CAPITULO 4

Clasificación y reconocimiento de patrones. Estado del arte en "remote sensing"

4.1. Definiciones y conceptos

Como ya se ha mencionado, con el incremento de la capacidad, la velocidad y las ventajas económicas de los dispositivos de procesamiento de señales, se ha visto un creciente esfuerzo por desarrollar sistemas sofisticados de tiempo real que emulen las habilidades humanas, entre ellas la visión, abarcando la identificación y clasificación de objetos (o seres) en grupos o categorías de acuerdo a sus similitudes o semejanzas.

El Reconocimiento de Patrones (Pattern recognition = PR) es el área de investigación que estudia la operación y el diseño de sistemas que reconocen patrones en los datos **[Fuk90]**. Las técnicas de PR se usan para clasificar automáticamente objetos y patrones y tomar decisiones **[Sch92][Tou74].**

El reconocimiento de patrones estadístico asume que la imagen puede contener uno o más objetos y que cada objeto pertenece a uno de varios tipos, categorías o clases de patrones.

Dada una imagen digital que contiene varios objetos, el proceso de reconocimiento de patrones consta de 3 etapas **[Bax94][Cas96][Gon92][Jai89].**

La primer etapa es llamada segmentación de la imagen, en donde los objetos de interés son aislados del resto de la imagen.

La segunda etapa, es la de extracción de rasgos, en donde los objetos son medidos. Una medida es un valor de alguna propiedad cuantificable del objeto. Un rasgo es una función de una o más medidas, computadas de tal forma que cuantifican algunas características importantes del objeto. Con estos rasgos se construye lo que se conoce como el vector de rasgos.

La tercera fase del reconocimiento de patrones es la clasificación. La idea básica es reconocer objetos utilizando rasgos. Existe un amplio conjunto de técnicas de clasificación. Todas las técnicas del reconocimiento de patrones asumen que N rasgos han sido detectados en imágenes, y que estos rasgos fueron normalizados de manera tal que pueden ser representados en el mismo espacio de medidas. Los rasgos para un objeto pueden ser representados en el espacio de rasgos N-dimensional.

La salida de este proceso es meramente una decisión sobre la clase a la que pertenece el objeto. Cada objeto es reconocido como perteneciente a un tipo particular y puede ser asignado a un conjunto de grupos preestablecidos (clases), que representan todos los tipos de objetos que se espera que existan en la imagen. Ocurre un error de clasificación si se realiza la asignación a una clase inapropiada, la probabilidad de que esto ocurra es un radio de error de clasificación.

Aplicaciones

Visión de computadora

Análisis/clasificación de señales de radar o sensores

Reconocimiento de rostros.

Identificación de huellas dactilares

Reconocimiento e interpretación de voz

Diagnosis médicas

Reconocimiento de caracteres, etc.

Ejemplo del uso de la representación abstracta

Consideremos un problema con dos clases: w1 y w2, donde w1 corresponde a la clase de las “manzanas” y w2 corresponde a la clase de las “granadas de mano”. En el espacio de patrones hay algunas diferencias entre los patrones, pi, resultantes de las manzanas, ya que no todas las manzanas tienen el mismo tamaño, forma, peso,

color, etc. Lo mismo pasa para la clase w2. También algunas manzanas y granadas de mano comparten atributos similares (por ejemplo: masa, volumen, peso). Así, basándonos en nuestro sistema de PR sobre una medición que consiste de patrones armados con el peso es probable que:

- Clasifiquemos (mal) algunas manzanas y granadas de mano como lo mismo.
- Clasifiquemos (mal) o distingamos entre manzanas pesadas y livianas (igualmente para w2)

Los enfoques de caja negra (implementados usando redes neuronales), requieren un buen sistema de entrenamiento y un conjunto de datos para entrenar.

Estructura de un sistema de PR

Posible algoritmo de retroceso o interacción

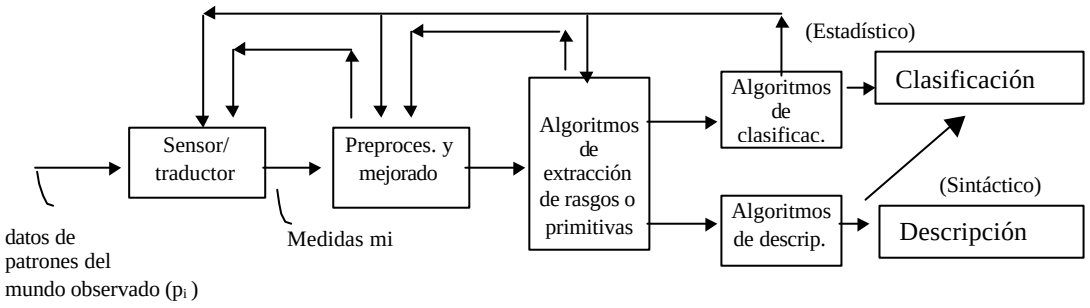


Figura 4.1 - Estructura de un sistema de PR

A los pasos antes vistos de un sistema de PR, se le agregan dos más, con lo cual tenemos las siguientes 5 etapas: 1) diseño de un localizador de objetos, 2) selección de rasgos, 3) diseño de un clasificador, 4) entrenamiento del clasificador, y 5) evaluación de su rendimiento o "performance" **[Cas96]**.

No es tema de interés aquí detallar el diseño de un localizador de objetos ni la selección de rasgos, pero sí se debe destacar la importancia de ambas etapas.

Diseño del clasificador: establece la estructura lógica del clasificador y su base matemática para la regla de decisión que se utilizará en la clasificación.

Para cada objeto encontrado el clasificador computa un valor para cada clase que indica el ***grado*** con el cual el objeto se parece a los objetos de la clase.

- El valor antes dicho se computa como *una función de los rasgos*, y se usa para seleccionar la clase más apropiada en la asignación.

- La mayoría de las reglas de decisión se reducen a una regla que utiliza un umbral (threshold) para particionar el espacio de rasgos en regiones disjuntas, una (o más) para cada clase. Cada región se corresponde con una clase simple.

Si los valores de los rasgos caen en una región particular, entonces el objeto se asigna a la correspondiente clase. Muchas veces, una o más de tales regiones pueden corresponderse con una clase llamada **desconocida**.

Entrenamiento del clasificador: una vez que las reglas de decisión se establecieron, se debe evaluar cuáles son los valores de **umbral** que separan las clases. Esto se hace, generalmente, por entrenamiento del clasificador usando un grupo de objetos conocidos.

El *conjunto de entrenamiento* es una colección de objetos de cada clase que fueron clasificados previamente por algún método preciso.

Medición del rendimiento: la precisión del clasificador puede ser directamente estimada tabulando su rendimiento sobre un conjunto de objetos de test. Para ello el conjunto debe ser suficientemente grande, y estar libre de errores.

Existen diferentes técnicas y es importante que el conjunto de test sea diferente del de entrenamiento.

4.2. Clasificación supervisada vs. no supervisada

En esta sección se presentan en forma breve algunos de los métodos de clasificación supervisada y no supervisada **[Mar94][www7]**, de forma tal de dar un contexto para la comparación entre dichos métodos y el propuesto en la presente tesis (**Capítulo 6**).

Podemos diferenciar a los clasificadores en las siguientes categorías:

1. **Clasificación Supervisada:** se trabaja con dos hipótesis:

a) las clases son de naturaleza **determinística** (se tiene un vector que representa a todos los objetos de una clase, y que se conoce como **vector prototipo**)

b) toda la información necesaria y suficiente para su diseño se encuentra disponible a priori.

Pueden usarse varios algoritmos de clasificación supervisada para asignar un pixel u objeto desconocido a una clase entre un conjunto de clases posibles. La elección particular del clasificador o regla de decisión depende de la naturaleza particular de los datos de entrada y de la salida esperada. Los algoritmos de clasificación paramétrica asumen que el vector de medidas observadas, $X_{c,}$ para cada clase y por cada rasgo, tiene una distribución "Gaussiana" por naturaleza. Los algoritmos de clasificación no paramétrica no tienen tal hipótesis **[Jen96]**.

