; Kao, H.A. A Cyber-Physical Systems Architecture for Industry 4.0-Based Manufacturing Systems. *Manufacturing Letters* **2015**, *3*, 18–23, doi:10.1016/j.mfglet.2014.12.001.
10. Sharma, A.B.; Ivančić, F.; NiculescuMizil, A.; Chen, H.; Jiang, G. Modeling and Analytics for Cyber-Physical Systems in the Age of Big Data. *Performance Evaluation Review* **2014**, *41*, 74–77, doi:10.1145/2627534.2627558.
11. Tao, F.; Qi, Q.; Liu, A.; Kusiak, A. Data-Driven Smart Manufacturing. *Journal of Manufacturing Systems* **2018**, *48*, 157–169, doi:10.1016/j.jmsy.2018.01.006.
12. Yin, S.; Ding, S.X.; Xie, X.; Luo, H. A Review on Basic Data-Driven Approaches for Industrial Process Monitoring. *IEEE Transactions on Industrial Electronics* **2014**, *61*, 6418–6428, doi:10.1109/TIE.2014.2301773.
13. Roblek, V.; Meško, M.; Krapež, A. A Complex View of Industry 4.0. *SAGE Open* **2016**, *6*, 215824401665398, doi:10.1177/2158244016653987.
14. Grier, D.A. The Radical Technology of Industrie 4.0. *Computer* **2017**, *50*, 120–120, doi:10.1109/mc.2017.109.
15. Hermann, M.; Pentek, T.; Otto, B. Design Principles for Industrie 4.0 Scenarios. In Proceedings of the Proceedings of the Annual Hawaii International Conference on System Sciences; IEEE Computer Society, March 7 2016; Vol. 2016-March, pp. 3928–3937.
16. Lade, P.; Ghosh, R.; Srinivasan, S. Manufacturing Analytics and Industrial Internet of Things. *IEEE Intelligent Systems* **2017**, *32*, 74–79, doi:10.1109/MIS.2017.49.

17. Madakam, S., Ramaswamy, R. and Tripathi, S. Internet of Things (IoT): A Literature Review. *Journal of Computer and Communications* **2015**, 164–173, doi:<http://dx.doi.org/10.4236/jcc.2015.35021>.
18. Yan, J., Meng, Y., Lu, L., & Li, L. Industrial Big Data in an Industry 4.0 Environment: Challenges, Schemes, and Applications for Predictive Maintenance. *IEEE Access* **2017**, 5, 23484–23491, doi:10.1109/ACCESS.2017.2765544.
19. Jeschke, S.; Brecher, C.; Meisen, T.; Özdemir, D.; Eschert, T. Industrial Internet of Things and Cyber Manufacturing Systems. In; 2017; pp. 3–19.
20. O'Donovan, P.; Leahy, K.; Bruton, K.; O'Sullivan, D.T.J. Big Data in Manufacturing: A Systematic Mapping Study. *Journal of Big Data* **2015**, 2, 20, doi:10.1186/s40537-015-0028-x.
21. Santos, M.Y.; Martinho, B.; Costa, C. Modelling and Implementing Big Data Warehouses for Decision Support. *Journal of Management Analytics* **2017**, 4, 111–129, doi:10.1080/23270012.2017.1304292.
22. Vora, R.; Garala, K.; Raval, P. An Era of Big Data on Cloud Computing Services as Utility: 360° of Review, Challenges and Unsolved Exploration Problems. In Proceedings of the Smart Innovation, Systems and Technologies; Springer Science and Business Media Deutschland GmbH, 2016; Vol. 51, pp. 563–574.
23. Givehchi, O.; Trsek, H.; Jasperneite, J. Cloud Computing for Industrial Automation Systems - A Comprehensive Overview. In Proceedings of the IEEE International Conference on Emerging Technologies and Factory Automation, ETFA; IEEE, 2013; pp. 1–4.
24. Huang, B.; Li, C.; Yin, C.; Zhao, X. Cloud Manufacturing Service Platform for Small- and Medium-Sized Enterprises. *International Journal of Advanced*

