Mundo Fesc E-ISSN: 2216-0388 P- ISSN: 2216-0353 Artículo Original

Detección de situaciones de emergencias usando el modelo Naive- Bayes de machine learning.

Detection of emergency situations using the Naive Bayes machine learning model.

Recibido: 26 de agosto de 2022
Aprobado: 4 de diciembre de 2022

Forma de citar: I.L. Vásquez Rojas, M.J. Vívas Cortez, "Detección de situaciones de emergencias usando el modelo Naive- Bayes de machine learning", *Mundo Fesc,* vol 13, no. 25, pp. 21-40 de 2023. https://doi.org/10.61799/2216-0388.1286

Iván Leonel Vásquez Rojas
Decanato Ciencias Económicas y Contables
vasquezivan@ucla.edu.ve
Universidad Lisandro Alvarado
Barquisimeto, Venezuela.

Miguel José Vívas Cortez*
Facultad de ciencias exactas y naturales
mjvivas@puce.edu.ec
Pontificia Universidad Católica del Ecuador
Quito, Ecuador.

***Autor para correspondencia:**
mjvivas@puce.edu.ec

Detección de situaciones de emergencias usando el modelo Naive- Bayes de machine learning

Resumen

En la actualidad, las redes sociales han ganado terreno en la generación y obtención de información al instante. Esta característica la hace de gran utilidad en la detección y advertencias de emergencias tales como accidentes viales, incendios, tormentas, inundaciones, etc. Esto ha motivado la generación de una gran cantidad de trabajos acerca del aprovechamiento de esta información para enfrentar los problemas generados por tales emergencias, trabajo como el de A. Kansal, Y. Singh, N. Kumar "Detection of forest fire using Machine Learning technique" [1] o de Chamorro Verónica "Clasificación de tweets mediante modelos de aprendizaje supervisado" [2], muestran el uso de técnicas de machine learning para la detección de situaciones extraordinarias. Tras estas situaciones catastróficas o de emergencias es necesario gestionar los servicios de atención y protección de la población.Problemas como caos informativo, incertidumbre en las necesidades y servicios pueden encontrar solución en la detección oportuna de cuáles eventos son realmente emergencias, así el propósito de este trabajo utiliza mensajes de X (Twitter) para clasificar cuáles emergencias en son reales o no. Se utiliza para ello, el algoritmo de machine Learning conocido como Naive-Bayes, en este problema de clasificación de los mensajes de X, para determinar las emergencias reales, con un resultado en la evaluación de la exactitud en la clasificación de emergencia real con una proporción del 73.4% entre las clasificadas como emergencias y clasifica las emergencias falsas con una precisión del 75.4% entre la clasificada como falsa. En general el modelo obtenido tiene una exactitud del 74.6% en sus pronósticos de clasificación. Se considera que la utilización de un modelo Naive-Bayes para un prototipo en la clasificación de los mensajes de emergencias de la red social X podría ser de gran utilidad con base a los resultados de la evaluación de su performance de clasificación.

Palabras clave: aprendizaje automático, Bayes, clasificación, emergencias, modelos, twitter

Detection of emergency situations using the Naive Bayes machine learning model

Abstract

Nowadays, social networks have gained ground in generating and obtaining instant information. This characteristic makes it very useful in the detection and warning of emergencies such as road accidents, fires, storms, floods, etc. This has led to the generation of a large number of works on the use of this information to address the problems generated by such emergencies, work such as that of A. Kansal, Y. Singh, N. Kumar "Detection of forest fire using Machine Learning technique" [1] or Chamorro Verónica "Classification of tweets using supervised learning models" [2], show the use of machine learning techniques for the detection of extraordinary situations. After these catastrophic or emergency situations, it is necessary to manage the services of care and protection of the population, problems such as information chaos, uncertainty in the needs and services can find a solution in the timely detection of which events are really emergencies, so the purpose of this work is to use messages from X (Twitter) to classify which emergencies are real or not. The machine learning algorithm known as Naive-Bayes is used in this problem of classification of X messages to determine the real emergencies, with a result in the evaluation of the accuracy in the classification of real emergencies with a proportion of 73.4% among those classified as emergencies and classifies false emergencies with an accuracy of 75.4% among those classified as false. Overall the model obtained has an accuracy of 74.6% in its classification predictions. It is considered that the use of a Naive-Bayes model for a prototype in the classification of emergency messages of the social network X could be of great use based on the results of the evaluation of its classification performance.

Keywords: machine learning, Bayes, classification, emergencies, models, twitter

Introducción

En la actualidad existen una gran cantidad de redes sociales a las cuales los usuarios dedican un alto porcentaje del tiempo del día. Estas redes sociales han afectado la cotidianidad de las personas. Unos de los cambios que se evidencia mayormente en la actualidad, es como las redes sociales están desplazando a los medios de información tradicionales como la radio, la televisión y otros como medios de información. Es común que las personas utilicen aplicaciones de Facebook, Twitter o Instagram para obtener información de interés o de igual forma proporcionar información de interés al momento. Así entre las investigaciones sobre la utilización de las redes sociales y los datos que generan los usuarios, destaca la de obtener información en situaciones de emergencias o riesgo. En este sentido es justo el desarrollo de este trabajo al obtener un modelo que permita utilizar las redes sociales para alertar situaciones de riesgo real y en consecuencia medidas paliativas. Aquí la palabra clave es modelar y como se define en [3] es "la capacidad de construir una interpretación probabilística de un fenómeno observado y la historia que va con este".

Una de las redes sociales más importante en la actualidad es Equis o también conocida como Twitter, así basado en los datos proporcionados por Kaggle [4], nuestro objetivo es construir un modelo de Machine Learning que permita clasificar los mensajes en Twitter sobre emergencias, esto es un modelo que prediga cuales mensajes se refieren a desastres o emergencias reales y cuáles no.

