



# ≡ Proyecto 1- Etapa 1

Apoyo a la detección de intentos de suicidio

**Grupo 4**

Daniela Ricaurte - 201822966  
Melissa Contreras - 202011876  
Julián Mora - 202012747

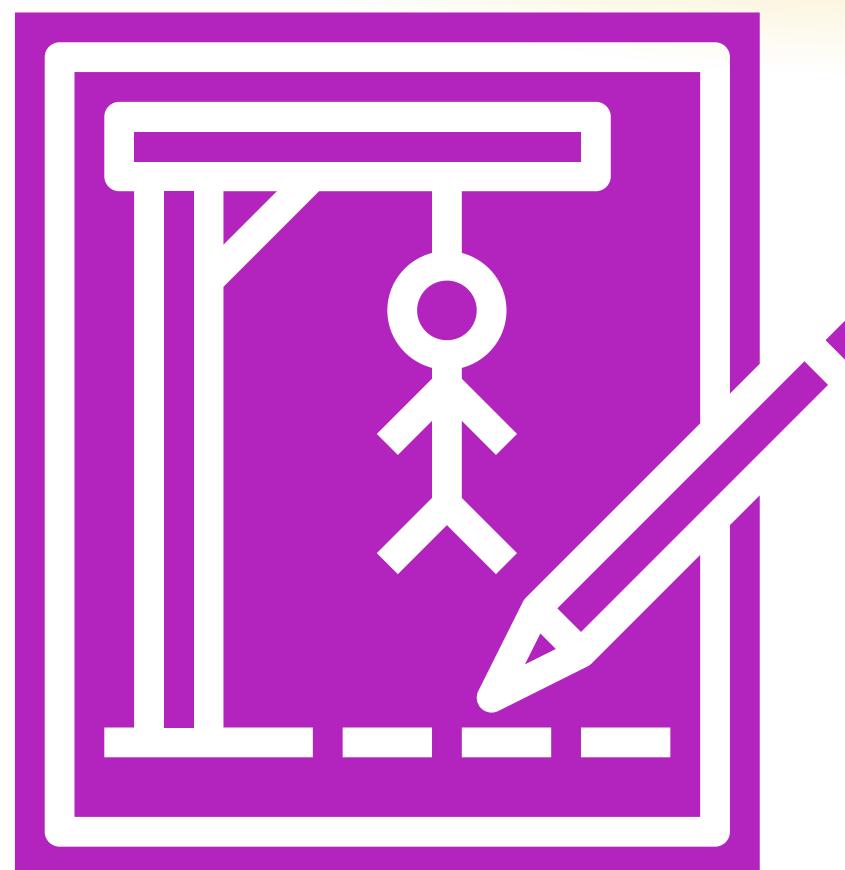
## Oportunidad

Mediante el uso de técnicas de NLP y text classification podemos indicar de los textos de redes sociales aquellos que pueden considerarse casos de suicidio para lograr dar apoyo oportuno y entender mejor esta enfermedad o incidencias.



## El problema

Incremento de casos de suicidio y deterioro de la salud mental.



Algoritmo Support Vector Machine-  
Máquinas de Vectores de Soporte

# ALGORITMO SVM

Por: Daniela Ricaurte



# FUNCIONAMIENTO

El algoritmo elige aquel hiperplano que clasifica con mayor precisión los datos en vez de maximizar el margen de distancias.

Es decir que de un lado de este indica que pertenece a la categoría y el otro que no pertenece a la categoría, es decir genera 2 subespacios.

## VENTAJAS

- Es un algoritmo que no requiere muchos datos de entrenamiento para producir resultados precisos.
- Es muy bueno en identificar una margen clara de separación de los datos.
- Es eficiente con la memoria
- Los datos no necesariamente deben de estar separados de forma lineal ya que el algoritmo cuenta con la técnica del truco de kernel, en la cual toma espacio de entrada de baja dimensión y lo transforma en un espacio de mayor dimensión, como podemos ver en las imágenes a continuación

# Metodología

1. Preparación de los datos
  - a. Limpieza
  - b. Tokenización
  - c. Normalización
  - d. Codificación de los datos
2. Vectorización de palabras
3. Entrenamiento del algoritmo



# Resultados

	precision	recall	f1-score	support
0	0.93	0.95	0.94	33042
1	0.93	0.91	0.92	25668
accuracy			0.93	58710
macro avg	0.93	0.93	0.93	58710
weighted avg	0.93	0.93	0.93	58710

# ALGORITMO Naive Bayes

Por: Julián Mora



# FUNCIONAMIENTO

El algoritmo se basa en el teorema de Bayes. Este teorema vincula la probabilidad de un evento A dado un evento B, con la probabilidad de el evento B dado el evento A.

$$\begin{aligned} P(B | A) = \\ (P(A | B) * P(B)) / (P(A)) \end{aligned}$$



Los conjuntos de datos son los siguientes:

- $P(B)$  es el a priori, la probabilidad de que la hipótesis B sea verdadera antes de la evidencia.
- $P(A|B)$  es el likelihood, probabilidad de que la hipótesis A sea verdadera dados los datos B.
- $P(A)$  es el likelihood marginal o evidencia, la probabilidad de observar los datos A.
- $P(B|A)$  es el a posteriori, la probabilidad final, en la cual B es verdadera dado A.

