



TRABAJO DE GRADO
Opción Seminario-Diplomado.

**ALGORITMO COMPUTACIONAL PARA EL ANÁLISIS Y TOMA DE DECISIONES
EN DATOS DE SOLICITUDES DE GESTIÓN DE CAPACIDAD SOFKA
TECHNOLOGIES, UTILIZANDO ESTRATEGIAS DE MACHINE LEARNING**

Corporación Universitaria Remington.

Facultad de Ingeniería

Tecnología en Desarrollo de Software

Estudiantes:

S.A Ynfante Valero

Nevardo A. Ospina Zuñiga

Tutor: Juan Carlos Briñez de León

Opción de Trabajo de grado Seminario-Diplomado.

2024.

Dedicatoria

A Dios, por ser mi guía y fortaleza en este camino de formación académica y profesional.

A mi familia, por su inquebrantable apoyo, comprensión y motivación durante este proceso.

Gracias por creer en mí y por alentarme a alcanzar mis metas académicas.

Finalmente, dedico este trabajo a mí mismo/a, como un recordatorio de que con esfuerzo, dedicación y perseverancia, se pueden alcanzar grandes objetivos. Que este logro sea el punto de partida para seguir creciendo y contribuyendo al conocimiento en mi campo de estudio.

Agradecimientos

A mis compañeros de clase, quienes compartieron conmigo este recorrido académico y con quienes compartí experiencias, desafíos y momentos de aprendizaje, les agradezco por su colaboración, compañerismo y amistad.

A la Corporación Universitaria Remington y su excelente cuerpo docente que compartieron sus conocimientos y experiencias para nuestro crecimiento profesional.

¡Gracias a todos los que hicieron posible este logro!

TABLA DE CONTENIDO

Resumen.....	5
Marco conceptual y contextual	6
Pregunta problema	8
Acercamiento a los datos	8
Descripción de variables	9
Posibles aplicaciones	10
Aproximaciones con gráficos	11
Objetivos.....	17
Objetivo general.....	17
Objetivos específicos	17
Desarrollo e implementación del aprendizaje.....	18
Modelo de toma de decisiones.....	25
Implementación en contextos reales	26
Resultados adicionales	28
Conclusiones.....	30
Referencias.....	31

ÍNDICE DE FIGURAS

Figura 1.	Resumen del DataFrame de Gestión de capacidad	9
Figura 2.	Análisis de los datos de Gestión de capacidad.....	11
Figura 3.	Distribución de Gestiones por País y Centro de Excelencia (CoE)	12
Figura 4.	Gráfico de Distribución de Gestiones por País y Centro de Excelencia (CoE)	12
Figura 5.	Relación entre Tipo de Cliente y Número de Personas Presentadas/Cantidad de Cierres	13
Figura 6.	Gráfico de relación entre Tipo de Cliente y Número de Personas Presentadas/Cantidad de Cierres	13
Figura 7.	Éxito de las Gestiones según el Tipo de Atención.....	14
Figura 8.	Gráfico de éxito de las Gestiones según el Tipo de Atención	14
Figura 9.	Tendencias a lo largo del Tiempo	15
Figura 10.	Gráfico de tendencias a lo largo del Tiempo	15
Figura 11.	Mapa de calor o correlación.....	16
Figura 12.	Gráfico de mapa de calor o correlación	16
Figura 13.	Cargar el dataset al entorno de Colab	19
Figura 14.	DataFrame	20
Figura 15.	Resumen del DataFrame de Gestión de capacidad	21
Figura 16.	Análisis de los datos de Gestión de capacidad.....	22
Figura 17.	Quitar columna indeseada a la dataframe	22
Figura 18.	Corregir errores de formato (por ejemplo, convertir a mayúsculas).....	23
Figura 19.	Analizando variables categóricas de Gestión de capacidad.....	23
Figura 20.	Cambiando rango de valores del dataframe	24

Resumen

En la industria de las tecnologías de la información (TI) está en constante evolución, Sofka Technologies, se ha posicionado como una empresa líder en consultoría y desarrollo de software, brindando soluciones innovadoras y adaptables para satisfacer las demandas del mercado. En este contexto dinámico, el Machine Learning (ML) ha emergido como una herramienta fundamental para analizar grandes volúmenes de datos, tomar decisiones informadas y automatizar procesos.

Específicamente, en el ámbito de la gestión de solicitudes de capacidad en Sofka Technologies, el ML representa una oportunidad para optimizar la asignación de recursos técnicos y humanos a proyectos, mejorar los flujos de trabajo y aumentar la eficiencia operativa. Esto es crucial para el éxito de los proyectos y la satisfacción del cliente.

El presente trabajo aborda la pregunta de cómo un algoritmo de Machine Learning puede optimizar la gestión de solicitudes de capacidad en Sofka Technologies, con el objetivo de mejorar la asignación de recursos y la eficiencia operativa.

Los datos utilizados provienen de registros internos de solicitudes de posiciones laborales que Sofka Technologies ha gestionado, recopilados por sus departamentos de Recursos Humanos. Estas solicitudes incluyen información como país, tipo de cliente, centro de excelencia, número de posiciones, fechas, tipo de atención, estado, año y calificación.

El análisis propuesto implica caracterizar y procesar estos datos, implementar algoritmos de Machine Learning, evaluar su desempeño y validar la toma de decisiones con nuevos datos. Las posibles aplicaciones incluyen la previsión de demanda por área y el análisis del mercado laboral por país, lo cual puede respaldar decisiones estratégicas de expansión y enfoque geográfico.

Marco conceptual y contextual

La industria de las tecnologías de la información (TI) está en constante crecimiento y se caracteriza por una rápida adaptación a las nuevas tecnologías y métodos de desarrollo de software. En este entorno dinámico, Sofka Technologies se ha consolidado como una empresa de consultoría y desarrollo de software especializada en brindar soluciones innovadoras a problemas complejos de TI. La adaptabilidad y la capacidad de gestionar activos y gestionar proyectos son esenciales para seguir siendo competitivos y satisfacer las demandas del mercado. Sofka. (2023, 6 octubre).

Por otra parte, el Machine Learning (ML) ha emergido como una herramienta clave en la industria de TI, permitiendo a las empresas analizar grandes volúmenes de datos para tomar decisiones informadas y automatizar procesos. En el contexto de la gestión de solicitudes de capacidad en Sofka Technologies, el ML ofrece una oportunidad para mejorar la asignación de recursos, optimizar los flujos de trabajo y aumentar la eficiencia operativa.

Las solicitudes de capacidad en una empresa de TI como Sofka Technologies implican la asignación de recursos técnicos y humanos para proyectos específicos. Un análisis efectivo de estas solicitudes mediante algoritmos de ML puede ayudar a identificar patrones, predecir necesidades futuras y asignar recursos de manera óptima, lo cual es crucial para el éxito de los proyectos y la satisfacción del cliente.