- Manufacturing Technology* **2013**, 65, 1261–1272, doi:10.1007/s00170-012-4255-4.
25. Wuest, T.; Weimer, D.; Irgens, C.; Thoben, K.D. Machine Learning in Manufacturing: Advantages, Challenges, and Applications. *Production and Manufacturing Research* **2016**, 4, 23–45, doi:10.1080/21693277.2016.1192517.
 26. Marrella, A.; Monreale, A.; Kloepper, B.; Krueger, M.W. Privacy-Preserving Outsourcing of Pattern Mining of Event-Log Data - A Use-Case from Process Industry. In Proceedings of the 2016 IEEE International Conference on Cloud Computing Technology and Science (CloudCom); IEEE, December 2016; pp. 545–551.
 27. Fahmideh, M.; Beydoun, G. Big Data Analytics Architecture Design—An Application in Manufacturing Systems. *Computers and Industrial Engineering* **2019**, 128, 948–963, doi:10.1016/j.cie.2018.08.004.
 28. Diez-Olivan, A.; Del Ser, J.; Galar, D.; Sierra, B. Data Fusion and Machine Learning for Industrial Prognosis: Trends and Perspectives towards Industry 4.0. *Information Fusion* **2019**, 50, 92–111, doi:10.1016/j.inffus.2018.10.005.
 29. Sharda, R.; Delen, D.; Turban, E. Business Intelligence, Analytics, and Data Science: A Managerial Perspective. **2016**.
 30. Sarnovsky, M.; Bednar, P.; Smatana, M. Big Data Processing and Analytics Platform Architecture for Process Industry Factories. *Big Data and Cognitive Computing* **2018**, 2, 3, doi:10.3390/bdcc2010003.
 31. Atat, R.; Liu, L.; Wu, J.; Li, G.; Ye, C.; Yang, Y. Big Data Meet Cyber-Physical Systems: A Panoramic Survey. *IEEE Access* **2018**, 6, 73603–73636, doi:10.1109/ACCESS.2018.2878681.

32. Lee, J.; Bagheri, B.; Kao, H.A. A Cyber-Physical Systems Architecture for Industry 4.0-Based Manufacturing Systems. *Manufacturing Letters* **2015**, *3*, 18–23, doi:10.1016/j.mfglet.2014.12.001.
33. Hinojosa-Palafox, E.A.; Rodríguez-Elías, O.M.; Hoyo-Montaño, J.A.; Pacheco-Ramírez, J.H. *Trends and Challenges of Data Management in Industry 4.0*; Zhang J., Dresner M., Zhang R., Hua G., S.X., Ed.; Springer Singapore, 2020;
34. Borrison, R.; Klöpper, B.; Chioua, M.; Dix, M.; Sprick, B. Reusable Big Data System for Industrial Data Mining - A Case Study on Anomaly Detection in Chemical Plants. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer Nature Switzerland, 2018; Vol. 11314 LNCS, pp. 611–622.
35. Amruthnath, N.; Gupta, T. A Research Study on Unsupervised Machine Learning Algorithms for Early Fault Detection in Predictive Maintenance. In Proceedings of the 2018 5th International Conference on Industrial Engineering and Applications, ICIEA 2018; IEEE, 2018; pp. 355–361.
36. Pimminger, S.; Wagner, S.; Kurschl, W.; Heinzlreiter, J. Optimization as a Service: On the Use of Cloud Computing for Metaheuristic Optimization. In; 2013; pp. 348–355.
37. Parra, V.M.; Halgamuge, M.N. Performance Evaluation of Big Data and Business Intelligence Open Source Tools: Pentaho and Jaspersoft. In; 2018; pp. 147–176.
38. Kitchenham, B.; Charters, S. *Guidelines for Performing Systematic Literature Reviews in Software Engineering*; 2007; Vol. 5;.

