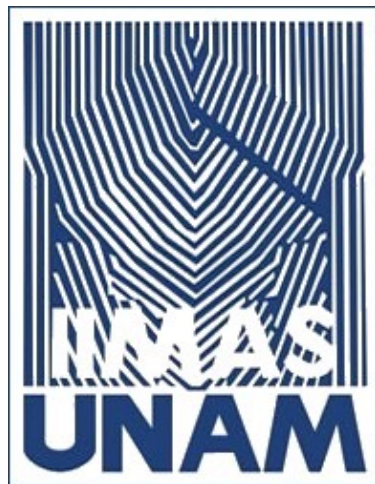


# Procesamiento de Lenguaje Natural

## Clasificación de textos



Dra. Helena Gómez Adorno

[helena.gomez@iimas.unam.mx](mailto:helena.gomez@iimas.unam.mx)

Dra. Gemma Bel

[gbele@iingen.unam.mx](mailto:gbele@iingen.unam.mx)

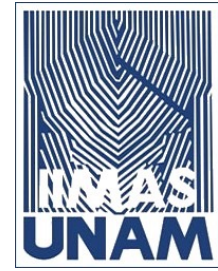


Correo del curso:

[pln.cienciadedatos@gmail.com](mailto:pln.cienciadedatos@gmail.com)

Asistente:

Luis Ramon Casillas



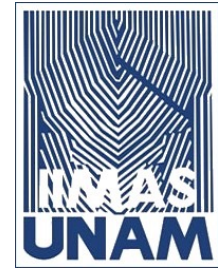
# Clasificación de textos

- Autoría
- Español vs. Inglés
- Género
- Reseñas positivas vs. negativas
- Área temática

# Género

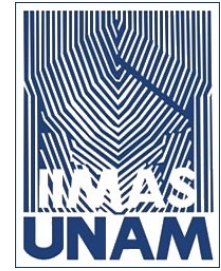


- Elaborado por un equipo de 26 científicos dirigido por el Centro de Investigación del Clima de la Universidad de Nueva Gales del Sur, el Diagnóstico demuestra de manera convincente que los efectos del calentamiento global han empeorado en los últimos tres años. (Somerville et al 2011)
- El huracán Sandy azotó a unas 290 millas de la costa del Atlántico Medio el domingo por la noche, y el Centro Nacional de Huracanes informó que se esperaba que la monstruosa tormenta llegara a la costa con vientos casi huracanados e inundaciones por marejadas ciclónicas potencialmente mortales.
- Era el día amplio: las ocho o las nueve; la tormenta furiosa, en lugar de las baterías; y alguien llamando y llamando a mi puerta. '¿Cuál es el problema?' Lloré. '¡Un naufragio! ¡Cerca! " Salté de la cama y pregunté, ¿qué naufragio? —Una goleta, de España o Portugal, cargada de frutas y vino. ¡Date prisa, señor, si quieres verla! ... "



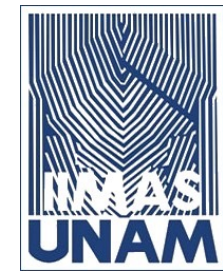
# Características

- ¿Qué palabras dan una pista sobre el género?
- ¿Qué otra información podríamos utilizar?



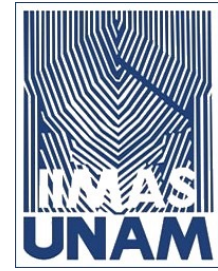
# Características

- ¿Qué palabras dan una pista sobre el género?
- ¿Qué otra información podríamos utilizar?



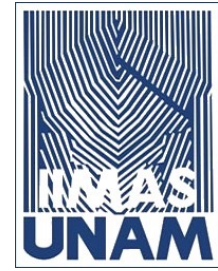
# Reseñas positivas vs. negativas

- Odio los auriculares del iphone, me lastiman los oídos
- Bueno, me encanta mi iPhone 6+. La duración de la batería es simplemente increíble. Tener un 40% al final del día es genial.
- Recibí un correo electrónico de el director esta mañana y él ha exigido que se terminen 10 créditos de esta beca para el lunes.
- ¿Amas lo que haces de por vida? Lo hago y me divierto :-)



# Características

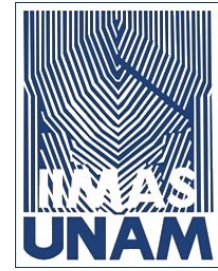
- ¿Qué palabras dan una pista sobre el sentimiento?
- ¿Qué otra información podríamos utilizar?



# Otras tareas de clasificación en PLN

- Etiquetado de categorías gramaticales
- Análisis de dependencias sintácticas (parsing)
- Desambiguación del sentido de las palabras
- Datos demográficos en las redes sociales (por ejemplo, género)
- Detección de entidades nombradas





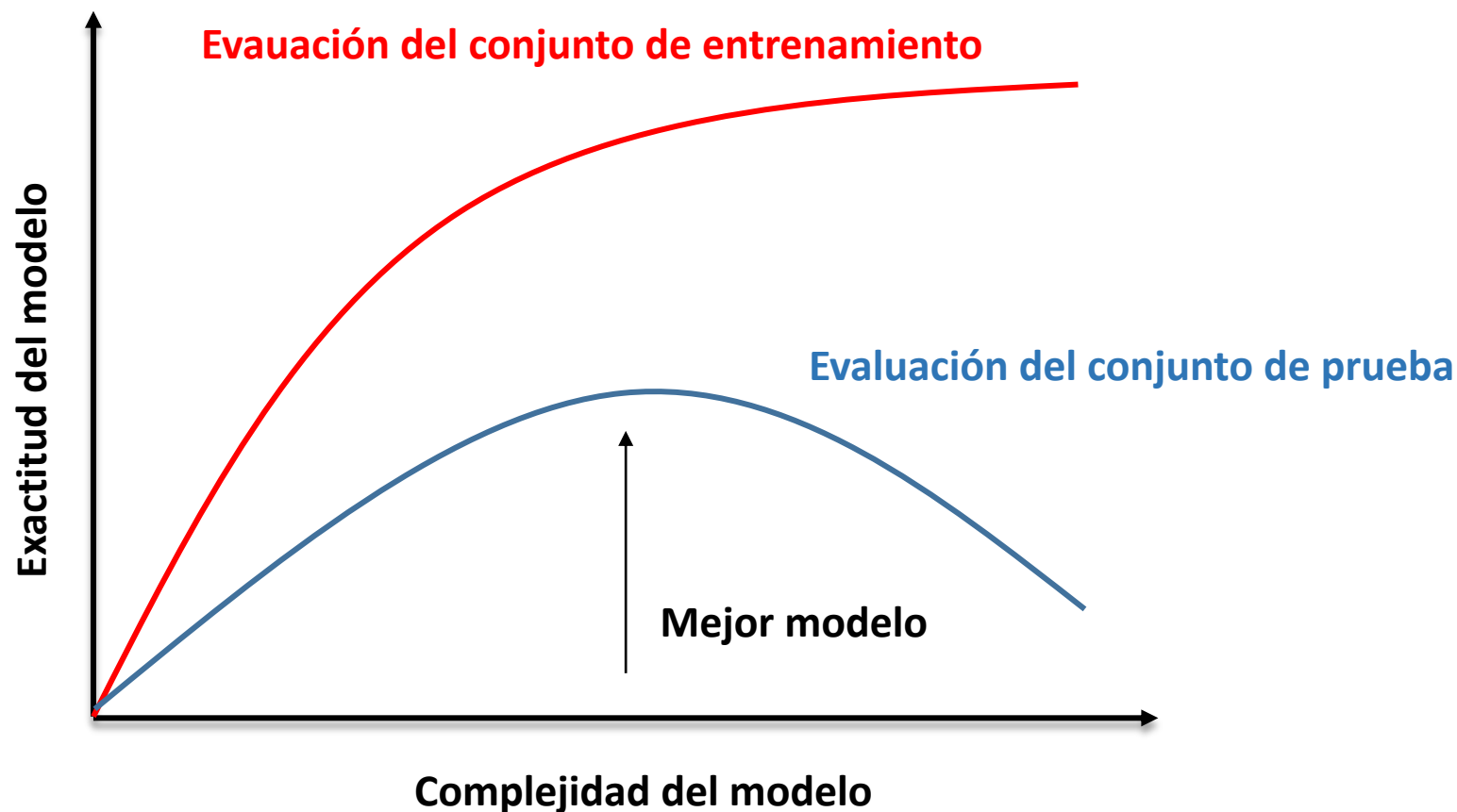
# Tipos de clasificación

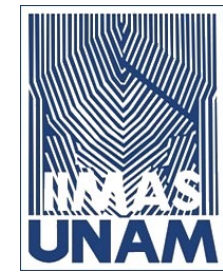
- Binario (p. ej., Adulto? Verdadero, Falso)
- Clases múltiples (p. ej., edad: niño, adolescente, adulto)
- Multi-etiqueta (p. ej., niño, mexicano)
- Resultado de la agrupación en clústeres (sin etiquetas)

