UNIVERSIDAD DE MATANZAS

FACULTAD DE CIENCIAS TÉCNICAS

DEPARTAMENTO DE INFORMÁTICA

<Algoritmos de minería de datos para el procesamiento estadístico de información asociada a la gestión sostenible de playas>

Trabajo de Diploma para optar por el título de Ingeniero Informático

Autor: <Yadian Noda Rodriguez>

Tutor: <Ing. Eduardo Javier Berrio Turiño

Dra. Liz Pérez Martínez>

Matanzas, 2022

Pensamientos

“Quien con el corazón y la sangre se enfrenta a lo imposible, triunfa”

Ernesto Guevara de la Serna

“Debemos sentirnos legítimamente orgullosos de la obra que la Revolución ha hecho en la educación”

Fidel Castro Ruz

“El futuro pertenece a quienes creen en la belleza de sus sueños”

Eleanor Roosevelt

“Quien no tenga iniciativa propia es esclavo del pensamiento ajeno”

Ernesto Guevara de la Serna

Dedicatoria

Dedico este trabajo a mi familia que ha estado pendiente de mis estudios y me ha guiado siempre por el buen camino. En especial lo dedico a mi mamá que es quien me da fuerzas cada día para seguir adelante, a ella debo quien soy, ha estado conmigo en todo momento y los resultados obtenidos son fruto de su dedicación y el esfuerzo conjunto que hemos realizado durante todos estos años.

Agradecimientos

Este trabajo de diploma es fruto del apoyo y dedicación de muchas personas, a las que quiero expresarle mi más sincera gratitud

En primer lugar agradezco a Dios que ha estado conmigo en todo momento y a él debo el éxito recibido.

A mi madre que es la persona más importante en mi vida, que ha estado siempre a mi lado, me ha dado las fuerzas necesarias para continuar en los momentos que me he sentido derrotado y me ha dedicado su tiempo, amor y dedicación. Por esto y muchas cosas más le doy gracias.

A mi padrastro, quien desde chiquito me ha inculcado buenos valores y ha estado siempre pendiente de mis estudios y que no me falte lo necesario para avanzar.

A mi papá que ha estado pendiente de mí y me ha ayudado a cumplir mis metas.

A mi familia que ha sido lo más importante para mí, ha estado presente en cada éxito, en cada fracaso y me han apoyado todo el tiempo.

A mis tutores, el profesor Eduardo y la Profesora Liz que han sido incondicionales y su ayuda ha sido imprescindible todo este tiempo. Considero que sin ellos hubiera sido imposible llegar al final.

A mi novia que ha estado conmigo en los buenos y malos momentos y su apoyo ha sido fundamental en todo el tiempo que llevamos juntos.

A mi hermano que ha sido de gran apoyo durante todos estos años.

A mis compañeros de aula y en especial a esos que estuvieron becados todos estos años conmigo, hablo de Adrian, Luis Raúl y mi gran amigo Yaisel, a quien considero como un hermano y durante estos 5 años ha sido incondicional y su ayuda sido fundamental para llegar al final. También agradezco a Roxana, quien me ha dado su apoyo y en conjunto con Yaisel me ha ayudado a vencer muchos obstáculos.

Al estudiante Leonardo que se ha ganado mi amistad y también ha sido fundamental su ayuda para lograr el resultado final de este proyecto.

A mis jefes del MININT que han estado siempre pendientes de mí, de qué necesito, que problemas tengo y como ayudarme a solucionarlo. Han sido como mi familia todos estos años.

A mis profesores que me han ayudado a formarme como profesional.

A mis vecinos, que son mi segunda familia y han estado al tanto de mi progreso año tras año.

A esos familiares que por desgracia ya no están entre nosotros pero sé que donde quiera que se encuentren están orgullosos de mí. Siempre los tengo presente y viven en mi corazón.

A todos muchas gracias

Declaración de autoría

Yo, Yadian Noda Rodriguez, declaro que soy el único autor del trabajo algoritmo de minería de datos para el procesamiento estadístico de información asociada a la gestión sostenible de playas y autorizo a la Universidad de Matanzas, y en especial, a la Facultad de Ciencias Técnicas, a que hagan el uso que estimen pertinente. Para que así conste firmo la presente a los 28 días del mes de noviembre del 2022

Firma del autor

Yadian Noda Rodriguez

Firma de los tutores

Ing. Eduardo J. Berrio Turiño

Dra. Liz Pérez Martínez

Opinión del tutor

DATOS PERSONALES DEL TUTOR

Nombre y apellidos: Eduardo Javier Berrio Turiño.

Centro de trabajo: Universidad de Matanzas.

Organismo a que pertenece: Ministerio de Educación Superior – MES.

Cargo que ocupa: Profesor e Investigador.

Especialidad de la que es graduado: Ingeniero Informática. Universidad de Matanzas, 2018.

Categoría docente o investigativa: Asistente.

DATOS DE LA TESIS Y EL DIPLOMANTE

Nombre y apellidos: Yadian Noda Rodriguez.

Centro de estudio: Universidad de Matanzas sede “Camilo Cienfuegos”.

Título de la Tesis: Algoritmo de minería de datos para el procesamiento estadístico de información asociada a la gestión sostenible de playas

OPINION SOBRE EL TRABAJO

La tesis presentada posee gran actualidad, pues intenta resolver un problema real presente en toda Cuba y en especial en nuestro territorio matancero, y de gran importancia para el mismo.

El tutor de este trabajo de diploma considera que, durante su ejecución, el estudiante mostró las cualidades que a continuación se detallan:

Independencia y capacidad de investigación. Fueron jornadas adentrándose en temas complejos y nuevos que profundizan en gran medida los recibidos durante la carrera, logró captar con rapidez y profesionalidad el conocimiento necesario para enfrentar el problema planteado. Fue consecuente con los aspectos tanto metodológicos como de la investigación científica propiamente. Esto le permitió una feliz culminación del método desarrollado, de la documentación y de las pruebas realizadas.

En el trabajo se aprecia rigor, manifestado desde el tratamiento de los conceptos estudiados y referenciados en la bibliografía, hasta las conclusiones, lo que ha contribuido a la correcta solución de los problemas encontrados.

Una gran cantidad de clases y métodos, un producto bien concebido y validaciones correctamente realizadas para culminar su investigación, unido a una excelente planificación de tiempo y recursos, dieron una gran calidad al trabajo obtenido.

También fueron horas de revisión, discusión y consenso en las que demostró notables cualidades para la investigación. El trabajo que hoy presenta y que sintetiza un periodo de aprendizaje no solo académico.

Como resultado se derivó en la obtención de un producto de software al nivel de las exigencias y expectativas. Por todo lo anteriormente señalado, considero que el

estudiante Yadian Noda Rodriguez reúne los requisitos para el título de Ingeniero Informático y espero le sea otorgada la mejor calificación de este Tribunal.

Ing. Eduardo J. Berrio Turiño
Dpto. Informática
Universidad de Matanzas
Noviembre/2022

Resumen

Desarrollar modelos predictivos para parámetros ambientales del monitoreo de playas constituye el objetivo principal de esta investigación. Una herramienta informática que apoye la toma de decisiones permitirá una eficiente gestión de estas zonas costeras, de forma tal que se eviten errores que se comenten en la actualidad. Para ello se emplearon tecnologías que demostraron ser competentes para el logro del objetivo de la investigación. La aplicación de técnicas de minería de datos permitió capturar los patrones pasados y replicarlos, además de realizar estimaciones con datos nuevos o fuera de muestra, así como inferir comportamientos y resultados futuros, en aras de anticipar posibles situaciones de deterioro que comprometan la sostenibilidad ambiental. Los experimentos diseñados para comparar los resultados en la clasificación al emplear los modelos predictivos, demuestran que el porcentaje de error oscila entre el 5% y el 10%, lo que demuestra un grado muy bueno (alto), de acuerdo con las escalas de comprobación. Por lo que podemos afirmar que la implementación de herramientas predictivas constituye un paso significativo hacia al objetivo estratégico de convertir información en conocimiento, evidentemente dicho conocimiento será trascendental para un mejor desempeño futuro.

Abstract

Developing predictive models for environmental parameters of beach monitoring is the main objective of this research. A computer tool that supports decision-making will allow efficient management of these coastal areas, in such a way as to avoid errors that are currently being made. For this, technologies that proved to be competent to achieve the objective of the investigation were used. The application of data mining techniques made it possible to capture past patterns and replicate them, in addition to making estimates with new or out-of-sample data, as well as inferring future behaviors and results, in order to anticipate possible deterioration situations that compromise environmental sustainability. The experiments designed to compare the results in the classification when using the predictive models, show that the percentage of error oscillates between 5% and 10%, which demonstrates a very good degree (high), according to the verification scales. Therefore, we can affirm that the implementation of predictive tools constitutes a significant step towards the strategic objective of converting information into knowledge, obviously said knowledge will be transcendental for better future performance.

Índice general

Índice de tablas

Índice de figuras

Introducción

La gestión integral y sostenible del litoral permite manejar de forma integrada las distintas funciones de la playa y los servicios ecosistémicos que ella provee. Para ello, se siguen un conjunto de acciones conducentes al logro de determinados fines en el marco del uso de los recursos de la franja costera. En este sentido, tanto recursos materiales como humanos se combinan, distribuyen y disponen para cumplir dichos objetivos, con la necesidad de una constante evaluación de los efectos para corregir posibles desvíos.

En Cuba se presenta la paradoja de que las playas, a pesar de ser el eje fundamental de la actividad turística y de una enraizada tradición cultural y recreativa de la población local, experimentan en los últimos años un deterioro generalizado de sus potencialidades para tales propósitos. Entre las causas de este deterioro, debido al impacto del cambio climático se encuentran: la erosión costera, el aumento del nivel del mar, el retroceso de sus límites y la contaminación ambiental, fenómenos que se han incrementado en los dos últimos siglos. Las construcciones sobre las dunas litorales, las extracciones de arena y la tala de la vegetación costera son acciones que contribuyen a la destrucción de las playas, debido al crecimiento acelerado de la actividad humana asociada con el desarrollo del turismo, por lo que conocer el comportamiento futuro de las playas, o sea saber hacia dónde se dirigen los parámetros de la calidad de la playa y las posibles consecuencias positivas y negativas, sería una herramienta de gran apoyo e importancia para los gestores de playa donde el actuar a tiempo se hace imprescindible.

A lo anterior se agrega que, en la mayoría de los casos, lo que constituye el detonante de una decisión es el tiempo límite en el que se debe tomar. Lo cual hace que el verdadero objetivo de un sistema de apoyo a las decisiones sea proporcionar la mayor cantidad de información relevante en el menor tiempo posible, con el fin de decidir lo más adecuado.

Con el desarrollo de las tecnologías informáticas ha sido posible la manipulación y el almacenamiento de información en formato electrónico, lo que conlleva a la necesidad de procesarla automáticamente para facilitar la realización de varias tareas como la clasificación y la predicción de la información.

En cualquier caso, la gestión ambiental cubana ha estado caracterizada por cambios bruscos e inesperados en direcciones muchas veces contrapuestas, que nos han llevado, en los últimos años, a replantearnos el empleo de las técnicas normalmente utilizadas para el tratamiento de una realidad que de tan cambiante se ha convertido en incierta.

En el proceso de análisis e interpretación de esta información es de vital importancia la selección de los indicadores que permitan hacer una medición más eficiente del resultado de la gestión en correspondencia a las particularidades que definen los procesos. Debemos saber que los indicadores en si no constituyen un fin, sino que sus resultados son el efecto de diversas causas que se generan en los procesos y que por ello es imprescindible su adecuada interpretación a la hora de tomar decisiones que posibiliten la solución de los problemas.

El objetivo de las técnicas de predicción es obtener estimaciones o pronósticos de valores futuros de una serie temporal a partir de la información histórica contenida en la serie observada hasta el momento actual. Estas técnicas no requieren la especificación de los factores que determinan el comportamiento de la variable, sino que se basan únicamente en la modelización del comportamiento sistemático de la serie. Se consideran tres modelos posibles del comportamiento sistemático de una serie temporal: modelo estacionario (sin tendencia), modelo con tendencia lineal y modelo con estacionalidad. La técnica de predicción adecuada dependerá del modelo de comportamiento de la serie.

En muchas ocasiones ocurren problemas relacionados con la obtención de resultados estadísticamente significativos, que se derivan del uso inapropiado de técnicas estadísticas y econométricas, además de la recurrente ausencia de datos o en otros casos y la duplicidad de los mismos.

Es en este sentido, que los modelos predictivos de minería de datos son apropiados para tratar problemas que tienen evidentes relaciones no lineales a largo plazo y donde se requieren pronósticos con un elevado nivel de fiabilidad formal (bondad del ajuste) y confiabilidad (apropiada selección de variables a relacionar).

La inexistencia de una herramienta informática que asista la toma de decisiones y viabilice la actividad humana en la gestión de playas, constituye un obstáculo en el objetivo estratégico de convertir la información en conocimiento; dicho conocimiento será trascendental para lograr un desarrollo sostenible. Por lo que, para la gestión ambiental cubana, contar con herramientas que apoyan la toma de decisiones es crucial para alcanzar el tan ansiado desarrollo sostenible.

La adopción voluntaria de sistemas de gestión de la calidad y del medio ambiente en las playas supone un cambio sustancial en el enfoque de la ordenación de los usos y la explotación de estos espacios litorales por parte de algunos municipios. Tras examinar las debilidades de la gestión habitual de las playas, se recogen y consideran las distintas normas y sistemas de gestión aplicables. Este nuevo enfoque es necesario para mantener los beneficios económicos y sociales a largo plazo que proporciona el turismo

litoral. La gestión racional en el uso de las playas, apoyada en estos nuevos instrumentos, deberá satisfacer a todas las partes interesadas: los turistas, la población local, el medio ambiente y las generaciones futuras.

Lo anteriormente descrito deriva en el siguiente **problema científico** de investigación: ¿Cómo desarrollar un modelo de minería de datos que permita predecir comportamientos futuros y apoyar el proceso de toma de decisiones en el monitoreo de playas?

Para dar solución al problema antes expuesto se traza como **objetivo general**: Desarrollar modelos predictivos para indicadores ambientales que apoyen la toma de decisiones para una eficiente gestión ambiental empresarial.

Los **objetivos específicos** definidos para dar cumplimiento al objetivo general son:

1. Analizar los trabajos más importantes relacionados con la utilización de modelos predictivos.
2. Diseñar las propuestas de modelos predictivos.
3. Diseñar un interfaz web para el empleo del modelo predictivo
4. Implementar las propuestas de modelos predictivos.
5. Implementar la interfaz web
6. Validar las propuestas mediante la realización de experimentos.
7. Validar el módulo web mediante pruebas de software

Los métodos y técnicas utilizados para el desarrollo de la investigación son los siguientes:

- Métodos teóricos:
  - Método de análisis histórico – lógico: permitió estudiar la trayectoria y desarrollo de los modelos predictivos existentes, así como el proceso de análisis de indicadores ambientales.
  - Método de **análisis y síntesis**: este se precisó durante la revisión bibliográfica y el análisis de los resultados, lo que permitió descomponer lo complejo en sus partes y cualidades, la división del todo en sus múltiples relaciones para luego unir las partes analizadas, descubrir las relaciones y características generales entre ellas.