Este tipo de problema encuentra una solución en los algoritmos obtenidos a partir del método de Naive Bayes en el área de machine learning, básicamente el algoritmo de Naive-Bayes funciona usando teoría de probabilidad y más específicamente estadística Bayesiana para problemas de clasificación, que se resume en resolver problemas donde las variables respuestas son cualitativas como se ve en [5], [6] y [7]. Dentro de los métodos de clasificación es frecuente calcular la probabilidad que determinada observación pertenezca a las categorías de la variable cualitativa como base del proceso de clasificación. Entonces la predicción de una respuesta cualitativa para una observación particular (este proceso se conoce como clasificación) involucra la asignación de la observación a una determinada categoría.

Dentro de las técnicas de clasificación útiles para predecir respuestas cualitativas o clasificación tenemos los modelos de regresión logística, la K-vecindad cercana, análisis del discrimínate lineal, análisis del discriminante cuadrático, los cuales podemos ver en los trabajos de [8] y [9] por último el algoritmo de Naive- Bayes, muy útil para los trabajos de clasificación de textos como se indica en los estudios [10],[11],[12] y [13] siendo este último método el que utilizaremos en el desarrollo de este trabajo para la clasificación de los mensajes de Twitter.

Modelo Naive Bayes

El modelo de clasificación de Naive –Bayes, es usado frecuentemente para la clasificación de textos, este funciona muy bien en estos casos debido a que este tipo de problema en el cual se tiene información de numerosos atributos que deben ser considerados al mismo tiempo con la finalidad de estimar en base a estos atributo, la probabilidad de un resultado.

Antes de ahondar más detalles del modelo de Naive- Bayes consideremos algunos aspectos importantes de acuerdo a [14]: la probabilidad, concierne al estudio de la incertidumbre, podríamos ver como la porción de veces que un evento ocurre o el que tanto ocurre un determinado evento. Entonces al final queremos medir o cuantificar valores de incertidumbre, lo que sucede es que no determinamos las probabilidades directamente de los resultados de un experimento, en lugar de esto trabajamos con variables numéricas más convenientes para determinar las probabilidades, aquí entra en juego la idea de las variables aleatorias.

Teorema (de Bayes). La probabilidad condicional de un evento A, dado que otro evento B ocurrió esta dada por la ecuación (1):

$$P(A/B)=\frac{P(A\ y\ B)}{P(B)} \quad (1)$$

Donde $P(B)>0$.

La probabilidad conjunta $P(AyB)$ también denotada por $P(A\cap B)$, es la probabilidad que evalúa la posibilidad que ocurran dos eventos de manera simultánea.

Definición: Dados dos eventos A y B, se dicen que ellos son independientes si y solo si la ocurrencia de uno de estos eventos no afecta la probabilidad de ocurrencia del otro evento.

Fórmula condicional de eventos independientes: Dos eventos A y B son independientes si se cumple cualquiera de estas condiciones de las ecuaciones (2) y (3):

a.- $P(A/B)=P(A)$ (2)
b.- $P(A/B)=P(B)$ (3)

Fórmula de la multiplicación

a.- Dados dos eventos A y B cualesquiera, entonces se cumple que en la ecuación (4):

$$P(A\ y\ B)=P(A).P(A/B) \quad (4)$$

b. Si los eventos A y B son independientes de las ecuaciones (2) y (3) se reducen (4) a la ecuación (5):

$$P(A \text{ y } B)=P(A).P(B) \qquad (5)$$

Observación: dado que $P(A \text{ y } B)=P(B \text{ y } A)$ se tiene que ecuación (6):

$$P(A/B)=\frac{P(A \text{ y } B)}{P(B)}=\frac{P(B/A).P(A)}{P(B)} \qquad (6)$$

La expresión final (6) es útil para clasificar (o filtrar) observaciones en un problema bajo condiciones establecidas o donde se tiene evidencia o conocimiento de probabilidades condicionales, un caso para su uso es cuando se desea filtrar correos no deseados bajo ciertas características. Por ejemplo los correos con la palabra "oferta" considerarlos como no deseados (spam), entonces lo lógico sería clasificar este correo según el valor de la probabilidad de la ecuación (7):

$$P(\text{span/oferta})=\frac{P(\text{oferta/span}).P(\text{spam})}{P(\text{oferta})} \qquad (7)$$

Es decir si esta probabilidad en (7), es mayor a un determinado número clasificar como spam. Cuando se conoce cuántos correos no deseados tienen la palabra "oferta", cuantos correos son no deseados y cuantos tienen la palabra "oferta".

Teorema de Probabilidad Total.

Definición: Un conjunto de eventos $\{E1, E2,,..., En\}$, mutuamente excluyentes y colectivamente exhaustivos de un espacio muestral S, se les conoce como una partición de Ω. O equivalentemente

- $\Omega= E1\cup E2\cup...,\cup En$ (colativamente exhaustivo)
- $Ei\cap Ej=\emptyset$ (son disjuntos)

Teorema (probabilidad total). Sea $\{E1,E2,...,En\}$, una partición de un espacio muestral Ω, tal que la $P(Ei)>0,i=1,...n$. Entonces para cualquier evento A, se cumple que ecuación (8):

$$P(A)=P(E1).P(A/E1)+P(E2).P(A/E2)+\cdots+P(En).P(A/En) \qquad (8)$$

La regla de Bayes en el teorema de Bayes en (1), junto al teorema de probabilidad total permite obtener el siguiente resultado también conocido como regla de Bayes.

