Tema 3: Máquina de Soporte Vectorial

2018-2019

Gonzalo A. Aranda-Corral

Ciencias de la Computación e Inteligencia Artificial
Universidad de Huelva

Contenido

Índice

Historia

- Pre 1980:
  - Casi todos los métodos de Aprendizaje de Máquina se basaban en superficies de decisión lineal.
  - Las propiedades teóricas del aprendizaje lineal funcionaban bien.
- 1980's
  - Los árboles de decisión y métodos NN permiten un aprendizaje eficiente de superficies de decisión no lineales.
  - Pocas bases teóricas, además de que los métodos usados se veían afectados por lo mínimos locales.

http://www.svms.org/history.html

Historia

- Pre 1980:
  - Casi todos los métodos de Aprendizaje de Máquina se basaban en superficies de decisión lineal.
  - Las propiedades teóricas del aprendizaje lineal funcionaban bien.
- 1980's
  - Los árboles de decisión y métodos NN permiten un aprendizaje eficiente de superficies de decisión no lineales.
  - Pocas bases teóricas, además de que los métodos usados se veían afectados por lo mínimos locales.

http://www.svms.org/history.html

Historia

- Pre 1980:
  - Casi todos los métodos de Aprendizaje de Máquina se basaban en superficies de decisión lineal.
  - Las propiedades teóricas del aprendizaje lineal funcionaban bien.
- 1980's
  - Los árboles de decisión y métodos NN permiten un aprendizaje eficiente de superficies de decisión no lineales.
  - Pocas bases teóricas, además de que los métodos usados se veían afectados por lo mínimos locales.

http://www.svms.org/history.html

Historia

- Pre 1980:
  - Casi todos los métodos de Aprendizaje de Máquina se basaban en superficies de decisión lineal.
  - Las propiedades teóricas del aprendizaje lineal funcionaban bien.
- 1980's
  - Los árboles de decisión y métodos NN permiten un aprendizaje eficiente de superficies de decisión no lineales.
  - Pocas bases teóricas, además de que los métodos usados se veían afectados por lo mínimos locales.

`http://www.svms.org/history.html`

Historia

- Pre 1980:
  - Casi todos los métodos de Aprendizaje de Máquina se basaban en superficies de decisión lineal.
  - Las propiedades teóricas del aprendizaje lineal funcionaban bien.
- 1980's
  - Los árboles de decisión y métodos NN permiten un aprendizaje eficiente de superficies de decisión no lineales.
  - Pocas bases teóricas, además de que los métodos usados se veían afectados por lo mínimos locales.

http://www.svms.org/history.html

Historia

- 1990's
  Se desarrollan algoritmos de aprendizaje eficientes para funciones no lineales en la teoría de aprendizaje computacional.

- 1992:
  - Se introduce el concepto de Maquinas de Soporte Vectorial. Boser, Guyon & Vapnik. Su uso se ha vuelto popular desde entonces.
  - Algoritmo teóricamente bien fundamentado: Desarrollado de la Teoría de Aprendizaje Estadístico (Vapnik & Chervonenkis) desde la década de los 60's.
  - Buen desempeño empírico: éxito en diversas aplicaciones (bioinformatica, texto, reconocimiento de imágenes, etc.)

`http://www.svms.org/history.html`

Historia

- 1990's
  Se desarrollan algoritmos de aprendizaje eficientes para funciones no lineales en la teoría de aprendizaje computacional.

- 1992:
  - Se introduce el concepto de Maquinas de Soporte Vectorial. Boser, Guyon & Vapnik. Su uso se ha vuelto popular desde entonces.
  - Algoritmo teóricamente bien fundamentado: Desarrollado de la Teoría de Aprendizaje Estadístico (Vapnik & Chervonenkis) desde la década de los 60's.
  - Buen desempeño empírico: éxito en diversas aplicaciones (bioinformatica, texto, reconocimiento de imágenes, etc.)

`http://www.svms.org/history.html`

Historia

- 1990's
  Se desarrollan algoritmos de aprendizaje eficientes para funciones no lineales en la teoría de aprendizaje computacional.

- 1992:
  - Se introduce el concepto de Maquinas de Soporte Vectorial. Boser, Guyon & Vapnik. Su uso se ha vuelto popular desde entonces.
  - Algoritmo teóricamente bien fundamentado: Desarrollado de la Teoría de Aprendizaje Estadístico (Vapnik & Chervonenkis) desde la década de los 60's.
  - Buen desempeño empírico: éxito en diversas aplicaciones (bioinformatica, texto, reconocimiento de imágenes, etc.)

http://www.svms.org/history.html

Historia

- 1990's
  Se desarrollan algoritmos de aprendizaje eficientes para funciones no lineales en la teoría de aprendizaje computacional.

- 1992:
  - Se introduce el concepto de Maquinas de Soporte Vectorial. Boser, Guyon & Vapnik. Su uso se ha vuelto popular desde entonces.
  - Algoritmo teóricamente bien fundamentado: Desarrollado de la Teoría de Aprendizaje Estadístico (Vapnik & Chervonenkis) desde la década de los 60's.
  - Buen desempeño empírico: éxito en diversas aplicaciones (bioinformatica, texto, reconocimiento de imágenes, etc.)

