

TRABAJO DE GRADO

Opción Seminario-Diplomado.

**ALGORITMO COMPUTACIONAL PARA EL ANÁLISIS Y TOMA DE
DECISIONES EN PROYECCIONES DE PRECIOS DEL GANADO VACUNO EN PIE,
UTILIZANDO ESTRATEGIAS DE MACHINE LEARNING**

Corporación Universitaria Remington.

Nombre de la facultad: Ingenierías

Nombre del programa académico: Ingeniería Industrial

Estudiantes:

Pedro Jamith López Viloría

Tutor: Juan Carlos Briñez de León

Opción de Trabajo de grado Seminario-Diplomado.

2024

Dedicatoria

A MIS PADRES QUE ME DIERON LA VIDA, A MI ESPOSA E HIJOS, QUIENES ME IMPULSARON A INCIAR ESTA CARRERA Y ESTUVIERON A MI LADO DURANTE TODO ESTE PROCESO, BRINDANDOME SIEMPRE ESA VOZ DE ALIENTO Y EL APOYO NECESARIO PARA NUNCA DESISTIR, ESTE LOGRO ES TANTO MIO COMO DE ELLOS.

Agradecimientos

A DIOS, CREADOR DEL TODO, A LA ORGANIZACIÓN PARA LA QUE PRESTO MIS SERVICIOS, QUE ME BRINDO EL TIEMPO Y LAS HERRAMIENTAS PARA PODER ESTUDIAR, A MIS COMPAÑEROS DE CARRERA, A LA UNIVERSIDAD Y A TODOS LOS DOCENTES QUE ME BRINDARON SU APOYO Y APORTE EN ESTE PROCESO PARA CONVERTIRME EN INGENIERO.

Tabla de Contenido

INDICE DE TABLAS	5
INDICE DE FIGURAS.....	5
Resumen.....	7
Palabras clave.....	8
Marco conceptual y contextual	8
Pregunta problema:	9
Acercamiento a los datos.	10
Descripción de variables.	11
Aproximaciones con gráficos - analítica.....	12
Objetivos:	16
Objetivo general.....	16
Objetivos específicos.	16
Desarrollo e implementación del aprendizaje.....	16
Preparación de los datos.....	17
Identificando y eliminando filas con valores faltante	18
Identificando y eliminando filas con valores duplicados.....	19
Modelo de toma de decisiones	19
Entrenando los modelos escogidos:	21
Validación del modelo	23

	5
Resultados de los modelos	24
Conclusiones y trabajos futuros	27
Referencias.....	28

INDICE DE TABLAS

Tabla 1 <i>Datos ingresados y resultados predictivos</i>	25
---	----

INDICE DE FIGURAS

Ilustración 1 Imagen del web origen del conjunto de datos.....	11
Ilustración 2 Lista de variables contenidas en el data set	12
Ilustración 3 Grafico de var precio por Kg del ganado Sem1	13
Ilustración 4 Grafico de Dispersión Semanas 1 y 4	14
Ilustración 5 Gráfico de la Caja Sem 1.....	15
Ilustración 6 Organización de los datos para cargar al algoritmo	17
Ilustración 7 Limpieza de los datos y eliminación de datos nulos	18
Ilustración 8 Eliminación de filas con datos duplicados	19
Ilustración 9 Importando Pandas y cargando data set	20

Ilustración 10 Definiendo la matriz en Numpy	21
Ilustración 11 Cargando y entrenando el modelo 1, KNN	22
Ilustración 12 Cargando y entrenando el modelo 2, MLP	22
Ilustración 13 Cargando las funciones que permitan ingresar los nuevos datos	23
Ilustración 14 Ingresando los nuevos datos.....	24
Ilustración 15 Gráfico comparativos datos reales Vs Datos de Modelos 1 y 2.....	26

Resumen

El mercado colombiano de la carne, en especial la carne bobina se ve enfrentado diariamente a fluctuaciones considerables en los precios que perciben los productores, son muchas las variables a considerar y que afectan de gran manera estos precios, y que no permiten que se marque una tendencia que se pudiera pronosticar de una forma fácil y asertiva, los productores se ven enfrentados a la incertidumbre del rumbo que van a tomar los precios, lo que los obliga a vender sin tener la seguridad que se van a obtener los mejores márgenes de ganancia, limitando así las decisiones de futuras inversiones que garanticen una estabilidad en sus negocios y aportar al crecimiento de la economía del país.

Esta preocupación por la incertidumbre de la tendencia de los precios, este trabajo busca crear una herramienta basada en Machine Learning, que alimentándola con los datos históricos de los comportamientos y variaciones que ha tenido el precio por Kilogramo, se pueda predecir como será el comportamiento del precio por Kilogramo del ganado vacuno en pie en un lapso de tiempo determinado, con lo cual los productores pueden tener una guía y tomar una decisión si se arriesgan a vender o esperan a que se presente una mejor opción de venta.