# Generalización, sobreajuste y falta de ajuste

- **Generalización**: capacidad de un algoritmo de proporcionar predicciones precisas para datos nuevos que no se habían visto anteriormente.
- Supuestos:
  - Los **datos futuros no vistos** tendrán las mismas propiedades que los conjuntos de entrenamiento actuales.
  - Se espera que los modelos que son precisos en el conjunto de entrenamiento sean precisos en el conjunto de prueba.
  - Pero eso puede no ocurrir si el modelo entrenado se ajusta muy específicamente al conjunto de entrenamiento.
- Modelos demasiado complejos para la cantidad de datos de entrenamiento **se sobreajustan** y no generalizarán bien a los nuevos ejemplos.
- Modelos demasiado simples, que ni siquiera funcionan bien en los datos de entrenamiento, les **falta ajuste** y tampoco generalizarán bien.

# La relación entre la complejidad del modelo y el rendimiento en entrenamiento y prueba





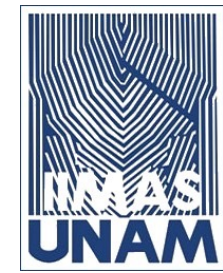
# Clasificadores generativos vs. discriminatorios

- **Generativo**

- Aprenden la distribución de probabilidad conjunta  $P(\mathbf{x}, \mathbf{y})$ .
- La atención se centra en cómo las características y la variable de destino ocurren juntas:  $P(\mathbf{x}, \mathbf{y})$ .
- Para predecir, la regla de Bayes permite transformar a una distribución condicional:  $P(\mathbf{y}|\mathbf{x})$ .

- **Discriminativo**

- Proporcionar un modelo para la variable objetivo.
- Intentan encontrar límites que separen clases.
- Límites pueden ser estrictos o blandos según el algoritmo.
- Inferir salidas basadas en entradas:  $P(\mathbf{y}|\mathbf{x})$



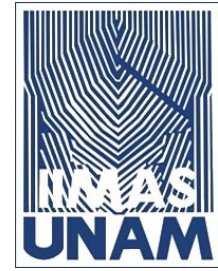
# Clasificadores generativos vs. discriminativos

- **Generativo**

- Aprenden la distribución de probabilidad conjunta  $P(\mathbf{x}, \mathbf{y})$ .
- La atención se centra en cómo las características y la variable de destino ocurren juntas:  $P(\mathbf{x}, \mathbf{y})$ .
- Para predecir, la regla de Bayes permite transformar a una distribución condicional:  $P(\mathbf{y}|\mathbf{x})$ .

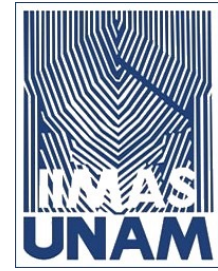
- **Discriminativo**

- Proporcionar un modelo para la variable objetivo.
- Intentan encontrar límites que separen clases.
- Límites pueden ser estrictos o blandos según el algoritmo.
- Inferir salidas basadas en entradas:  $P(\mathbf{y}|\mathbf{x})$



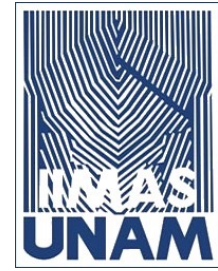
# Representación de Bolsa de Palabras BOW

- Vector para representar todas las palabras posibles
  - Valor = número de veces que la palabra aparece en el documento
  - Unigramas
- Vector o matriz para representar todos los posibles pares de palabras
  - Valor = número de veces que aparece bigrama en el documento
  - Bigramas
- **Bolsa** porque hemos descartado información sobre el orden, la sintaxis, los límites de las oraciones, los párrafos



# Representación de vectores de características

- Unigramas
- Bigramas
- N-gramas
- Adición de características de meta-nivel
  - Longitud del documento, longitud de la oración, posición de la oración, posición de la palabra, nombre del autor



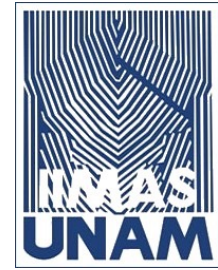
# Representación de vectores de características

- ¿Qué tan grande será el vector de unigramas?
- ¿Qué tan grande será el vector de bigramas?



# Preprocesamiento

- Al preprocesar, se puede realizar lo siguiente:
  - Eliminar menciones y URL
  - Tokenizar la cadena en palabras.
  - Eliminar las palabras vacías como "y, es, a, sobre, etc."
  - Derivar o convertir cada palabra en su raíz. Como bailarina, bailando, bailada, se convierte en “baila”. Se puede utilizar el ***porter stemmer*** para solucionar este problema.
  - Convertir todas las palabras a minúsculas.



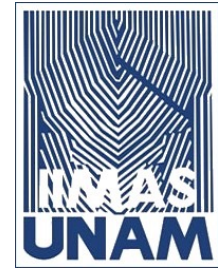
# Ejemplo

- Por ejemplo, el siguiente tweet "*@Usuario2 y @Usiario1 están realizando un GRAN curso de PLN en <https://iimas.unam.mx> !!!*" después de que el preprocesamiento se convierte en  
[está, realiza, gran, curso, PLN].
- Pueden ver cómo eliminamos las menciones, los convertimos en palabras, eliminamos las palabras vacías, realizamos la derivación y convertimos todo a minúsculas.

# Representación de bolsa de palabras

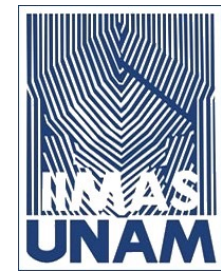
El huracán Sandy azotó a unas 290 millas de la costa del Atlántico Medio el domingo por la noche, y el Centro Nacional de Huracanes informó que se esperaba que la monstruosa tormenta llegara a la costa con vientos casi huracanados e inundaciones por marejadas ciclónicas potencialmente mortales.

el	3
huracán	1
sandy	1
azotó	1
a	2
equipo	0
efectos	0
millas	1
de	1



# Pesos

- **Reseñas** positivas vs negativas
  - Sorprendente vs. maravilloso vs. horrible
- **Género**: twitter vs. novelas vs. artículos de revistas científicas
  - Exclamado vs. neurotransmisor vs. LOL



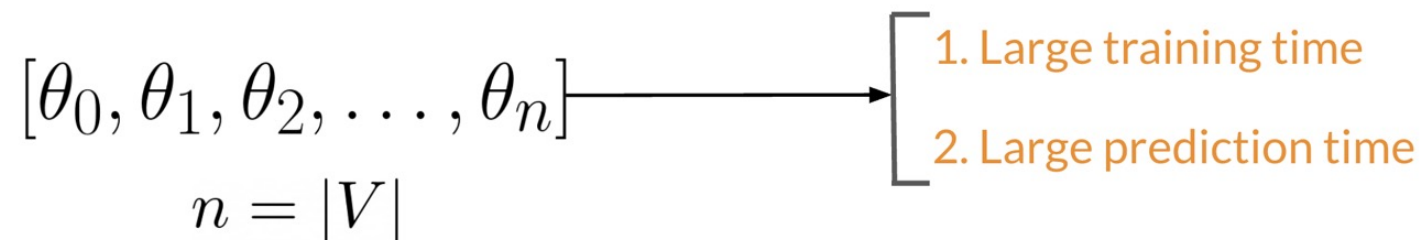
# Aprender pesos a partir de datos

- Datos de entrenamiento (corpus)
  - Texto etiquetado
  - Oraciones o párrafos extraídos de fuentes (en línea)
    - Twitter
    - Novelas
    - Artículos
    - Noticias
- Vector de características: bolsa de palabras más predicción de salida

# Vocabulario y extracción de características

- Dado algún texto, puede representarlo como un vector de dimensión  $V$ , donde  $V$  corresponde al tamaño de su vocabulario. Si tuviéramos el tweet "*I am happy because I am learning NLP*", entonces pondríamos su frecuencia en el índice correspondiente para cualquier palabra en el tweet, y un 0 en caso contrario.
- A medida que  $V$  aumenta, el vector se vuelve más disperso. Además, terminamos teniendo muchas más características y terminamos entrenando más parámetros de  $\theta$ .

