

Clasificación

PROFESOR: PATRICIO DE LA CUADRA

AYUDANTE: GABRIEL DURÁN

Objetivos de la clasificación

- Asignar clases a un conjunto de elementos según algún criterio específico.
 - Ejemplos:
 - Clasificar estilos musicales.
 - Identificación de una persona mediante la voz
- Generalmente se extrae un set de características, o descriptores, a cada elemento y éstos son clasificados utilizando distancias en dicho espacio multidimensional. Otros algoritmos, como las redes neuronales artificiales, aprenden relaciones entre dichos descriptores, en función de éstas asignan afinidades a cada clase y se asigna la clase con mayor afinidad.
- Un algoritmo de clasificación compara dichos descriptores y asigna el grupo al que es clasificado cada elemento.

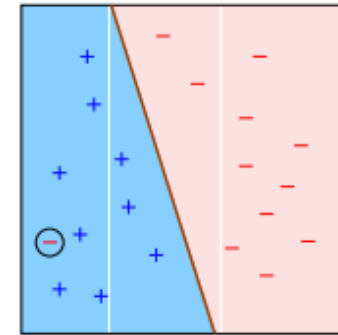
Tipos de clasificadores

- Clasificación supervisada: Se tiene un set de de elementos previamente etiquetados, llamado set de entrenamiento y luego un nuevo elemento es clasificado según los datos previamente aprendidos.
- Clasificación No supervisada: No se tienen datos previamente etiquetados y muchas veces incluso se desconoce el número de clases. Por lo tanto un algoritmo agrupa los elementos según su similitud, lo que se conoce como *clustering*.
- Algunos clasificadores son:
 - KNN
 - LDA
 - Redes Neuronales
 - SVM
 - K-Means (*clustering*)
 - Random forest
 - Naive Bayes

Clasificadores

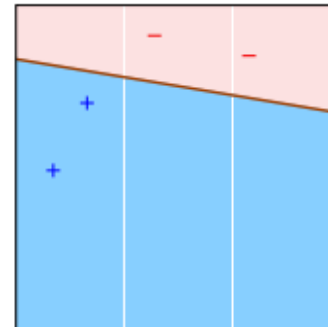
- Los clasificadores lo hacen en base a los datos disponibles. Pocos datos, o datos de mala calidad (mal etiquetados, poco representativos, etc) implicarán un clasificador deficiente.
- El objetivo es lograr un buen desempeño de clasificación sobre los datos utilizados para desarrollar el sistema, sin perder generalidad. Cuando el modelo se ajusta demasiado a los datos de entrenamiento, se dice que el modelo está sobre entrenado (overfitting).

Good:

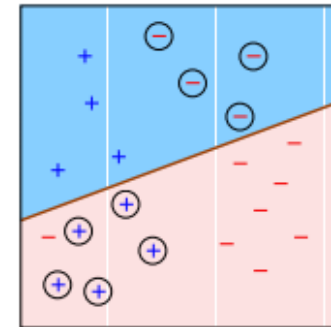


sufficient data
low training error
simple classifier

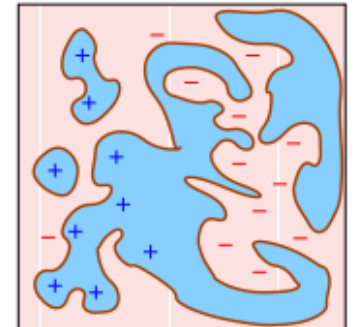
Bad:



insufficient data



training error
too high



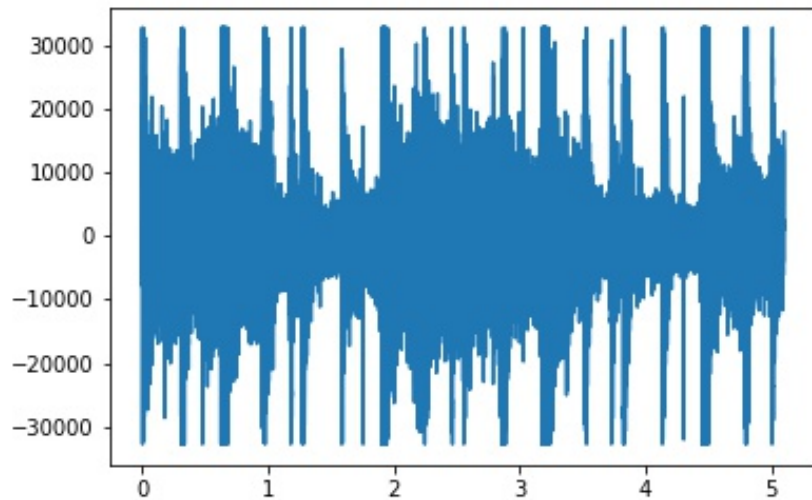
classifier
too complex

Extracción de descriptores

- En audio la información suele ser contenida en un gran número de elementos, hay mucha información redundante o inútil para efectos de clasificación. Por lo tanto es necesario hacer una reducción de cantidad de elementos basada en descriptores. Típicamente se utilizan los descriptores temporales y/o frecuenciales vistos en capítulos anteriores.
- Según el objetivo de clasificación se debe elegir un determinado conjunto de descriptores. Por ejemplo si se desea clasificar instrumentos musicales, probablemente no sean muy útiles descriptores de intensidad, pero sí de timbre.
- Existen métodos para hacer una selección del 'mejor' subconjunto de características (SBS, SFS, Búsqueda exhaustiva). Otros métodos realizan una transformación de las características, creando nuevas características a partir de una combinación lineal del set original. De esta manera se ordenan según cuánta información aportan a la clasificación (PCA, PLSR, etc).

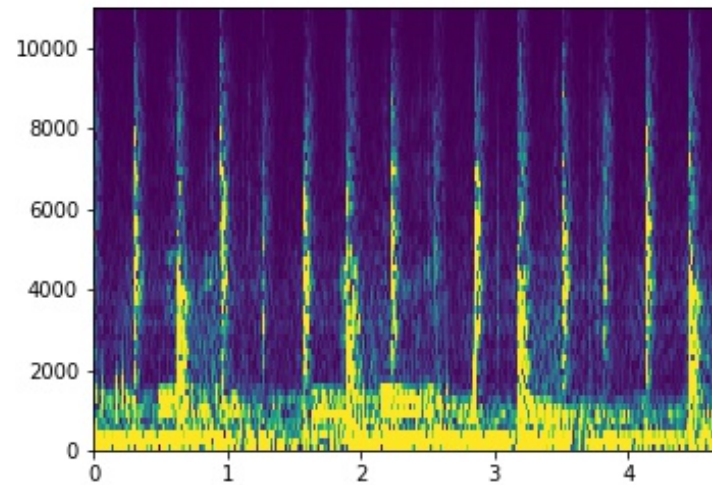
¿Por qué utilizar descriptores?

- Para un audio de 5,1 segundos con $sr=22050$:
- Waveform:



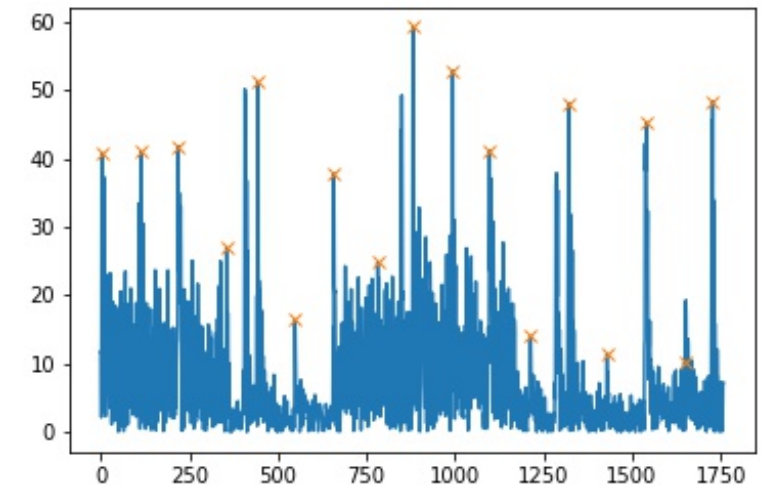
112469 x 1

STFT:



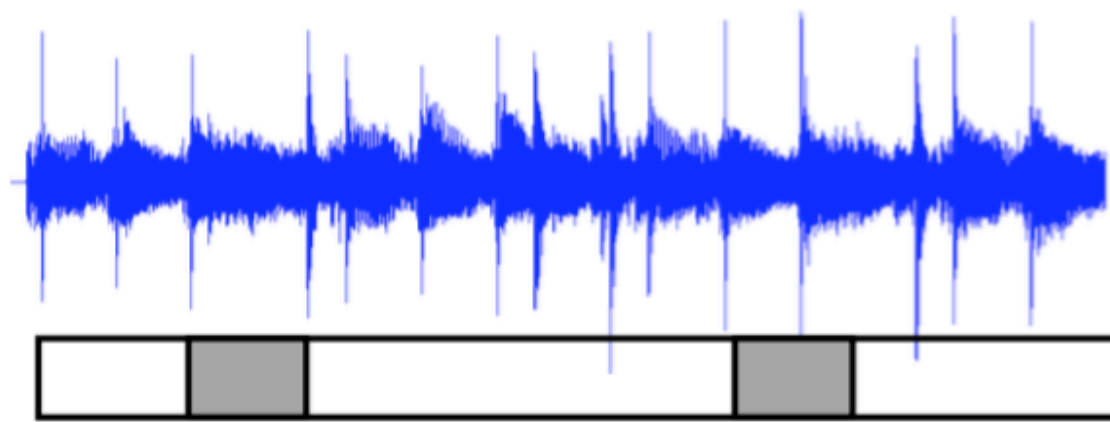
1759 x 65

Spectral Flux



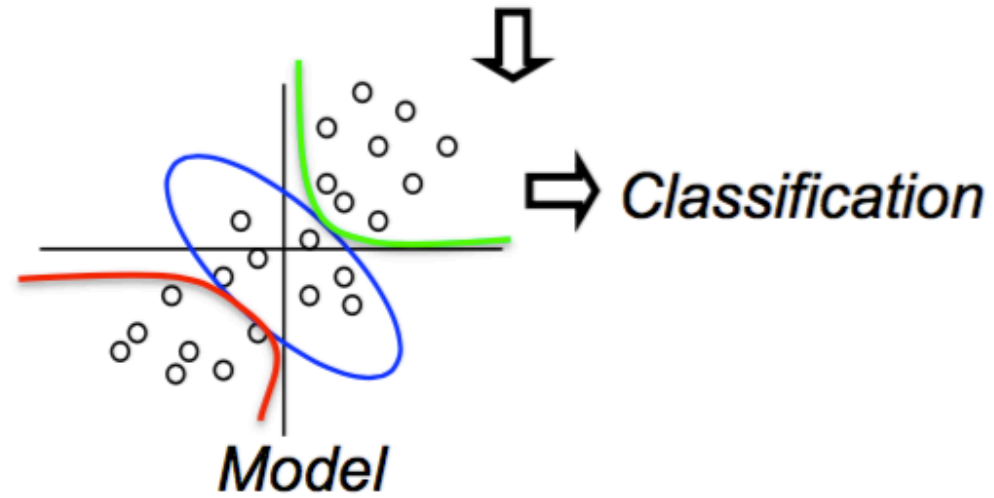
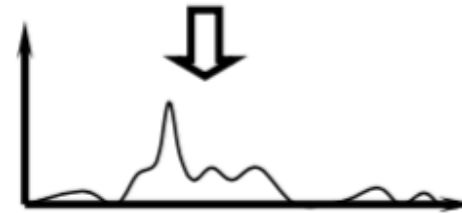
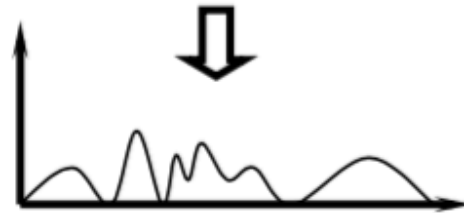
1759 x 1

- SE REDUCE LA CANTIDAD DE INFORMACIÓN EN UNA RELACIÓN 1/64 Y EL DESCRIPTOR ES MÁS REPRESENTA MEJOR AL AUDIO FRENTE A UN CLASIFICADOR.