Hoy en día y con el creciente auge de la inteligencia artificial y con la utilización de algoritmos de Machine Learning y aprendizaje autónomo, se pueden crear herramientas que le permitan mitigar la incertidumbre a los productores y obtener así un mejor margen de ganancia en sus procesos.

Palabras clave

Precio de ganado, Machine Learning, regresiones, predicción, Clustering, Algoritmos, Python, Google Colab.

Marco conceptual y contextual

Durante los últimos años y a lo largo del tiempo, tanto los productores como los comercializadores de ganado vacuno en pie, especialmente los productores se han visto enfrentados a una gran volatilidad y variabilidad del precio por kilogramo del ganado vacuno en pie en sus diferentes tipos y en las diferentes etapas de cría o de levante, son muchas las variables que afectan directa e indirectamente en la variabilidad de este precio, siendo las de mayor impacto los factores climatológicos como los cambios de temperatura y la variación en las precipitaciones, la oferta del mercado y Las enfermedades epidémicas entre otros, el precio varía de acuerdo a cada región y a cada ciudad, entre mayor o menor sea el precio, así mismo será la disposición de los productores para ofrecer una mayor o menor cantidad de producto tal como lo mencionó (Coronado, E. A., Ricardo, J. B., & Núñez, O. C. (2018)).

El problema principal al que se enfrentan tanto los productores como los comercializadores al momento de vender y de comprar ganado vacuno en pie, es que no existe una tendencia clara ni definida del comportamiento del precio por kilogramo, es entonces cuando los productores se enfrentan a la incertidumbre si venden o esperan a que el precio sea masa favorable para obtener un mayor margen de utilidades, o por el contrario el precio no se recupere y terminen vendiendo a un precio más bajo, trayendo consigo no solo una menor ganancia sino que se corre el riesgo de tener pérdidas significativas, que terminan afectando la economía de las empresas y por ende de las regiones productoras y comercializadoras de ganado vacuno, el no

vender el ganado en un punto específico de la etapa productiva trae consigo una serie de gastos adicionales tales como la alimentación y la manutención de los animales que al final terminan sumando y encareciendo más los costos de producción, sin tener la garantía de poder vender a un mejor precio, este no es un problema menor, debido a que la ganadería es una actividad que se practica en todas las regiones del país, siendo esta un renglón socio económico de vital importancia para el desarrollo de nuestro país, tal como lo mencionó (Gómez Arenas, E. V. (2021)).

Hoy en día los algoritmos de Machine Learning están siendo utilizados y entrenados mediante la utilización de algoritmos de inteligencia artificial, para ayudar a los productores en la toma de decisiones informadas, lo que convierte a estos en una poderosa herramienta que brindaría un apoyo fundamental a la hora de estimar y predecir el comportamiento del precio por Kg al que se puede llegar a comercializar el ganado vacuno en pie, es así como un pequeño o gran productor con unos sencillos pasos se puede anticipar y estar preparado para una fluctuación en los precios, de esta forma puede organizar su producción y generar estrategias que le permitan establecer el mejor momento y las mejores condiciones de favorabilidad para la venta de los lotes y así obtener y maximizar las utilidades obtenidas, tal como lo mencionado por Zapata Bustamante, J. C. (2024).

Pregunta problema:

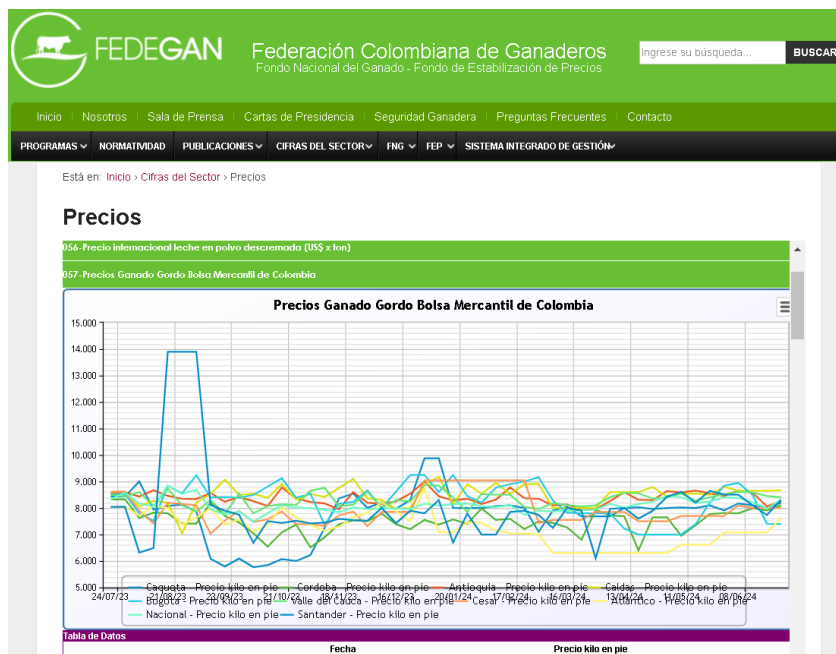
¿Cómo desarrollar una estrategia computacional para ayudar en la predicción de la variabilidad del precio por Kg del ganado vacuno en pie, a partir de algoritmos de Machine Learning?