Sea $\{E1,E2\}$, una partición de un espacio muestral Ω, tal que la $P(Ei)>0,i=1,2$. Entonces para cualquier evento B, se cumple que ecuación (9):

$$P(E_1/B)=\frac{P(E_1).P(B/E_1)}{\sum_i P(E^i).P(B/E^i)} \qquad (9)$$

Variable aleatoria X, es una descripción numérica de los posibles resultados de un experimento. Una variable aleatoria, es vista como una función que toma elementos en el espacio muestral y le asigna un valor numérico.

Conceptos frecuentes en probabilidad son:

- Espacio muestral Ω, que es el conjunto de todos los posibles resultados de un experimento.
- Espacio de eventos A, es el espacio cuyos elementos son los eventos A. Un evento A es un subconjunto del espacio muestral Ω. Un evento lo definimos como el conjunto formado por ninguno, uno o más resultados de un experimento.
- La probabilidad P, a cada evento A se asocia un número $P(A)$ que mide la posibilidad o valor de creencia que el evento pueda ocurrir.
- Así una variable aleatoria X es una función de Ω en Γ, donde Γ lo conoceremos como espacio objetivo y sus elementos como estados.

Variables aleatorias discretas y continuas: dependiendo si el espacio objetivo Γ es discreto o continuo, de esa misma manera nos referiremos al tipo de variable aleatoria. Podemos entonces ver que para una variable aleatoria Y, el conjunto $\{Y=0\}=\{E\_1,E\_2,\ldots,E\_r\}$, donde los E_1 son puntos muestrales en Ω a los cuales X asigna 0. Entonces $P(X=x)$ es la suma de las probabilidades de los puntos muestrales a los que se le asigna el valor de x.

Definición: La distribución de probabilidad para una variable aleatoria X, puede representarse mediante una fórmula, una tabla o una gráfica, que proporciona $P(X=x)\ \forall x$

Teorema: Cualquier distribución de variable aleatoria discreta se cumple que:

- $0 \leq P(x\_i) \leq 1$
- $\Sigma P(x\_i)=1$

Cuando el espacio objetivo es discreto se puede determinar la probabilidad que una variable aleatoria tome un valor particular $x \in \Gamma$ y lo denotamos con $P(X=x)$. Cuando el espacio objetivo es continuo lo natural es determinar la probabilidad que una variable aleatoria este en un intervalo.

Definición: Si X es una variable aleatoria continua entonces la función de distribución de probabilidad de X, está dada por $F(x)=P(X<x)$ para todo $-\infty \leq x \leq \infty$

Definición: Si F(x) es una función de distribución de X, entonces: ecuación (10)

$$f(x)=\frac{dF(X)}{dx}=F'(Y) \qquad (10)$$

Se define como la función de densidad de X, y así se tiene que ecuación (11):

$$F(x)=P(X<x)=\int_{-\infty}^{x} f(t)dt \quad (11)$$

Teorema: Si $f(x)$ es una funcion de densidad de una variable aleatoria X, entonces:

1.- $f(x)\geq 0, \forall x$
2.- $F(\infty)=P(X<\infty)=\int_{-\infty}^{\infty} f(t)dt$

Probabilidad multivariable.

A fin de extender los conceptos y propiedades en distribuciones de probabilidad con más de una variable aleatoria a continuación se muestran solo con dos variables, y el caso general es similar.

Definición: Si X_1 y X_2 son dos variables aleatorias discretas, entonces la probabilidad conjunta de X_1 y X_2 esta dada por:

$$F(x_1,x_2)=P(x_1,x_2)=P(X_1=x_1,X_2=x_2)$$

Teorema: Si X_1 y X_2 son dos variables aleatorias discretas con función de probabilidad conjunta $F(x_1,x_2)=P(x_1,x_2)$, entonces se cumple:

- $0\leq P(x_1,x_2)\leq 1 \forall x_1,x_2$
- $\sum_{x_1,\,x_2} P(x_1,x_2)=1$

Definición: Si X_1 y X_2 son dos variables aleatorias continuas, la función de distribución de probabilidad conjunta está dada por:

$$F(x_1,x_2)=P(x_1,x_2)=P(X_1<x_1,X_2<x_2)$$

Definición: Si X_1 y X_2 son dos variables aleatorias continuas, con función de distribución de probabilidad conjunta dada por $F(x_1,x_2)$. Si existe una función no negativa $f(x_1,x_2)$ tal que:

$$F(x_1,x_2)=\int_{-\infty}^{x_1}\int_{-\infty}^{x_2} f(t_1,t_2)dt_1\,dt_2 \quad \text{para} \quad -\infty<x_1,x_2<\infty$$

Entonces se dice que son variables aleatorias continuas conjuntas y $f(x_1,x_2)$ es la función de densidad conjunta de X_1,X_2.

Teorema: Si $f(x_1,x_2)$ es una función de densidad continua conjunta de X_1,X_2. entonces:

$$1.- f(x_1,x_2)\geq 0, \forall x_1,x_2$$
$$2.- \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(t_1,t_2)dt_1\,dt_2=1$$

Definición: sea $F(x_1)$ y $F(x_2)$ funciones de distribución de X_1,X_2 respectivamente y $F(x_1,x_2)$ la función de distribución conjunta de X_1,X_2. Entonces se dice que X_1 y X_2 son independientes sí y solo sí: (ecuación (12)

$$F(x_1,x_2)=F(x_1).F(x_2)\ \forall x_1,x_2 \quad (12)$$

Teorema: Si X_1 y X_2 son dos variables aleatorias discretas con función de probabilidad conjunta $F(x_1,x_2)=P(x_1,x_2)$, entonces se cumple:

