ISSN 2415-2323

La Salle Universidad Bolivia

REVISTA DE INVESTIGACIÓN ESTUDIANTIL

ILUMINATE

INGENIERÍA DE SISTEMAS

VOLUMEN 12



2020

La Salle Universidad Bolivia

REVISTA DE INVESTIGACIÓN ESTUDIANTIL

ILUMINATE

INGENIERÍA DE SISTEMAS

VOLÚMEN 12

2020

ILUMINATE VOL. 12

DIRECCIÓN
Lic. Wilma Peñafiel

EDITOR RESPONSABLE:
Lic. Wilma Peñafiel (wpenafile@ulasalle.edu.bo)

Comité Editorial (ULASALLE):
Lic. Martha Heredia, Lic. Javier Heredia, Ing. José Laruta

Comité Editorial Externo
M.Sc. Marceo Saavedra, Escuela Militar de Ingeniería

Comité Internacional
Ing. José Gallardo, Universidad Católica del Norte, Antofagasta-Chile

Diseño y Diagramación:
Ing. Roberto Mamani

Impresión: Universidad La Salle Bolivia

Depósito Legal: 4-3-28-09
ISSN 2415-2323

Dirección: Calle Jorge Carrasco, esquina Las Palmas, No 450
Teléfonos: 2723588, 2723598
e-mail: wpenafiel@ulasalle.edu.bo

Noviembre de 2020
La Paz- Bolivia

Visibilidad
Indexada en Revistas Bolivianas

INDICE

EDITORIAL 9
Jose Eduardo Laruta Espejo

ARTÍCULOS ORIGINALES

Visualización y análisis de los jugadores del fifa 2021 aplicando algoritmos de machine learning 13
Albert Omar Araya Cory
José Antonio Aguilar Fernández

Exploración de datos y sistema de recomendación de libros 35
Lozada Gómez José Adolfo
Arana Sánches Esmeralda Yennifer

Análisis de riesgo para préstamos bancarios 47
Juan Salvador Vilca Paredes

Predicción de la clasificación final en la liga española de fútbol 61
Hugo Herrera Lunario,
Ramiro Paucara Ochoa

Predicción de supervivencia ante la insuficiencia cardiaca 77
Choque Forra Daniela Patricia, Mamani Mamani Jessyca Liset

POLÍTICA EDITORIAL E INSTRUCTIVO PARA LOS AUTORES 103

PRESENTACIÓN

El Volumen 12, de la Revista Iluminate contiene seis artículos: de los cuales, el artículo editorial a cargo de un docente, cinco artículos son originales.

En estos tiempos tan difíciles, en medio de una pandemia, con muchas limitaciones, se ha podido presentar este volumen con el esfuerzo, dedicación de los estudiantes que se han atrevido a publicar. También se debe ponderar el apoyo incondicional de docentes de la carrera de Ing. de Sistemas, que han coadyuvado a la realización de presentación de trabajos y su correspondiente publicación.

El artículo, " Exploración de datos y sistema de recomendación de libros", analiza las propiedades de cada libro (libros más leídos, autores más consultados y calificación promedio de un autor o libro) y las relaciones que pueden tener dentro de un conjunto de datos. En base al análisis añaden un sistema de recomendación para la utilización de los usuarios.

El artículo "Análisis de riesgo para préstamos bancarios", analiza los sistemas financieros e identifica si es prudente o no realizar un préstamo a los clientes según el monto del préstamo que solicitan y los datos que se tienen de los mismos.

El artículo "Predicción de la clasificación final en la liga española de fútbol", propone el uso de herramientas de machine learning, en particular del algoritmo de regresión logística para intentar predecir la clasificación final del siguiente campeonato de la Liga española de fútbol.

El artículo "Predicción de supervivencia ante la Insuficiencia Cardiaca", predice la supervivencia ante la insuficiencia cardiaca, contando con un dataset que contiene registros clínicos de pacientes con insuficiencia cardiaca, mediante la aplicación de diferentes algoritmos de Aprendizaje Automático de los cuales se seleccionó el mejor modelo para luego utilizarlo en una interfaz gráfica.

El artículo "Visualización y Análisis de los Jugadores del FIFA 2021 Aplicando Algoritmos de Machine Learning", realiza un análisis comparativo sobre las

características de los jugadores de futbol del FIFA entre los años 2020 – 2021 y la vida real. Para el desarrollo de esta investigación se utilizó dos Dataset del videojuego.

Muchas felicidades a todos los participantes que hicieron posible esta publicación, así como a los docentes, investigadores y comité editorial nacional e internacional, que apoyaron en las aventuras de nuestros estudiantes ayudando a aterrizar las ideas planteadas.

M.Sc. Wilma Peñafiel
Comité Editorial

Artículo Editorial

El impacto del internet en la formación de profesionales en tecnología de talla mundial

Tradicionalmente, el desarrollo de los últimos avances en la tecnología ha estado bastante alejado de nuestra realidad en el pasado, sin embargo, esta situación ha cambiado drásticamente en la actualidad. Debido al contexto geográfico, político y cultural, y a pesar de rescatables iniciativas gubernamentales en la última década, la cantidad de recursos económicos y humanos necesarios hizo casi imposible explorar aspectos tecnológicos de vanguardia de manera local y sin la necesidad de millonarias inversiones en infraestructura aparte de una red de conexión a internet.

Sin embargo, la reciente generalización del uso de internet ha permitido romper muchas barreras de acceso a la información nunca antes pensadas. Ya no es necesario viajar para poder conocer avances recientes en la tecnología o incluso contactar con aquellas personas y equipos responsables de tales avances. Universidades e instituciones que van a la vanguardia han liberado su material y lo han dejado disponible para que todo el que esté interesado pueda acceder a el y estudiarlo.

Esta apertura engloba bastantes áreas del conocimiento; tutoriales, manuales, libros y documentales sobre casi cualquier tema pueden ser encontrados y en muchos casos de manera gratuita. Esto se acentúa en el área de desarrollo de software y TICs debido a que los únicos requisitos para poder desarrollarse profesionalmente en estas áreas son una computadora y conexión a internet. Además, la facilidad de acceso se ha incrementado con la constante reducción del costo de aparatos electrónicos y capacidad de cómputo disponible a los consumidores.

En este sentido, uno de los fenómenos más interesantes en esta llamada "democratización del acceso a la información" gracias a internet fue la creación de los denominados MOOCs o cursos abiertos masivos en línea. En los cuales se presentaron a profesores, equipos y laboratorios de las universidades más prestigiosas del mundo (Stanford, MIT, Harvard, EPFL) compartiendo sus conocimientos en plataformas abiertas en línea. Tradicionalmente este conocimiento solamente

estaba accesible para matriculados en dichas universidades. Estos cursos masivos tuvieron gran éxito, pues enrolaban a miles e incluso decenas de miles de estudiantes, consumiendo información y formándose en temas de vanguardia desde cualquier parte del mundo. Iniciativas como Coursera, Udacity, EDx son algunas de las pioneras en el mundo de los MOOCs y continúan ofreciendo material abierto y contenido de alta calidad en sus plataformas.

Todos los elementos anteriormente mencionados causaron un incremento exponencial en el interés en el desarrollo de software y TICs. A su vez, se comenzaron a generar iniciativas y emprendimientos que usaban nueva tecnología y equipos distribuidos en varias partes del globo. Este interés no ha dejado de crecer desde finales de la década del 2000 y ha creado tanto nuevos profesionales capacitados como la necesidad de mano de obra calificada en temas tecnológicos. Y un aspecto bastante interesante de este fenómeno es que la demanda de profesionales supera en una gran proporción a la disponibilidad de recursos humanos calificados.

Finalmente, en el ámbito académico, todos los aspectos explorados anteriormente han impactado y siguen impactando de gran manera a la forma en la que las universidades preparan a sus profesionales. Los profesores, ahora ya no son los dueños del conocimiento de un área en específico; su papel ha migrado a ser facilitadores y catalizadores de la exploración de los estudiantes. Un profesor no podrá competir con la disponibilidad inmediata de la información encontrada en internet, pero si deberá ser capaz de filtrarla y preparar la mejor forma de presentarla a sus estudiantes. A su vez, el profesor tendrá el papel fundamental de guiar la exploración atendiendo de forma eficaz las dudas y problemas que los estudiantes puedan enfrentar.

En la presente revista, se podrá apreciar los resultados de ese proceso de exploración y experimentación realizada por los estudiantes en forma de artículos científicos. Estos reportes tienen la característica de incluir temas de vanguardia, previamente considerados demasiado complejos o alejados de la realidad académica boliviana. No obstante, gracias a la curiosidad y dedicación de los autores se ha logrado explorar conceptos sobre los cuales, en la actualidad, se construyen emprendimientos e iniciativas con gran impacto en la sociedad. Muchos de los algoritmos y temas explorados en varios de los artículos de la presente edición se usan actualmente en

empresas de la envergadura de Google o Netflix para el análisis de los datos. Esto es una muestra del potencial que tiene el adecuado uso de los recursos en internet y el valor de la guía del profesor para poder explotar tanto la información disponible como la capacidad de análisis y experimentación de sus estudiantes.

Jose Eduardo Laruta Espejo
Docente Universidad La Salle

BLOQUE
"A"

Artículo Original Pág. 13 - 34 ISSN 2415-2323

ILUMATE VOLUMEN 12

VISUALIZACIÓN Y ANÁLISIS DE LOS JUGADORES DEL FIFA 2021 APLICANDO ALGORITMOS DE MACHINE LEARNING

Visualization and Analysis of FIFA 2021 Teams Applying Machine Learning Algorithms

Albert Omar Araya Cory
albertaraya944@gmail.com
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle- Bolivia

José Antonio Aguilar Fernández
jaaguilarfernandez@outlook.es
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle- Bolivia

Resumen

El objetivo de la presente investigación es el de realizar un análisis comparativo sobre las características de los jugadores de futbol del FIFA entre los años 2020 – 2021 y la vida real. Para el desarrollo de esta investigación se utilizó dos Dataset del videojuego. Se aplicó métodos de visualización de datos con Python y se realizó un análisis para la detección de anomalías mediante algoritmos de aprendizaje no supervisado.

Palabras clave

Pandas, Numpy, Seaborn, Scikit-Learn, Matplotlib, Dataset, Video juego, Clustering, Futbol, Python

Abstract

The objective of this research is to carry out a comparative analysis on the characteristics of FIFA soccer players between the years 2020 - 2021 and real life. For the development of this research, two videogame datasets were used. Data visualization methods were applied with Python and an analysis was carried out for the detection of anomalies using unsupervised learning algorithms.

Key words

Pandas, Numpy, Seaborn, Scikit-Learn, Matplotlib, Dataset, Video Game, Clustering, Soccer, Python

Introducción

La Federación Internacional de Fútbol Asociación (FIFA) es la institución que gobierna a todas las federaciones de futbol del planeta, esta institución cada año brinda un informe acerca del rendimiento de los jugadores registrados, estos Datasets son de carácter público y ayudan al desarrollo del Big Data del futbol. El aprendizaje no supervisado es un método de aprendizaje automático donde un modelo se ajusta a las observaciones.

El siguiente proyecto presenta un dataset que contiene datos relevantes y específicos acerca de los jugadores de futbol registrados en la FIFA hasta el año 2021, también se visualizara el club al que pertenecen, de esta manera estaremos realizando análisis en profundidad y visualización de estos datos, además de aplicar aprendizaje supervisado para la regresión lineal y para la detección de anomalías mediante tareas de Clustering.

Problemática

Los equipos de hoy tienen jugadores excelentes que benefician a su equipo en cada partido que se juega y de esta manera muchas veces algunos de

estos jugadores son infravalorados y algunos otros sobrevalorados. Por esta razón al momento de contratar a un nuevo integrante existe la probabilidad de que el jugador beneficie o no al club al que se está integrando.

Objetivo General

Analizar los Datasets completos, para determinar cuáles son los jugadores infravalorados y sobrevalorados en base a distintas características que se obtendrá a través de un análisis profundo utilizando herramientas de machine Learning.

Los objetivos específicos son: Mediante un análisis exploratorio realizar gráficos de visualización. Detectar anomalías. Analizar y explicar los distintos gráficos obtenidos.

Referentes conceptuales

FIFA ha sido un videojuego al que atrae muchas personas en todo el mundo y ha aparecido en muchas ediciones. Su último lanzamiento será en octubre de este año, es decir, FIFA 2021. Este conjunto de datos contiene las características de los jugadores en el juego.

El conjunto de datos contiene datos de la clasificación de los jugadores, su edad, su nacionalidad, la posición en la que juegan y su potencial de crecimiento en el juego. Es posible que los datos de algunos jugadores y sus clubes no sean muy precisos, ya que la ventana de transferencia aún está abierta y es posible que se realicen cambios en etapas posteriores. (Mishra, 2020).

DATASETS

Para este proyecto se utilizaron tres Dataset en los cuales se trabajó y se creó un dataset único ya normalizado listo para ser analizado y visualizado de una manera correcta aplicando los distintos algoritmos.

Dataset 1

fifa-FUT-Data fue el primer dataset que se utilizó y se normalizó para las funciones necesarias al principio de las pruebas del proyecto. (Kafagy, 2019).

Dataset 2

players_21 fue el segundo dataset que se utilizós para poder realizar una actualización mas reciente respecto a las características de los jugadores. (Leone, 2020).

Dataset 3

AprendizajeAutomatico dataset combinado de los dos anteriores dataset el que fue normalizado e implementado en el proyecto. (AlbertAC, 2020).

Análisis y Visualización de los datos de los jugadores del FIFA 2021

Actualmente cada equipo fútbol tiene a un jugador estrella que según los entrenadores son considerados el mejor, entonces los espectadores comienzan a pensar el ¿por qué? estos jugadores se consideran los mejores.

Surgen varias preguntas al respecto, para analizar detalles de muchos de estos jugadores, se comienza a formar varios conceptos, analizando elementos como: si es el potencial, la edad o es el salario que sustentan, hace que estos jugadores sean considerados los mejores.

De esta manera se ha propuesto a analizar a varios jugadores junto a sus equipos y ver que tanto es el aporte de estos según la posición en la que juega, el potencial que tiene, el valor que tiene en el mercado y el pago actual que recibe por temporadas de juego.

Métodos.

Importación de las librerías

Para realizar el análisis y la parte visual gráfica se necesita de algunas librerías necesarias como **numpy**, **pandas**, **matplotlib, seaborn** y **scikit-learn**.

Figura 1. Importación de Librerías

```
import numpy as np
import pandas as pd
from matplotlib import pyplot as plt
from sklearn.cluster import KMeans
import seaborn as sns
from TeamClass import Team
```

Fuente: Elaboración propia

Importación del Dataset ya normalizado

Después de importar las librerías es necesario importar el Dataset ya normalizado para que de esta manera se tenga una visualización en tablas, así como en filas y columnas.

Tabla 1. Dataset normalizado

	sofifa_id	first_position	second_position	third_position	hits	team	height_cm	weight_kg	value_eur	wage_eur	..
0	158023	ST	CF	RW	299	FC Barcelona	170	72	95.5	565.0	..
1	20801	ST	LW	NaN	276	Juventus	187	83	58.5	405.0	..
2	190871	CAM	LW	NaN	186	Paris Saint-Germain	175	68	105.5	290.0	..
3	203376	CB	NaN	NaN	127	Liverpool	193	92	78.0	200.0	..
4	200389	GK	NaN	NaN	47	AtlÃ□Â©tico Madrid	188	87	77.5	125.0	..

.	shooting	passing	dribbling	defending	physic	name	age	nationality	potential	overall
.	92.0	92.0	96.0	39.0	66.0	L. Messi	33	Argentina	94	94
.	93.0	82.0	89.0	35.0	78.0	Cristiano Ronaldo	35	Portugal	93	93
.	85.0	87.0	95.0	32.0	58.0	Neymar Jr	28	Brazil	92	92
.	60.0	70.0	71.0	90.0	86.0	V. van Dijk	29	Netherlands	92	91
.	NaN	NaN	NaN	NaN	NaN	J. Oblak	27	Slovenia	93	91

Fuente: (Elaboración propia)

En la presente tabla se obsera algunas caracterisitcas que se encontró en el Dataset

Definiendo funciones para encontrar al peor y al mejor jugador

Figura 2. Funciones para encontrar al peor y al mejor jugador

```
def searchWorstTeam(team,pos):
    Worst = team[team.first_position == pos][team.potential==team.potential[team.first_position == pos].min()]
    return Worst
def searchBestPlayer(team,pos):
    BestPlayer = team[team.first_position == pos][team.potential==team.potential[team.first_position == pos].max()]
    return BestPlayer
```

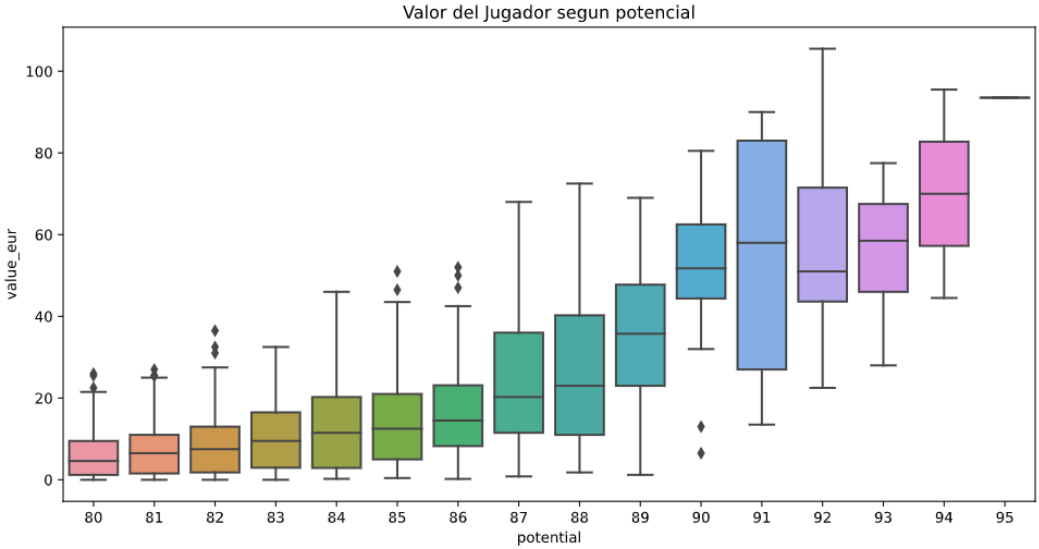
Fuente: Elaboración propia

Visualizaciones del Dataset

Gráfico de comparación del potencial y el valor del mercado del jugador en Euros

Seleccionamos los jugadores que tengan un potencial arriba de 80 y vemos el valor que tienen según su potencial.

Figura 3. Gráfico de comparación del potencial y el valor del mercado del jugador en Euros.



Fuente: (Elaboración propia)

Observaciones

En el gráfico podemos observamos que existen jugadores que están evaluados de una manera errónea ya que algunos valen más, pero no tienen el mayor potencial.