39. Whitmore, A.; Agarwal, A.; Da Xu, L. The Internet of Things—A Survey of Topics and Trends. *Information Systems Frontiers* **2015**, *17*, 261–274, doi:10.1007/s10796-014-9489-2.
40. Sharma, S. Expanded Cloud Plumes Hiding Big Data Ecosystem. *Future Generation Computer Systems* **2016**, *59*, 63–92, doi:https://doi.org/10.1016/j.future.2016.01.003.
41. Pertel, V.M.; Saturno, M.; Deschamps, F.; Loures, E.D.R. Analysis of It Standards and Protocols for Industry 4.0. *24th International Conference on Production Research* **2017**, 622–628, doi:10.12783/dtetr/icpr2017/17681.
42. Atif, M. U., & Shah, M.A. OptiSEC: In Search of an Optimal Sensor Cloud Architecture. In Proceedings of the Proceedings of the 23rd International Conference on Automation & Computing; 2017; p. 1.6.
43. Jararweh, Y., Al-Ayyoub, M., Benkhelifa, E., Vouk, M., & Rindos, A. SDIoT: A Software Defined Based Internet of Things Framework. *Journal of Ambient Intelligence and Humanized Computing* **2015**, *6*, 453–461.
44. Stankevichus, I. Data Acquisition as Industrial Cloud Service. *Master These* **2016**, 41.
45. Basanta-Val, P. An Efficient Industrial Big-Data Engine. *IEEE Transactions on Industrial Informatics* **2018**, *14*, 1361–1369, doi:10.1109/TII.2017.2755398.
46. Santos, M.Y.; Oliveira e Sá, J.; Costa, C.; Galvão, J.; Andrade, C.; Martinho, B.; Lima, F.V.; Costa, E. A Big Data Analytics Architecture for Industry 4.0. In Proceedings of the Advances in Intelligent Systems and Computing; 2017; Vol. 570, pp. 175–184.
47. Chen, K.; Li, X.; Wang, H. On the Model Design of Integrated Intelligent Big Data Analytics Systems. *Industrial Management & Data Systems* **2015**, *115*, 1666–1682, doi:10.1108/IMDS-03-2015-0086.

48. Mishra, N.; Lin, C.C.; Chang, H.T. A Cognitive Adopted Framework for IoT Big-Data Management and Knowledge Discovery Prospective. *International Journal of Distributed Sensor Networks* **2015**, *2015*, doi:10.1155/2015/718390.
49. Cao, B.; Wang, Z.; Shi, H.; Yin, Y. Research and Practice on Aluminum Industry 4.0. In Proceedings of the Proceedings of 6th International Conference on Intelligent Control and Information Processing, ICICIP 2015; Institute of Electrical and Electronics Engineers Inc.: Wuhan, China, January 20 2016; pp. 517–521.
50. Li, J.; Tao, F.; Cheng, Y.; Zhao, L. Big Data in Product Lifecycle Management. *International Journal of Advanced Manufacturing Technology* **2015**, *81*, 667–684, doi:10.1007/s00170-015-7151-x.
51. Bai, Y.; Sun, Z.; Deng, J.; Li, L.; Long, J.; Li, C. Manufacturing Quality Prediction Using Intelligent Learning Approaches: A Comparative Study. *Sustainability* **2017**, *10*, 1–15, doi:10.3390/su10010085.
52. Gustavsson, M.; Wänström, C. Assessing Information Quality in Manufacturing Planning and Control Processes. *International Journal of Quality and Reliability Management* **2009**, *26*, 325–340, doi:10.1108/02656710910950333.
53. Bass, L.; Clements, P.; Kazman, R. *Software Architecture in Practice, Third Edit*; Upper Sadd.; Addison Wesley: NJ., 2013; ISBN 0321154959.
54. Hinojosa-Palafox, E.A.; Rodriguez-Elias, O.M.; Hoyo-Montano, J.A.; Pacheco-Ramirez, J.H. Towards an Architectural Design Framework for Data Management in Industry 4.0. In Proceedings of the Proceedings - 2019 7th International Conference in Software Engineering Research and Innovation, CONISOFT 2019; IEEE, 2019; pp. 191–200.