Abarzúa, E., Contreras, F., & Robles, J. F. (2002). Señala que La gestión de personas en las empresas ha experimentado una notable evolución a lo largo del tiempo. Inicialmente, se centraba en tareas operativas y reactivas, como la administración de personal y el cumplimiento de

normativas laborales. Posteriormente, se orientó hacia un enfoque más estratégico y proactivo, alineado con los objetivos organizacionales y considerando al capital humano como una ventaja competitiva clave.

El área se ha replanteado, abordando temas como la cultura organizacional, el aprendizaje continuo, la concertación de actores y el desarrollo de competencias esenciales. Las tendencias actuales apuntan hacia una gestión de personas que potencie las capacidades de la empresa para enfrentar los desafíos de un entorno cambiante, mediante la atracción, retención y desarrollo del talento, así como la promoción de un clima laboral favorable y la generación de compromiso de los colaboradores. (Abarzúa, Contreras, & Robles, 2002, p. 12)

Crespo Márquez, A. (2022). Señala que en la era de la Industria 4.0, la gestión inteligente del mantenimiento de activos se ha convertido en un gran desafío y objetivo principal para las empresas en su proceso de transformación digital. Esta nueva gestión implica mejorar el rendimiento de los planes de mantenimiento, incorporar el mantenimiento basado en condición y un modelo de datos que soporte técnicas de analítica predictiva y minería de datos.

El estudio presenta una metodología para extraer información de bases de datos de activos y revisar técnicas de Machine Learning (ML) aplicadas en Condition-Based Maintenance (CBM). Se selecciona la técnica más apropiada y se integra en la toma de decisiones mediante reglas de interpretación. Además, se ilustran casos de uso en activos críticos y flotas, generando recomendaciones para mejorar la eficiencia energética y optimizar los planes de CBM existentes. Crespo Márquez, A. (2022).

La información obtenida de la página nos demuestra cómo el análisis de datos, la extracción de conocimiento mediante modelos de ML y su aplicación en la gestión del mantenimiento a través de recomendaciones son fundamentales para impulsar la transformación digital hacia la Industria 4.0 y el diseño de soluciones avanzadas basadas en machine learning para la toma de decisiones en la gestión de activos.

Pregunta problema

¿Cómo un algoritmo de Machine Learning puede optimizar la gestión de solicitudes de capacidad en Sofka Technologies para mejorar la asignación de recursos y eficiencia operativa?

Acercamiento a los datos

Los datos contenidos en el dataset "Gestión Capacidad-Seminario.csv" son registros internos de la empresa Sofka Technologies. Sofka. (2023, 6 octubre). La página oficial nos dice que es una compañía de tecnología que brinda experiencias de servicio para la transformación tecnológica, co-creando soluciones de alto impacto al servicio de las personas y su entorno. Este conjunto de datos se refiere a un registro de solicitudes de posiciones laborales que Sofka Technologie ha recibido y gestionado para diferentes países, clientes, áreas de negocio y tipos de atención (interna/externa). Recopilados y administrados por sus departamentos de Recursos Humanos.

Nuestro conjunto de datos proviene de un proceso de recopilación, originándose inicialmente en las solicitudes creadas a través de la plataforma Team Tailor. Estas solicitudes fundamentales para nuestra operación, pasan posteriormente a un archivo de Excel distinto al que se presenta para análisis. Este archivo secundario contiene información detallada y sensible, incluyendo datos de costos y nombres de clientes, cuya confidencialidad es primordial. Por tal motivo, se realiza una

selección cuidadosa de los datos necesarios para nuestro análisis, excluyendo cualquier información que pueda comprometer la privacidad o la seguridad de la información del cliente. El Excel dedicado al análisis se actualiza diariamente al cierre de la jornada, incorporando las novedades y garantizando que el análisis se realice con la información más actual y relevante. Este enfoque nos permite mantener un equilibrio entre el rigor analítico y el respeto por la confidencialidad de los datos, asegurando al mismo tiempo que las decisiones y estrategias derivadas de este análisis sean informadas y efectivas.

Descripción de variables

Figura 1. Resumen del DataFrame de Gestión de capacidad

```

▶ # Utilizar el método .info() para obtener un resumen del DataFrame de Gestión de capacidad
  Conjunto_Datos.info()

↳ <class 'pandas.core.frame.DataFrame'>
  RangeIndex: 273 entries, 0 to 272
  Data columns (total 12 columns):
  #   Column                Non-Null Count  Dtype
  ---  ---
  0   Id                     273 non-null   int64
  1   Pais                   273 non-null   object
  2   TipoCliente           273 non-null   object
  3   CoE                    273 non-null   object
  4   # Posiciones          273 non-null   int64
  5   Fecha de solicitud    273 non-null   object
  6   Tipo Atencion         273 non-null   object
  7   Personas Presentadas  273 non-null   int64
  8   STATUS                 273 non-null   object
  9   Q Cierre               273 non-null   int64
  10  Año                    273 non-null   int64
  11  CALIFICADA (S/N/P)    273 non-null   object
  dtypes: int64(5), object(7)
  memory usage: 25.7+ KB

```

Descripción variable a variable de DataFrame de Gestión de capacidad

- Id: Identificador único de cada solicitud. Hay 273 registros únicos.

- País: El país de origen de la solicitud. Hay 12 países únicos, siendo "Ecuador" el más frecuente con 111 menciones.
- TipoCliente: Clasificación del cliente. Hay 3 tipos únicos, Nuevo, On Going y propuesta.
- CoE: Centro de Excelencia asignado a la solicitud. Existen 7 únicos: Agile, Calidad, RPA, Desarrollo, CLOUD, DevOps, AI.
- #Posiciones: Número de posiciones solicitadas. Todas las solicitudes tienen 1 posición indicada, ya que por solicitud si son varias se crea una fila por cada posición.
- Fecha de solicitud: Fecha en que se realizó la solicitud.
- Tipo Atención: Indica si la atención fue interna, externa o la solicitud fue una propuesta.
- Personas Presentadas: Número de personas presentadas para la solicitud, variando de 0 a 5.
- Status: Estado de la solicitud, con 4 estados únicos, Completed, Open, Standby, Canceled.
- Q Cierre: Indicador cuantitativo del cierre.
- Año: Año de la solicitud, oscilando entre 2023 y 2024.
- Calificada (S/N/P): Indica si la solicitud fue calificada, con tres posibles valores ("S", "N", "P"). Sí, NO y Propuesta.

Posibles aplicaciones

1. Previsión de demanda: Utilizando técnicas de análisis de series de tiempo y modelado predictivo, se pueden estimar las futuras demandas de personal por área, lo que permite una mejor planificación de recursos.

2. Análisis de mercado laboral: Al examinar los datos por país, se puede obtener una perspectiva de los mercados laborales más activos para la empresa, lo que puede influir en decisiones de expansión o enfoque geográfico.
3. Análisis de solicitudes: La distribución de las gestiones de capacidad por centro de excelencia (CoE) y la relación entre el tipo de cliente (nuevo o en curso).