Acercamiento a los datos: Los datos utilizados para el desarrollo del presente trabajo, utilizados para predecir el comportamiento del precio por Kg del ganado vacuno gordo en pie, fueron tomados de la página oficial de Fedegan, quien es la Federación Colombiana de Ganaderos, una entidad gremial sin ánimo de lucro la cual funciona desde el año 1963, En su condición de gremio cúpula de la ganadería colombiana, FEDEGÁN agrupa, en calidad de afiliadas, a las organizaciones gremiales ganaderas regionales y locales, como también a otro tipo de entidades vinculadas a la actividad ganadera nacional.

El nombre del archivo es “057-Precios Ganado Gordo Bolsa Mercantil de Colombia”, en este encontramos un histórico semanal y por departamentos colombianos, que va desde julio de 2023, hasta junio de 2024, en el cual se nos presenta el precio del Kg al cual se comercializó el ganado vacuno gordo en pie, la intención de estos datos es poder llevar un registro de las variaciones y fluctuaciones que ha tenido en precio por kg del ganado en Colombia, y la decisión es un rango del precio por Kg, con el objetivo de poder determinar el precio por Kg de comercialización del ganado en pie y obtener el mayor margen de utilidades de una empresa productora de cualquier tamaño y cantidad de unidades Kg comercializados.

Ilustración 1

Imagen del web origen del conjunto de datos



Descripción de variables.

El data set con el cual se realizaron las predicciones del siguiente trabajo, consta de cuatro columnas, las tres primeras definidas como datos de entrada contienen los precios por Kg del ganado vacuno en pie, donde se tomó como semana uno dos y tres de un mes calendario y una cuarta columna en la cual se estableció el dato a predecir o semana cuatro y última semana de un mes calendario.

Ilustración 2

Lista de variables contenidas en el data set

```

<class 'pandas.core.frame.DataFrame'>
RangeIndex: 45 entries, 0 to 44
Data columns (total 4 columns):
#   Column                Non-Null Count  Dtype
---  -
0   Sem 1                 45 non-null     int64
1   Sem 2                 45 non-null     int64
2   Sem 3                 45 non-null     int64
3   Prediccion (Sem 4)   45 non-null     int64
dtypes: int64(4)
memory usage: 1.5 KB

```

Como se puede observar las variables contenidas en el conjunto de datos, son de tipo numérico, siendo todas ellas números enteros, para este caso las cuatro variables nos indican el precio por kilogramo al cual se comercializó el ganado gordo en pie, en la bolsa mercantil de Colombia. Siendo \$ 6.400 Kilo el valor mínimo de comercialización para todas las semanas analizadas y \$ 8.203 Kilo el valor máximo para las semanas 1,2 y 3, mientras que para la semana 4 o semana de predicción el valor máximo estuvo en \$ 8.330 Kilo.

Aproximaciones con gráficos - analítica.

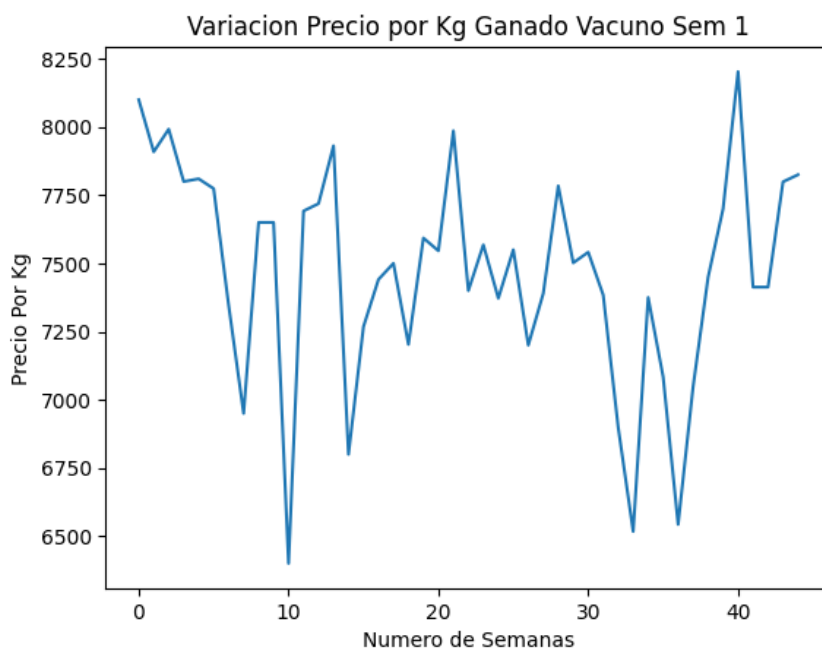
Como se puede observar en el gráfico siguiente, existe una gran variabilidad del comportamiento de los datos del precio por Kg del ganado, ya que observamos picos muy elevados del precio que estuvo por encima de los \$ 8.000 y caídas muy bruscas del precio, el cual osciló por debajo de los \$ 6.500 Kg, lo que nos deja ver una diferencia aproximada de \$ 1.800

Kg, esto se puede considerar como una diferencia bastante considerable, si un productor realiza su venta en uno de estos días donde el precio estuvo en su punto más bajo, le puede generar pérdidas significativas y obtener un menor margen de ganancia.