Gráfico de comparación entre el potencial y el sueldo del jugador en Euros

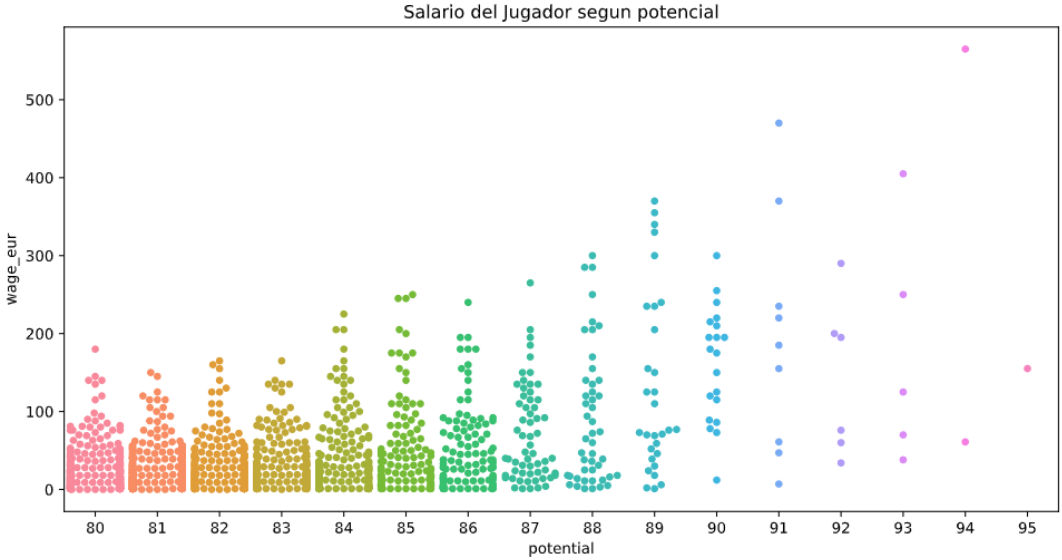
Seleccionamos los jugadores que tengan un potencial arriba de 80 y vemos el valor que tienen según su potencial

Figura 4. Vemos el valor que tienen según su potencial.

```
x = players_dataset.potential[players_dataset.potential>=80]
y = players_dataset.wage_eur[players_dataset.potential>=80]
plt.figure(figsize=(12,6))
sns.swarmplot(x,y)
plt.title('Salario del Jugador segun potencial')
```

Fuente: Elaboración propia

Figura 5. Gráfico de comparación entre el potencial y el salario del jugador en Euros.



Fuente: Elaboración propia

Observaciones

Se puede observar que existen algunas anomalías en la visualización de los datos, en primer lugar, existen jugadores que perciben un sueldo mayor teniendo un potencial menor que el de otros jugadores. Estos datos reflejan que existen jugadores sobrevalorados en el mercado.

Gráfico de comparación entre la posición el sueldo de los jugadores

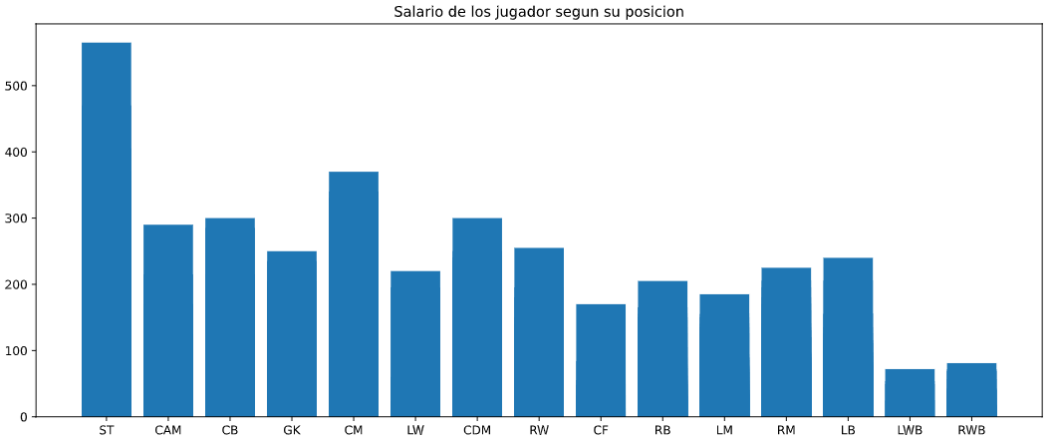
Seleccionamos los jugadores que tengan un potencial arriba de 80 y vemos el valor que tienen según su posición.

Figura 6. Vemos el valor que tienen según su posición.

```
y = players_dataset.wage_eur[players_dataset.potential>=80]
x = players_dataset.first_position[players_dataset.potential>=80]
plt.figure(figsize=(15,6))
plt.bar(x,y)
plt.title('Salario de los jugador segun su posicion')
```

Fuente: (Elaboración propia)

Figura 7. Gráfico de comparación entre la posición el salario de los jugadores



Fuente: Elaboración propia

Observaciones

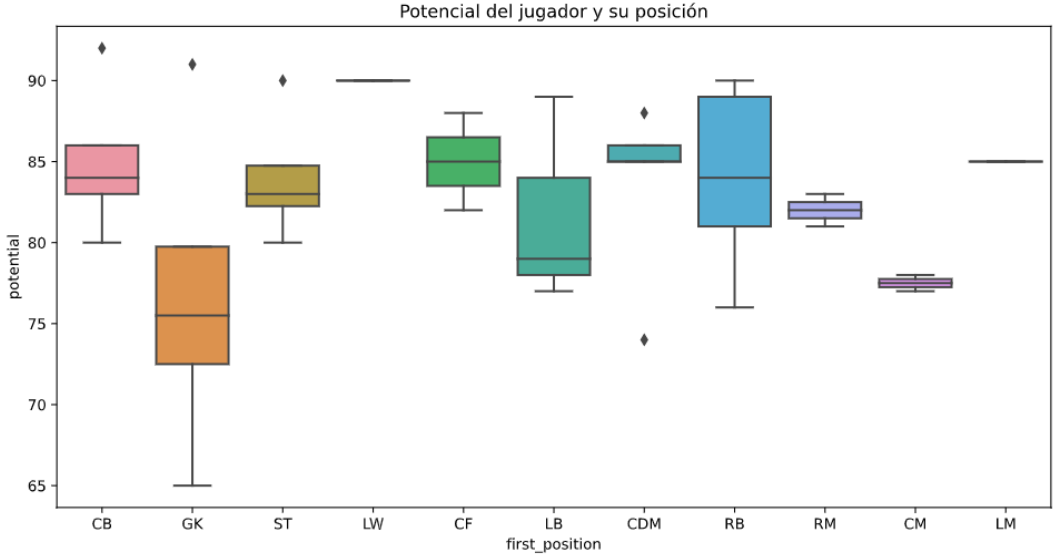
Según el gráfico podemos ver que los delanteros son los mejores pagados en las plantillas seguidos de los volantes de creación.

Figura 8. Análisis de la composición de un equipo de futbol

```
name = 'Liverpool'
players_team = players_dataset[players_dataset.team == name]
y = players_team.potential
x = players_team.first_position
plt.figure(figsize=(12,6))
sns.boxplot(x,y)
plt.title('Potencial del jugador y su posición')
```

Fuente: Elaboración propia

Figura 9. Gráfico de comparación entre el potencial y la posición del jugador.



Fuente: Elaboración propia

Observaciones

Según el gráfico se puede observar que existen jugadores que no aportan

mucho en el equipo, caso en el que encontramos a un GK (Portero) que tiene un rendimiento demasiado bajo.

Analizando cual es jugador con el potencial bajo

Figura 10. Analizando cual es jugador con el potencial bajo

```
pos = 'GK'
worst = searchWorstTeam(players_team,pos)
worst[['name','nationality','first_position','potential','value_eur','wage_eur']]
```

Fuente: Elaboración propia

Tabla 2. Extracción del peor jugador.

	name	nationality	first_position	potential	value_eur	wage_eur
10562	A. Lonergan	England	GK	65	0.15	8.0

Fuente: Elaboración propia

Figura 11. A. Lonergan según el análisis de datos el peor jugador del dataset y esta muy infravalorado



Fuente: (BeSoccer, 2020)

Aplicación de la Regresión Lineal

Declaración de hipótesis

Figura 12. Funcion de la hipótesis.

```
def h(X,theta):
    return np.dot(np.transpose(theta),X)
```

Fuente: Elaboración propia

Declaración del costo

Figura 13. Función consto

```
def J(X,y,theta):
    m = X.shape[1]
    return (1/(2*m))*np.sum(np.square(h(X,theta)-y))
```

Fuente: Elaboración propia

Declaración de la derivada de la función de costo

Figura 14. Derivada de la función

```
def dJ(X,y,theta):
    m=X.shape[1]
    return (1/m)*np.dot((h(X,theta)-y),np.transpose(X)).reshape((-1,1))
```

Fuente: Elaboración propia

Declaración de x como la columna que almacenara los valores de los jugadores y del potencial de los mismos

Figura 15. Almacenamiento de valores de los jugadores.

```
x = players_dataset[['value_eur']]
y = players_dataset.potential
x.to_numpy()
y.to_numpy()
y = np.array(y).reshape(1,-1)
x = np.array(x).reshape(1,-1)
```

Fuente: Elaboración propia

Declaración de Alpha y las iteraciones del modelo

Figura 16. Pruebas para determinar su almacenamiento.

```
alpha = 0.01
iterations = 400
```

Fuente: Elaboración propia

Declaración theta

Figura 17. Theta como random. Fuente: (Elaboración propia)

```
theta = np.random.random((X.shape[0],1))
theta
```

```
array([[0.58673882],
       [0.00666722]])
```

Fuente: Elaboración propia

Obtención del costo inicial

Figura 18. Obtención del costo inicial.

```
registro_grads = []
registro_J = []
J_init = J(X,y,theta)
print(f'Costo inicial:{J_init}')
```

```
Costo inicial:2573.490966374446
```

Fuente: Elaboración propia

Obtención del costo final

Figura 19. Obtención del costo final

```
for i in range(iterations):
    dtheta = dJ(X,y,theta)
    theta = theta-alpha*dtheta
    registro_J.append(J(X,y,theta))
print(f'Costo Final:{J(X,y,theta)}')
```

```
Costo Final:14.968570365001174
```

Fuente: Elaboración propia

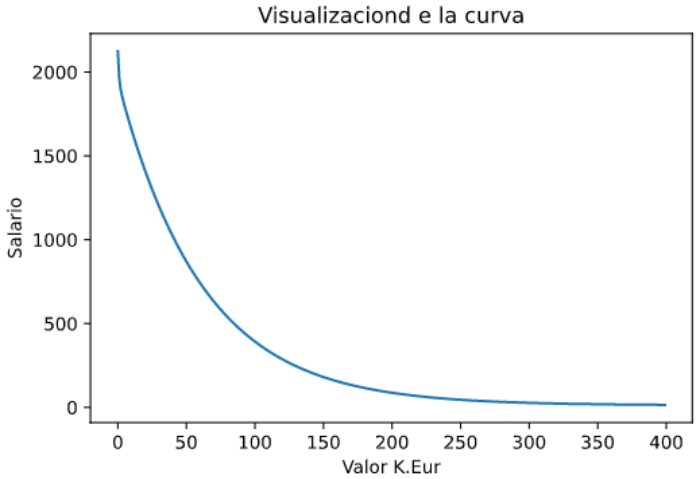
Visualización de la curva

Figura 20. Impresión de la curva.

```
it = list(range(iterations))
plt.plot(it,registro_J)
plt.title('Visualizaciond e la curva')
plt.xlabel('Valor K.Eur')
plt.ylabel('Salario')
plt.show()
```

Fuente: Elaboración propia

Figura 21. Gráfica de una curva que haciende de 0 2000.



Fuente: Elaboración propia

Observaciones

Se puede observar que existe una constante entre el valor del jugador y el salario que persive.

Visualización de datos para una tarea de detección de anomalías

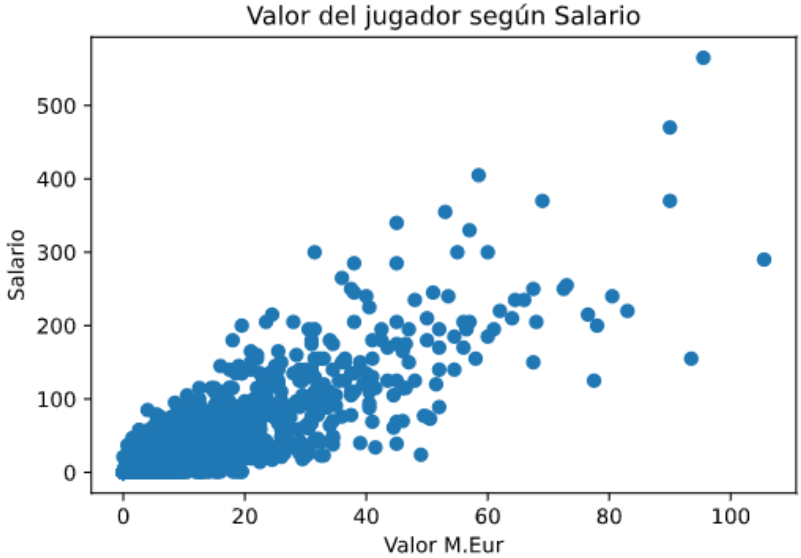
Gráfico del valor del mercado y el sueldo de los jugadores

Figura 22. Imprimimos las anomalias que existen.

```
x = players_dataset.value_eur[players_dataset.potential>=75]
y = players_dataset.wage_eur[players_dataset.potential>=75]
plt.title('Valor del jugador según Salario')
plt.xlabel('Valor M.Eur')
plt.ylabel('Salario')
plt.scatter(x,y)
```

Fuente: Elaboración propia

Figura 23. Valor del jugador según salario.



Fuente: Elaboración propia

Observaciones

Podemos observar que existen un jugadores que no cumplen con la constante, por lo cual realizaremos una tara de clustering para ver cual es la tendencia.

Aplicando Clustering al gráfico del valor del mercado y el sueldo de los jugadores

Figura 24. Imprimimos el clustering del grafico.

```
X = pd.concat([x,y], axis=1)
kmeans = KMeans(n_clusters=1, init='k-means++', max_iter=300, random_state=0)
pred_y = kmeans.fit_predict(X)
plt.scatter(x, y)
plt.scatter(kmeans.cluster_centers_[:, 0], kmeans.cluster_centers_[:, 1], s=300, c='red')
plt.title('Valor del jugador según Salario')
plt.xlabel('Valor M.Eur')
plt.ylabel('Salario')
plt.show()
```

Fuente: Elaboración propia

Observaciones

Despues de realizar una tarea de Clustering vemos que la mayoria de jugadores valen menos de 20 millones y persiben sueldos menores de 100 mil euros. Se podria decir que esto es una constante y es el salario promedio de los jugadores de futbol.

Aplicando al Clustering de 4 etapas sueldos y su valor de los jugadores

Figura 25. Codigo de implementación para graficar con Matplotlib.

```
from matplotlib.patches import Ellipse

def draw_ellipse(position, covariance, ax=None, **kwargs):
    ax = ax or plt.gca()

    # Convert covariance to principal axes
    if covariance.shape == (2, 2):
        U, s, Vt = np.linalg.svd(covariance)
        angle = np.degrees(np.arctan2(U[1, 0], U[0, 0]))
        width, height = 10 * np.sqrt(s)
    else:
        angle = 0
        width, height = 10 * np.sqrt(covariance)

    # Draw the Ellipse
    for nsig in range(1, 4):
        ax.add_patch(Ellipse(position, nsig * width, nsig * height,
                             angle, **kwargs))

def plot_gmm(gmm, X, label=True, ax=None):
    ax = ax or plt.gca()
    labels = gmm.fit(X).predict(X)
    if label:
        ax.scatter(X[:, 0], X[:, 1], c=labels, s=40, cmap='viridis', zorder=2)
    else:
        ax.scatter(X[:, 0], X[:, 1], s=40, zorder=2)
    ax.axis('equal')

    w_factor = 0.2 / gmm.weights_.max()
    for pos, covar, w in zip(gmm.means_, gmm.covariances_, gmm.weights_):
        draw_ellipse(pos, covar, alpha=w * w_factor)
```

Fuente: Elaboración propia

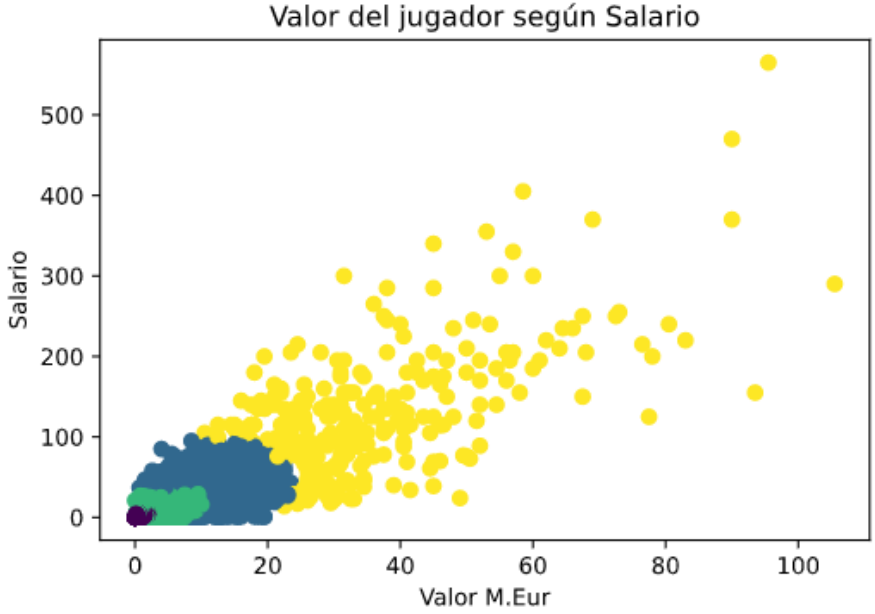
Figura 26. Imprimir los graficos.

```
X = pd.concat([x,y], axis=1)
X = X.to_numpy()
gmm = GaussianMixture(n_components=4).fit(X)
labels = gmm.predict(X)
plt.scatter(X[:, 0], X[:, 1], c=labels, s=40, cmap='viridis');
plt.title('Valor del jugador según Salario')
plt.xlabel('Valor M.Eur')
plt.ylabel('Salario')
plt.show()
```

Fuente: Elaboración propia

Visualziación grafica de las 4 etapas

Figura 27. grafica de las 4 etapas.



Fuente: Elaboración propia

Observaciones

Se puede observar que exisisten 4 segmentos marcados, donde existen jugadores que ganan lo que valen, pero en la cuarta etapa existen jugadores que valen más pero no se les paga lo que corresponde, con excepción de algunos que valen menos en el mercado pero ganan mas que otros.

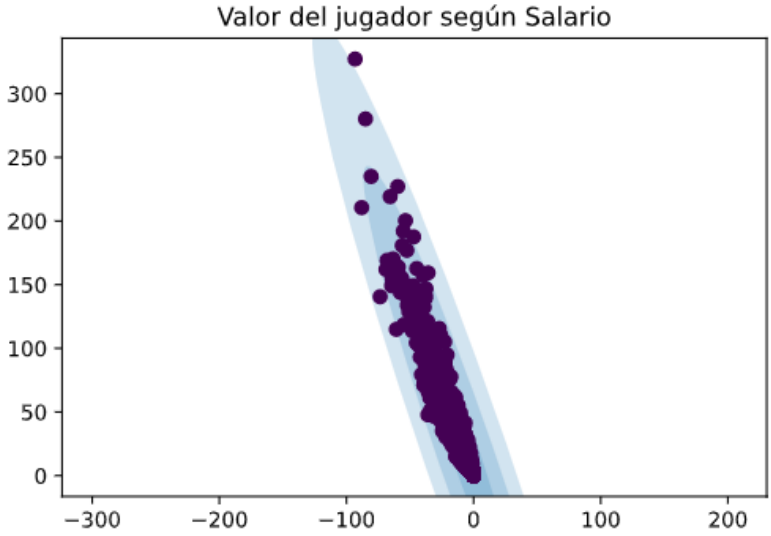
Visualización de jugadores fuera del rango de con los mejores sueldos y el valor del mercado

Figura 28. Utilizando las librerias de sckityleran para imprimir la grafica.

```
from sklearn.mixture import GaussianMixture
X = pd.concat([x,y], axis=1)
X = X.to_numpy()
rng = np.random.RandomState(13)
X_stretched = np.dot(X, rng.randn(2, 2))
gmm = GaussianMixture(n_components=1, covariance_type='full', random_state=42)
plot_gmm(gmm, X_stretched)
plt.title('Valor del jugador según Salario')
plt.show()
```

Fuente: Elaboración propia

Figura 29. jugadores fuera del rango de con los mejores sueldos y el valor del mercado.



