SELECCIÓN DE VARIABLES

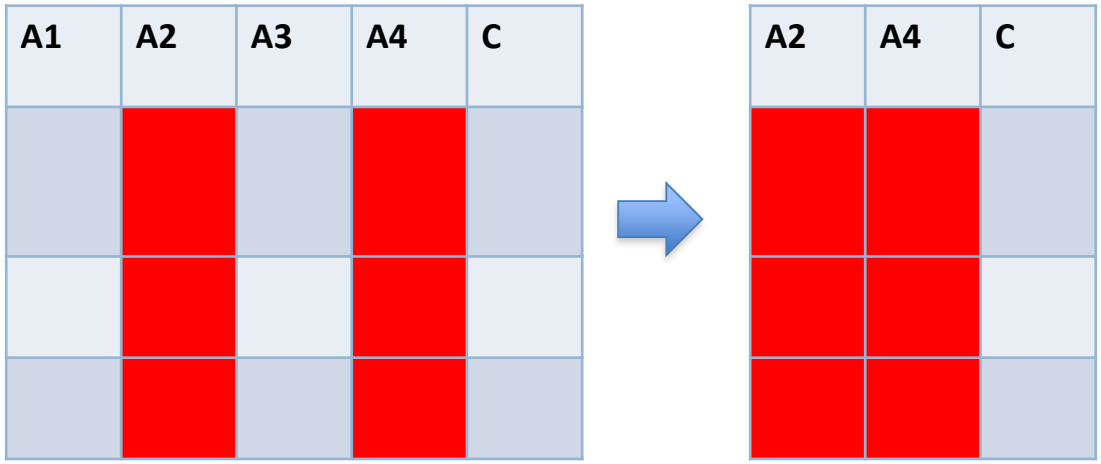
Reducción de datos

Tipos de reducción de datos

- Selección de variables
  - Consigue una reducción de datos mediante la eliminación de variables irrelevantes o redundantes
- Selección de instancias
  - Consiste en seleccionar un subconjunto de los ejemplos del dataset original para conseguir que la técnica de minería de datos consiga los mismos resultados que con todos los ejemplos
- Discretización
  - Transforma las variables numéricas en categóricas con un número finito de intervalos
- Generación de variables e instancias
  - Extiende tanto la selección de variables e instancias permitiendo la modificación de los valores que representan cada ejemplo o variable

Reducción de variables / características

- Transformar un data set de tal forma que el nuevo dataset tenga menos variables (columnas) que el original



Motivación de la reducción de variables

- Las variables redundantes, irrelevantes o el ruido en los datos pueden confundir al algoritmo de minería de datos
  - En un data set con más variables que ejemplos es imposible estimar todos los parámetros necesarios para especificar el modelo
- Los modelos con menos variables son más fáciles de explicar y entender
- La reducción de la dimensionalidad ayuda en la visualización
- Tipos de técnicas
  - Selección
  - Transformación

Selección vs. transformación

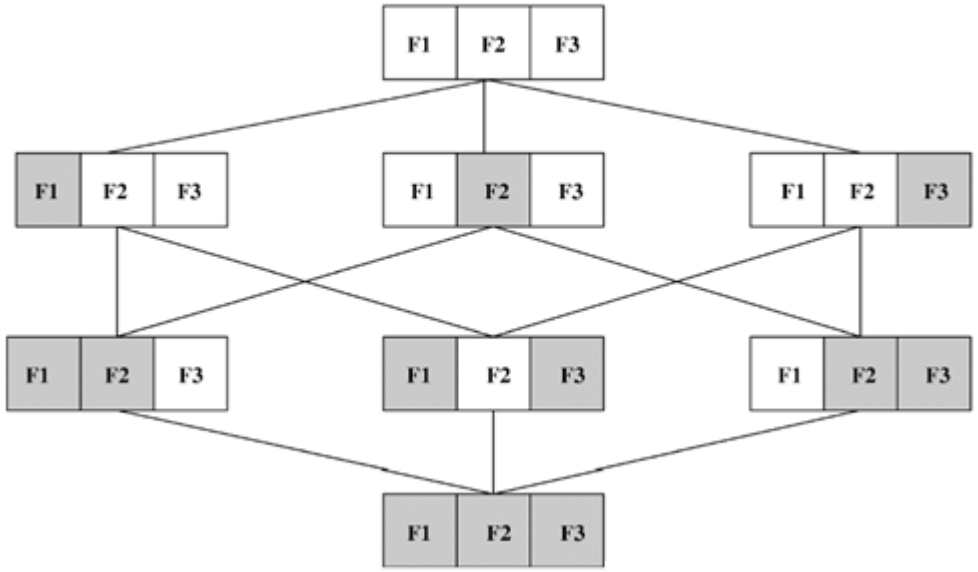
- Selección de variables: eligen algunas de las variables del data set inicial
  - Ventajas
    - Interpretabiliad del modelo obtenido
    - No alteran la representación inicial de las variables
    - Conservan la semántica original de las variables
  - Desventajas
    - Pueden implicar sobre-entrenamiento
- Transformación de variables: construyen nuevas variables a partir de las originales
  - Ventaja
    - Procedimientos bien definidos matemáticamente: PCA, MDS, LLE, FA
  - Desventaja
    - Dificultad en la interpretación de los modelos generados