55. Kazman, R.; Klein, M.; Clements, P. ATAM: Method for Architecture Evaluation. *Cmusei* **2000**, *4*, 83, doi:(CMU/SEI-2000-TR-004, ADA382629).
56. Zhou, Z.H. *Ensemble Methods: Foundations and Algorithms*; Ralf Herbrich and Thore Graepel (Microsoft Research Ltd.), Ed.; Chapman and Hall/CRC: 6000 Broken Sound Parkway NW, 2012; ISBN 9781439830055.
57. Ademujimi, T.T.; Brundage, M.P.; Prabhu, V. V. A Review of Current Machine Learning Techniques Used in Manufacturing Diagnosis. In Proceedings of the IFIP Advances in Information and Communication Technology; Springer International Publishing, Cham, 2017; Vol. 513, pp. 407–415.
58. Ademujimi, T.T.; Brundage, M.P.; Prabhu, V. V. A Review of Current Machine Learning Techniques Used in Manufacturing Diagnosis. In Proceedings of the IFIP Advances in Information and Communication Technology; 2017; Vol. 513, pp. 407–415.
59. Bagozi, A.; Bianchini, D.; De Antonellis, V.; Marini, A.; Ragazzi, D. Big Data Summarisation and Relevance Evaluation for Anomaly Detection in Cyber Physical Systems. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Springer Verlag, 2017; pp. 429–447.
60. Bagozi, A.; Bianchini, D.; De Antonellis, V.; Marini, A. Big Data Exploration for Smart Manufacturing Applications. In *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*; Springer Nature, 2018; Vol. 11234 LNCS, pp. 487–501 ISBN 9783030029241.
61. Bagozi, A.; Bianchini, D.; De Antonellis, V.; Marini, A.; Ragazzi, D. Summarisation and Relevance Evaluation Techniques for Big Data Exploration: The Smart Factory Case Study. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); 2017.

62. Chandola, V.; Banerjee, A.; Kumar, V. Anomaly Detection: A Survey. *ACM Computing Surveys* **2009**, *41*, doi:10.1145/1541880.1541882.
63. Karkouch, A.; Mousannif, H.; Al Moatassime, H.; Noel, T. Data Quality in Internet of Things: A State-of-the-Art Survey. *Journal of Network and Computer Applications* **2016**, *73*, 57–81, doi:10.1016/j.jnca.2016.08.002.
64. Lazarevic, A.; Kumar, V. Feature Bagging for Outlier Detection. In Proceedings of the Proceeding of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining - KDD '05; ACM Press: New York, New York, USA, 2005; p. 157.
65. Aggarwal, C.C. *Outlier Analysis*. In *Data Mining*; Springer, Cham., 2015;
66. Zhao, Y.; Hryniewicki, M.K. XGBOD: Improving Supervised Outlier Detection with Unsupervised Representation Learning. *Proceedings of the International Joint Conference on Neural Networks* **2018**, 2018-July, doi:10.1109/IJCNN.2018.8489605.
67. Goldstein, M.; Uchida, S. A Comparative Evaluation of Unsupervised Anomaly Detection Algorithms for Multivariate Data. *PLoS ONE* **2016**, *11*, e0152173, doi:10.1371/journal.pone.0152173.
68. Wang, J.; Ma, Y.; Zhang, L.; Gao, R.X.; Wu, D. Deep Learning for Smart Manufacturing: Methods and Applications. *Journal of Manufacturing Systems* **2018**, *48*, 144–156, doi:10.1016/j.jmsy.2018.01.003.
69. Aggarwal, C.C. *Outlier Analysis*. In *Outlier Analysis*; Springer Nature: Cham, Switzerland, 2013; Vol. 9781461463, pp. 1–446 ISBN 9781461463962.
70. Hecht-nielsen, R. Replicator Neural Networks for Universal Optimal Source Coding. *Science, New Series*, **2016**, *269*, 1860–1863.
71. Holden, A.J.; Robbins, D.J.; Stewart, W.J.; Smith, D.R.; Schultz, S.; Wegener, M.; Linden, S.; Hormann, C.; Enkrich, C.; Soukoulis, C.M.; et al. Reducing the