- Método **inductivo - deductivo**: su uso fue necesario tanto en la revisión bibliográfica, como en el análisis de los resultados, de modo que permitió arribar a conclusiones que se infirieron a partir de propiedades y relaciones existentes entre los elementos que conforman el fenómeno objeto de estudio.

- Los métodos empíricos, utilizados mediante las siguientes técnicas:
  - **Observación**: permitió entender el proceso de análisis de los indicadores ambientales y se obtuvo la información primaria acerca de los objetos investigados.
  - **Entrevistas**: aportaron datos esenciales a la investigación puesto que el entrevistado es la persona que propuso el desarrollo de la investigación en primer lugar.

Principales **aportes** de la investigación:

- El **teórico-investigativo**, al integrar los procedimientos tradicionales más utilizados por autores relacionados con los indicadores ambientales a través de diferentes pasos que permiten orientar metodológicamente la secuencia de acciones lógicas a desarrollar para la conformación de los modelos; y los elementos a tener en cuenta para la continuidad de la investigación.
- El **práctico**, al desarrollar una herramienta informática que asista la toma de decisiones y viabilice la actividad humana en el logro de los objetivos de desarrollo sostenible.
- El **económico**, al elaborar modelos capaces de predecir cómo se comportarán los indicadores ambientales, de modo que se posibilite tomar decisiones que maximicen la gestión ambiental del país.

El **resultado esperado** con este trabajo es lograr que el sistema analice los indicadores ambientales y generalice su comportamiento, lo que permite realizar pronósticos y que constituyan un instrumento para la toma de decisiones.

A partir de lo planteado anteriormente, la investigación queda estructurada en: introducción, tres capítulos, conclusiones, recomendaciones y referencias bibliográficas, según sigue:

- Una Introducción, donde se caracteriza la situación problemática y se fundamenta el problema científico a resolver.
- Un primer capítulo donde se recoge el marco teórico referencial del tema y los principales conceptos que constituyen la base teórica de la investigación, así como

el análisis de las principales tendencias tecnológicas y el estudio de los antecedentes que enmarcan la problemática planteada.

- Un capítulo segundo donde se diseña la propuesta de solución a partir de la conformación de los modelos predictivos, sobre la base de lo analizado en el capítulo primero.
- Un tercer capítulo donde se analizan los resultados obtenidos para el pronóstico de indicadores a partir de los pronósticos obtenidos.
- Un apartado de conclusiones donde se verifica el cumplimiento de los objetivos trazados al inicio de la investigación.
- Las recomendaciones en la cual se plasman una serie de propuestas encaminadas a la continuidad de esta investigación.
- Y las referencias de la bibliografía citada.

Capítulo I "Marco Teórico-Referencial"

1.1 Introdución del Capítulo

Las técnicas de pronósticos tienen como objetivo general obtener un valor futuro bajo la incertidumbre, donde se toma como base la información histórica y se trata de encontrar un patrón de comportamiento que permita conocer cuál va a ser el compartimiento en el futuro. Los métodos de pronósticos se dividen en Cualitativos y Cuantitativos. Los métodos cualitativos tienen como característica principal, depender del conocimiento y experiencia de los expertos, dentro de estos métodos encontramos las siguientes técnicas:

- **Ajuste Curva subjetiva**: Es creada por experto y se basa en el conocimiento del negocio, aplicada en las predicciones de nuevos productos, donde se analiza, para productos con características similares, cuál va a ser el comportamiento desde el lanzamiento de este, hasta la estabilización de las ventas, la complejidad está en cómo definir la forma que tendrá la curva.
- **Delphi**: Utiliza el juicio del experto. Existen varios expertos que establecen el pronóstico. No hay datos, todo se hace por intuición. Por otra parte, las técnicas cuantitativas de predicción consisten en encontrar un patrón en los datos disponibles para poder proyectarlo al futuro, es decir, estos métodos requieren de un análisis de la información para poder efectuar la predicción de la variable que sea de interés.

1.2 Antecedentes

La minería de datos es un proceso ampliamente utilizado en la investigación de diferentes bigdatas. Su uso va en ascenso exponencial dado que con el pasar de los años se acumulan cada vez más datos. Diversos trabajos similares a esta investigación se han realizado en diferentes campos de aplicación.

- Modelos ARIMA univariante de series temporales para la producción y demanda de agua en el distrito de Lambayeque, periodo 2002 – 2017. El presente estudio tuvo como objetivo principal determinar el modelo univariante que permita predecir el comportamiento de la producción y demanda de agua en el distrito de Lambayeque, periodo 2002 al 2017. El análisis de los datos de la serie, se realizó mediante la metodología de Box – Jenkins, para identificar el modelo que mejor se adecue a los datos observados. En función a ello se determinó: El mejor modelo que explica el comportamiento de la producción de agua en el periodo indicado es el modelo SARMA (1,0,2) (1,0,0), con un Error Cuadrático Medio = 16 932.67, con Error Absoluto Medio = 13 254.61, el Error Porcentual Absoluto Medio = 3.98% y con coeficientes estimados AR (1) = 0.608, MA (1) = 0.299, MA (2) = -0.248, SAR (1) =

0.215. Mientras que el modelo ARIMA (0 ,1 ,1), con un Error Cuadrático Medio = 3 062.74, con Desviación Absoluta de la media = 2 303.09, el Porcentaje de Error Medio Cuadrado Absoluto = 1.44% y con coeficientes estimados MA (1) = 0.377 es el que explica mejor el comportamiento de la demanda de agua en el periodo analizado. (López Jiménez & Villanueva Vásquez, 2020)

- Desarrollo e implementación de modelos de Machine Learning para aplicaciones de gestión y eficiencia energética. El propósito del proyecto es el desarrollo de un sistema de Aprendizaje Automático que permita realizar predicciones de consumo de suministros eléctricos. También debe permitir la detección de errores de datos de series temporales de consumo para su posterior análisis y corrección. Para ello se toma como datos de entrenamiento el histórico de series temporales de cada uno de los suministros eléctricos proporcionados por las comercializadoras. (Ruiz Brückel, 2020)

- Análisis de datos en entornos inteligentes basados en el Internet de las cosas. La investigación e innovación contribuyen de manera decisiva a la lucha contra el cambio climático. Las TIC pueden reducir un 20 % de las emisiones mundiales de CO2 de aquí a 2030. El Aprendizaje Automático es útil para detectar ineficiencias de las ciudades modernas que contribuyen a la inestabilidad climática. Análisis iniciales sugieren que la conversión a edificios inteligentes gracias a la sensorización del Internet de las Cosas (IdC), junto con el análisis de datos, podría ser una opción para abordar estos problemas. Para ello, hemos identificado las siguientes necesidades de los edificios: mejorar las decisiones de gasto, reducir el consumo de energía, mejorar la eficiencia operativa y capacitar a sus usuarios con conocimientos energéticos; y las siguientes necesidades relativas a los análisis: cumplir los requisitos Big Data (volumen, velocidad, variedad, etc.), proporcionar mecanismos de fusión de datos, identificación de patrones de movilidad humana, reducción de información redundante en tiempo real, mejora de la predicción de series temporales mediante la selección de características y la gobernanza de datos. (González Vidal, 2020)

- Análisis sobre el uso de la red social Facebook en el proceso de enseñanza-aprendizaje por medio de la ciencia de datos. Hoy día, los avances tecnológicos han provocado el desarrollo de nuevas estrategias de enseñanza-aprendizaje. De hecho, las redes sociales han adquirido gran relevancia durante la planeación de las actividades escolares. En particular, esta investigación mixta analiza el uso de Facebook como medio de difusión, comunicación, aprendizaje, interacción y

colaboración durante la realización de las prácticas de laboratorio en la asignatura “Desarrollo de aplicaciones para los negocios”. La minería de datos permite establecer los modelos predictivos sobre el impacto de Facebook durante el diseño de la interfaz web, para ello se consideran las técnicas bayesiana y árbol de decisión (ciencia de datos). La muestra está compuesta por 69 estudiantes de la Licenciatura en Gestión de Negocios y Tecnologías de Información. Por medio del enfoque cuantitativo y cualitativo, este estudio analiza el empleo de esta red social en el proceso de enseñanza-aprendizaje relacionado con el diseño de la interfaz web, las instrucciones HTML, el lenguaje de programación PHP, la aplicación WampServer y la base de datos MYSQL. Asimismo, el método ANOVA evalúa el rendimiento académico de los grupos experimental y control por medio de la calificación en el proyecto práctico. Los resultados obtenidos permiten afirmar que Facebook representa una alternativa tecnológica para mejorar la organización e implementación de las experiencias educativas en el siglo XXI. (Salas Rueda & Salas Rueda, 2019)

- Impacto del Diplomado Dirección y Gestión Empresarial en la Capacitación de Cuadros y Ceservas, en la filial “Rubén Martínez Villena”, Universidad de Artemisa. La investigación se desarrolló en la filial Rubén Martínez Villena, ubicada en el municipio Alquízar, perteneciente a la Universidad de Artemisa, provincia del mismo nombre, con el objetivo de evaluar el impacto del Diplomado en Dirección y Gestión Empresarial en la Capacitación de los Cuadros y Reservas. El estudio se realizó con un enfoque dialéctico-materialista, el empleo de métodos teóricos y empíricos entre ellos: histórico y lógico, análisis y síntesis, sistémico, revisión documental y la observación, así como el empleo de la minería de datos para el análisis, evaluación y representación de los resultados. Los que revelan la participación de los directivos de las entidades de los grupos empresariales de la provincia, las calificaciones obtenidas en los diferentes módulos impartidos en el diplomado, donde predominan las calificaciones de excelente. Finalmente se evidencian algunas de las innovaciones propuestas por los diplomantes que se encuentran implementadas o en fase de implementación. (Pérez Almeneiro & Sánchez Batista, 2019)

- Aprendizaje profundo para la extracción de aspectos en opiniones textuales. La extracción de aspectos en opiniones textuales es una tarea muy importante dentro del análisis de sentimientos o minería de opiniones, que permite lograr mayor exactitud al analizar la información y contribuir a la toma de decisiones. El aprendizaje profundo agrupa varios algoritmos o estrategias que han obtenido resultados relevantes en diversas tareas del procesamiento del lenguaje natural. Existen varios

artículos de revisión sobre el análisis de sentimientos que abordan el aprendizaje profundo como una de las técnicas existentes para la extracción de aspectos; sin embargo, no existen artículos de revisión que se dediquen exclusivamente al empleo del aprendizaje profundo en el análisis de sentimiento. El objetivo de este artículo consiste en ofrecer un análisis crítico y comparativo de las principales propuestas y trabajos de revisión que emplean estrategias de aprendizaje profundo para la extracción de aspectos, para ello se profundiza en la forma de representación, modelos, resultados y conjuntos de datos empleados en esta tarea. En esta propuesta se hace el análisis de 89 artículos publicados durante el período 2011 a 2019 de modo que resaltan sus principales aciertos, fisuras, y retos de investigación. Finalmente, proponemos algunas direcciones de investigación futuras. (López Ramos & Arco García, 2019)

- Métodos de series temporales en los estudios epidemiológicos sobre contaminación atmosférica: En este trabajo realizado por un colectivo al mando de **Marc Saez** se relacionan las variaciones en el número diario de muertos mayores de 70 años (todas las causas, CIE-9:001-799) en Barcelona, 1991-1995, con las variaciones en los niveles diarios promedio de contaminación por humos negros. Este aplica las series temporales para intentar construir un modelo explicativo de la evolución temporal de la variable dependiente, con el fin de cuantificar los efectos de factores de riesgo. (Saez, et. al., 1999)

- Técnicas de predicción económicas. Un documento realizado por María Pilar González Casimiro, que investiga la importancia de las series temporales en la predicción de datos económicos. Plantea la importancia de estos estudios para la toma de decisiones, para lo que se argumenta la fiabilidad que brindan los modelos vistos. (Casimiro, 2009)

- Uso de los modelos de series temporales como técnica de análisis de los diseños conductuales. En el anuario de psicología de 1981(2), se realizó un importante estudio para el desarrollo conductual. El mismo se centró en el uso de series temporales y su aplicación en dicho campo, que permitió llegar a la conclusión y cito al autor “(…) los modelos de series temporales resuelven satisfactoriamente uno de los problemas más importantes implícitos en los diseños conductuales: La dependencia serial. Los análisis de series temporales, tienen en cuenta el grado de dependencia existente entre las observaciones y permiten obtener inferencias válidas sin que por ello el investigador tenga que violar supuestos básicos del modelo estadístico o introducir variaciones a fin de soslayar dicho problema.” (Arnau, 1981)

- Análisis de series de tiempo en el pronóstico de almacenamiento de productos perecederos. Los autores realizaron un estudio sobre la relevancia de incorporar pronósticos en la demanda de almacenamiento en productos perecederos dentro de la cadena de frío deriva de su importancia económica y social. Este caso de estudio presenta una empresa con tendencia de crecimiento dedicada al almacenamiento de productos perecederos e incorpora técnicas de pronósticos de series de tiempo, en el volumen de ingreso y egreso de los productos en una cámara frigorífica, con el fin de estimar el volumen de almacenamiento para prever los requerimientos de instalaciones adicionales, personal y materiales necesarios para la movilidad de los productos. (Juárez, 2016)

1.3 Mineria de Datos

La minería de datos es un campo de la estadística y las ciencias de la computación referido al proceso de detectar la información procesable de los conjuntos grandes de datos. El término es un concepto de moda, y es frecuentemente mal utilizado para referirse a cualquier forma de datos a gran escala o procesamiento de la información. En el uso de la palabra, el término clave es el descubrimiento, comúnmente se define como "la detección de algo nuevo", para esto utiliza el análisis matemático para deducir los patrones y tendencias que existen. Normalmente, estos patrones no se pueden detectar mediante la exploración tradicional porque las relaciones son demasiado complejas o porque hay demasiados datos. El objetivo general del proceso de minería de datos consiste en extraer información de un conjunto de datos y transformarla en conocimiento. Para entender su significado es imprescindible conocer los términos relacionados en esta:

- Datos: son cualquier hecho, número o texto que puede ser procesado por una computadora. Hoy día, las organizaciones acumulan grandes cantidades, y cada vez mayores, en diferentes formatos y diferentes bases de datos
- Información: los patrones, asociaciones, o relaciones entre todos estos datos pueden proporcionar información. Por ejemplo, el análisis del punto de venta de datos de transacciones puede dar información sobre qué productos se venden y cuándo.
- Conocimiento: la información puede ser convertida en conocimiento acerca de los patrones históricos y las tendencias futuras. Por ejemplo, la información resumida sobre las ventas de supermercados minoristas puede ser analizada a la luz de los esfuerzos de promoción para facilitar el conocimiento del comportamiento de compra

del consumidor. Por lo tanto, un fabricante o distribuidor puede determinar qué elementos son los más susceptibles a los esfuerzos de promoción.