- $0\leq P(x_1,x_2)\leq 1 \forall x_1,x_2$
- $\sum_{x_1x_2} P(x_1,x_2)= 1$

Definición: Si X1 y X2 son dos variables aleatorias continuas, la función de distribución de probabilidad conjunta está dada por:

$$F(x_1,x_2)=P(x_1,x_2)=P(X_1<x_1,x_2<x_2)$$

Definición: Si X1 y X2 son dos variables aleatorias continuas, con función de distribución de probabilidad conjunta dada por $F(x_1,x_2)$. Si existe una función no negativa $f(x_1,x_2)$ tal que:

$$F(x_1,x_2)=\int_{-\infty}^{x_1}\int_{-\infty}^{x_2} f(t_1,t_2)dt_1dt_2 \quad \text{para} \quad -\infty<x_1,x_2<$$

Entonces se dice que son variables aleatorias continuas conjuntas y $f(x_1,x_2)$ es la función de densidad conjunta de X_1,X_2.

Teorema: Si $f(x_1,x_2)$ es una función de densidad continua conjunta de X_1,X_2. Entonces

$$1.-\ f(x_1,x_2)>\ 0,\ \forall x_1,x_2$$

$$2.-\ \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} f(t_1,t_2)dt_1d\overline{t_2}=1$$

Definición: sea $F(x_1)$ y $F(x_2)$ funciones de distribución de X_1,X_2 respectivamente y $F(x_1,x_2)$ la función de distribución conjunta de X_1,X_2. Entonces se dice que X_1 y X_2 son independientes sí y solo sí: (ecuación 13)

$$F(x_1,x_2)=F[(x_1).F(x_2)\ \forall x_1,x_2 \quad (13)$$

Si no se dicen dependientes.

Notese que la definición anterior es consistente con la definición de independencia de eventos.

Esta propiedad de independencia fundamenta el modelo de Naive-Bayes, puesto que en este se fundamenta en asumir que las variables poseen esta característica.

Algoritmo de Naive-Bayes

De acuerdo a [15] y [16], básicamente el algoritmo de Naive-Bayes, funciona usando teorema de Bayes para problemas de clasificación. Un ejemplo clásico de clasificación consiste en determinar los emails spam, como lo muestra [17]. Este asume que todas las características son igualmente importantes e independientes. Esta condición es raramente cierta en la vida real, pero a pesar de esta asunción Naive Bayas trabaja bien. Una explicación es que no importa el valor preciso de la probabilidad estimada siempre y cuando la predicción sea precisa. Entonces supongamos que queremos clasificar una observación X en una de las k-clases, con k>=2. Sean $E1, E2,, Ek$, las k-clases de la variable respuesta, tal que la $P(Ei)>0$, $i=1,n$. Entonces por el teorema de Bayes en (10), se tiene que: (ecuación (14)

$$P(Y=Ei/X=x)=\frac{P(E).P(X=x/Y=E_i)}{P(E1).P(X=x/Y=E_i)+...+P(E_n).P\ (X=x/Y=E_n)} \quad (14)$$

Para simplificar la notación hagamos $p_i(x)=P(Y=E_i/X=x)$ que es la probabilidad que una observación pertenezca a la clase i.

En la ecuación anterior (14) en lugar de calcular directamente pix como lo hacíamos en el modelo logístico vamos a estimar $P(E_i)$ y $P(X=x/Y=E_i)$. Generalmente la estimación de PEi es fácil si tenemos una muestral aleatoria de la población, simplemente calculamos la fracción de observaciones de entrenamiento que pertenecen a la i-ésima clase. Sin embargo estimar $P(X=x/Y=Ei)$ es un poco más complejo y como veremos será necesario hacer algunas suposiciones. Recordemos que el clasificador de Bayes de la fórmula (14), clasifica una observación x a la clase para el cual pix es el mayor. Entonces si logramos encontrar una forma de estimar $P(X=x/Y=E_i)$, podemos usar la ecuación (14) con la finalidad de aproximar el estimador de Bayes.

Antes de desarrollar el método Naive Bayes, consideremos $f_i(x)$ la función de densidad p-dimensional para una observación x en la i-ésima clase, el método se fundamenta en asumir la independencia de los predictores por cada clase así de (11) se tiene: ecuación (15)

$$f_i(x)=f_{i1}(x1) \times f_{i2}(x_2)\times...\times f_{iP}(x_P) \quad (15)$$

Para todo $i=1,...,n$ y f_{ij} es la función de densidad del j-ésimo predictor entre las observaciones de la clase E_i (si la variable es discreta entonces $f_{ij}=p(x_j/E_i)$ es la probabilidad del j-esimo predictor entre las observaciones de la clase E_i) Así no se tiene asociación de los predictores en cada clase. Esta asunción la mayoría de las veces no ocurre pero incluso así este asunción es hecha obteniéndose buenos resultados.

Así asumiendo esta independencia como en (15) se tiene que:

$$P(Y=Ei/X=x)=\frac{P(E_i)\times P(X=x/Y=E_i)}{P(E_1)\times P(X=x/Y=E_1)+...+P(E_n).P\,(X=x/Y=E_n)}$$

$$=\frac{P(E_i)\times f_{i1}(x_1 f_{i2} x_2\times...\times f_{ip}(x_p)}{\sum_{l=1}^{n} P(E_i)\times f_{l1}\,(x_1)\times f_{i2}\,(x_2)\times...\times f_{lp}(x_p)}$$

Para estimar la unidimensional función de densidad fij usando los datos de entrenamientos tenemos varias opciones:

1. Con X_j cuantitativa, entonces podemos asumir que $X_j/Y=E_n$ es aproximadamente normal. Esto es, asumimos que para cada clase el j-esimo predictor es descrito por una distribución normal.