- Dimensionality of of Data with Neural Networks. *Science Publications* **2006**, 313, 504–507.
72. Zhao, Y.; Nasrullah, Z.; Li, Z. PyOD: A Python Toolbox for Scalable Outlier Detection. *ournal of Machine Learning Research* **2019**, 20, 1–7.
73. Shyu, M.L.; Chen, S.C.; Sarinnapakorn, K.; Chang, L. Principal Component-Based Anomaly Detection Scheme. *Studies in Computational Intelligence* **2006**, 9, 311–329, doi:10.1007/11539827-18.
74. Hardin, J.; Rocke, D.M. Outlier Detection in the Multiple Cluster Setting Using the Minimum Covariance Determinant Estimator. *Computational Statistics and Data Analysis* **2004**, 44, 625–638, doi:10.1016/S0167-9473(02)00280-3.
75. Schölkopf, B.; Platt, J.C.; Shawe-Taylor, J.; Smola, A.J.; Williamson, R.C. Estimating the Support of a High-Dimensional Distribution. *Neural Computation* **2001**, 13, 1443–1471, doi:10.1162/089976601750264965.
76. Breunig, M.M., Kriegel, H.P., Ng, R.T. and Sander, J. LOF: Identifying Density-Based Local Outliers. *ACM sigmod record* **2000**, 29, 93–104, doi:10.1016/s0020-7292(09)60373-8.
77. Tang, J.; Chen, Z.; Fu, A.W.C.; Cheung, D.W. Enhancing Effectiveness of Outlier Detections for Low Density Patterns. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **2002**, 2336, 535–548.
78. Rfdo, O.D.; Ri, H.; Dqg, R.; Duelq, Q.; Duelq, Q.R.I.; Vlfdo, S.K.; Ri, V.; Rxwolhu, D.Q.; Ghvljqhg, L. V; Lv, Z.; et al. Discovering Cluster-Based Local Outliers. *Pattern Recognition Letters* **2003**, 24, 1641–1650.
79. Papadimitriou, S., Kitagawa, H., Gibbons, P. B., & Faloutsos, C. Loci: Fast Outlier Detection Using the Local Correlation Integral. In Proceedings of the Proceedings 19th International Conference on Data Engineering (Cat. No. 03CH37405); IEEE, 2003; pp. 315–326.

80. Goldstein, M.; Dengel, A. Histogram-Based Outlier Score (Hbos): A Fast Unsupervised Anomaly Detection Algorithm. *KI-2012: Poster and Demo Track 2012*, 59–63.
81. Ramaswamy, S.; Rastogi, R.; Shim, K. Efficient Algorithms for Mining Outliers from Large Data Sets. *SIGMOD Record (ACM Special Interest Group on Management of Data)* **2000**, 29, 427–438, doi:10.1145/335191.335437.
82. Angiulli, F.; Pizzuti, C. Fast Outlier Detection in High Dimensional Spaces. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)* **2002**, 2431 LNAI, 15–27.
83. Kriegel, H.-P.; Kroger, P.; Schubert, E.; Zimek, A. Outlier Detection in Axis-Parallel Subspaces of High Dimensional Data BT - Advances in Knowledge Discovery and Data Mining. *Advances in Knowledge Discovery and Data Mining* **2009**, 5476, 831–838.
84. Kriegel, H.; Schubert, M. Angle-Based Outlier Detection in High-Dimensional Data. In Proceedings of the Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining; ACM, 2008; pp. 444–452.
85. Janssens, J.H.M., Huszár, F., Postma, E.O. and van den Herik, H.J. *Stochastic Outlier Selection*; Tilburg, The Netherlands., 2012;
86. Liu, Y.; Li, Z.; Zhou, C.; Jiang, Y.; Sun, J.; Wang, M.; He, X. Generative Adversarial Active Learning for Unsupervised Outlier Detection. *EEE Transactions on Knowledge and Data Engineering*. **2019**, 1–13.
87. Achtert, E.; Kriegel, H.P.; Reichert, L.; Schubert, E.; Wojdanowski, R.; Zimek, A. Visual Evaluation of Outlier Detection Models. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and*