2. **Clasificación No Supervisada:** las clases no son conocidas a priori. Se recurre a un agrupamiento natural ("**clustering"**).

No existe ningún reconocimiento externo que guíe el diseño de las funciones discriminantes. La única información que se requiere es el vector de rasgos, sin embargo, algunos algoritmos necesitan conocer el número de clases. Trabajan recibiendo vectores de rasgos y agrupándolos para formar clases.

4.3. Clasificadores utilizados en el área de sensado remoto

Clasificación supervisada

- Método lineal basado en regionalización [Gon92][Mar94]

- Un clasificador particiona el espacio de rasgos (R) en regiones de decisión etiquetadas como clases. Si se usan regiones de decisión para la asignación de una clase posible única, estas regiones deben cubrir todo R y ser disjuntas
- El borde de cada región de decisión es un límite de decisión.
- Con este punto de vista, la clasificación de un vector desconocido x, resulta algo simple: **determinamos la región de decisión (en R) en la que cae x, y asignamos x a esa clase**.

A pesar de que esta estrategia de clasificación es directa, la determinación de las regiones de decisión es un desafío. Las funciones discriminantes en el caso de n rasgos, pasan a ser hiperplanos

- Método lineal basado en distancia
(Clasificador de mínima distancia) [Gon92][Lil94]

Supongamos que se tienen M clases y que cada una está representada por un vector prototipo:

$$m_j = \frac{1}{N_j} \sum_{x \in w_j} x \quad j = 1 \ldots M \qquad \textbf{(Ec 4.1)}$$

N_j**:** es el número de vectores patrones para la clase w_j.
La sumatoria se toma sobre estos vectores.

Una forma de determinar de qué clase es miembro un vector patrón desconocido x, es **asignarlo a la clase más cercana a su prototipo**. Es decir, medimos su similitud con cada clase computando su distancia al vector prototipo de la clase.

Se puede usar la distancia Euclidea para determinar la proximidad o cualquier combinación pesada de rasgos. Si usamos la distancia "euclideana", se reduce el problema a computar las medidas de distancia

$$d_j = \left\| x - m_j \right\| \quad j = 1 \ldots M \qquad \textbf{(Ec 4.2 a)}$$

$\|a\| = \left(a^T a\right)^{1/2}$ es la norma euclideana. **(Ec 4.2 b)**

Entonces, asignamos x a la clase $w_{i,}$ si $d_i(x)$ es la menor distancia encontrada. Esto es, la menor distancia implica la mejor concordancia o "matching" en esta formulación. No es difícil demostrar que esto es equivalente a evaluar las funciones:

$$d_j = x^T m_j - \frac{1}{2} m_j^T m_j \quad j = 1 \ldots M \qquad \textbf{(Ec 4.3)}$$

y se asigna x a la clase $w_{i,}$ si $d_i(x)$ lleva al valor numérico más grande.

El límite de decisión entre la clase w_i y w_j para un clasificador de mínima distancia es:

$$d_{ij} = d_i(x) - d_j(x) = 0 \qquad \textbf{(Ec 4.4)}$$

En la clasificación de datos sensados remotamente, es necesario que el usuario provea los vectores medios para cada clase en cada banda.

La ventaja de este método es que es simple y eficiente computacionalmente. El problema es que es insensible a diferentes grados de varianza en los datos de respuesta espectral. Por lo que no es recomendable para aplicaciones en las que las clases espectrales están cercanas unas de otras en el espacio de medidas y tienen alta varianza.

- Método de clasificación del paralelepípedo [Jen96][www7]

Se puede introducir sensibilidad a la varianza de la categoría considerando un rango de valores en cada conjunto de entrenamiento. El rango puede estar dado, por ejemplo, por el valor más bajo y más alto en cada banda.

El método se puede definir entonces como sigue. Se usan para clasificar los datos de entrenamiento en n bandas espectrales. Se tiene un vector de medias n-dimensional, $M_c = (U_{ck}, U_{c2}, \ldots, U_{cn})$ donde U_{ck} es el valor medio de los datos de entrenamiento obtenidos para la clase c en la banda k, entre m clases posibles. Se llama S_{ck} a la desviación estándar de los datos de entrenamiento de la clase c en la banda k.

Usando un umbral basado en la desviación estándar y siendo Bv_{ijk} el valor del pixel de la banda k en las coordenadas espaciales (i,j), el algoritmo del paralelepípedo decide que Bv_{ijk} está en la clase c, si y sólo si:

$$(U_{ck} - S_{ck}) \leq Bv_{ijk} \leq (U_{ck} + S_{ck}) \qquad \textbf{(Ec 4.5)}$$

c = 1, 2,..., m. Número de clases

k = 1, 2, 3, ..., n. Número de bandas

Si definimos los límites (inferior y superior) de decisión como:

$$L_{ck} = U_{ck} - S_{ck} \quad y \quad H_{ck} = U_{ck} + S_{ck} \qquad \textbf{(Ec 4.6)}$$

El algoritmo del paralelepípedo se convierte en:

$$L_{ck} \leq Bv_{ijk} \leq H_{ck} \qquad \textbf{(Ec 4.7)}$$

Los límites de decisión forman un paralelepípedo n-dimensional en el espacio de rasgos. Si el valor de un pixel está por arriba del umbral inferior y por debajo del superior para todas las bandas evaluadas, se asigna el pixel a la clase. Cuando el

pixel no satisface este criterio para ninguna de las clases se asigna a la clase "desconocida".

El algoritmo del paralelepípedo es computacionalmente eficiente para clasificar datos sensados remotamente, sin embargo algunos paralelepípedos se superponen. Esto provoca que un pixel puede satisfacer el criterio para más de una clase. La superposición se debe en parte a que las distribuciones de las categorías que exhiben correlación o alta covarianza no se representan correctamente por regiones de decisión rectangulares. La covarianza es la tendencia de los valores espectrales de variar similarmente en dos bandas, visto esto en un diagrama se observan como nubes elongadas e inclinadas. Por ello, se dice que los datos no se ajustan a regiones de decisión rectangulares.

Clasificadores estadísticos óptimos

Hasta ahora se usaron *vectores prototipos fijos*, representativos de cada clase, con lo que se admite un comportamiento determinístico de los elementos de una clase.

Hay situaciones en las que los vectores de algunas clases presentan una dispersión significativa respecto a su media y por eso no conviene usar la hipótesis determinística sino un enfoque estadístico

Las consideraciones probabilísticas se vuelven importantes para el reconocimiento de patrones dada la aleatoriedad bajo la cual las clases patrones son generadas.

- Clasificador de Bayes [Dud73][Fuk90][Sch92]

El enfoque "bayesiano" ha sido usado para el reconocimiento de objetos cuando la distribución de los objetos no es tan directa. En general, hay una superposición significativa en los valores de rasgos de diferentes objetos. En el momento de decidir se encontrarían varias clases como posibles candidatas a las que asignar el objeto. Para tomar una buena decisión se puede usar el enfoque de Bayes.

En este enfoque se utiliza el conocimiento probabilístico sobre los rasgos y la frecuencia de los objetos. Supongamos que conocemos que la probabilidad de los objetos de la clase w_j es $P(w_j)$. Esto significa que a priori conocemos que la probabilidad de que un objeto de la clase j aparezca es $P(w_j)$, y entonces en ausencia de cualquier otro conocimiento podemos minimizar la probabilidad de error, asignando el objeto desconocido a la clase para la cual $P(w_j)$ es máxima.

Las decisiones sobre la clase de un objeto se realizan usualmente basándose en observaciones de rasgos. Supongamos que la probabilidad $P(x/w_j)$ fue dada. La misma representa la probabilidad de que el objeto pertenezca a la clase w_j dado que el vector de rasgos observados es x. Basándonos en este conocimiento podemos calcular la probabilidad a posteriori **$P(w_j/x)$.**

La probabilidad a posteriori es la probabilidad que, para la información y observaciones dadas, el objeto desconocido pertenece a la clase w_j. Usando la regla de Bayes:

$$P(w_j/x)=\frac{P(x/w_j)P(w_j)}{P(x)} \quad donde P(x)=\sum_{j=1}^{m} P(x/w_j)P(w_j) \qquad \textbf{(Ec 4.8)}$$

El objeto desconocido debería ser asignado a la clase con la mayor probabilidad a posteriori.