Feature vector 1

Feature vector 2



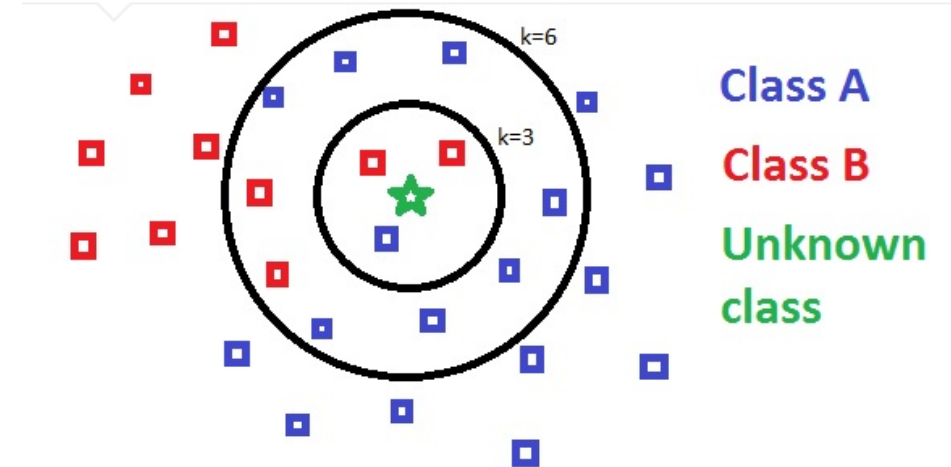
⇒ *Classification*

Clasificadores: Supervisados

- Se debe contar con un set de datos previamente etiquetados. Esta fase es crítica y es vital contar con que las etiquetas están bien hechas.
- Existe una fase en entrenamiento en la cual el algoritmo 'aprende' de los datos que se le muestran. Si hay datos mal etiquetados el aprendizaje se puede ver muy perjudicado.
- Cuando llega un nuevo elemento, es analizado según lo aprendido por el algoritmo y luego clasificado.
- Para poder entrenar un modelo y luego validarlo, antes del entrenamiento se realiza una partición aleatoria de la base de datos. Una parte será utilizada exclusivamente para entrenar el modelo, que luego será testeado sobre la otra porción de la base de datos. Generalmente se utiliza un 70% de los datos para entrenar y 30% para testear, pero es un parámetro a decidir.

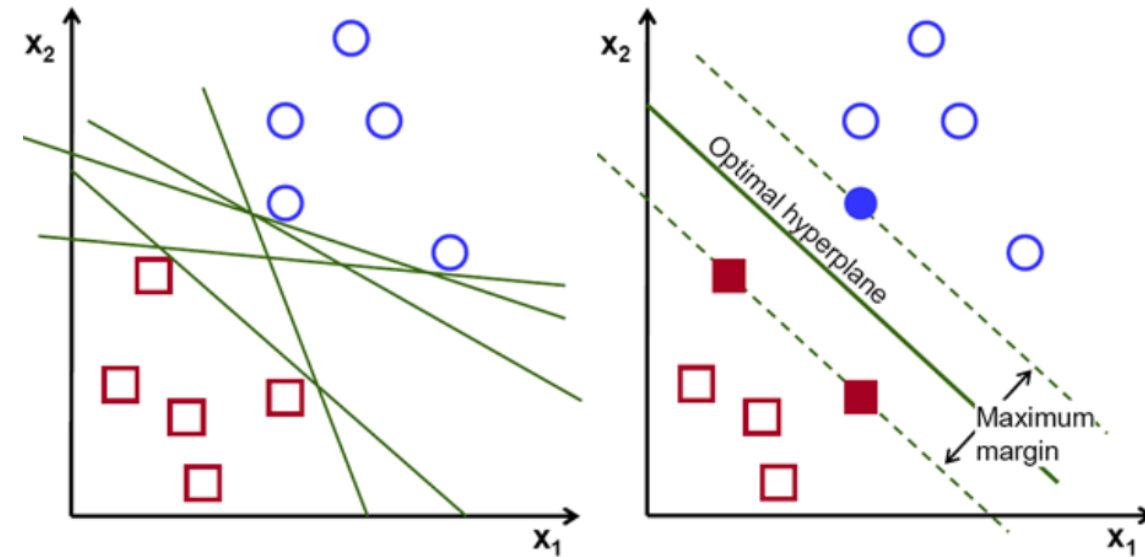
Clasificadores: KNN (k-nearest neighbors)