I am happy because I am learning NLP



# Extracción de características con frecuencia

- Dado un corpus con tweets positivos y negativos de la siguiente manera:

Positive tweets

I am happy because I am learning NLP

I am happy

Negative tweets

I am sad, I am not learning NLP

I am sad

- Tenemos que codificar cada tweet como un vector. Anteriormente, este vector era de dimensión  $V$ .
- Podemos representar con un vector de dimensión 3. Para ello, debemos crear un diccionario para mapear la palabra, y la clase en donde apareció (positivo o negativo) a la cantidad de veces que esa palabra apareció en su clase correspondiente.

# Extracción de características con frecuencia

I am sad, I am not learning NLP

Vocabulary	PosFreq (1)	NegFreq (0)
I	3	3
am	3	3
happy	2	0
because	1	0
learning	1	1
NLP	1	1
sad	0	2
not	0	1

I am sad, I am not learning NLP

$$X_m = [1, \sum_w freqs(w, 1), \sum_w freqs(w, 0)]$$

$$X_m = [1, \sum_w freqs(w, 1), \sum_w freqs(w, 0)]$$



# Extracción de características con frecuencia

I am sad, I am not learning NLP

Vocabulary	PosFreq (1)	NegFreq (0)
I	3	3
am	3	3
happy	2	0
because	1	0
learning	1	1
NLP	1	1
sad	0	2
not	0	1

$$X_m = [1, \sum_w freqs(w, 1), \sum_w freqs(w, 0)]$$

8

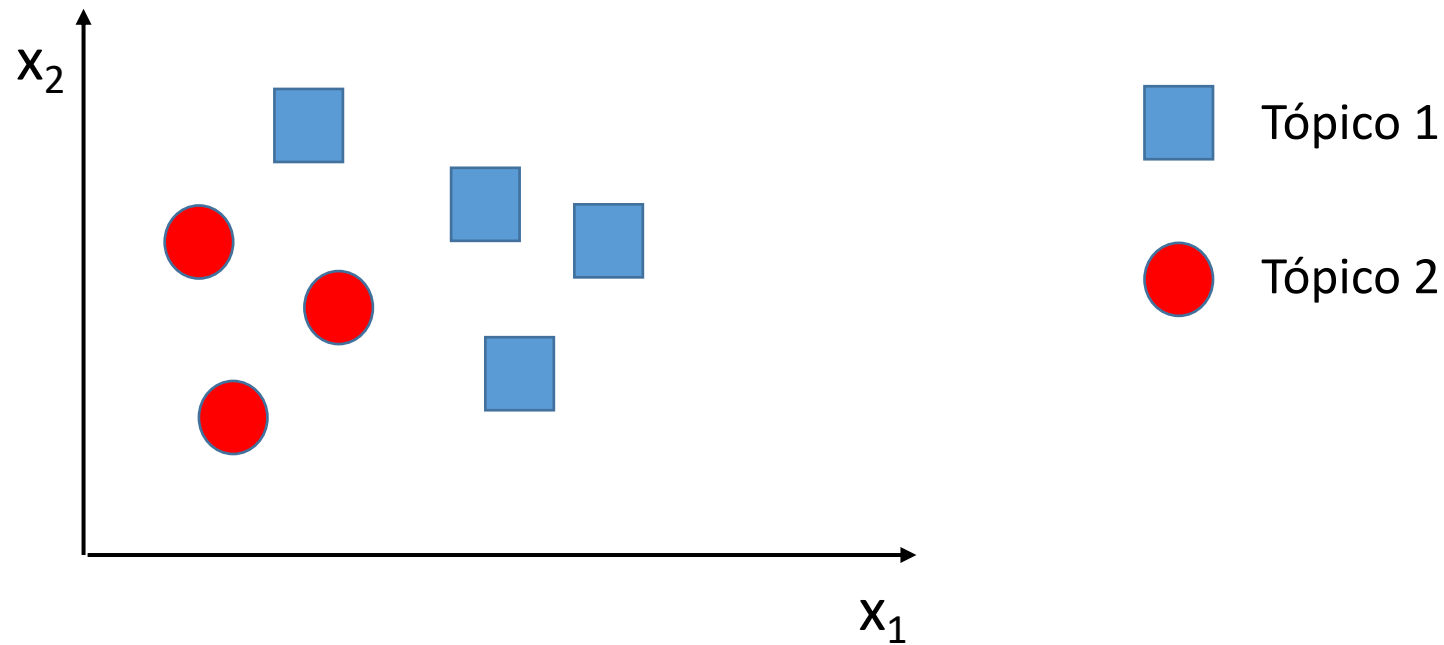
I am sad, I am not learning NLP

$$X_m = [1, \sum_w freqs(w, 1), \sum_w freqs(w, 0)]$$

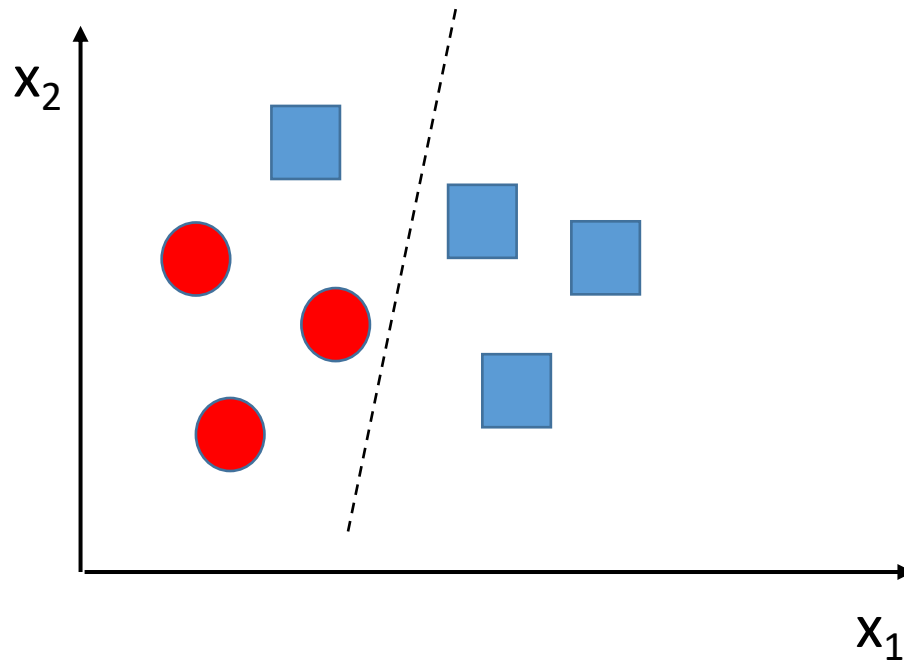
11

$$X_{[I \text{ am sad, I am not learning NLP}]} = [1, 8, 11]$$

# Clasificación del espacio vectorial



# Límite lineal



■ Tópico 1

● Tópico 2

# Representación de vectores de características

$x_1$	el	3
$x_2$	huracán	1
$x_3$	sandy	1
$x_4$	azotó	1
$x_5$	a	2
$x_6$	equipo	0
$x_7$	efectos	0
$x_8$	millas	1
$x_9$	de	1
$y_1$	<b>novela</b>	<b>0</b>
$y_2$	<b>noticia</b>	<b>1</b>
$y_3$	<b>Revista Científica</b>	<b>0</b>