1.3.1 Métodos de series temporales

Los métodos de series temporales utilizan datos históricos como base para estimar resultados futuros, se asume que la variable a predecir es función de las observaciones de estas variables en periodos de tiempos anteriores. En este tipo de análisis pueden estar involucrados los siguientes componentes:

- **Tendencia**: Es el componente de largo plazo que representa el crecimiento o declinación de la serie. En términos intuitivos, la tendencia caracteriza el patrón gradual y consistente de las variaciones de la serie, que es consecuencia de "fuerzas persistentes" que afectan el crecimiento o la reducción de la serie.
  También se conoce como **Tendencia secular** y suele obtenerse o describirse mediante ajuste a una función matemática o por medias móviles o alisamiento exponencial. (CEACES, Series Temporales, 2022)
- **Ciclo**: Es la fluctuación en forma de onda alrededor de la tendencia. Una de las fluctuaciones cíclicas más comunes en series de tiempo son las llamadas ciclo económico, la cual está representada por fluctuaciones ocasionadas por períodos recurrentes de prosperidad de modo que se alterne la recesión, sin embargo, dichas fluctuaciones no necesitan ser causadas por cambios en los 20 factores económicos, como, por ejemplo; en la producción agrícola donde la fluctuación cíclica puede estar ocasionada por los cambios climáticos.
  También se le llama variaciones cíclicas y se consideran oscilaciones periódicas que se producen con una frecuencia superior a un año, suelen deberse a la alternancia de etapas de prosperidad económica (crestas) con etapas de depresión. (CEACES, Series Temporales, 2022)
- **Variación Estacional**: Son patrones periódicos que se repiten año tras año, factores como el clima y las costumbres ocasionan estos tipos de patrones.
  Tienen gran relación con el comportamiento de los agentes económicos al variar la época del año. (CEACES, Series Temprales, 2022)
- **Fluctuaciones irregulares o irregularidad:** Son movimientos erráticos que siguen un patrón indefinido o irregular. Estos movimientos representan lo que queda de la serie después de haber restado las demás componentes.
  Este método también se conoce como variación errática y recoge la variabilidad en el comportamiento de la serie que se debe a pequeñas causas impredecibles. (CEACES, Series Temporales, 2022)

Muchas de las fluctuaciones irregulares son causadas por hechos inusuales que no se pueden predecir, como son los sismos, huracanes, guerras, entre otros.
Algunas técnicas de series de tiempo encontradas son:

1. Método Ingenuo: este método asume que la variable a predecir tiene igual valor a su medición anterior.
2. Promedio Móvil: la predicción es el resultado del promedio de n observaciones anteriores.
3. Métodos de Descomposición. Como su nombre lo indica, descomponen la serie de tiempo en sus cuatro componentes con el objetivo de describir cada una de ellas por separado para lograr así describir y predecir en conjunto la serie de tiempo.
4. Suavizamiento Exponencial. El objetivo es filtrar o suavizar la serie para tener una mejor idea del comportamiento de la tendencia y por tanto tener un pronóstico más confiable. Dentro de los métodos de suavizamiento encontramos los métodos de Holt Winters.
5. Metodología de Box-Jenkins. Proporciona una colección más extensa de modelos de Predicción además de ser un procedimiento más sistemático para ayudar a identificar el modelo adecuado

1.3.2 Métodos por análisis de regresiones

El análisis de regresión es una técnica estadística para estudiar la relación entre variables. El término regresión fue introducido por Francis Galton en 1886. Su trabajo se centró en la descripción de los rasgos físicos de los descendientes (variable A) a partir de los de sus padres (variable B). Con el estudio de la altura de padres e hijos a partir de más de mil registros de grupos familiares, se llegó a la conclusión de que los padres muy altos tenían una tendencia a tener hijos que heredaban parte de esta altura, pero que revelaban también una tendencia a regresar a la media. Galton generalizó esta tendencia bajo la "ley de la regresión universal": «Cada peculiaridad en un hombre es compartida por sus descendientes, pero en media, en un grado menor». El objetivo de la regresión es descubrir la relación funcional entre la entrada y la salida de este sistema, para poder así predecir la salida de este cuando se le presenta un dato de entrada nuevo. En un análisis de regresión simple existe una variable respuesta o dependiente (y) y una variable explicativa o independiente (x). El propósito es obtener una función sencilla de la variable explicativa, que sea capaz de describir lo más ajustadamente posible la variación de la variable dependiente. La variable explicativa puede estar formada por un vector de una sola característica o puede ser un conjunto de n

características, atributos o dimensiones (regresión múltiple). La regresión se utiliza para predecir una medida basándonos en el conocimiento de otra y la intención final es que dado un vector de entrada se persigue predecir un valor de salida a partir de una función generada mediante la supervisión previamente observada de un conjunto de entrenamiento inicial.

1.3.2.1 Regresión lineal

Este método consiste en construir a partir de los datos de entrada una función lineal (línea recta) que atraviese los puntos con la mínima variación entre cada observación y el punto predicho por la función.

Como los valores observados de la variable dependiente difieren generalmente de los que predice la función, esta posee un error. La función más eficaz es aquella que describe la variable dependiente con el menor error posible o, dicho en otras palabras, con la menor diferencia entre los valores observados y predichos. La diferencia entre los valores observados y predichos (el error de la función) se denomina variación residual o residuos. Para estimar los parámetros de la función se utiliza el ajuste por mínimos cuadrados. Es decir, se trata de encontrar la función en la cual la suma de los cuadrados de las diferencias entre los valores observados y esperados sea menor. Sin embargo, con este tipo de estrategia es necesario que los residuos o errores estén distribuidos normalmente y que varíen de modo similar a lo largo de todo el rango de valores de la variable dependiente. Estas suposiciones pueden comprobarse a partir del examen de la distribución de los residuos y su relación con la variable dependiente.

1.3.2.2 Modelos de regresión lineal en Python

Dos de las implementaciones de modelos de regresión lineal más utilizadas en Python son: Scikit-Learn y Statsmodels. Aunque ambas están muy optimizadas, Scikit-Learn está orientada principalmente a la predicción, por lo que no dispone de apenas funcionalidades que muestren las muchas características del modelo que se deben analizar para hacer inferencia. Statsmodels es mucho más completo en este sentido.

- **Método de los mínimos cuadrados:** El método de mínimos cuadrados es adecuado para procesar un conjunto de datos. No tiene que pasar exactamente por cada punto, sino en función de la distancia de la imagen a cada punto de datos y la función mínima.

  Se considera un procedimiento de análisis numérico en la que, dados un conjunto de datos (pares ordenados y familia de funciones), se intenta determinar la función continua que mejor se aproxime a los datos (línea de

regresión o la línea de mejor ajuste), de modo que se proporcione una demostración visual de la relación entre los puntos de los mismos. En su forma más simple, busca minimizar la suma de cuadrados de las diferencias ordenadas (llamadas residuos) entre los puntos generados por la función y los correspondientes datos.

- **Método de Gradiente descendiente:** El gradiente descendiente es la base de aprendizaje en muchas técnicas de machine learning. Por ejemplo, es fundamental en deep learning para entrenar redes neuronales. También es necesario para la regresión logística. Y en muchos casos, al hacer regresión lineal o polinómica es mejor usar el método del gradiente descendiente que el de los mínimos cuadrados.

1.3.3 Regresión no lineal

Este método consiste en conseguir una función polinomial de mayor grado que permita minimizar el error.

1.3.3.1 Modelos de regresión no lineal en Python:

- **Método de validación cruzada:** La validación cruzada o cross-validation es una técnica utilizada para evaluar los resultados de un análisis estadístico y garantizar que son independientes de la partición entre datos de entrenamiento y prueba. Consiste en repetir y calcular la media aritmética obtenida de las medidas de evaluación sobre diferentes particiones. Se utiliza en entornos donde el objetivo principal es la predicción y se quiere estimar la precisión de un modelo que se llevará a cabo a la práctica. Es una técnica muy utilizada en proyectos de inteligencia artificial para validar modelos generados.

- **Método de retención:** El método de retención o holdout method consiste en dividir en dos conjuntos complementarios los datos de muestra, realizar el análisis de un subconjunto (denominado datos de entrenamiento o training set), y validar el análisis en el otro subconjunto (denominado datos de prueba o test set), de forma que la función de aproximación sólo se ajusta con el conjunto de datos de entrenamiento y a partir de aquí calcula los valores de salida para el conjunto de datos de prueba (valores que no ha analizado antes). La ventaja de este método es que es muy rápido a la hora de computar. Sin embargo, este método no es demasiado preciso debido a la variación de resultados obtenidos

para diferentes datos de entrenamiento. La evaluación puede depender en gran medida de cómo es la división entre datos de entrenamiento y de prueba, y por lo tanto puede ser significativamente diferente en función de cómo se realice esta división.

1.4 Métodos Machine Learning

1.4.1 Redes Neuronales

Las Redes de Neuronas Artificiales (M, 1993) (Artificial Neuronal Networks, ANN) son modelos de aprendizaje inspirados en el sistema nervioso de los animales. Se componen de una serie de unidades, denominadas neuronas, que de forma simplificada simulan la funcionalidad de las neuronas biológicas.

Estos modelos son extremadamente flexibles, pudiéndose obtener un número virtualmente infinito de modelos de redes distintos al variar el tipo de neuronas en cada capa y la forma de interconectarlas en red, así como el mecanismo de aprendizaje.

De lo mejor de las redes neuronales es la capacidad de aprender de la experiencia, a partir de conocimiento almacenados en los pesos asociados a las conexiones entre neuronas, el poder de utilizar información con alto nivel de ruido y/o incompleta para ser procesada con resultados satisfactorios, la capacidad de reconocer y organizar datos que no habían sido vistos con anterioridad. Todo esto hace que las redes neuronales se han convertido en una gran herramienta para el reconocimiento de patrones, clasificación y agrupación de datos.

El entrenamiento de las redes neuronales puede ser clasificado en tres tipos: Entrenamiento supervisados, sin supervisión y entrenamiento por refuerzo. Los algoritmos de entrenamiento supervisados requieren el uso de ejemplos del que debería ser el comportamiento adecuado de la red en presencia de entradas específicas.

1.4.1.1 Perceptrón Multicapa (MLP)

Un Perceptrón Multicapa (Multilayer Perceptron, MLP) es un modelo de red neuronal artificial formada por múltiples capas de neuronas. Este es uno de los modelos más usados de redes neuronales artificiales.

La arquitectura de un Perceptrón multicapa consta de varias capas de neuronas, en las que las salidas de una capa son las entradas de la siguiente. Las conexiones entre neuronas tienen un valor de ponderación o peso y cada neurona tiene una función de activación con la que genera su salida en función de las entradas. Las capas pueden clasificarse en tres tipos:

- Capa de entrada: Constituida por aquellas neuronas que representan cada entrada a la red, estas neuronas no producen procesamiento.
- Capas Ocultas: Formada por aquellas neuronas cuyas entradas provienen de capas anteriores y las salidas pasan a neuronas de capa posteriores.
- Capa Salida: Neuronas cuyos valores de salida corresponden con las salidas de toda la red.

Estas redes son populares porque son capaces de aproximar cualquier función continua en un intervalo hasta el nivel deseado. (Funahashi, 1989). Usualmente para pronósticos o modelos de regresión se utilizan una capa de entrada con tantas neuronas como variables independientes del modelo, una capa oculta con función de activación Sigmoide donde los nodos se establecen experimentalmente, y una capa de salida con una neurona y función de activación lineal.

1.5 Herramientas y tecnologías

1.5.1 Python

Python es un lenguaje de programación que te permite trabajar rápidamente para integrar los sistemas de manera más eficaz. (Foundation, 2022), Dentro de sus principales características destacan (Jaume, 2010):

- Es fácil de utilizar.
- Es un lenguaje “completo”; no sirve solo para programar scripts.
- Tiene gran variedad de estructuras de datos incorporadas al propio lenguaje.
- Tiene una gran cantidad de bibliotecas (libraries).
- Permite la programación modular, orientada a objetos y su uso como un lenguaje imperativo tradicional.
- Es interpretado.
- Esto facilita el desarrollo (aunque ralentice la ejecución).
- Se puede utilizar desde un entorno interactivo. Se puede extender fácilmente

1.5.1.1 Librerías de Python

1) **Series temporales:**

- **Keras:** es una librería de alto nivel utilizada para prototipado rápido, investigación de vanguardia y soluciones productivizadas. Entre las características de Keras destacan su interfaz simple y optimizada para casos

comunes, la modularidad mediante el uso de bloques y la facilidad de adaptar estos bloques para aplicar nuevos descubrimientos estado del arte.

Keras es una API diseñada para seres humanos, no para máquinas. Keras sigue las mejores prácticas para reducir la carga cognitiva: ofrece API consistentes y simples, minimiza la cantidad de acciones del usuario requeridas para casos de uso comunes y proporciona mensajes de error claros y accionables. También tiene una extensa documentación y guías para desarrolladores. (Google, 2022)

- **TensorFlow:** es una de las librerías open source más importantes de Deep Learning y ha sido creada por Google. Está formada por un ecosistema flexible de herramientas, librerías y recursos de la comunidad y ayuda a los investigadores a innovar mediante la aplicación de técnicas de Machine Learning. Además, permite la compilación y entrenamiento de modelos de Machine Learning de una forma sencilla con la utilización de sus API's. (SOLVER, 2022)

- **Scikit-Learn:** esta biblioteca contiene varias herramientas eficientes para aprendizaje automático y modelado estadístico, la misma incluye clasificaciones, regresión, y reducción de dimensionalidad.

  La gran variedad de algoritmos y utilidades de Scikit-learn la convierten en la herramienta básica para empezar a programar y estructurar los sistemas de análisis de datos y modelado estadístico. Los algoritmos de Scikit-Learn se combinan y depuran con otras estructuras de datos y aplicaciones externas como Pandas o PyBrain. (BSD, 2022)

2) Valores numéricos y estructura de datos

- **Numpy**: se caracteriza por ser la librería de procesamiento de arrays por excelencia. Contiene una gran colección de funciones que permiten realizar operaciones lógicas, redimensiones, búsquedas y aplicar estadísticas entre otras muchas. El núcleo de la librería se basa en los objetos ndarray, los cuales permiten encapsular arrays de n dimensiones sobre los que se pueden realizar las operaciones antes descritas de una forma muy eficiente. (GitHub, 2022)

- **Pandas:** es una herramienta de manipulación y análisis de datos de código abierto rápida, potente, flexible y fácil de usar, construida sobre el lenguaje de programación Python. (NumFOCUS, 2022)

Una de las principales virtudes que tiene esta librería es la carga desde distintos orígenes. Entre los orígenes que acepta encontramos archivos de texto plano como CSV, ficheros en el extendido formato Excel y cargas directas desde bases de datos SQL, entre otros orígenes de datos. Todas estas funciones de datos contienen la información en formato tabular y pandas permite representar este tipo de datos a la perfección mediante el uso de su estructura principal, el DataFrame.