Aproximaciones con gráficos

Figura 2. Análisis de los datos de Gestión de capacidad

[44] #Análisis de los datos de Gestión de capacidad
Conjunto_Datos.describe()

	Id	# Posiciones	Personas Presentadas	Q Cierre	Año
count	273.000000	273.0	273.000000	273.000000	273.000000
mean	137.000000	1.0	1.190476	1.637363	2023.468864
std	78.952517	0.0	0.818210	0.945381	0.499946
min	1.000000	1.0	0.000000	1.000000	2023.000000
25%	69.000000	1.0	1.000000	1.000000	2023.000000
50%	137.000000	1.0	1.000000	1.000000	2023.000000
75%	205.000000	1.0	2.000000	2.000000	2024.000000
max	273.000000	1.0	5.000000	4.000000	2024.000000

- Número de Registros: Hay 273 registros en el dataset.
- Número de Posiciones: Todos los registros tienen 1 posición abierta.
- Personas Presentadas: En promedio, 1.19 personas se presentan por posición, con un mínimo de 0 y un máximo de 5.
- Cierres (Q Cierre): El número promedio de cierres es de 1.64, variando entre 1 y 4.
- Año: Los registros están distribuidos entre los años 2023 y 2024.

Figura 3. Distribución de Gestiones por País y Centro de Excelencia (CoE)

```

import matplotlib.pyplot as plt
import seaborn as sns

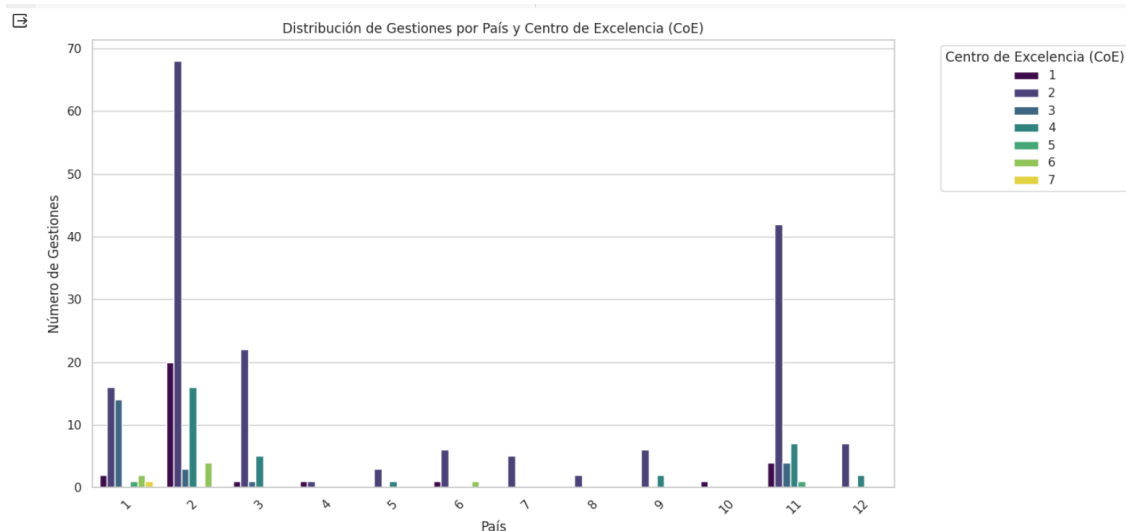
# Ajustes de visualización
sns.set(style="whitegrid")

# Gráfico de barras para mostrar la distribución de gestiones por país y CoE
plt.figure(figsize=(14, 7))
sns.countplot(data=Conjunto_Datos, x='País', hue='CoE', palette='viridis')
plt.title('Distribución de Gestiones por País y Centro de Excelencia (CoE)')
plt.xlabel('País')
plt.ylabel('Número de Gestiones')
plt.xticks(rotation=45)
plt.legend(title='Centro de Excelencia (CoE)', bbox_to_anchor=(1.05, 1), loc='upper left')

# Mostrar el gráfico
plt.tight_layout()
plt.show()

```

Figura 4. Gráfico de Distribución de Gestiones por País y Centro de Excelencia (CoE)



El gráfico muestra la distribución de gestiones por país, con un desglose por Centro de Excelencia (CoE). Se puede observar cómo varía la cantidad de gestiones entre los diferentes países y cómo ciertos CoEs tienen una presencia más fuerte en algunos países que en otros.

Figura 5. Relación entre Tipo de Cliente y Número de Personas Presentadas/Cantidad de Cierres de Cierres

```

# Relación entre Tipo de Cliente y Número de Personas Presentadas/Cantidad de Cierres
# Gráficos de cajas para comparar el número de personas presentadas y cantidad de cierres según el tipo de cliente

plt.figure(figsize=(14, 6))

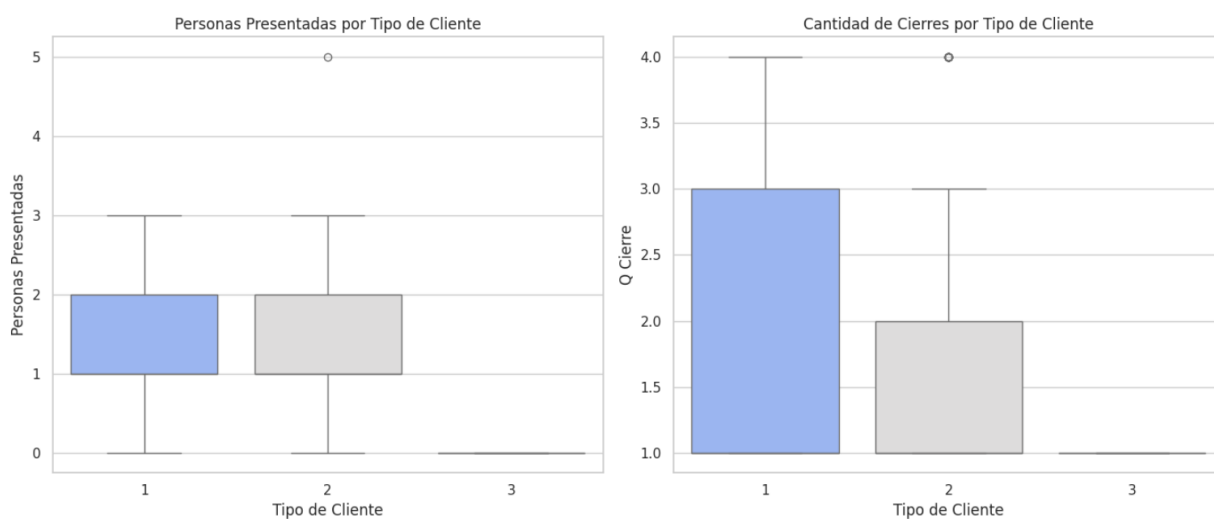
# Número de Personas Presentadas por Tipo de Cliente
plt.subplot(1, 2, 1)
sns.boxplot(data=Conjunto_Datos, x='TipoCliente', y='Personas Presentadas', palette='coolwarm')
plt.title('Personas Presentadas por Tipo de Cliente')
plt.xlabel('Tipo de Cliente')
plt.ylabel('Personas Presentadas')

# Cantidad de Cierres por Tipo de Cliente
plt.subplot(1, 2, 2)
sns.boxplot(data=Conjunto_Datos, x='TipoCliente', y='Q Cierre', palette='coolwarm')
plt.title('Cantidad de Cierres por Tipo de Cliente')
plt.xlabel('Tipo de Cliente')
plt.ylabel('Q Cierre')

plt.tight_layout()
plt.show()