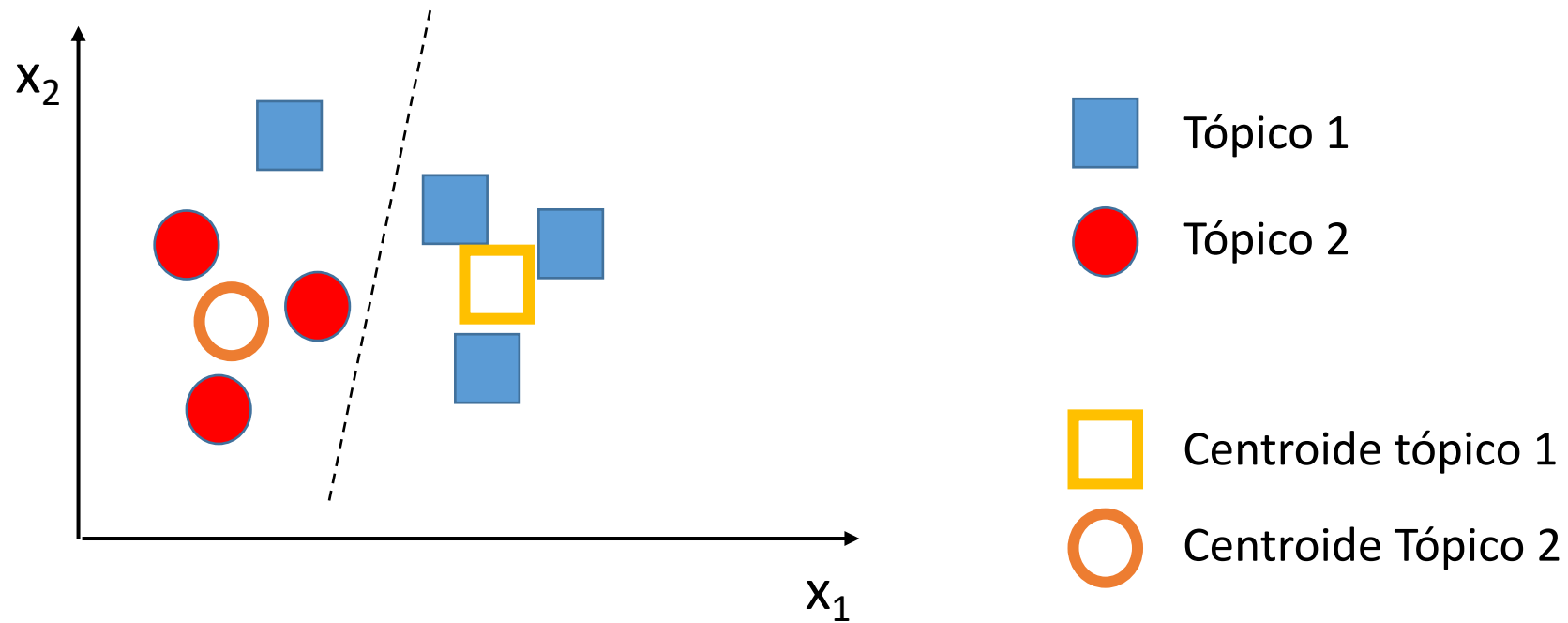
El huracán Sandy azotó a unas 290 millas de la costa del Atlántico Medio el domingo por la noche, y el Centro Nacional de Huracanes informó que se esperaba que la monstruosa tormenta llegara a la costa con vientos casi huracanados e inundaciones por marejadas ciclónicas potencialmente mortales.

One-hot vector representation

# Clasificación mediante centroides

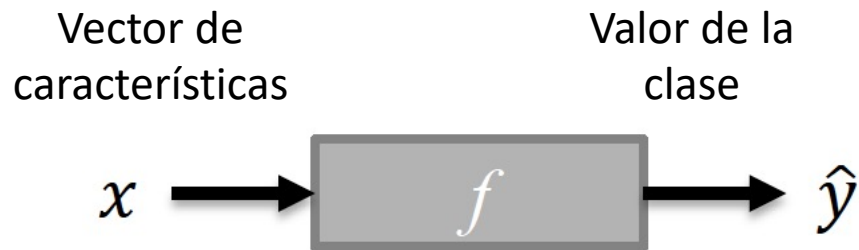
- *Centroide*
  - El punto más representativo de una clase.
- Calcule el *centroide* encontrando el promedio vectorial de miembros de clase conocidos
- El límite de decisión es una línea equidistante de los dos *centroides*
- El nuevo documento en un lado de la línea va en la **clase 1** y en el otro lado va en la **clase 2**

# Clasificación usando centroides



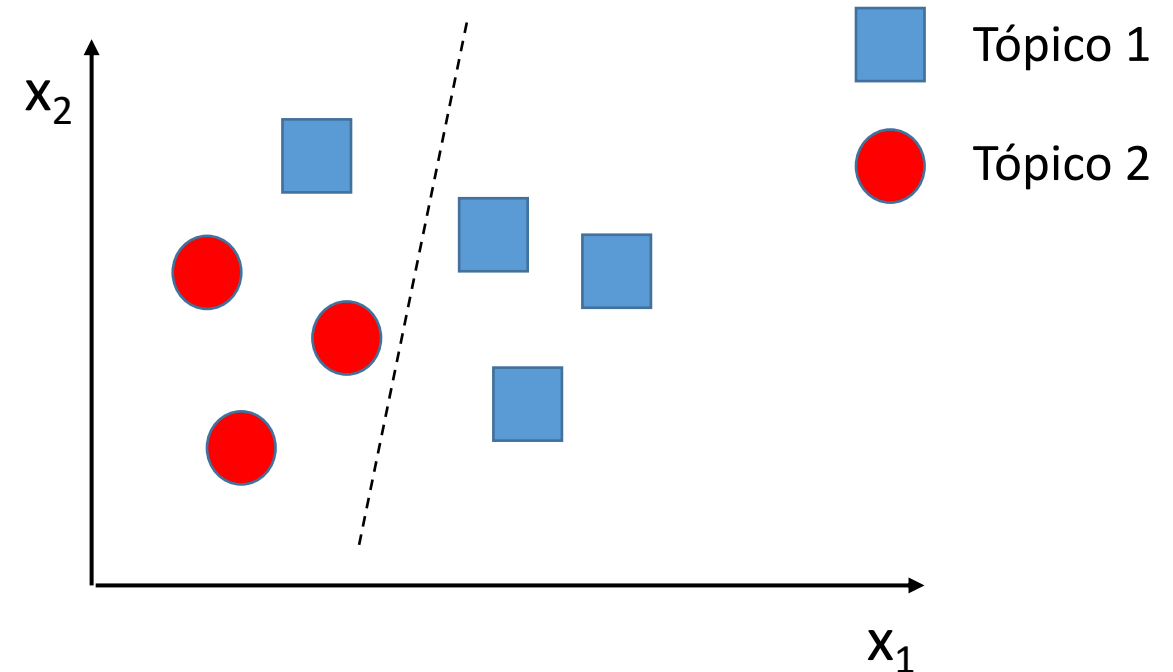
# Separadores lineales

- Cómo separamos estos dos grupos de ejemplo de entrenamiento con una línea recta?



$$f(x, w, b) = \text{sign}(w \circ x + b)$$

$$= \text{sign}(\sum w[i]x[i] + b)$$



$$f(x) = \Theta X + b$$

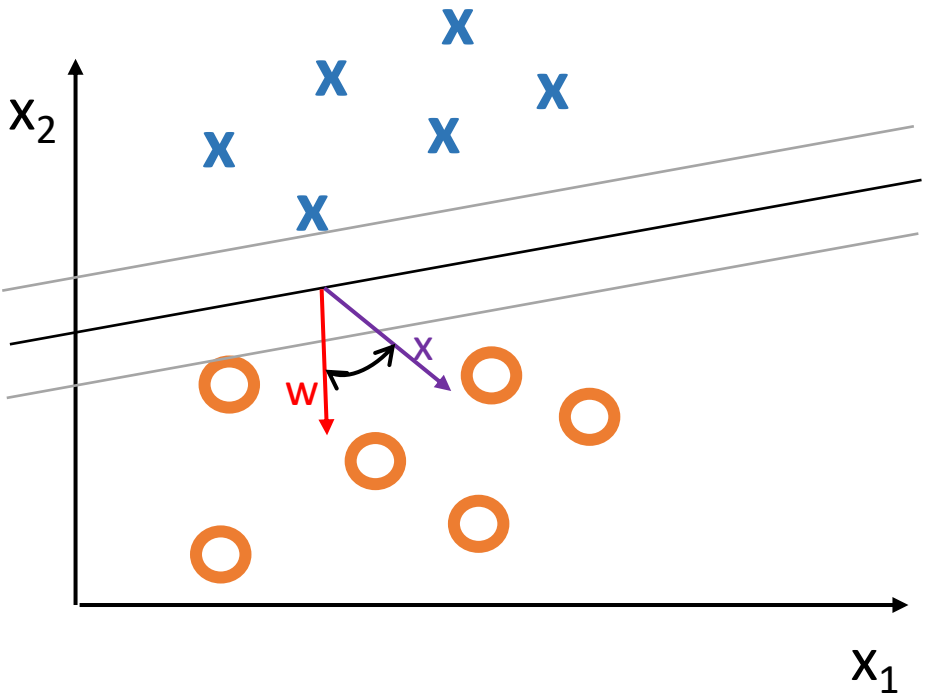
Espacio bidimensional:  $w_1x_1 + w_2x_2 = b$