- En el espacio de los descriptores se busca un número 'k' de los puntos más cercanos, conocidos como vecinos. Se hace una votación entre las clases de éstos y se asigna la clase con mayor cantidad de vecinos cercanos al nuevo elemento.
- 'k' es un parámetro que debe ser elegido cuidadosamente.
- PROS:
 - Fácil de entender e intuitivo.
 - Rápido, no requiere tiempo de entrenamiento.
- CONS:
 - Para clasificar se necesitan TODOS los datos. Si la Base de datos es grande, es problemático.
 - Fuertemente dependiente del 'k' elegido.
 - Fuertemente dependiente del pre procesamiento de los datos



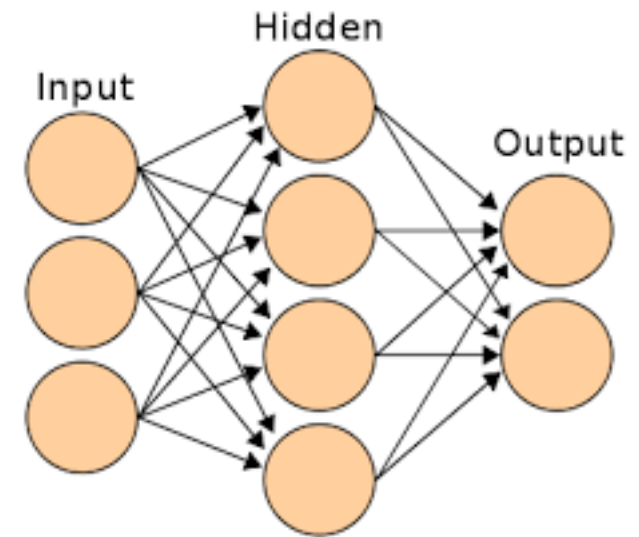
Clasificadores: SVM (Support vector machines)

- Algoritmo de optimización que busca el hiperplano que maximiza el margen de separación entre clases.
- Se pueden utilizar distintos 'kernels' como función cuyos parámetros son variables. Típicos kernels son lineales, gaussianos (RBF) o polinomiales.
- PROS:
 - Solución óptima
 - Según el kernel, se pueden 'aprender' formas complejas.
- CONS:
 - El entrenamiento requiere una gran capacidad computacional.



Clasificadores: ANN(Artificial Neural Networks)

- En la capa de entrada los descriptores son llamadas neuronas de entrada.
- Se crean combinaciones no lineales entre los descriptores en etapas conocidas como capas ocultas, que tienen un número pre definido de neuronas.
- Los ponderadores que multiplican el valor de cada neurona se inicializan aleatoriamente y luego son aprendidos mediante un algoritmo de optimización llamado 'backpropagation'.
- PROS:
 - Poco susceptible a pre procesar datos, es robusto.
 - Puede encontrar fronteras de decisión complejas de describir matemáticamente.
- CONS:
 - El entrenamiento requiere una gran capacidad computacional.
 - Suele requerir una gran cantidad de datos para aprender correctamente.
 - Puede encontrar mínimos locales y no ser una solución óptima.



Clasificadores: No Supervisados

- Si no se cuenta con datos previamente etiquetados, la única forma de clasificar es agrupar los datos de forma no supervisada.
- Para esto se clasifican juntos los elementos que son similares bajo algún criterio. Generalmente el número de clusters debe ser definido previamente, pero algunas variantes de algoritmos permiten definir un número óptimo.
- Es difícil establecer una medida de desempeño objetiva para analizar el resultado, puesto que no se tiene información previa de los datos. Una opción es analizar el resultado gráficamente (o auditivamente si se clasifican sonidos), mientras que una forma cuantitativa es obtener la correlación entre clases (que debe ser pequeña) y la correlación intra clases (que debe ser alta).

Clasificadores: K-Means

- Algoritmo iterativo en el que se calculan el centroide de los centroides de cada clase (para comenzar los centroides se eligen aleatoriamente) y cada elemento es clasificado según el centroide de la clase más cercana. Luego se vuelve a calcular los centroides según los datos clasificados en el paso anterior.
- Se espera que el algoritmo vaya convergiendo y en las últimas iteraciones los elementos ya no cambien de clase.
- PROS:
 - Es sencillo e intuitivo.
 - Existen versiones optimizadas que funcionan rápido.
- CONS:
 - Asume que los clusters son 'esféricos', no maneja bien formas geométricas complejas.