El comportamiento de los gráficos para las semanas restantes es muy similar al de la primera semana analizado anterior mente, por lo cual no se analizan en detalle.

Ilustración 3

Grafico de var precio por Kg del ganado Sem1

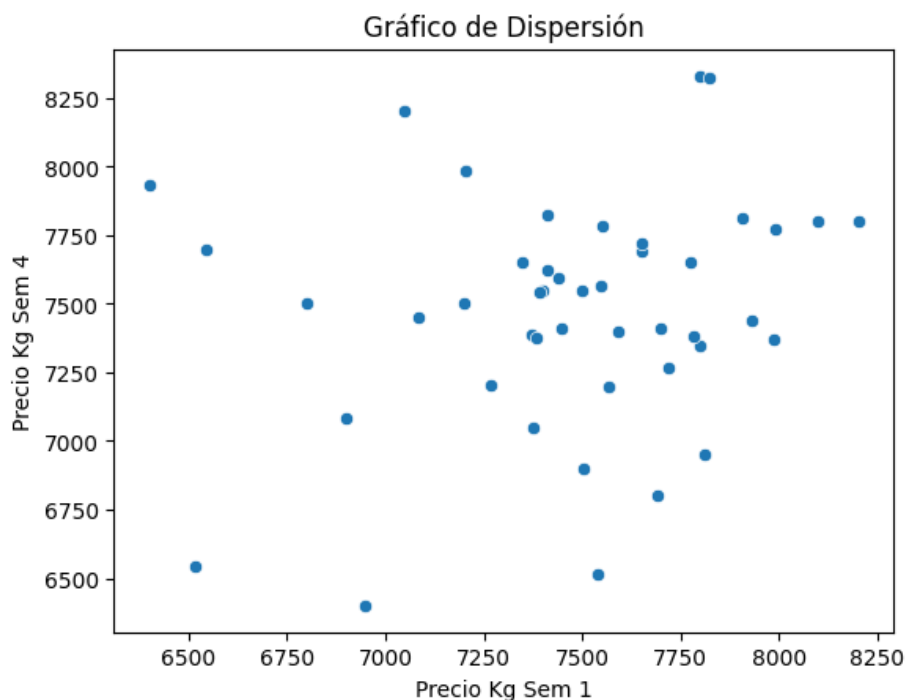


Para el siguiente análisis se realizó un gráfico de dispersión para las variables sem 1 y sem 4, y se puede observar que los comportamientos entre ambas semanas es muy parecido, ya que existen datos puntuales por debajo y por encima de los valores promedio para ambas variables, pero en su gran mayoría observamos un cumulo de puntos o de datos en la misma

franja, que corresponde entre los \$ 7.250 Kg y \$ 7.750 Kg, con lo cual podemos determinar, que si un productor logra vender un lote de producción con un precio que oscile entre estos valores, estaría dentro de un rango normal, con lo cual conseguiría una venta optima y un buen margen de ganancias.

