Universidad Simón Bolívar
Decanato de Estudios Profesionales
Coordinación de Ingeniería de la Computación

@títuloProyecto

Por:
Alejandro Flores V.

Realizado con la asesoría de:
Emely Arráiz B.

PROYECTO DE GRADO
Presentado ante la Ilustre Universidad Simón Bolívar
como requisito parcial para optar al título de
Ingeniero de Computación

Sartenejas, septiembre de 2014

UNIVERSIDAD SIMÓN BOLÍVAR
DECANATO DE ESTUDIOS PROFESIONALES
COORDINACIÓN DE INGENIERÍA DE LA COMPUTACIÓN

ACTA FINAL PROYECTO DE GRADO

@tíTuloProyecto

Presentado por:
Alejandro Flores V.

Este Proyecto de Grado ha sido aprobado por el siguiente jurado examinador:

Emely Arráiz B.

@jurado1

@jurado2

Sartenejas, @día de @mes de @año

Resumen

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Palabras clave: @palabra1, @palabra2, @palabra3.

Agradecimientos

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Índice general

Índice de figuras

Índice de Tablas

Acrónimos y Símbolos

KDD	Knowledge Discovery in Databases
MD	Minería de Datos
SI	Selección de Instancias
NN	Nearest Neighbor

$\in$	Relación de pertenencia, «*es un elemento de*»

Dedicatoria

A @personasImportantes, por @razonesDedicatoria.

Introducción

El avance de la ciencia y la tecnología durante las últimas décadas ha traido como consecuencia un aumento sin precedentes en la candidad de datos generados y recopilados por la actividad humana. El *Proyecto Genoma Humano*, el *Instituto SETI* y el *Gran Colisionador de Hadrones*, tienen algo en común: generan una enorme cantidad de datos, por lo que resulta imposible usarlos y mucho menos analizarlos de forma tradicional.

Por esta razón, nuevos cambos de estudio, como el Descubrimiento de Conocimiento en Bases de Datos (KDD) y Minería de Datos (DM), emergen para afrontar el creciente problema que se genera al intentar usar y analizar enormes cantidades de datos.

Bajar complejidad, disminuir los datos.

Capítulo 1

Selección de Instancias

En este capítulo se describe el proceso de reducción de datos y sus diferentes estrategias. En particular, se hace especial énfasis en el problema de *Selección de Instancias*: se define formalmente, se describen sus principales características, y se realiza un breve análisis del estado del arte.

1.1. Reducción de Datos

Como parte del proceso de "*Knowledge Discovery in Databases*" (*KDD*), la fase de *Preprocesamiento de los Datos* juega un rol fundamental para la aplicación efectiva de técnicas de *Minería de Datos* (*MD*). Una de las estrategias de mayor uso durante la fase de preprocesamiento es la de *Reducción de Datos*.

El problema de *Reducción de Datos* consiste en decidir qué datos deben ser utilizados durante la aplicación de algoritmos de *MD* con el objetivo de construir modelos representativos de los datos originales. Dicha decisión debe basarse en la relevancia de los datos con respecto a los objetivos que se persiguen, o inclusive, por limitaciones técnicas. En términos prácticos, la importancia del problema de *Reducción de Datos* radica en los siguientes factores: *a*) *Tiempo y Espacio*: Mientras mayor sea el número de datos a utilizar, mayor será el espacio necesario para almacenarlos y el tiempo requerido para analizarlos. *b*) *Sensibilidad al ruido*: Al aumentar el número de instancias en el conjunto de datos, también lo hace

la probabilidad de aparición de datos atípicos, inconsistentes o redundantes. Su eliminación se vuelve necesaria para evitar un impacto negativo en los modelos de representación creados a partir de los datos.

En función de estos criterios, y basados en la definición de los datos, se han formulado diferentes estrategias para llevar a cabo la fase de reducción. En los procesos de *KDD*, el conjunto de datos está definido en función de un conjunto de clases Ω y un conjunto T de n observaciones de un evento, cada observación con m mediciones, donde:

Definición 1. Una **instancia** t_i (con $i = 1 \dots n$) es una observación del evento; donde $t_i = (v_{i,1}, v_{i,2}, \dots, v_{i,m})$ es una tupla de m valores/mediciones (un punto en un espacio m-dimensional). Adicionalmente, cada instancia en t_i pertenece a la *clase* $\omega_{t_i} \in \Omega$.

Definición 2. Un **atributo** p_j (con $j = 1 \dots m$) define el conjunto de mediciones «*de un mismo tipo*» para todas las observaciones, *i.e.* $p_j = \{v_{i,j} \mid i = 1 \dots n\}$. Cada atributo puede presentarse en diferentes formatos: *nominales*, *discretos*, o *continuos*.