En espacios n-dimensionales:  $\Theta X = \sum_{i=1}^n w_i x_i = w_1x_1 + w_2x_2 + \dots + w_nx_n$

Se puede agregar  $w_0=1, x_0=b$  para tomar en cuenta el sesgo

Se pasa la salida de  $f(x)$  por la función de signo, mapeando valores negativos a -1 y valores positivos a 1.

# Límite de decisión





# Límite de decisión

## Ejemplo

$$\Theta X = b$$

Asumimos  $b = 0$

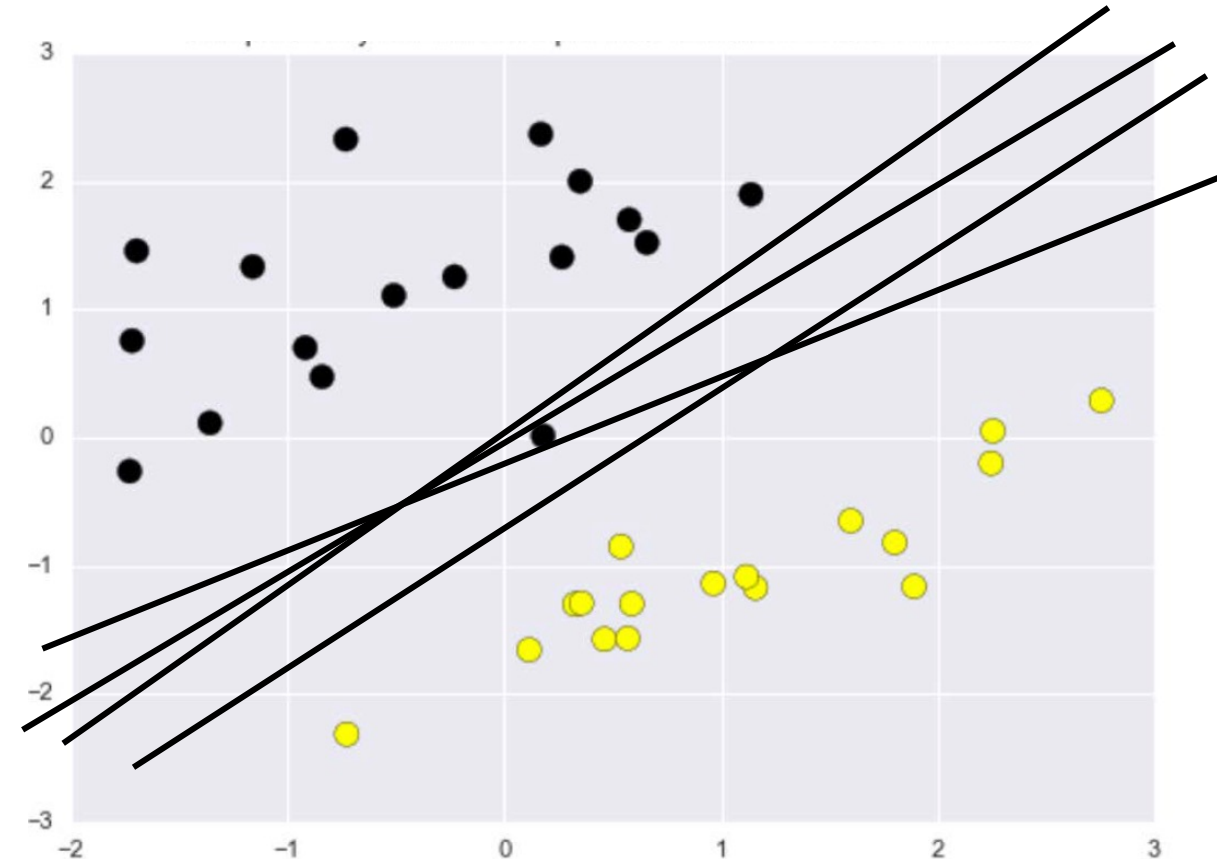
“El huracán Sandy azotó a unas”

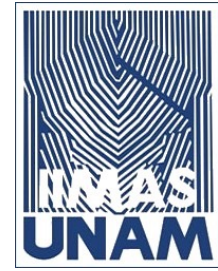
$$\Theta X = .2 * 1 + 2 * 1 + .5 * 1 + .3 * 2 + 1 * 1 + .5 * 1 = 4.8 > 0$$

Noticia		otro	
el	.2	el	-.2
huracán	1	huracán	-.5
sandy	.5	sandy	-.7
azotó	.3	azotó	-.8
a	1	a	-.1
equipo	.7	equipo	-.8
efectos	.5	efectos	-.5
millas	.5	millas	-.5
de	.3	de	-.3

# Cómo encontrar el límite lineal?

- ¿Cómo encontrar el vector de pesos  $\Theta$ ?
- Muchos métodos
  - Perceptrón
  - Árboles de decisión
  - Regresión logística
- Problema
  - Hay un número infinito de límites lineales si las clases están separadas linealmente
  - Margen máximo: máquinas de vectores de soporte (svm)





# Entrenamiento usando optimización

- Seleccionar valores para  $w$
- Calcular  $f(x)$
- Comparar la salida de  $f(x)$  con las etiquetas y calcular la pérdida
- Ajustar  $w$



# Usando una función de pérdida

- Datos de entrenamiento
- $x_1 x_2 \dots x_n$  (entrada)
- $y_1 y_2 \dots y_n$  (etiquetas)
- Algoritmo que devuelve  $f(x)$  con predicciones  $\hat{y}$
- Función de pérdida  $L(\hat{y}, y)$
- Parámetros de la función aprendida  $(\Theta, b)$  configurados para minimizar  $L$

# Pérdida

- Dado: conjunto de entrenamiento etiquetado,  $L$ , función parametrizada  $f(x, \Theta)$  donde  $\Theta = w_1 \dots w_n, b$
- Pérdida en todo el corpus = pérdida promedio en todos los ejemplos de entrenamiento:

$$L(\theta) = \frac{1}{n} \sum_{i=1}^n L(f(x_i; \theta), y_i)$$

- Conjuntos de algoritmos de entrenamiento  $\Theta$  para minimizar  $L$ :

$$\theta = \underset{\theta}{\operatorname{argmin}} L(\theta) = \underset{\theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^n L(f(x_i; \theta), y_i)$$

- Para evitar el sobreajuste, se agrega un parámetro de regularización  $R(\Theta)$ :

$$\theta = \underset{\theta}{\operatorname{argmin}} \frac{1}{n} \sum_{i=1}^n L(f(x_i; \theta), y_i) + \lambda R(\theta)$$

# Funciones de pérdida

- Puede ser una función arbitraria que mapea dos vectores a un escalar
- Pérdida de bisagra (Hinge loss) (binaria)
  - Salida del clasificador:  $\hat{y}$ , datos etiquetados:  $y$  (-1,1)
  - $\hat{Y}$  es correcto si  $\hat{y} * y > 1$  (la regla de clasificación es el signo ( $\hat{y}$ ))

$$L_{\text{hinge}(\text{binary})}(\hat{y}, y) = \max(0, 1 - \hat{y} \cdot y)$$

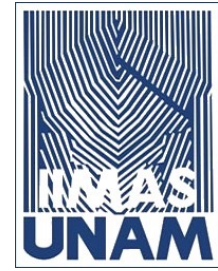
- La pérdida es 0 cuando  $\hat{y}$ ,  $y$  comparten el mismo signo  $|\hat{y}| = 1$ , lineal en caso contrario
- Intenta lograr una clasificación correcta con un margen de  $\leq 1$