Ilustración 4

Grafico de Dispersión Semanas 1 y 4

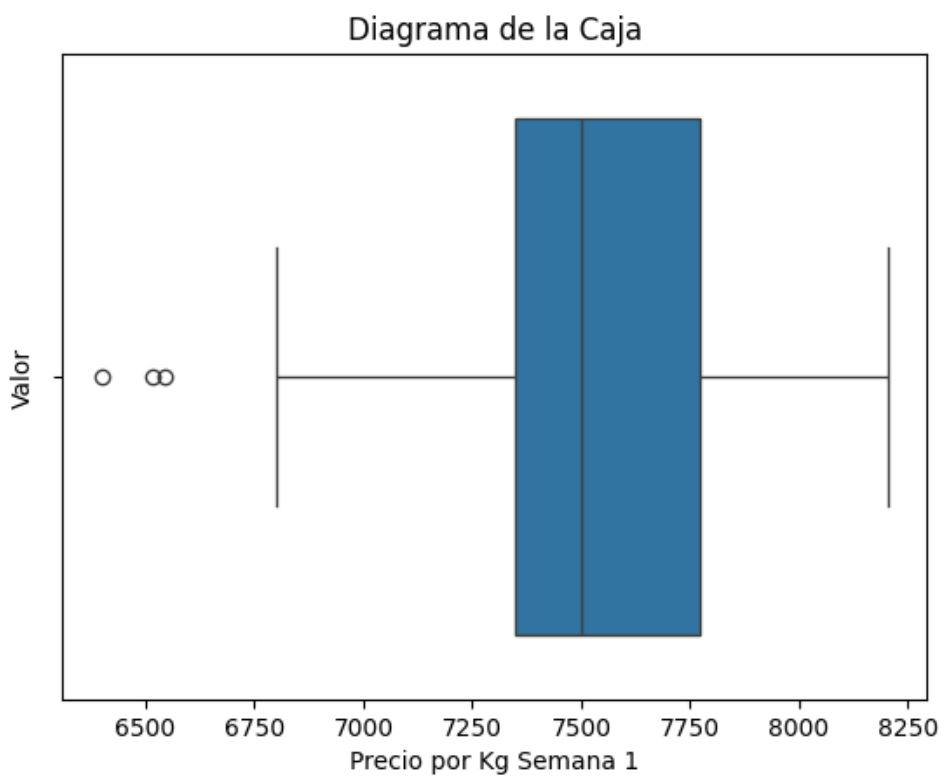


En el siguiente grafico el cual corresponde a un gráfico de cajas, podemos observar la ubicación y distribución de los datos por cuartiles, de precios por Kg semanal del ganado vacuno en pie, para la semana 1. Para este caso tenemos que la mediana de los datos la tenemos en 7.500, para el segundo cuartil el 50% de los datos (Q2), podemos observar que los datos se

encuentran más cercanos los unos de los otros, lo que nos indica que la variación fue menor, y este está dado por los valores entre 7.260 y 7.500 \$ por Kg, para el tercer cuartil el 75% de los datos (Q3) nos muestra que los datos presentan una mayor separación entre sí, lo que significa que hubo una mayor variación de los precios, y estos se encuentran ubicados entre los valores 7.500 y 7.760 \$ por Kg,

Ilustración 5

Gráfico de la Caja Sem 1



Objetivos:

Objetivo general.

Implementar un algoritmo computacional para el análisis y toma de decisiones a partir de datos del precio por kilogramo del ganado vacuno en pie, utilizando estrategias de machine Learning.

Objetivos específicos.

Caracterizar y procesar los datos de interés, con miras a la toma de decisiones informadas.

Implementar un algoritmo de Machine Learning para la toma de decisiones a partir de los datos de interés.

Evaluar y analizar el desempeño de los algoritmos implementados para la toma de decisiones.

Validar el funcionamiento de toma de decisiones a partir de datos nuevos.

Desarrollo e implementación del aprendizaje

Para el desarrollo del presente trabajo, una vez encontrados y extraídos los datos de la fuente de origen, se procedió a su organización y limpieza, la cual se realizó de forma manual utilizando un libro de Excel, luego se eliminaron variables que no correspondían o no se consideraban significativas para la toma de decisiones, se implementó un modelo de machine Learning en regresión, se evaluaron los resultados, y se puso a prueba con datos de la misma fuente, pero de

otros departamentos registrados. A continuación, se desarrolla cada una de las etapas de la presente metodología.

Preparación de los datos

Organización de los datos: para nuestro caso de estudio el primer paso que ejecutamos, fue realizar un filtro y escoger los datos de un solo departamento, aquí escogimos el departamento de Córdoba, por ser nuestro departamento de residencia y los datos se ajustan a nuestra realidad. Luego se organizaron los datos en columnas, debido a que el modelo elegido para trabajar corresponde a un modelo de regresión y este exige que los datos se encuentren en columnas donde se deben definir unas variables de entrada y una variable de salida, los datos se deben guardar una forma secuencial de tal forma que el segundo dato de la primera variable de entrada, se debe convertir en el primer dato de la segunda variable de entrada, y el tercer dato de la primera variable de entrada, se debe convertir en el primer dato de la tercera variable de entrada y así sucesivamente.

Ilustración 6

Organización de los datos para cargar al algoritmo

Ub	Fecha	Precio Kg en Pie
Nacional	lun, jul 24, '23	8.325
Antioquia	lun, jul 24, '23	8.551
Valle del Cauca	lun, jul 24, '23	8.5
Bogotá	lun, jul 24, '23	8.515
Santander	lun, jul 24, '23	8.046
Córdoba	lun, jul 24, '23	8.325
Caquetá	lun, jul 24, '23	8.431
Caldas	lun, jul 24, '23	8.524
Atlántico	lun, jul 24, '23	8.037
Cesar	lun, jul 24, '23	8.618
Nacional	lun, jul 31, '23	8.58
Antioquia	lun, jul 31, '23	8.596
Valle del Cauca	lun, jul 31, '23	8.436

Sem 1	Sem 2	Sem 3	Prediccion (Sem 4)
8100	7909	7992	7800
7909	7992	7800	7810
7992	7800	7810	7774
7800	7810	7774	7347
7810	7774	7347	6950
7774	7347	6950	7650
7347	6950	7650	7650
6950	7650	7650	6400
7650	7650	6400	7692
7650	6400	7692	7719
6400	7692	7719	7931
7692	7719	7931	6800

Identificando y eliminando filas con valores faltante

Para lograr un resultado acertado y evitar errores al momento de ejecutar el algoritmo, se implementó el método **Dropna**, este método nos permite eliminar las filas con datos faltantes o que no pertenezcan a la misma serie de datos con la que estamos trabajando en nuestro data set, y nos arroja una nueva serie de datos con los datos filtrados y optimizados para realizar en este caso la predicción, como se puede observar en este ejercicio, no se detectaron filas con valores faltantes.