Para el clasificador de Bayes se tiene en cuenta el riesgo condicional. Si el clasificador decide que x proviene de la clase w_j, incurrió en una pérdida llamada L_{ij}.

Como el patrón x pudo pertenecer a cualquiera de *m* clases a considerar, el promedio de pérdida incurrida en asignar x a la clase w_j es

(Ec 4.9)

$$r_j(x)=\sum_{k=1}^{m} L_{kj} P(w_k/x) \quad \textit{esto se llama riesgo o pérdida promedio condicional}$$

Si reemplazamos en la fórmula anterior usando **(Ec 4.8)** obtenemos:

$$r_j(x) = \left(\frac{1}{P(x)}\right) \sum_{k=1}^{m} L_{kj}\, P(x/w_k) P(w_k) \qquad \textbf{(Ec 4.10)}$$

donde $P(x/w_k)$ es la función de densidad de probabilidad de los patrones de la clase w_k y $P(w_k)$ es la probabilidad de ocurrencia de la clase w_k

Dado que 1/P(x) es positivo y común a todos los $r_j(x)$ j= 0, ..., m se puede sacar de la ecuación sin afectar el orden relativo de estas ecuaciones desde la de menor a la de mayor valor.

Entonces se reducen a:

$$r_j(x) = \sum_{k=1}^{m} L_{kj} P(x/w_k) P(w_k) \qquad \textbf{(Ec 4.11)}$$

El clasificador tiene M clases posibles entre las que puede elegir para cualquier patrón desconocido x. Se computa: $r_1(x)$, $r_2(x)$, ..., $r_m(x)$ para cada patrón x , y se asigna el patrón a la clase con menor pérdida. El total de pérdida con respecto a todas las decisiones será mínimo.

Finalmente, se establece la siguiente regla de decisión. Sean w1,..., wm las posibles clases entre las que el clasificador puede asignar el objeto desconocido.

$r_i(x) < r_j(x)$ Þ x Î wi $\quad \forall\, i, j = 1, 2, ..., m.\ i \neq j$

Existen otros enfoques que utilizan como base la filosofía del clasificador de Bayes. Por ejemplo, en sensado remoto puede ser útil realizar la clasificación teniendo en cuenta las diferentes consecuencias, y por lo tanto también costos, asociados con cada tipo de error. Existen aplicaciones donde se busca minimizar el costo del error de la clasificación, en lugar del error general de clasificación **[Bru2000]**.

- Clasificador de Máxima Verosimilitud [Fuk90][Jen96][Lil94][www2]

Este clasificador asume que los datos de entrenamiento para cada clase en cada banda tienen una distribución normal o Gaussiana. En otras palabras, los datos con histogramas bi-modales o trimodales en una sola banda no son ideales.

Utiliza como estadísticas el vector de media (M) y la matriz de covarianza (V) de cada clase para cada banda. La regla que se aplica para un vector desconocido x es: se decide por la clase c, si y sólo si: $P_c >= P_i$ con i=1, 2, ..., *m* (clases posibles), donde P_c es la probabilidad de que x pertenezca a la clase c.

Entonces para clasificar un vector de medidas x de un pixel u objeto desconocido, se debe calcular el valor de Pc para cada clase. Luego, se asigna el objeto a la clase con mayor probabilidad. Si todas las probabilidades están por debajo de un umbral dado por un analista, se asigna a la clase desconocida.

Esta ecuación asume que cada clase tiene igual probabilidad de ocurrir, es decir las $P(w_j)$ con j=1, ..., *m* tienen todas igual valor. Esto difiere del clasificador de Bayes que no realiza tal suposición. Bayes aplica dos factores de peso a la estimación de la probabilidad. Primero, el analista determina la probabilidad a priori, o verosimilitud anticipada de ocurrencia para cada clase en la escena dada. Segundo, hay un peso asociado con el riesgo de clasificar mal para cada clase. Estos dos factores juntos actúan para minimizar el costo de clasificar mal, resultando en una clasificación teóricamente óptima.

Tanto el clasificador de Máxima verosimilitud como el de Bayes tienen un alto costo computacional. Esto se incrementa particularmente cuando se tiene un gran número de canales espectrales involucrados y/o muchas clases entre las que se debe distinguir.

Entre las aplicaciones de sensado remoto que utilizan este clasificador se encuentra el estudio de mares congelados en las regiones polares para entender el clima global y los procesos geofísicos que gobiernan los cambios climáticos **[Rem2000]**.

Enfoque neuronal [Bis95][Fuk90][www5]

La alternativa de ***computación neuronal*** surge de intentar imitar la forma en que los sistemas neuronales biológicos guardan y manipulan información. Esto lleva a una clase de sistemas neuronales artificiales denominados redes neuronales. Se los considera técnicas de modelado sofisticado, capaces de modelizar funciones extremadamente complejas. En particular, las redes neuronales son no lineales. Durante muchos años los modelos lineales han sido la técnica más frecuentemente usada, dado que los mismos eran conocidos por encontrar buenas soluciones en problemas de optimización. Sin embargo, en aquellos problemas donde la aproximación lineal no resultaba válida, los modelos no se comportaban correctamente.

La unidad básica de las redes neuronales, la neurona artificial, simula las cuatro funciones básicas de las neuronas naturales, pero son mucho más simples. Básicamente, una neurona biológica recibe una entrada de otras fuentes, las combina de algún modo, realiza una operación no lineal en el resultado y luego da como salida el resultado final.

Es importante aclarar que, generalmente, las redes neuronales toman una entrada numérica y producen también una salida numérica. La función de transferencia de una unidad se elige de manera tal que acepte una entrada en cualquier rango y la salida sea limitada a un cierto rango. Este último hecho, junto con el que hay que exigir que la información esté en forma numérica, implica que las soluciones neuronales requieren de etapas de pre y post procesamiento para cuando se trabaja en aplicaciones reales **[Bis95]**.

Cada entrada se multiplica por un peso de conexión, estos pesos serán representados como wij (**ver gráfico 4.1**). En el caso más simple, en que los productos son sumados, se aplica una función de transferencia para generar un resultado que es la salida.

Para implementar una red neuronal se debe atravesar por un período de prueba y error en el diseño de la misma antes de encontrar el diseño satisfactorio. El diseño se torna en un tema complejo para quienes optan por utilizar una red neuronal.

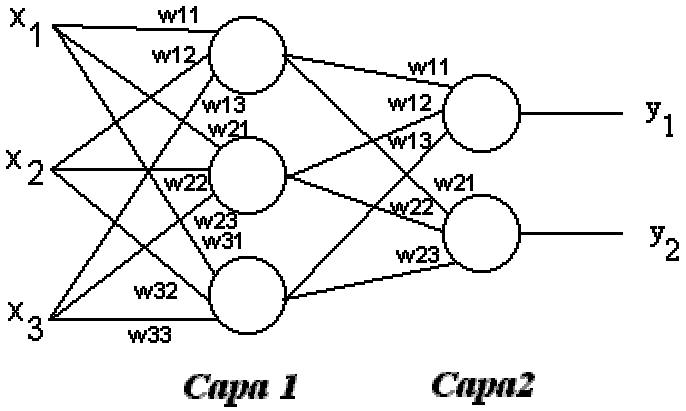


Gráfico 4.1 – Ejemplo de la estructura de una red neuronal

Diseñar una red neuronal consiste en [Dug96]:

- Arreglar las neuronas en varias capas.
- Decidir el tipo de conexión entre las neuronas de diferentes capas, así como entre las neuronas de una misma capa.
- Decidir la forma en que una neurona recibe una entrada y produce una salida.
- Determinar el peso de cada conexión dentro de la red, permitiendo que la red aprenda los valores apropiados de los pesos de conexión, usando el conjunto de datos de entrenamiento.