2. Con Xj cuantitativa, entonces otra opción es usar un no paramétricos estimación de f_{kj}. Una sencilla forma para hacer esto es haciendo un histograma para las observaciones del f_{kj} predictor con cada clase. Entonces

3. Con X_j cuantitativa, entonces. Podemos simplemente contar la proporción de observaciones para el j-esimo predictor correspondiente a cada clase. Por ejemplo supongamos que $X_j\in\{1,2,3\}$ y tenemos 100 observaciones en la k-esima clase. Supongamos que el j-esimo predictor toma sobre los valores 1,2 y 3 en 32, 55 y 13 de esas observaciones respectivamente. Entonces estimamos f_{kj} como

Para ilustrar como trabaja Naive- Bayes usando el tercer método descrito anteriormente, supongamos que se desea clasificar objetos, en dos diferentes tipos de clases A o B, estos objetos cumplen con las características w1, ~w2, ~w3 y w4, las cuales asumimos independientes, entonces nuestro método se basa en determinar la probabilidad:

$$P(A/w1\cap\sim w2\cap\sim w3\cap w4)=\frac{P(w1\cap\sim w2\cap\sim w3\cap w4/A).P(A)}{P(w1\cap\sim w2\cap\sim w3\cap w4)}$$

El desarrollo de la fórmula será computacionalmente difícil de calcular cada vez que agregamos más características al problema que es lo que normalmente sucede, dada esta situación se necesitará una gran cantidad de memoria para almacenar las probabilidades de todas las posibles intersecciones. Así si asumimos lo que hace el algoritmo de Naive-Bayes sobre la independencia la expresión anterior esta se transformaría en:

$$P(A/w1\cap \sim w2 \cap \sim w3\cap w4) = \frac{P(w1\cap \sim w2 \cap \sim w3 \cap w4/A).P(A)}{P(w1\cap \sim w2 \cap \sim w3 \cap w4)}$$

$$= \frac{P(w1/A).P(\sim w2/A).P(\sim w3/A).P(w4/A).P(A)}{P(w1).P(\sim w2).P(\sim w3).P(w4)}$$

Y en consecuencia la probabilidad que sea B, dado que tiene las características w1, ~w2, ~w3 y w4, es:

$$P(B/w1\cap \sim w2 \cap \sim w3 \cap w4) = \frac{P(w1/B).P(\sim w2/B).P(\sim w3/B).P(w4/B).P(B)}{P(w1).P(\sim w2).P(\sim w3).P(w4)}$$

Las expresiones resultantes serán las que participen en el algoritmo pues son menos difíciles de calcular para determinar la probabilidad condicional establecida.

Tipos de clasificadores Naive Bayes:

- **Multinomial Naive Bayes:** es usado mayormente para la clasificación de documentos, este algoritmo usa la frecuencia de las palabras presentes en el texto para determinar a cual clase pertenece.

- **Bermoulli Naive Bayes:** similar al multinomial con la diferencia que en vez de considerar la frecuencia de las palabras establece la variable como booleana, es decir si está o no en el texto.

- **Gaussiano Naive Bayes:** aquí se considera el caso que las variables predictoras son continuas.

El algoritmo nave Bayes es el más utilizado en el análisis sentimental, filtrado y clasificación de textos.

Posiblemente una de las características nunca suceda esto agrega un inconveniente adicional, resultando en una probabilidad condicional que sea spam (o no sea) igual a cero (o probabilidad de 100%). Pero las restantes características podrían ser usualmente asociadas con mensajes spam, entonces una alternativa para evitar este inconveniente es usar el estimador de Laplace.

Evaluación del performance del modelo seleccionado

Evaluar el performance de los algoritmos de machine Learning es una de las tareas fundamentales en su construcción, preguntas de “Como medir el éxito de un modelo” o “como podría saber si he tenido éxito” son respondidas mediante la evaluación de los

modelos de machine Learning.

A propósito de medir el performance de un modelo de machine Learning la mejor forma es la cual capture si el clasificador es exitoso en su propósito. Cuando desarrollamos un modelo de clasificación generalmente se tienen los elementos siguientes valores actuales de la clase, valores predichos de la clase y probabilidad estimada de la predicción. Entonces los valores de la clase actual y los valores predichos serán parte clave de la evaluación.

Si consideramos una clasificación binaria entonces solo existen dos posibles clases, en el caso de tener más de dos clases los métodos son fácilmente extensibles a estos casos. Existen varias formas de medir el performance en modelos de clasificación ver [18]: Exactitud, matriz de confusión, log-loss, AUC y precisión- recall.

La exactitud, es la más común medida del performance de un modelo de clasificación, es una métrica que mide la frecuencia en que el clasificador hace correctamente la predicción. Se define como el cociente entre el número correcto de predicciones entre el número total de puntos en el conjunto de prueba (ecucación 16).

$$\text{Exactitud} = \frac{\text{número de predicciones correctas}}{\text{número total de puntos}} \quad (16)$$

Una matriz o tabla de confusión, muestra detalladamente las correctas e incorrectas clasificaciones en cada clase. Las clases de interés son conocidas como clase positiva, mientras que las otras son conocidas como clases negativa. Por otro lado donde el modelo hace una correcta predicción lo conocemos como verdadero y donde el modelo se equivoca la conocemos como falso, entonces en una clasificación de dos clases se observará lo siguiente (tabla I).