Fuente: Elaboración propia

Observaciones

Al igual que el gráfico de 4 etapas podemos observar que exisisten 4 secmentos marcados, en este grafico vemos que solo existe un jugadopr que según la constante vale menos de lo que le pagan, este jugador es Lionel Messi que según los ultimos tiempos vajo su valor en el mercado.

Como se ha podido observar a lo largo del documento se exite un pequeño circulo rojo interlineado en donde existe un jugador que ha sido considerado el mejor de la historia hasta esta fechas y es:

Figura 30. Lionel Messi el jugador mas sobrevalorado en los utlimos tiempos.



Fuente: (As, 2020)

El jugador está evaluado por los 100,00 euros y es el mejor ya que es conocido por su larga travesía de partidos jugados y ganados en su largo recorrido. Por el momento sigue manteniendo la mejor racha jugando en el Barcelona.

Conclusión

Después de realizar este análisis se detectaron varias anomalías. En primer lugar, existen jugadores que no deberían ser centrados para el club puesto que según estadísticas y características de los jugadores estos no aportaran en mucho al equipo.

En segundo lugar, existen jugadores que actualmente valen menos de lo que solían valer, esto puede deberse al rendimiento que ahora tienen en sus respectivos equipos, en vista de ello, estos jugadores reciben salarios exorbitantes, los cuales consiguen la constante de salario vs valor del jugador, de igual manera existen jugadores que según sus estadísticas valen más de lo que perciben salarialmente. En conclusión, si bien existen jugadores que se les paga lo que valen, existen jugadores que no perciben el sueldo que merecen, esto se pude dar por varios motivos que pueden ser ajenos a las

estadísticas del jugador. Estos datos fueron extraídos de las características que tienen los jugadores en un videojuego, si bien estos datos son realizados por la FIFA no son datos exactos porque cuando se habla de futbol todo puede pasar.

Proyecto

Repositorio para ver el proyecto completo.
https://github.com/AlbertAC/ProyectoFinal/blob/master/SystemFifa.ipynb

Referencias Bibliográficas

- AlbertAC. (22 de 10 de 2020). *GitHub*. Obtenido de AprendizajeAutomatico: https://github.com/AlbertAC/ProyectoFinal/blob/master/SystemFifa.ipynb
- As. (2020). *As.* Obtenido de https://resultados.as.com/resultados/ficha/deportista/messi/15167/
- BeSoccer. (2020). *BeSoccer.* Obtenido de https://www.besoccer.com/player/a-lonergan-12304
- Kafagy, A. (26 de 11 de 2019). *GitHub.* Obtenido de fifa-FUT-Data: https://github.com/kafagy/fifa-FUT-Data
- Leone, S. (26 de 10 de 2020). *Kaggle*. Obtenido de FIFA 21 complete player dataset: https://www.kaggle.com/stefanoleone992/fifa-21-complete-player-dataset?select=players_21.csv
- Mishra, A. (15 de 09 de 2020). *Kaggle.* Obtenido de Conjunto de datos de jugador completo de FIFA 2021: https://www.kaggle.com/aayushmishra1512/fifa-2021-complete-player-data

Artículo Recibido: 23-09-2020
Artículo Aceptado: 11-10-2020

Artículo Original Pág. 35 - 46 ISSN 2415-2323

ILUMATE VOLUMEN 12

EXPLORACIÓN DE DATOS Y SISTEMA DE RECOMENDACIÓN DE LIBROS

Data Exploration & Book's System Recommendation

Lozada Gómez José Adolfo
jalgoz95@gmail.com
Instituto de Investigación en Ciencia y Tecnología,
Universidad La Salle-Bolivia

Arana Sánches Esmeralda Yennifer
aranaesmeralda7@gmail.com
Instituto de Investigación en Ciencia y Tecnología,
Universidad La Salle-Bolivia

Resumen

La presente investigación se realizó con la intención de aplicar los conocimientos adquiridos del área de Aprendizaje Automático en las publicaciones de libros en la red. El Objetivo fue analizar las propiedades de cada libro (libros más leídos, autores más consultados y calificación promedio de un autor o libro) y las relaciones que pueden tener dentro de un conjunto de datos. En base al análisis, añadir un sistema de recomendación para el uso de los usuarios. La metodóloga empleada fue cuantitativa de diseño de investigación no experimental tipo de estudio correlacional. Los resultados fueron, que utilizando el Machine Learning se pudo analizar los datos y así

poder brindar un sistema de recomendación.

Palabras claves

Análisis de datos, Relación de datos, Kmeans Clustering, KNN.

Abstract

This research was carried out with the intention of applying the knowledge acquired from the area of Machine Learning in the publication of books on the web. The objective was to analyze the properties of each book (most read books, most consulted authors and average rating of an author or book) and the relationships they may have within a data set. Based on the analysis, add a recommendation system for user use. The methodology used was quantitative non-experimental research design type of correlational study. The results were that using Machine Learning it was possible to analyze the data and thus be able to provide a recommendation system.

Key words

Data analysis, Data relations, Kmeans Clustering, KNN.

Introducción

La principal idea del proyecto es analizar y reconocer las propiedades más importantes con las que cuenta cada libro del dataset, encontrar las relaciones existentes dentro, en este caso información sobre los libros existentes, explorarlo y verificar que los datos sean correctos y estén libres de anomalías para su próxima ejecución en un sistema de recomendación.

Para la solución del problema, se debe reconocer el tamaño y las columnas del dataset para una noción básica de su estructura, utilizando la librería de NumPy. A continuación, seguir con la exploración de los datos usando las

gráficas de la librería Seaborn. Para utilizar los datos los libros se dividirán en grupos usando los métodos de Kmeans y Neighbors de la librería Sklearn que nos ayudarán a identificar al tipo de preferencia según las búsquedas y gustos del usuario.

Referentes Conceptuales

El aprendizaje automático es un paradigma que puede referirse al aprendizaje de experiencias pasadas (que en este caso son datos previos) para mejorar el rendimiento futuro. El único enfoque de este campo son los métodos de aprendizaje automático. El aprendizaje se refiere a modificación o mejora del algoritmo basado en "experiencias" pasadas automáticamente sin ninguna ayuda externa de humano. (Das & Narayan Behera, 2017).

El aprendizaje automático es la ciencia de enseñar a las computadoras a hacer predicciones basadas en datos. En un nivel básico, el aprendizaje automático implica dar a una computadora un conjunto de datos y pedirle que haga una predicción. Al principio, la computadora tendrá muchas predicciones incorrectas, sin embargo, en el transcurso de miles de predicciones, la computadora actualizará su algoritmo para hacer mejores predicciones (Norman, 2017, pág. 60).

Tipos de aprendizaje según (Cambronero, 2006):

> **Aprendizaje inductivo:** Creamos modelos de conceptos a partir de generalizar ejemplos simples. Buscamos patrones comunes que expliquen los ejemplos. Se basa en el razonamiento inductivo: Obtiene conclusiones generales de información especifica. El conocimiento

obtenido es nuevo. No preserva la verdad (nuevo conocimiento puede invalidar lo obtenido). No tiene una base teórica bien fundamentada.

Aprendizaje analítico o deductivo: Aplicamos la deducción para obtener descripciones generales a partir de un ejemplo de concepto y su explicación. Se basa en el razonamiento deductivo: Obtiene conocimiento mediante el uso de mecanismos bien establecidos. Este conocimiento no es nuevo (ya está presente implícitamente). Nuevo conocimiento no invalida el ya obtenido. Se fundamenta en la lógica matemática.

Aprendizaje analógico: Busca soluciones a problemas nuevos y encontrar similaridades con problemas ya conocidos y adaptando sus soluciones.

Aprendizaje genético: Aplica algoritmos inspirados en la teoría de la evolución para encontrar descripciones generales a conjuntos de ejemplos. Aprendizaje conexionista: Busca descripciones generales mediante el uso de la capacidad de adaptación de redes de neuronas artificiales.

Uno de los objetivos que persigue el Machine Learning es clasificar automáticamente los objetos o datos. En base a esto podríamos mencionar según (APD, 2019) a tres clases de algoritmos:

Aprendizaje supervisado

En el aprendizaje supervisado, la máquina se enseña con el ejemplo. De este modo, el operador proporciona al algoritmo de aprendizaje automático un conjunto de datos conocidos que incluye las entradas y salidas deseadas, y el algoritmo debe encontrar un método para determinar cómo llegar a esas entradas y salidas.

Mientras el operador conoce las respuestas correctas al problema, el algoritmo identifica patrones en los datos, aprende de las observaciones y hace predicciones. El algoritmo realiza predicciones y es corregido por el operador, y este proceso sigue hasta que el algoritmo alcanza un alto nivel de precisión y rendimiento.

Aprendizaje sin supervisión

Aquí, el algoritmo de aprendizaje automático estudia los datos para identificar patrones. No hay una clave de respuesta o un operador humano para proporcionar instrucción. En cambio, la máquina determina las correlaciones y las relaciones mediante el análisis de los datos disponibles.

En un proceso de aprendizaje no supervisado, se deja que el algoritmo de aprendizaje automático interprete grandes conjuntos de datos y dirija esos datos en consecuencia. Así, el algoritmo intenta organizar esos datos de alguna manera para describir su estructura. Esto podría significar la necesidad de agrupar los datos en grupos u organizarlos de manera que se vean más organizados.

A medida que evalúa más datos, su capacidad para tomar decisiones

sobre los mismos mejora gradualmente y se vuelve más refinada.

Aprendizaje por refuerzo:

El aprendizaje por refuerzo se centra en los procesos de aprendizajes reglamentados, en los que se proporcionan algoritmos de aprendizaje automáticos con un conjunto de acciones, parámetros y valores finales.

Al definir las reglas, el algoritmo de aprendizaje automático intenta explorar diferentes opciones y posibilidades, monitorizando y evaluando cada resultado para determinar cuál es el óptimo.

Para este caso se utilizó el algoritmo de clasificación no supervisada **k-means**. A continuación, algunos conceptos de las librerías que se utilizaron.

NumPy, o Python numérico, es una biblioteca basada en Python para cálculos matemáticos y matrices de procesamiento. Python no admite estructuras de datos en más de una dimensión, y los contenedores como listas, tuplas y diccionarios son unidimensionales. Los contenedores y tipos de datos incorporados en Python no se pueden reestructurar en más de una dimensión y tampoco se prestan a cálculos complejos. Estos inconvenientes son limitaciones para algunas de las tareas involucradas al analizar datos y construir modelos, lo que hace que los arreglos sean una estructura de datos vital. (Rajagopalan, 2020).

Pandas es una librería que proporciona estructuras de datos y herramientas de análisis de datos de alto rendimiento para el lenguaje de programación Python (Vidal, 2019).

La librería Sklearn sirve para ilustrar el manejo practico de los algoritmos de aprendizaje automático, además implementa numerosos algoritmos de aprendizaje automático, así como diversos procedimientos relacionados y, junto con la característica sencillez de Python, proporciona un entorno rápido y cómodo para su uso (Jose Manuel Robles, 2020).

El algoritmo K-means, creado por MacQueen en 1967 es el algoritmo de clustering más conocido y utilizado ya que es de muy simple aplicación y eficaz. Sigue un procedimiento simple de clasificación de un conjunto de objetos en un determinado número K de clústeres, K determinado a priori (Cambronero, 2006).

El análisis de clúster o Clustering agrupa un conjunto de objetos de datos en clústers o grupos de manera que en cada grupo los objetos sean similares entre sí y disimiles de los objetos de otros grupos. En la actualidad, existen distintas técnicas de agrupamiento que permiten cumplir con esta tarea. En búsqueda de un algoritmo más natural se hizo uso del concepto de densidad atómica de los elementos como base para generar uno nuevo. El algoritmo propuesto tiene como ventajas, poseer un método concreto de selección de centroides, además de tener mejores agrupamientos que otros algoritmos basados en centroides como k-means y k-medoids (Medina, 2016).

El algoritmo consta de tres pasos según (Laruta, 2020):

1. **Inicialización:** Una vez hayamos seleccionados el número de grupos, k, estableceremos k centroides en el espacio de los datos pudiendo escogerlos de forma aleatoria.

2. **Asignación objetos a los centroides:** Cada objeto de los

datos deberá ser asignado al centroide más cercano que se encuentre.

3. **Actualización centroides:** Se debe actualizar la posición que tiene cada centroide tomando como nuevo centroide la posición promedio de los objetos pertenecientes a dicho grupo.

La asignación de objetos a los centroides y la actualización de centroides va a ser repetitiva hasta que los centroides no se muevan o caso contrario se muevan por debajo de una distancia umbral en cada paso.

Este algoritmo se lo podría definir como una función de optimización de la suma de las distancias de cada objeto al centroide de su clúster.

Tiene varias ventajas entre las principales están que es un método rápido y sencillo, sin embargo, para ello se requieres decidir el valor de K y el resultado final que se obtenga dependerá mucho de donde estén iniciados los centroides.

Metodología

La metodóloga empleada fue cuantitativa de diseño de investigación no experimental tipo de estudio correlacional.

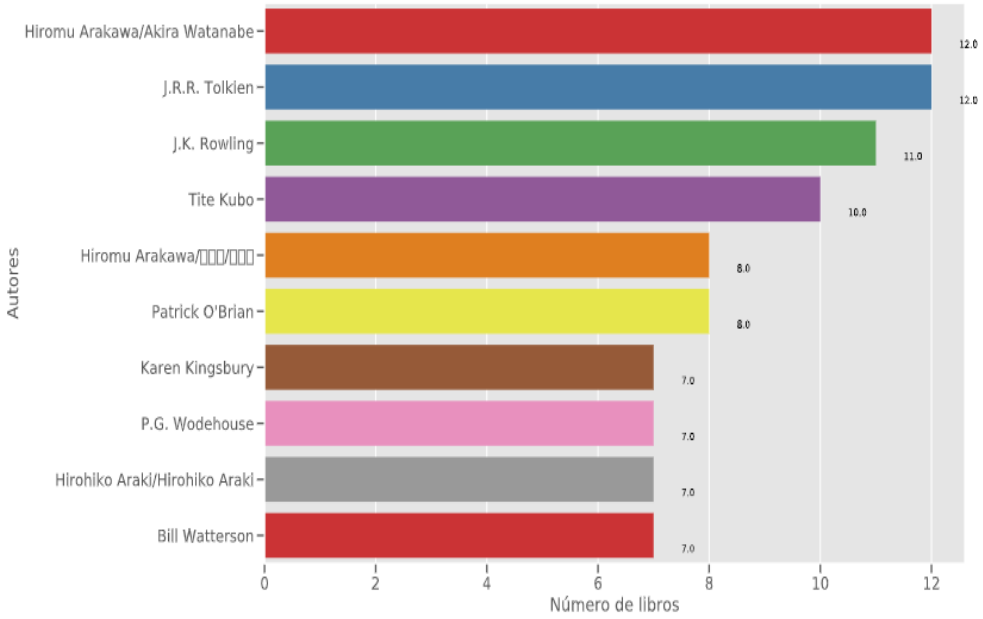
Los instrumentos utilizados fueron:

- Lenguaje de programación Python y las librerías Numpy, Pandas, Seaborn y Sklearn.
- Editor de código Visual Studio Code

Resultados y discusión:

Como resultado se pudo encontrar algunas ambigüedades en la relación de datos, como ser la calificación promedio y el número de reseñas, debe haber persistencia al cambio de datos en las columnas y tendremos como resultado un dataset que esté libre de anomalías, así como una exploración gráfica e informativa en la división de grupos en los libros, y la buena ejecución en el sistema de recomendación.

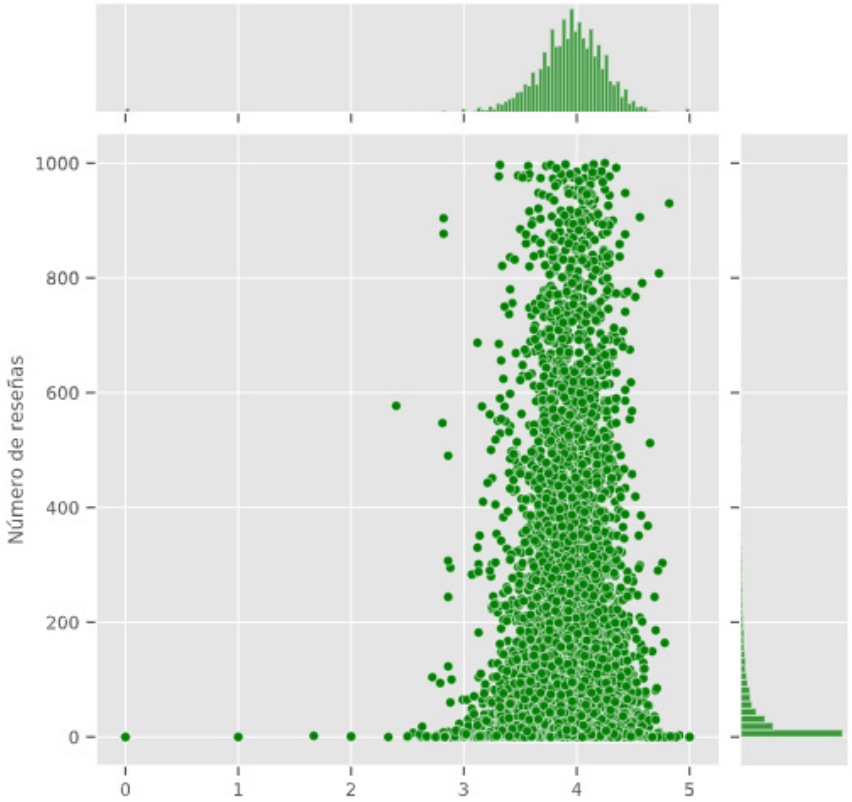
Fig 1. Cantidad de libros



Fuente Elaboración propia (VS Code, 2020)

La figura 1 muestra la cantidad de libros que existen por autor en el software que se hizo correr.

