Universidad Autónoma de Nuevo León

Facultad de Ciencias Físico Matemáticas

Maestria en Ciencia de Datos

Aprendizaje Automático
Articulo

Autor:
Alan Fernando Mejia Aranda

Marzo 2023

1. Introducción

Lo que observaran a continuación es un estudio estadístico de una muestra de datos financieros de una empresa de la rama automotriz, es una empresa internacional por lo que tiene presencia tambien en Estados Unidos, esta empresa cuenta con distintos tipos de productos que estan segmentados por Cateogrias. En esta ocasion nos enfocaremos en una categoria que es Accesorios, dentro de esta categoria nos encontraremos productos como cubrevolantes, tapetes, portaplacas, llaveros, entre muchos otros productos, como sabemos que muchas veces este tipo de productos se venden a menores precios por el tipo de producto, el principal objetivo de esta categoria es cumplir su plan de venta en unidades. El principal indicador es las unidades que vende, con esto lo que queremos conocer es cuales son las variables que están afectando al momento de que un cliente tome la decisión de comprar o no este tipo de producto. En esta ocasion se cuenta con una problematica debido a que los gerentes de categoria estan nerviosos por cumplir su plan de venta por lo que haremos el siguiente analisis con las siguientes variables:

1. *AUR*: Esta variable representa el precio promedio al que se están vendiendo los articulos, es decir del total de la venta por periodo o por semana cual es el precio promedio al que se vendio cada unidad, esta variable la estamos tomando en consideración debido a que usualmente hacemos incrementos de precio debido al incremento de costo de nuestros productos o decrementos debido a alguna promoción o oportunidad del mercado, esta variable sigue una escala de intervalo y es continua.

2. *AUC*: Esta variable representa el costo promedio al que nos están vendiendo los artículos que estamos vendiendo, es decir del total de la venta por periodo o por semana cual fue el costo promedio de unidad vendida, esta puede subir debido a incremento de precio de materias primas que el proveedor nos refleja a nosotros en el costo o puede bajar por negociaciones o por tipo de cambio si el proveedor no nos vende el producto en MXP, esta variable sigue una escala de intervalo y es continua.

3. *Closure Rate*: Es un indicador con el que contamos que nos representa el porcentaje de cierre de ventas, cuando una persona busca un producto se crea una búsqueda y cuando una persona compra un producto se genera una transacción, por lo que el "closure rate" es el numero de transacciones entre el número de búsquedas, así vemos el porcentaje con el que se cierra la venta, esta variable sigue una escala de intervalo y es continua.

4. *Instock*: Es el indicador que nos dice cual es el porcentaje de inventario con el que contamos, esta variable la tomamos en consideración debido a que puede ser un factor por el cual no se esta vendiendo el producto, esta variable sigue una escala de intervalo y es continua..

5. *Debito a Ingreso*: Proporción del pago total de la deuda mensual del representante dividido por el ingreso mensual autoinformado, excluyendo la hipoteca.

6. *Units USA*: Son las unidades que vende esta empresa, pero en Estados Unidos, esto debido a que puede que lo que este afectando en el ritmo de las Unidades sea un factor externo, como la temporalidad, un ejemplo puede ser cuando es Enero y las empresas sufren una caída importante en sus ventas, esta variable sigue una escala de intervalo y es continua.

7. *Units*: Unidades vendidas.

8. *Target (Variable de Interés)*: Nos indica si cumpple con el plan de venta (1) o no cumple con el plan de venta (0).

Para lograr la predicción mas acertada nos enfocamos tanto en el apredizaje no supervisado y en el aprendizaje supervisado. El aprendizaje no supervisado es una técnica de aprendizaje automático en la que el modelo debe encontrar patrones y relaciones en los datos sin la ayuda de etiquetas o respuestas conocidas previamente.

El aprendizaje supervisado es un tipo de algoritmo de aprendizaje automático que utiliza datos etiquetados para entrenar un modelo y hacer predicciones o clasificaciones precisas. El objetivo del algoritmo es aprender una función que pueda mapear las entradas a las etiquetas de salida.