Tabla I. Predicción del modelo

	Predicción del modelo		
Valor real		Si	No
	Si (positiva)	Verdadero positivo TP	Falso positivo FP
	No (negativa)	Falso negativo FN	Verdadero negativo TN

La sensibilidad del modelo también llamado clase positiva mide la proporción de elementos positivos correctamente clasificados entre los que son realmente positivos. (ecuación 17)

$$\text{sensibilidad} = \frac{\text{TP}}{\text{TP+FN}} \quad (17)$$

La especificidad del modelo también llamada clase negativa mide la proporción de observaciones negativas correctamente clasificadas entre las que son realmente negativas (Ecucación 18).

$$\text{especificidad} = \frac{\text{TP}}{\text{TP+FP}} \quad (18)$$

AUC, la visualización son útiles para entender el performance del algoritmo de machine Learning en general. La visualización representa como un modelo trabaja a través de un rango amplio de condiciones. Dado que dos algoritmos tienen diferentes sesgos es posible que dos modelos con similar exactitud en su performance tengan grandes diferencias en como ellos alcanzan su exactitud. La curva característica de operación de receptor ROC, es comúnmente usada para examinar el contraste entre la clase de verdaderos positivos mientras evita los falsos positivos. Las características de clasificación en la curva ROC pueden ser medidos mediante el área bajo la curva (AUC) que van desde 0.5 para sin ningún valor predictivo hasta 1 para clasificadores perfectos. Valores cercanos a 1 no indican un modelo con un alto performance.

Clasificación de los mensajes de X.

Twitter es una de las redes sociales más populares del mundo con más de 300 millones de usuarios, generándose alrededor de 500 millones de tweets (publicaciones) diarios y más de 350 mil por minuto, la cual principalmente utiliza mensajes de textos para compartir información de diversas índoles, deporte, música, política, publicidad, sucesos o acontecimientos y muchos más. Así es natural que se utilice como medio de información instantánea para los casos de emergencias. Con la finalidad de clasificar los mensajes en Twitter para conocer cuales se refieren a emergencias reales y cuáles no, usaremos los paquetes tm, Snowball, wordcloud, e1071, tydiverse en el lenguaje de programación R y el conjunto de datos obtenidos de Kaggle [3]. Estos datos son provistos en un archivo el cual contiene la variables id, keywords, text (donde están los mensajes) y la variable target (etiqueta) que muestra la clasificación de estos, en emergencias reales o no con los números 1 y 0 respectivamente. Además contiene 7613 observaciones o mensajes como en la figura 1.

id	keyword	location	text	target
1			Our Deeds are the Reason of this #earthquake May ALLAH Forgive us all	1
4			Forest fire near La Ronge Sask. Canada	1
5			All residents asked to 'shelter in place' are being notified by officers. No other evacuation or shelt	1
6			13,000 people receive #wildfires evacuation orders in California	1
7			Just got sent this photo from Ruby #Alaska as smoke from #wildfires pours into a school	1
8			#RockyFire Update => California Hwy. 20 closed in both directions due to Lake County fire - #CAfire	1
10			#flood #disaster Heavy rain causes flash flooding of streets in Manitou, Colorado Springs areas	1
13			I'm on top of the hill and I can see a fire in the woods...	1
14			There's an emergency evacuation happening now in the building across the street	1
15			I'm afraid that the tornado is coming to our area...	1
16			Three people died from the heat wave so far	1
17			Haha South Tampa is getting flooded hah- WAIT A SECOND I LIVE IN SOUTH TAMPA WHAT AM I GOI	1
18			#raining #flooding #Florida #TampaBay #Tampa 18 or 19 days. I've lost count	1
19			#Flood in Bago Myanmar #We arrived Bago	1
20			Damage to school bus on 80 in multi car crash #BREAKING	1
23			What's up man?	0

Figura 1. Clasificación de mensajes de emergencia reales o no

Este conjunto de datos lo dividimos en dos, un primer conjunto con el 75% de los datos (5710 mensajes) para entrenar el modelo y el segundo conjunto con el 25% de los datos (1903 mensajes) para evaluar el modelo obtenido.

Algunas palabras pueden aparecer tanto en los mensajes clasificados como emergencias reales y en los clasificados como no emergencias, pero el algoritmo Naive Bayes toma en cuenta las frecuencias de estas palabras en los mensajes para así establecer una probabilidad que determina como un mensaje se clasificará. Por ejemplo en la figura 2 se representan en un words cloud las palabras más frecuentes en los mensajes de emergencias.

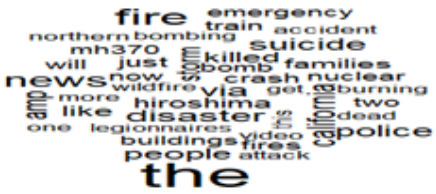

Figura 2. Words cloud las palabras más frecuentes en los mensajes de emergencias.

Procesamiento de datos

En esta primera parte se trata de darle estructura a los datos como en [19], realizamos el trabajo de minado de texto con el uso de R y utilizando herramientas que proporciona el paquete tm en R. Se muestran métodos de creación y manipulación de corpus, procesamientos de los textos y creación de matrices de documentos.

La estructura fundamental llamada corpus se ven como objetos que almacenan colecciones de documentos, la definición de los corpus permite realizar operaciones con estos, junto con sus documentos, además proporciona accesibilidad a los documentos y a los metadatos.

```
tw_corpus<-VCorpus(VectorSource(tw_datos$text)
```

Transformación de los datos.