```

Figura 6. Gráfico de relación entre Tipo de Cliente y Número de Personas Presentadas/Cantidad de Cierres



Los gráficos de cajas muestran cómo varían el número de personas presentadas y la cantidad de cierres entre los clientes nuevos y los clientes en curso (On Going). A partir de estos gráficos, se puede observar la distribución y la variabilidad de estas métricas en ambos tipos de clientes, incluyendo los valores medianos, los rangos intercuartílicos, y la presencia de valores atípicos.

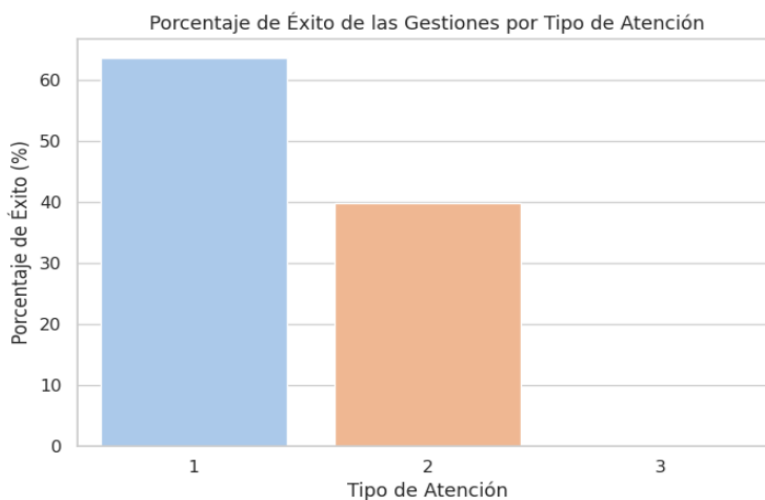
Figura 7. Éxito de las Gestiones según el Tipo de Atención

```
# Crear una columna indicadora de éxito de las gestiones
Conjunto_Datos[Éxito] = Conjunto_Datos[STATUS] == 2

# Gráfico de barras para mostrar la proporción de éxito por tipo de atención
plt.figure(figsize=(7, 5))
sns.barplot(data=Conjunto_Datos, x='Tipo Atencion', y='Éxito', estimator=lambda x: sum(x) / len(x) * 100, palette='pastel', ci=None)
plt.title("Porcentaje de Éxito de las Gestiones por Tipo de Atención")
plt.xlabel("Tipo de Atención")
plt.ylabel("Porcentaje de Éxito (%)")

plt.tight_layout()
plt.show()
```

Figura 8. Gráfico de éxito de las Gestiones según el Tipo de Atención



El gráfico de barras muestra el porcentaje de éxito de las gestiones según el tipo de atención (interna o externa). Este análisis visual proporciona una idea clara de cómo el tipo de atención puede influir en la probabilidad de completar con éxito las gestiones.

Figura 9. Tendencias a lo largo del Tiempo

```

# Convertir la columna 'Fecha de solicitud' a datetime
Conjunto_Datos['Fecha de solicitud'] = pd.to_datetime(Conjunto_Datos['Fecha de solicitud'], dayfirst=True)

# Crear una columna 'Mes-Año' para agrupar los datos por mes y año
Conjunto_Datos['Mes-Año'] = Conjunto_Datos['Fecha de solicitud'].dt.to_period('M')

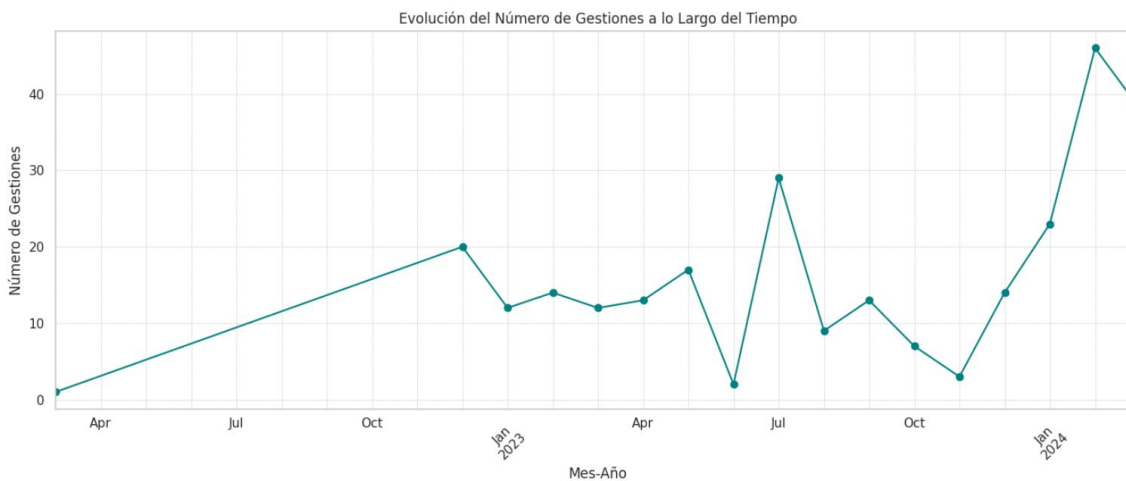
# Agrupar los datos por 'Mes-Año' y contar el número de gestiones
gestiones_por_mes = Conjunto_Datos.groupby('Mes-Año').size()

# Gráfico de líneas para mostrar la evolución del número de gestiones a lo largo del tiempo
plt.figure(figsize=(14, 6))
gestiones_por_mes.plot(kind='line', marker='o', linestyle='-', color='teal')
plt.title("Evolución del Número de Gestiones a lo Largo del Tiempo")
plt.xlabel("Mes-Año")
plt.ylabel("Número de Gestiones")
plt.xticks(rotation=45)
plt.grid(True, which='both', linestyle='--', linewidth=0.5)

plt.tight_layout()
plt.show()