2. Marco Teoríco

El objetivo del aprendizaje automático es desarrollar herramientas y técnicas capaces de automatizar actividades humanas que consumen mucho tiempo de manera precisa y oportuna. Existen los siguientes algoritmos supervisados:

- Regresión Logística.
- Árbol de Decisión.
- Bosque Aleatorio.
- K Vecinos más Cercanos.

2.1. Regresión Logística

El objetivo de la Regresión Logística es encontrar la relación entre un conjunto de variables predictoras (independientes) y una variable de resultado (dependiente) binaria, toma valores en dos categorías posibles, como "sí/no", "verdadero/falso" o "1/0". La Regresión Logística es un algoritmo de aprendizaje supervisado que se utiliza comúnmente en problemas de clasificación binaria.

La Regresión Logística utiliza la función logística (también conocida como sigmoide) para modelar la probabilidad de que una instancia pertenezca a una categoría. La función logística tiene la siguiente forma matemática:

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

Donde z es la suma ponderada de las variables predictoras:

$$z = \beta_0 + \beta_1 x_1 + \beta_2 x_2 + ... + \beta_n x_n$$

Aquí, β_0 es la intersección (sesgo), β_1 a β_n son los coeficientes de regresión y x_1 a x_n son las variables predictoras.

La función logística toma cualquier valor de entrada y lo transforma en un valor entre 0 y 1, i la probabilidad resultante es mayor que un umbral de decisión (por ejemplo, 0.5), se clasifica la instancia en la categoría positiva; de lo contrario, se clasifica en la categoría negativa.

El modelo de Regresión Logística se ajusta mediante la maximización de la función de verosimilitud logarítmica, que se define como la probabilidad de observar los datos dados los parámetros del modelo. La función de verosimilitud logarítmica es:

$$\ell(\beta) = \sum_{i=1}^{m} [y_i \log(\sigma(z_i)) + (1 - y_i) \log(1 - \sigma(z_i))]$$

Donde m es el número de instancias, y_i es la etiqueta real de la instancia i y z_i es la suma ponderada de las variables predictoras para la instancia i.

2.2. Árbol de Decisión

La construcción de un árbol de decisión se puede ver como la tarea de dividir el espacio de características en regiones cada vez más pequeñas y homogéneas, donde cada región corresponde a una hoja del árbol. Los árboles de decisión se utilizan comúnmente en problemas de clasificación, pero también se pueden utilizar en problemas de regresión.

El aprendizaje supervisado "Árbol de Decisión" es un algoritmo que se utiliza para la clasificación y regresión en problemas de aprendizaje automático.

El objetivo de un árbol de decisión es crear un modelo que prediga el valor de una variable objetivo mediante la evaluación de una serie de reglas de decisión simples derivadas de las características de los datos.

Un árbol de decisión se construye mediante la partición de los datos de entrenamiento en subconjuntos más pequeños y homogéneos basados en las características de los datos.

La función de costo más comúnmente utilizada para los árboles de decisión es la entropía, que se define como:

$$H(T) = -\sum_{i=1}^{c} p(i|T) \log_2 p(i|T)$$

donde T es un nodo del árbol, c es el número de clases, y $p(i|T)$ es la proporción de muestras en el nodo T que pertenecen a la clase i. La entropía mide la cantidad de incertidumbre o desorden en un conjunto de muestras.

El algoritmo de construcción del árbol de decisión selecciona la característica que minimiza la entropía después de la división, lo que se puede calcular mediante la ganancia de información, que se define como:

$$IG(D_p, f) = H(D_p) - \sum_{j=1}^{m} \frac{N_j}{N_p} H(D_j)$$

donde D_p es el conjunto de datos en el nodo padre, f es la característica que se está evaluando, m es el número de valores posibles de la característica f, N_j es el número de muestras que pertenecen a la j-ésima rama, y N_p es el número total de muestras en el nodo padre.