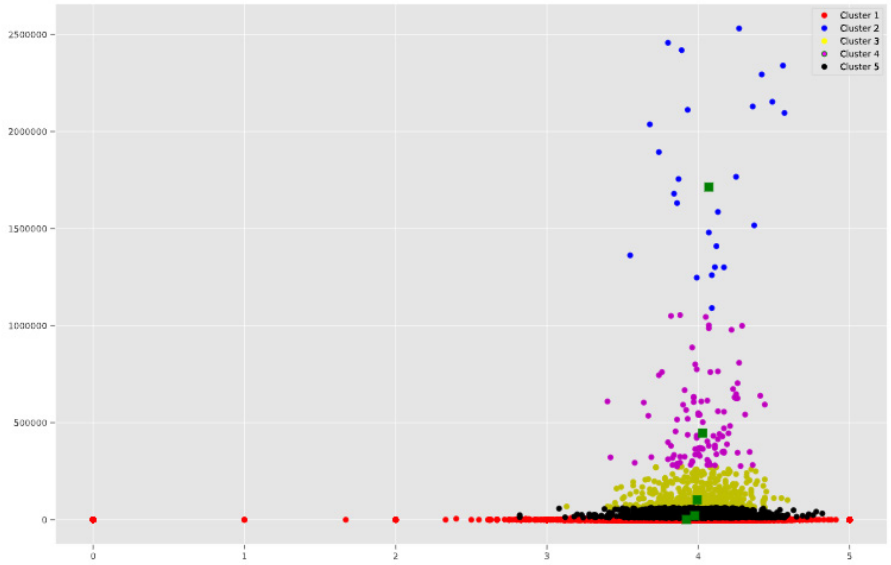
Fig 2. Número de reseñas y calificación promedio



Fuente Elaboración propia (VS Code, 2020)

No sé puede afirmar o negar si existe una relación entre el número de reseñas y la calificación promedio

Fig 3. División en clústeres del dataset



Fuente Elaboración propia (VS Code, 2020)

Conclusiones

Se logra ver que todas las herramientas que ofrece python al momento de un análisis de los datos como también para los diversos métodos de aprendizaje automático son potentes, fáciles de usar y entender, obteniendo una gran información sobre el conjunto de datos ya trabajados, dando a los desarrolladores poder manipular y usarlos a un beneficio final, en este caso para un sistema de recomendación.

Referencias

- APD, R. (04 de 04 de 2019). apd. Obtenido de apd: https://www.apd.es/algoritmos-del-machine-learning/#:~:text=Este%20tipo%20de%20algoritmos%20por,de%20caracter%C3%ADsticas%2C%20utilizando%20la%20probabilidad.

- Cambronero, C. G. (2006). ALGORITMO DE APRENDIJAZE KNN & KMEANS. 7.
- Das, K., & Narayan Behera, R. (2 de 2017). www.ijircce.com. Obtenido de www.ijircce.com: https://www.researchgate.net/profile/Rabi-Behera-2/publication/316273553_A_Survey_on_Machine_Learning_Concept_Algorithms_and_Applications/links/59017eb94585156502a094fd/A-Survey-on-Machine-Learning-Concept-Algorithms-and-Applications.pdf
- Jose Manuel Robles, R. C. (2020). Big data para cientificos sociales. Madrid: RALI, S.A.
- Medina, O. M. (2016). SEDICI. Retrieved from http://sedici.unlp.edu.ar/handle/10915/56757
- Moya, R. (29 de 05 de 2016). Jarroba. Obtenido de Jarroba: https://jarroba.com/k-means-python-scikit-learn-ejemplos/
- Norman, A. T. (2017). Aprendizaje Automática en Acción. Tektime .
- Rajagopalan, G. (23 de 12 de 2020). SpringerLink. Obtenido de SpringerLink: https://link.springer.com/chapter/10.1007/978-1-4842-6399-0_5#citeas
- Vidal, J. H. (2019). Aprendizaje automático en el diseño de un detector de estrés a partir de señales biomédicas. Universidad de La Laguna.

Artículo recibido: 23-09-2020
Artículo aceptado: 05-11-2020

Artículo Original Pág. 47 - 60 ISSN 2415-2323

ILUMATE VOLUMEN 12

ANÁLISIS DE RIESGO PARA PRÉSTAMOS BANCARIOS

Risk analysis for bank loans

Juan Salvador Vilca Paredes
slvdrvlc@gmail.com
Instituto de Investigación en Ciencia y Tecnología,
Universidad La Salle - Bolivia

Resumen

En estos tiempos de pandemia, existen muchas dificultades a la hora de saber a quién prestar, inclusive si el cliente es una persona que demuestra solvencia. El objetivo es identificar si es prudente o no realizar un préstamo a los clientes según el monto del préstamo que solicitan y los datos que se tienen de los mismos. Es un problema de clasificación para predecir si es recomendable o no un préstamo. Se debe predecir valores discretos basados en un conjunto dado de variables independientes. Se empleo la metodología cuantitativa con diseño de investigación no experimental en el tipo de estudio correlacional. Los resultados fueron que no siempre las hipótesis pueden estar alineadas con el análisis y evaluación de los datos, pues los resultados demostraron que los solicitantes con más ingresos no necesariamente eran los que tenían mayor posibilidad de acceder al préstamo.

Palabras claves

Aprendizaje automático, ciencia de datos, clasificación, python, regresión logística

Abstract

In these times of pandemic, there are many difficulties when it comes to knowing who to lend to, even if the client is a person who demonstrates solvency. The objective is to identify whether or not it is prudent to make a loan to clients based on the amount of the loan they request and the data they have about them. It is a classification problem to predict whether or not a loan is recommended. Discrete values must be predicted based on a given set of independent variables. The quantitative methodology was used with a non-experimental research design in the correlational type of study. The results were that the hypotheses could not always be aligned with the analysis and evaluation of the data, since the results showed that the applicants with the highest income were not necessarily the ones with the greatest possibility of accessing the loan

Key words

Machine learning, data science, classification, python, logistic regression

Introducción

La predicción de préstamos es un problema muy común de la vida real que cada banco enfrenta. Si se hace correctamente, puede ahorrar muchas horas de trabajo. Es importante tener en cuenta que las predicciones por lo general son de apoyo para la toma de decisiones, es decir, que apoyan a profesionales a tomar decisiones en sus áreas y no así reemplazarlos en sus tareas.

El trabajo consiste en ayudar a una empresa bancaria que realiza préstamos con presencia en todas las áreas urbanas, semiurbanas y rurales. El cliente primero solicita un préstamo hipotecario después de que la compañía valida la elegibilidad del cliente para el préstamo. Sin embargo, hacer esto manualmente lleva mucho tiempo. Se propone realizar la predicción de la aprobación de los clientes para el préstamo ayudando de esta manera a respaldar la elegibilidad del préstamo en función de la información del cliente.

Se utilizó el modelo de regresión logística para realizar una clasificación binaria donde las predicciones fueron:

Y/1(si) para indicar que el préstamo es recomendable.
N/0(no) para indicar que el préstamo no es recomendable.

Referentes conceptuales.

La Regresión Logística según Amat (2016), desarrollada por David Cox en 1958, es un método de regresión que permite estimar la probabilidad de una variable cualitativa binaria en función de una variable cuantitativa.

Es un problema de clasificación en el que tenemos que predecir si se aprobase o no un préstamo. En un problema de clasificación, tenemos que predecir valores discretos basados en un conjunto dado de variables independientes, la clasificación puede ser de dos tipos:

Clasificación binaria: en esta clasificación tenemos que predecir cualquiera de las dos clases dadas. Por ejemplo: clasificar el género como masculino o femenino, predecir el resultado como ganar o perder, etc., o como en este paso aceptar o rechazar la solicitud.

Clasificación multiclase: Aquí tenemos que clasificar los datos en tres o más clases. Por ejemplo: clasificar el género de una película como comedia, acción o romántico, clasificar frutas como naranjas, manzanas o peras, etc.

> Según Amat (2016):
> "Una de las principales aplicaciones de la regresión logística es la de clasificación binaria, en el que las observaciones se clasifican en un grupo u otro dependiendo del valor que tome la variable empleada como predictor. Por ejemplo, clasificar a un individuo desconocido como hombre o mujer en función del tamaño de la mandíbula" (p.3).

Para la predicción de la variable objetivo se aplica el modelo de regresión

logística para predecir el resultado binario.

Según Fernandez (2011):
"Con el fin de estimar **β** = β_1 β_2 L βk y analizar el comportamiento del modelo estimado se toma una muestra aleatoria de tamaño n dada por ($X_{i,}$ $Y_{i)\,i}$ = $_1$, $_2$, ...,$_n$ donde el valor de las variables independientes es X_i = ($X_{i1,}$ X_{i2} , ... , X_{ik}) e Y_i ∈ [0.1] es el valor observado de Y en el i-ésimo elemento de la muestra" (p.4)

La regresión logística, a pesar de su nombre, es un modelo lineal para clasificación en lugar de regresión. La regresión logística también se conoce en la literatura como regresión logit, clasificación de máxima entropía (MaxEnt) o clasificador log-lineal. En este modelo, las probabilidades que describen los posibles resultados de un solo ensayo se modelan utilizando una función logística.

Métodos.

Se emplean las siguientes librerías de Python:

- pandas
- numpy
- seaborn
- matplotlib
- sklearn

Pandas es una de las librerías de python más útiles para los científicos de datos. Las estructuras de datos principales en pandas son Series para datos en una dimensión y DataFrame para datos en dos dimensiones.

Estas son las estructuras de datos más usadas en muchos campos tales como finanzas, estadística, ciencias sociales y muchas áreas de ingeniería. Pandas destaca por lo fácil y flexible que hace la manipulación de datos y el análisis de datos.

NumPy proporciona una estructura de datos universal que posibilita el análisis de datos y el intercambio de datos entre distintos algoritmos. Las estructuras de datos que implementa son vectores multidimensionales y matrices con capacidad para gran cantidad de datos.
Además, esta librería proporciona funciones matemáticas de alto nivel que operan en estas estructuras de datos.

Seaborn es una librería gráfica basada en matplotlib, especializada en la visualización de datos estadísticos. Se caracteriza por ofrecer un interfaz de alto nivel para crear gráficos estadísticos visualmente atractivos e informativos.

Seaborn considera la visualización como un aspecto fundamental a la hora de explorar y entender los datos. Se integra muy bien con la librería de manipulación de datos pandas.

Matplotlib es la librería gráfica de python estándar y la más conocida. Se puede usar matplotlib para generar gráficos de calidad necesaria para publicarlas tanto en papel como digitalmente.

Con matplotlib se puede crear muchos tipos de gráficos: series temporales, histogramas, espectros de potencia, diagramas de barras, diagramas de errores, etc.

Scikit-learn es una librería de python para Machine Learning y Análisis de Datos. Está basada en NumPy, SciPy y Matplotlib. Las ventajas principales de scikit-learn son su facilidad de uso y la gran cantidad de técnicas de aprendizaje automático que implementa.

Con scikit-learn se puede realizar aprendizaje supervisado y no supervisado. Podemos usarlo para resolver problemas tanto de clasificación y como de regresión.

Es muy fácil de usar y aprender porque tiene una interfaz simple y muy

consistente. Se puede verificar que el interfaz es consistente cuando es posible cambiar de técnica de machine learning cambiando sólo una línea de código.

Otro punto a favor de scikit-learn es que los valores de los hiper-parámetros tienen unos valores por defecto adecuados para la mayoría de los casos.
Es una de las herramientas más eficientes que contiene muchas funciones incorporadas que se pueden usar para modelar en Python.

Específicamente se utiliza la clase sklearn.linear_model. Logistic Regression para implementar el modelo, el cual tiene los siguientes métodos:

Fig. 1 Métodos de la clase LogisticRegression

`decision_function`(X)	Predecir puntuaciones de confianza para muestras.
`densify`()	Convierta la matriz de coeficientes a un formato de matriz densa.
`fit`(X, y [, sample_weight])	Ajuste el modelo de acuerdo con los datos de entrenamiento proporcionados.
`get_params`([profundo])	Obtenga parámetros para este estimador.
`predict`(X)	Predecir etiquetas de clase para muestras en X.
`predict_log_proba`(X)	Predecir el logaritmo de las estimaciones de probabilidad.
`predict_proba`(X)	Estimaciones de probabilidad.
`score`(X, y [, sample_weight])	Devuelve la precisión media en las etiquetas y los datos de prueba dados.
`set_params`(** params)	Establezca los parámetros de este estimador.
`sparsify`()	Convierta la matriz de coeficientes a formato disperso.

Fuente: Elaborado por: Sklearn

También es importante mencionar que las pruebas fueron realizadas en Google Colab.

Colab es un servicio en la nube, basado en los Notebooks de Jupyter, que permite el uso gratuito de las GPUs y TPUs de Google, con librerías como: Scikit-learn, PyTorch, TensorFlow, Keras y OpenCV. Con Python 2.7 y 3.6, aún no está disponible para R y Scala.

Aunque tiene algunas limitaciones, es una herramienta ideal, no solo para

practicar y mejorar nuestros conocimientos en técnicas y herramientas de Data Science y Machine Learning, sino también para el desarrollo de aplicaciones de deep learning, sin tener que invertir en recursos hardware o en la nube.

Con Colab se pueden crear notebooks o importar los que ya tengamos creados, además de compartirlos y exportarlos cuando queramos. Esta fluidez a la hora de manejar la información también es aplicable a las fuentes de datos que usemos en nuestros proyectos (notebooks), de modo que podremos trabajar con información contenida en nuestro propio Google Drive, unidad de almacenamiento local, github e incluso en otros sistemas de almacenamiento en la nube.

Resultados y discusión

Hipótesis

Es muy útil e importante desarrollar algunas hipótesis en trabajos de aprendizaje automático, es necesario entender el problema a detalle entendiendo el enunciado del problema a fondo y antes de mirar los datos, con una lluvia de ideas de tantos factores como sea posible que puedan afectar el resultado

Salario: los solicitantes con altos ingresos deberían tener más posibilidades de aprobación del préstamo.

Historial previo: los solicitantes que hayan pagado sus deudas anteriores deberían tener mayores posibilidades de aprobación del préstamo.

Monto del préstamo: la aprobación del préstamo también debe depender del monto del préstamo. Si el monto del préstamo es menor, las posibilidades de aprobación del préstamo deben ser altas.

Plazo del préstamo: el préstamo por un período de tiempo y una cantidad

menores debería tener mayores posibilidades de aprobación.
EMI (pago mensual estimado/cuota mensual): Cuanto menor sea la cantidad para pagar mensualmente para reembolsar el préstamo, mayores serán las posibilidades de aprobación del préstamo.

El dataset tiene las siguientes características:

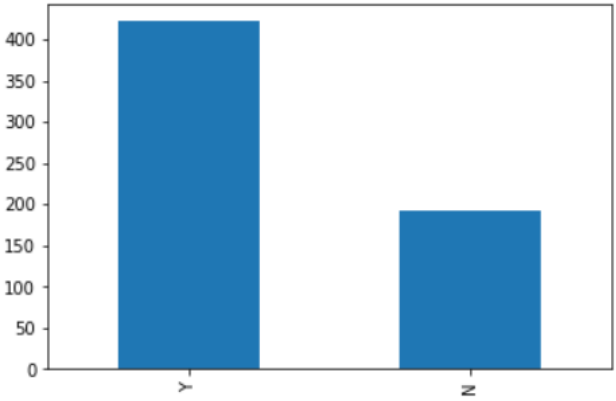
Loan_ID: ID de préstamo único
Gender: Genero (Masculino / Femenino)
Married: Estado civil (S / N)
Dependents: número de dependientes (0, 1, 2, 3+)
Education: Educación del solicitante (graduado / no graduado)
Self_Employed: Trabajadores por cuenta propia (S / N)
ApplicantIncome: Ingresos del solicitante
CoapplicantIncome: Ingresos del garante
LoanAmount: Monto del préstamo en miles de dólares
Loan_Amount_Term: Plazo del préstamo
Credit_History: Historial crediticio (S / N)
Property_Area: área de ubicación del inmueble (Urbano / Semiurbano / Rural)
Loan_Status: Estado del préstamo (S / N) esta es la variable objetivo

Las características de las variables del dataset son:

- Características categóricas: estas características tienen categorías (género, casado, autónomo, historial crediticio, estado del préstamo).
- Características ordinales: variables en características categóricas que tienen algún orden (dependientes, educación, área de propiedad).
- Características numéricas: estas características tienen valores numéricos (Ingreso del solicitante, Ingreso del garante, Monto del prestamo, Plazo del préstamo).

Primero se realiza un análisis unitario de las variables, empezando por la variable objetivo Loan_Status que es categórica.

Fig. 2 Tabla de solicitantes de préstamo aprobados y rechazados

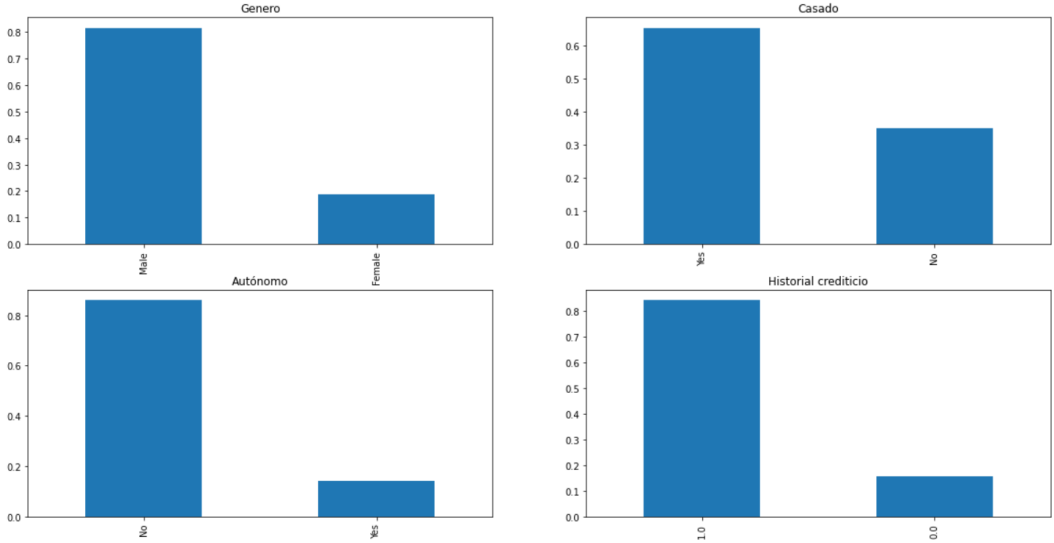


Fuente: Elaboración propia

Donde se puede observar que, casi el doble de solicitantes es aprobado para el crédito.

Analizando las variables independientes categóricas se tiene:

Fig.3 Tablas de Genero, Casado, Autónomo e Historial crediticio del solicitante



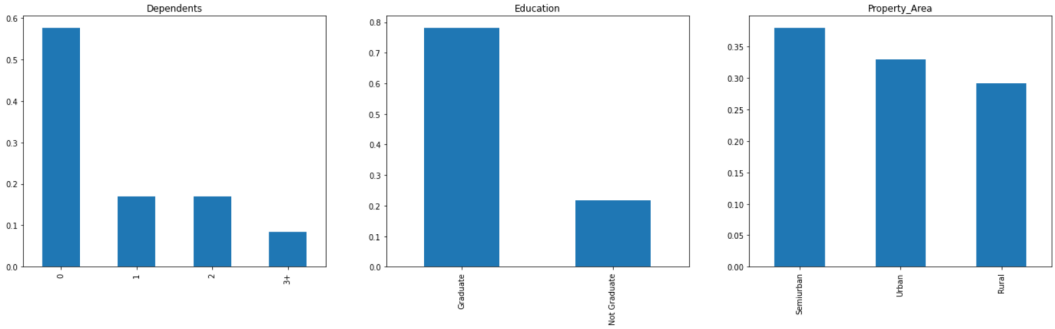
Fuente: Elaboración propia

Se puede ver en los gráficos de barras anteriores que:

- El 80% de los solicitantes son hombres.
- Alrededor del 65% de los solicitantes están casados.
- Alrededor del 15% de los solicitantes son autónomos.
- Alrededor del 85% de los solicitantes tienen historial de crédito.

Analizando las variables independientes ordinales se tiene:

Fig. 4 Tabla de Dependientes, Educación y área de la propiedad del solicitante



Fuente: Elaboración propia

- La mayoría de los solicitantes no tienen dependientes.
- Alrededor del 80% de los solicitantes son graduados.
- La mayoría de los solicitantes son del área semiurbana.