Ilustración 7

Limpieza de los datos y eliminación de datos nulos

```
# Identificar valores faltantes
print(Dataset.isnull().sum())

# Eliminar filas con valores faltantes
Dataset = Dataset.dropna()
Dataset.head(7)
```

Sem 1	0
Sem 2	0
Sem 3	0
Prediccion (Sem 4)	0

dtype: int64

	Sem 1	Sem 2	Sem 3	Prediccion (Sem 4)
0	8100	7909	7992	7800
1	7909	7992	7800	7810
2	7992	7800	7810	7774
3	7800	7810	7774	7347
4	7810	7774	7347	6950
5	7774	7347	6950	7650
6	7347	6950	7650	7650

Identificando y eliminando filas con valores duplicados

Siguiendo con la depuración de los datos, se optó por la implementación de método `.DROP()`, el cual nos proporciona para este caso la eliminación de filas con valores repetidos, con esto se busca tener una mayor confiabilidad en el resultado final del ejercicio y en la toma de decisiones, para este caso al aplicar el método, no se encontraron valores duplicados, por lo que no se eliminaron filas repetidas y el data set quedó con los mismos datos iniciales.

Ilustración 8

Eliminación de filas con datos duplicados

```
# Identificar y eliminar duplicados
print('Número de filas antes de eliminar duplicados:', len(Dataset))
Dataset = Dataset.drop_duplicates()
print('Número de filas después de eliminar duplicados:', len(Dataset))

Número de filas antes de eliminar duplicados: 45
Número de filas después de eliminar duplicados: 45
```

Modelo de toma de decisiones

El modelo escogido para la toma de decisiones del presente trabajo fue un modelo de **Regresión**, con predicción en series de tiempo, para la utilización de este modelo es necesario conocer como fue el comportamiento de los datos a través del tiempo, y realizando un estudio de estos podemos predecir como va a ser su comportamiento en un futuro o predecir un dato siguiente.

- El modelo de toma de decisiones escogido, se trabajó en la plataforma de Google Colab, esta es una herramienta de Google con la cual podemos programar y ejecutar Python en el navegador, sin que sea necesario tener el programa instalado en un equipo.
- El primer paso fue la importación de las librerías “Pandas” de Python, la cual es una librería especializada, cuyo objetivo es el manejar y analizar estructuras de datos, esta nos permite trabajar en diferentes formatos y ficheros tales como Excel, CSV y SQL, para este trabajo se utilizaron ficheros de Excel y CSV.
- Con la utilización del método “.head”, logramos obtener una visualización de como quedan cargados los datos y así validar si corresponden con los archivos deseados para trabajar, este método por defecto nos muestra las primeros 5 filas de datos.

Ilustración 9

Importando Pandas y cargando data set

```
import pandas as pd
from google.colab import files
uploaded = files.upload()
for filename in uploaded.keys():
    Dataset = pd.read_excel(filename)

Dataset.head(7)
```

Elegir archivos Ningún archivo seleccionado Upload widget is on

Saving precios semanal ganado vacuno colombia.xlsx

	Sem 1	Sem 2	Sem 3	Prediccion (Sem 4)
0	8100	7909	7992	7800
1	7909	7992	7800	7810
2	7992	7800	7810	7774
3	7800	7810	7774	7347
4	7810	7774	7347	6950
5	7774	7347	6950	7650
6	7347	6950	7650	7650

- Debido a que se está utilizando un modelo de regresión, es necesario ajustar los datos, para que estos correspondan a una matriz numérica y definir cuales son los datos de entrada y el dato resultado o proyectado, para ello utilizamos e importamos la biblioteca Numpy de Python, la cual es una herramienta avanzada en los lenguajes de programación, que nos da soporte para manipular y crear matrices.

En esta entrada también definimos las variables de entrada que se establecieron desde la columna 0 a la 3 y la variable de salida o resultado, definida por la columna (-1), que para este caso es la ultima columna o columna 4.

Ilustración 10

Definiendo la matriz en Numpy

```
#Ajustando los datos a una matriz numérica
import numpy as np
Dataset_Matriz = np.array(Dataset)
#Dividiendo los datos en entrada y salida
Entradas = Dataset_Matriz[:,0:3]
Salida = Dataset_Matriz[:, -1]
```

Entrenando los modelos escogidos:

Primer Modelo, se utilizó un algoritmo de machine Learning que utilizan IA, el algoritmo “**KNeighborsRegressor**”, también conocido como algoritmo k de vecinos más cercanos, que es un tipo de algoritmos el cual utilizan la proximidad de los datos para hacer predicciones, este modelo es muy funcional siempre y cuando sabemos que los datos no se salen del rango.