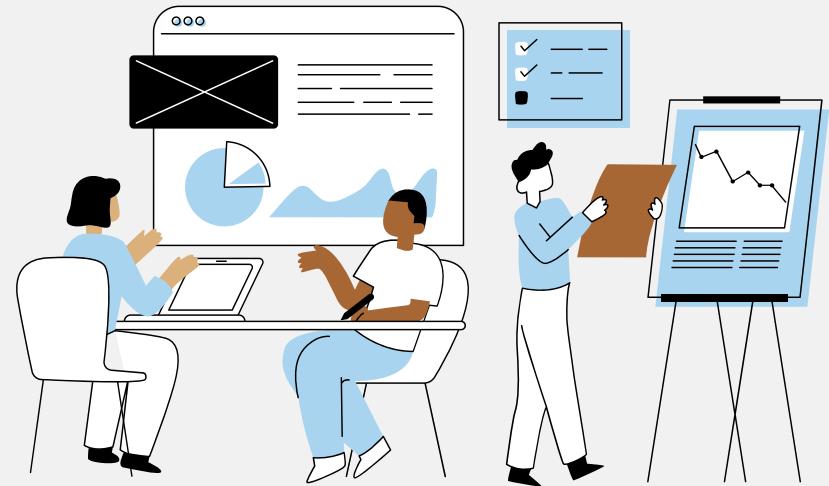


# Aplicaciones

- La aplicación que tiene el teorema de bayes a problemas donde se quiere conocer la probabilidad de las causas dados los efectos.
- Uno de los ejemplos más conocidos sobre la aplicación del teorema de Bayes es conocer la probabilidad de que un paciente con síntomas X tenga una enfermedad Y, cuando lo que normalmente se conoce es el porcentaje de pacientes de la enfermedad Y que tienen síntomas X.
- Este ejemplo puede ser escalado a nuestro caso de interés pues necesitamos identificar, a partir de las actitudes que tienen los usuarios en los foros de la fuente (síntomas X), cuales de ellos pueden estar atentando contra su integridad física (enfermedad Y).



# Resultados



```
In [52]: y_pred = classifier.predict(Test_X)  
print(classification_report(Test_Y,y_pred))
```

	precision	recall	f1-score	support
non-suicide	0.96	0.84	0.90	33024
suicide	0.82	0.95	0.88	25686
accuracy			0.89	58710
macro avg	0.89	0.90	0.89	58710
weighted avg	0.90	0.89	0.89	58710

## Pruebas

```
In [53]: examples = ['My name is Daniela','I want to kill myself']  
example_counts = vectorizer.transform(examples)  
predictions = classifier.predict(example_counts)  
predictions
```

```
Out[53]: array(['non-suicide', 'suicide'], dtype='<U11')
```



# ALGORITMO LABEL SPREADING

Por: Melissa Contreras

# FUNCIONAMIENTO

Es un algoritmo de aprendizaje semi-supervisado que consiste en crear una matriz con ayuda de un kernel RBF (Función de Base Radial) la cual será utilizada para determinar los pesos de los bordes. Esta matriz tiene ceros en su diagonal porque un borde no debe conectarse consigo mismo.

≡

Squared Euclidean distance

$$w_{ij} = e^{\frac{-||x_i - x_j||^2}{2\sigma^2}}, \text{ where } i \neq j. \quad w_{ii} = 0$$

RBF Kernel

$$W = \begin{bmatrix} 0 & w_{21} & \cdots & w_{1j} \\ w_{12} & 0 & & \ddots \\ \vdots & & & 0 \\ w_{i1} & & & 0 \end{bmatrix}$$

# FUNCIONAMIENTO

Symmetric Normalized Laplacian      Affinity matrix from step 1

$$S = D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$$

A diagonal matrix with its  $(i,i)$ -element equal to the sum of the  $i$ -th row of  $W$

Se coge la anterior matriz de afinidad y la normaliza simétricamente dando como resultado una matriz laplaciana.

- Multiplicación de matrices
- El parámetro alfa permite el soft clamping controlando proporción de información . Entonces, si alfa esta alrededor de 0, toda la información de la etiqueta original, si esta alrededor de 1 se puede reemplazar la mayor parte de información de la etiqueta original.
- Este proceso tiene una limitada cantidad de iteraciones.

Matrix F contains label vectors, where  $t$  is simply an iteration number      Hyperparameter alpha used for soft clamping      Matrix Y contains the original labels

$$F(t + 1) = \underbrace{\alpha S F(t)}_{\text{Symmetric Normalized Laplacian matrix from step 2}} + \underbrace{(1 - \alpha) Y}_{\text{First term, label information from neighbors}} + \underbrace{\alpha Y}_{\text{Second term, original label infomation}}$$

# ¡Gracias!