```

Figura 10. Gráfico de tendencias a lo largo del Tiempo



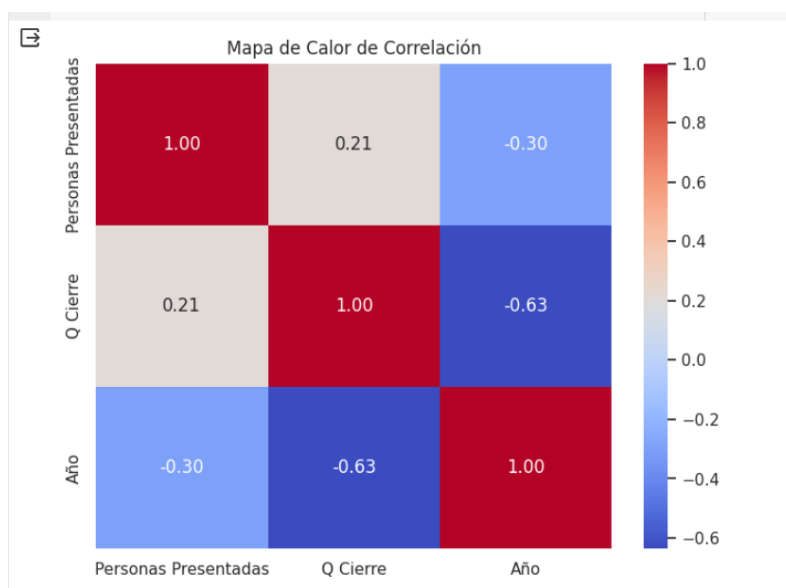
El gráfico de líneas muestra la evolución del número de gestiones a lo largo del tiempo, destacando cualquier variación mensual y las tendencias generales entre los años 2023 y 2024. A través de este análisis, podemos identificar periodos de mayor o menor actividad y cómo estas tendencias podrían correlacionarse con eventos específicos o cambios en la estrategia de gestión de capacidad.

Figura 11. Mapa de calor o correlación

```
# Calcular la matriz de correlación
matriz_correlacion = Conjunto_Datos[['Personas Presentadas', 'Q Cierre', 'Año']].corr()

# Generar el mapa de calor de correlación
plt.figure(figsize=(8, 6))
sns.heatmap(matriz_correlacion, annot=True, cmap='coolwarm', fmt=".2f")
plt.title('Mapa de Calor de Correlación')
plt.show()
```

Figura 12. Gráfico de mapa de calor o correlación



El mapa de calor muestra la matriz de correlación entre el número de personas presentadas, la cantidad de cierres (Q Cierre) y el año. Cada celda proporciona el coeficiente de correlación para un par de variables, donde un valor cercano a 1 indica una correlación positiva fuerte, un valor cercano a -1 indica una correlación negativa fuerte, y un valor cercano a 0 indica poca o ninguna correlación lineal.

Objetivos

Objetivo general

Implementar un algoritmo computacional para el análisis y toma de decisiones a partir de datos de Solicitudes de Gestión de capacidad Sofka Technologies, utilizando estrategias de machine learning.

Objetivos específicos

- Caracterizar y procesar los datos de interés, con miras a la toma de decisiones informadas.
- Implementar un algoritmo de Machine learning para la toma de decisiones a partir de los datos de interés.
- Evaluar y analizar el desempeño de los algoritmos implementados para la toma de decisiones.

Desarrollo e implementación del aprendizaje

En el entorno dinámico y competitivo de la industria TI, donde la capacidad de adaptarse a las demandas del mercado rápidamente cambiantes es crucial, el aprendizaje adquirido en el curso de Machine Learning (ML) se suma como una herramienta transformadora para afrontar estos desafíos. Este conocimiento no solo amplía la comprensión teórica de la ciencia de datos y los algoritmos inteligentes, sino que su aplicación práctica se convierte en la piedra angular de la optimización de procesos y la toma de decisiones basadas en datos en tiempo real. Para Sofka Technologies, el aprendizaje de ML significa una capacidad ampliada para analizar de manera efectiva solicitudes de gestión de grandes volúmenes, mejorando significativamente la asignación de recursos técnicos y humanos.

Las estrategias de ML permiten entonces anticipar, analizar y optimizar la asignación de equipos, ayudando así a fortalecer la capacidad de respuesta a las necesidades del cliente, asegurando una ejecución más eficiente y efectiva del proyecto. Aplicar este conocimiento no solo fortalece la competitividad en el mercado, sino que también marca un hito en la evolución hacia una empresa basada en datos, donde las decisiones se basan en análisis profundos y predictivos, abriendo nuevas fronteras en la gestión de proyectos y la satisfacción del cliente.

Procesamiento de los datos

Para el procesamiento de los datos debemos cargar el dataset al entorno de Colab (Google Colaboratory), para convertirlo en un DataFrame de panda.

Figura 13. Cargar el dataset al entorno de Colab

```

# Importar las bibliotecas necesarias:
# pandas se usa para el manejo y análisis de estructuras de datos.
# files de google.colab permite la interacción con el sistema de archivos de Colab.
import pandas as pd
from google.colab import files

# Subir archivos desde el sistema local.
# Al ejecutar esta línea, se abrirá una interfaz de usuario para seleccionar y subir archivos desde tu computadora.
uploaded = files.upload()

# Iterar sobre los archivos subidos. En este caso, aunque se suba más de uno, se procesará el primero.
# Esto es útil si estás esperando que los usuarios suban un único archivo o si solo te interesa el primero que se suba.
for filename in uploaded.keys():
    # Cargar el archivo CSV en un DataFrame de pandas.
    # Se asume que el archivo está delimitado por ';', común en regiones donde la coma se usa como separador decimal.
    Conjunto_Datos = pd.read_csv(filename, sep=';')

# Mostrar las primeras filas del DataFrame para obtener una vista preliminar de los datos.
# Esto es útil para verificar rápidamente que los datos se han cargado correctamente y entender su estructura básica.
Conjunto_Datos.head()

```

Elegir archivos: Ninguno archivo selec. Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.

Saving Gestión Capacidad-SeminarioV2.csv to Gestión Capacidad-SeminarioV2.csv

	Id	País	TipoCliente	CoE	# Posiciones	Fecha de solicitud	Tipo Atención	Personas Presentadas	STATUS	Q Cierre	Año	CALIFICADA (S/N/P)
0	1	USA	Nuevo	AGILE	1	27/03/2022	INTERNA	1	CANCELED	2	2023	S
1	2	USA	Nuevo	DESARROLLO	1	1/12/2022	EXTERNA	2	CANCELED	1	2023	S
2	3	Ecuador	On Going	DESARROLLO	1	5/12/2022	EXTERNA	3	COMPLETED	1	2023	S

En la preparación inicial de los datos para su análisis, seguimos una secuencia de pasos clave, ilustrados en la figura mencionada. El proceso comienza con la importación de las bibliotecas esenciales: utilizamos pandas, un poderoso paquete para el manejo y análisis de datos en Python, y files de google.colab, que nos permite interactuar con el sistema de archivos de Colab, facilitando así la carga de archivos directamente desde nuestra computadora local.