A continuación se presentan las estrategias de *Reducción de Datos* más estudiadas en la literatura:

- **Selección de Instancias** [BL97, LM02]
  Busca la reducción del conjunto de datos mediante la selección de un subconjunto de instancias, de forma tal que dicho subconjunto conserve las capacidades de representación del conjunto original.
  La sección 1.2 está dedicada a describir esta estrategia en amplitud.

- **Selección de Atributos** [BL97, LM98]
  Esta técnica permite eliminar atributos del conjunto de datos original, que no contribuyen (o que influyen negativamente) a la construcción de un modelo representativo.

- **Discretización de Atributos** [FI93, LHTD02]
  Esta estrategia busca convertir atributos *continuos* en *discretos* (cuantificando el espacio de posibles valores), o disminuir el número de valores *discretos* (combinando valores adyacentes).

1.2. Selección de Instancias

Dado un conjunto inicial de instancias $T = \{t_i \mid i = 1 \dots n\}$ donde $t_i = (v_{i,1}, v_{i,2}, \dots, v_{i,m})$ y $\omega_i \in \Omega$ (siendo Ω el conjunto de posibles clases para las instancias en T), el problema de *Selección de Instancias* (*SI*) consiste en seleccionar un $R \subseteq T$ que mantenga (o mejore) la capacidad de representación del conjunto original T.

Este problema puede ser reformulado como un *problema de optimización*, donde se busca el $R^* \subseteq T$ de menor cardinalidad, que mantenga (o mejore) la capacidad de representación del conjunto original.

En particular, la literatura se ha enfocado en la aplicación del problema de *SI* para su uso en clasificadores [GK14, Tou02]. El subconjunto seleccionado se usa como conjunto de entrenamiento, en base al cuál el clasificador estima la clase $\hat{\omega}$ de instancias previemente desconocidas. En este sentido, el problema de optimización de *SI* busca conseguir un $R^* \subseteq T$ *consistente* y de cardinalidad mínima, donde:

Definición 3. Un conjunto R es **consistente** con T, *si y solo si* toda instancia $t \in T$ es clasificada correctamente (*e.i.* $\hat{\omega}_t = \omega_t$) mediante el uso de un clasificador M y las instancias en R como conjunto de entrenamiento.

Este caso particular del problema de *SI* también es conocido como *Selección de Prototipos* (*SP*), dado que se encarga de seleccionar un conjunto de instancias que sirvan como prototipos para un clasificador dado.

La complejidad del problema de selección ha sido estudiada por diferentes autores: *Bien* y *Tibshirani* [BT12] describen la reducción del problema de *SI* al problema de *Conjunto de Cobertura* ("*Set Cover*" en inglés), cuya versión de optimización es NP-Dura. Adicionalmente, *Wilfong* [Wil91] y *Zukhba* [Zuk10] muestran que el problema de *SP* es NP-Completo.

En general, la literatura relacionada con el problema de *SI* se ha enfocado en el uso de clasificadores k-NN por su simplicidad, y sobretodo, por su capacidad de representación de modelos sin información adicional sobre la distribución de los datos. A continuación se describen los clasificadores NN.

1.2.1. Vecino Más Cercano (NN)

Inicialmente descrita por *Fix* y *Hodges* [FH51], la regla del *Vecino Más Cercano* ("*Nearest Neighbor*", *NN*) es una regla de inferencia que estima la clase $\hat{\omega}_x$ de un punto x en un espacio m-dimensional. Dado un conjunto T de instancias de entrenamiento y una función de distancia φ entre dos puntos (en el espacio de m dimensiones):

$$\hat{\omega}_x = \omega_{t^*} \ , \ t^* = \underset{t \in T}{\arg\min}\, \varphi(t, x) \tag{1.1}$$

La generalización de la regla de inferencia NN se conoce como el clasificador k-NN: dado un $k \in \mathbb{N}$, se estima la clase $\hat{\omega}_x$ de un punto x en función a la clase de las k instancias más cercanas a x. En general, se usa la estrategia del «*voto de la mayoría*», asignando la clase más común entre las k instancias más cercanas. En particular, el clasificador 1-NN corresponde a la regla NN.

k-NN es un clasificador no paramétrico, de *aprendizaje perezoso* (debido a que la etapa de aprendizaje consiste en guardar el conjunto de entrenamiento), caracterizado por su sencillez en términos de implementación. Esa simplicidad, y su probada utilidad para numerosas aplicaciones, han hecho del clasificador k-NN uno de los más estudiados en la literatura.

Uno de los trabajos de mayor relevancia es el de *Cover* y *Hart* [CH67], quienes mostraron que cuando el número de instancias de entrenamiento tiende a infinito, el clasificador k-NN garantiza un error no mayor al doble de la tasa de error de Bayes: la menor tasa de error posible para un clasificador dado. Adicionalmente, probaron que para un conjunto de entrenamiento de cardinalidad finita, el clasificador 1-NN es admisible dentro de la clase de clasificadores k-NN: *e.i.* No existe $k > 1$ tal que k-NN tenga menor probabilidad de error frente a 1-NN, para toda posible distribución de los datos.