Analizando las variables independientes numéricas que tiene:

- La mayoría de los datos en la distribución de los ingresos de los solicitantes están hacia la izquierda, lo que significa que no se distribuye normalmente.
- El diagrama de caja confirma la presencia de muchos valores atípicos / extremos.
- Lo más probable es que sea por la disparidad de ingresos en la sociedad.
- Con el hecho de que estamos mirando a persona con diferentes niveles educativos.

Analizando la distribución de la variable LoanAmount se tiene:

- Se observa muchos valores atípicos en el diagrama de caja de esta variable.
- Pero la distribución es bastante normal.

El Historial crediticio contra estado de préstamo refleja:

- Que las personas con historial crediticio tienen más probabilidades de obtener la aprobación de sus préstamos visto de otra manera al solicitar un préstamo la primera vez es difícil.

Vemos la correlación entre todas las variables numéricas donde:

- Vemos que las variables más correlacionadas son Ingreso del solicitante con monto del préstamo e historial crediticio con estado del préstamo.

En el tratamiento del dataset:

- Hay menos valores perdidos en las características Sexo, Casado, Dependientes, Historial de crédito y Autónomo, por lo que podemos completarlos usando la moda de las características.
- Usamos la mediana para completar los valores nulos, ya que antes vimos que el monto del préstamo tiene valores atípicos, por lo que la media no será el enfoque adecuado, ya que se ve muy afectada por la presencia de valores atípicos.
- Para manejar variables categóricas, existen otros métodos. como necesitamos convertir todas las variables categóricas en numéricas, se utilizará el método get_dummies.

En la construcción del modelo:

Sklearn requiere la variable de destino en un conjunto de datos separado por lo que eliminamos nuestra variable de destino del conjunto de datos y la guardamos en otro conjunto de datos.

Después se crean variables ficticias para las variables categóricas. La variable ficticia convierte las variables categóricas en una serie de 0 y 1, lo que las hace mucho más fáciles de cuantificar y comparar.

Se entrena el modelo en el conjunto de datos de entrenamiento y se harán predicciones para el conjunto de datos de prueba. Una forma de hacerlo es

dividir el conjunto de datos en dos partes: entrenamiento y validación.

Usando la función train_test_split de sklearn para dividir el conjunto de datos. Con LogisticRegression, fit () y precision_score de sklearn ajustamos el modelo de regresión logística.

Se debe predecir Loan_Status y calcular su precisión con el método predict () y accuracy_score ().

Donde las predicciones son casi un 80% precisas, es decir, se logró identificado correctamente el 80% del estado del préstamo.

Metodología:

Se empleo la metodología cuantitativa con diseño de investigación no experimental en el tipo de estudio correlacional.

Conclusiones:

Hoy en día, el negocio de préstamos se vuelve cada vez más popular y muchas personas solicitan préstamos por diversas razones. Sin embargo, hay casos en los que las personas no devuelven la mayor parte del monto del préstamo al banco, lo que resulta en una gran pérdida financiera. Si se logra clasificar de manera eficiente a los solicitantes por adelantado, evitaría en gran medida la pérdida financiera.

En este estudio, primero se limpió el conjunto de datos y se realizó el análisis de datos exploratorios. Se cubrieron las estrategias para abordar tanto los valores perdidos como los conjuntos de datos desequilibrados.

A partir del análisis exploratorio de datos y evaluación individual de las variables, se puede generar información a partir de los datos y entender cómo se relaciona cada una de las variables individuales con la variable objetivo. También se puede notar que la librería de sklearn nos facilita el

trabajo enormemente al permitir trabajar con hiper parámetros por defecto que por lo general son los más adecuados y en este caso se pudo obtener un resultado aceptable.

Se pudo evaluar que no siempre las hipótesis pueden estar alineadas con el análisis y evaluación de los datos, por ejemplo, en este caso los solicitantes con más ingresos no necesariamente eran los que tenían mayor posibilidad de acceder al préstamo, por otro lado, la variable que más impacto tenía era el historial crediticio del solicitante.

Referencias.

- Amat, J. (2016, 8). *Regresion logistica simple y multiple.* Retrieved from ciencia de datos: https://www.cienciadedatos.net/documentos/27_regresion_logistica_simple_y_multiple
- Borrajo, D. (2006). *Aprendizaje autom´atico.* Madrid.
- Fernandez, S. d. (2011). *Regresión logística.* Madrid.
- Garreta, R. (2013). Learning scikit-learn: Machine Learning in Python.
- Mitchel, T. (1997). Machine Learning. McGraw Hill.
- Murphy, K. (2012). *Machine Learning: A probabilistic perspective.* The MIT Press.
- Sklearn. (2020, 12 2). *Logistic Regression.* Retrieved from scikit-learn: https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html

Artículo recibido: 22-09-2020
Artículo aceptado: 05-11-2020

Artículo Original Pág. 61 - 75 ISSN 2415-2323

ILUMATE VOLUMEN 12

PREDICCIÓN DE LA CLASIFICACIÓN FINAL EN LA LIGA ESPAÑOLA DE FÚTBOL

Prediction of the final classification in the Spanish soccer league

Hugo Herrera Lunario
hugolunario@gmail.com
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle-Bolivia

Ramiro Paucara Ochoa
ramiropaucara@gmail.com
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle-Bolivia

Resumen

En este trabajo se propone el uso de herramientas de machine learning, en particular del algoritmo de regresión logística para intentar predecir la clasificación final del siguiente campeonato de la Liga española de fútbol. El método es experimental de tipo exploratorio, los datos que se utilizaron se armaron en base a datos disponibles en el sitio web de Kaggle, se hizo uso de 3 conjuntos de datos en los cuales se tiene información de los jugadores, el historial de partidos desde 1996 hasta 2020 y el conjunto de equipos que participaran en el siguiente campeonato con su ranking actual de la FIFA. Las principales variables para predecir los resultados fueron el ranking FIFA de cada equipo, el talento de sus jugadores y el historial de

enfrentamientos. Los experimentos realizados sobre el conjunto de datos muestran cuantitativamente que el modelo desarrollado es apropiado para predecir una clasificación de los equipos de la Liga española en el siguiente año.

Palabras claves

Machine learning, regresión logística, Kaggle, ranking.

Abstract

In this work, the use of machine learning tools is proposed, in particular the logistic regression algorithm to try to predict the final classification of the next championship of the Spanish Soccer League. The method is experimental of an exploratory type, the data used was assembled based on data available on the Kaggle website, 3 data sets were used in which there is information on the players, the history of matches since 1996 to 2020 and the set of teams that will participate in the next championship with their current FIFA rankings. The main variables to predict the results were the FIFA ranking of each team, the talent of its players and the history of confrontations. The experiments carried out on the data set show quantitatively that the model developed is appropriate to predict a classification of the teams of the Spanish league in the following year.

Key words

Machine learning, logistic regression, Kaggle, ranking.

Introducción

La liga de fútbol española comúnmente conocida como La Liga es la primera liga nacional de fútbol de España, siendo una de las ligas deportivas profesionales más populares del mundo.

Actualmente, está formado por 20 equipos distribuidos de forma bastante uniforme por todo el país, pero principalmente de las regiones más desarrolladas: Madrid, Barcelona y País Vasco. Los cuatro mejores equipos están clasificados para la Liga de Campeones, mientras que los tres equipos peor ubicados (posiciones 18-20) son relegados a la segunda división.

Las opciones de aplicación de los algoritmos de aprendizaje automático en el fútbol son diversas ya que comprenden el sector de la industria de las apuestas deportivas; el análisis del desempeño de los jugadores que conforman el equipo; la formación táctica del equipo y plan de juego; la cantidad de goles anotados por encuentro hasta el posible ganador antes de que comience el encuentro. (ARIAS, 2019).

Se utilizará la metodología de aprendizaje automático comprendido por los pasos de construcción del conjunto de datos, categorización de los datos, posteriormente un preprocesamiento de la información y limpieza de los datos, luego se realizará la extracción de características acompañadas de validación de datos que permitan la construcción del modelo de predicción que finalmente tendrá como salida una categorización en una tabla de clasificación. (ARIAS, 2019).

El resultado de este experimento basado en aprendizaje automático mediante la técnica de la regresión logística tiene como fin apoyar la toma de decisiones de los equipos del fútbol profesional para generar un fútbol más competitivo, por otra parte, puede ser de utilidad para las empresas patrocinadoras para ver la eficiencia partido tras partido de los equipos de fútbol para promover sus productos y apoyar de forma económica a los equipos del fútbol profesional.

Referentes conceptuales.

Métodos.

Regresión Logística Binaria. La regresión logística es un método estadístico

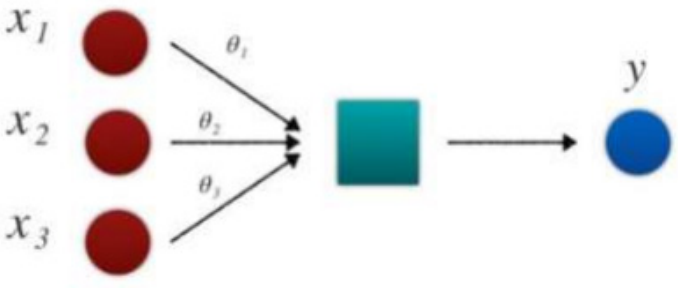
para analizar un conjunto de datos en el que hay una o más variables independientes que determinan un resultado. El resultado se mide con una variable dicotómica (en la que solo hay dos resultados posibles). Se utiliza para predecir un resultado binario dado un conjunto de variables independientes. Para representar el resultado binario o categórico, se utilizan variables ficticias. También la regresión logística se puede expresar como un caso especial de regresión lineal, es decir, cuando la variable de resultado es categórica, donde se está utilizando el registro de probabilidades como variable dependiente. En palabras simples, predice la probabilidad de ocurrencia de un evento al ajustar los datos a una función. (Hernández, 2019)

Este modelo logístico binario según Ponce (2020) se usa para estimar la probabilidad de una respuesta binaria basada en una o más variables (características) predictoras (o independientes). Permite decir que la presencia de un factor de riesgo aumenta la probabilidad de un resultado dado en un porcentaje específico.

Como todos los análisis de regresión, la regresión logística es un análisis predictivo.
La regresión logística se utiliza para describir datos y para explicar la relación entre una variable binaria dependiente y una o más variables independientes nominales, ordinales, de intervalo o de relación. (Gandhi, 2018).

En la siguiente figura se observa un modelo de regresión logística.

Fig. 1. Representación gráfica de un modelo de regresión logística.



Fuente: (ARIAS, 2019)

Esta función de probabilidad es la ' Función sigmoidal' como se observa en la siguiente figura:

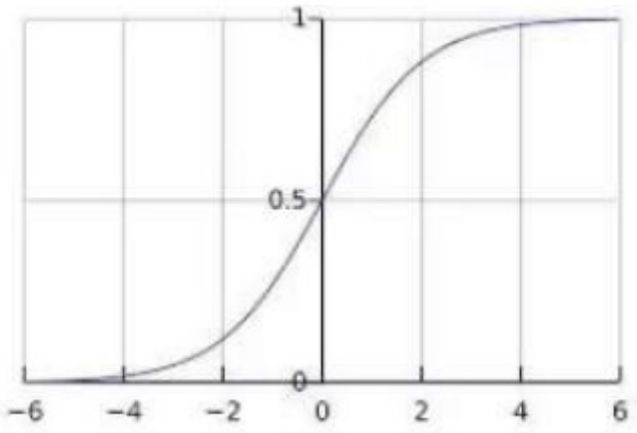
Fig. 2. Función sigmoidal

$$\frac{1}{1 + e^{(-z)}}$$

Fuente: (ARIAS, 2019)

En la siguiente figura se muestra gráficamente la función, se vería de la siguiente forma:

Fig. 3. Función sigmoidal



Fuente: (ARIAS, 2019)

Regresión Logística Multinomial. La regresión logística multinomial es un método de clasificación que generaliza la regresión logística a problemas a problemas multiclase, es decir, con más de dos posibles resultados discretos. Es un modelo que se utiliza para predecir las probabilidades de los diferentes resultados posibles de una variable dependiente distribuida categóricamente, dado un conjunto de variables independientes. La regresión logística multinomial es una solución particular para los problemas de clasificación que utilizan una combinación lineal de las características observadas y algunos parámetros específicos del problema para estimar la probabilidad de cada valor particular de la variable dependiente. (Catalan, 2021).

Al igual que en otras formas de regresión lineal, la regresión logística multinomial utiliza una función predictiva lineal $f(k, i)$ para predecir la probabilidad de que la observación i tenga un resultado i, de la siguiente forma:

Fig. 4. Función de predicción regresión logística multinomial

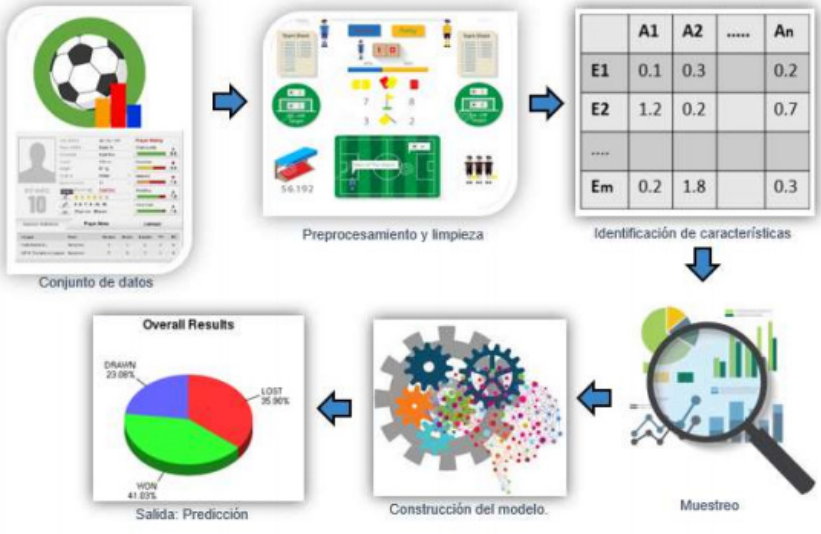
$$f(k,i) = \boldsymbol{\beta}_k \cdot \mathbf{x}_i$$

Fuente: (Catalan, 2021)

donde Xi es el vector de variables explicativas que describen la observación i, βk es un vector de ponderaciones correspondientes al resultado k, y la puntuación (Xi, k) es la puntuación asociada con la asignación de la observación i a la categoría k. En la teoría de la elección discreta, donde las observaciones representan personas y los resultados representan elecciones, la puntuación se considera la utilidad asociada con la persona i que elige el resultado k. El resultado predicho es el que tiene la puntuación más alta.

Para realizar el proceso de predicción de la clasificación final de la liga española de fútbol, es necesario realizar una serie de etapas en donde cada una se complementa directamente con la anterior, de esta forma se describen 6 pasos como se puede observar en la siguiente figura.

Fig. 5. Metodología aplicada predicción de clasificación



Fuente: (ARIAS, 2019)

Conjunto de datos

Se cuenta con información en tres conjuntos de datos:

- Equipos: Este dataset contiene los datos de los jugadores en cada equipo. (Kishan, 2020).
- La Liga_Matches_1995-2020: Este dataset contiene el historial de partidos en que se enfrentaron los equipos que participan de la liga, desde 1996 hasta el 2020. (Kishan, 2020).
- Copa_20_21: Este dataset contiene la composición del total de equipos que participan de la Liga con su ranking FIFA actual. (Tadhg, 2018).

Preprocesamiento y limpieza de datos.

En esta etapa se realiza la selección de datos más importantes para el proceso

de clasificación con el objetivo de conservar solo las características o los atributos más relevantes. (Dominguez, 2019).

Se eliminaron varios campos irrelevantes para esta predicción, como ser: sueldo de jugadores, fecha de vencimiento de contratos, nuero de camiseta, etc. También se relleno datos faltantes con el promedio correspondiente de cada columna.

Identificación de características

Los métodos de selección de características ayudan a reducir las dimensiones sin perder mucho la información total. También ayuda a dar sentido a las características y su importancia. (AED - Asociación Española de Directivos, 2019).

Usamos las características de si un equipo juega en casa o fuera de casa, estas características se transforman en una codificación en una variable discreta para su procesamiento posterior.

Muestreo

En esta etapa se va a clasificar el conjunto de datos en dos de manera que una sea para realizar el entrenamiento y la segunda sea para realizar las pruebas.

Puede haber muchos más datos seleccionados disponibles de los que necesita para trabajar. Más datos pueden resultar en tiempos de ejecución mucho más largos para los algoritmos y mayores requisitos computacionales y de memoria. Se tomó una muestra representativa más pequeña de los datos seleccionados que pueden ser mucho más rápidos para explorar y crear prototipos de soluciones antes de considerar el conjunto de datos completo.

Construcción del modelo de predicción.

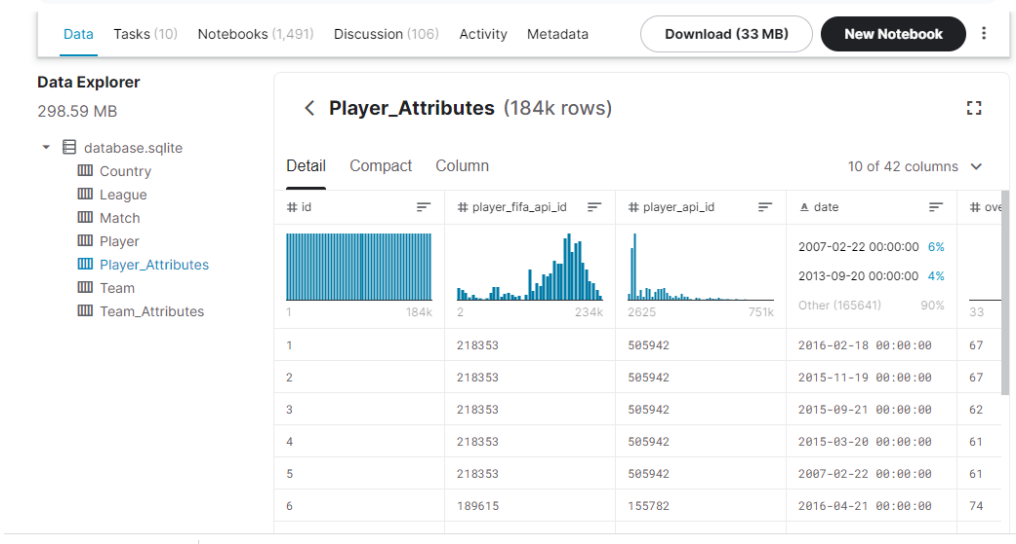
En esta etapa, se aplica los clasificadores de aprendizaje automático necesarios para realizar predicción de la clasificación final de la liga española de fútbol.

Salida.
En esta etapa se va a obtener el resultado que genera el modelo predictivo donde se indica el resultado del modelo desarrollado.

Resultados y discusión

Para construir el conjunto de datos, fue necesario indagar varias fuentes las cuales debían contar con la información relacionada a los partidos de fútbol de la liga española, por otra parte, la información debía ser confiable en su totalidad, finalmente se optó por seleccionar la fuente de datos de la página kaggle.

Fig. 6. Obtencion de informacion del sitio web Kaggle.com



Fuente: Elaboracion Propia

Para el conjunto de datos obtenido de la fase anterior, fue necesario construir la variable overall en el conjunto de datos de los partidos, uno del equipo local, otro para el equipo visitante y un campo adicional de la diferencia entre overall del equipo local y del visitante, en base a la información de los jugadores de cada equipo como se observa en la siguiente ilustración.