`http://www.svms.org/history.html`

Índice

Conceptos básicos

- Una máquina de SVM se puede utilizar para tareas de clasificación y regresión
  - aunque nosotros nos vamos a centrar en la clasificación
- Es una tarea de aprendizaje supervisado
  - Los datasets incluyen la clase a la que pertenece
  - El aprendizaje consiste en encontrar la división entre las clases

Conceptos básicos

- Una máquina de SVM se puede utilizar para tareas de clasificación y regresión
  - aunque nosotros nos vamos a centrar en la clasificación
- Es una tarea de aprendizaje supervisado
  - Los datasets incluyen la clase a la que pertenece
  - El aprendizaje consiste en encontrar la división entre las clases

Conceptos básicos

- Una máquina de SVM se puede utilizar para tareas de clasificación y regresión
  - aunque nosotros nos vamos a centrar en la clasificación
- Es una tarea de aprendizaje supervisado
  - Los datasets incluyen la clase a la que pertenece
  - El aprendizaje consiste en encontrar la división entre las clases

Conceptos básicos

- Una máquina de SVM se puede utilizar para tareas de clasificación y regresión
  - aunque nosotros nos vamos a centrar en la clasificación
- Es una tarea de aprendizaje supervisado
  - Los datasets incluyen la clase a la que pertenece
  - El aprendizaje consiste en encontrar la división entre las clases

Conceptos básicos

- Una máquina de SVM se puede utilizar para tareas de clasificación y regresión
  - aunque nosotros nos vamos a centrar en la clasificación
- Es una tarea de aprendizaje supervisado
  - Los datasets incluyen la clase a la que pertenece
  - El aprendizaje consiste en encontrar la división entre las clases

Conceptos básicos

- La tarea de clasificación se limita a clasificación **binaria**: **Pertenece a una clase o no**
  - Por ejemplo, un SVM puede aprender a reconocer la actividad fraudulenta de una tarjeta de crédito examinando cientos o miles de informes de actividades para determinar si existe un fraude o no.
- Se puede extender a **multiclase**
  - puede aprender a reconocer dígitos manuscritas examinando una gran colección de imágenes digitales de manuscritos.
  - Para n clases se construyen n clasificadores binarios que deciden si pertenece a esa clase o no

Conceptos básicos

- La tarea de clasificación se limita a clasificación **binaria**: **Pertenece a una clase o no**
  - Por ejemplo, un SVM puede aprender a reconocer la actividad fraudulenta de una tarjeta de crédito examinando cientos o miles de informes de actividades para determinar si existe un fraude o no.
- Se puede extender a **multiclase**
  - puede aprender a reconocer dígitos manuscritas examinando una gran colección de imágenes digitales de manuscritos.
  - Para n clases se construyen n clasificadores binarios que deciden si pertenece a esa clase o no

Conceptos básicos

- La tarea de clasificación se limita a clasificación **binaria**: **Pertenece a una clase o no**
  - Por ejemplo, un SVM puede aprender a reconocer la actividad fraudulenta de una tarjeta de crédito examinando cientos o miles de informes de actividades para determinar si existe un fraude o no.
- Se puede extender a **multiclase**
  - puede aprender a reconocer dígitos manuscritas examinando una gran colección de imágenes digitales de manuscritos.
  - Para n clases se construyen n clasificadores binarios que deciden si pertenece a esa clase o no

Conceptos básicos

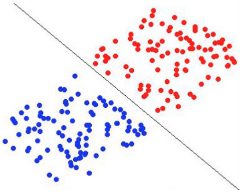
- La tarea de clasificación se limita a clasificación **binaria**: **Pertenece a una clase o no**
  - Por ejemplo, un SVM puede aprender a reconocer la actividad fraudulenta de una tarjeta de crédito examinando cientos o miles de informes de actividades para determinar si existe un fraude o no.
- Se puede extender a **multiclase**
  - puede aprender a reconocer dígitos manuscritas examinando una gran colección de imágenes digitales de manuscritos.
  - Para n clases se construyen n clasificadores binarios que deciden si pertenece a esa clase o no

Conceptos básicos

- La tarea de clasificación se limita a clasificación **binaria**: **Pertenece a una clase o no**
    - Por ejemplo, un SVM puede aprender a reconocer la actividad fraudulenta de una tarjeta de crédito examinando cientos o miles de informes de actividades para determinar si existe un fraude o no.
- Se puede extender a **multiclase**
    - puede aprender a reconocer dígitos manuscritas examinando una gran colección de imágenes digitales de manuscritos.
    - Para n clases se construyen n clasificadores binarios que deciden si pertenece a esa clase o no

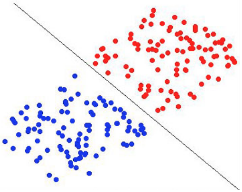
Conceptos básicos

- Una SVM **aprende la superficie** decisión de dos clases distintas de los puntos de entrada.
- Como un clasificador binario, forma una “**frontera de decisión**” dejando a un lado los elementos de una clase y al otro los que no pertenecen a la misma.



Conceptos básicos

- Una SVM **aprende la superficie** decisión de dos clases distintas de los puntos de entrada.
- Como un clasificador binario, forma una “**frontera de decisión**” dejando a un lado los elementos de una clase y al otro los que no pertenecen a la misma.



Conceptos básicos

- De forma general, **Clasificar** significa aprender la siguiente función de mapeo:

$$clasificacion : X \to Y$$

  donde X es un objeto, e Y es la etiqueta de clase que el clasificador le asignaría.

- El caso más simple: **clasificación binaria**, sea $x \in \mathbb{R}^n$ e $y \in \{-1, +1\}$

Conceptos básicos

- De forma general, **Clasificar** significa aprender la siguiente función de mapeo:

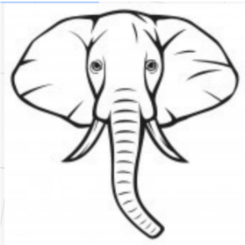
$$\mathit{clasificacion} : X \to Y$$

  donde X es un objeto, e Y es la etiqueta de clase que el clasificador le asignaría.