Una vez que tenemos el corpus definido, es necesario tener uniformidad en el texto, esto puede ser visto como un análisis de texto, aquí descubrimos información relevante que posiblemente no está a la vista, con el procesamiento de texto, extraemos palabras relacionadas, eliminamos ambigüedades como palabras de uso frecuentes signos o símbolos, damos uniformidad a los mensajes textos, hacemos limpieza en los datos, en fin transformamos en un formato adecuado para la fase de entrenamiento del modelo. Para ello usamos las funciones dentro de tm_map(), que aplican a todos los

elementos del corpus. Para terminar la fase de preparación vamos a dividir el mensaje en términos individuales mediante tokenization , un token es un elemento simple de texto o simplemente una palabra, a partir de esto se genera una matriz de frecuencia donde se indica la frecuencia de cada palabra. En esta etapa usaremos una función llamada DocumentTermMatrix() la cual toma un corpus y crea una estructura llamada DTM en el cual sus filas son los SMS (documentos o mensaje) y en las columnas están las palabras (o términos) y las frecuencias de cada palabra por mensaje (o documento).

```
tw_dtm2<-DocumentTermMatrix(tw_corpus,control = list(
 tolower=TRUE,
 removeNumbers=TRUE,
 stopwords=TRUE,
 removePunctuation=TRUE,
 stemming=TRUE,
 stripWhitespace= TRUE
))
```

Un vistazo a una parte de la matriz DTM obtenida nos muestra su estructura, los 7613 documentos (mensajes) y los 18394 términos (palabras) totales de los mensajes y las frecuencias de estas en cada mensaje.

```
<<DocumentTermMatrix (documents: 7613, terms: 18394)>>
Non-/sparse entries: 70234/139963288
Sparsity           : 100%
Maximal term length: 51
Weighting          : term frequency (tf)
Sample             :

Terms
Docs  amp bomb fire get just like new now via will
 1541   0    2    0   0    0    0   0   0   0    0
 1832   1    0    0   0    0    0   0   0   0    0
 1910   0    0    0   0    0    0   0   0   0    0
 1943   0    0    0   0    0    0   0   1   0    0
 2149   0    0    0   1    1    0   0   0   0    0
 4189   0    0    1   0    0    0   0   0   0    0
 4199   0    0    0   0    0    0   0   0   0    0
 4458   0    0    0   0    0    0   0   0   0    0
 5981   2    0    0   0    0    2   0   0   0    0
 6129   0    0    0   0    0    0   0   0   1    0
```

Entonces se separan los datos en dos conjuntos (tw_dtm_train) con un 75% para entrenar el modelo y otro (tw_dtm_test) con el restante 25 % de los datos para evaluar

la precisión del modelo obtenido.

Conjunto de entrenamisento:
tw_dtm_train<-tw_dtm2[1:5710,]
tw_dtm_test<-tw_dtm2[5711:7613,]
Reducir la dimensión.

Seguido se remueven las palabras con baja frecuencia en cada uno de los conjuntos de entrenamiento y el de evaluación. Eliminamos entonces cualquier palabra que aparezca en menos de 5 mensaje, así se reducirá significativamente las matrices y no afecta relaciones inherentes de la matriz.

El clasificador que usamos en R está basado en el clasificador Bernoulli Naive-Bayes, el cual trabaja con variables categóricas, así que generamos una matriz que asigna "no" en las casillas de frecuencia cero y "si" a las casillas de frecuencia distinta de cero en la matriz de frecuencias obtenidas anteriormente. Tanto para el conjunto de entrenamiento y de evaluación.

```
conver_counts<-function(x){x<-ifelse(x>0,"Yes","No")}
tw_train1<-apply(tw_dtm_freq_train,MARGIN = 2,conver_counts)
```

Naive Bayes con R.

Así teniendo los conjuntos de las filas de mensajes convertidos en formato que puede ser representado por un modelo estadístico, apliquemos el algoritmo Naive Bayes del paquete e10741 al conjunto de entrenamiento para obtener el modelo de predicción. En [20] se ilustra una forma de hacerlo.

Este utilizará la presencia o ausencia de las palabras para estimar la probabilidad que un mensaje dado sea una emergencia verdadera o no.

```
>tw_classifier<-naiveBayes(twF_train,tw_train_label)
  0    1
3296 2414
```

Evaluación del modelo.

Para evaluar el clasificador que obtuvimos, usaremos los mensajes en el conjunto de prueba junto con las con las etiquetas almacenadas en el vector de prueba. Así generamos una predicción, entonces comparamos las clases pronosticadas por el modelo con la clasificación real utilizando una tabla de valoración cruzada y determinamos las métricas de exactitud, sensibilidad y especificidad.

```
tw_test_pred<-predict(tw_classifier,twF_test)
str(tw_test_pred)
CrossTable(tw_test_pred,tw_test_label,prop.chisq = FALSE,prop.c = FALSE,
        prop.r = FALSE,
        dnn = c("predic","actual") )
```

	actual		
predic	0	1	Row Total
0	834	271	1105
	0.438	0.142	
1	212	586	798
	0.111	0.308	
Column Total	1046	857	1903

Así la clase positiva la definimos como (857/1903) y la clase negativa (1046/1903) como los mensajes

- Verdaderos positivos: clasificados correctamente en la clase de interés 586
- Verdaderos negativos: clasificados correctamente en la clase de no interés 834
- Falsos positivos: clasificado incorrectamente en la clase de interés 271
- Falsos negativos: clasificado incorrectamente en la clase de no interés 212

Usando las ecuaciones (14), (15) y (16) se tienen los siguientes resultados:

Exactitud=0.746

Sensibilidad=0.734

Especificidad=0.754

AUC: Area under the curve= 0.7406, este valor representa el área bajo la curva en la figura 3.