Selección de variables (SV)

- ¿Son todas las variables útiles para la clasificación?
- No-monotonía: no existe relación entre el número de variables y el rendimiento
- La SV pretende elegir atributos que sean relevantes para una aplicación y lograr el máximo rendimiento con el mínimo esfuerzo
- El resultado de la SV sería:
  - Menos datos → los algoritmos pueden aprender más rápidamente
  - Mayor exactitud → el clasificador generaliza mejor
  - Resultados más simples → más fácil de entender y visualizar

Selección de variables (SV)

- La SV puede ser afrontado como un problema de búsqueda
  - Se puede representar como un array binario
    - Tantos elementos como variables
      - 0: no se selecciona la variable asociada
      - 1: se selecciona la variable asociada
- Cardinalidad del espacio de búsqueda: 2^N
  - N: número de variables



Selección de variables (SV)

- Problema muy complejo
  - Hay 2^N subconjuntos de variables posibles
    - Para N=50 implica que tenemos 1,125,899,906,842,624 subconjuntos de variables posibles
    - Si procesáramos cada subconjunto en 0,001segundos, nos costaría evaluar todos 35,702 años
      - N=30 → 12.43 días
      - N=20 → 0,29 horas
    - En todos los casos, excepto con pocas variables (N<20 por ejemplo), evaluar todos los subconjuntos es intratable
  - Al menos debemos encontrar una buena solución
    - Por tanto, no podemos garantizar encontrar el subconjunto óptimo
    - Lo mejor que podemos hacer es encontrar una buena solución evaluando algunos de los subconjuntos
- La selección de variables es un proceso en el que se escoge un subconjunto de variables óptimo para un cierto criterio
  - El criterio determina la forma de evaluar los subconjuntos de variables

Selección de variables (SV)

- Factores a tener en cuenta en la búsqueda
  - Estrategia de búsqueda
    - Exhaustiva
    - Heurística
    - No determinista
  - Tipo de técnica utilizada
    - Filtro
    - Envoltorio (Wrapper)
      - Dirección de la búsqueda
        - Hacia adelante (Sequential Forward Search)
        - Hacia atrás (Sequential Backward Search)
    - Embebida
  - Tipo de salida de la búsqueda
    - Ranking
    - Subconjunto de variables

Tipo de búsqueda

- Búsqueda exhaustiva
  - Se exploran los 2^N subconjuntos de variables
  - Garantiza encontrar el subconjunto óptimo
  - No es factible con muchas variables
- Búsqueda heurística
  - Utiliza una heurística para realizar la búsqueda
    - La elección de la heurística es clave
  - Seguramente proporciona un subconjunto sub-óptimo
  - La complejidad de la búsqueda es $O(N)$
- Búsqueda no determinista
  - Combina las dos anteriores
  - Genera subconjuntos de variables aleatoriamente
    - No hay que esperar hasta que la búsqueda acabe
    - Se comprueba si el subconjunto actual es mejor que el que se tenía

Salida: algoritmos Subconjunto de Atributos

- Devuelven un subconjunto de atributos optimizado según algún criterio de evaluación
- Algoritmo
  - Entrada
    - X: conjunto de N variables original, C: criterio de evaluación
  - Salida
    - X': subconjunto de variables
  - Pseudo-código
    - $Subconjunto = []$
    - Repetir
      - $S_k = GenerarSubconjunto(\mathrm{X})$
      - $Si\ existeMejora(Subconjunto, S_k, C)$
        - $Subconjunto = S_k$
    - Hasta *Criterio de parada*

Salida: algoritmos de Ranking

- Devuelven una lista de atributos ordenados según algún criterio de evaluación
- Algoritmo
  - Entrada
    - X: conjunto de N variables original, C: criterio de evaluación
  - Salida
    - X': conjunto de variables ordenado de acuerdo a C
  - Pseudo-código
    - Lista = []
    - For $i = 1$ to N do
      - V_i = Evaluar(X_i, C)
      - Insertar X_i en Lista de acuerdo a V_i
    - End for