- El caso más simple: **clasificación binaria**, sea $x \in \mathbb{R}^n$ e $y \in \{-1, +1\}$

Ejemplo simple

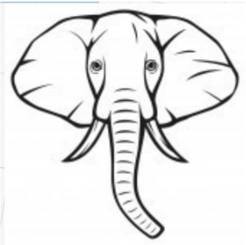
- Supongamos que tenemos imagenes de 50 tigres y de 50 elefantes.
  m=100



- Y las digitalizamos en imagenes de 100 x 100 pixeles, entonces tenemos , $x \in \mathbb{R}^n$,
  n= 10 000

Ejemplo simple

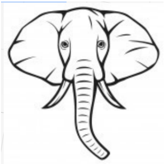
- Supongamos que tenemos imagenes de 50 tigres y de 50 elefantes.
  m=100

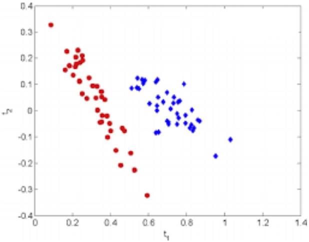


- Y las digitalizamos en imagenes de 100 x 100 pixeles, entonces tenemos , $x \in \mathbb{R}^n$,
  n= 10 000

Ejemplo simple

- Ahora, si introdujeramos una fotografía **NUEVA**, **¿es un tigre o un elefante?**[1]





[1] Asumiendo por supuesto, que es uno u otro.

¿**Cómo** lo hacemos?

- "Construimos" un **hiperplano** que separe las muestras positivas (+1) de las negativas (-1).
- Los puntos x^i del hiperplano satisfacen la ecuación: $wx + b = 0$ $(\theta^T \cdot x = 0)$
- Los puntos que estén por encima, o por debajo del hiperplano corresponderán con cada una de las etiquetas.
  - $wx + b > 0 \rightarrow y = +1$
  - $wx + b < 0 \rightarrow y = -1$
- es decir, cada **nueva imagen** que llegue (x), la **sustituimos** en la ecuación $wx + b$ y según el **resultado** (positivo o negativo) asignamos una **etiqueta**.

Conceptos básicos

¿**Cómo** lo hacemos?

- "Construimos" un **hiperplano** que separe las muestras positivas (+1) de las negativas (-1).
- Los puntos x^i del hiperplano satisfacen la ecuación: $wx + b = 0$ $(\theta^T \cdot x = 0)$
- Los puntos que estén por encima, o por debajo del hiperplano corresponderán con cada una de las etiquetas.
  - $wx + b > 0 \quad \rightarrow \quad y = +1$
  - $wx + b < 0 \quad \rightarrow \quad y = -1$
- es decir, cada **nueva imagen** que llegue (x), la **sustituimos** en la ecuación $wx + b$ y según el **resultado** (positivo o negativo) asignamos una **etiqueta**.

Conceptos básicos

¿**Cómo** lo hacemos?

- "Construimos" un **hiperplano** que separe las muestras positivas (+1) de las negativas (-1).
- Los puntos x^i del hiperplano satisfacen la ecuación: $wx + b = 0$ $(\theta^T \cdot x = 0)$
- Los puntos que estén por encima, o por debajo del hiperplano corresponderán con cada una de las etiquetas.
  - $wx + b > 0 \rightarrow y = +1$
  - $wx + b < 0 \rightarrow y = -1$
- es decir, cada **nueva imagen** que llegue (x), la **sustituimos** en la ecuación $wx + b$ y según el **resultado** (positivo o negativo) asignamos una **etiqueta**.

Conceptos básicos

¿**Cómo** lo hacemos?

- "Construimos" un **hiperplano** que separe las muestras positivas (+1) de las negativas (-1).
- Los puntos x^i del hiperplano satisfacen la ecuación: $wx + b = 0$ $(\theta^T \cdot x = 0)$
- Los puntos que estén por encima, o por debajo del hiperplano corresponderán con cada una de las etiquetas.
  - $wx + b > 0 \;\rightarrow\; y = +1$
  - $wx + b < 0 \;\rightarrow\; y = -1$
- es decir, cada **nueva imagen** que llegue (x), la **sustituimos** en la ecuación $wx + b$ y según el **resultado** (positivo o negativo) asignamos una **etiqueta**.

Conceptos básicos

¿**Cómo** lo hacemos?

- “Construimos” un **hiperplano** que separe las muestras positivas (+1) de las negativas (-1).
- Los puntos x^i del hiperplano satisfacen la ecuación: $wx + b = 0$ $(\theta^T \cdot x = 0)$
- Los puntos que estén por encima, o por debajo del hiperplano corresponderán con cada una de las etiquetas.
  - $wx + b > 0 \ \rightarrow \ y = +1$
  - $wx + b < 0 \ \rightarrow \ y = -1$
- es decir, cada **nueva imagen** que llegue (x), la **sustituimos** en la ecuación $wx + b$ y según el **resultado** (positivo o negativo) asignamos una **etiqueta**.

Conceptos básicos

¿**Cómo** lo hacemos?

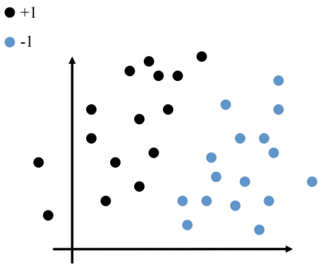
- "Construimos" un **hiperplano** que separe las muestras positivas (+1) de las negativas (-1).
- Los puntos x^i del hiperplano satisfacen la ecuación: $wx + b = 0$ $(\theta^T \cdot x = 0)$
- Los puntos que estén por encima, o por debajo del hiperplano corresponderán con cada una de las etiquetas.
  - $wx + b > 0 \;\rightarrow\; y = +1$
  - $wx + b < 0 \;\rightarrow\; y = -1$

- es decir, cada **nueva imagen** que llegue (x), la **sustituimos** en la ecuación $wx + b$ y según el **resultado** (positivo o negativo) asignamos una **etiqueta**.