Se utilizan estas redes como medios para desarrollar adaptativamente los coeficientes de las funciones discriminantes, a través de presentaciones sucesivas de conjuntos de patrones de entrenamiento.

Existen diferentes arquitecturas de redes entre las que se mencionan la de “Multilayer Perceptron”, “RBF (Radial Basis Function)”, Kohonen, etc. No se

desarrollará aquí una descripción detallada de cada una de estas arquitecturas, sólo se verán algunos conceptos generales.

La primera de ellas, una de las más populares, permite modelizar funciones complejas, y el número de capas y de neuronas determinan dicha complejidad. Para estas redes se debe determinar el valor adecuado para los parámetros libres, es decir los pesos y los valores de umbral que permitan minimizar el error.

Los algoritmos de entrenamiento de la red pueden ser supervisados o no supervisados, sin embargo los primeros son los más comunes. En los supervisados, el usuario propone un conjunto de datos de entrenamiento, que contienen ejemplos de datos de entrada junto con su correspondiente salida. La red aprende a inferir la relación entre ambas. Un algoritmo de entrenamiento supervisado conocido y muy usado es el de "Back Propagation", donde se calcula el vector gradiente de la superficie de error. Los datos de entrenamiento determinarán también el correcto funcionamiento de la red. La cantidad de datos de entrenamiento debe ser tal, que permita encontrar el mínimo error de la superficie y no estancarse en mínimos locales.

Las redes de Kohonen son diseñadas para realizar entrenamiento no supervisado, donde sólo tenemos las variables de entrada y se intenta aprender de la estructura de los datos. Dichas redes pueden aprender a reconocer agrupamientos de datos similares y pueden también relacionar clases similares entre sí. Cuando se encuentran datos nuevos, la red no puede reconocerlos y esto indica una novedad. Una red Kohonen sólo tiene dos capas: la de entrada y la de salida de unidades radiales. Se entrenan mediante un algoritmo iterativo, comenzando con un conjunto de centros radiales aleatorios, el algoritmo los ajusta gradualmente para reflejar los agrupamientos en los datos de entrenamiento.

La principal ventaja encontrada en utilizar redes neuronales para aplicaciones de sensado remoto es su posibilidad de hacer clasificaciones a partir de múltiples

fuentes. Las redes que se utilizan en sensado remoto, generalmente, contienen una capa de entrada que está compuesta por nodos que se activan con los datos de la imagen, y una capa de salida cuyos nodos representan las clases posibles para las que se hace el entrenamiento. Ambas capas están separadas por una o más capas escondidas. Un nodo de una capa se engancha con todos los nodos de las capas por debajo. El número de iteraciones, el de capas escondidas y los radios de aprendizaje se determinan por prueba y error durante el proceso de entrenamiento, ya que no existe una metodología precisa para la selección de la red óptima en una determinada aplicación.

Existen algunos trabajos realizados en el campo de sensado remoto utilizando redes neuronales, algunos de ellos combinándolas con técnicas de "wavelets" **[www2]**.

Clasificación no supervisada [Lan99][www12]

Existen situaciones en las que se dispone de un conjunto de datos relacionados por ciertas características, pero no se conoce a priori la clasificación de los mismos. Es más, puede no conocerse a priori la cantidad de clases presentes. En estos casos, se pueden usar los métodos de agrupamiento para dar un entendimiento de la estructura de los datos y una medida numérica útil acerca de las características de los mismos.

Además, los métodos de agrupamiento permiten reducir la cantidad de información agrupando ítems de datos similares. Como se ha dicho, una de las razones para utilizar algoritmos de clasificación es la obtención de herramientas automáticas para ayudar a construir categorías o taxonomías.

El problema del "clustering" se puede ver como el problema de la taxonomía biológica, donde las plantas y animales son clasificados según ciertas características en reinos y familias, "creados" a partir de información anterior de otras plantas y animales. Aquellas plantas o animales que no se correspondan adecuadamente con alguna de las categorías existentes podrían ser miembros de una nueva categoría o "cluster".

Los métodos de agrupamiento pueden ser clasificados en dos tipos: por partición o jerárquicos.

Estos últimos trabajan mezclando pequeños "clusters" o dividiendo "clusters" grandes. Los métodos pertenecientes a este tipo, difieren en la forma de decidir cuáles son los "clusters" pequeños que deben unirse o cuáles los grandes que deben dividirse. En todos los casos, el resultado final del algoritmo es un árbol de "clusters" que muestra la forma en que están relacionados. El resultado del agrupamiento estaría determinado por el nivel de corte de dicho árbol.

Por otro lado, los métodos de agrupamiento por partición intentan descomponer el conjunto de datos en clases disjuntas. Típicamente el criterio global busca minimizar alguna medida de disimilitud entre las muestras pertenecientes a una misma clase mientras que busca maximizar la diferencia entre diferentes clases.

A continuación se detallan algunos métodos de agrupamiento analizados.

a) Con número de clases conocidas.

- ***Método de las k-medias*** **[Mar94][www10][www11]**

El nombre de este algoritmo hace referencia a que se conoce a priori que existe un número k de clases o patrones involucrados en el problema. Es un algoritmo sencillo, pero muy eficiente, siempre y cuando se conozca el valor de k con exactitud.

Partiendo de un conjunto de objetos a clasificar: X_1, X_2, ..., X_p, el algoritmo realiza las siguientes operaciones:

1. Se seleccionan al azar entre los elementos a agrupar k vectores, de forma de constituir los centroides de las k clases. Recordar que k es el número de clases, ingresado por el usuario.

2. Se realiza un proceso recursivo, en el que en una cierta iteración genérica n se distribuyen todas las muestras entre las k clases de acuerdo a la menor distancia de la muestra y los centroides de dichas clases.

3. Una vez redistribuidos los elementos a agrupar entre las diferentes clases, es preciso calcular nuevamente o actualizar los centroides de las clases.
4. Se repite el proceso (a partir de 2) hasta que no existan cambios significativos en los centroides de las clases respecto de la iteración anterior.

De acuerdo un trabajo de investigación realizado sobre estos métodos de agrupamiento, se obtuvieron los siguientes resultados de acuerdo a dos opciones implementadas para el paso 1:

Opción 1: Se eligen los k primeros puntos sin importar si hay valores repetidos.

Opción 2: Se eligen los primeros k puntos distintos.

Para la opción 1, la imagen se recorre por filas hasta reunir k elementos distintos. En caso de no haber en la imagen k puntos distintos, se reduce el valor de k a la cantidad de valores encontrados.

Breve análisis del método

Este método es fuertemente dependiente del k seleccionado por el usuario.

Si seleccionamos un valor de k mayor que la cantidad de clases existentes en la imagen de entrada, se observa que se crean clases ficticias.

Para imágenes cuyos patrones se encuentran dispersos, puede observarse que se crean clases que no contienen elementos. Esto se debe a que los centros están muy separados entre sí y que la variedad de patrones es inferior al número de clases.

También se puede observar que cuando se tiene un valor de k mayor que la cantidad real de clases existentes, y se elige la opción 1 para seleccionar los centros iniciales, no se suele lograr una clasificación correcta.

- ***Método Isodata*** **[Tou74][www8][www9]**

ISODATA es el acrónimo de la definición en inglés: Iterative Self-organizing Data Analysis Techniques, al que se le agregó la letra A para conseguir una palabra más fácilmente pronunciable.

Este método está basado en el algoritmo de k-medias, con el agregado de una cantidad de parámetros y operaciones que llevan a mejorar ciertos aspectos del mismo. Esto a su vez, le da un mayor contenido heurístico y el usuario deberá tener conocimiento de la medida en que influye cada uno de estos parámetros en el resultado.

A diferencia del algoritmo de las k-medias, el valor de k utilizado en el método Isodata es un valor esperado de clases, no un número exacto. Por ello, el algoritmo inicialmente considera un número A de clases, que a lo largo de la ejecución se trata de aproximar a k.