Fig. 7. Codigo del modelo

```
# Agregando el pontencial del equipo al dataset de partidos
potencial = equipos.groupby('Club').mean()['Overall']
partidos = partidos.merge(potencial,
                          left_on=['HomeTeam'],
                          right_on=['Club'])
partidos = partidos.merge(potencial,
                          left_on=['AwayTeam'],
                          right_on=['Club'],
                          suffixes=('_local', '_visitante'))

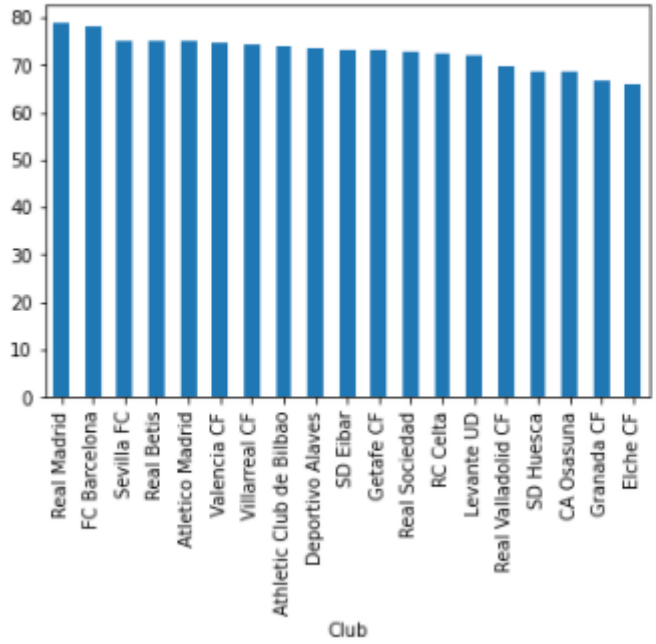
# Agregar diferencia de potencial entre los equipos
partidos['dif_potencial'] = partidos['Overall_local'] - partidos['Overall_visitante']
```

Fuente: Elaboracion Propia

El siguiente análisis, corresponde a la calidad de los jugadores por equipo, como se muestra en la siguiente ilustración, como se puede ver los grandes candidatos de acuerdo a la calidad de sus jugadores son Real Madrid, Barcelona, Sevilla.

Fig. 8. Potencial por equipo

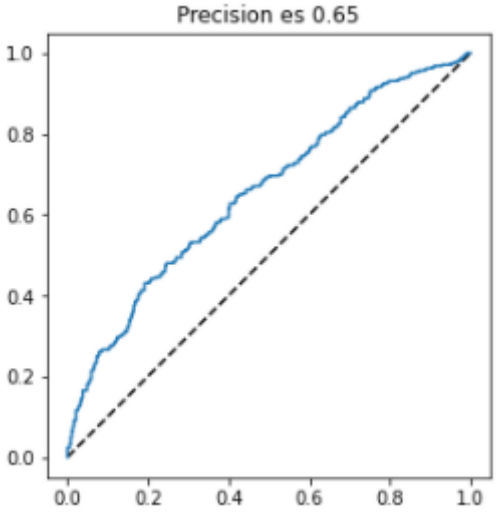
```
plot = potencial.sort_values( ascending=False).plot(kind='bar')
```



Fuente: Elaboracion Propia.

Al ejecutar el algoritmo, se muestra en la ilustración siguiente la precisión del modelo:

Fig. 9. Precision del modelo



Fuente: Elaboracion Propia

Finalmente se asigna una probabilidad de victoria del local o visitante en cada partido según los campos predictores de cada partido, posteriormente se suman los puntos por cada victoria o empate de cada equipo y al final se realiza una sumatoria de los puntos al final de todos los partidos todos contra todos, y el resultado es el siguiente:

Fig. 10. Clasificacion final

	equipo	puntos
0	Real Madrid	54
1	FC Barcelona	51
2	Sevilla FC	48
3	Atletico Madrid	45
4	Valencia CF	36
5	Villarreal CF	33
6	CA Osasuna	33
7	Athletic Club de Bilbao	30
8	Elche CF	30
9	Deportivo Alaves	27
10	Granada CF	24
11	Real Betis	21
12	SD Huesca	21
13	Getafe CF	15
14	Real Valladolid CF	15
15	Levante UD	12
16	RC Celta	9
17	Real Sociedad	6
18	SD Eibar	3

Fuente: Elaboracion Propia.

Conclusiones:

En el transcurso del desarrollo del presente trabajo se identificaron inconvenientes en base a la información del conjunto de datos, la precisión de predicción no subió lo esperado, pues ya que los atributos del conjunto de datos son limitados y no cuentan con características relevantes que en efecto puedan impactar en quien será el ganador de un partido de fútbol, como, por

ejemplo, datos de tipo climatológico, factores emocionales, puntuación del árbitro, factores sociales, etc.

En el presente trabajo de investigación, como se pudo observar la precisión del modelo construido es cercano al 65%, es decir, no existe un criterio claro y definido para predecir la clasificación final de un campeonato en función de la probabilidad que se obtuvo como resultado.

Referencias:

- AED - Asociación Española de Directivos. (2019). *Machine Learning, Inteligencia Artificial y Big Data. Lo que todo directivo debe saber .* Madrid: Accenture.
- ARIAS, E. (2019). *Repositorio Universidad Catolica de Colombia.* Obtenido de https://repository.ucatolica.edu.co/
- Catalan, A. (2021). Predicción y clasificación con Data Science. *Revista de Marina*.
- Dominguez, K. (2019). Inteligencia artificial. *Pro Magazine*, 42-43.
- Gandhi. (2018). *towardsdatascience.* Obtenido de http://www.towardsdatascience.com
- Hernández, A. (2019). *TRES MODELOS PREDICTIVOS DEPOTENCIAL ARQUEOLÓGICO.* Obtenido de https://es.scribd.com/document/411566036/Modelos-Predictivos-de-Potencial-Arqueologico
- Kishan, K. (2020). *kaggle.* Obtenido de https://www.kaggle.com/kishan305/la-liga-results-19952020/version/4
- Ponce, H. (2020). Buscando la integridad academica de la inteligencia artificial. *Integridad academica*, 6-8.
- Tadhg, F. (2018). *kaggle*. Obtenido de https://www.kaggle.com/tadhgfitzgerald/fifa-international-soccer-mens-ranking-1993now

Artículo recibido: 09-09-2020
Artículo Aceptado: 09-11-2020

Artículo Original Pág. 77 - 101 ISSN 2415-2323

ILUMATE VOLUMEN 12

PREDICCIÓN DE SUPERVIVENCIA ANTE LA INSUFICIENCIA CARDIACA

Prediction of Survival in Heart Failure

Choque Forra Daniela Patricia
dpcfjcrc@gmail.com
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle-Bolivia

Mamani Mamani Jessyca Liset
jessycamamani20@gmail.com
Instituto de Investigaciones en Ciencia y Tecnología,
Universidad La Salle-Bolivia

Resumen

Justificación: En visto que actualmente las enfermedades son ya comunes entre nosotros, una que tiene un amplio conjunto es la Insuficiencia Cardiaca, que puede surgir por varias razones y estas aplicar en que tan mortal puede llegar a ser. El objetivo es predecir la supervivencia ante la insuficiencia cardiaca, contando con un dataset que contiene registros clínicos de pacientes con insuficiencia cardiaca. Se utilizó diferentes algoritmos de Aprendizaje Automático de los cuales se seleccionó el mejor modelo para luego utilizarlo en una interfaz gráfica. También se realizó el análisis de datos para observar la influencia que tienen respecto a la Insuficiencia Cardiaca. La metodología empleada fue cuantitativa de diseño exploratorio. Se logró presentar un análisis de cada factor en la insuficiencia cardiaca, tomando puntos como el

género, o algún otro tipo de enfermedad.

Palabras clave: Corazón, insuficiencia, predicción.

Abstract

Justification: Given that currently diseases are already common among us, one that has a wide range is Heart Failure, which can arise for various reasons and these apply to how deadly it can be. The objective is to predict survival in heart failure, counting on a dataset that contains clinical records of patients with heart failure. Different Machine Learning algorithms were used from which the best model was selected and then used in a graphical interface. Data analysis was also carried out to observe the influence they have regarding Heart Failure. The methodology used was quantitative with an exploratory design. An analysis of each factor in heart failure was presented, taking points such as gender, or some other type of disease.

Key words: Heart, failure, prediction.

Introducción

Muchas veces, no es posible saber cuándo puede una persona tener insuficiencia cardiaca, y a medida que avanza el tiempo, esto se hace cada vez más difícil de saber o confundir ciertos síntomas que se pueden producir por este problema.

El proyecto trata sobre la supervivencia ante la insuficiencia cardiaca y la aplicación de modelos de aprendizaje automático para predecir la supervivencia ante la Insuficiencia Cardiaca. Ante ello conoceremos que es la Insuficiencia Cardiaca y que es Aprendizaje Automático.

Insuficiencia Cardiaca

La insuficiencia cardíaca (IC) es un síndrome de disfunción ventricular. La

insuficiencia ventricular izquierda causa disnea y fatiga, mientras que la insuficiencia ventricular derecha promueve la acumulación de líquido en los tejidos periféricos y el abdomen. Los ventrículos pueden verse involucrados en forma conjunta o por separado. El diagnóstico inicial se basa en la evaluación clínica y se confirma con radiografías de tórax, ecocardiografía y medición de las concentraciones plasmáticas de péptido natri uréticos. El tratamiento consiste en la educación del paciente, diuréticos, inhibidores de la enzima convertidor de la angiotensina (ECA), bloqueantes de los receptores de angiotensina II, betabloqueantes, antagonistas de la aldosterona, inhibidores de la neprilisina, marcapasos/desfibriladores implantables y otros dispositivos especializados y la corrección de la causa o las causas del síndrome de IC. (Howlett, 2020).

Causas

La insuficiencia cardíaca casi siempre es una afección prolongada (crónica), pero se puede presentar repentinamente. Puede ser causada por muchos problemas diferentes del corazón.

La enfermedad puede afectar únicamente el lado derecho o el lado izquierdo del corazón. Más frecuentemente, ambos lados del corazón resultan comprometidos. (Allen LA, 2019).

La insuficiencia cardíaca ocurre cuando:

- Su miocardio no puede bombear (expulsar) la sangre del corazón muy bien. Esto se denomina insuficiencia cardíaca sistólica o insuficiencia cardíaca con una fracción de eyección reducida (HFrEF, por sus siglas en inglés). (Allen LA, 2019).
- El miocardio está rígido y no se llena de sangre fácilmente. Esto se denomina insuficiencia cardíaca diastólica o insuficiencia cardíaca con una eyección preservada (HFpEF, por sus siglas en inglés). (Allen LA, 2019).

Referentes Conceptuales

Aprendizaje Automático

El proceso de aprendizaje automático es similar al de la minería de datos. Ambos sistemas buscan entre los datos para encontrar patrones. Sin embargo, en lugar de extraer los datos para la comprensión humana –como es el caso de las aplicaciones de minería de datos– el aprendizaje automático utiliza esos datos para detectar patrones en los datos y ajustar las acciones del programa en consecuencia. Los algoritmos del aprendizaje automático se clasifican a menudo como supervisados o no supervisados. Los algoritmos supervisados pueden aplicar lo que se ha aprendido en el pasado a nuevos datos. Los algoritmos no supervisados pueden extraer inferencias de conjuntos de datos. (Rouse, 2017).

Conociendo ambos conceptos se puede empezar con el objetivo del proyecto.

Descripción de datos

El presente proyecto para la "Predicción de Supervivencia ante la Insuficiencia Cardiaca" utilizó un dataset que contiene registros clínicos de pacientes con insuficiencia cardiaca. Este dataset cuenta con los siguientes datos según Kaggle-Serpro:

- **Age:** La edad del paciente.
- **Anaemia:** Si el paciente tiene anemia.
- **Creatinine_phosphokinase:** El número de Creatinina fosfoquinasa que tiene el paciente.
- **Diabetes:** Si el paciente tiene Diabetes.
- **Ejection_fraction:** La fracción de eyección del corazón del paciente.
- **High_blood_pressure:** Si el paciente tiene hipertensión.
- **Platelets:** El número de plaquetas del paciente.
- **Serum_creatinine:** El nivel de Suero de creatinina en el paciente.
- **Serum_sodium:** El nivel de Sodio sérico en el paciente.

- **Sex:** El género del paciente.
- **Smoking:** Si el paciente fuma.
- **Time:** Días de tratamiento del paciente.
- **DEATH_EVENT:** Si el paciente sobrevive o no.

Análisis de datos

Se importó el Dataset utilizado en este proyecto el cual es "Registros clínicos de insuficiencia cardíaca. Nuestro dataset cuenta con 299 registros clínicos con los datos ya mencionados.

División aleatoria del Dataset al 90%

Se realizó una separación de datos donde se tomará aleatoriamente el 90% de los datos para el análisis y entrenamiento, el 10% será utilizado para prueba.

Comienzo del Análisis de datos

Se inició a analizar el dataset tomando algunos factores para observar su influencia en la predicción de la supervivencia ante la Insuficiencia cardiaca. Se acompañará con gráficos.

La edad y el género un factor importante para predecir la muerte ante IC

En este primer caso se tomó en cuenta la edad y género del paciente para observar si es un factor importante para predecir la muerte ante IC.

Metodología empleada

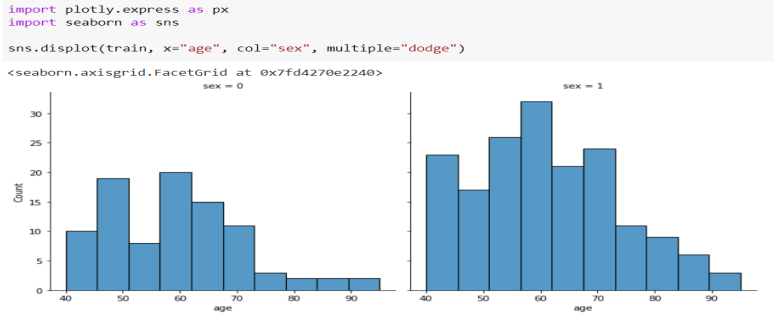
La metodología empleada fue cuantitativa de diseño exploratorio, es un tipo de investigación para estudiar el problema respecto algunas fases. Para obtener nuestros los datos de estudio realizamos un 'Grupo Focal', donde

expusimos lo investigado y llegamos a tomar los puntos más relevantes. Así mismo investigaciones en línea y bibliográfica. Identificando el problema, y proponiendo una herramienta de ayuda.

Para esto empleamos 4 modelos de aprendizaje KNN, arboles de decisión, regresión lineal, SVM y Random forest, estos nos ayudaran a determinar la predicción. Todo fue desarrollado en Python y sincrónicamente en Google Colab, usando librerías como Pandas, NumPy, SKLearn y otros.

Resultados Hallados

Fig 1: Análisis de comparación en población entre hombres y mujeres



Fuente: Elaboración Propia

Resultados: Según el grafico se pudo observar que se tienen más pacientes entre 50 a 72 años Se observa que se tiene más pacientes hombres. Lo cual indica que tienden a padecer IC.

Análisis de porcentaje de Supervivencia según el género

Se selecciona el género y sexo. Se le asigna a cada uno el evento de muerte para determinar cuántos hombres y mujeres sobreviven o no sobreviven ante la IC.

Fig. 2: Análisis de Supervivencia entre Hombres y Mujeres



Fuente: Elaboración Propia

Resultados: Según el grafico se puede observar que:

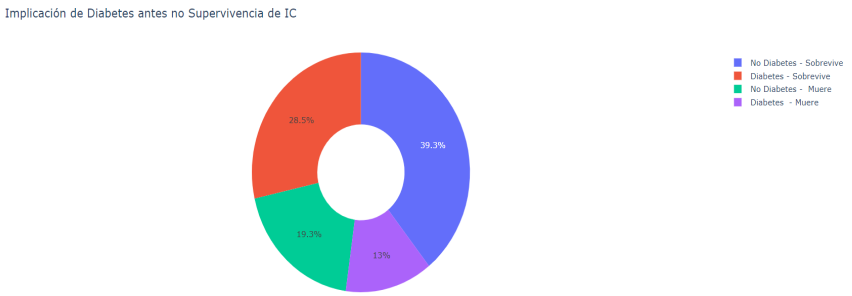
- El 22.7% de las mujeres sobreviven ante la IC.
- El 12.1% de las mujeres no sobreviven ante la IC.
- El 44.3% de los hombres sobreviven ante la IC.
- El 20.8% de los hombres no sobreviven ante la IC.

Probabilidad de una persona con diabetes o sin diabetes sobrevive ante la IC

La diabetes está estrechamente asociada a la insuficiencia cardíaca y, de hecho, se estima que cerca del 40% de los pacientes con insuficiencia cardiaca son diabéticos. A su vez, la diabetes acelera la evolución de esta enfermedad cardiaca. (prnoticias, 2016).

Se selecciona los datos donde una persona con diabetes o sin diabetes sobrevive o no sobrevive ante la IC.

Fig.3: Análisis de Probabilidad de Diabetes



Fuente: Elaboración Propia

Resultados: Según el grafico se puede observar que:

- El 39.3% de personas que no tienen diabetes sobreviven ante la IC.
- El 28.5% de personas que tienen diabetes sobreviven ante la IC.
- El 19.3% de personas que no tienen diabetes no sobreviven ante la IC.
- El 20.8% de personas que tienen diabetes no sobreviven ante la IC.

Probabilidad de que una persona con un bajo porcentaje de sangre que sale del corazón en cada contracción no sobreviva ante la IC

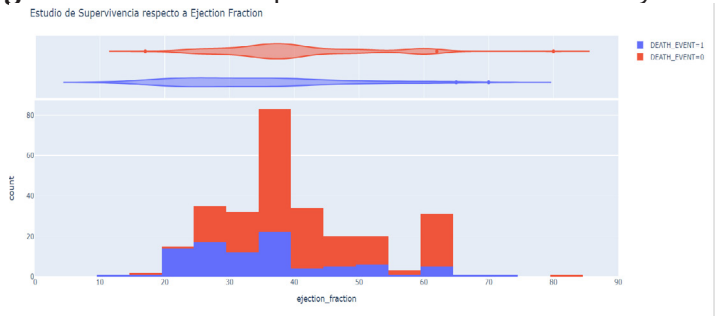
La fracción de eyección es una medida que puede medir la salud del corazón. Un número bajo de fracción de eyección puede ser un indicador de insuficiencia cardíaca. (Mankad, 2019).

Números de fracción de eyección:

- 50 a 70%: función cardíaca normal
- 40 a 55%: función cardíaca por debajo de lo normal. Puede indicar daño cardíaco previo por ataque cardíaco o miocardiopatía.
- Más del 75%: puede indicar una afección cardíaca como la miocardiopatía hipertrófica, una causa común de paro cardíaco repentino.
- Menos del 40%: puede confirmar el diagnóstico de insuficiencia cardíaca.

Se selecciona el dato de fracción de eyección y se lo relaciona con el dato de evento de muerte.

Fig. 4: Análisis entre poblaciones de factor de eyección



Fuente: Elaboración Propia

Resultados: Se puede observar que una mayoría se encuentra debajo de lo normal. Y a causa de que un porcentaje que está debajo de lo normal no han sobrevivido ante la IC.