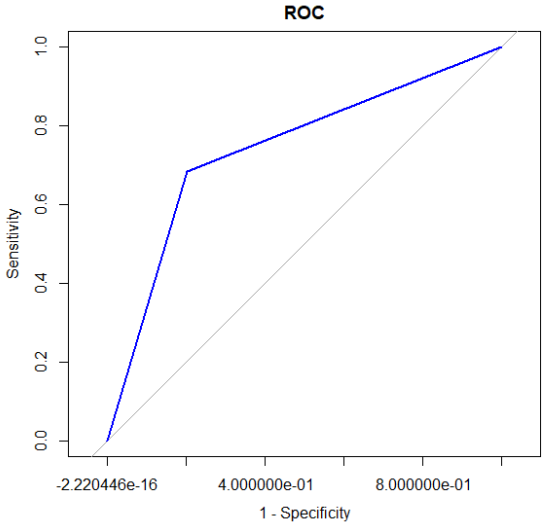

Figura 3. área bajo la curva

Conclusiones

Se notó como la utilización de método de Naive-Bayes para la detección de emergencias o catástrofes reales basadas en los mensajes de X (Twitter) genera un modelo según la evaluación capaz de clasificar una emergencia real con una proporción de 73.4% entre las clasificadas como emergencias (sensibilidad) y clasifica las emergencias falsas con una precisión de 75.4% (especificidad) entre las clasificada como falsa. En general el modelo obtenido tiene una exactitud de 74.6% en sus pronósticos.

Observamos también que el modelo se equivoca 11.1% de la veces al clasificar como emergencias a mensajes que realmente no lo son. También lo hace 14.1% de la veces con los que realmente son emergencias. El AUC o área bajo la curva con un valor de 0.74 aproximadamente nos confirma un nivel aceptable en el performance del modelo para ´predecir las emergencias reales y las no reales.

La utilización de modelos de machine learning para la seguridad, riesgo, gestión de situaciones de emergencias y desastre ha venido en aumento, en la actualidad cada vez es más frecuente el uso del análisis de datos y pronostico para la toma de decisiones, por las implicaciones claras que acarrea buenas o malas decisiones, siendo un factor decisivo en la gestión de situaciones de emergencias. Con este trabajo mostramos el uso de uno de los algoritmos en tales situaciones, donde la información de las redes sirve como fuente de dato. Esto demuestra un campo naturalmente creciente donde la aplicación de modelos y las técnicas de machine learning pueden aportar a soluciones de problemas en estos casos. El uso de diferentes modelos con la finalidad de determinar cual tiene mejor performance en la clasificación podría resultar en un prototipo útil en la

solución de problemas de esta clase.

Es claro que con el aprovechamiento del desarrollo del machine learning y la ciencia de datos se están abriendo oportunidades y retos en estos trabajos y en consecuencia tendrá nueva información que derive en predicciones con mejor exactitud. Nuevos métodos más complejos son construidos, más escalables para grandes cantidades de datos sobrepasando los modelos convencionales. En definitiva el uso de métodos de machine learning puede contribuir a mejorar la gestión de situaciones de emergencias, apoyando el análisis y la toma de decisiones. Además como herramienta que permite optimizar los recursos en tales situaciones.

Referencias

[1] A. Kansal, Y. Singh, N. Kumar y V. Mohindru, "Detection of forest fire using Machine Learning technique", *Journal of an university of Architecture & Technology,* 2020

[2] V. Chamorro, *Clasificación de tweets mediante modelos de aprendizaje supervisado*, Madrid, España: Universidad complutense, 2018

[3] J. Marin, *Bayesian Essentia*l *with R*, N.Y. USA: Springer, 2014

[4] M. Yasser, Machine Learning and Data Science Community, "Predicting tweeter sentiment", 2021. [Online]. Avalaivable: https://www.kaggle.com/code/yasserh/predicting-twitter-sentiments-top-ml-models

[5] F. Harrell, *Regression Modeling Strategies*, Nashville, USA: Springer, 2015.

[6] O. Basheer, "Application of Naïve Bayes to students ´performance classification", *Asian Journal probability and statistics*. 2023

[7] Z. Zhang, "Sentiment Analysis of twitter comments using Naïve Bayes Classifier", *Communications in humanities research*, 2023

[8] K. Chitra, "Netflix ranking by combination of k-nearest neighbor and singular value decomposition", *International Journal of computational Science and engineering,* 2020

[9] T. Fontalvo, "Aplicación del análisis discriminante para evaluar el mejoramiento de los indicadores financieros en las empresas del sector de extracción de petróleo y gas natural en Colombia", *Soluciones del posgrado EIA*, 2011

[10] G. Castillo, "Técnica de clasificación bayesiana para identificar posible plagio en información textual", *Revista cubana de ciencias informáticas,* 2014

[11] L. Dubiau, "Análisis de sentimientos sobre un Corpus en español", Facultad de

ingeniería, Universidad Buenos Aires, 2013

[12] D. Buzic, "Lyrics Classification using Naive Bayes", *International convention on information and communication Technology*, 2018

[13] H. Ramadhan, "Sentiment analysis on Indonesia-English code-mixed data", *International conference for convergence in technology*. April 2023.

[14] D. Wackerly, *Estadística matemática con aplicaciones.* Miami Florida: Thomson, 2002.

[15] G. James, D. Witten, T. Hastie, R. Tibshirani, *An introduction to Statistical Learning,* Springer, 2021.

[16] B. Lantz., Machine Learning with R. Pack Publishing LTD, 2019. [Online]. Available:https://www.packtpub.com/product/machine-learning-with-r-third-edition/9781788295864

[17] T. Hastie, *The elements of statistical learning*, California, USA: Springer, 2008.

[18] A. Zheng, *Evaluating Machine Learning Models*, USA: O´Really, 2015.

[19] I. Feinerer, "Text mining Infrastructure in R", *Journal of statistical software,* 2008

[20] J. Bosco, Naive Bayes con R para clasificación de textos. R Pubs by Rstudio, 2018. [Online]. Available: https://rpubs.com/jboscomendoza/naive_nayes_con_r_clasificacion_texto