Como ya se mencionó anteriormente, existen situaciones en las cuales el método de las k-medias genera clases con muy pocos elementos o simplemente vacías. Por ello, la primer acción del ISODATA es eliminar estas clases y lo hace utilizando el parámetro θ_N, que representa la cantidad mínima de elementos que debe contener una clase para considerarse como tal.

Se vio para k-medias, que cuando el número de clases consideradas es menor que la cantidad real de clases existentes, se producen dispersiones muy grandes, precisamente causadas por los agrupamientos forzados de elementos.

El algoritmo ISODATA utiliza como criterio de división:

- La relación entre la dispersión por agrupamiento (Dj dispersión del agrupamiento j) y la dispersión global (D) y el número de elementos del agrupamiento.

Este criterio es muy débil cuando se trata de clases con dispersiones no uniformes, ya que un alto porcentaje de clases verificarán la condición Dj > D. Esto no podría solucionarse mediante θ_N ya que si se aumenta demasiado el valor de este parámetro se descartarían demasiadas clases en el paso de eliminar agrupamientos.

- La cantidad de clases existentes hasta el momento. En este caso θ_S (desviación típica máxima) se utiliza para solucionar el problema de tener un número real de clases menor que el esperado y una dispersión baja.

Suponiendo que el número de agrupamientos sea demasiado alto, se producirán clases cuyos centros están muy cercanos. Para solucionar este problema, el método ISODATA une agrupamientos teniendo en cuenta la distancia entre los centros. Se utiliza el parámetro θ_C como cota superior para el valor de las distancias a considerar, y el parámetro L como límite de la cantidad máxima de clases a unir por iteración.

Ventajas y Desventajas

- Provee mejores resultados que el método de k-medias
- Es fuertemente dependiente de los parámetros ingresados, por lo que hay que tener suficiente conocimiento sobre ellos.
- Permite una mayor interacción con el usuario, mediante el ajuste de los parámetros. Si se tiene un conocimiento del tipo de imagen con la que se trabaja se logran los resultados deseados.

b) Con número de clases desconocidas.

- ***Algoritmo adaptativo*** **[Mar94]**

Éste es un método heurístico incremental que emplea únicamente dos parámetros muy relacionados:

ô: umbral de distancia para crear agrupamientos

θ: fracción de ô que determina *total confianza*.

La líneas generales del algoritmo son las siguientes:

1. La parte esencial del algoritmo es crear agrupamientos en base al umbral de distancia ô (ponderado por θ). El primer agrupamiento se establece arbitrariamente.
2. Cuando un patrón se asigna a un agrupamiento, se recalcula el centro del agrupamiento. Este cálculo puede hacer que algunos patrones cambien de estado, eliminándose del agrupamiento al que pertenecían o asignándose a un agrupamiento.
3. Los cambios de estado son posibles porque se hacen repetidas pasadas del conjunto de patrones utilizado para agrupar. Este proceso repetitivo se termina cuando no hay reasignaciones: la partición se considera estable.
4. Se utiliza una parte, N, del total de patrones a agrupar, M, para definir los agrupamientos.

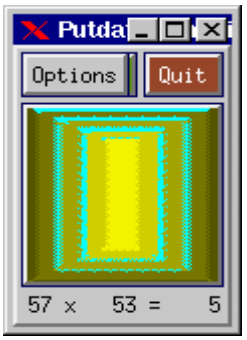
Uno de los problemas más graves de este método es su fuerte dependencia del orden de presentación de los patrones de entrada. Dado que los centros se van adaptando a medida que ingresan los patrones, se puede llegar a clasificaciones erróneas si se trata de clases solapadas (**Figura 4.2 a y b**).

Figura 4.2 a - Imagen original

Figura 4.2 b - Imagen resultante de aplicar método adaptativo con parámetros q = 0,5 y t =1. Clases solapadas

También, es importante la elección de los valores θ y τ, ya que esto incidirá en el tamaño de la región de aceptación, indeterminación o rechazo. (**Figura 4.3**)

Figura 4.3- Resultado de aplicar el método con un valor de $\tau = 50$ y $\theta = 0{,}5$. Al aumentar τ las clases se van uniendo y quedan menos clases, pero más distinguibles. El color celeste pertenece a la clase de rechazo.



Para la imagen de la **Figura 4.2 a,** el método no encuentra una solución aceptable. Para clases con centros perfectamente separados el método da buenos resultados como puede verse en la **Figura 4.4.**

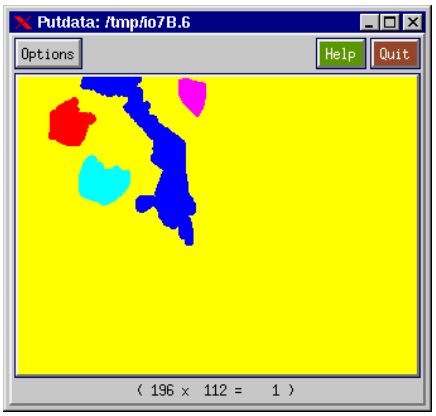


Figura 4.4 - Resultado de aplicar el método. Se obtienen las mismas clases que en la imagen original.

Ventajas:

- Es un método simple
- Es rápido ya que realiza cálculos sencillos
- Se utiliza un almacenamiento mínimo ya que los patrones se recorren secuencialmente, y sólo se guarda la información de los centros existentes hasta el momento.

- No es necesario conocer a priori el número de clases

Desventajas:

- Supone agrupamientos compactos y separados claramente. Presenta problemas cuando hay solapamientos.
- Está sesgado por los primeros patrones
- Depende del orden de presentación de los patrones
- Depende de los valores de θ y τ seleccionados.
- No permite descartar clases con muy poca cantidad de elementos.

- ***Algoritmo de Batchelor y Wilkins*** **[Mar94]**

Este algoritmo, propuesto por Batchelor y Wilkins, es llamado también de máxima distancia. Se trata de un método heurístico incremental que emplea un único parámetro:

f: fracción de la distancia media entre los agrupamientos existentes. Se usa para calcular un umbral de distancia para decidir si se crea un agrupamiento. 0 <= f <= 1.

Las líneas generales del algoritmo son las siguientes:

1. Se crea un agrupamiento si la distancia de un patrón al agrupamiento más cercano supera un valor umbral. El primer agrupamiento se establece arbitrariamente.
2. En este algoritmo, a diferencia del adaptativo, el umbral de distancia no es fijo y se calcula en base al parámetro y a la distancia media entre los agrupamientos existentes en el momento de su evaluación.
3. El aprendizaje termina cuando no se crean nuevos agrupamientos.

Resumiendo, trata de mejorar la dependencia de los datos de entrada que tiene el método anterior, tomando como criterio para la creación de clases un factor (factor de creación, f) de la distancia media entre los "clusters" existentes.

Esto si bien introduce una mejora en la clasificación, agrega tiempo de cálculo ya que todos los patrones que no han sido clasificados hasta el momento son inspeccionados para ver cual puede convertirse en el próximo centro. Esto implica que la cantidad de consultas sobre cada patrón es del orden n^2.

En cuanto a la memoria, este método está en desventaja con respecto al anterior, ya que es necesario guardar información sobre los patrones no clasificados.

Dado que los centros de los "clusters" no son recalculados a medida que se incorporan patrones a las clases, se pueden presentar problemas en caso de clases muy dispersas.