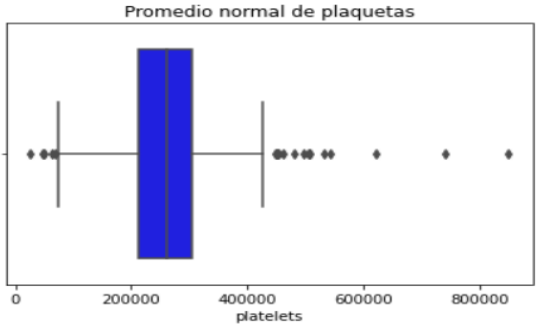
El desbalance en el promedio normal de plaquetas una razón para determinar la Supervivencia ante IC

Las plaquetas, son células sanguíneas producidas en la médula ósea y que son responsables por el proceso de coagulación sanguínea. La cantidad normal de plaquetas en la sangre es de 150,000 a 450,000 por micro litro (mcL) o 150 a 400 × 109/L. Los rangos de los valores normales pueden variar ligeramente.

Se selecciona el dato de número de plaquetas en la sangre para observar si los pacientes se encuentran o no dentro de lo normal. (AEAL, 2017).

Fig. 5: Análisis del promedio normal de plaquetas

```
sns.boxplot(x = train.platelets, color = 'blue')
plt.title("Promedio normal de plaquetas")
plt.show()
```



Fuente: Elaboración Propia

Resultados: Observamos que una mayoría se encuentra dentro de lo normal en el número de plaquetas en la sangre y se encuentran pocos valores atípicos.

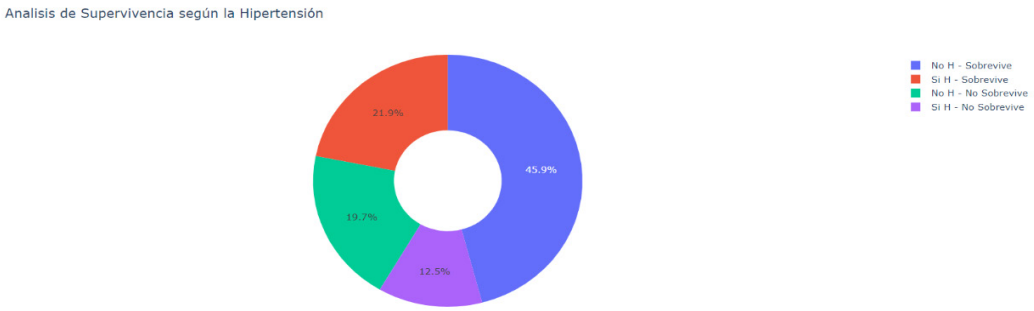
La hipertensión un factor importante ante la supervivencia de IC

El corazón es el órgano encargado de bombear la sangre oxigenada a través de las arterias hacia todo el organismo. Al avanzar, la sangre ejerce una presión contra las paredes de las arterias, que se mide como presión arterial.

La hipertensión arterial se define por la detección de promedios de la presión arterial sistólica (“máxima”) y/o diastólica (“mínima”) por encima de los límites establecidos como normales. Dicho límite es de 140 mmHg para la sistólica y de 90 mmHg para la diastólica.

Se selecciona los datos donde una persona con hipertensión o sin hipertensión sobrevive o no sobrevive ante la IC (Universitario, 2018).

Fig. 6: Análisis de Supervivencia según la hipertensión



Fuente: Elaboración Propia

Resultados: Según el grafico se puede observar que:

- El 45.9% de personas que no tienen hipertensión sobreviven ante la IC.
- El 21.9% de personas que tienen hipertensión sobreviven ante la IC.
- El 19.7% de personas que no tienen hipertensión no sobreviven ante la IC.
- El 12.5% de personas que tienen hipertensión no sobreviven ante la IC.

El alto nivel de enzimas un factor para la No Supervivencia ante la IC

Las enzimas son proteínas complejas que producen un cambio químico específico en todas las partes del cuerpo. Por ejemplo, pueden ayudar a descomponer los alimentos que consumimos para que el cuerpo los pueda usar. La coagulación de la sangre es otro ejemplo del trabajo de las enzimas.

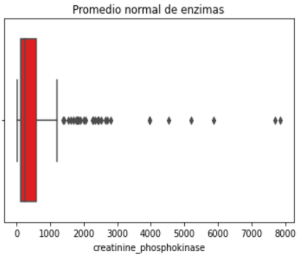
Normalmente se encuentran niveles bajos de estas proteínas y enzimas en la sangre, pero si el músculo cardíaco está lesionado, como por un ataque cardíaco, las proteínas y enzimas se escapan de las células dañadas del músculo cardíaco y aumentan sus niveles en el torrente sanguíneo.

Valores normales de CPK total: de 10 a 120 microgramos por litro (mcg/L). (J., 2019).

Se selecciona el dato del nivel de encimas para observar si los pacientes se encuentran o no dentro de lo normal.

Fig. 7: Promedio normal de Enzinas

```
sns.boxplot(x = train.creatinine_phosphokinase, color = 'red')
plt.title("Promedio normal de enzimas")
plt.show()
```



Fuente: Elaboración Propia

Resultados: Según la gráfica observamos que una mayoría se encuentra dentro de lo normal en el nivel de enzimas y se encuentran pocos valores atípicos.

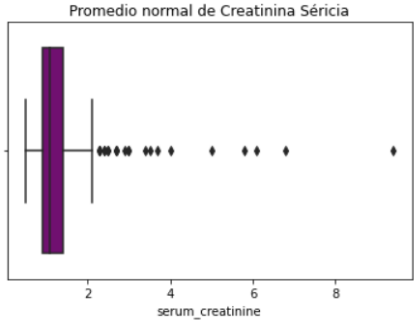
El alto nivel de creatinina sérica un factor para la No Supervivencia ante la IC

La creatinina es un producto de desecho generado por los músculos como parte de la actividad diaria. Normalmente, los riñones filtran la creatinina de la sangre y la expulsan del cuerpo por la orina.

Un resultado normal de nivel de creatinina sérica es de 0.7 a 1.3 mg/dL (de 61.9 a 114.9 µmol/L) para los hombres y de 0.6 a 1.1 mg/dL (de 53 a 97.2 µmol/L) para las mujeres. Las mujeres con frecuencia tienen niveles de creatinina más bajos que los hombres. Esto se debe a que ellas frecuentemente tienen menor masa muscular. (Hinkle J, 2014).

Se selecciona el dato del nivel de creatinina sérica para observar si los pacientes se encuentran o no dentro de lo normal.

Fig. 8: Promedio normal de Creatina sérica



Fuente: Elaboración Propia

Resultados: Según la gráfica observamos que una mayoría se encuentra dentro de lo normal en el nivel de creatinina sérica y se encuentran pocos valores atípicos.

Algoritmo de Predicción

En el presente proyecto se utilizará 5 algoritmos de predicción y se identificara cual es más preciso para la predicción ante la IC.

Entrenamiento de Datos para definir la Supervivencia de un Paciente con IC
En primer lugar, se importará las librerías que utilizaremos como:

- **Pandas:** Es una biblioteca de software escrita como extensión de NumPy para manipulación y análisis de datos para el lenguaje de programación Python. En particular, ofrece estructuras de datos y operaciones para manipular tablas numéricas y series temporales.
- **Numpy:** Es una biblioteca para el lenguaje de programación Python que da soporte para crear vectores y matrices grandes multidimensionales, junto con una gran colección de funciones matemáticas de alto nivel para operar con ellas.
- **Matplotlib:** Es una biblioteca para la generación de gráficos a partir de datos contenidos en listas o arrays en el lenguaje de programación

Python y su extensión matemática NumPy. Proporciona una API, pylab, diseñada para recordar a la de MATLAB.

También se importará el dataset que contiene los registros clínicos de personas que tiene IC.

Datos seleccionados para el entrenamiento

Se seleccionarán los siguientes datos para realizar nuestro entrenamiento ya que sé que se consideran importantes.

- **Age:** La edad, ya que se tienen más pacientes de la tercera edad.
- **Sex:** El sexo ya que se observó según el análisis de datos que hay diferencia de sobrevivir ante la IC entre hombres y mujeres.
- **Creatinine_phosphokinase:** El nivel de enzimas más alto de lo normal, podría significar que tiene inflamación del músculo cardíaco (miocardio) o que está teniendo o ha tenido recientemente un ataque al corazón.
- **Ejection_fraction:** Una medición de la fracción de eyección inferior al 40% puede ser evidencia de insuficiencia cardíaca o miocardiopatía.
- **High_blood_pressure**: La hipertensión supone una mayor resistencia para el corazón, que responde aumentando su masa muscular (hipertrofia ventricular izquierda). A la larga, está hipertrofia patológica (al contrario de la hipertrofia que se produce con el ejercicio) acaba siendo perjudicial, pudiendo producir angina, arritmias e insuficiencia cardíaca. (Torres, 2017).
- **Platelets:** Ya que tener un numero fuera de lo normal de plaquetas en la sangre puede producir un ataque cardiaco.
- **Serum_creatinine:** Un nivel alto de creatinina puede ser a causa de que un paciente tiene hipertensión.
- **Time:** Los días de tratamiento son importantes ya que a menores días de tratamiento es muy probable que el paciente no sobreviva, mientras que a mayores días de tratamiento es más probable que el paciente si sobreviva.

Importación de los módulos y división aleatoria del Dataset al 80%

Se importa los diferentes módulos que utilizaremos de la librería sklearn.

Scikit-learn es la librería más útil para Machine Learning en Python, proporciona algoritmos de aprendizaje supervisados y no supervisados.

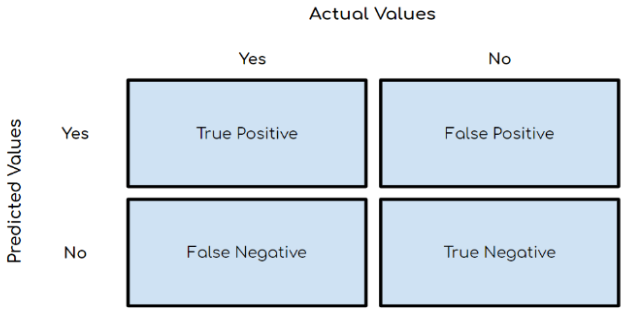
Aquí tomaremos el conjunto del dataset dividido al 80% de los datos donde el 20% serán los datos de prueba, es decir, son aquellos datos que no vio el algoritmo de predicción.

Se seleccionan los datos que tomaremos para el entrenamiento del dataset Se utiliza el train_test_split. La función train_test_split nos permite dividir un dataset en dos bloques, típicamente bloques destinados al entrenamiento y validación del modelo (llamemos a estos bloques “bloque de entrenamiento “ y “bloque de pruebas”).

En estos modelos se utilizará la función Accuracy_score y una matriz de confusión

- **Accuracy score:** Es la exactitud (accuracy) mide el porcentaje de casos que el modelo ha acertado.
- **Matriz de confusión:** La matriz de confusión es una herramienta muy útil para valorar cómo de bueno es un modelo clasificación basado en aprendizaje automático. En particular, sirve para mostrar de forma explícita cuándo una clase es confundida con otra, lo cual nos, permite trabajar de forma separada con distintos tipos de error.

Fig. 9: Matriz de confusión



Fuente: Elaboración Propia

Modo de uso de cada modelo

Datos de entrenamiento para cada modelo

- x_train serán los datos de entrada.
- y_train los de salida.

Se crea una instancia del modelo

Ejemplo:

- reg_log = LogisticRegression ()

Se entrena el modelo con los datos de entrenamiento.

Ejemplo:

- reg_log.fit(x_train,y_train)

Se usa el modelo entrenado para obtener las predicciones con datos nuevos.

Ejemplo:

- reg_log_pred = reg_log.predict(x_test)

Se almacena el porcentaje de exactitud del modelo.

Ejemplo:

- accuracy_score(y_test, reg_log_pred)

Se pasa el porcentaje de exactitud del modelo y la predicción para graficar su matriz de confusión.

Ejemplo:

- cm = confusión_matrix(y_test, reg_log_pred)

Modelo de Regresión Logística

La regresión logística es una técnica de aprendizaje supervisado para clasificación. Es muy usada en muchas industrias debido a su escalabilidad y explicabilidad. (Martinez, 2020).

Utilizando este modelo, su respectiva matriz de confusión nos dio como resultado los siguientes datos:

- 40 verdaderos positivos.
- 3 falsos positivos.
- 5 falsos negativos.
- 12 verdaderos negativos.

Modelo de KNN

K-Nearest-Neighbor es un algoritmo basado en instancia de tipo supervisado de Machine Learning. Puede usarse para clasificar nuevas muestras (valores discretos) o para predecir (regresión, valores continuos). Al ser un método sencillo, es ideal para introducirse en el mundo del Aprendizaje Automático. Sirve esencialmente para clasificar valores buscando los puntos de datos "más similares" (por cercanía) aprendidos en la etapa de entrenamiento (ver 7 pasos para crear tu ML) y haciendo conjeturas de nuevos puntos basado en

esa clasificación. (Na8, 2018)

Utilizando este modelo, su respectiva matriz de confusión nos dio como resultado los siguientes datos:

- 36 verdaderos positivos.
- 7 falsos positivos.
- 13 falsos negativos.
- 4 verdaderos negativos.

Modelo de Arboles de Decisión

Según Ferero (2020), son un tipo de algoritmos de aprendizaje supervisado (i.e., existe una variable objetivo predefinida).

Principalmente usados en problemas de clasificación.

Las variables de entrada y salida pueden ser categóricas o continuas.

Divide el espacio de predictores (variables independientes) en regiones distintas y no sobrepuestas.

Parámetros de Árbol de decisión:

- **max_leaf_nodes:** Los mejores nodos se definen como una reducción relativa de la impureza.
- **random_state:** Controla la aleatoriedad del estimador. Las características siempre se permutan aleatoriamente en cada división.
- **criterion:** La función para medir la calidad de una división. Los criterios admitidos son “gini” para la impureza de Gini y “entropía” para la ganancia de información. En nuestro caso utilizaremos la entropía.

Utilizando este modelo, su respectiva matriz de confusión nos dio como resultado los siguientes datos:

- 43 verdaderos positivos.
- 0 falsos positivos.
- 6 falsos negativos.
- 11 verdaderos negativos.

Soporte de Maquina Vectorial SVM

En el aprendizaje automático, las máquinas de vectores de soporte (SVM, también redes de vectores de soporte) son modelos de aprendizaje supervisado con algoritmos de aprendizaje asociados que analizan los datos utilizados para el análisis de clasificación y regresión.

Las SVM se pueden utilizar para resolver varios problemas del mundo real:

- Las SVM son útiles en la categorización de texto e hipertexto, ya que su aplicación puede reducir significativamente la necesidad de instancias de entrenamiento etiquetadas tanto en la configuración estándar inductiva como transductiva.
- La clasificación de imágenes también se puede realizar utilizando SVM.
- Clasificación de datos satelitales como datos SAR utilizando SVM supervisado.
- Los caracteres escritos a mano se pueden reconocer utilizando SVM.
- El algoritmo SVM se ha aplicado ampliamente en las ciencias biológicas y otras.

Utilizando este modelo, su respectiva matriz de confusión nos dio como resultado los siguientes datos:

- 43 verdaderos positivos.
- 0 falsos positivos.
- 17 falsos negativos.
- 0 verdaderos negativos.

Random Forest

Un modelo Random Forest está formado por un conjunto de árboles de decisión individuales, cada uno entrenado con una muestra ligeramente distinta de los datos de entrenamiento generada mediante bootstrapping). La predicción de una nueva observación se obtiene agregando las predicciones de todos los árboles individuales que forman el modelo.

Parámetros de Random Forest:

- **max_features:** La cantidad de características que se deben considerar al buscar la mejor división:
  - Si es int, considere las max_featurescaracterísticas en cada división.
  - Si es flotante, entonces max_features una fracción y las características se consideran en cada división.int (max_features * n_features)
  - Si es "automático", entonces max_features=sqrt(n_features).
  - Si es "sqrt", entonces max_features=sqrt(n_features)(igual que "auto").
  - Si es "log2", entonces max_features=log2(n_features).
  - Si es Ninguno, entonces max_features=n_features.
- **max_depth:** La profundidad máxima del árbol. Si es None, los nodos se expanden hasta que todas las hojas sean puras o hasta que todas las hojas contengan menos de min_samples_split muestras.
- **random_state:** Controla tanto la aleatoriedad

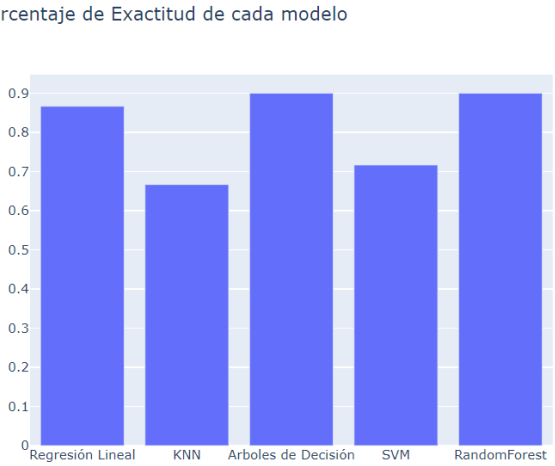
Utilizando este modelo, su respectiva matriz de confusión nos dio como resultado los siguientes datos:

- 40 verdaderos positivos.
- 3 falsos positivos.
- 3 falsos negativos.
- 14 verdaderos negativos.

Evaluación de exactitud de los modelos

Se realiza un gráfico de barras y se observa que modelo tiene más precisión según su porcentaje.

Fig. 10: Gráfico según exactitud de cada modelo



Fuente: Elaboración Propia

- **Regresión Lineal:** 87% de precisión.
- **KNN:** 67% de precisión.
- **Arboles de Decisión:** 90% de precisión.
- **SVM:** 72% de precisión.
- **Random Forest:** 90% de precisión.

Se puede observar que el modelo de Arboles de Decisión y el modelo Random Forest tienen el mismo porcentaje de precisión, pero gracias a la matriz de confusión de cada uno podemos identificar cual es el mejor.

En la matriz de confusión del modelo de Arboles de decisión se tiene 0 falsos positivos mientras que en la matriz de confusión de Random Forest se tiene 3 falsos positivos, entonces podemos determinar que el mejor modelo es el de

Arboles de decisión. Ya que tener falsos positivos nos trae peligros.

Almacenamiento del mejor modelo

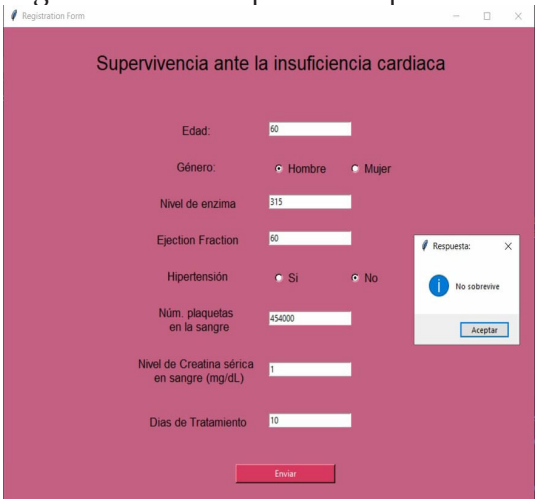
Ya que se ha obtenido el mejor modelo de predicción para nuestro caso se lo almacena, para utilizarlo en una interfaz gráfica.

Interfaz Grafica

Se cuenta con una interfaz gráfica para ingresar los datos de prueba y este nos indique el evento de muerte.