El proceso de construcción del árbol se puede detener mediante la definición de un criterio de detención, como la profundidad máxima del árbol o el número mínimo de muestras requeridas para dividir un nodo. Una vez que se ha construido el árbol, se pueden utilizar diferentes estrategias de poda para reducir su complejidad y evitar el sobreajuste. La poda consiste en eliminar algunas de las ramas del árbol que no aportan información relevante o que son demasiado específicas para el conjunto de entrenamiento, lo que puede mejorar la capacidad de generalización del modelo.

2.3. Bosque Aleatorio

Una de las ventajas de Bosque Aleatorio es su capacidad para manejar grandes conjuntos de datos con muchas características. Además, puede manejar conjuntos de datos con valores

perdidos y valores categóricos.

El aprendizaje supervisado de Bosque Aleatorio, es un algoritmo que se utiliza para la clasificación, regresión y otras tareas de aprendizaje automático. Es una técnica de ensamblado que combina múltiples árboles de decisión para producir un modelo más robusto y preciso.

El algoritmo de Bosque Aleatorio funciona creando múltiples árboles de decisión a partir de diferentes muestras de los datos de entrenamiento y características aleatorias, y combinando sus predicciones para producir una predicción final. Cada árbol de decisión se entrena en una muestra aleatoria de los datos de entrenamiento y utiliza una subconjunto aleatorio de las características para cada división de nodo.

El proceso de entrenamiento del modelo se puede resumir en los siguientes pasos:

1. Seleccionar una muestra aleatoria de los datos de entrenamiento.
2. Seleccionar un subconjunto aleatorio de características para cada árbol.
3. Entrenar un árbol de decisión en la muestra de entrenamiento y características seleccionadas.
4. Repetir los pasos 1-3 un número predeterminado de veces para crear múltiples árboles de decisión.
5. Combinar las predicciones de los árboles para producir una predicción final.

La fórmula para la predicción en Bosque Aleatorio es la siguiente:

$$\hat{y} = \frac{1}{B} \sum_{i=1}^{B} h_i(\mathbf{x})$$

donde $\hat{y}$ es la predicción final, $n_{arboles}$ es el número de árboles en el bosque, $f_i(x)$ es la predicción del i-ésimo árbol y x es el vector de características de entrada.

Siendo así, combina múltiples árboles de decisión para producir un modelo más preciso y robusto. Se utiliza comúnmente en problemas de clasificación, regresión y otras tareas de aprendizaje automático.

2.4. K Vecinos Más Cercanos

El algoritmo se basa en la premisa de que los puntos de datos similares tienden a agruparse en el mismo espacio. Por lo tanto, se puede utilizar la cercanía en el espacio de características para determinar la clase a la que pertenece un punto de datos desconocido.

El clasificador de vecinos más cercanos asigna una instancia según la clase de sus vecinos más cercanos. Se conoce más comúnmente como el vecino más cercano k (k-NN) ya que a menudo es más beneficioso considerar más de un vecino (ver Henley y Hand, 1996).

El algoritmo KNN se puede resumir en los siguientes pasos:

1. Calcular la distancia entre el punto de datos desconocido y todos los puntos de datos conocidos en el conjunto de entrenamiento.
2. Seleccionar los K puntos de datos más cercanos al punto desconocido.
3. Asignar la clase más común entre los K vecinos más cercanos al punto desconocido como la clase del punto desconocido.

La distancia entre dos puntos de datos se puede calcular utilizando diferentes métricas, pero la más común es la distancia euclidiana. La fórmula para la distancia euclidiana entre dos puntos A y B en un espacio de n dimensiones se puede expresar como:

$$d(A, B) = \sqrt{\sum_{i=1}^{n}(A_i - B_i)^2}$$

donde A_i y B_i son las coordenadas del punto A y B en la i-ésima dimensión.

Una vez que se han calculado las distancias, se seleccionan los K vecinos más cercanos al punto desconocido. Luego, la clase más común entre los K vecinos más cercanos se asigna al punto desconocido. En el caso de una clasificación binaria, la clase más común se puede determinar por mayoría de votos. En el caso de una clasificación múltiple, se puede utilizar el voto ponderado por la distancia para determinar la clase más probable.