Índice

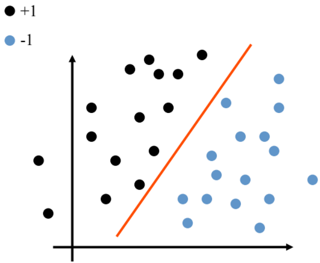
Idea inicial de separación

- Dado un conjunto de datos… que suponemos inicialmente "separable",



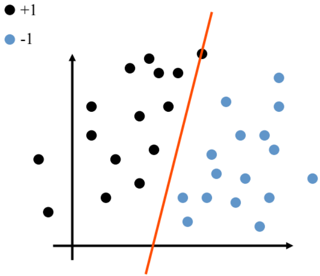
Idea inicial de separación

- Es decir, podemos establecer un hiperplano de separación que deje a todos los miembros de la misma clase en los mismos lados.



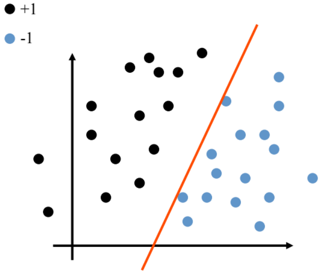
Idea inicial de separación

- Pero ese hiperplano, así definido, no es único.



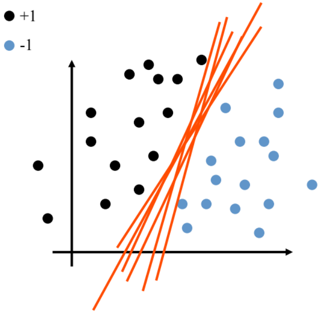
Idea inicial de separación

- Pero ese hiperplano, así definido, no es único.



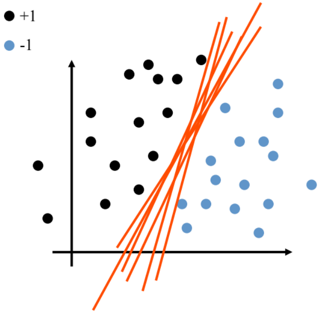
Idea inicial de separación

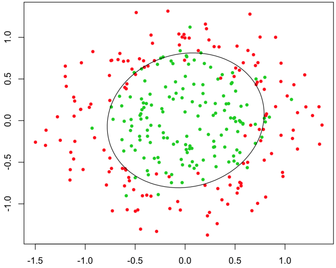
- De hecho, podemos encontrar infinitos hiperplanos que cumplan esa condición.
  - o ninguno



Idea inicial de separación

- De hecho, podemos encontrar infinitos hiperplanos que cumplan esa condición.
  - o ninguno





Idea inicial de separación

- La idea, entonces, es añadir más carácterísticas (condiciones) a ese hiperplano para que sea único.
- Eso va a depender de la naturaleza de los datos.
- Podemos encontrar 3 tipos de conjuntos de datos:
  - Separables linealmente
  - Quasi separables
  - No separables (linealmente)

Idea inicial de separación

- La idea, entonces, es añadir más carácterísticas (condiciones) a ese hiperplano para que sea único.
- Eso va a depender de la naturaleza de los datos.

- Podemos encontrar 3 tipos de conjuntos de datos:
  - Separables linealmente
  - Quasi separables
  - No separables (linealmente)

Idea inicial de separación

- La idea, entonces, es añadir más carácterísticas (condiciones) a ese hiperplano para que sea único.
- Eso va a depender de la naturaleza de los datos.

- Podemos encontrar 3 tipos de conjuntos de datos:
  - Separables linealmente
  - Quasi separables
  - No separables (linealmente)

Idea inicial de separación

- La idea, entonces, es añadir más carácterísticas (condiciones) a ese hiperplano para que sea único.
- Eso va a depender de la naturaleza de los datos.
- Podemos encontrar 3 tipos de conjuntos de datos:
    - Separables linealmente
    - Quasi separables
    - No separables (linealmente)

Idea inicial de separación

- La idea, entonces, es añadir más carácterísticas (condiciones) a ese hiperplano para que sea único.
- Eso va a depender de la naturaleza de los datos.
- Podemos encontrar 3 tipos de conjuntos de datos:
  - Separables linealmente
  - Quasi separables
  - No separables (linealmente)

Idea inicial de separación

- La idea, entonces, es añadir más carácterísticas (condiciones) a ese hiperplano para que sea único.
- Eso va a depender de la naturaleza de los datos.
- Podemos encontrar 3 tipos de conjuntos de datos:
  - Separables linealmente
  - Quasi separables
  - No separables (linealmente)

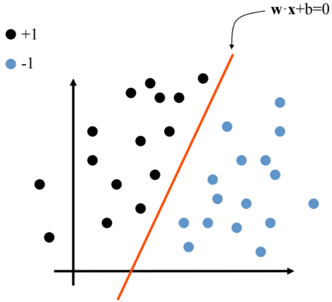
Índice

Linealmente Separables

- Establecemos que la ecuación del hiperplano es: w·x+b = 0

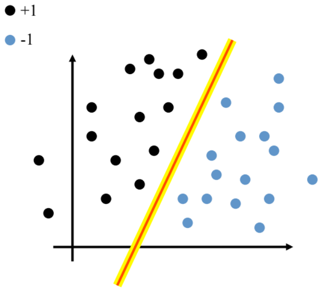
$$h_\theta(x) \equiv \theta^T \cdot X = 0$$