3) Gráficos y visualización

- **Matplotlib:** Es una librería de python especializada en la creación de gráficos en dos dimensiones. Permite crear y personalizar los tipos de gráficos más comunes, entre ellos: diagramas de barras, histogramas, diagramas de sectores, diagramas de caja y bigotes, diagramas de violín, diagramas de dispersión o puntos, diagramas de líneas, diagramas de áreas, diagramas de contorno, mapas de color y combinaciones de todos ellos.

  Matplotlib es una biblioteca completa para crear visualizaciones estáticas, animadas e interactivas en Python. Matplotlib hace que las cosas fáciles sean fáciles y las difíciles sean posibles. (Hunter, 2022)
- **Seaborn**: Seaborn es una biblioteca de visualización de datos de Python basada en matplotlib. Proporciona una interfaz de alto nivel para dibujar gráficos estadísticos atractivos e informativos. (Waskom, 2022), Está construido en la parte superior de la biblioteca matplotlib y también está estrechamente integrado a las estructuras de datos de pandas. Seaborn tiene como objetivo hacer de la visualización la parte central de la exploración y comprensión de los datos. Proporciona API orientadas al conjunto de datos, de modo que podamos cambiar entre diferentes representaciones visuales para las mismas variables para una mejor comprensión del conjunto de datos.

1.5.2 Visual Studio Code

Editor de código fuente independiente que se ejecuta en Windows, macOS y Linux. La elección principal para desarrolladores web y JavaScript, con extensiones para admitir casi cualquier lenguaje de programación. (Build, 2022)

1.5.3 PostgresSql

PostgreSQL es un potente sistema de base de datos relacional de objetos de código abierto con más de 30 años de desarrollo activo que le ha valido una sólida reputación de fiabilidad, solidez de características y rendimiento. Un sistema de base de datos relacionales es un sistema que permite la manipulación de acuerdo con las reglas del álgebra relacional. Los datos se almacenan en tablas de columnas y renglones. Con el uso de llaves, esas tablas se pueden relacionar unas con otras. Es gratuito y libre, además de que nos ofrece una gran cantidad de opciones avanzadas. De hecho, es considerado el motor de base de datos más avanzado en la actualidad. Una característica interesante de PostgreSQL es el control de concurrencias multiversión o MVCC por sus siglas en inglés. Este método agrega una imagen del estado de la base de datos a cada transacción. Esto nos permite hacer transacciones eventualmente consistentes, ofreciéndonos grandes ventajas en el rendimiento. No se requiere usar bloqueos de lectura al realizar una transacción lo que nos brinda una mayor escalabilidad. También PostgreSQL tiene Hot-Standby. Este permite que los clientes hagan búsquedas (sólo de lectura) en los servidores mientras están en modo de recuperación o espera. Así podemos hacer tareas de mantenimiento o recuperación sin bloquear completamente el sistema (Platzi, s.f.). PostgreSQL aporta mucha flexibilidad a nuestros proyectos, por ejemplo, nos permite la utilización de varios lenguajes como R, Python y Ruby. Permitiéndonos la utilización del primer caso para el desarrollo de este proyecto. (Group, 2022)

1.6 Métodos de validación:

En la literatura encontró que en machine learning el uso masivo de las técnicas de validación cruzadas (cross validation) como las más utilizadas por los investigadores para entrenamiento y validación del modelo.

Existen varios tipos de cross validation, la más utilizada es la de k-iteraciones o K-Folds, la cual consiste en dividir el conjunto de datos en k subconjunto. La técnica consiste en dejar uno de los subconjuntos como datos de validación y el resto (k–1) como datos de entrenamiento. El proceso de validación cruzada es repetido durante k iteraciones, con cada uno de los posibles subconjuntos de datos de validación. Finalmente, se selecciona la que mayor capacidad de generalización posea. (Arlot, 2010).

Cross validation asume que la data es independiente y distribuida idénticamente. Ambas asunciones podrían no mantenerse para series de tiempo, como los datos están usualmente autocorrelacionados.

1.7 Evaluación de la exactitud del pronóstico

El objetivo de las medidas de error, es obtener un claro y robusto resumen de la distribución del error. Es una práctica común calcular la medida de error a partir de la función de perdida y la generación del promedio. Dado ***yt*** ser el valor observado ***ŷt*** el valor predicho, el error es calculado por ***e = yt - ŷt***.

1.8 Conclusiones del capítulo

Después de haber realizado el estudio de la teoría y los conceptos en torno a la minería de datos y a los modelos predictivos específicamente; así como luego de analizar las principales tendencias tecnológicas en este sentido, podemos concluir que:

1. El inapropiado uso de técnicas estadísticas y econométricas, además de la recurrente ausencia de datos o en otros casos, la duplicidad de los mismos conlleva a problemas relacionados con la obtención de resultados estadísticamente significativos a nivel formal (baja confiabilidad).
2. Los modelos predictivos son apropiados para tratar problemas con evidentes relaciones no lineales a largo plazo y donde se requieren pronósticos con un el evado nivel de fiabilidad formal (bondad del ajuste) y confiabilidad (apropiada selección de variables a relacionar).
3. Se definieron las tecnologías que mejor se ajustan a los requerimientos del problema detectado

Capítulo II "Solución teórica del problema científico"

2.1 Introducción del Capítulo

El desarrollo de modelos predictivos de minería de datos trae consigo una ardua labor de análisis y diseño, de modo que se alcance una propuesta de solución lo suficientemente genérica para modelar y predecir cualquier entrada de datos que se realice. En este capítulo se describirá la solución propuesta, a partir del análisis de los requerimientos del software.

2.2 Carga de datos en Python

Para crear un modelo de aprendizaje automático es necesario cargar los datos a partir de los cuales construiremos el mismo. Existen distintas formas de realizar esta tarea, las cuales dependen del formato que tengan los datos, su ubicación o los recursos queramos utilizar. (Telefónica Tech, 2022)

Los datos que se utilizan en esta investigación son extraídos de la base de datos del sistema de gestión de playas, desarrollado por el Observatorio Ambiental Costa Atenas de la Universidad de Matanzas, dicha base de datos se desarrolló en PostgreSQL.

La base de datos posee los datos de las mediciones realizadas a los parámetros del monitoreo ambiental que se realiza en las playas.

Para cargar los datos desde Python fue necesaria la librería pyscopg2, este permite la conexión de dicha base de datos con el marco de trabajo Django. Posteriormente se construyó la función leer(parámetro), que se le pasa el nombre del parámetro del cual queremos cargar sus mediciones.

```
if(Beach.objects.filter(id=beach_id).count() and Beach_Monitoring.obje
cts.filter(beach_id=beach_id).filter(parameter=Parameter.objects.get(p
arameter_name=parameter)).count()>=4):
            beach = Beach.objects.get(id=beach_id)
            monitoreo = Beach_Monitoring.objects.filter(beach_id=beach
_id).filter(parameter=Parameter.objects.get(parameter_name=parameter))
            p_data = []
            for i in monitoreo:
                p={}
                p['Fecha'] = i.datetime
                p['Valor'] = i.value
                p_data.append(p)
```

Figura 1. Código para la conexión con la base de datos PostgreSQL

En el código anterior se puede observar la conexión con la base de datos y como el software adquiere los datos necesarios de la misma. Con la base de datos cargada correctamente en el programa podemos pasar a la transformación de los datos en una serie temporal.

2.3 Diseño del modelo predictivo

2.3.1 Transformación de los datos en una serie temporal

Una serie temporal se obtiene con la medición de una variable (o un conjunto de variables) de manera regular a lo largo de un período de tiempo. Las transformaciones de los datos de serie temporal suponen una estructura de archivo de datos en la que cada caso (fila) representa un conjunto de observaciones para un momento diferente y la duración del tiempo entre los casos es uniforme.

Una serie de procedimientos de transformación de datos que se proporciona en el sistema básico es útil en el análisis de series temporales. Estas transformaciones solo se aplican a los datos basados en columna, donde cada campo de serie temporal contiene los datos para una sola serie temporal. (IBM, Transformaciones de los datos, 2022)

- El procedimiento Definir fechas (en el menú Datos) genera las variables de fecha que se utilizan para establecer la periodicidad y para distinguir entre períodos históricos, de validación y de previsión. Predicciones está diseñado para trabajar con las variables creadas por el procedimiento Definir fechas.
- El procedimiento Crear serie temporal (del menú Transformar) crea nuevas variables de series temporales como funciones de variables de series temporales existentes. Se incluyen aquí funciones que utilizan observaciones vecinas para el suavizado, el promedio y la diferenciación.
- El procedimiento Reemplazar los valores perdidos (del menú Transformar) reemplaza los valores perdidos del sistema y los valores perdidos del usuario por estimaciones basadas en uno de varios métodos. Los valores perdidos al principio o fin de una serie no suponen un problema especial; sencillamente acortan la longitud útil de la serie. Las discontinuidades que aparecen en mitad de una serie (datos *incrustados* perdidos) pueden ser un problema mucho más grave.

Los datos que contamos en el proyecto fueron medidos mensualmente por lo que la frecuencia de la función es 12 y desde el año 2000. En nuestro caso no es necesario definir ningún otro parámetro para la función. Plantear la fecha final de la muestra

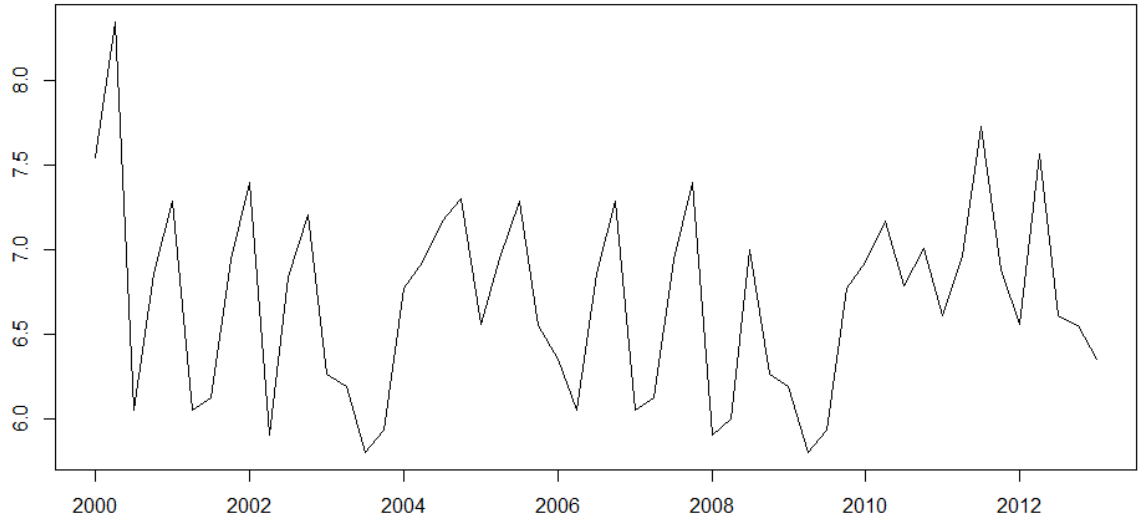
no es necesario puesto que está en constante crecimiento, según la cantidad de datos proporcionados por la base de datos, esta calculará la fecha final a partir de la inicial.

2.3.2 Análisis de la serie temporal

El análisis de series temporales es una técnica estadística que se ocupa de los datos de series temporales y el análisis de tendencias. Los datos de series temporales siguen intervalos de tiempo periódicos que se midieron en intervalos de tiempo regulares o se recopilaron en intervalos de tiempo particulares. En otras palabras, una serie temporal es simplemente una serie de puntos de datos ordenados en el tiempo, y el análisis de series temporales es el proceso de dar sentido a dichos datos. (TIBCO, 2022)

La forma más sencilla de comenzar el análisis de una serie temporal es mediante su representación gráfica. Las series temporales se representan en gráficos de frecuencias, los cuales son diagramas de líneas, en los que el tiempo se representa en el eje de abscisas (x), y la variable cuya evolución en el tiempo estudiamos en el eje de ordenadas (y). Para diagramas de dispersión simples, se usará plot. Sin embargo, existen métodos de trazado para muchos objetos en Python, incluidas funciones, marcos de datos y objetos de densidad.

Figura 2. Representación de serie temporal para el parámetro Oxígeno disuelto (OD)



Si todas las variables aleatorias que componen el proceso están idénticamente distribuidas, independientemente del momento del tiempo en que se estudie el proceso, entonces la serie es estacionaria.

Es decir, la función de distribución de probabilidad de cualquier conjunto de k variables (donde k es un número finito) del proceso debe mantenerse estable (inalterable) al desplazar las variables s períodos de tiempo tal que, si P(Yt+1,Yt+2,…,Yt+k) es la función de distribución acumulada de probabilidad, entonces: (Parra, 2019)

$P(Yt+1,Yt+2,\ldots,Yt+k)=P(Yt+1+s,Yt+2+s,\ldots,Yt+k+s),\ \forall t,k,s$

Una serie de tiempo se dice que es estrictamente estacionaria si sus propiedades no son afectadas por los cambios a lo largo del tiempo. Se deben cumplir tres criterios básicos para poder considerar a una serie de tiempo como estacionaria: (Briega, 2016)

- La media de la serie no debe ser una función de tiempo; sino que debe ser constante.
- La varianza de la serie no debe ser una función del tiempo.
- La covarianza de la serie no debe ser una función del tiempo.

Para verificar si la serie es estacionaria se calcula el p-valor y en los casos que el mismo es mayor que 0.05 se aplica la prueba Augmented Dickey-Fuller (ADF), y se hacen las diferenciaciones necesarias hasta que el p-valor<0.05.

2.3.3 Método de descomposición

La descomposición de series de tiempo es una tarea estadística que divide una serie de tiempo en varios componentes, cada uno de los cuales representa una de las categorías subyacentes de patrones. El método más utilizado es la descomposición basada en tasas de cambio.

Esta es una técnica importante para todo tipo de análisis de series de tiempo, especialmente para el ajuste estacional. Se busca construir, a partir de una serie temporal observada, una serie de series de componentes (que podrían usarse para reconstruir el original mediante sumas o multiplicaciones) donde cada una de estas tiene una determinada característica o tipo de comportamiento. Por ejemplo, las series de tiempo generalmente se descomponen en:

- T_t, el componente de tendencia en el tiempo t, que refleja la progresión a largo plazo de la serie (variación secular). Existe una tendencia cuando hay una dirección persistente de aumento o disminución en los datos. El componente de tendencia no tiene por qué ser lineal.
- C_t, el componente cíclico en el tiempo t, que refleja fluctuaciones repetidas pero no periódicas. La duración de estas fluctuaciones depende de la naturaleza de la serie temporal.

- S_t, el componente estacional en el tiempo t, que refleja la estacionalidad (variación estacional). Existe un patrón estacional cuando una serie de tiempo está influenciada por factores estacionales. La estacionalidad ocurre durante un período fijo y conocido (por ejemplo, el trimestre del año, el mes o el día de la semana).
- I_t, el componente irregular (o "ruido") en el tiempo t, que describe influencias irregulares y aleatorias. Representa los residuos o el resto de la serie temporal después de que se hayan eliminado los demás componentes.
- Por tanto, una serie de tiempo que utiliza un modelo aditivo puede considerarse como:

$y_t = T_t + C_t + S_t + I_t$,

Mientras que un modelo multiplicativo sería:

$y_t = T_t \times C_t \times S_t \times I_t$.