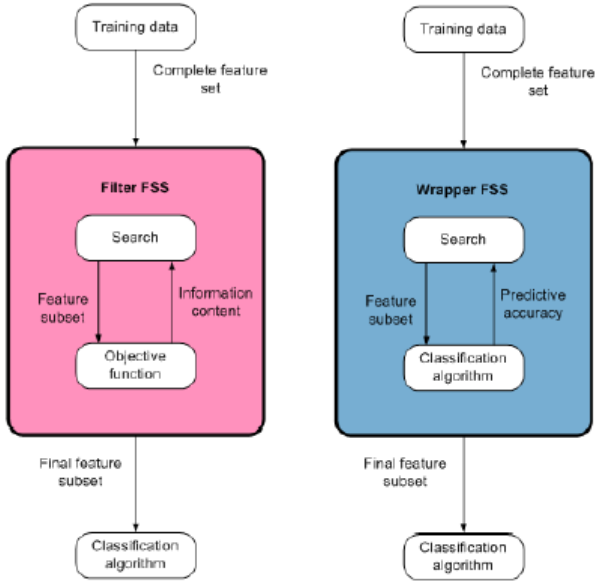
Salida: algoritmos de Ranking - Validación

- Seleccionar las Z primeras variables
  - Z valor prefijado
- Seleccionar las variables que tengan evaluaciones, V_i, menores que un umbral
  - Prefijado
  - Calculado dinámicamente de acuerdo a los V_i
- Evaluación de las variables con un clasificador

Variables	A1 A2 A3 A4 A5 A6 A7 A8 A9
Ranking	A5 A7 A4 A3 A1 A8 A6 A2 A9
Mejor Subconjunto	80 82 81 83 83 **85** 84 83 84
	A5 A7 A4 A3 A1 A8 (6 variables)

Aproximaciones para acometer SV

- Técnicas de filtro, envoltorio (wrapper) y embebidas
  - Filtros: basadas en la características intrínsecas de los datos
  - Wrapper: basadas en el conocimiento del clasificador
  - EMBEBIDOS: la selección se hace dentro del propio clasificador
- Requisito (filtros y wrappers): función de rendimiento que mida la calidad del conjunto de variables seleccionado

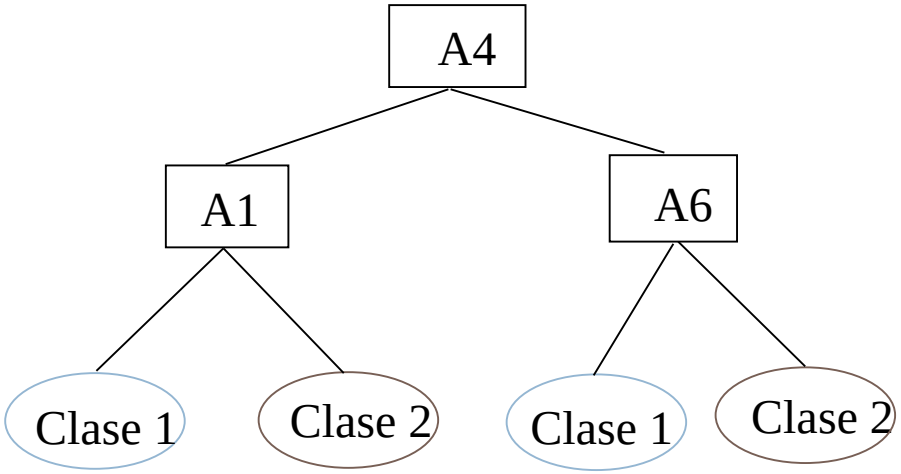


Técnicas embebidas

- El algoritmo de aprendizaje del clasificador incluye la búsqueda del subconjunto óptimo de variables
  - La búsqueda es realiza en el espacio de búsqueda que combina variables e hipótesis (clasificadores)
  - Las técnicas embebidas son específicas del algoritmo de aprendizaje
- Los árboles de decisión son un ejemplo de técnica embebida
  - Ofrecen un balance entre el rendimiento del subconjunto de variables utilizadas y el tiempo requerido para encontrarlo
  - Obtienen un subconjunto sub-óptimo de variables

Técnicas embebidas

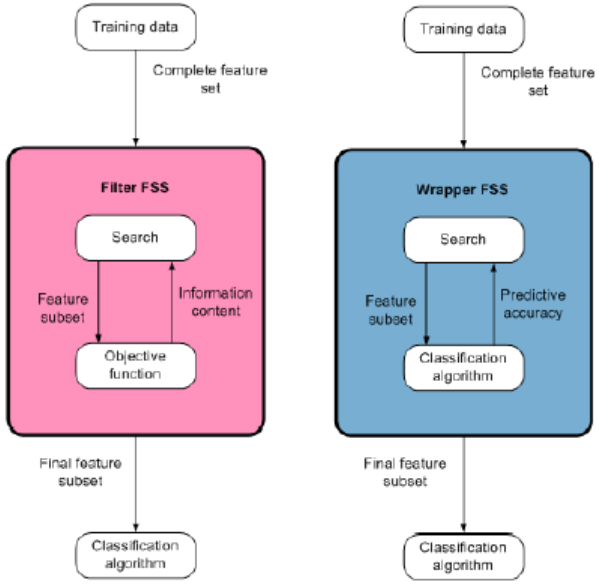
- Ejemplo: árbol de decisión
- Conjunto de variables inicial: {A1, A2, A3, A4, A5, A6}