Para la imagen original presentada en la **Figura 4.2 a**, este método no logra resolver adecuadamente el problema de la clasificación. Para valores altos de f (factor de creación), el método tiende a decrementar la cantidad de clases. Para valores más pequeños se obtienen imágenes con mayor cantidad de clases, pero solapadas (**Figura 4.5**).

a) Valor de f = 0.5 **b) Valor de f = 0.25** **c) Valor de f = 0.1**

Figura 4.5 - Aplicación del algoritmo de Batchelor & Wilkins

Un valor de f intermedio, no llega a encontrar todas las clases. Como se observa a continuación en la **Figura 4.6**:

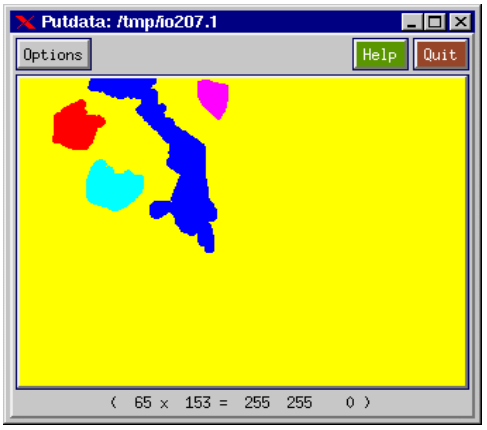


Figura 4.6 a- Imagen Original

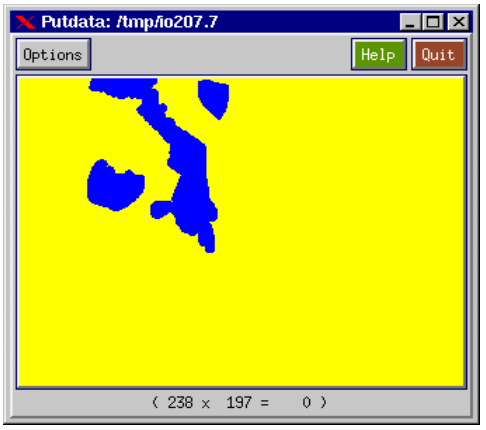


Figura 4.6 b - Para valores de f > 0.6, no se logran los resultados esperados. Para valores de f <= 0.5 dio el resultado deseado.

Con este ejemplo de la **Figura 4.5** se quiere mostrar que mediante este método para imágenes sin solapamientos, donde las clases están perfectamente separadas se pueden llegar a obtener buenos resultados dando el valor adecuado al parámetro f.

Ventajas:

- Es un método simple
- No es necesario conocer a priori el número de clases

Desventajas:

- El resultado cambia con la elección del primer patrón
- Requiere almacenamiento adicional para registrar los patrones que aún no han sido clasificados.

- Requiere una gran cantidad de cálculo por cada patrón a clasificar, ya que calcula la máxima de las mínimas distancias.

- Es necesario cuantificar la fracción de distancia entre clases a considerar para decidir si se debe crear un nuevo agrupamiento o no.

4.4. Métodos basados en modelos biofísicos

Dentro de los métodos que se utilizan para la clasificación de imágenes obtenidas con sensores remotos, encontramos un conjunto de procedimientos que permiten el análisis de datos espectrales. Entre ellos, tenemos el análisis de mezclas espectrales (spectral mixture analysis = SMA) **[Pin98] [Ust93][Ust98][www3]**. Los enfoques de mezcla asumen que las observaciones multiespectrales pueden ser entendidas como combinaciones de firmas espectrales extremas (llamadas miembros extremos) de objetos individuales presentes en la escena. SMA es un marco de trabajo integrado y estructurado que simultáneamente resuelve el problema de pixel de mezcla, la calibración y variaciones en la geometría de la luz. Los resultados se muestran en términos de proporciones de miembros que pueden ser relacionados con unidades observacionales estándares. La forma general de la ecuación de SMA para cada banda se expresa como:

$$R_b = \sum_{em=1}^{N} F_{em}\, R_{em,b} + E_b \qquad \textbf{(Ec 4.12)}$$

donde R_b *es la radiancia en la banda b*
F_{em} *es el coeficiente de fracción de cada miembro*
R_{em} *pesando su radiancia en la banda b*
E_b *es un término de error para la radiancia no modelada en la banda b*

Los miembros extremos son elegidos para explicar los materiales espectralmente distintos que forman la vaina convexa del volumen espectral.

El método es relativamente insensible a los rasgos sutiles de absorción, y produce errores de cuantificación significativos debido a la variabilidad de los miembros

extremos que surgen de mezclas lineales y no lineales (por ejemplo, por geometría de la luz y dispersión) en un pixel.

El SMA es útil para aplicaciones de mapeo de vegetación debido a la continua variación en la composición de un pixel en la escala de muchos de los satélites. Cuando se trabaja con los miembros extremos (tipos espectrales puros o clases) en el SMA, esto puede ser similar en concepto a los clasificadores tradicionales, aunque no están construidos de la misma manera. Esto es, pueden ser considerados como bloques de construcción básicos a partir de las cuales los pixeles varían en proporciones de componentes ambientales comunes (suelo, vegetación, etc.). Las clases definidas en los algoritmos de clustering son generalmente determinadas por otro criterio, por ejemplo, la categoría taxonómica (soja, maíz, etc.).

Los índices de vegetación (ver **Capítulo 2**, **Sección 7**) pueden verse como modelos lineales de dos miembros extremos, e intentan directamente estimar una planta dada a partir de la manipulación de las mediciones espectrales.

Existe otra técnica conocida como foreground/background analysis (FBA) desarrollada por Smith et al. como una mejora del SMA **[Smi94]**. En esta técnica, las medidas espectrales son divididas en dos grupos las del fondo (background) y las del primer plano (foreground) que comprenden un subconjunto seleccionado de espectros los cuales enfatizan la presencia de una firma espectral de interés.

4.5. Método de razonamiento evidencial

Como se dijo, esta investigación tiene como objetivo estudiar técnicas de clasificación de imágenes obtenidas por sensado remoto. Se vienen produciendo numerosos avances en el campo de “remote sensing”, tales como resoluciones espaciales y espectrales altas y que pueden poseer bandas de información numerosas y propiedades estadísticas diversas (por ejemplo, imágenes hiperespectrales). Todo esto, sumado a la necesidad de combinar datos de múltiples fuentes auxiliares (ejemplo: modelos digitales de elevación, datos climatológicos, datos temáticos de

un sistema de información geográfico) ha llevado a buscar salidas alternativas a los métodos de clasificación convencionales **[Ped95 a]**.

Se presenta aquí un posible enfoque alternativo a los métodos de clasificación tradicionales basado en la teoría de Dempster – Shafer **[Dem67][Sha76][Mur98]**. Este método ha sido utilizado en aplicaciones para clasificación de bosques y de hielos permanentes en Canadá utilizando imágenes multiespectrales **[Ped93]**.

La teoría matemática de la evidencia fue propuesta por Shafer (1976) como una extensión y refinamiento de la Regla de Dempster de combinación (Dempster, 1967). La misma provee una base general y heurística para integrar cuerpos distintos de información a partir de fuentes independientes.

A pesar que la teoría es general en naturaleza, y podría ser aplicada a cualquier problema que requiera agrupamiento de información para determinar la mejor respuesta a partir de un conjunto de opciones, será aquí presentada en términos de la nomenclatura y principios de la clasificación de sensado remoto. Para un pixel dado, la tarea de una clasificación es asignar el pixel a un miembro de un conjunto de clases.

Definiciones

Definición 4.1 **- Marco de discernimiento:**

En la teoría de evidencia **[Gar95][Rus92]**, el conjunto de todas las clases posibles constituye el marco explícito dentro del cual la discriminación ocurre, y se refiere como el marco de discernimiento (U). Por ejemplo: si estamos tratando de determinar la enfermedad de un paciente, el marco de discernimiento es el conjunto de todas las enfermedades posibles que puede presentar el mismo.

A cada subconjunto S de U se le asocia una medida de **soporte o evidencia** y una de **plausibilidad**.

Definición 4.2 **- Evidencia o soporte:**

El soporte se define como la masa o evidencia que se tiene a favor de una clase. Usualmente es un número real entre 0 y 1, inclusive.

Definición 4.3 **- Función de creencia o vector de evidencia:**

Para una fuente de datos dada, el conjunto de todas las masas sobre el marco de discernimiento es una función de creencia, la cual en esta investigación también se llama **vector de evidencia**.

Ejemplo: sea U={A, B, C}, y m(A)= 0.24, m(B)=0.5, m(C)= 0.10 siendo m el soporte para una cierta fuente.