# Hinge Loss (multi-clase)

- Salida del clasificador:  $\hat{Y} = \hat{y}_1 \dots \hat{y}_n$
- $Y$  es un vector one-hot para la clase correcta
- Predicción =  $\operatorname{argmax}_i \hat{y}_i$

$$L_{\text{hinge}(\text{multiclass})}(\hat{y}, y) = \max(0, 1 - \hat{y}_{[t]} \cdot \hat{y}_{[k]})$$

- donde  $t$  es la clase correcta y  $k$  es la clase de puntuación más alta donde  $k \neq t$
- Califica la clase correcta por encima de todas las demás clases con un margen de al menos 1



# Regularización

- Considerar el caso en el que uno o más documentos están mal etiquetados
  - El texto de una novela puede estar mal etiquetado como una red social si se publica como una cita.
- El clasificador intentará aprender pesos que promuevan palabras características de novelas como predictores de redes sociales.
- El sobreajuste también puede ocurrir cuando los documentos de las redes sociales en el conjunto de entrenamiento no son representativos.



# Dos regularizadores comunes

- Regularización L2

- Mantiene baja la suma de cuadrados de los valores de los parámetros

$$R_{L2}(W) = ||W||_2^2 = \sum_{i,j} (W_{[i,j]})^2$$

- Deterioro de peso o antes de Gauss (aquí W son pesos sin incluir b)
  - Prefiere disminuir el parámetro con alto peso en 1 que 10 parámetros con bajo peso

- Regularización L1

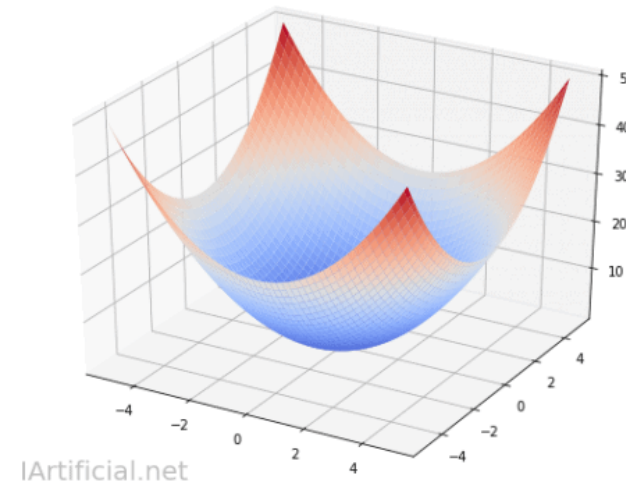
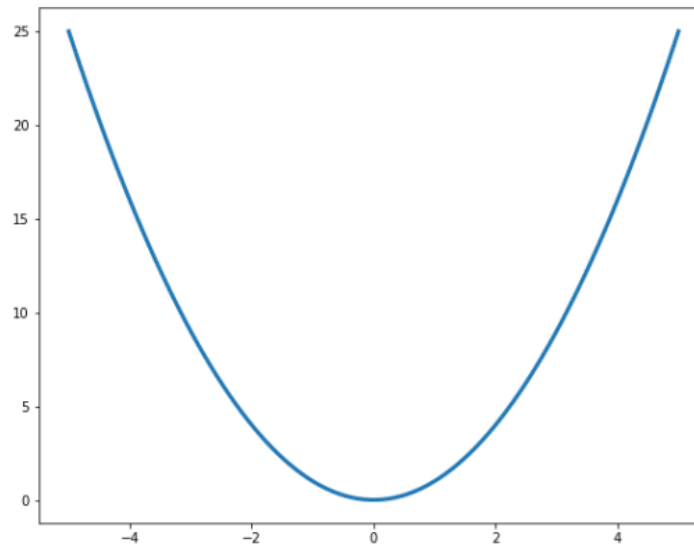
- Mantiene baja la suma del valor absoluto de los parámetros

$$R_{L1}(W) = ||W||_1 = \sum_{i,j} |W_{[i,j]}|$$

- Castiga uniformemente características con valor alto y bajo

# Optimización basada en gradiente

- Repetir hasta  $L < \text{margen}$
- Calcular  $L$  sobre el conjunto de entrenamiento
- Calcular gradientes de  $\Theta$  con respecto a  $L$
- Mover los parámetros en la dirección opuesta al gradiente



IArtificial.net

# Descenso del gradiente estocástico

---

**Algorithm 1** Online Stochastic Gradient Descent Training

---

*Input:*

- Function  $f(\mathbf{x}; \Theta)$  parameterized with parameters  $\Theta$ .
- Training set of inputs  $\mathbf{x}_1, \dots, \mathbf{x}_n$  and desired outputs  $\mathbf{y}_1, \dots, \mathbf{y}_n$ .
- Loss function  $L$ .

---

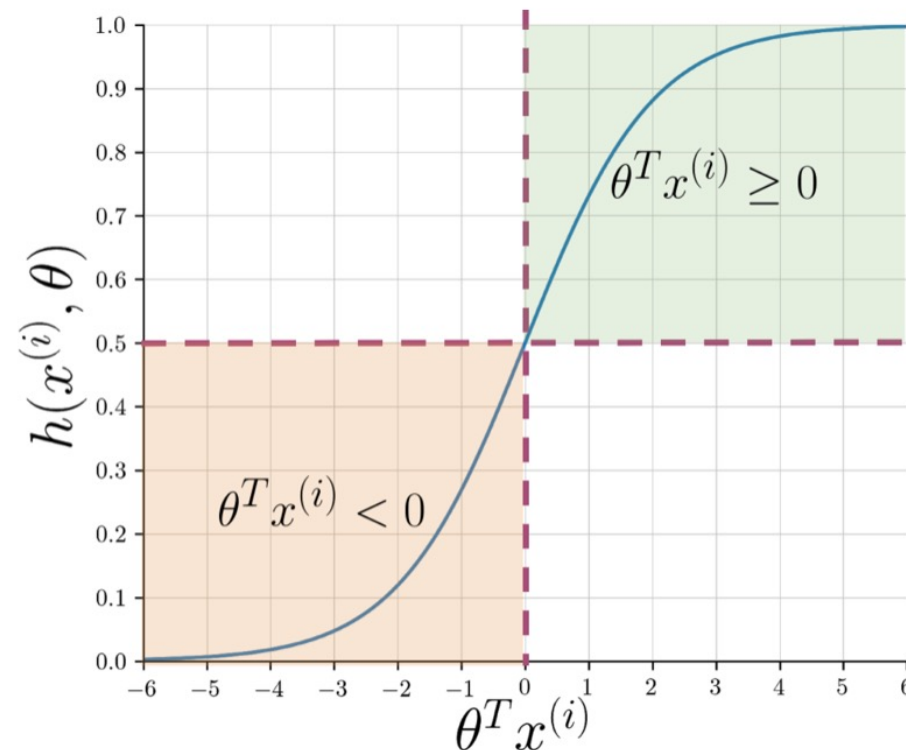
```
1: while stopping criteria not met do
2:   Sample a training example  $\mathbf{x}_i, \mathbf{y}_i$ 
3:   Compute the loss  $L(f(\mathbf{x}_i; \Theta), \mathbf{y}_i)$ 
4:    $\hat{\mathbf{g}} \leftarrow$  gradients of  $L(f(\mathbf{x}_i; \Theta), \mathbf{y}_i)$  w.r.t  $\theta$ 
5:    $\Theta \leftarrow \Theta - \eta_t \hat{\mathbf{g}}$ 
6: return  $\Theta$ 
```

---

# Regresión Logística

- La regresión logística hace uso de la función **sigmoide** que genera una probabilidad entre 0 y 1. La función sigmoide con algún parámetro de peso  $\theta$  y alguna entrada  $x(i)$  se define como sigue.