Para aplicar un modelo ARIMA ajustado es necesaria la transformación de la serie temporal en otra que sea aproximadamente estacionaria. Para esto se emplean técnicas como la diferenciación y logaritmos en dependencia de la no estacionariedad. Para determinar si la serie es o no estacionaria es necesario realizar pruebas de presencia de raíz unitaria, porque en caso afirmativo esto implicaría la no estacionariedad.

Según las fuentes consultadas, las pruebas más utilizadas para este fin son aplicadas a nuestra serie para el análisis de raíz unitaria.

Prueba Dickey-Fuller aumentada (ADF) para el análisis de raíz unitaria. Esta prueba es una versión aumentada de la prueba Dickey-Fuller para un conjunto más amplio y más complejo de modelos de series de tiempo. La estadística Dickey-Fuller Aumentada (ADF), utilizada en la prueba, es un número negativo. Cuanto más negativo es, más fuerte es el rechazo de la hipótesis nula de que existe una raíz unitaria para un cierto nivel de confianza. Para la realización de esta prueba dentro del modelo se implementó la función adfuller_test()

```
def adfuller_test(sales):
    result=adfuller(sales)
        labels = ['Estadística de prueba ADF','p-value','#Retrasos
utilizados','Número de observaciones']

    for value,label in zip(result,labels):
```

```
     print(label+' : '+str(value) )

    if result[1] <= 0.05:
        print("Fuerte evidencia en contra de la hipótesis nula (Ho), rechazar la hipótesis nula. Los datos son estacionarios")
    else:
       print("Evidencia débil contra la hipótesis nula, lo que indica que no es estacionaria ")
```

Figura 3. Función para realizar la prueba Augmented Dickey-Fuller (ADF)

Esta prueba da como resultado un p-value = 0.023466, a partir de un nivel de significancia del 95%, se rechaza la hipótesis nula por ser 0.023466<0.05. Por tanto, la serie es estacionaria.

```
 def adf_test(dataset):
            dftest = adfuller(dataset, autolag = 'AIC')
            print("1. ADF : ",dftest[0])
            print("2. P-Valor : ", dftest[1])
            return dftest[1]
dfv2= pd.DataFrame(p_data)
            dfv2 ['Valor'] = pd.to_numeric(dfv2 ['Valor'])
            dfv2 ['Fecha'] = pd.to_datetime(dfv2 ['Fecha'])
            dfv2 = dfv2.set_index('Fecha')
            dfv2 = dfv2.sort_values('Fecha')
            df= dfv2
            cont=0
            while(adf_test(dfv2)>0.05 and cont<5):
                dfv2=dfv2.diff().dropna()
                cont=cont+1
                adf_test(dfv2)
                print("Cant de diferenciaciones: ",cont)
```

Figura 4. Código para calcular el p-valor y realizar la prueba ADF

El número de diferencias a tomar de una serie es una aplicación de llamar recursivamente la función de diferencia n veces.

Una manera simple de ver una diferencia única (o "de primer orden") es verla como x (t) - x (t-k) donde k es el número de rezagos para volver. Las diferencias de orden superior son simplemente la reaplicación de una diferencia a cada resultado anterior. Esta función toma dos argumentos de nota. El primero es el retraso, que es el número de períodos, y el segundo son las diferencias, que es el orden de la diferencia. De igual forma la librería `statsmodels`, cuyo parámetro es la serie que queremos analizar y nos ayuda a determinar el número de diferenciaciones necesarias para que la serie sea estacionaria en media.

2.3.4 Modelo ARIMA

En estadística y econometría, en particular en series temporales, un modelo autorregresivo integrado de promedio móvil o ARIMA (acrónimo del inglés autoregressive integrated moving average) es un modelo estadístico que utiliza variaciones y regresiones de datos estadísticos con el fin de encontrar patrones para una predicción hacia el futuro. Se trata de un modelo dinámico de series temporales, es decir, las estimaciones futuras vienen explicadas por los datos del pasado y no por variables independientes. Fue desarrollado a finales de los sesenta del siglo XX. Box y Jenkins han desarrollado modelos estadísticos para series temporales que tienen en cuenta la dependencia existente entre los datos, esto es, cada observación en un momento dado es modelada a partir de los valores anteriores. Los análisis se basan en un modelo explícito.

Se suele expresar como ARIMA (p, d, q) donde los parámetros p, d y q son números enteros no negativos que indican el orden de las distintas componentes del modelo respectivamente, las componentes autorregresivas, integrada y de media móvil. Cuando alguno de los tres parámetros es cero, es común omitir las letras correspondientes del acrónimo. AR se refiere a la componente autorregresiva, I a la integrada y MA a la media móvil. Por ejemplo, ARIMA (0, 1, 0) se puede expresar como I (1) y ARIMA (0, 0,1) como MA (1).

El modelo ARIMA se puede representar como:

$$Y_t = -(\Delta^d Y_t - Y_t) + \phi_0 + \sum_{i=1}^{p} \phi_i \Delta^d Y_{t-i} - \sum_{i=1}^{q} \theta_i \varepsilon_{t-i} + \varepsilon_t$$

Donde d corresponde a las d diferencias que son necesarias para convertir la serie original en estacionaria, ϕ 1 , … , ϕ p son los parámetros pertenecientes a la parte

"autorregresiva" del modelo, θ 1 , … , θ q los parámetros pertenecientes a la parte "medias móviles" del modelo, ϕ 0 es una constante, y εt es el término de error.
Se debe tomar en cuenta que:
$\Delta Y_t = Y_t - Y_{t-1}$

Condiciones Necesarias para el Modelo ARIMA

- Los datos deben ser estacionarios, esto significa que las propiedades de la serie no dependen del momento en que se capturan. Una serie de ruido blanco y series con comportamiento cíclico también pueden considerarse series estacionarias.

- Los datos deben ser univariantes, ARIMA trabaja en una sola variable. La regresión automática tiene que ver con la regresión de los valores pasados.

Identificación practica del modelo: Identificar un modelo significa utilizar los datos recogidos, así como cualquier información de cómo se genera la serie temporal objeto de estudio, para sugerir un conjunto reducido de posibles modelos, que tengan muchas posibilidades de ajustarse a los datos. Ante una serie temporal empírica, se deben encontrar los valores (p, d, q) más apropiados.
Como la serie temporal presentó una tendencia, lo primero fue aplicar una diferenciación, de orden d. Una vez diferenciada la serie, una buena estrategia consiste en comparar los correlogramas de la función de autocorrelación (ACF) y la función de autocorrelación parcial (ACFP), proceso que suele ofrecer una orientación para la formulación del modelo orientativo.
Los procesos autorregresivos presentan función de autocorrelación parcial (ACFP) con un número finito de valores distinto de cero. Un proceso AR(p) tiene los primeros p términos de la función de autocorrelación parcial distintos de cero y los demás son nulos. En la práctica se considera que una muestra dada proviene de un proceso autorregresivo de orden p si los términos de la función de autocorrelación parcial son casi cero a partir del que ocupa el lugar p. Un valor se considera casi cero cuando su módulo es inferior a $\frac{2}{\sqrt{T}}$. Los programas de ordenador constituyen la franja $(-\frac{2}{\sqrt{T}} ; \frac{2}{\sqrt{T}})$ y detectan los valores de la ACFP que caen fuera de ella.
Los procesos de medias móviles presentan función de autocorrelación con un número finito de valores distintos de cero. Un proceso MA(q) tiene los primeros q términos de la función de autocorrelación distintos de cero y los demás son nulos. Las dos propiedades descritas son muy importantes con vistas a la identificación de un proceso mediante el análisis de las funciones de autocorrelación y autocorrelación parcial.

Para realizar los autocorrelogramas en Python se implementó el código que se muestra a continuación para la autocorrelación parcial y simple.

```
df['Sales First Difference'] = df[parametro] - df[parametro].shift(1)
df['Seasonal First Difference']=df[parametro]- df[parametro].shift(12)
fig = plt.figure(figsize=(12,8))
ax1 = fig.add_subplot(211)
fig = sm.graphics.tsa.plot_acf(df['Seasonal First
Difference'].dropna(),lags=15,ax=ax1)
ax2 = fig.add_subplot(212)
fig = sm.graphics.tsa.plot_pacf(df['Seasonal First
Difference'].dropna(),lags=15,ax=ax2)
plt.show()
```

Figura 5. Código para determinar la diferenciación y generar los gráficos autocorrelación parcial (PACF) y simple (ACF)

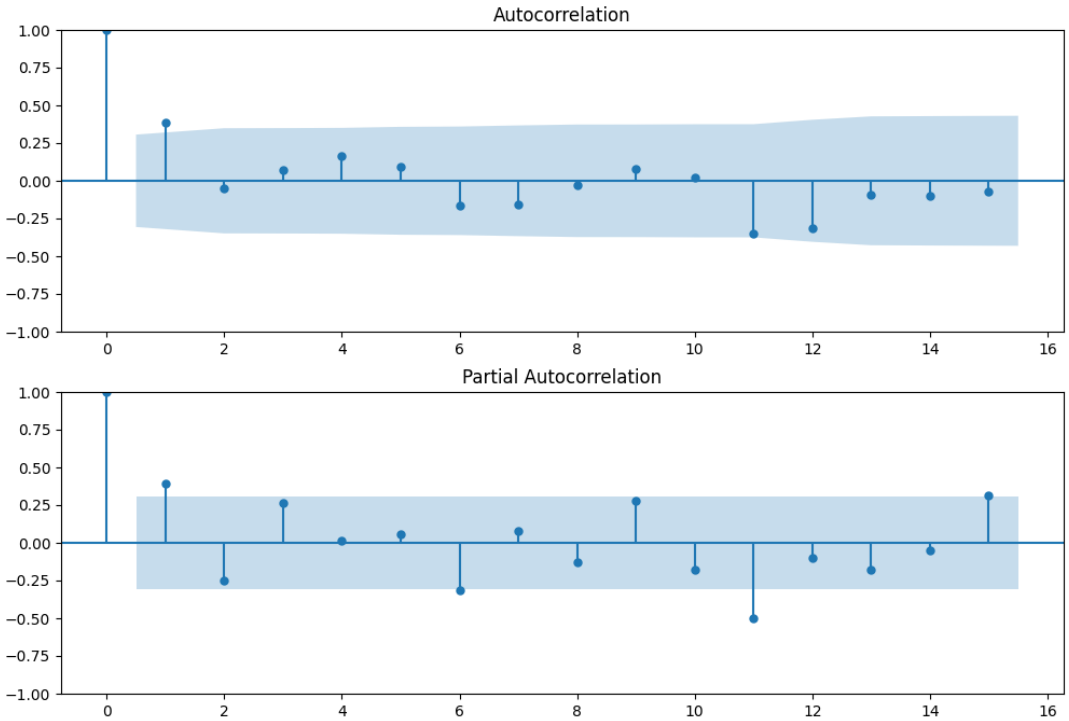


Figura 6. Gráficos autocorrelación parcial (PACF) y simple (ACF) del parámetro OD

Como el modelo ARIMA tendrá una parte estacional se analizarán los gráficos en los primeros retardos para la parte estacionaria y los retardos en cada periodo para el análisis de la parte estacional.

Con el análisis del grafico de autocorrelación parcial podemos ver que el primer retardo es diferente de 0 y que con el aumento de los periodos la función tiende a disminuir. A partir del aumento de los periodos pueden considerarse marcados el primer rezago.
A partir del gráfico de autocorrelación simple podemos ver que el primer retardo es diferente de 0 y que a partir de esta comienzan a variar entre positivos y negativos, esto indica que se toma el primero. Con el aumento de los periodos la función tiende a disminuir. A partir del aumento de los periodos pueden considerarse marcados en el rezago.
A partir de aquí se derivan dos variantes para el modelo para el parámetro OD:
ARIMA (1, 1, 1)
SARIMAX (1, 1, 1, 12)
La librería `statsmodels.tsa` tiene las funciones ARIMA () y SARIMAX (), para la construcción de estos tipos de modelos, donde se debe pasar como parámetro la serie de datos que se desea modelar y el orden, que son los valores de los modelos para los datos ajustados.

```
model=ARIMA(df[parametro],order=(1,1,1))
model_fit=model.fit()
model_fit.summary()
```

a)

```
model=sm.tsa.statespace.SARIMAX(df[parametro],order=(1,1,
1),seasonal_order=(1,1,1,12))
results=model.fit()
```

b)

Figura 7. a) Código para crear modelo ARIMA en python a partir de librería statsmodels

b) Código para crear modelo SARIMAX en python con el empleo de librería statsmodels

2.4 Diseño del módulo web

2.4.1 Pila del producto (SPRINT BLACKLOG)

La **pila producto Scrum** (Product Backlog Scrum, en inglés) es una lista ordenada donde van enlistados todos los elementos que el Product Owner cree se necesitarán para llevar a cabo el proyecto. Es, en palabras más sencillas, una lista priorizada de todas las cosas que el cliente quiere.

Para la construcción de la pila del producto hay que tener presente una serie de aspectos:

- **Código:** Muestra un código elegido por el equipo de trabajo para identificar cada pila del producto.
- **Prioridad:** Muestra en una escala de “baja, media y alta” la prioirdad del desarrollo de la pila del producto por el equipo de desarrollo.
- **Descripción:** Muestra los requisitos o funcionalidades para la realización del producto.
- **Estimado(hrs):** Muestra el estimado en horas que necesita el equipo de trabajo para realizar cada requisito de la pila de producto.

Tabla 1. Pila de producto

PILA DE PRODUCTO			
Código	**Prioridad**	**Descripción**	**Estimado (hrs)**
p01	Alta	Investigar la plataforma tecnológica	24
p02	Alta	Analizar y diseñar la Base de Datos del monitoreo	72
p03	Alta	Diseñar de la Interface de Usuario	56
p09	Media	Crear Vista de Monitoreo	73
p22	Alta	Gestionar Monitoreo	90
p23	Alta	Crear Mecanismo de Predicciones	30
p25	Alta	Crear el Mecanismo del Reporte de monitoreo	12
p26	Baja	Crear Mecanismo de Búsqueda de parámetros	10

2.4.1.1 Requisitos No Funcionales

Requisitos No Funcionales.

1) El módulo del sistema debe ajustarse a los colores y diseño del sistema en general.
2) El módulo del sistema de contener colores acordes al diseño original del sistema.
3) El módulo del sistema debe contener el logotipo del Observatorio de Costa Atenas.
4) Toda funcionalidad del sistema y transacción de negocio debe responder al usuario en menos de 5 segundos.