- Conjunto de variables seleccionadas: {A1, A4, A6}

Aproximaciones para acometer SV

- Técnicas de filtro, envoltorio (wrapper) y embebidas
  - FILTROS: basadas en la características intrínsecas de los datos
  - Wrapper: basadas en el conocimiento del clasificador
  - Embebidos: la selección se hace dentro del propio clasificador
- Requisito (filtros y wrappers): función de rendimiento que mida la calidad del conjunto de variables seleccionado



Evaluación: técnicas de filtro

- La reducción se realiza ANTES de aplicar el algoritmo de aprendizaje

- Se calcula la relevancia de las variables y se eliminan las de menor importancia
  - La relevancia de las variables se mide solamente en función de las propiedades intrínsecas de los datos
- Ventajas
  - Fácilmente escalables a conjuntos de datos de muchas variables
  - Computacionalmente simple y rápido
  - Independencia del clasificador
- Desventajas
  - No tiene en cuenta la interacción con el clasificador
  - La mayoría de estas técnicas son uni-variable (ignoran las dependencias entre las variables)

Selección de variables (SV)

- ¿Cómo estimamos la calidad del subconjunto de variables? Tipos de métricas
  - Métricas de distancia
    - Variables que ayudan a separar mejor entre las clases del problema
  - Métricas de información
    - Calcular la ganancia de información (Teoría de información) de las variables
  - Métricas de dependencia
    - Calcular la correlación entre las variables y entre las variables y las clases
  - Métricas de consistencia
    - Inconsistencia: cuando existen 2 ejemplos iguales pero que son de diferentes clases
    - Estas métricas tratan de encontrar el conjunto mínimo de variables que mantienen el nivel de consistencia del dataset inicial
  - Métricas de rendimiento
    - Calcular el rendimiento del modelo en base a las variables
- Las métricas se aplican a
  - Multi-variable: subconjunto de variables
  - Uni-variable: variables individuales (generan un ranking)

Selección de variables (SV)

	Ventajas	Desventajas
Uni-variable	Rapidez Escalabilidad Independencia del clasificador	Ignoran las dependencias entre las variables Ignoran la interacción con el clasificador
Multi-variable	Modelado de las dependencias entre las variables Independencia del clasificador Complejidad mejor que la de los wrappers	Más lentos que las técnicas uni-variable Menos escalabilidad que las técnicas uni-variable Ignoran la interacción con el clasificador

Algoritmos para selección de variables

- Tipos de filtros

Uni-variable	Multi-variable
A. Paramétricos t-test ANOVA Mutual information B. No paramétricos p-metric Man-Whitney Kruskal-Wallis BSS/WSS Permutation tests	Relief Correlation feature selection (CFS) Markov blanket

  - Paramétrico: se conoce la distribución subyacente de los datos
  - No paramétrico: no se conoce la distribución de los datos

Filtros: t-test

- Filtro uni-variable y paramétrico
- Numero de clases = 2
  - $N_1 + N_2 = N$
- Para cada variable i
  - Calcular la media de los valores de los ejemplos, μ^i
  - Calcular la media de los valores de los ejemplos de cada clase, μ_C^i
  - Calcular la desviación estándar de los valores de los ejemplos, σ^i
  - Calcular la desviación estándar de los valores de los ejemplos de cada clase, σ_C^i
  - Calcular p-valor
    - Varianzas iguales $t^i = \frac{\mu_1^i - \mu_2^i}{\sigma^i * \sqrt{\frac{1}{N_1} - \frac{1}{N_2}}}$
    - Varianzas no iguales $t^i = \frac{\mu_1^i - \mu_2^i}{\sqrt{\frac{\sigma_1^i}{N_1} - \frac{\sigma_2^i}{N_2}}}$
- Ordenar de forma descendiente las variables de acuerdo a su p-valor absoluto