Entonces el vector de evidencia o función de creencia para esa fuente es: [0.24, 0.5, 0.10]

Definición 4.4 **- Plausibilidad:**

Además del soporte evidencial, la teoría considera una medida de **plausibilidad**, o la cantidad de evidencia que falla para refutar una proposición. La plausibilidad representa la evidencia que no refuta una proposición, y se calcula como 1 menos la suma del soporte para todas las otras proposiciones.

$$Pj = 1 - \sum_{i=1}^{m} s_i \;\; \forall i \neq j$$

donde Pj es la plausibilidad para la clase j y s_i es el soporte para la clase i.

En el contexto de una clasificación de sensado remoto, la plausibilidad para la clase Ci podría ser computada como 1- S(¬Ci), donde S representa el soporte evidencial. La verdadera factibilidad de una proposición está dentro del rango de valores posibles en el intervalo que va desde la medida de soporte a la de plausibilidad para la clase Ci, el cual es llamado **intervalo evidencial**. El uso del intervalo evidencial permite que tanto el soporte en favor del rótulo de clase y el nivel asociado de incertidumbre sean incluidos en una regla de decisión.

Ejemplo: supongamos que dos médicos examinan un paciente y están de acuerdo en que sufre o de varicela (V), o rubéola (R), o de sarampión (S). Entonces U={V, R, S}. Sin embargo, difieren en el diagnóstico teniendo cada uno en cuenta su conocimiento previo, análisis realizados al paciente, etc.:

Médico 1 (M1), Médico 2 (M2)

M1({V}) = 0.70

M1({R}) = 0.10

M1({S}) = 0

M2({V}) = 0.30

M2({R}) = 0.20

M2({S}) = 0.40

Como se puede apreciar en los valores anteriores, están de acuerdo en la baja creencia de que sea rubéola, pero el médico 2 cree más en un sarampión que en varicela. Sin embargo, el médico 1 parece tener la suficiente evidencia para asegurar que es varicela.

Los valores otorgados a cada posible subconjunto de interés de U constituyen el **soporte o evidencia** a su favor. Cada médico constituye una fuente diferente de información.

La función de creencia o vector de evidencia es:

Para la fuente M1 = [0.70, 0.10, 0]

Para la fuente M2 = [0.30, 0.20, 0.40]

Suma ortogonal [Dem67][Ped95 b]

Dados los vectores evidenciales (por ejemplo, calculados para un pixel en 1 conjunto de datos multifuente o en el caso anterior para cada médico), la tarea que queda es

combinar la evidencia de todas las fuentes en una forma unidimensional conteniendo una única medida de soporte y plausibilidad para cada clase.

La descomposición de los vectores evidenciales de una fuente específica en una función de masa resultante se alcanza por suma ortogonal usando la Regla de Dempster de combinación. El poder de esta regla puede ser aplicado a cualquier número de fuentes, cada una de las cuales conteniendo evidencia para un conjunto de clases.

La suma ortogonal de evidencia a partir de dos fuentes trabaja multiplicando secuencialmente la evidencia para una clase dada de una fuente por la evidencia de cada clase de la siguiente fuente. Luego, se aplica un factor de normalización que corrige para cualquier masa que haya sido adjudicada al conjunto vacío.

La suma ortogonal de dos vectores de evidencia m1 y m2 se denota por m1 $\oplus$ m2

Supongamos que trabajamos con conjuntos de clases simples, entonces la ecuación de la suma ortogonal de Dempster para determinar la nueva masa m' asignada a la proposición n-ésima de A puede escribirse como:

$$m'(A_n) = K^{-1} \sum_{A_i \cap A_j = A_n} m_1(A_i)\, m_2(A_j) \qquad \textbf{(Ec 4.13 a)}$$

$$K = 1 - \sum_{A_i \cap A_j = \Phi} m_1(A_i) m_2(A_j) \qquad \textbf{(Ec 4.13 b)}$$

Continuemos con el ejemplo anterior **[San2000]**:

Tabla 4.1 – Cálculo de suma ortogonal dos fuentes, de acuerdo al ejemplo

⊕	Fuente: médico 1			
Fuente: médico2	M1(V)=0.70	M1(R)=0.10	M1(S)=0	M1(Θ)=0.20
M2(V)=0.30	M1(V)*M2(V)=0.21	M1(R)*M2(V)=0.03 ϕ	M1(S)*M2(V)=0 ϕ	M1(Θ)*M2(V)=0.06
M2(R)=0.20	M1(V)*M2(R)=0.14 ϕ	M1(R)*M2(R)=0.02	M1(S)*M2(R)=0 ϕ	M1(Θ)*M2(R)=0.04
M2(S)=0.40	M1(V)*M2(S)= 0.28 ϕ	M1(R)*M2(S)=0.04 ϕ	M1(S)*M2(S)=0	M1(Θ)*M2(S)=0.08
M2(Θ)=0.10	M1(V)*M2(Θ)=0.07	M1(R)*M2(Θ)=0.01	M1(S)*M2(Θ)=0	M1(Θ)*M2(Θ)=0.02
M1 ⊕M2	$K^{-1} \sum$ m(V)=0.666	$K^{-1} \sum$ m(R)=0.137	$K^{-1} \sum$ m(S)=0.156	K=1- m(ϕ)=0.51
Plausibilidad	0.707	0.178	0.197	

La función de creencia o vector de evidencia resultante es:

M1 Å M2= [0.666, 0.137, 0.156]

Una regla de decisión puede ser elegir la clase con mayor soporte **[Lee87][Wil90]**. Para este ejemplo la clase con mayor soporte es la V (varicela). Otra opción es seleccionar la clase con mayor plausibilidad **[Kim90][Sri90]**. También autores como Peddle han optado por la mayor suma de soporte y plausibilidad **[Ped95 a]**.

Especificación de conocimiento

Una de las dificultades en la aplicación de la teoría matemática de evidencia para una aplicación dada, es que el marco evidencial no incluye especificación de cómo derivar o crear las medidas de soporte y de plausibilidad que se usan como entrada al procedimiento (esto también afecta la computación de medidas de incertidumbre)

En algunas instancias, esto no es un problema si la información disponible puede ser considerada evidencia apropiada a ser usada como entrada directa de la teoría de evidencia. Sin embargo, puede no ocurrir tal situación y se debe crear una interface para el razonamiento evidencial que permita derivar los valores de soporte y plausibilidad necesarios.

Obviamente, la habilidad de esta interface para trasladar la información disponible en evidencia, es crítica para lograr las ventajas ofrecidas por el razonamiento evidencial y afecta el grado de éxito obtenido.

Los datos de imagen de sensado remoto no proveen medidas directas de evidencia para la entrada al clasificador, y por eso se debe aplicar un proceso separado para derivar las medidas requeridas. Peddle propone computar las medidas de evidencia a partir de datos de entrenamiento en un marco de **clasificación supervisada [Ped95 b]**.

Este tipo de clasificación involucra al analista de la imagen para que determine el conjunto de clases con las que se trabajará y para cada clase, áreas representativas dentro de la imagen (entrenamiento del clasificador).

La **frecuencia de ocurrencia** de los valores de pixel individual constituye la base para formar vectores evidenciales para todas las clases.

Las premisas subyacentes para este método son que:

1) **los datos de entrenamiento contienen evidencia para el conjunto de clases**

2) **la frecuencia de ocurrencia de un valor dado en el conjunto de entrenamiento representa las magnitudes de soporte para la clase.**

Ejemplo:
Supongamos que se tiene un pixel a clasificar cuyos valores son (0.45, 0.9, 198); el primer valor se corresponde con la fuente 1, el segundo con la fuente 2 y así siguiendo, y además se tienen las siguientes cantidad de muestras para cada fuente y clase para el valor del pixel correspondiente:

Tabla 4.2 - Cantidad de muestras por fuente y clase. Tsn se refiere al total de muestras para una clase

	TSn	**Fuente 1** (0.45)	**Fuente 2** (0.9)	**Fuente 3** (198)
Clase1	223	56	23	10
Clase2	300	80	12	12
Clase3	70	1	34	14
Clase4	90	24	44	33

¿Cómo se deriva entonces la evidencia?