5) El sistema debe ser capaz de operar adecuadamente con hasta 10.000 usuarios con sesiones concurrentes.
6) Los datos modificados en la base de datos deben ser actualizados para todos los usuarios que acceden en menos de 2 segundos.
7) Los permisos de acceso al sistema podrán ser cambiados solamente por el administrador general del sistema.
8) El nuevo sistema debe desarrollarse a partir de análisis de patrones y recomendaciones de programación que incrementen la seguridad de datos.
9) El tiempo de aprendizaje del sistema por un usuario deberá ser menor a 4 horas.
10) El sistema debe proporcionar mensajes de error que sean informativos y orientados a usuario final.
11) La aplicación web debe poseer un diseño “Responsive” a fin de garantizar la adecuada visualización en múltiples computadores personales, dispositivos tableta y teléfonos inteligentes.
12) El sistema debe poseer interfaces gráficas bien formadas.
13) El sistema debe tener una disponibilidad del 99,99% de las veces en que un usuario intente accederlo.
14) El tiempo para iniciar o reiniciar el sistema no podrá ser mayor a 5 minutos.
15) La tasa de tiempos de falla del sistema no podrá ser mayor al 0,5% del tiempo de operación total.
16) La aplicación debe ser compatible con todas las versiones de navegadores web.
17) La aplicación deberá consumir menos de 500 Mb de memoria RAM al ser operada por un navegador.
18) El sistema deberá ser multiplataforma y ser compatible con los sistemas operativos Windows y Linux. Deberá funcionar con los siguientes requisitos mínimos de software: Sistema operativo: Windows XP o superior. Linux. Gestor de base de datos: Postgresql v8.0 o superior.
19) El sistema deberá proteger la información que se maneje, de acceso no autorizado y divulgación, a partir de los diferentes roles de los usuarios que empleen el sistema, es decir tener un control de usuarios.
20) El sistema deberá encontrarse disponible las 24 horas de todos los días para aquellos usuarios autorizados a acceder al sistema.

2.4.2 Pila de Sprint

A partir de la Pila de Producto se crea la Pila de Sprint que descompone las funcionalidades de la Pila de Producto en las tareas necesarias para construir un incremento: una parte completa y operativa del producto. Además, se planificarán de forma independiente y detallada cada sprint, para lo que se descompone el trabajo en unidades de tamaño adecuado para monitorear el avance a diario, e identificar riesgos y problemas sin necesidad de procesos de gestión complejos.

A continuación, se muestra la Pila de Sprint para los elementos de la Pila de producto elaborada con anterioridad.

Tabla 2. PILA DE SPRINT

PILA DE SPRINT					
Sprint	**Pila de Producto**	**Encargado**	**Fecha Inicial**	**Fecha Final**	**Valor**
1	Investigar la plataforma tecnológica Analizar y diseñar la base de Datos del monitoreo Diseñar la Interfaz de usuario	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	15/10/2022	26/10/2022 (11 días)	152
2	Crear Vista de Monitoreo Gestionar Monitoreo	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	27/10/2022	09/11/2022 (13 días)	163 (sumado: 295)
3	Crear Mecanismo de Predicciones Crear el Mecanismo del Reporte de monitoreo	Yadian Noda Rodriguez Yaisel González Hernández	10/11/2022	18/11/2022 (8 días)	52 (Sumado: 347)

	Crear Mecanismo de Búsqueda de parámetros	Eduardo J Berrio Turiño			

2.4.3 Planificación de Sprint del Proyecto

Tabla 3. SPRINT 1

SPRINT 1				
Pila del Producto	**Tareas**	**Encargado**	**Est. Inicial**	**Est. Total**
Investigar la plataforma tecnológica	- Definir las características mínimas del hardware - Seleccionar el gestor de base de datos - Seleccionar los lenguajes de programación y herramientas de desarrollo - Realizar pruebas rendimiento al hardware instalado - Realizar pruebas de rendimiento al gestor de base de datos	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	16	152
	- Comprobar que los lenguajes de programación y las herramientas seleccionadas cumplen las exigencias del proyecto	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	4	
Analizar y diseñar de la Base de Datos	- Analizar y definir los tipos de datos - Normalizar la base de datos - Crear las clases de la base de datos - Generar la base de datos	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	89	

	- Cargar datos de prueba	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	5	
Diseñar de la Interface de Usuario	- Seleccionar colores del diseño - Diseñar la interface de usuario - Realizar la maquetación del diseño - Crear la lógica de JavaScript del diseño	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	32	
	- Comprobar la funcionalidad y adaptabilidad del diseño en los diferentes dispositivos y navegadores	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	6	

Tabla 4. SPRINT 2

SPRINT 2				
Pila del Producto	**Tareas**	**Encargado**	**Est. Inicial**	**Est. Total**
Crear Vista de Monitoreo	- Crear y maquetar la vista - Generar el formulario de contacto - Crear lógica de envío de información - Crear la lógica de JavaScript para la validación del formulario	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	67	163

	- Comprobar la correcta funcionabilidad de la vista - Comprobar que la información sea correcta y con el formato adecuado según el diseño - Comprobar la validación de los datos del formulario	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	16	
Gestionar Monitoreo	- Crear y maquetar la vista general - Crear y maquetar la vista detalles - Recuperar la información de la base de datos y mostrarla en la vista correspondiente -Crear mecanismo de asignación de roles	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	62	
	- Comprobar la correcta funcionabilidad de la vista - Comprobar que la información sea correcta y con el formato adecuado según el diseño	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	18	

Tabla 5. SPRINT 3

SPRINT 3				
Pila del Producto	**Tareas**	**Encargado**	**Est. Inicial**	**Est. Total**
Crear Mecanismo de Predicciones	- Crear y maquetar la vista - Recuperar la información de la base de datos y mostrarla en la vista - Generar los formularios de la vista	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	25	52

	- Comprobar la correcta funcionabilidad de la vista - Comprobar que la información sea correcta y con el formato adecuado según el diseño	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	5	
Crear el Mecanismo del Reporte de monitoreo	- Crear y maquetar la vista general - Crear y maquetar la vista detalles - Recuperar la información de la base de datos y mostrarla en la vista correspondiente - Crear mecanismo de asignación de roles	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	10	
	- Comprobar la correcta funcionabilidad de la vista - Comprobar que la información sea correcta y con el formato adecuado según el diseño	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	2	
Crear Mecanismo de Búsqueda de parámetros	- Crear y maquetar la vista general - Crear y maquetar la vista detalles - Recuperar la información de la base de datos y mostrarla en la vista correspondiente - Crear mecanismo de asignación de roles	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	8	
	- Comprobar la correcta funcionabilidad de la vista - Comprobar que la información sea correcta y con el formato adecuado según el diseño	Yadian Noda Rodriguez Yaisel González Hernández Eduardo J Berrio Turiño	2	

2.4.4 Historias de Usuario

Las historias de usuario definen las funcionalidades del sistema y son elaboradas de manera colaborativa entre el dueño del producto y el equipo de desarrollo. Las historias de usuario deben redactarse de manera clara para todo el equipo incluyendo el dueño del producto. Al conjunto de todas las historias de usuario se lo conoce como el product backlog.

Dentro del modelo de las historias de usuarios se presentan los siguientes campos:

Código: Identificador de la historia de usuario.

Nombre: Nombre de la funcionalidad a desarrollar.

Tipo: Tipo de historia de usuario.

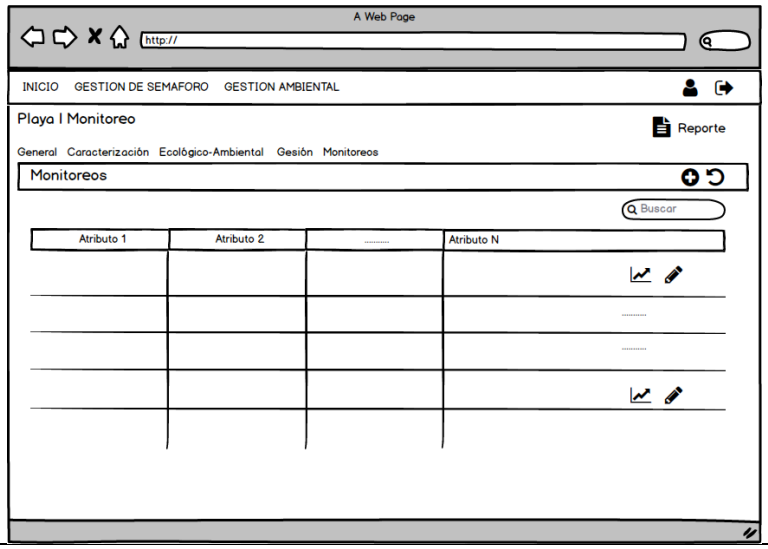
Actor: Quien la realiza.

HU Relacionada: Historia de Usuario con la que se vincula.

Descripción: Descripción de la historia de usuario, acción que el usuario va a realizar.

Resultado: Resultado esperado

Tabla 6. Historia de usuario Gestionar Monitoreo

<table>
<tr><th colspan="2">HISTORIA DE USUARIO</th></tr>
<tr><td>Código:hu01</td><td>Nombre: Gestionar Monitoreo</td></tr>
<tr><td>Tipo de HU: Funcional</td><td>Complejidad: Alta</td></tr>
<tr><td>Actor: Gestor</td><td>HU Relacionadas: No</td></tr>
<tr><td colspan="2">Descripción:
El usuario podrá insertar un nuevo monitoreo y modificar los monitoreos existentes</td></tr>
<tr><td>Prototipo
 
</td><td>Resultado:
Se crea un nuevo monitoreo con los datos suministrados por el usuario.</td></tr>
</table>

2.6 Conclusiones del Capítulo

El diseño descrito anteriormente demuestra que:

1. Se logró la implementación de modelos predictivos que permitirán procesar los datos y realizar inferencias futuras.
2. Es necesario un histórico de datos amplio para lograr valores de predicción óptimos.
3. Para el método de descomposición los modelos ARIMA (1, 1, 1), SARIMAX (1,1, 1, 12) resultaron ser los de mejor ajuste.
4. El levantamiento de los requisitos funcionales y su descripción fue fundamental para lograr un diseño más apropiado para la realización del proyecto.
5. Se proporciona una visión más completa del módulo a desarrollar dado que se modelan todos los procesos que intervienen en el mismo.
6. Se obtuvo el diseño de la interfaces web a partir de la metodología Scrum.

Capítulo III “Propuesta de solución práctica al problema científico”

3.1 Introducción del Capítulo

Con la propuesta de modelos ya terminada solo queda analizarla con el fin de corroborar los resultados que se obtienen. Se realizaron diferentes pruebas, con el objetivo de mejorar los resultados con cada una de ellas. Todos estos ensayos se explicarán en el presente capítulo, así como un pequeño análisis y comparación de los mismos.

3.2 Validación de los modelos

La prueba de Ljung-Box es un tipo de prueba estadística de si un grupo cualquiera de autocorrelaciones de una serie de tiempo son diferentes de cero. En lugar de probar la aleatoriedad en cada retardo distinto, esta prueba la aleatoriedad "en general" basado en un número de retardos.

Esta prueba también es conocida como la prueba Q de Ljung-Box, y está estrechamente relacionada con la prueba de Box-Pierce. Esta es una versión simplificada de la estadística de Ljung-Box para los cuales los estudios de simulación posteriores han demostrado un rendimiento deficiente.

La prueba de Ljung-Box se puede definir de la siguiente manera.

H0: Los datos se distribuyen de forma independiente (es decir, las correlaciones en la población de la que se toma la muestra son 0, de modo que cualquier correlación observada en los datos es el resultado de la aleatoriedad del proceso de muestreo).

Ha: Los datos no se distribuyen de forma independiente.

La estadística de prueba es:

$$Q = n(n+2)\sum_{k=1}^{h}\frac{p_k^2}{n-k}$$

donde n es el tamaño de la muestra, $p_k$es la autocorrelación de la muestra en el retraso, k y h es el número de retardos que se prueban. Por nivel de significación α, la región crítica para el rechazo de la hipótesis de aleatoriedad es

$$Q > x_{1-\alpha,h}^2$$

donde $x_{1-\alpha,h}^2$ es la α- cuantil de la distribución chi-cuadrado con m grados de libertad.

Para que el modelo seleccionado sea validado tiene que tener los residuales estacionarios, normalizados e independientes. Para esto se realiza la prueba de ruido

blanco o Ljung-Box. Un ruido blanco es una serie tal que su media es cero, la varianza es constante y no se puede correlacionar.

E(at)=0

Var(at)=σ2a

cov(at, at+h) =0

Se trata de un proceso en el que todas sus variables son independientes.

Modelo ARIMA:

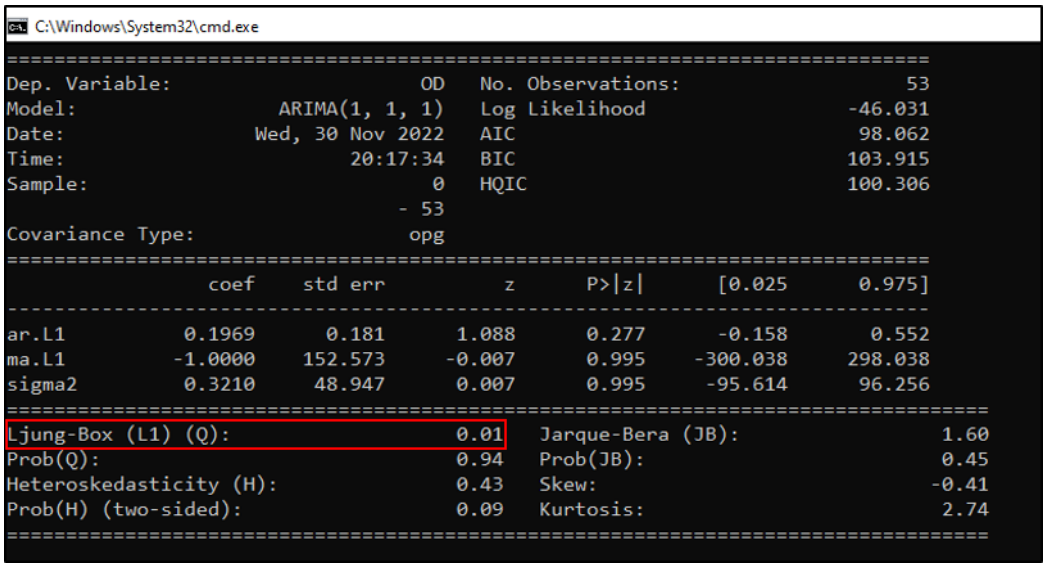


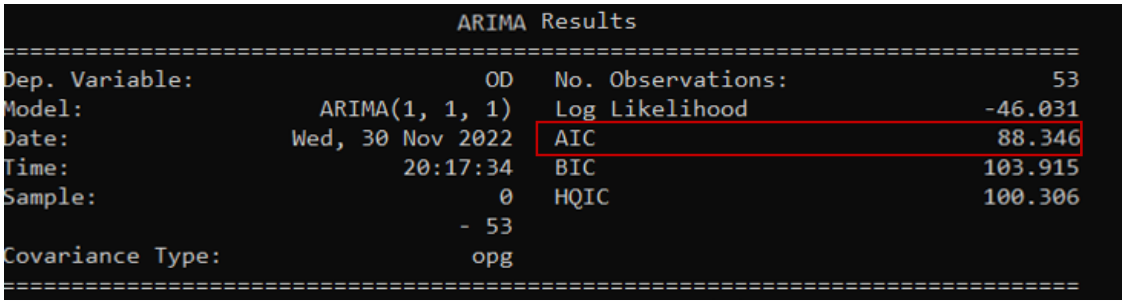
Figura 8. Resultado prueba Ljung-Box generado por python

Los resultados de este test aceptan en todos los modelos la hipótesis nula. Esto significa que los residuales se distribuyen como un ruido blanco. Por tanto, estos presentan estacionariedad, normalidad e independencia, lo cual implica que se está en presencia de modelos adecuados para la predicción.