Filtros: ANOVA

- Filtro uni-variable y paramétrico
- Numero de clases = M
  - $N_1 + \ldots + N_M = N$
- Descomposición de la varianza

$$\sum_{k=1}^{M}\sum_{j=1}^{N_k}\left(e_{j,k}^{i}-\mu^{i}\right)^2 = \sum_{k=1}^{M}\sum_{j=1}^{N_k}\left(e_{j,k}^{i}-\mu_k^{i}\right)^2 + \sum_{k=1}^{M} N_k * \left(\mu_k^{i}-\mu^{i}\right)^2$$

$$SS_{total} = SS_{within} + SS_{between}$$

  - $e_{j,k}^{i}$: valor del ejemplo j-ésimo (cuya clase es k) en la variable i
- Calcular el p-valor para cada variable i

$$t^{i} = \frac{(N-M) * SS_{between}}{(M-1) * SS_{within}}$$

- Ordenar de forma descendiente las variables de acuerdo a su p-valor

Filtros: Man-Whitney

- Filtro uni-variable y no paramétrico
- Numero de clases = 2
  - $N_1 + N_2 = N$
- Para cada variable i
  - Ordenar los valores de menor a mayor independientemente de su clase
  - Asignar rangos (menor valor = 1, …, mayor valor = N)
    - En caso de empate se reparten los rangos
  - Sumar los rangos correspondientes a cada clase: R_1 y R_2
  - Calcular

$$U_1 = N_1 * N_2 + \frac{N_1 * (N_1 + 1)}{2} - R_1$$

$$U_2 = N_1 * N_2 + \frac{N_2 * (N_2 + 1)}{2} - R_2$$

$$U^i = \bigvee (U_1, U_2)$$

- Ordenar de forma descendiente las variables de acuerdo a su valor U

Filtros: Kruskal-Wallis

- Filtro uni-variable y no paramétrico
- Numero de clases = M
  - $N_1 + \ldots + N_M = N$
- Para cada variable i
  - Ordenar los valores de menor a mayor independientemente de su clase
  - Asignar rangos (menor valor = 1, …, mayor valor = N)
    - En caso de empate se reparten los rangos
  - Sumar los rangos correspondientes a cada clase k, R_k
  - Calcular

$$W^i = \left(\frac{12}{N * (N+1)} * \sum_{k=i}^{M} \frac{R_k^2}{N_k} \right) - 3 * (N+1)$$

- Ordenar de forma descendiente las variables de acuerdo a su valor W

Filtros: Relief/ReliefF

- Filtro multi-variable
- Numero de clases = 2
  - $N_1 + N_2 = N$
- Inicializa un vector de pesos con valores 0
  - Tantos pesos como variables
- Se realizan N iteraciones
  - Elección de un ejemplo aleatoriamente, e
  - Se calcula el vecino más cercano de su clase, $NC(e)$, y de clase diferente, $NF(e)$
  - Para cada variable i se actualiza su peso

$$w(i) = w(i) + \frac{d\left(e_j^i - e_{NF(e_j)}^i\right)}{N} - \frac{\left(e_j^i - e_{NC(e_j)}^i\right)}{N}$$

- Salida: vector de pesos
- Idea: favorecer variables que tengan valores diferentes en ejemplos parecidos de diferente clase y valores iguales en ejemplos parecidos de la misma clase
- La fórmula para problemas multi-clase es más compleja (ReliefF)
  - Para cada clase se busca el vecino más cercano y se tiene en cuenta la probabilidad de ocurrencia de cada clase

Filtros: Correlation-based Feature Selection (CFS)

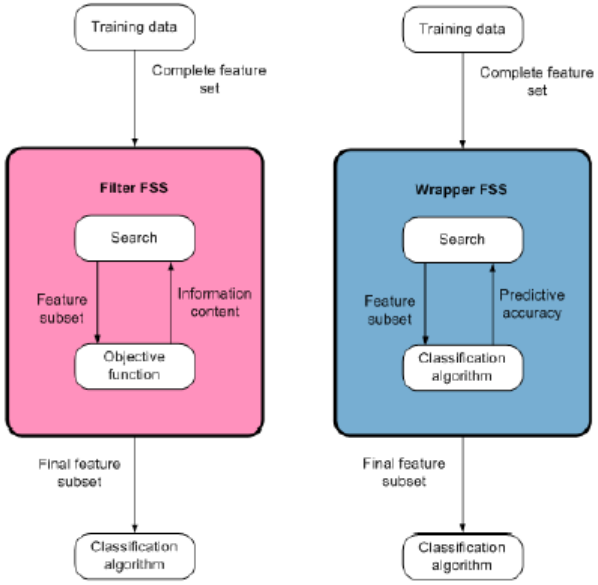
- Filtro multi-variable
- Evalúa subconjuntos de variables
- Idea: Seleccionar el subconjunto de variables S con correlación máxima con las clases y mínima correlación entre ellas

$$M(S) = \frac{n' * \overline{r_{cf}}}{\sqrt{n' + n' * (n' - 1) * \overline{r_{ff}}}}$$

  - n'= número de variables en S
  - $\overline{r_{cf}}$= correlación media entre las variables y las clases
  - $\overline{r_{ff}}$ = correlación media entre las variables
    - Para atributos discretos: correlación basada en la ganancia de información normalizada (Teoría de información)
    - Para atributos numéricos: correlación de Pearson
- Seleccionar el subconjunto de variables S que maximice $M(S)$
  - Las variables irrelevantes deberían ser ignoradas puesto que tienen poca correlación con las clases
  - Las variables redundantes deberían ser ignoradas puesto que tienen alta correlación con el resto de variables