Dónde la fórmula para el voto ponderado por la distancia en el caso de una clasificación múltiple:

$$\hat{y} = \arg\max_i \sum_{j=1}^{K} w_j I(y_j = i)$$

donde $\hat{y}$ es la clase predicha para el punto desconocido, w_j es el peso del j-ésimo vecino más cercano, y_j es la clase del j-ésimo vecino más cercano e $I(y_j = i)$ es una función indicadora que es 1 si la clase del j-ésimo vecino más cercano es i y 0 en caso contrario.

3. Metodología

Se tomaron en cuenta las variables mencionadas en la introduccion debido a que son las que afectan principalmente a la variable de Unidades, pero tambien sabemos que tiene un plan de ventas y es el principal objetivo por lo que nos plantearemos como variable de respuesta el Target.

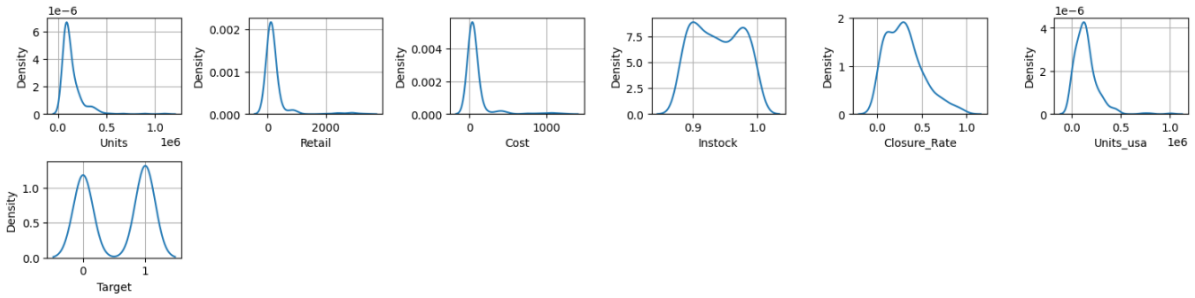
A continuación en la tabla 1 se muestran las primeras filas de nuestra base de datos de entrenamiento:

Figura 1: Variables

	Units	Retail	Cost	Instock	Closure_Rate	Units_usa	Target
0	392437	22	3	0.9060	0.0638	475304	1
1	69410	349	123	0.9062	0.8027	138188	0
2	248107	109	48	0.9440	0.3161	313928	1
3	89189	89	33	0.9560	0.2047	1434	0
4	87539	215	96	0.8933	0.5375	185985	0

A continuacion, graficamos cada una de nuestras variables, para ver si notabamos un patron en su comportamiento y asi decir que siguen algun tipo de distribucion, esto debido a que creemos que algunas de nuestras variables pueden seguir una distribucion normal, pero se ira revisando cada una de ellas con su grafico:

Figura 2: Densidad de Cada Variable.



Dicho esto se estandarizarán las variables para poder tener un mejor resultado en los modelos por aplicar dentro del aprendizaje supervisado. A su vez, se retira la variable de interes "Target" para continuar con la creación de los modelos.

Figura 3: Estandarizacion de datos.

	Units	Retail	Cost	Instock	Closure_Rate	Units_usa
0	3.394894	-0.851259	-0.821622	-0.466593	-0.824080	3.047843
1	-0.391691	2.141876	1.772973	-0.463640	1.888226	0.098756
2	1.703030	-0.054920	0.151351	0.094500	0.102046	1.636127
3	-0.159838	-0.237986	-0.172973	0.271687	-0.306873	-1.097566
4	-0.179180	0.915332	1.189189	-0.654116	0.914747	0.516884

Planteado lo anterior, se procedió a realizar la creación de los modelos donde la metodología inicial fue hacer iteraciones con la regresión logística, árbol de decisión, bosque aleatorio y k vecinos más cercanos. Para ello se hace el ajuste del modelo de acuerdo a los datos de entrenamiento, se procede a predecir la variable de interés "Target" en función a las regresoras de la base de entrenamiento, empieza a predecir el "Estatus del Préstamo" de la base de prueba de acuerdo a las variables regresoras de esta misma base y por último mostramos la precisión de cada uno de los modelos, comparandolos a su vez con las matrices de confusión para ver cuantos datos pudieron predecir correctamente cada uno de los modelos.