Es importante recordar que: **aprender significaría encontrar los w y la b** (θ)



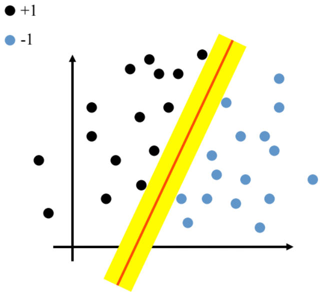
Linealmente Separables

- Definimos una distancia de **margen**,
que es la distancia del hiperplano al punto más cercano de la clase



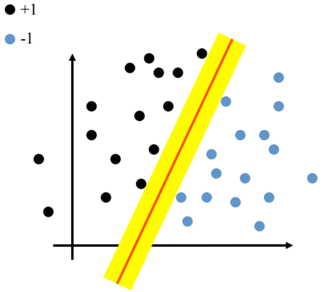
Linealmente Separables

- Y nos proponemos encontrar una **ecuación** del hiperplano que haga que **el margen sea lo más grande posible**



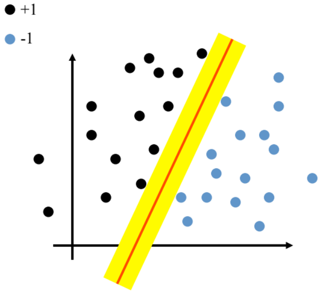
Linealmente Separables

- Como idea intuitiva, sería la recta o el plano que pasa justo por **medio** de las clases
- Esto trata de hacer al clasificador lo más **tolerante** posible



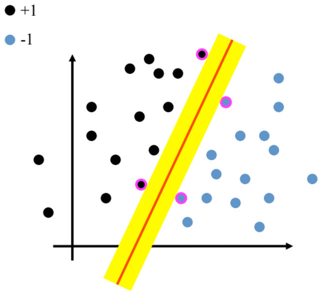
Linealmente Separables

- Como idea intuitiva, sería la recta o el plano que pasa justo por **medio** de las clases
- Esto trata de hacer al clasificador lo más **tolerante** posible



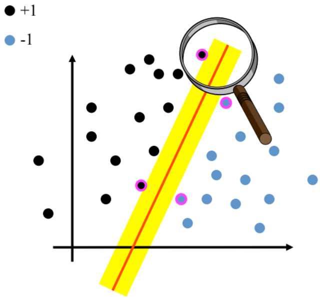
Linealmente Separables

- El, o los, puntos de cada clase **más cercanos** al hiperplano se denominan "**vectores soporte**"



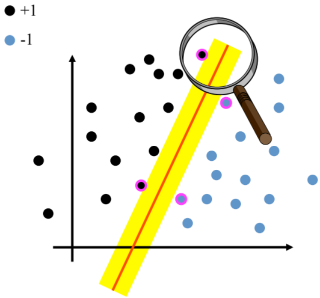
Linealmente Separables

- La **distancia** de todos los **vectores** soporte al **hiperplano** es la **misma**
- El hiperplano, por tanto, estará **a mitad** de camino entre uno y otro.



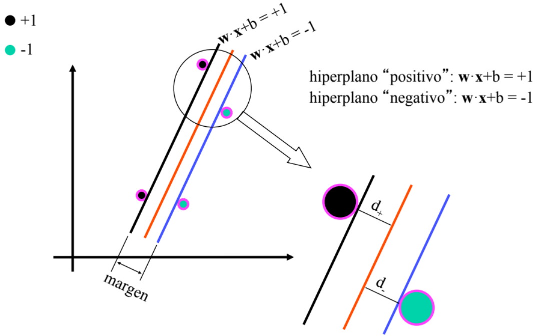
Linealmente Separables

- La **distancia** de todos los **vectores** soporte al **hiperplano** es la **misma**
- El hiperplano, por tanto, estará **a mitad** de camino entre uno y otro.



Linealmente Separables

- Hacemos dos **hiperplanos paralelos** al que buscamos y que pasen por los vectores soporte
- Las **ecuaciones** de estos las vamos a expresar así:



LINEALMENTE SEPARABLES

- Se quiere construir un **clasificador** que, en función de las coordenadas del punto nos diga a que clase pertenece
  - $w \cdot x + b > 0$ para $y_i = +1$. (hiperplano “positivo”)
  - $w \cdot x + b < 0$ para $y_i = -1$. (hiperplano “negativo”)

- Se quiere construir un **clasificador** que, en función de las coordenadas del punto nos diga a que clase pertenece
  - $w \cdot x + b > 0$ para $y_i = +1$. (hiperplano “positivo”)
  - $w \cdot x + b < 0$ para $y_i = -1$. (hiperplano “negativo”)

- Se quiere construir un **clasificador** que, en función de las coordenadas del punto nos diga a que clase pertenece
  - $w \cdot x + b > 0$ para $y_i = +1$. (hiperplano "positivo")
  - $w \cdot x + b < 0$ para $y_i = -1$. (hiperplano "negativo")

LINEALMENTE SEPARABLES

Notación:

- $\vec{w}$ es es un **vector normal** al hiperplano
- $\frac{|b|}{||w||}$ es la distancia perpendicular del hiperplano al origen.
- $||w||$ es la norma euclídea de w
- $< w, x_i >$ Es el producto escalar de w y x_i

Notación:

- $\vec{w}$ es es un **vector normal** al hiperplano
- $\dfrac{|b|}{||w||}$ es la distancia perpendicular del hiperplano al origen.
- $||w||$ es la norma euclídea de w
- $< w, x_i >$ Es el producto escalar de w y x_i