Aproximaciones para acometer SV

- Técnicas de filtro, envoltorio (wrapper) y embebidas
  - Filtros: basadas en la características intrínsecas de los datos
  - WRAPPER: basadas en el conocimiento del clasificador
  - Embebidos: la selección se hace dentro del propio clasificador
- Requisito (filtros y wrappers): función de rendimiento que mida la calidad del conjunto de variables seleccionado

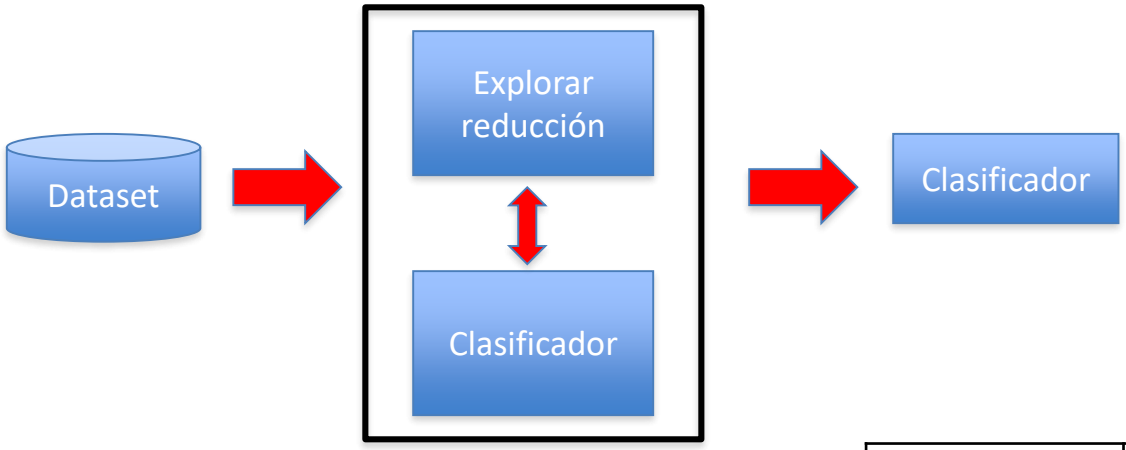


Evaluación: técnicas de envoltorio (wrapper)

- Los filtros tratan de estimar la calidad de la reducción
  - ¿Por qué no utilizar el clasificador a utilizar para determinar si la reducción es buena o mala?
- La evaluación del subconjunto de variables se obtiene mediante aprendizaje y evaluación de un clasificador
  - El clasificador utilizado para evaluar los subconjuntos se utiliza como caja negra
  - Está envuelto por el proceso de selección de variables
- Son técnicas multi-variable
- Se utilizan métodos heurísticos puesto que el número de subconjuntos posibles crece exponencialmente con el número de variables
- Mayor riesgo de sobre-entrenamiento
  - Uso de técnicas de validación para reducir el riesgo de sobre entrenamiento
- Técnicas computacionalmente costosas

Wrappers

- Esquema



- Ejemplo
  - Accuracy rate como medida de rendimiento
  - Obtener el conjunto mínimo de variables
    - Que maximice el accuracy

Subconjunto	Accuracy
$\{A_1,A_2,A_3\}$	98%
$\{A_1,A_2\}$	98%
$\{A_1,A_3\}$	77%
$\{A_2,A_3\}$	56%
$\{A_1\}$	89%
$\{A_2\}$	90%
$\{A_3\}$	91%
$\{\}$	85%

Salida: algoritmos Subconjunto de Atributos

- Devuelven un subconjunto de atributos optimizado según algún criterio de evaluación
  - Wrapper si el criterio es el rendimiento de un clasificador
- Algoritmo
  - Entrada
    - X: conjunto de N variables original, C: clasificador
  - Salida
    - X': subconjunto de variables
  - Pseudo-código
    - $Subconjunto = []$
    - Repetir
      - $S_k = GenerarSubconjunto(\mathrm{X})$
      - $Si\ hayMejoraRendimiento(Subconjunto, S_k, C)$
        - $Subconjunto = S_k$
    - Hasta *Criterio de parada*

Wrappers

- Componentes básicos
  - Estado inicial
  - Dirección de la búsqueda (pasos)
  - Criterio parada
- Evaluación del subconjunto de variables
  - Clasificador
  - Medida de rendimiento
- No hay una única opción mejor para cualquier componente
  - Probar varias y elegir la de mejor rendimiento

- NOTA: El mismo procedimiento se puede utilizar para filtros multi-variable
  - Solo cambiaría la evaluación del subconjunto de variables