Primero se importa y se carga la librería de KNN regressor, seguidamente se da la instrucción para que utilizando este código cree el modelo número 1 y luego utilizando la función “Fit()” se entrena el modelo utilizando los datos suministrados.

Ilustración 11

Cargando y entrenando el modelo 1, KNN

```
#Modelo para predicciones de series de tiempo (KNN)
from sklearn.neighbors import KNeighborsRegressor as KNMR
Modelo_1=KNMR()
Modelo_1.fit(Entradas,Salida)
```

```
▼ KNeighborsRegressor
KNeighborsRegressor()
```

Segundo modelo, se utilizó un modelo basado en machine Learning por redes neuronales denominado “**MLPRegressor**”, el MLP o preceptor multicapa, es una red de neurona artificial y se utiliza para el tipo de aprendizaje supervisado, se siguen los mismos pasos para cargar y entrenar el primer modelo.

Ilustración 12

Cargando y entrenando el modelo 2, MLP

```
#Modelo para series de tiempo complicadas (Sólo se conoce la dinámica)
from sklearn.neural_network import MLPRegressor
Modelo_2 = MLPRegressor()
Modelo_2.fit(Entradas, Salida)
```

```
▼ MLPRegressor
MLPRegressor()
```

Validación del modelo

- Ingresando los datos para generar una predicción,

Una vez tenemos los modelos cargados y entrenados, debemos ingresar datos y validar que tan efectivos son nuestros modelos de predicción, dado que para nuestro caso de estudio se utilizaron tres variables de entrada para obtener una de respuesta, se ingresaron tres líneas de código utilizando la función **“Input”** y la función **“float”** que nos permiten ingresar variables de tipo numérico, el mismo numero de variables con las que se entrenó el programa, son el mismo numero que se deben ingresar para obtener una respuesta.

Seguidamente se colocó la función **“Predict()”**, la cual al ser ejecutada obtenemos los resultados de predicción para los datos ingresados.

Ilustración 13

Cargando las funciones que permitan ingresar los nuevos datos

```
#Se ingresan los datos de entrada para generar una predicción
Nueva_entrada = np.zeros((1,3))
Nueva_entrada[0,0]=float(input('Ingrese el valor del precio por Kg actual: '))
Nueva_entrada[0,1]=float(input('Ingrese el valor del precio por Kg sem pasada: '))
Nueva_entrada[0,2]=float(input('Ingrese el valor del precio por Kg hace dos sem: '))

Proyeccion_1 = Modelo_1.predict(Nueva_entrada)
Proyeccion_2 = Modelo_2.predict(Nueva_entrada)

print('')
print('')
print('Según los datos ingresados, la proyección del precio por Kg para la siguiente sem usando KNN será: ',Proyeccion_1[0])
print('')
print('Según los datos ingresados, la proyección del precio por Kg para la siguiente sem usando ANN será: ',Proyeccion_2[0])
```

Al ejecutar las líneas de código, esta nos brinda la opción para colocar el primer valor o primera variable solicitada y así sucesivamente has completar el número de variables con las que se entrenó el programa, en este caso son tres.

Ilustración 14

Ingresando los nuevos datos.

```
print('')
print('')
print('Según los datos ingresados, la proyección del precio por Kg para la siguiente sem usando KNN será: ',Proyeccion_1[0])
print('')
print('Según los datos ingresados, la proyección del precio por Kg para la siguiente sem usando ANN será: ',Proyeccion_2[0])
```

Ingrese el valor del precio por Kg actual:

Resultados de los modelos

Para realizar la validación y eficiencia de los modelos, se tomaron diez datos de la misma fuente de la data set original, solo que se tomaron datos de otra región del país, estos datos fueron cargados por semanas, y los resultados se compararon con los datos reales de la semana cuatro.

Al comparar los resultados obtenidos para el modelo 1 y 2, podemos observar que los resultados del modelo 2, son los que mas se acercan a los resultados de la semana 4, solamente en uno de los casos el resultado del modelo 1, se acercó mas al resultado de la semana 4, por consiguiente, para la toma de decisiones nos quedamos con los resultados del Modelo 2

“MLPRegressor”