El siguiente paso implica ejecutar `files.upload()`, una instrucción que activa una interfaz de usuario para seleccionar y subir archivos desde nuestra computadora. Una vez subidos los archivos, utilizamos la instrucción `for filename in uploaded.keys():` para iterar a través de ellos. A pesar de que es posible subir múltiples archivos, este método procesa únicamente el primero en la lista.

Para cargar los datos en nuestro entorno de trabajo, empleamos `Conjunto_Datos = pd.read_csv(filename, sep=';')`, donde `pd.read_csv()` lee el archivo CSV seleccionado y lo convierte

en un DataFrame de pandas. Esta estructura de datos facilita la manipulación y análisis posterior gracias a su flexibilidad y eficiencia.

Finalmente, `Conjunto_Datos.head()` nos proporciona una vista preliminar de los primeros registros del DataFrame. Este paso es crucial para verificar la correcta carga de los datos y obtener una comprensión inicial de su estructura, lo que nos permite asegurar que estamos en el camino correcto para el análisis detallado que sigue.

Figura 14. DataFrame

Elegir archivos Ninguno archivo selec. Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable.
Saving Gestión Capacidad-SeminarioV2.csv to Gestión Capacidad-SeminarioV2.csv

	Id	Pais	TipoCliente	CoE	# Posiciones	Fecha de solicitud	Tipo Atencion	Personas Presentadas	STATUS	Q Cierre	Año	CALIFICADA (S/N/P)
0	1	USA	Nuevo	AGILE	1	27/03/2022	INTERNA	1	CANCELED	2	2023	S
1	2	USA	Nuevo	DESARROLLO	1	1/12/2022	EXTERNA	2	CANCELED	1	2023	S
2	3	Ecuador	On Going	DESARROLLO	1	5/12/2022	EXTERNA	3	COMPLETED	1	2023	S
3	4	Ecuador	On Going	DESARROLLO	1	5/12/2022	EXTERNA	2	CANCELED	3	2023	S
4	5	Ecuador	On Going	DESARROLLO	1	5/12/2022	INTERNA	2	CANCELED	3	2023	S

La DataFrame contiene información sobre solicitudes o posiciones, incluyendo el país, el tipo de cliente, el CoE (Centro de Excelencia o área funcional), el número de posiciones, la fecha de solicitud, el tipo de atención (interna, externa o Propuesta), el número de personas presentadas, el estado (cancelado o completado), el trimestre y año, y una calificación (S / N / P).

Figura 15. Resumen del DataFrame de Gestión de capacidad

```

Conjunto_Datos.info()

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 273 entries, 0 to 272
Data columns (total 12 columns):
#   Column                Non-Null Count  Dtype
---  ---
0   Id                     273 non-null   int64
1   Pais                   273 non-null   object
2   TipoCliente            273 non-null   object
3   CoE                     273 non-null   object
4   # Posiciones           273 non-null   int64
5   Fecha de solicitud     273 non-null   object
6   Tipo Atencion          273 non-null   object
7   Personas Presentadas  273 non-null   int64
8   STATUS                 273 non-null   object
9   Q Cierre               273 non-null   int64
10  Año                    273 non-null   int64
11  CALIFICADA (S/N/P)    273 non-null   object
dtypes: int64(5), object(7)
memory usage: 25.7+ KB

```

Para la figura anterior nos proporciona una visión general de la estructura y contenido del DataFrame, lo cual es útil para comprender los datos antes de realizar cualquier análisis o procesamiento adicional, donde nos muestra la salida de la información general del DataFrame sobre la gestión de capacidad de la empresa. A continuación se muestra con más detalles:

Registro: este dataset contiene un total de registro de 273.

VARIABLES: Id, País, TipoCliente, CoE, # Posiciones, Fecha de solicitud, Tipo Atención, Personas Presentadas, STATUS, Q Cierre, Año y CALIFICADA (S/N/P).

Los tipos de datos que tenemos en nuestro dataFrame son:

int64 (Números enteros): Id, # Posiciones, Personas Presentadas, Q Cierre, Año:

object (Cadenas de texto): País, TipoCliente, CoE, Fecha de solicitud, Tipo Atención, STATUS, CALIFICADA (S/N/P)

Figura 16. Análisis de los datos de Gestión de capacidad

```
[ ] #Análisis de los datos de Gestión de capacidad
Conjunto_Datos.describe()
```

	Id	# Posiciones	Personas Presentadas	Q Cierre	Año
count	273.000000	273.0	273.000000	273.000000	273.000000
mean	137.000000	1.0	1.190476	1.637363	2023.468864
std	78.952517	0.0	0.818210	0.945381	0.499946
min	1.000000	1.0	0.000000	1.000000	2023.000000
25%	69.000000	1.0	1.000000	1.000000	2023.000000
50%	137.000000	1.0	1.000000	1.000000	2023.000000
75%	205.000000	1.0	2.000000	2.000000	2024.000000
max	273.000000	1.0	5.000000	4.000000	2024.000000

Teniendo en cuenta la figura anterior tenemos el siguiente análisis proporciona información valiosa sobre la distribución de las diferentes variables en el conjunto de datos de Gestión de capacidad. Puede ser útil para comprender patrones, identificar posibles problemas o preparar los datos para un análisis más profundo.

Si realizamos el análisis por año y personas presentadas, podemos observar que:

Las solicitudes abarcan los años 2023 y 2024, además que el 75% de las solicitudes son del año 2024, mientras que el 25% restante son del año 2023.

Personas presentadas:

El número de personas presentadas varía de 0 a 5, con una media de 1.19 personas y la mayoría de las solicitudes (50%) tienen solo 1 persona presentada.

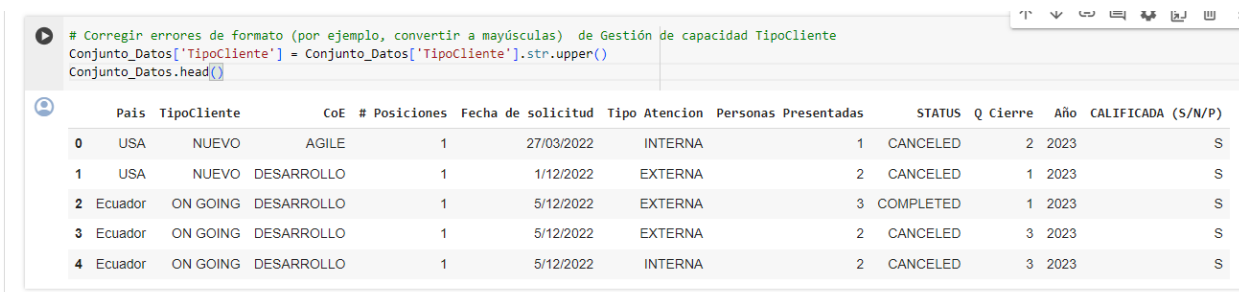
Figura 17. Quitar columna indeseada a la dataframe

```
[ ] #Quitando columnas indeseadas
Conjunto_Datos=Conjunto_Datos.drop(['Id'],axis=1)
#resumen de los datos
Conjunto_Datos.head()
```