$$h(x^{(i)}, \theta) = \frac{1}{1 + e^{-\theta^T x^{(i)}}}$$



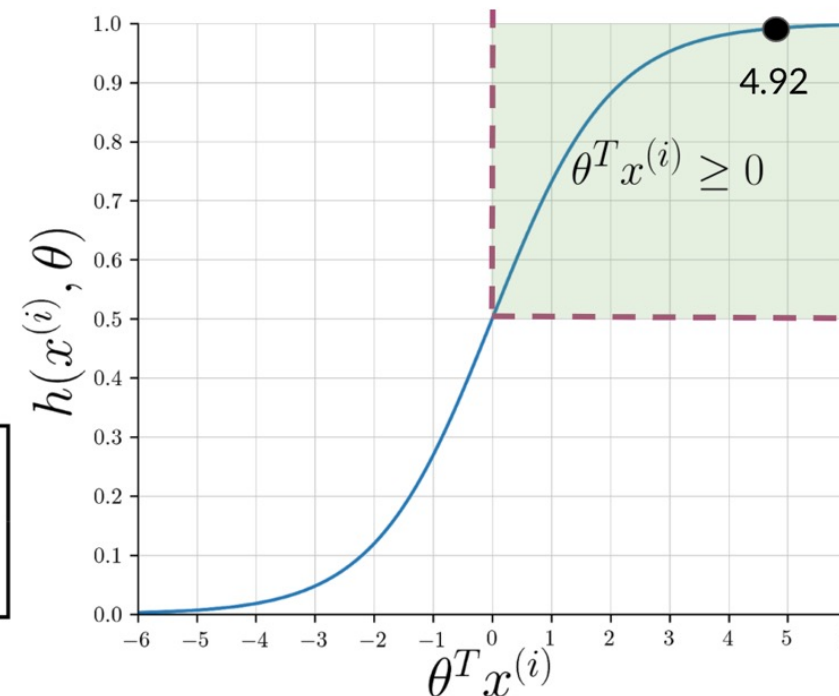
# Regresión Logística

- Tener en cuenta que a medida que  $\theta^T x^{(i)}$  se acerca cada vez más a  $-\infty$ , el denominador de la función sigmoide se hace cada vez más grande y, como resultado, el sigmoide se acerca a 0. Por otro lado, a medida que  $\theta^T x^{(i)}$  se acerca más y más para  $\infty$  el denominador de la función sigmoide se acerca a 1 y, como resultado, la sigmoide también se acerca a 1. Ahora, dado un tweet, puede transformarlo en un vector y ejecutarlo a través de su función sigmoide para obtener una predicción de la siguiente manera:

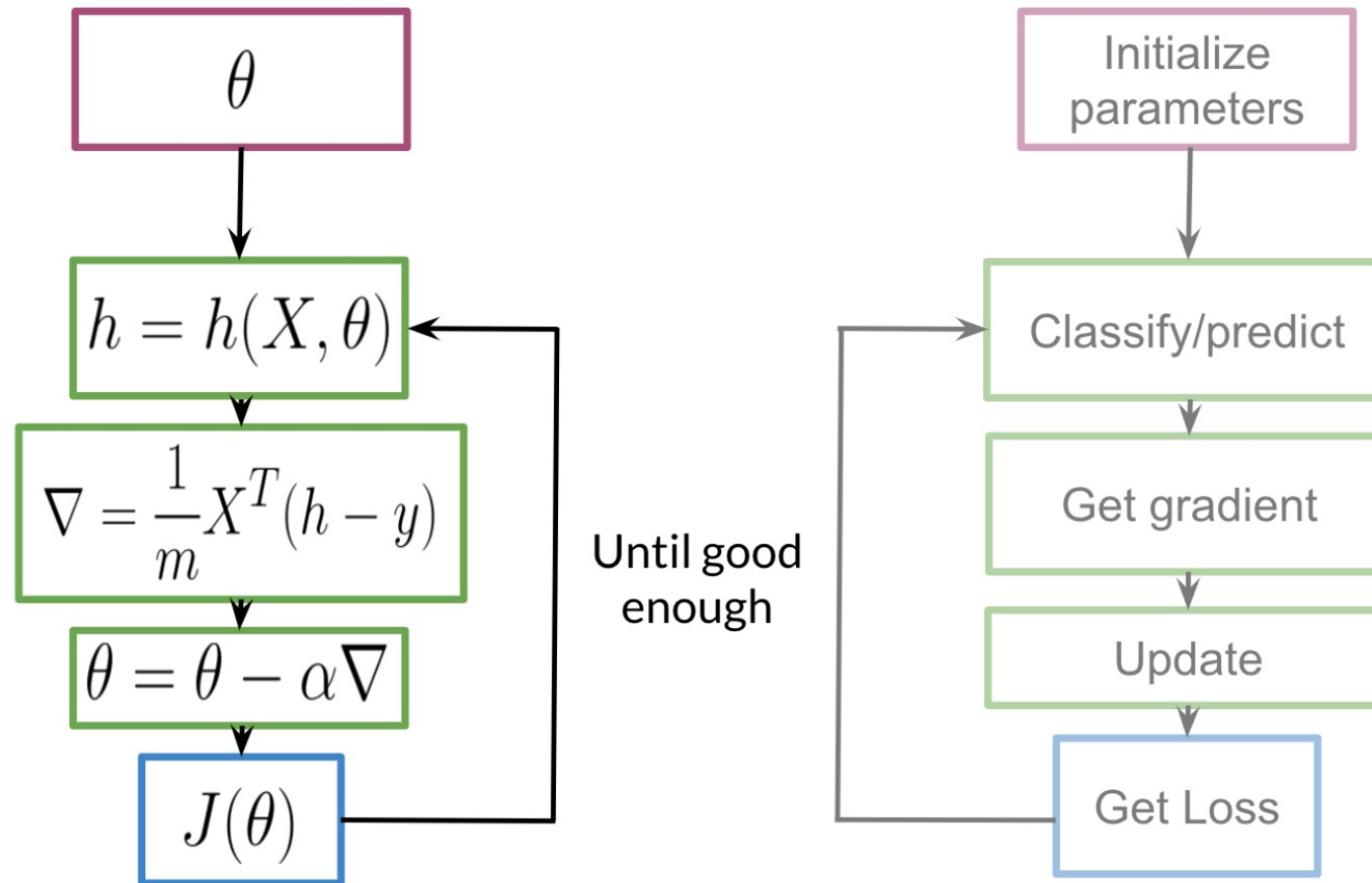
*"@Usuario2 y @Usiario1 están realizando un GRAN curso en <https://iimas.unam.mx> !!!"*

*[realiza, gran, curso].*

$$x^{(i)} = \begin{bmatrix} 1 \\ 3476 \\ 245 \end{bmatrix} \quad \theta = \begin{bmatrix} 0.00003 \\ 0.00150 \\ -0.00120 \end{bmatrix}$$