LINEALMENTE SEPARABLES

Notación:

- $\vec{w}$ es es un **vector normal** al hiperplano
- $\dfrac{|b|}{||w||}$ es la distancia perpendicular del hiperplano al origen.
- $||w||$ es la norma euclídea de w
- $< w, x_i >$ Es el producto escalar de w y x_i

Notación:

- $\vec{w}$ es es un **vector normal** al hiperplano
- $\frac{|b|}{||w||}$ es la distancia perpendicular del hiperplano al origen.
- $||w||$ es la norma euclídea de w
- $< w, x_i >$ Es el producto escalar de w y x_i

Conceptos matemáticos

... y el problema su puede expresar así:

- minimizar: $||w||$ sujeto a: $< w, x_i > +b = 0$

- Pero el problema se puede transformar para que quede más fácil de manejar!
- Se usan multiplicadores de Lagrange (α_i).

$$L_p \equiv \frac{1}{2}||w||^2 - \sum_{i=1}^{m} \alpha_i y_i (w \cdot x_i + b) + \sum_{i=1}^{m} \alpha_i$$

Conceptos matemáticos

... y el problema su puede expresar así:

- minimizar: $||w||$ sujeto a: $< w, x_i > +b = 0$
- Pero el problema se puede transformar para que quede más fácil de manejar!
- Se usan multiplicadores de Lagrange (α_i).

$$L_p \equiv \frac{1}{2}||w||^2 - \sum_{i=1}^{m} \alpha_i y_i (w \cdot x_i + b) + \sum_{i=1}^{m} \alpha_i$$

... y el problema su puede expresar así:

- minimizar: $||w||$ sujeto a: $< w, x_i > +b = 0$

- Pero el problema se puede transformar para que quede más fácil de manejar!
- Se usan multiplicadores de Lagrange (α_i).

$$L_p \equiv \frac{1}{2}||w||^2 - \sum_{i=1}^{m} \alpha_i y_i (w \cdot x_i + b) + \sum_{i=1}^{m} \alpha_i$$

Conceptos matemáticos

Haciendo cálculos y pasando el problema “al dual” , se convierte en:

- maximizar: $L_D = \sum_{i=1}^{m} \alpha_i + \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j x_i x_j$
- sujeto a: $w = \sum_{i=1}^{m} \alpha_i y_i x_i$ y $\sum_{i=1}^{m} \alpha_i y_i = 0$
- **Fácil** de resolver con un sistema algebraico

Conceptos matemáticos

Haciendo cálculos y pasando el problema “al dual” , se convierte en:

- maximizar: $L_D = \sum_{i=1}^{m} \alpha_i + \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j x_i x_j$
- sujeto a: $w = \sum_{i=1}^{m} \alpha_i y_i x_i$ y $\sum_{i=1}^{m} \alpha_i y_i = 0$
- **Fácil** de resolver con un sistema algebraico

Conceptos matemáticos

Haciendo cálculos y pasando el problema “al dual” , se convierte en:

- maximizar: $L_D = \sum_{i=1}^{m} \alpha_i + \sum_{i=1}^{m} \sum_{j=1}^{m} \alpha_i \alpha_j y_i y_j x_i x_j$
- sujeto a: $w = \sum_{i=1}^{m} \alpha_i y_i x_i$ y $\sum_{i=1}^{m} \alpha_i y_i = 0$
- **Fácil** de resolver con un sistema algebraico

Índice

QUASI SEPARABLES

- Los problemas **reales** se caracterizan normalmente por poseer ejemplos **ruidosos** y no ser perfecta y linealmente separables.

- La **estrategia** para este tipo de problemas reales es relajar el grado de separabilidad del conjunto de ejemplos, **permitiendo** que haya **errores** de clasificación en algunos de los ejemplos del conjunto de entrenamiento.

- Sin embargo, sigue siendo un **objetivo** el encontrar un **hiperplano óptimo** con el **menor error posible**

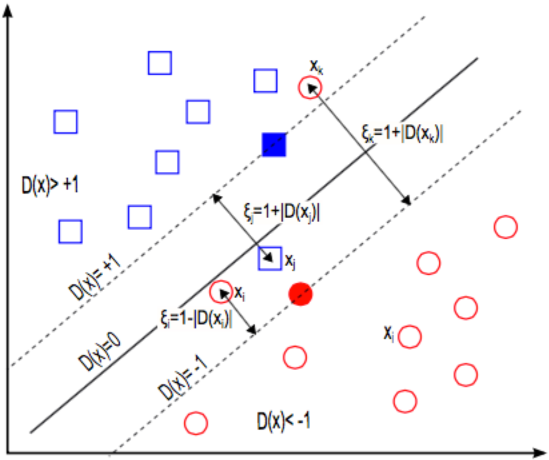
QUASI SEPARABLES

- Los problemas **reales** se caracterizan normalmente por poseer ejemplos **ruidosos** y no ser perfecta y linealmente separables.
- La **estrategia** para este tipo de problemas reales es relajar el grado de separabilidad del conjunto de ejemplos, **permitiendo** que haya **errores** de clasificación en algunos de los ejemplos del conjunto de entrenamiento.
- Sin embargo, sigue siendo un **objetivo** el encontrar un **hiperplano óptimo** con el **menor error posible**