- Lecture Notes in Bioinformatics*) **2010**, 5982 LNCS, 396–399, doi:10.1007/978-3-642-12098-5_34.
88. Hofmann, M., & Klinkenberg, R. *RapidMiner: Data Mining Use Cases and Business Analytics Applications*; CRC Press, 2013;
 89. Komsta, L. Outliers: Tests for Outliers. R Package Version 0.14 Available online: <http://www.r-project.org>.
 90. Verma, C.; Pandey, R. Big Data Representation for Grade Analysis through Hadoop Framework. In Proceedings of the Proceedings of the 2016 6th International Conference - Cloud System and Big Data Engineering, Confluence 2016; Institute of Electrical and Electronics Engineers Inc, Ed.; IEEE, 2016; pp. 312–315.
 91. Alla, S. *Big Data Analytics with Hadoop 3*; Amey Varangaonkar, Shetty, V., Dsa, C., Sawant, S., Eds.; Packt Publishing Ltd.: Birmingham, UK, 2018; ISBN 9781788628846.
 92. Raj, P.; Raman, A.; Nagaraj, D.; Duggirala, S.; Systems, C. *High-Performance Big-Data Analytics. Computing Systems and Approaches*; Sammes, A.J. (Centre for Forensic Computing Cranfield University, Shrivenham Campus Swindon, U., Ed.; Springer, 2015; ISBN 9783319207438.
 93. Ramos López, J.I. Despliegue de Un Clúster Hadoop Con Cloudera En Un Sistema de Virtualización Basado En Proxmox, Universidad de Valladolid, 2019.
 94. Lu, J.; Feng, J. A Survey of Mapreduce Based Parallel Processing Technologies. *China Communications* **2014**, 11, 146–155, doi:10.1109/CC.2014.7085615.
 95. Ambari Available online: <https://ambari.apache.org/> (accessed on 24 May 2021).

96. Manage Projects Faster & Collaborate Better | Hive Available online: <https://hive.com/> (accessed on 17 June 2021).
97. Pig, A.; Started, G.; Involved, G. Welcome to Apache Pig! Available online: <https://pig.apache.org/> (accessed on 17 June 2021).
98. Foundation, A.S. Apache Spark™ - Unified Analytics Engine for Big Data. *Apache Spark* 2018.
99. Apache HBase – Apache HBase™ Home. *DZone Refcardz* 2015, 1–113.
100. Apache Tez – Welcome to Apache TEZ® Available online: <https://tez.apache.org/> (accessed on 8 July 2021).
101. Apache Kafka Available online: <https://kafka.apache.org/> (accessed on 8 July 2021).
102. GitHub - Apache/Hcatalog: Mirror of Apache HCatalog Available online: <https://github.com/apache/hcatalog> (accessed on 17 June 2021).
103. Welcome to Apache Solr - Apache Solr Available online: <https://solr.apache.org/> (accessed on 17 June 2021).
104. Apache NiFi. *wikipedia* 2019.
105. Apache ZooKeeper Available online: <https://zookeeper.apache.org/> (accessed on 24 May 2021).
106. Apache Storm. Apache Storm Available online: <https://storm.apache.org/> (accessed on 24 May 2021).
107. Streaming Analytics Manager Overview Available online: https://docs.cloudera.com/HDPDocuments/HDF3/HDF-3.3.0/sam-overview/content/streaming_analytics_manager_overview.html (accessed on 24 May 2021).