3.2.1 Selección del modelo

Para seleccionar el modelo que mejor ajuste posee nos apoyaremos en AIC. El criterio de información de Akaike (AIC) es una medida de la calidad relativa de un modelo estadístico, para un conjunto dado de datos. Como tal, el AIC proporciona un medio para la selección del modelo. AIC maneja un sacrificio entre la bondad de ajuste del modelo y la complejidad del mismo. Se basa en la entropía de información: se ofrece una estimación relativa de la información perdida cuando se utiliza un modelo determinado para representar el proceso que genera los datos.

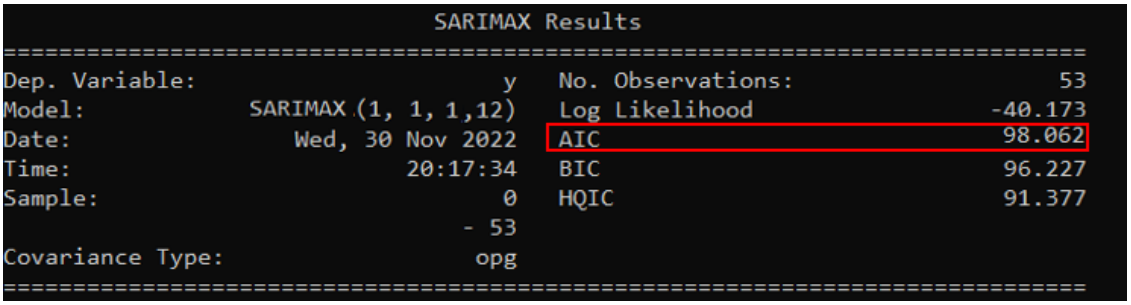
AIC no proporciona una prueba de un modelo en el sentido de probar una hipótesis nula, es decir AIC puede decir nada acerca de la calidad del modelo en un sentido absoluto. Para conocer el AIC de los modelos se vuelve a utilizar la función summary(), pero esta vez se le pasaran los modelos como parámetros. El modelo que tenga menor AIC será el más ajustado a los datos, porque será menor la información perdida con dicho modelo.



ARIMA Results

Dep. Variable:	OD	No. Observations:	53
Model:	ARIMA(1, 1, 1)	Log Likelihood	-46.031
Date:	Wed, 30 Nov 2022	AIC	88.346
Time:	20:17:34	BIC	103.915
Sample:	0	HQIC	100.306
	- 53		
Covariance Type:	opg		

Figura 9. Resumen Modelo ARIMA (1, 1, 1)



SARIMAX Results

Dep. Variable:	y	No. Observations:	53
Model:	SARIMAX(1, 1, 1,12)	Log Likelihood	-40.173
Date:	Wed, 30 Nov 2022	AIC	98.062
Time:	20:17:34	BIC	96.227
Sample:	0	HQIC	91.377
	- 53		
Covariance Type:	opg		

Figura 10. Resumen Modelo SARIMAX (1, 1, 1, 12)

3.2.2 Predicción de datos

Después de obtener el modelo más ajustado a los datos es posible realizar las estimaciones. Para esto nos auxiliamos de la función forecast de la librería de igual nombre. Esta función es utilizada para predecir tanto series temporales como modelos de series temporales, solamente es necesario especificar el modelo entrenado que deseamos predecir, así como la cantidad de predicciones.

```
df['forecast']=results.predict(start=53,end=65,dynamic=True)
```

53	6.037710
54	6.656255
55	6.443005
56	6.410401
57	6.691202
58	6.676702
59	6.667485
60	6.661627
61	6.657903
62	6.655537
63	6.654032
64	6.653076
65	6.652468

Figura 11. Predicción Realizada por el Modelo

Los valores mostrados anteriormente equivalen a las predicciones de la variable OD con el modelo más ajustado (1, 1, 1).

3.2.3 Análisis de los resultados

Para evaluar el modelo planteado para la solución se utilizó el indicador de Porcentaje de Error Medio Absoluto (MAPE) por su fácil interpretación, el cual se calcula con la siguiente ecuación:

$$\boldsymbol{MAPE} = \frac{\sum_{t=1}^{n} \left| \frac{(y_t - \hat{y_t})}{y_t} (100) \right|}{n}$$

Dónde:

$\boldsymbol{y_t}$: Es el valor observado (valor real del indicador)

$\boldsymbol{\hat{y_t}}$: Es el valor pronosticado (predicción del indicador)

$\boldsymbol{n}$: Es la cantidad de observaciones

Según MAPE la clasificación del pronóstico depende del % de error obtenido como se muestra en la tabla siguiente:

Tabla 6 Clasificaciones de MAPE

% de Error MAPE	Clasificación del pronóstico
Menor de 10%	Alta precisión
10% -20%	Buena precisión

20% - 50%	Precisión razonable
Mayor del 50%	Poco fiable

Para continuar con el ejemplo de OD, al ejecutar el MAPE se obtuvo un valor aproximado de 4.581749743056838% que si nos fijamos en la tabla está por debajo del 10% y por lo tanto se considera un pronóstico de alta precisión, lo que nos demuestra que es un modelo altamente confiable para la predicción de escenarios futuros.

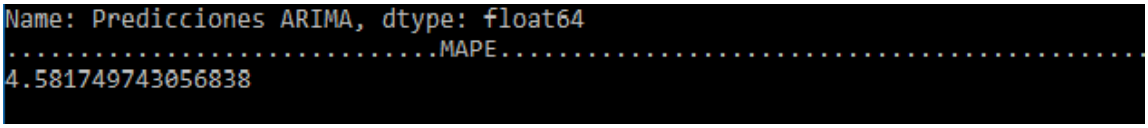


Figura 12. Cálculo del MAPE

3.3 Validación módulo web monitoreo

3.3.1 Descripción de la propuesta de solución

El módulo Monitoreo del sistema de gestión de playas se desarrolló para elevar la eficiencia y eficacia en el proceso que se lleva a cabo a la hora de realizar el monitoreo ambiental en las playas a través de la recolección y almacenamiento de datos mediante mediciones u observaciones de variables previamente identificadas, los indicadores, historial de datos que posteriormente es aprovechado por el modelo de predicción antes desarrollado. El módulo cuenta con varias funcionalidades las cueles se especifican a continuación:

Gestionar Monitoreo

Gestionar Parámetros

Generar Reportes

Graficar datos y predicciones

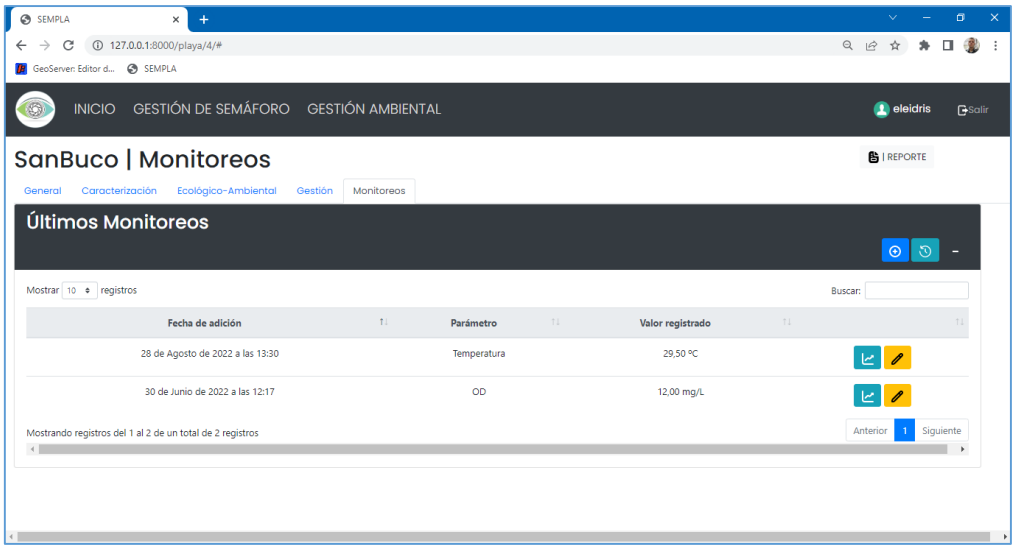


Figura 12. Vista de Monitoreo

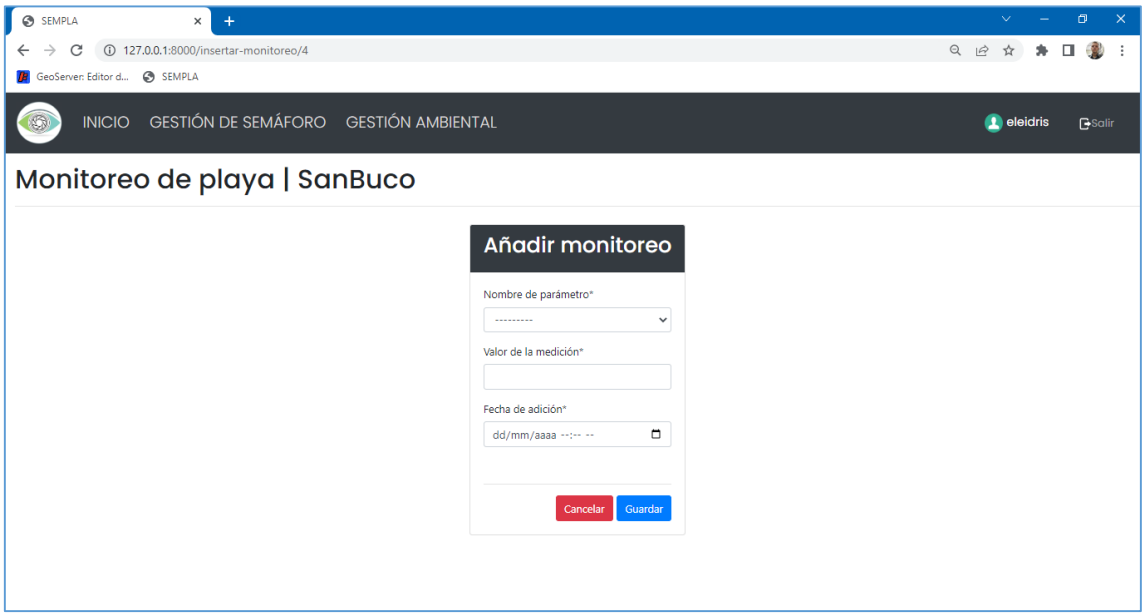


Figura 13. Añadir monitoreo

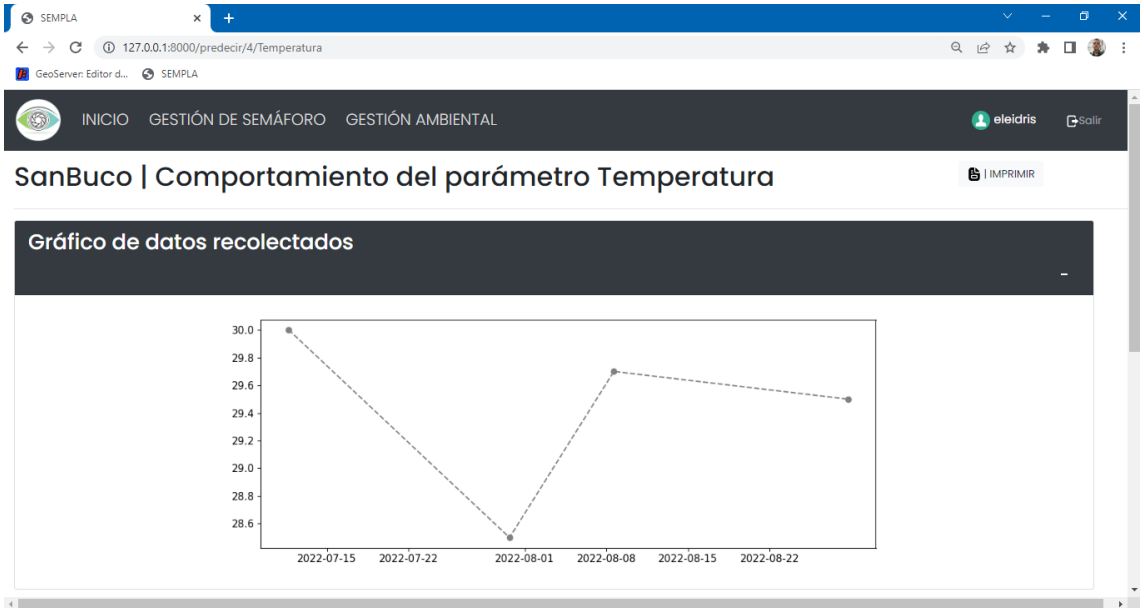


Figura 15. Gráfico de Comportamiento OD

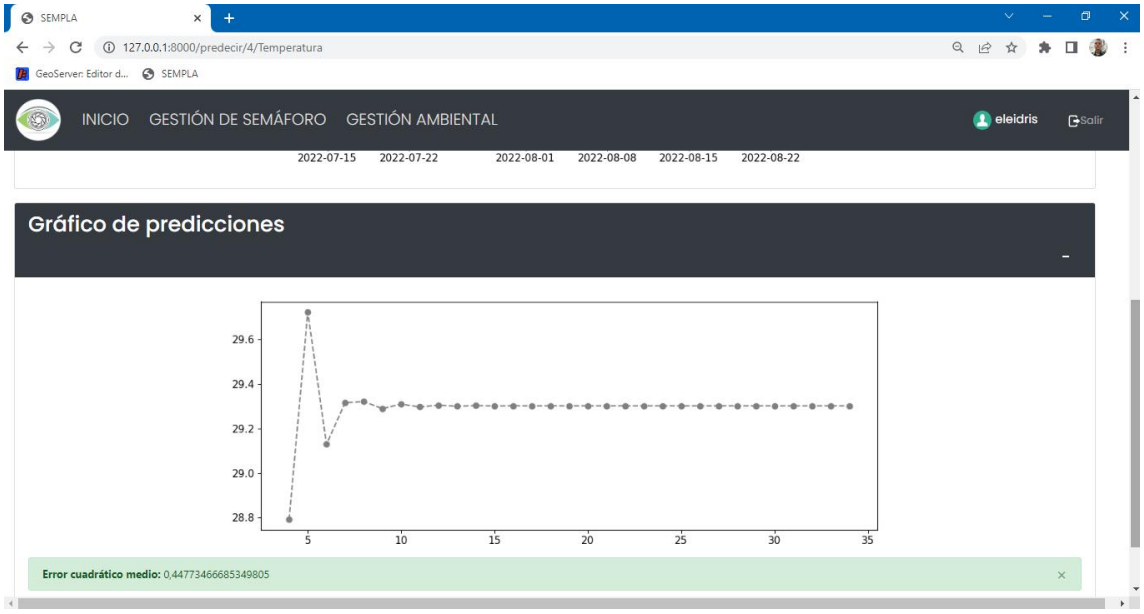


Figura 15. Gráfico de predicciones del parámetro OD

3.3.2 Pruebas

La prueba de software es el proceso de evaluar y verificar que un producto o aplicación de software hace lo que se supone que debe hacer. Los beneficios de las pruebas incluyen la prevención de errores, la reducción de los costos de desarrollo y la mejora del rendimiento.(IBM, 2022)

Se aplicaron al software un conjunto de pruebas que dejaron evidenciado el buen funcionamiento del mismo. A continuación se muestra una breve introducción a algunas de estas pruebas, dígase pruebas funcionales, pruebas de integración, pruebas de aceptación y pruebas de caja negra, y se muestra en profundidad el mecanismo empleado para la realización de las pruebas de aceptación y las pruebas de caja negra.