El metodo Smote funciona seleccionando una muestra de la clase minoritaria y encontrando sus k-vecinos más cercanos para seleccionar uno de ellos al azar e interpolar una nueva muestra entre la muestra original y el vecino elegido, creando sintéticamente una nueva muestra de la clase minoritaria y repitiendo este proceso hasta que tengamos equilibrio entre ambas clases, lo veremos aplicado mas adelante.

Este proceso se realiza porque el desequilibrio de clases puede hacer que los clasificadores se sesguen hacia la clase mayoritaria y les resulte difícil identificar correctamente los ejemplos de la clase minoritaria, lo que conduce a un rendimiento deficiente y puntuaciones de recuperación bajas.

La fórmula utilizada para la generación de nuevas instancias a través de SMOTE es la siguiente:

$$X_{nueva} = X_i + \lambda(X_{zi} - X_i)$$

4. Resultados

Planteada la metodología anterior en nuestro primer método donde se realiza el ajuste a partir de los datos de entrenamiento donde se observaron las siguientes precisiones para cada modelo.

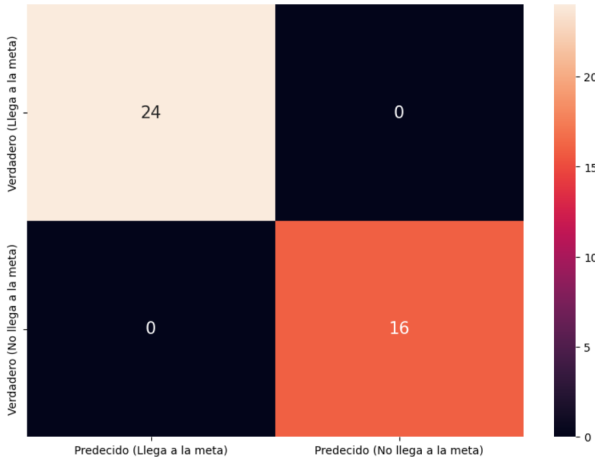
Tabla 1: Precisión de los Modelos.

Modelo	Precisión
Regresión Logística.	97.85 %
Árbol de Decisión.	100.00 %
Bosque Aleatorio.	100.00 %
K Vecinos más Cercanos.	86.33 %

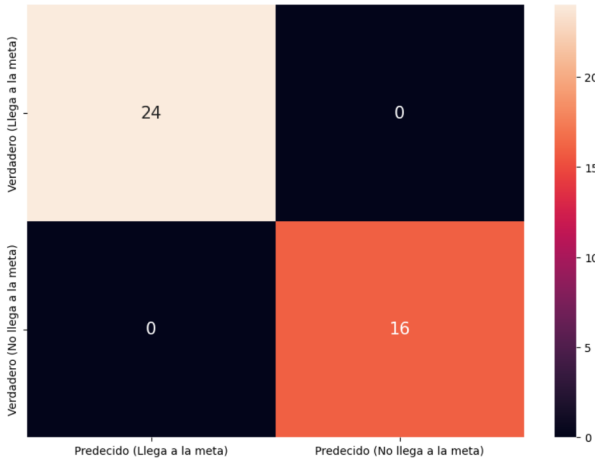
Para este método la precisión se obtiene comparando la variable de interés predecida de entrenamiento contra la verdadera de esa misma base de datos. A continuación mostramos las matrices de confusión de los 2 mejores modelos:

Figura 4: Matriz de Confusión.

(a) Árbol de Decisión



(b) Bosque Aleatorio



En ambos modelos obtuvimos todos los resultados bien predecidos.

Realizamos el procedimiento de "SMOTE" en los datos para que los clasificadores no tengan un sesgo hacia la clase mayoritaria.