QUASI SEPARABLES

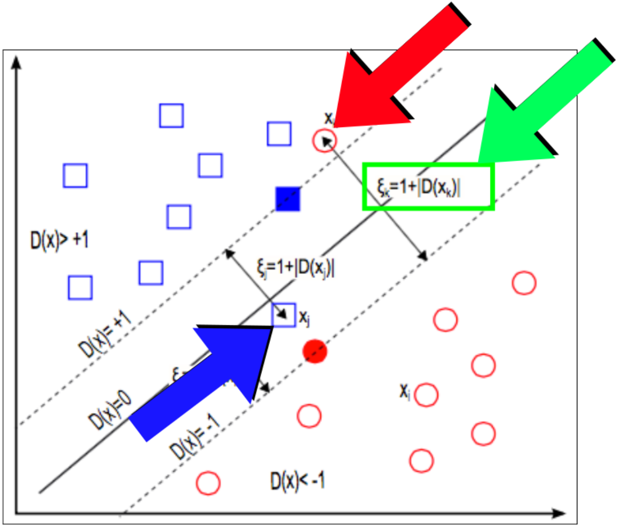
- Los problemas **reales** se caracterizan normalmente por poseer ejemplos **ruidosos** y no ser perfecta y linealmente separables.
- La **estrategia** para este tipo de problemas reales es relajar el grado de separabilidad del conjunto de ejemplos, **permitiendo** que haya **errores** de clasificación en algunos de los ejemplos del conjunto de entrenamiento.
- Sin embargo, sigue siendo un **objetivo** el encontrar un **hiperplano óptimo** con el **menor error posible**

- Gráficamente



QUASI SEPARABLES

- Gráficamente



QUASI SEPARABLES

- En la formulación anterior, un ejemplo es no-separable si no cumple la condición:

$$y_i(< w, x_i > +b) \geq 1 \qquad i = 1, \ldots, n$$

- Introducimos un factor de relajación (ξ), por lo que los hiperplanos quedan:
  - $< w, x_i > +b \geq 1 - \xi_i \qquad$ para $y = +1$
  - $< w, x_i > +b \leq 1 + \xi_i \qquad$ para $y = -1$
- volveríamos a elaborar cálculos para el mínimo de error.

QUASI SEPARABLES

- En la formulación anterior, un ejemplo es no-separable si no cumple la condición:

$$y_i(< w, x_i > +b) \geq 1 \qquad i = 1, \ldots, n$$

- Introducimos un factor de relajación (ξ), por lo que los hiperplanos quedan:
  - $< w, x_i > +b \geq 1 - \xi_i \qquad$ para $y = +1$
  - $< w, x_i > +b \leq 1 + \xi_i \qquad$ para $y = -1$

- volveríamos a elaborar cálculos para el mínimo de error.

QUASI SEPARABLES

- En la formulación anterior, un ejemplo es no-separable si no cumple la condición:

$$y_i(< w, x_i > +b) \geq 1 \qquad i = 1, \ldots, n$$

- Introducimos un factor de relajación (ξ), por lo que los hiperplanos quedan:
  - $< w, x_i > +b \geq 1 - \xi_i$ para $y = +1$
  - $< w, x_i > +b \leq 1 + \xi_i$ para $y = -1$

- volveríamos a elaborar cálculos para el mínimo de error.

QUASI SEPARABLES

- En la formulación anterior, un ejemplo es no-separable si no cumple la condición:

$$y_i(< w, x_i > +b) \geq 1 \qquad i = 1, \ldots, n$$

- Introducimos un factor de relajación (ξ), por lo que los hiperplanos quedan:
  - $< w, x_i > +b \geq 1 - \xi_i \qquad$ para $y = +1$
  - $< w, x_i > +b \leq 1 + \xi_i \qquad$ para $y = -1$

- volveríamos a elaborar cálculos para el mínimo de error.

QUASI SEPARABLES

- En la formulación anterior, un ejemplo es no-separable si no cumple la condición:

$$y_i(< w, x_i > +b) \geq 1 \qquad i = 1, \dots, n$$

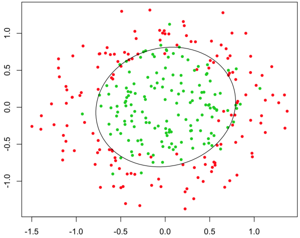
- Introducimos un factor de relajación (ξ), por lo que los hiperplanos quedan:
  - $< w, x_i > +b \geq 1 - \xi_i \qquad$ para $y = +1$
  - $< w, x_i > +b \leq 1 + \xi_i \qquad$ para $y = -1$

- volveríamos a elaborar cálculos para el mínimo de error.

Índice

No separables (linealmente)

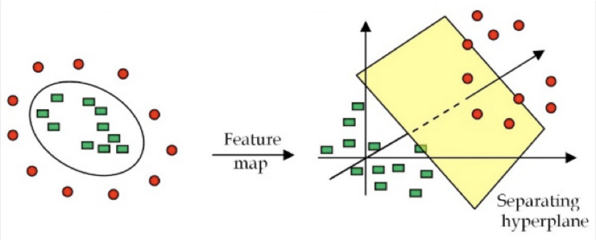
- Los datos originales no pueden ser separados por una superficie lineal (pérdida de convexidad)



No separables (linealmente)

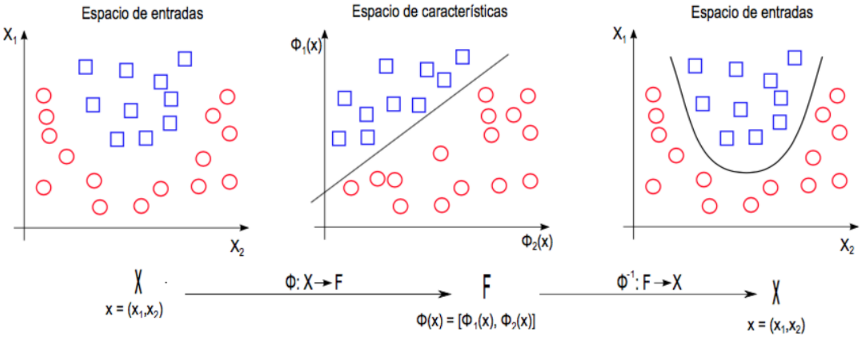
Idea:

- Mapear los datos, por medio de una **función KERNEL**, a un espacio de características en un espacio **dimensional más alto**, donde se busca la máxima separación entre clases.