108. Schema Registry Overview | Confluent Documentation Available online: <https://docs.confluent.io/platform/current/schema-registry/index.html> (accessed on 24 May 2021).
109. Capilla, R.; Jansen, A.; Tang, A.; Avgeriou, P.; Babar, M.A. 10 Years of Software Architecture Knowledge Management: Practice and Future. *Journal of Systems and Software* **2016**, *116*, 191–205, doi:10.1016/j.jss.2015.08.054.
110. Bass, L.; Klein, M.; Bachmann, F. Quality Attribute Design Primitives and the Attribute Driven Design Method. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); 2002; Vol. 2290, pp. 169–186.
111. Rayana, S. Outlier Detection DataSets (ODDS) Available online: <http://odds.cs.stonybrook.edu/>.
112. Zhao, Y.; Rossi, R.A.; Akoglu, L. Automating Outlier Detection via Meta-Learning. *arXiv* **2020**.
113. Fawcett, T. An Introduction to ROC Analysis. *Pattern recognition letters* **2006**, *27*, 861–874.
114. Aggarwal, C.C.; Sathe, S. *Outlier Ensembles: An Introduction*; First edit.; Springer: Cham, Switzerland, 2017; ISBN 9783319547657.
115. Rayana, S.; Zhong, W.; Akoglu, L. Sequential Ensemble Learning for Outlier Detection: A Bias-Variance Perspective. *Proceedings - IEEE International Conference on Data Mining, ICDM* **2017**, 1167–1172, doi:10.1109/ICDM.2016.117.
116. F. Keller, E. Muller, K.B. HiCS: High-Contrast Subspaces for Density-Based Outlier Ranking. In Proceedings of the IEEE ICDE Conference; IEEE, 2012; pp. 1037–1048.

117. Marz, N. *Principles and Best Practices of Scalable Real-Time Data Systems*; 2015; Vol. 37;.
118. Raza, Ali; Zafar, Shaista; Rahman, Saeed Ur; Khattak, U. Software Architecture Evaluation Methods: A Comparative Study. *International Journal of Computing and Communication Networks* **2019**, 1, 1–9.
119. Lei, Y.; Jia, F.; Lin, J.; Xing, S.; Ding, S.X. An Intelligent Fault Diagnosis Method Using Unsupervised Feature Learning Towards Mechanical Big Data. *IEEE Transactions on Industrial Electronics* **2016**, 63, 3137–3147, doi:10.1109/TIE.2016.2519325.
120. Qi, G.; Zhu, Z.; Erqinhu, K.; Chen, Y.; Chai, Y.; Sun, J. Fault-Diagnosis for Reciprocating Compressors Using Big Data and Machine Learning. *Simulation Modelling Practice and Theory* **2018**, 80, 104–127, doi:10.1016/j.simpat.2017.10.005.
121. Madhusudana, C.K.; Budati, S.; Gangadhar, N.; Kumar, H.; Narendranath, S. Fault Diagnosis Studies of Face Milling Cutter Using Machine Learning Approach. *Journal of Low Frequency Noise Vibration and Active Control* **2016**, 35, 128–138, doi:10.1177/0263092316644090.
122. Vamsi, I.V.; Abhinav, N.; Verma, A.K.; Radhika, S. Random Forest Based Real Time Fault Monitoring System for Industries. In Proceedings of the 2018 4th International Conference on Computing Communication and Automation, ICCCA 2018; IEEE, 2018; pp. 1–6.
123. Ranjan, G.S.K.; Kumar Verma, A.; Radhika, S. K-Nearest Neighbors and Grid Search CV Based Real Time Fault Monitoring System for Industries. In Proceedings of the 2019 IEEE 5th International Conference for Convergence in Technology, I2CT 2019; IEEE, 2019; p. 5.
124. Bezerra, C.G.; Costa, B.S.J.; Guedes, L.A.; Angelov, P.P. An Evolving Approach to Unsupervised and Real-Time Fault Detection in Industrial