Dirección de la búsqueda

- Hay 2^N posibles subconjuntos de las N variables iniciales
- Métodos de selección de variables heurísticos
  - Determinan las mejores variables, bajo la suposición de independencia
    - Utilizando test de significancia
    - Utilizando un clasificador
  - Introducción escalonada de la mejor variable
    - La mejor variable es elegida la primera
    - Después la siguiente mejor, ...
  - Eliminación escalonada de la peor variable
    - Se va eliminando la peor variable del subconjunto
  - Métodos combinados de eliminación y selección de la mejor variable
  - Métodos no deterministas
    - Aleatoriedad

Dirección de la búsqueda

- Sequential Forward Selection (SFS)
  - Empieza con un conjunto vacío de variables, S
  - Añade una variable (la mejor) a S de acuerdo algún criterio que mide la calidad de las variables
    - Elimina la variable añadida de las variables a examinar
  - El proceso de inserción se realiza hasta que se cumpla una condición de parada
    - Conjunto original de variables
    - Umbral para el número de variables a añadir
    - Generación de todos los posibles subconjuntos de variables

Dirección de la búsqueda

- Sequential Backward Selection (SBS)
  - Empieza con el conjunto total de variables, S
  - Elimina una variable a S (la peor) de acuerdo algún criterio que mide la calidad de las variables
    - Elimina la variable eliminada de las variables a examinar
  - El proceso de eliminación se realiza hasta que se cumpla una condición de parada
    - Se obtenga un subconjunto con una sola variable
    - Umbral para el número de variables a eliminar (mantener)
    - Generación de todos los posibles subconjuntos de variables

Técnicas de SFS Y SBS: Búsqueda Hill Climbing

- Método de búsqueda simple y rápido
  1. S = estado inicial
  2. CS = cambioLocal(S)
  3. Evaluar todas las soluciones en CS
  4. S' = mejor solución de CS
  5. Si calidad(S') > calidad(S)
     - S = S'
       - S = S + S' o S = S – S'
     - Ir al paso 2
  6. Else
     - Devolver S
- Estado inicial
  - Todas las variables
  - Ninguna variable
  - Subconjunto de variables
- Cambio local
  - Añadir cualquier posible variable
  - Eliminar cualquier posible variable

Dirección de la búsqueda: comentarios

- Tanto Sequential Forward Selection (SFS) como Sequential Backward Elimination (SBE)
  - Procedimientos de búsqueda simples: complejidad $O(N^2)$ (vs. $O(2^N)$)
  - Precio de la simplicidad
    - Pueden quedarse atrapados en óptimos locales
- SBE es más lento que SFS
  - Es más costoso aprender clasificadores con muchas variables que con pocas
    - Wrappers
- SFS tiende a seleccionar menos variables que SBE
- Intuitivamente, ¿cual preferirías?
  - Podría ser mejor mantener más variables puesto que el clasificador podría manejar mejor algunas irrelevancias y/o redundancias que quedarse atrapado en mínimos locales con pocas variables