Los resultados de precisión para los modelos fueron los siguientes:

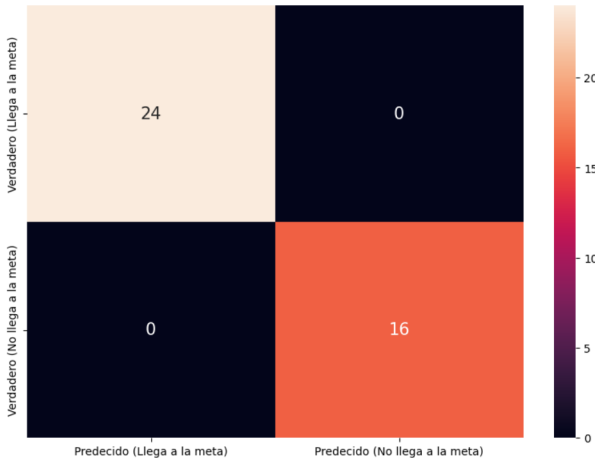
Tabla 2: Precisión de los Modelos.

Modelo	Precisión
Regresión Logística.	97.88 %
Árbol de Decisión.	100.00 %
Bosque Aleatorio.	100.00 %
K Vecinos más Cercanos.	72.22 %

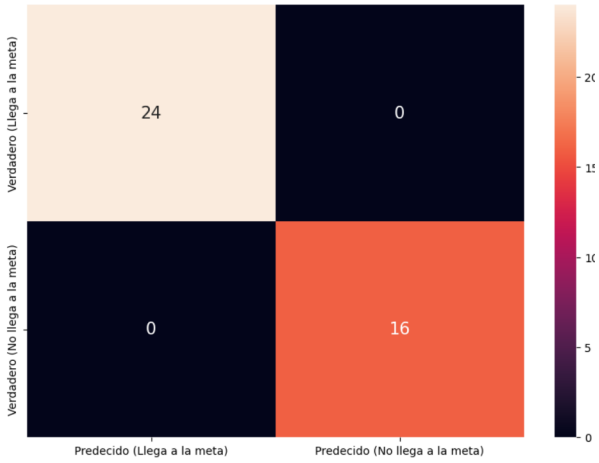
Considero que esto no cambia mucho, por la base de datos es muy pequena y por esta razon no cambia mucho de un modelo a otro.

Figura 5: Matriz de Confusión con SOMTE.

(a) Regresión Logística



(b) Árbol de Decisión



Observamos como las matrices de confusion permanecen de la misma manera.

5. Conclusiones

Este ejercicio puede servir mucho para algun equipo de Planeacion, crear alguno de estos modelos y hacer sus planeaciones con estos tipos de variables y ver si en realidad podremos llegar a los planes puestos por la compania y no ser sorprendidos en caso de que no se llegue a los planes de venta de unidades.

Se consideran las variables que creemos que mas influyen en esta variable de respuesta, pero en realidad existen mas variables que son categoricas que debemos encontrar la manera de meterla en este tipo de analisis.

Por el numero de datos es complicado que cambie de un metodo a otro, con una base mas amplia podriamos notar mayores diferencias, pero aun con esta las notamos, pero en cuanto a los porcentajes de precision de un modelo a otro.

Este tipo de proyectos, se pueden explotar con bases amplias, pero venia trabajando esta base, las bases de usar cada uno de los metodos estan y se puede implementar con datos masivos para lograr notar mayor diferencia entre cada uno de los metodos.

6. Referencias

Hand, D. & Zhou, F. (2009). Evaluating models for classifying customers in retail banking collections. Journal of the Operational Research Society, 61, 1540–1547. 23, 91, 124, 180

Lee, S. (2005). Application of logistic regression model and its validation for landslide susceptibility mapping using gis and remote sensing data. International Journal of Remote Sensing, 26, 1477–1491. 25

Kennedy, K. (2013). Credit scoring using machine learning. Doctoral thesis. Technological University Dublin. doi:10.21427/D7NC7J.

Bishop, C. M. (2006). Pattern recognition and machine learning. New York: Springer.