- Processes. *Expert Systems with Applications* **2016**, 63, 134–144, doi:10.1016/j.eswa.2016.06.035.
125. Kolokas, N.; Vafeiadis, T.; Ioannidis, D.; Tzovaras, D. A Generic Fault Prognostics Algorithm for Manufacturing Industries Using Unsupervised Machine Learning Classifiers. *Simulation Modelling Practice and Theory* **2020**, 103, 102109, doi:10.1016/j.simpat.2020.102109.
126. Xiao, W. A Probabilistic Machine Learning Approach to Detect Industrial Plant Faults: PHM15 Data Challenge. *Proceedings of the Annual Conference of the Prognostics and Health Management Society, PHM 2015*, 718–726.
127. Prognostics and Health Management Society PHM Data Challenge 2015 Available online: <https://www.phmsociety.org/events/conference/phm/15/data-challenge>.
128. Kim, H.; Ha, J.M.; Park, J.; Kim, S.; Kim, K.; Jang, B.C.; Oh, H.; Youn, B.D. Fault Log Recovery Using an Incomplete-Data-Trained FDA Classifier for Failure Diagnosis of Engineered Systems. In Proceedings of the Proceedings of the Annual Conference of the Prognostics and Health Management Society, PHM; 2015; pp. 736–745.
129. Chen, H.-M.; Kazman, R.; Haziyevev, S.; Hrytsay, O. Big Data System Development: An Embedded Case Study with a Global Outsourcing Firm. In Proceedings of the 2015 IEEE/ACM 1st International Workshop on Big Data Software Engineering; 2015; pp. 44–50.
130. Xiaofeng, L.; Jing, L. Research on Big Data Reference Architecture Model. In Proceedings of the 2020 3rd International Conference on Artificial Intelligence and Big Data, ICAIBD 2020; IEEE, 2020; pp. 205–209.
131. Trunzer, E.; Vogel-Heuser, B.; Chen, J.K.; Kohnle, M. Model-Driven Approach for Realization of Data Collection Architectures for Cyber-Physical Systems of

- Systems to Lower Manual Implementation Efforts. *Sensors (Switzerland)* **2021**, *21*, 1–20, doi:10.3390/s21030745.
132. Molano, J.I.R.; Bravo, L.E.C.; Santana, E.R.L. Data Architecture for the Internet of Things and Industry 4.0. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); Fukuoka, Japan, 2017; Vol. 10387 LNCS, pp. 283–293.
 133. Amruthnath, N.; Gupta, T. Fault Class Prediction in Unsupervised Learning Using Model-Based Clustering Approach. In Proceedings of the 2018 International Conference on Information and Computer Technologies, ICICT 2018; 2018; pp. 5–12.
 134. Kolokas, N.; Vafeiadis, T.; Ioannidis, D.; Tzovaras, D. Forecasting Faults of Industrial Equipment Using Machine Learning Classifiers. In Proceedings of the 2018 IEEE (SMC) International Conference on Innovations in Intelligent Systems and Applications, INISTA 2018; IEEE, 2018; pp. 1–6.
 135. Kolokas, N.; Vafeiadis, T.; Ioannidis, D.; Tzovaras, D. Anomaly Detection in Aluminium Production with Unsupervised Machine Learning Classifiers. In Proceedings of the IEEE International Symposium on INnovations in Intelligent SysTems and Applications, INISTA 2019 - Proceedings; IEEE, 2019; pp. 1–6.
 136. Prognostics and Health Management Society PHM Data Challenge 2015.
 137. Xie, C.; Yang, D.; Huang, Y.; Sun, D. Feature Extraction and Ensemble Decision Tree Classifier in Plant Failure Detection. In Proceedings of the Proceedings of the Annual Conference of the Prognostics and Health Management Society, PHM; 2015; pp. 727–735.