	Pais	TipoCliente	CoE	# Posiciones	Fecha de solicitud	Tipo Atencion	Personas Presentadas	STATUS	Q Cierre	Año	CALIFICADA (S/N/P)
0	USA	Nuevo	AGILE	1	27/03/2022	INTERNA	1	CANCELED	2	2023	S
1	USA	Nuevo	DESARROLLO	1	1/12/2022	EXTERNA	2	CANCELED	1	2023	S
2	Ecuador	On Going	DESARROLLO	1	5/12/2022	EXTERNA	3	COMPLETED	1	2023	S
3	Ecuador	On Going	DESARROLLO	1	5/12/2022	EXTERNA	2	CANCELED	3	2023	S
4	Ecuador	On Going	DESARROLLO	1	5/12/2022	INTERNA	2	CANCELED	3	2023	S

La columna 'Id' se elimina del DataFrame porque para nosotros no aporta mucho valor analítico para el procesamiento de los datos, y al no proporcionar información sustancial sobre los atributos de los datos que tenemos en el dataframe. Y así simplificamos el conjunto de datos, facilitando el procesamiento y análisis posteriores.

Figura 18. Corregir errores de formato (por ejemplo, convertir a mayúsculas)



```
# Corregir errores de formato (por ejemplo, convertir a mayúsculas) de Gestión de capacidad TipoCliente
Conjunto_Datos['TipoCliente'] = Conjunto_Datos['TipoCliente'].str.upper()
Conjunto_Datos.head()
```

	País	TipoCliente	CoE	# Posiciones	Fecha de solicitud	Tipo Atencion	Personas Presentadas	STATUS	Q Cierre	Año	CALIFICADA (S/N/P)
0	USA	NUEVO	AGILE	1	27/03/2022	INTERNA	1	CANCELED	2	2023	S
1	USA	NUEVO	DESARROLLO	1	1/12/2022	EXTERNA	2	CANCELED	1	2023	S
2	Ecuador	ON GOING	DESARROLLO	1	5/12/2022	EXTERNA	3	COMPLETED	1	2023	S
3	Ecuador	ON GOING	DESARROLLO	1	5/12/2022	EXTERNA	2	CANCELED	3	2023	S
4	Ecuador	ON GOING	DESARROLLO	1	5/12/2022	INTERNA	2	CANCELED	3	2023	S

Para este paso realizamos una corrección de formato, convirtiendo a mayúsculas la columna "TipoCliente" en el conjunto de datos. Este proceso se realiza con el fin de asegurar la consistencia y estandarización de los datos. Al convertir la columna "TipoCliente" a mayúsculas, se eliminan inconsistencias de formato, como combinaciones de mayúsculas y minúsculas, lo que facilita el análisis y procesamiento posterior de los datos.

Esta estandarización de los datos también se realizó para: País, CoE, Tipo Atención, Status y Calificación

Figura 19. Analizando variables categóricas de Gestión de capacidad.

```
[ ] #Analizando variables categóricas de Gestión de capacidad TipoCliente
Conjunto_Datos['TipoCliente'].unique()

array(['Nuevo', 'On Going', 'Propuesta'], dtype=object)
```

Analizar las variables categóricas se realiza con el fin de comprender mejor la distribución y características de los datos categóricos en el conjunto de datos. Las variables categóricas son aquellas que toman valores discretos que representan diferentes categorías o grupos.

En este caso específico, el código está creando un arreglo de los valores únicos presentes en la columna "TipoCliente" utilizando la función `unique()`. Los valores únicos encontrados son "Nuevo", "On Going" y "Propuestas". Y así conocer las diferentes categorías presentes y su distribución proporciona una visión general de los datos, por lo que es un paso esencial en el proceso de comprensión, exploración y preparación de los datos para su posterior análisis o modelado.

Para este análisis también se realizaron a las variables: País, TipoCliente, Coe, Status, Calificada(S/N/P)

Figura 20. Cambiando rango de valores del dataframe



```
# Cambiando rango de valores del dataframe País
Estrategia = {'USA':1,'ECUADOR':2, 'PANAMÁ':3, 'URUGUAY':4, 'CANADÁ':5, 'MÉXICO':6, 'GUATEMALA':7, 'CHILE':8, 'BRASIL':9, 'UK':10, 'COLOMBIA':11, 'PERÚ':12}
Conjunto_Datos['País']=Conjunto_Datos['País'].map(Estrategia)
Conjunto_Datos.head()
```

	País	TipoCliente	Coe	# Posiciones	Fecha de solicitud	Tipo Atencion	Personas Presentadas	STATUS	Q Cierre	Año	CALIFICADA (S/N/P)
0	1	1	AGILE	1	27/03/2022	INTERNA	1	CANCELED	2	2023	S
1	1	1	DESARROLLO	1	1/12/2022	EXTERNA	2	CANCELED	1	2023	S
2	2	2	DESARROLLO	1	5/12/2022	EXTERNA	3	COMPLETED	1	2023	S
3	2	2	DESARROLLO	1	5/12/2022	EXTERNA	2	CANCELED	3	2023	S
4	2	2	DESARROLLO	1	5/12/2022	INTERNA	2	CANCELED	3	2023	S

El motivo de este proceso es reemplazar los valores de los países por número, por ejemplo USA = 1, Ecuador = 2, el objetivo principal de este proceso es asignar una codificación numérica a los diferentes países que se encuentran en el DataFrame. La finalidad de este proceso es poder trabajar con algoritmos de aprendizaje automático que requieren que las variables categóricas sean codificadas numéricamente. De la misma manera se realizan los cambios para los valores de las variables, TipoCliente, Coe, Status, Calificada(S/N/P)

Modelo de toma de decisiones

Definición del Problema

El primer paso en nuestro modelo de toma de decisiones implica identificar y definir claramente el problema que se desea resolver. En este caso, el problema radica en optimizar la gestión de solicitudes de capacidad en Sofka Technologies, lo que incluye la asignación eficiente de recursos técnicos y humanos a proyectos, la mejora de los flujos de trabajo, y el incremento de la eficiencia operativa.

Recopilación y Preparación de Datos

La base es un conjunto de datos robusto y bien preparado. Los datos utilizados provienen de registros internos de solicitudes de posiciones laborales gestionadas por el área de Gestión de Capacidad de Sofka Technologies. Este paso incluye la limpieza, normalización y transformación de los datos para garantizar que el modelo pueda procesarlos eficientemente.