Tabla 1*Datos ingresados y resultados predictivos*

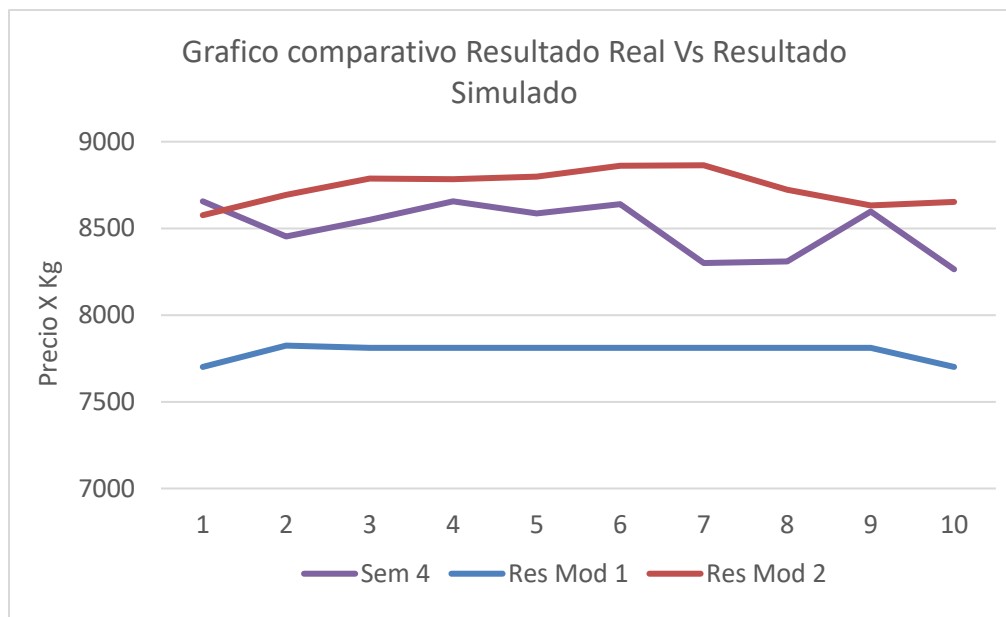
Sem 1	Sem 2	Sem 3	Sem 4	Res Mod 1	Res Mod 2	Dif Mod 1	Dif Mod 2
8251	8066	8578	8656	7701	8576	-955	-80
8066	8578	8656	8453	7824	8693	-629	240
8578	8656	8453	8550	7811	8787	-739	237
8656	8453	8550	8656	7811	8784	-845	128
8453	8550	8656	8586	7811	8799	-775	213
8550	8656	8586	8640	7811	8862	-829	222
8656	8586	8640	8301	7811	8864	-490	563
8586	8640	8301	8309	7811	8722	-498	413
8640	8301	8309	8597	7811	8632	-786	35
8301	8309	8597	8264	7701	8652	-563	388

Nota. Esta tabla nos muestra los datos ingresados al programa, Sem 1,2 y3 y el resultado real de la semana 4, y lo compara con los resultados de los modelos 1 y 2, para final mente mostrarnos la diferencia entre el resultado de ambos modelos y el real de los datos.

Como lo podemos observar en el siguiente gráfico, los resultados del modelo 2, son los que mas se asemejan a los resultados reales de la semana 4, incluso en algunos puntos del gráfico, los datos se cruzan, lo que nos indica que el resultado obtenido fue igual al resultado real de la tabla de datos, mientras que el modelo 1 siempre se mantuvo por debajo del comportamiento de los datos deseados, con un comportamiento muy lineal y bastante alejado de la realidad.

Ilustración 15

Gráfico comparativos datos reales Vs Datos de Modelos 1 y 2



Conclusiones y trabajos futuros

En este trabajo se materializó la construcción de un algoritmo para la toma de decisiones informadas en la proyección del precio por kilogramo del ganado vacuno en pie, utilizando algoritmos de machine learning.

En cuanto a los modelos utilizados, el algoritmo con mejor desempeño fue **“MLPRegressor”**, también llamado modelo 2, y el de menos desempeño fue el del modelo 1 **“KNeighborsRegressor”**,

Para el caso de la toma de decisiones, este trabajo opta por el modelo 2 **“MLPRegressor”**,

Para trabajos futuros, se proyecta realizar un ejercicio con cargue de datos con mayor información, y que estos datos correspondan con históricos clasificados para cada ciclo estacionario y por departamentos, ya que estos factores son determinantes en las variaciones y fluctuaciones de los precios por kilogramos que se paga para el ganado vacuno en pie, aparte de la fuente original de los datos, se plantea la recolección de datos de otras fuentes, tales como las principales subastas ganaderas del país y armar un conjunto de datos mas robusto con lo cual se espera que el nivel de precisión y asertividad de los resultados sea mayor, luego poder validar en tiempo real los resultados que arroja el mercado y los resultado predictivos obtenidos mediante el modelo de machine learning.

Referencias

Zapata Bustamante, J. C. (2024). ¿ Es posible pronosticar el precio por kilogramo en el mercado porcícola como una herramienta de gestión de riesgo?.

Coronado, E. A., Ricardo, J. B., & Núñez, O. C. (2018). La oferta de ganado vacuno para ceba en el departamento de Sucre, Colombia: un enfoque ARDL. *Revista de Economía del Caribe*, (21), 57-74.

Gómez Arenas, E. V. (2021). Caracterización del sistema de producción ganadero en el municipio del Valle de San José-Santander.

¿Qué es KNN? | IBM. (s. f.). <https://www.ibm.com/mx-es/topics/knn>