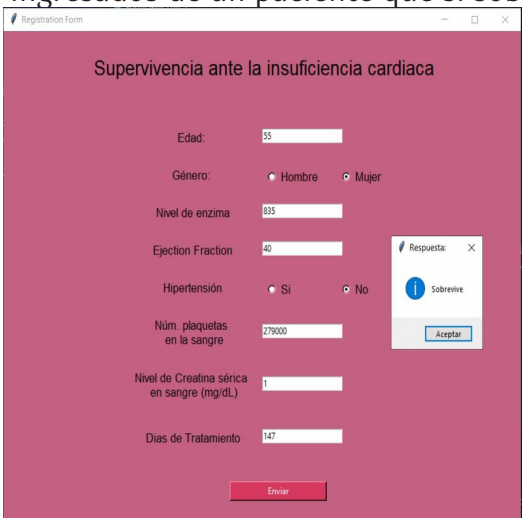
Interfaz

Fig. 11: Datos ingresados de un paciente que no sobrevivió a la IC



Fuente: Elaboración Propia

Fig. 12: Datos ingresados de un paciente que si sobrevivió a la IC



Fuente: Elaboración Propia

Conclusión

Como conclusión se obtiene que gracias a los algoritmos de Machine Learning se puede llegar a predecir si una persona con insuficiencia cardiaca puede o no sobrevivir. Y esto puede se puede aplicar en el mundo real lo cual sería de gran ayuda para hospitales que traten estas enfermedades.

Utilizar Machine Learning de manera adecuada puede llegar a ser muy beneficiosa para diferentes áreas del mundo real.

En base al objetivo podemos demostrar que varios factores no son letales ante la IC, pero factores básicos que afectan están entre la edad y el género. El cual comprobando datos ya existentes da una correcta predicción , con un 90% de exactitud por Random forest.

Enlace del Proyecto

El proyecto se lo puede encontrar en GitHub:
https://github.com/jessie4242/ProyectoAprendizajeAutomatico_InsuficienciaCardiaca

Referencias

AEAL. (20 de Febrero de 2017). *AEAL*. Obtenido de AEAL: http://www.aeal.es/leucemia-mieloide-cronica-espana/1-la-sangre-y-la-medula-osea-normales/

Allen LA, S. L. (2019). Management of patients with cardiovascular disease approaching end of life. En B. E. Tomaselli GF, *Braunwald's Heart Disease: A Textbook of Cardiovascular Medicine.* (pág. 31). Philadelphia.

Ferrero, R. (18 de Agosto de 2020). *Maxima Información*. Obtenido de Maxima Información: https://www.maximaformacion.es/blog-dat/que-son-los-arboles-de-decision-y-para-que-sirven/

Hinkle J, C. K. (28 de Agosto de 2014). *MedlinePlus*. Obtenido de MedlinePlus: https://medlineplus.gov/spanish/pruebas-de-laboratorio/prueba-de-creatinina/

Howlett, J. G. (Febrero de 2020). *MANUAL MSD*. Obtenido de MANUAL MSD Versión para profesionales: https://www.msdmanuals.com/es/professional/trastornos-cardiovasculares/insuficiencia-card%C3%ADaca/insuficiencia-card%C3%ADaca-ic

J., V. L. (2 de Julio de 2019). *MedlinePlus.* Obtenido de MedlinePlus: https://medlineplus.gov/spanish/ency/article/002353.htm#:~:text=Las%20enzimas%20son%20prote%C3%ADnas%20complejas,del%20trabajo%20de%20las%20enzimas.

Mankad, R. (14 de Septiembre de 2019). *Mayo Clinic*. Obtenido de Mayo Clinic org: https://www.mayoclinic.org/es-es/ejection-fraction/expert-answers/faq-20058286

Martinez, J. (2 de Septiembre de 2020). *IArtificial*. Obtenido de IArtificial.net: https://www.iartificial.net/como-usar-regresion-logistica-en-

python/

Na8. (10 de Julio de 2018). *Aprende Machine Learning*. Obtenido de Aprende Machine Learning: https://www.aprendemachinelearning.com/clasificar-con-k-nearest-neighbor-ejemplo-en-python/

Prnoticias, R. (21 de Abril de 2016). *prnoticias*. Obtenido de prnoticias: https://historico.prnoticias.com/salud/sala-de-prensa-pr-salud/20151837-diabetes-e-insuficiencia-cardiaca-una-relacion-bidireccional#inline-auto1611

Rouse, M. (Enero de 2017). *Search Datacenter*. Obtenido de Search Datacenter Tech Target: https://searchdatacenter.techtarget.com/es/definicion/Aprendizaje-automatico-machine-learning

Torres, F. (12 de Abril de 2017). *HC marbella*. Obtenido de HC marbella: https://www.hcmarbella.com/es/hipertension-que-es-y-como-nos-afecta/

Universitario, S. d. (17 de Mayo de 2018). Hospital Universitario. Obtenido de Hospital Universitario: http://hospital.uncuyo.edu.ar/la-importancia-de-prevenir-y-tratar-la-hipertension-arterial

Artículo recibido: 06-09-2020
Artículo Aceptado: 05-11-2020

Misión

Nuestra Misión es publicar, divulgar trabajos de investigación generados en el ámbito académico de Las Ciencias de la Computación a nivel nacional e internacional, elaborado principalmente por estudiantes e investigadores para aportar a la producción científica de nuestra universidad y de nuestro país.

Política Editorial

Iluminate, es una publicación editada por la Carrera de Ingeniería de Sistemas de la Universidad La Salle de Bolivia, con **frecuencia de publicación anual**, busca divulgar trabajos de investigación del area de las ciencias de la computación generados en el ambiente Universitario por estudiantes y en entornos colaborativos con otras universidades. Se reciben contribuciones en español e inglés. El Artículo candidato a publicación debe cumplir con las normas que aparecen en las instrucciones para los autores. Luego de su recepción, el artículo se somete a evaluación por pares, los que recomiendan su aceptación o rechazo.

La revista solicita a los autores con preferencia remitir artículos originales u originales cortos, artículos inéditos en español o inglés con carácter científico, que serán valorados por el comité editor.

La revista se divulga en forma impresa y electrónica, esta ultima para ampliar el espectro de difusión, nos interesa que profesionales, pequeñas, medianas y grandes empresas, organismos, tomadores de decisiones, otros investigadores, docentes y estudiantes universitarios, puedan acceder a sus páginas y establecer referencias para potenciar el desarrollo local y regional.

Licenciamiento

Los artículos que sean seleccionados para formar parte de la revista deberán ser autorizados por su autores o autores, para su difusión a través de cualquier medio, mediante formato establecido para dicho fin. Los artículos de la revista, permanecen en acceso abierto disponibles en la página web institucional como en los diferentes índices y bases de datos a los cuales la revista está suscrita.

Instrucciones para la presentación de Artículos

1. La entrega de artículos y afiliación

Los artículos, serán recibidos y registrados en Dirección Académica de la Universidad La Salle:

Editor en Jefe - Revista Iluminate
wpenafiel@ulasalle.edu.bo;
Instituto de Investigaciones, Universidad La Salle
Calle Jorge Carrasco, esquina Las Palmas, No. 450, zona Bolognia- La Paz, Bolivia
Tel: 591-2-2723588 - 2723598 / Fax: 591-2-2723588
http://www.ulasalle.edu.bo/Iluminate

La universidad entregará, en este caso, acuse de recepción por la misma vía.

Se entregarán en formato digital en una versión actualizada de Word. Las fechas de recepción

y aceptación del artículo son importantes, y deberán ocupar un lugar en el artículo de acuerdo con el formato para la presentación de artículos.

En caso de ser el primer aporte de la revista, el autor deberá presentar en formato digital, nombres y apellidos completos, grado académico, lugar y fecha de nacimiento, lugar de trabajo actual, cargo actual y dirección correo electrónico, para su respectiva afiliación.

Los autores publicados en Iluminate se afiliarán automáticamente a la publicación para recibir periódicamente la revista en formato impreso.

2. Proceso de arbitraje

Una vez recibido el artículo y que se ha comprobado que cumple con los requisitos de forma indicados en las instrucciones para el autor, se acusará recibo mediante correo electrónico y se iniciará el proceso de selección.

Los revisores evaluadores serán designados por el Comité Editorial. Se empleará la metodología "revisores ciegos", quienes evaluarán los aspectos propios de la especialidad tratada.

Los artículos recibidos y admitidos por el Comité Editorial serán enviados, con la información de los autores a dos (2) evaluadores de reconocido prestigio en el área y materia del artículo remitido, también se remitirá el formulario de evaluación.

Los evaluadores tendrán cuatro semanas para revisar el artículo y entregar el formulario de evaluación lleno. Los árbitros evaluadores pueden sugerir modificaciones al artículo original que deben ser realizados por el autor para la aceptación de su artículo. El autor tiene (15) días hábiles para entregar el artículo reformulado. En caso de existir diferencia de opinión entre los dos árbitros evaluadores, se procederá a enviar el artículo a un tercer evaluador o al Editor que dirimirá la controversia.

La evaluación negativa se podrá justificar en los siguientes casos:

a) La clasificación propuesta no concuerda con el desarrollo del artículo.
b) Si presenta deficiencias en la redacción o el desarrollo científico.
c) Si la información contenida no respeta los derechos de autor.
d) Si el artículo no corresponde al eje temático de la revista.

3. Conflicto de Interés

Iluminate aplicará mecanismos internos de resguardo y cautela, para evitar conflicto de interés que afecte a la objetividad en todo el proceso editorial y que involucren al comité editorial y a los evaluadores anónimos en el proceso de arbitraje.

4. Aceptación de artículos

Una vez concluida la etapa de evaluación, será comunicada al autor la decisión sobre la aceptación o rechazo del artículo por parte del Comité Editorial.

El artículo deberá presentar la fecha de entrega y aceptación, éstas serán impresas de acuerdo con las instrucciones para los autores. Adicionalmente el autor deberá llenar el formulario de autorización para la publicación del artículo o revista escrita y/o digital

El comité editorial, declara que las opiniones y comentarios expresados en los artículos son de responsabilidad exclusiva de sus autores.

La revista Iluminate comunicará al autor, el informe final de evaluación. Dicha Comunicación tendrá lugar en un plazo máximo de seis (6) meses, a contar desde la fecha de recepción del artículo.

5. Publicación

El artículo aceptado será publicado en el siguiente número de revista disponible de espacio

para el efecto.

6. Afiliación de Evaluadores

Un árbitro evaluador será parte de la base de datos de Iluminate al momento de ser nombrado por el comité editorial y haber aceptado la primera solicitud de evaluación de un artículo. Para su registros será necesario ingresar el curriculum vitae del evaluador con los siguientes datos: nombres y apellidos completos, lugar de estudio, y dirección correo electrónico.

7. Conformación del Comité Editorial

El comité editorial será conformado por disposición de Consejo de la carrera de Ing. de Sistemas de la Universidad La Salle en Bolivia, nombrando para el efecto los siguientes estamentos; Editor, Editor responsable, Consejo editorial, Consejo de Arbitraje, Colaboradores, Diseño y Diagramación.

El Consejo Editorial aprobará la incorporación de nuevos evaluadores.

8. Compromisos y Responsabilidades éticas

Cada autor presentará una declaración responsable de autoría y originalidad respecto de los siguientes aspectos:

- Inédito. No ce acepta material previamente publicado. Los autores son responsables de obtener los oportunos permisos para reproducir parcialmente (texto, tablas o figuras) de otras publicaciones y de citar su procedencia correctamente. Las opiniones expresadas en los artículos publicados son responsabilidad del autor/es
- Autoría. En la lista de autores firmantes deben figurar únicamente aquellas personas que han contribuido intelectualmente al desarrollo del trabajo. Haber colaborado en la recolección de datos no es, por si mismo, criterio suficiente de autoría. Iluminate declina cualquier responsabilidad sobre posibles conflictos derivados de la autoría de los trabajos que se publiquen.
- Consentimiento informado. Los autores deben mencionar que los procedimientos empleados en sujetos de experimentación han sido realizados con consentimiento informado.

Instrucciones para los autores

Todos los artículos serán publicados en español, con título en español e inglés, con resúmenes en español e inglés.

1. Instrucciones respecto al tipo de artículo

Los trabajos se estructurarán de acuerdo con las secciones y normas que a continuación se exponen:

a) ***Artículos originales (investigación científica).*** Documento que presenta, de manera detallada, los resultados originales de proyectos terminados de investigación.

Los **artículos originales**, de **reporte de caso** y **cortos**, deben cumplir con la siguiente estructura: la **introducción** debe resaltar la importancia de la investigación, presentar la literatura relacionada y entregar antecedentes necesarios para comprender la hipótesis de los autores, terminando con un párrafo que indique claramente los objetivos de la investigación. El **Método** debe tener suficiente información que permita a otro investigador replicar el ensayo y lograr los mismos resultados, así como la inclusión del diseño experimental, el análisis estadístico y las referencias de los métodos ya publicados. Los **resultados y discusión** se deben presentar en forma clara, apoyados con cuadros y figuras, con el análisis estadístico y los antecedentes de otros investigadores. Las **conclusiones**

deben redactarse de acuerdo con los objetivos de la investigación explicando claramente los principales resultados de la investigación. Las **referencias** deben contener todos los documentos consultados.

b) ***Artículo de reflexión.*** Documento que presenta resultados de investigación terminada desde una perspectiva analítica, interpretativa o crítica del autor, sobre un tema específico, recurriendo a fuentes originales. Los **artículos de reflexión** deben mostrar: a)intención analítica; b)propósito interpretativo; c)posición crítica. La estructura: **Resumen, Introducción, Desarrollo del tema, Conclusiones y referencias**.

c) Artículo de revisión. Documento resultado de una investigación terminada donde se analizan, sistematizan e integran los resultados de investigaciones publicadas o no publicadas, sobre un campo en ciencia o tecnología, con el fin de dar cuenta de los avances y las tendencias de desarrollo. Se caracteriza por presentar una cuidadosa revisión bibliográfica de por lo menos 40 referencias. Los **artículos de revisión** deben tener: **Resumen, Introducción, Desarrollo del tema, Conclusiones y referencias**.

d) Artículo original corto. Documento breve que presenta resultados originales preliminares o parciales de una investigación científica o tecnológica, que por lo general requieren de una pronta difusión. Tiene la misma estructura de los artículos originales.

e) Reporte de caso. Documento que presenta los resultados de un estudio sobre una situación particular con el fin de dar a conocer las experiencias técnicas y metodológicas consideradas en un caso específico. Incluye una revisión sistemática comentada de la literatura sobre casos análogos.

f) Revisión de tema. Documento resultado de la revisión crítica de la literatura sobre un tema en particular.

g) Cartas al editor. Posiciones críticas, analíticas o interpretativas sobre los documentos publicados en la revista, que a juicio del Comité Editorial constituyen un aporte importante a la discusión del tema por parte de la comunidad científica de referencia.

h) Editorial. Documento escrito por el editor, un miembro del comité editorial o un investigador invitado sobre orientaciones en el dominio temático de la revista.

i) Traducción. Traducciones de textos clásicos o de actualidad o transcripciones de documentos históricos o de interés particular en el dominio de publicación de la revista.

j) Documento de reflexión no derivado de investigación. Reflexión sobre un tema en particular, el cual no requiere ser derivado de una investigación científica o tecnológica

k) Reseña bibliográfica.

l) Otros

En todos los casos el artículo debe presentar aspectos de originalidad y contener las referencias bibliográficas.

Adicionalmente el autor deberá llenar el formulario de autorización para la publicación de artículos y/o revistas en formatos impresos y digitales.

2. Instrucciones respecto al formato del artículo

Los artículos seleccionados serán el centro de la revista, por lo que deben ajustarse a las siguientes especificaciones de forma:

a) Título del artículo (en español e inglés) no debe tener más de 15 palabras. El título estará

escrito con letras minúsculas y en negritas con tipo Times New Román de 12 puntos y centrado, este debe contener máximo 15 palabras. (En español e inglés).

b) Lista de Autores: Con los nombres completos y apellidos en el orden que deben aparecer. Adicionalmente el autor principal anotará su correo electrónico, para la correspondencia de editores y lectores; lugar y fecha de investigación y financiamiento si la tuvo.

c) Resumen y palabras clave: Se presenta un resumen en español y en inglés máximo de 100 palabras. Palabras clave en español e inglés (hasta un máximo de diez palabras).

d) Texto del artículo: Cuando los artículos son originales y original corto deberán tener las siguiente partes:

Introducción: El autor debe establecer el propósito del estudios, resumir su fundamento lógico.

Materiales y métodos: Incluye la selección de procedimientos para el trabajo experimental, y se indentifican los métodos y equipos con suficiente detalle.

Resultados y Discusión: Debe presentarse en secuencia lógica con sus re4spectivas tablas y gráficas y los comentarios de los principales hallazgos. En la Discusión enfatizar los aspectos más importantes del estudio.

Conclusiones: Anotar una o más conclusiones

Los demás artículos deberán tener como mínimo resúmen, palabras clave, el desarrollo del tema y las conclusiones.

e) El artículo debe estar capturado electrónicamente en una versión actualizada de Word, su extensión no debe exceder las doce (15) páginas en una columna y espacio sencillo, letras Times New Roman de 10 puntos, en hojas tamaño carta.

f) Los títulos principales deberán siempre ser precedidos de una línea en blanco, salvo que con ellos se comience una nueva página. Los títulos principales se escribirán centrados utilizando letras mayúsculas y en negritas con tipo Times New Román de 10 puntos. Se dejará una línea en blanco después de cada título principal. Cada título tendrá al menos dos líneas de texto que le continúen al final de cada hoja.

g) Las tablas deberán intercalar en el texto, ubicándose lo más cerca posible a la sección de texto que las refiere, pero no será aceptable, por fines de uniformidad, dividir la página en varias columnas para intercalar tabla en el texto. Se deberán centrar con respeto al formato del texto y deberá existir por lo menos una línea en blanco antes y después de cada tabla. La numeración de tablas deberá hacerse de forma secuencial. Los títulos de las tablas deberán ir en su parte inferior.

h) Las figuras se deberán intercalar con el texto, ubicándose lo más cerca posible a la sección de texto que las refiere, pero no será aceptable, por fines de uniformidad, dividir la página en varias columnas para intercalar figuras con el texto. Se deberán centrar con respecto al formato del texto y deberá existir por lo menos una línea en blanco antes y después de cada figura. La numeración de figuras deberá hacerse de forma secuencial. Los títulos tendrán como máximo una extensión de dos líneas. Las figuras no deberán enmarcarse, salvo que se trate de gráficas donde esto último se justifique. Asimismo, en la instrucción de formato de imagen de Word, deberá asegurarse que la figura se defina como un objeto en línea con el

texto, con la finalidad de que el control de la posición de la imagen en el mismo sea fácil de manipular por los editores en caso que se requiera algún ajuste menor en el artículo.

i) Si desea incluir una sección de agradecimientos, esta se ubicará después de la sección de conclusiones y antes de la sección de referencias, utilizando el mismo formato que para títulos y secciones principales.

j) Las fuentes bibliográficas deben ser citadas a lo largo del texto, en formato ISO.

k) Cualquier tema no cubierto en lo anterior será resuelto por el editor coordinador en el Consejo editorial y deberán ser consultados al siguiente correo electrónico: wpenafiel@ulasalle.edu.bo

La Salle
Universidad
Bolivia

CAMPUS BOLOGNIA:
Av. Jorge Carrasco esq.
Calle Las Palmas Nº 450 (Bolognia)
Telfs.: (591-2) 2723588
(591-2) 2723598

www.ulasalle.edu.bo
e-mail: wpenafiel@ulasalle.edu.bo