Selección de Características

Este proceso implica identificar las variables más relevantes que influyen en la toma de decisiones. Por ejemplo, país, tipo de cliente, centro de excelencia, número de posiciones, fechas, tipo de atención, estado, año, y calificación pueden ser características vitales para predecir la demanda de recursos y mejorar la asignación de proyectos.

Desarrollo del Modelo de Machine Learning

Con los datos preparados y las características seleccionadas, el siguiente paso es elegir y entrenar un modelo de ML adecuado para el problema. Se pueden explorar diferentes algoritmos y modelos como por ejemplo regresión Lineal y Logística, Árboles de Decisión y Bosques Aleatorios (Random Forests), Máquinas de Soporte Vectorial (SVM), Redes Neuronales y Deep Learning, Modelos de Series Temporales (ARIMA, LSTM), entre otros.

Implementación en contextos reales

Previsión de Demanda con Series Temporales

Contexto de Implementación:

La gestión eficaz de los recursos humanos en proyectos tecnológicos requiere anticipar las necesidades futuras de personal. Utilizando modelos de series temporales y técnicas de modelado predictivo, Sofka Technologies puede prever la demanda de trabajadores especializados en diferentes áreas, permitiendo una asignación y planificación de recursos más precisa y eficiente.

Proceso de Implementación:

- **Recolección de Datos:** Recopilación de datos históricos sobre solicitudes de personal, proyectos realizados, y asignaciones de recursos por área.

- Selección de Modelos: Implementación de modelos de series temporales como ARIMA o LSTM para capturar tendencias, ciclicidad, y factores estacionales en la demanda de personal.
- Validación y Ajuste: Evaluación de los modelos mediante técnicas como la validación cruzada temporal, ajustando los parámetros para mejorar la precisión de las predicciones.
- Integración y Acción: Integración de las predicciones en los sistemas de planificación de recursos de la empresa para guiar la contratación y la asignación de personal a proyectos.

Impacto Real: La capacidad de predecir con precisión la demanda de personal permite a Sofka Technologies optimizar su fuerza laboral, reducir costos operativos y mejorar la satisfacción del cliente al asegurar la entrega oportuna de proyectos.

Análisis de Mercado Laboral con Aprendizaje Automático

Contexto de Implementación:

Comprender los mercados laborales donde opera Sofka Technologies es vital para tomar decisiones estratégicas informadas sobre expansión y enfoque geográfico. Mediante el análisis de datos de solicitudes de personal por país, la empresa puede identificar tendencias de mercado, áreas de crecimiento, y potenciales oportunidades de expansión.

Proceso de Implementación:

- Recolección y Preparación de Datos: Agregación de datos sobre solicitudes de personal, incluyendo ubicación geográfica, tipo de proyecto, y resultados de estas solicitudes.

- **Análisis con ML:** Uso de técnicas de clustering, como K-Means, para segmentar los países según la demanda de personal y tipos de proyectos. Implementación de modelos predictivos para identificar factores que influyen en el éxito de las solicitudes en diferentes mercados.
- **Visualización de Datos:** Creación de dashboards interactivos que muestran los resultados del análisis, facilitando la identificación de patrones y tendencias.
- **Decisiones Estratégicas:** Utilización de insights generados para guiar decisiones sobre dónde concentrar esfuerzos de expansión o inversión, basándose en el análisis del mercado laboral.

Impacto Real: Este enfoque permite a Sofka Technologies adaptarse dinámicamente a las condiciones del mercado, identificar mercados emergentes con alta demanda de servicios tecnológicos, y enfocar sus estrategias de expansión donde tenga mayores oportunidades de crecimiento y éxito.

Resultados adicionales

- **Previsión de demanda y análisis de mercado laboral:** El proyecto logró aplicar el aprendizaje de ML para prever la demanda de personal por área, permitiendo una planificación de recursos más precisa. Además, se realizó un análisis detallado del mercado laboral por país, lo cual puede influir en decisiones estratégicas de expansión y enfoque geográfico.
- **Optimización de la gestión de solicitudes de capacidad:** A través del uso de algoritmos de ML, el proyecto mejoró significativamente la gestión de solicitudes de capacidad,

optimizando la asignación de recursos técnicos y humanos a proyectos, y mejorando los flujos de trabajo y la eficiencia operativa.

- Implementación de modelos de ML en contextos reales: El proyecto demostró cómo la implementación de modelos de ML en contextos reales, como la previsión de demanda con series temporales y el análisis de mercado laboral, puede tener un impacto positivo en la gestión de recursos humanos y la toma de decisiones estratégicas.

Conclusiones

La implementación del proyecto mostró cómo el aprendizaje automático puede convertirse en una herramienta transformadora en la industria de TI, especialmente en la gestión de solicitudes de capacidad. La capacidad de analizar grandes volúmenes de datos y tomar decisiones basadas en estos análisis es esencial para la adaptabilidad y eficiencia del campo tecnológico.

El proyecto enfatizó la importancia de las estrategias de aprendizaje automático para mejorar la toma de decisiones técnicas y de recursos humanos. La introducción de algoritmos informáticos permitió asignar recursos de manera más eficiente que antes, lo que optimiza los flujos de trabajo y aumenta la satisfacción del cliente.

La aplicación de los conocimientos teóricos adquiridos durante un seminario o diploma en un contexto real y dinámico enriqueció significativamente el aprendizaje. Las experiencias prácticas en el desarrollo y aplicación de un algoritmo informático en el análisis de datos de control específicos mostraron la viabilidad y el efecto positivo de la integración de la teoría y la práctica en el campo del aprendizaje automático: proceso de caracterización, procesamiento y análisis. El análisis de datos destacó el. Desafíos y oportunidades relacionados con la gestión de datos de proyectos de aprendizaje automático. La necesidad de limpiar, normalizar y transformar datos subraya la importancia de una preparación adecuada de los datos para maximizar la eficacia de los algoritmos de aprendizaje automático.

Referencias

Sofka. (2023, 6 octubre). *Conoce a Sofka Technologies | Nosotros*.

<https://www.sofka.com.co/es/nosotros/>

Abarzúa, E., Contreras, F., & Robles, J. F. (2002). Evolución de la gestión de personas en las empresas: del Departamento de Personal a la Gerencia de las Capacidades Organizacionales.

Psykhe, 11(2). <https://pensamientoeducativo.uc.cl/index.php/psykhe/article/view/20121>

Crespo Márquez, A. (2022). Diseño de soluciones avanzadas basadas en técnicas de machine learning para la toma de decisiones en gestión de activos.

<https://idus.us.es/handle/11441/135570>

Sánchez, C. (29 de enero de 2020). *Figuras*. Normas APA (7ma edición). <https://normas-apa.org/estructura/figuras/>

Sánchez, C. (08 de febrero de 2019). Cita de parafraseo. Normas APA (7ma edición).

<https://normas-apa.org/citas/cita-de-parafraseo/>