Pruebas funcionales

Las pruebas funcionales se llevan a cabo para comprobar las características críticas para el negocio, la funcionalidad y la usabilidad. Las pruebas funcionales garantizan que las características y funcionalidades del software se comportan según lo esperado sin ningún problema. Valida principalmente toda la aplicación con respecto a las especificaciones mencionadas en el documento Software Requirement Specification (SRS). Los tipos de pruebas funcionales incluyen pruebas unitarias, pruebas de interfaz, pruebas de regresión, además de muchas. (Loadview-Testing, 2022)

Pruebas de integración

Las pruebas de integración dentro del software testing chequean la integración o interfaces entre componentes, interacciones con diferentes partes del sistema, como un sistema operativo, sistema de archivos y hardware o interfaces entre sistemas. Las pruebas de integración son un aspecto clave del software testing. (Trans-TI, 2022)

Pruebas de aceptación

El objetivo de las pruebas de aceptación es validar que un sistema cumpla con el funcionamiento esperado, lo que permite al usuario de dicho sistema que determine su aceptación, desde el punto de vista de su funcionalidad y rendimiento. (Cillero, 2022)

Tabla 7 Prueba de aceptación 1

Código: 01	**HU:** Agregar zona
Descripción: Verificar los requisitos de la zona que se va a agregar	
Condiciones de ejecución: Que la aplicación se encuentre en ejecución	
Entradas: Los datos de la zona	
Resultado esperado:	

Se agrega la zona al sistema
Evaluación: Prueba satisfactoria

Tabla 8 Prueba de aceptación 2

Código: 02	**HU:** Buscar parámetro
Descripción: Verificar que el parámetro se encuentra en el sistema	
Condiciones de ejecución: Que la aplicación se encuentre en ejecución	
Entradas: El nombre del parámetro	
Resultado esperado: Se muestra el parámetro que se buscó	
Evaluación: Prueba satisfactoria	

Tabla 9 Prueba de aceptación 3

Código: 03	**HU:** Generar reporte
Descripción: Verificar que hayan datos agregados en el sistema	
Condiciones de ejecución: Que la aplicación se encuentre en ejecución	
Entradas: Datos	
Resultado esperado: Se genera el reporte	
Evaluación: Prueba satisfactoria	

Pruebas de caja negra

Las pruebas de caja negra, conocidas también como black box testing, pueden definirse como una técnica donde se busca la verificación de la funcionalidad del software o aplicación analizada, sin tomar, referente la estructura del código interno, las rutas de tipo internas ni la información referente a la implementación. Esto quiere decir que la prueba se lleva a cabo con desconocimiento del funcionamiento del sistema interno,

debido a que se enfoca en la entrada y salida de un software, a partir de sus especificaciones y requisitos de manera que se puede asegurar que el objetivo de las pruebas de caja negra está relacionado con la validación de los recursos funcionales del software o aplicación que se busca examinar. (Redacción KeepCoding, 2022)

Tabla 10 Prueba de caja negra 1

Funcionalidad			Gestionar parámetro	
Código			C01	
Pre-Requisito			Usuario registrado en el sistema con el rol de gestor	
No	Nombre	Descripción	Respuesta esperada	Respuesta obtenida
1	Adicionar	El gestor agrega un nuevo parámetro	Parámetro agregado satisfactoriamente	OK
2	Modificar	El gestor modifica un parámetro existente	Parámetro modificado satisfactoriamente	OK
3	Eliminar	El gestor elimina un parámetro existente	Parámetro eliminado satisfactoriamente	OK

Tabla 11 Prueba de caja negra 2

Funcionalidad			Parámetros en blanco	
Código			C02	
Pre-Requisito			Usuario registrado en el sistema con el rol de gestor	
No	Nombre	Descripción	Respuesta esperada	Respuesta obtenida
1	Validación de campo	Se verifica que todos los campos obligatorios estén llenos	Se adicionó el parámetro correctamente	OK

Tabla 12 Prueba de caja negra 3

Funcionalidad	Ver comportamiento de parámetro
Código	C03

Pre-Requisito			Usuario registrado en el sistema con el rol de gestor	
No	Nombre	Descripción	Respuesta esperada	Respuesta obtenida
1	Obtención de gráfico de parámetro	El gestor accede al gráfico de un parámetro en específico	Se muestra el gráfico del parámetro	OK

3.4 Conclusiones del Capítulo

En este capítulo se expuso todo lo referente a la experimentación, discusión y análisis de los resultados obtenidos, una vez detallados estos aspectos se arribaron a las siguientes conclusiones:

1. De los modelos de descomposición, el modelo ARIMA es muy aconsejable en datos similares a estos.
2. De los modelos ARIMA analizados el que más se ajusta a los datos es ARIMA (1,1,1) según el criterio de información de Akaike.
3. Los resultados obtenidos en la predicción de indicadores comprobaron que los modelos implementados son fiables y que sus pronósticos tienen un ¨alto grado de precisión¨.
4. Los modelos predictivos son derivados de la simulación y no de la subjetividad de los investigadores, lo cual provee de solidez y rigor en la toma de decisiones, abriendo un mayor espectro para su uso a partir de sus propiedades estadísticas.
5. El hecho de que las predicciones del software sean muy cercanas a la realidad permite emitir criterios acertados para evaluar una situación en un espacio de tiempo determinado.

Conclusiones

Como resultado de esta investigación quedaron satisfechos los objetivos trazados y se arribó a las siguientes conclusiones:

1. El estudio realizado sobre los antecedentes, el estado actual de la temática, la bibliografía y documentos relacionados con el objeto de estudio, permitió aportar los elementos necesarios para dar solución a la problemática planteada.
2. Los antecedentes encontrados, vinculados al tema no le dan solución al problema planteado por lo que no es factible su utilización.
3. Se utilizaron las herramientas de software más factibles para la construcción de la solución.
4. Se logró la implementación de modelos predictivos que permitirán procesar los datos y realizar inferencias futuras.
5. Es necesario un histórico de datos amplio para lograr valores de predicción óptimos.
6. Para los métodos analizados el modelo ARIMA (1,1,1) resultó ser el que mejores resultados arroja.
7. Los resultados obtenidos en la predicción de indicadores comprobaron que los modelos implementados son fiables y que sus pronósticos tienen un ¨alto grado de precisión¨.
8. Los modelos predictivos son derivados de la simulación y no de la subjetividad de los investigadores, lo cual provee de solidez y rigor en la toma de decisiones, abriendo un mayor espectro para su uso a partir de sus propiedades estadísticas.
9. El hecho de que las predicciones del software sean muy cercanas a la realidad permite emitir criterios acertados para evaluar una situación en un espacio de tiempo determinado.

Recomendaciones

A partir de vista del alcance del presente trabajo y teniendo en cuenta el momento de desarrollo del mismo, se proponen las siguientes recomendaciones:

1. Actualizar las series que sirven como base a los pronósticos, con la incorporación de valores reales de las variables empleadas, para renovar las previsiones.
2. Realizar la comprobación periódica del cumplimiento de los valores pronosticados.

Referencias

López Jiménez, K. J., & Villanueva Vásquez, W. J. (2020). *Modelos Arima univariante de series temporales para la producción y demanda de agua en el distrito de Lambayeque, periodo 2002 – 2017.* Lambayeque.

Vásquez Mejía, E. J., & Chavez Gonzales, S. (2019). TRABAJO TEÓRICO EXPERIMENTAL Predicción del consumo de energía eléctrica residencial de la Región Cajamarca mediante modelos Holt-Winters. *Ingeniería Energética, 40*(3), 181-191.

Arnau, J. (1981). *Uso de los modelos de series temporales como técnica de análisis de los diseños conductuales.* Barcelona.

Becker, R. A., Chambers, J. M., & Wilks, A. R. (1988). *The New S Language.* Wadsworth & Brooks/Cole.

Briega, R. E. (2016). Series de tiempo con Python. En R. E. Briega, *Matemáticas, análisis de datos y python.*

BSD. (8 de Junio de 2022). *Scikit-Learn, herramienta básica para el Data Science en Python*. Obtenido de Scikit-Learn, herramienta básica para el Data Science en Python: https://www.master-data-scientist.com/scikit-learn-data-science/

Build, M. (1 de junio de 2022). *Visual Studio*. Obtenido de Visual Studio: https://visualstudio.microsoft.com/es/

Casimiro, M. P. (2009). *Técnicas de predicción económicas.* País Vasco.

CEACES. (6 de Junio de 2022). *Series Temporales*. Obtenido de Series Temporales: https://www.uv.es/ceaces/series/series.htm

CEACES. (11 de Junio de 2022). *Series Temporales*. Obtenido de Series Temporales: https://www.uv.es/ceaces/series/series.htm

CEACES. (11 de Junio de 2022). *Series Temporales*. Obtenido de Series Temporales: https://www.uv.es/ceaces/series/series.htm

CEACES. (11 de Junio de 2022). *Series Temprales*. Obtenido de Series Temprales: https://www.uv.es/ceaces/series/series.htm

Cillero, M. (25 de junio de 2022). *Pruebas de aceptación*. Obtenido de Pruebas de aceptación: manuel.cillero.es

Foundation, P. S. (1 de Junio de 2022). *Python*. Obtenido de Python: https://www.python.org

GitHub. (8 de Junio de 2022). *NumPy*. Obtenido de NumPy: numpy.org

González Vidal, A. (2020). *Análisis de datos en entornos inteligentes basados en el Internet de las cosas.* Murcia: Universidad de Murcia.

Google. (8 de Junio de 2022). *Keras*. Obtenido de Keras: keras.io

Group, P. G. (8 de Junio de 2022). *PostgreSQL*. Obtenido de PostgreSQL: https://www.postgresql.org/

Hunter, J. D. (8 de Junio de 2022). *Malplotlib*. Obtenido de Malplotlib: https://matplotlib.org/

IBM. (25 de noviembre de 2022). *¿Qué es la prueba de software?* Obtenido de ¿Qué es la prueba de software?: www.ibm.com

IBM. (15 de noviembre de 2022). *Transformaciones de los datos.* Obtenido de Transformaciones de los datos: https://www.ibm.com/docs/es/spss-statistics/28.0.0?topic=series-data-transformations

IBM. (15 de noviembre de 2022). *Transformaciones de los datos de serie temporal.* Obtenido de Transformaciones de los datos de serie temporal: https://www.ibm.com/docs/es/spss-statistics/25.0.0?topic=transformations-time-series-data

Jaume, U. (2010). *Ingeniería Informáatica. Procesadores de lenguaje. Python: Conceptos básicos y ejercicios.* Castellón: Universitat Jaume I.

Juárez, e. a. (2016). *Análisis de series de tiempo en el pronóstico de la demanda de almacenamiento de productos perecederos.*

Laudon, J., & K., L. (2006). Sistemas de información gerencial. *Administración de la empresa digital.*

Loadview-Testing. (25 de noviembre de 2022). *Tipos de pruebas de software: diferencias y ejemplos*. Obtenido de Tipos de pruebas de software: diferencias y ejemplos: www.loadview-testing.com

López Ramos, D., & Arco García, L. (2019). *Aprendizaje profundo para la extracción de aspectos en opiniones textuales.* Santiago de Cuba: Universidad de Oriente.

M, S. (1993). Neuronal Networks for Statistical Modeling.

NumFOCUS. (8 de Junio de 2022). *Pandas*. Obtenido de Pandas: pandas.pydata.org

Parra, F. (2019). *Estadística y Machine Learning con R.* Bookdown. Obtenido de https://bookdown.org/content/2274/series-temporales.html

Pérez Almeneiro, A. M., & Sánchez Batista, N. I. (2019). Impacto del diplomado dirección y gestión empresarial en la capacitación de cuadros y reservas, en la filial “Rubén Martínez Villena”, universidad de artemisa. *Revista Caribeña de Ciencias Sociales*.

Redacción KeepCoding. (25 de noviembre de 2022). *¿Qué son las pruebas de caja blanca?* Obtenido de ¿Qué son las pruebas de caja blanca?: www.keepcoding.io

Redacción KeepCoding. (25 de noviembre de 2022). *¿Qué son las pruebas de caja negra?* Obtenido de ¿Qué son las pruebas de caja negra?: keepcoding.io

RStudio. (26 de 11 de 2017). *RPubs*. Obtenido de https://rpubs.com/palominoM/series

Ruiz Brückel, T. (2020). *Desarrollo e implementación de modelos de Machine Learning para aplicaciones de gestión y eficiencia energética.* Catalunya.

Saez, et. al. (mar de 1999). MÉTODOS DE SERIES TEMPORALES EN LOS ESTUDIOS EPIDEMIOLÓGICOS SOBRE CONTAMINACIÓN ATMOSFÉRICA. *Scielo, 73*(2).

Salas Rueda, R. A., & Salas Rueda, R. D. (2019). Análisis sobre el uso de la red social Facebook en el proceso de enseñanza-aprendizaje por medio de la ciencia de datos. *Revista de Comunicación de la SEECI*, 50.

SOLVER. (6 de Junio de 2022). *SOLVER Itilligent Analitics*. Obtenido de SOLVER Itilligent Analitics: https://iasolver.es/6-librerias-de-python-para-machine-learning/

Telefónica Tech. (14 de noviembre de 2022). *Python para todos: 5 formas de cargar datos para tus proyectos de Machine Learning.* Obtenido de Python para todos: 5 formas de cargar datos para tus proyectos de Machine Learning: https://empresas.blogthinkbig.com/python-5-formas-de-cargar-datos-csv-proyectos-machine-learning/

TIBCO. (15 de noviembre de 2022). *¿Qué es el análisis de series temporales?* Obtenido de ¿Qué es el análisis de series temporales?: https://www.tibco.com/es/reference-center/what-is-time-series-analysis#:~:text=El%20an%C3%A1lisis%20de%20series%20temporales%20es%20una%20t%C3%A9cnica%20estad%C3%ADstica%20que,en%20intervalos%20de%20tiempo%20particulares.

Trans-Tl. (25 de noviembre de 2022). *¿Qué son las pruebas de integración en el software testing?* Obtenido de ¿Qué son las pruebas de integración en el software testing?: trans-tl.com

Waskom, M. (8 de Junio de 2022). *Seaborn*. Obtenido de Seaborn: https://seaborn.pydata.org/