**Universidad Rey Juan Carlos**

**Máster Oficial en Visión Artificial**

**“Biometría Informática”**

**Curso académico 2012/2013**

**Orión García Gallardo**

**Práctica 1: Comparativa de técnicas de clasificación supervisada.**

**Introducción**

El objetivo de los clasificadores supervisados es construir un modelo de distribución conciso de etiquetas con respecto a unas determinadas características. El clasificador resultante se usa para asignar etiquetas a las instancias de test donde los valores de las características es conocido, pero el valor de la etiqueta se desconoce [1].

En la actualidad hay numerosas técnicas de clasificación supervisada. Entre los algoritmos existentes cabe destacar:

* Clasificadores basados en distancias.
* Clasificadores bayesianos.
* Redes neuronales.
* Support Vector Machines (SVM).
* Boosting (AdaBoost).

Para la realización de esta práctica se han escogido dos clasificadores para compararlos entre ellos. Dichos clasificadores han sido un Normal Bayes Classifier (Clasificador bayesiano normal) y un Support Vector Machine (máquina de vectores soportados).

Por un lado, un clasificador bayesiano normal (NB) es un clasificador simple que se basa en el teorema de Bayes [2]. En este clasificador se asume que los vectores de característica de cada clase están normalmente distribuidos (no necesariamente independientemente distribuidos). De este modo la distribución completa de los datos se asume que es una mezcla Gausiana con un componente por clase. Usando unos datos de entrenamiento el algoritmo estima el vector media y las matrices covarianza de cada clase y después las usa para hacer la predicción [3].

# Por otro lado, una máquina de vectores soportados (SVM) es una técnica de reconocimiento de patrones principalmente empleada para resolver problemas de clasificación y regresión [4]. En los SVMs se mapean los vectores de características en un espacio de dimensión mayor usando algunas funciones de kernel. A continuación se construye una función lineal óptima discriminante (o un hiperplano que encaje con los datos de prueba). Finalmente, la solución es óptima cuando los márgenes entre el hiperplano de separación y el vector de características más cercano de dos clases distintas es máximo.

# Descripción de la solución

# Entre el amplio abanico de opciones disponibles para implementa esta práctica se ha optado por el lenguaje de programación C++ y la librería de manejo de imágenes OpenCV. Esta librería contiene una gran cantidad de funcionalidad principalmente orientada a la visión por computación en tiempo real. Está desarrollada por Intel pero es libre bajo la licencia open source BSD. OpenCV permite hacer uso de forma fácil y sencilla de numerosos algoritmos ya implementados, entre ellos, los clasificadores que vamos a usar en esta práctica. De esta manera uno puede centrarse en seleccionar los parámetros más convenientes para los algoritmos y en organizar los entrenamientos para obtener los resultados más óptimos.

# De entre los datos proporcionados se han seleccionado sólo los etiquetados con los valores 0, 2 y 8 (0: pronasal, 2: entocanthion derecho, 8: alar derecho). Una vez filtrados los datos se ha procedido a crear los clasificadores. Aunque actualmente sólo hay dos algoritmos de clasificación en el sistema, el diseño elegido permite añadir de forma rápida y sencilla otros clasificadores si se desea. El paso previo al entrenamiento del clasificador requiere de la configuración de los parámetros del algoritmo. En el caso del clasificador NB ningún parámetro necesitaba ser configurado con anterioridad. Sin embargo, en el caso del clasificador SVM si se han buscado los parámetros óptimos con respecto a los datos de entrenamiento con los que se disponía. Para ello se ha hecho uso de la funcionalidad aportada por la librería OpenCV y en concreto de la función train\_auto de la clase SVM. Una vez establecidos estos parámetros se ha procedido al entrenamiento del sistema. Para ello previamente se han dividido los datos en 10 conjuntos con el objetivo de usar 9 conjuntos para el entrenamiento y el restante para testear los resultados. Dicho proceso se ha repetido 10 veces donde en cada una de las iteraciones era uno el conjunto de test. Dicha metodología de evaluación de algoritmos se denomina Cross-validation (validación cruzada) y es una forma muy eficiente de evaluar dado un conjunto finitos de datos de entrenamiento [5]. Una vez finalizadas las iteraciones del entrenamiento los resultados obtenidos se han mostrado por pantalla por cada uno de los algoritmos.

# Datos aplicados

# Los datos utilizados para el desarrollo, entrenamiento y evaluación de los clasificadores vienen exclusivamente a partir del fichero “prac1\_fichPuntosFaciales” aportado en la práctica.

# Resultados

# En cuanto a los resultados obtenidos se ha apreciado que ambos algoritmos llegan a clasificar correctamente el 100% de los casos. De este modo, los 60 puntos en 3D, de los tres tipos seleccionados previamente, se han clasificado correctamente en sus respectivas clases.

# Conclusiones

# Los resultados obtenidos no son significativos a la hora de comparar algoritmos entre sí. Esto se debe a que se dispone de una muestra, muy pequeña, donde las clases están claramente disjuntas. Sin embargo, no hay que pasar por alto que ambos algoritmos serían muy eficientes a la hora de clasificar puntos tridimensionales de determinadas características de las caras humanas.

**Referencias**

[1] SB Kotsiantis, "Supervised Machine Learning: A Review of Classification Techniques," Informatica, vol. 31, pp. 249-268, 2007.

[2] Bayes, Thomas, and Price, Richard (1763). ["An Essay towards solving a Problem in the Doctrine of Chance. By the late Rev. Mr. Bayes, communicated by Mr. Price, in a letter to John Canton, M. A. and F. R. S."](http://www.stat.ucla.edu/history/essay.pdf)

[3] K. Fukunaga. “Introduction to Statistical Pattern Recognition.” second ed., New York: Academic Press, 1990.

[4] C. Burges. “A tutorial on support vector machines for pattern recognition”, Knowledge Discovery and Data Mining 2(2), 1998.

[5] Devijver, Pierre A.; and Kittler, Josef; “Pattern Recognition: A Statistical Approach.”, Prentice-Hall, London, GB, 1982.