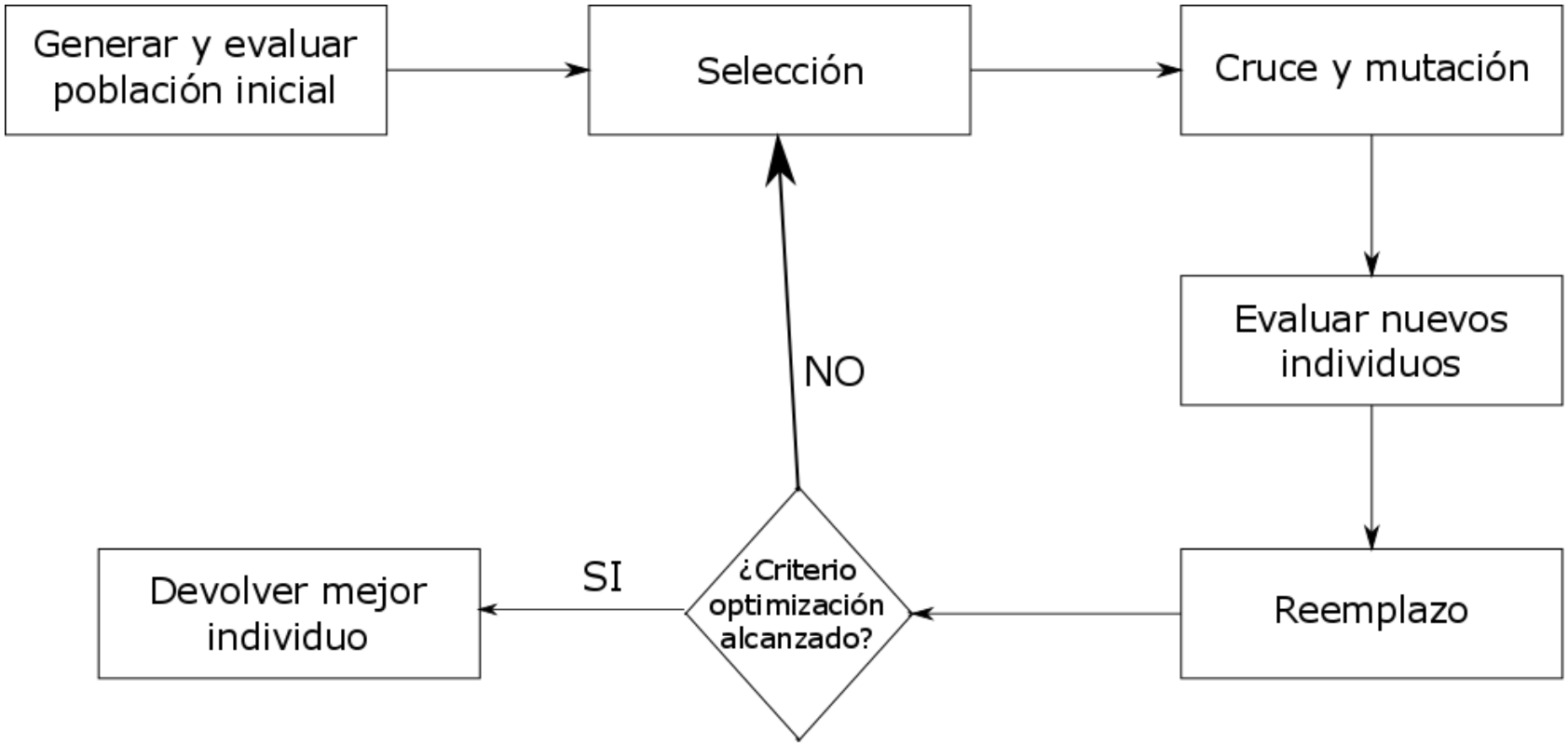
Dirección de la búsqueda

- Búsqueda no determinista (aleatoria) (BA)
  - Comienza la búsqueda en una dirección aleatoria
  - La elección de añadir o eliminar variables es aleatoria
  - Trata de evitar obtener subconjuntos sub-óptimos de variables
    - Evitando seguir una dirección de búsqueda única
  - A diferencia de SFS y SBS el tamaño del subconjunto no puede ser pre-determinado
- Se suelen utilizar técnicas como los algoritmos genéticos

Búsqueda: Algoritmos genéticos

- Algoritmos de optimización bio-inspirados
- Los algoritmos genéticos se rigen por un esquema iterativo en el que manipulan una población de soluciones candidatas al problema
- A través de las iteraciones, la población evoluciona
  - Se manipula por medio de un procedimiento inspirado por los principios de la selección natural y la genética
    - Selección
    - Cruce
    - Mutación
    - Reemplazo
- Este procedimiento no imita la vida
  - Simplemente está inspirado por su funcionamiento

Búsqueda: Algoritmos genéticos



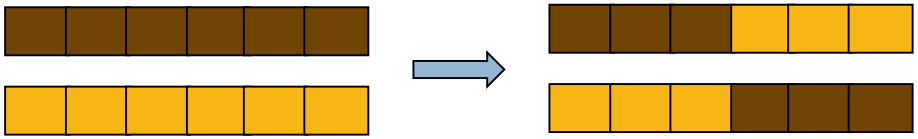
Búsqueda: Algoritmos genéticos - términos

- Individuo
  - Posible solución al problema: cadena de bits (0, 1)
    - Cada bit (gen) está asociado a una variable e indica si ésta es seleccionada o no
- Población
  - Conjunto de individuos
- Evaluación (fitness)
  - Consiste en asignar un valor que indique la calidad del individuo
    - Llamada al aprendizaje del clasificador con las variables asociadas a genes con valor 1
    - Evaluación del modelo aprendido con los ejemplos de entrenamiento
- Selección
  - Proceso que suele premiar a los mejores individuos
    - Suele estar basado en la evaluación de cada individuo
  - Los buenos individuos sobrevivirán y crearán más descendientes en la siguiente población
    - Los malos individuos tienden a desaparecer

Búsqueda: Algoritmos genéticos

- Cruce
  - Intercambio de partes de las soluciones
  - El cruce tomará dos individuos de la población (padres) y con una cierta probabilidad, Pc, generará dos descendientes

Cruce en 1 punto



- Mutación
  - Idea: introducir pequeñas variaciones (aleatorias) a los individuos
  - Para representación binaria: inversión de bits
    - Seleccionamos un gen del individuo y con una cierta probabilidad, Pm, se invierte su valor
      - Si el gen tenía valor 1 se cambia a 0 y viceversa
- Reemplazo
  - Determina como se combina la población inicial (de padres) con la de descendientes