Tabla 4.3 a - Frecuencia de Ocurrencia para la clase 1 [Ped91]

Clase 1 (TSn)	Fuente 1						
Valor de pixel	0	0.2	0.4	0.45	0.6	0.8	1
Cantidad de ocurrencias	2	34	78	56	40	13	0

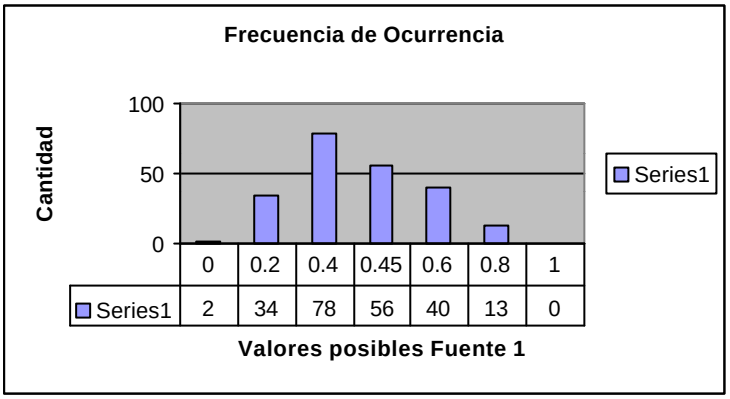


Figura 4.6 – Frecuencia de Ocurrencia para la fuente 1

Tabla 4.3 b - Frecuencia de Ocurrencia para la clase 2

Clase2	Fuente 1						
Valor de pixel	0	0.2	0.4	0.45	0.6	0.8	1
Cantidad de Ocurrencias	4	5	50	80	120	40	1

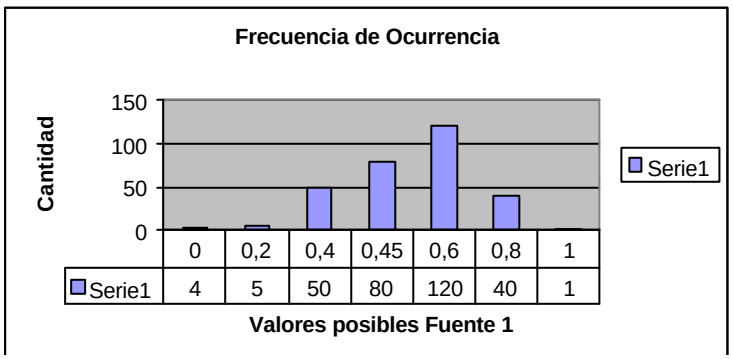


Figura 4.7 – Frecuencia de Ocurrencia para la fuente 1

Tabla 4.5 - Evidencia derivada

	TSn	Fuente 1	Fuente 2	Fuente 3
Clase1	223	56/223=0.251	0.103	0.044
Clase2	300	0.266	0.04	0.04
Clase3	70	0.014	0.485	0.2
Clase4	90	0.266	0.488	0.366

Transformación de Bin

Para datos cuantitativos (por ejemplo, radio de medidas), se puede transformar la frecuencia de ocurrencia para extender el conocimiento de los datos de entrenamiento a un rango dinámico mayor de los datos de la fuente, y reducir cualquier desvío asociado con las cuentas de frecuencias individuales.

Esto se realiza utilizando una función que depende de la distancia, y es lineal y multiplicativa. Se basa en dos principios:

- Si un valor I ocurre en los datos de entrenamiento para la clase c, entonces valores similares también son indicativos de la clase (ejemplo: para valores cuantitativos, los datos que van de + /- 1 pertenecen a la clase c)

- La probabilidad, p, de que valores similares representen la clase c incrementa con la proximidad a I, es decir, p(I+/- 1 ∈ c) > p(I+/- 2 ∈ c)

El usuario puede especificar el tamaño de bin para cada fuente individual. La propagación de la evidencia se produce en forma simétrica.

La función a aplicarse para un valor de dato de entrenamiento i es la siguiente:

$$f(j) = f(j) + a \times (b - 2 \times |i - j|) \qquad \textbf{(Ec 4.14)}$$

Donde:

a= f(i)

b= es el tamaño de “bin” especificado por el usuario

f(j)= evidencia para el dato j

y se debe cumplir que |i-j| < b/2

Esta función debe aplicarse para cada valor de muestra de todas las clases para las fuentes que el usuario indique.

El enfoque de "transformación de bin" permite que la evidencia sea distribuida consistente y objetivamente a partir de los datos de múltiples fuentes, formatos, y escalas de medición diferentes. Sin embargo, **esta transformación no puede ser aplicada a cualquier tipo de datos**, ya que por ejemplo si una de las fuentes es el resultado de una clasificación anterior, no tendría sentido extender la distribución de los datos para esa fuente en particular.

El usuario también podría tener la opción de especificar factores de peso distintos para las fuentes individuales, si hay suficiente información a priori sobre la importancia relativa de cada fuente de datos para la clasificación.

4.6. Matriz de confusión e indicador Khat

Una de las formas de expresar la precisión de la clasificación es la matriz de error, también llamada de confusión o tabla de contingencia. Las matrices de error son simplemente tablas mostrando las observaciones de datos reales de suelo en comparación con la información derivada de los datos sensados remotamente. Son matrices cuadradas, con el número de filas y columnas igual al número de categorías cuya precisión en la clasificación se está evaluando.

La matriz se desarrolla a partir de la clasificación de pixeles del conjunto de entrenamiento, se lista la cantidad de pixeles por tipo categoría (fila) versus los pixeles clasificados en cada categoría de tipo de suelo (columna).

A partir de estas matrices se pueden estudiar varios errores de clasificación como los de omisión (exclusión) y los de comisión (inclusión). Los pixeles pertenecientes a la diagonal son los que han sido correctamente clasificados, mientras que el resto representan errores.

Error de comisión: corresponden a los elementos de la columna (no pertenecientes a la diagonal)

Error de omisión: corresponde a los elementos de la fila que no pertenecen a la diagonal.

Se pueden obtener otros descriptores, como por ejemplo la precisión general, que se computa como el número total de pixeles correctamente clasificados sobre el total de pixeles de referencia.

Para analizar la matriz de error en más detalle se calcula un coeficiente de acuerdo. El mismo se conoce como **KAPPA (KIA)**, y también como **KHAT**. Esta medida estadística fue introducida por el psicólogo COHEN en 1960 y fue adoptada para la evaluación de la precisión en campos de sensado remoto por Congalton y Mead (1983) **[Con83][Ros86]**.

KIA permite evaluar la precisión de una clasificación en contraposición con los datos reales de suelo. La siguiente es la fórmula para calcular este coeficiente.

$$\kappa = \frac{N\sum_{i=1}^{r} x_{ii} - \sum_{i=1}^{r}(x_{i+} * x_{+i})}{N^2 - \sum_{i=1}^{r}(x_{i+} * x_{+i})} \qquad \textbf{(Ec 4.15)}$$

r	número de filas de la tabla de clasificación
X_{ii}	número de combinaciones de la diagonal
x_{i+}	total de observaciones en la fila i
x_{+i}	total de observaciones en la columna i
N	número total de observaciones

Resumiendo, se tiene que

$$k = \frac{Po - Ap}{1 - Ap} \qquad \textbf{(Ec 4.16)}$$

donde Po es la precisión observada y *Ap* es el acuerdo probable.

4.7. Conclusiones del capítulo

En este capítulo se presentan los métodos de clasificación más conocidos y utilizados, incluyendo ejemplos para algunos de ellos.

También, se introduce una división de los mismos en supervisados y no supervisados. El primer grupo se caracteriza por tener un vector prototipo para cada clase y tener un maestro que guía el proceso de clasificación, mientras que en el segundo, se trabaja con una agrupación natural de los datos de acuerdo a características que los hacen similares. Por otra parte, no existe ningún supervisor que guíe tal proceso.

El método de razonamiento evidencial tal como fue presentado por autores como Peddle es explicado aquí.

La razón de este capítulo es proveer un contexto para desarrollar el método que se presenta en esta tesis y su comparación con los métodos de clasificación más tradicionales. ◆