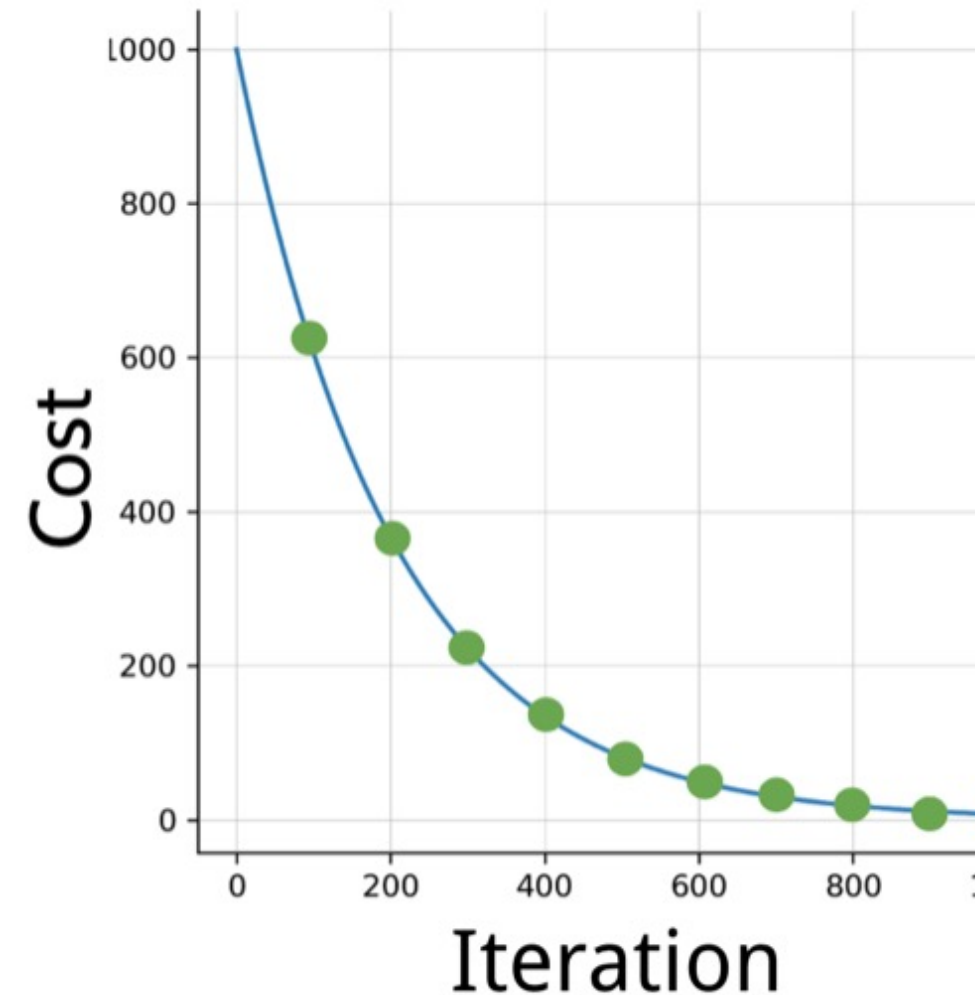
# Entrenamiento Regresión Logística





# Entrenamiento Regresión Logística

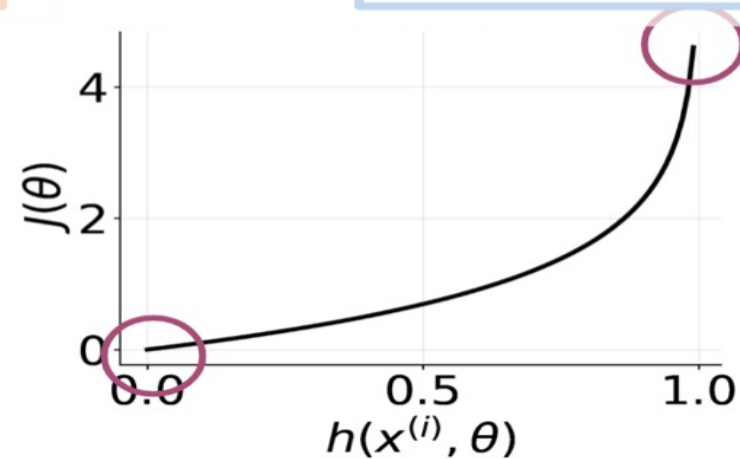
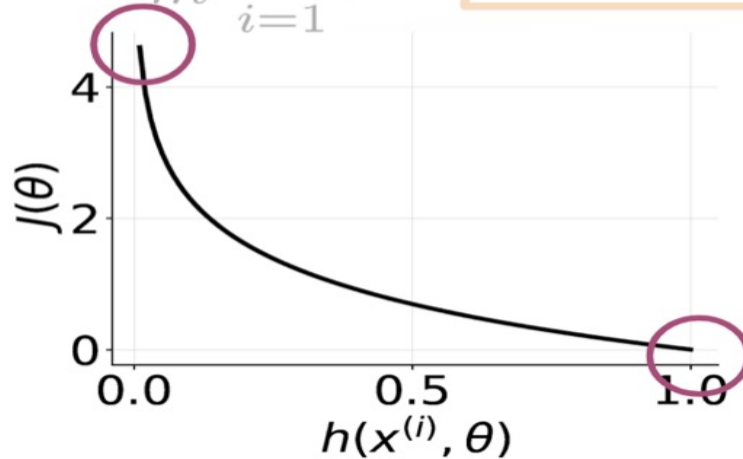
- Inicializar el parámetro  $\theta$ , que se usará en la sigmoide, luego calcular el gradiente que se usará para actualizar  $\theta$ , y luego calcular el costo. Sigue haciéndolo hasta que sea lo suficientemente bueno.
- **Nota:** En pocas palabras, el gradiente te permite aprender el valor de  $\theta$  para que puedas predecir la etiqueta del documento.
- Por lo general, se sigue entrenando hasta que el costo converja. Si dibujamos el número de iteraciones versus al costo, debería ver algo como esto.



# Regresión Logística: costo

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log h(x^{(i)}, \theta) + (1 - y^{(i)}) \log(1 - h(x^{(i)}, \theta))]$$

$$J(\theta) = -\frac{1}{m} \sum_{i=1}^m [y^{(i)} \log h(x^{(i)}, \theta) + (1 - y^{(i)}) \log(1 - h(x^{(i)}, \theta))]$$



Si  $y = 1$  y predice algo cercano a 0, la función obtiene un costo cercano a  $\infty$ . Lo mismo se aplica para entonces  $y = 0$  y predice algo cercano a 1. Por otro lado, si obtiene una predicción igual a la etiqueta, obtiene un costo de 0. En cualquier caso, está tratando de minimizar  $J(\theta)$ .



# Regresión Logística: gradiente

- La forma general del descenso del gradiente se calcula así:

---

```
Repeat {
     $\theta_j := \theta_j - \alpha \frac{\partial}{\partial \theta_j} J(\theta)$ 
}
```

- Para cada j, podemos trabajar la derivada usando cálculo:

---

```
Repeat {
     $\theta_j := \theta_j - \frac{\alpha}{m} \sum_{i=1}^m (h(x^{(i)}, \theta) - y^{(i)}) x_j^{(i)}$ 
}
```

- Una implementación vectorizada  $\theta := \theta - \frac{\alpha}{m} X^T (H(X, \theta) - Y)$

# Regresión Logística: evaluación

- Para probar el modelo, ejecutar un subconjunto de los datos, conocido como el conjunto de **validación**, en el modelo para obtener predicciones. Las predicciones son los resultados de la función sigmoide. Si la salida es  $\geq 0.5$ , la asignaría a una clase positiva. De lo contrario, lo asignaría a una clase negativa.

- $X_{val} \ Y_{val} \ \theta$

$$h(X_{val}, \theta)$$

$$pred = h(X_{val}, \theta) \geq 0.5$$

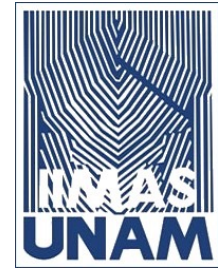
$$\begin{bmatrix} 0.3 \\ 0.8 \\ 0.5 \\ \vdots \\ h_m \end{bmatrix} \geq 0.5 = \begin{bmatrix} \underline{0.3 \geq 0.5} \\ \underline{0.8 \geq 0.5} \\ \underline{0.5 > 0.5} \\ \vdots \\ pred_m \geq 0.5 \end{bmatrix} = \begin{bmatrix} \underline{0} \\ \underline{1} \\ \underline{1} \\ \vdots \\ pred_m \end{bmatrix}$$

# Regresión Logística: evaluación

- En la práctica, dado el conjunto de datos  $X$ , normalmente se dividen en tres componentes.  $X_{train}$ ,  $X_{val}$ ,  $X_{test}$ . La distribución suele variar según el tamaño del conjunto de datos. Sin embargo, una división de 80,10,10 suele funcionar bien. Para calcular la exactitud, usamos la ecuación:

$$\text{Accuracy} \longrightarrow \sum_{i=1}^m \frac{(pred^{(i)} == y_{val}^{(i)})}{m}$$

- En otras palabras, se repasan todos los ejemplos de entrenamiento,  $m$  de ellos, y luego para cada predicción, si fue correcto, agregas uno. Luego divide por  $m$ .



# Tarea

- Revisar los laboratorios en el siguiente orden:
  1. `practica1-1preprocesamiento.ipynb`
  2. `practica1-2frecuenciaPalabras.ipynb`
  3. `practica1-3regresionLogistica.ipynb`
- Completar la implementación de la regresión logística en:
  - `practica1-tareaRegresionLogistica.ipynb`