No separables (linealmente)

- El funcionamiento completo sería algo así



No separables (linealmente)

- La función usada para cambiar de espacio se denomina función **Kernel**
- La idea es cambiar la norma por una función **kernel** que sea el producto escalar de una función ($K = \phi \cdot \phi$)

No separables (linealmente)

- La función usada para cambiar de espacio se denomina función **Kernel**
- La idea es cambiar la norma por una función **kernel** que sea el producto escalar de una función ($K = \phi \cdot \phi$)

No separables (linealmente)

Se suelen usar funciones de kernel **estándar**:

Lineal	$K(a, b) = a \cdot b$
Polinómicas	$K(a, b) = (\gamma a \cdot b + c)^d$
Funciones de Base Radial (RBF)	$K(a, b) = exp(-\frac{(a-b)^2}{2\sigma^2})$
Sigmoide (o NN)	$K(a, b) = tanh(\gamma a \cdot b + c)$

No separables (linealmente)

No todas las funciones pueden ser Kernel

- Necesitan ser factorizables: $K(a, b) = \phi(a)\phi(b)$
- Cumplir la condición de Mercer:

No separables (linealmente)

No todas las funciones pueden ser Kernel

- Necesitan ser factorizables: $K(a, b) = \phi(a)\phi(b)$
- Cumplir la condición de Mercer:

Teorema (Condición de Mercer) Existe una transformación $\mathbf{\Phi}$ y una expansión en series $K(\mathbf{x}_i,\mathbf{x}_i) = \sum \mathbf{\Phi}(\mathbf{x}_i)\cdot\mathbf{\Phi}(\mathbf{x}_j)$ si y solo si para cualquier $g(\mathbf{x})$ para la que la integral $\int g(\mathbf{x})^2 \, d\mathbf{x}$ sea finita, se tiene $\int K(\mathbf{xy})g(\mathbf{x})g(\mathbf{y})d\mathbf{x}d\mathbf{y} \geq 0$

No separables (linealmente)

- Con la utilización de esta propiedad, y reescribiendo el problema, se reduce a minimizar:

$$L_p = \frac{1}{2}||w||^2 + C(\sum_{i=1}^{n} \xi_i)$$

- Sujeto a las restricciones:

$$z_i(w \cdot \Phi(x_i) + b) - 1 + \xi_i \geq 0 \qquad \xi_i \geq 0$$

- el vector w es combinación de los vectores soporte transformados

$$\bar{w} = \sum_{i=1}^{n^*} \alpha_i z_i \bar{\Phi}(\bar{s}_i)$$

No separables (linealmente)

- Con la utilización de esta propiedad, y reescribiendo el problema, se reduce a minimizar:

$$L_p = \frac{1}{2}||w||^2 + C(\sum_{i=1}^{n} \xi_i)$$

- Sujeto a las restricciones:

$$z_i(w \cdot \Phi(x_i) + b) - 1 + \xi_i \geq 0 \qquad \xi_i \geq 0$$

- el vector w es combinación de los vectores soporte transformados

$$\bar{w} = \sum_{i=1}^{n^*} \alpha_i z_i \bar{\Phi}(\bar{s}_i)$$

No separables (linealmente)

- Con la utilización de esta propiedad, y reescribiendo el problema, se reduce a minimizar:

$$L_p = \frac{1}{2}||w||^2 + C(\sum_{i=1}^{n} \xi_i)$$

- Sujeto a las restricciones:

$$z_i(w \cdot \Phi(x_i) + b) - 1 + \xi_i \geq 0 \qquad \xi_i \geq 0$$

- el vector w es combinación de los vectores soporte transformados

$$\bar{w} = \sum_{i=1}^{n^*} \alpha_i z_i \bar{\Phi}(\bar{s}_i)$$

Índice

Implementación

- El cálculo de parámetros del hiperplano es un problema de optimización cuadrática.
- Existen varios **algoritmos especializados** para resolver rápidamente el problema QP, sobre todo basados en **heurísticas** para romper el problema en, trozos más pequeños y manejables.

Implementación

- El cálculo de parámetros del hiperplano es un problema de optimización cuadrática.
- Existen varios **algoritmos especializados** para resolver rápidamente el problema QP, sobre todo basados en **heurísticas** para romper el problema en, trozos más pequeños y manejables.

- Un método común es el algoritmo de optimización secuencial mínima (SMO) de Platt[2]
  - Descompone el problema en sub-problemas 2-dimensionales que se pueden resolver analíticamente, eliminando la necesidad de un algoritmo de optimización numérica.

[2] Artículo original

- Un método común es el algoritmo de optimización secuencial mínima (SMO) de Platt[2]
  - Descompone el problema en sub-problemas 2-dimensionales que se pueden resolver analíticamente, eliminando la necesidad de un algoritmo de optimización numérica.

[2] Artículo original

Bibliografía

Ha ido apareciendo a lo largo del tema, pero además:

- A Training Algorithm for Optimal Margin Classifiers
  Bernhard E. Boser, Isabelle Guyon y Vladimir Vapnik
  (en Moodle)
- Tutorial sobre Máquinas de Vectores Soporte (SVM)
  Enrique J. Carmona Suárez
  (en Moodle)
- ... internet ...