Adicionalmente algunos trabajos en geometría computacional han contribuido significativamente en la comprensión del problema. En este sentido, la regla NN para espacios euclideanos puede definirse de forma alternativa en función de

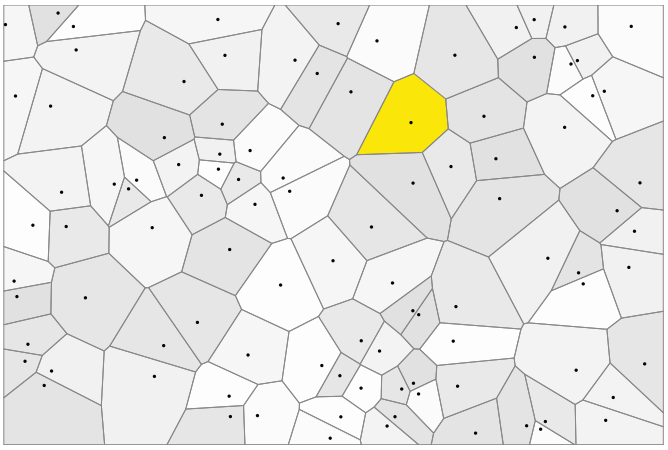

FIGURA 1.1: Diagrama de Voronoi para instancias en un espacio $\mathbb{R}^2$. En amarillo la Celda de Voronoi de un punto $t \in T$, representando el espacio de puntos para los que t es su *vecino más cercano*.

Diagramas de Voronoi [Vor08]; donde el espacio m-dimensional se encuentra particionado en *Celdas de Voronoi*, cada una definida por una instancia $t \in T$ donde t es el *vecino más cercano* para todos los puntos dentro del espacio dentro de dicha celda (ver Figura 1.1). Esto ha permitido el desarrollo de nuevos enfoques para la búsqueda de vecinos más cercanos basados en *Diagramas de Voronoi*, como el descrito por *Kolahdouzan* y *Shahabi* [KS04].

Similarmente, esta relación ha permitido avances importantes en términos de complejidad. En particular, mediante el uso de *kd*-trees [Ben75] (árboles de búsqueda binaria en múltiples dimensiones) se ha logrado disminuir la complejidad en tiempo de clasificación, de $\mathcal{O}(n)$ (de un enfoque "ingenuo" revisando todas las instancias) a $\mathcal{O}(\log n)$, a costas de un aumento en el tiempo necesario para el entrenamiento del clasificador: de $\mathcal{O}(1)$ a $\mathcal{O}(n \log n)$.

Sin embargo, los clasificadores k-NN presentan ciertas propiedades desalentadoras; el problema de conseguir el vecino más cercano de un punto dado, requiere –en cualquiera de los casos– almacenar todas las instancias de entrenamiento: *e.i.* $\mathcal{O}(n)$ en espacio. Adicionalmente, trabajos más recientes [KL04] muestran que en espacios euclideanos de altas dimensiones la búsqueda del vecino más cercano requiere $\mathcal{O}(n)$ en tiempo: un fenómeno conocido como la «*maldición de la dimensionalidad*» ("*curse of dimensionality*" en inglés). Finalmente, según *Shwartz* y *David* [SSBD14] los clasificadores NN tienden a sobreajustar el modelo con respecto al conjunto de entrenamiento (*overfitting* en inglés); efecto que puede mitigarse aumentando el k del clasificador [DGKL94, SSBD14], y eliminando instancias del

conjunto de datos [GKK13].

1.3. Algorítmos de aproximación para SI

Debido a la complejidad del problema de *SI*, la literatura se ha enfocado en la definición de heurísticas para conseguir soluciones aproximadas. De nuevo, el uso de clasificadores k-NN es una práctica extendida a lo largo de estos trabajos, por lo que la mayoria de estas estrategias de reducción se basan en conceptos base de este tipo de clasificadores.

En este sentido, se definen dos términos recurrentes en las descripciones de diferentes algoritmos de aproximación para *SI*. Dado un conjunto de instancias Q cualquiera:

Definición 4. Los **asociados** en Q de una instancia $q \in Q$ son aquellas instancias en Q que pertenecen al conjunto de k instancias más cercanas a q:

$$asociados_Q(q) = \{q' \in Q \mid q \in kNN(q')\} \tag{1.2}$$

Definición 5. Los **enemigos** en Q de una instancia $q \in Q$ son aquellas instancias en Q con una *clase* diferente a la *clase* de q:

$$enemigos_Q(q) = \{q' \in Q \mid \omega_{q'} \neq \omega_q\} \tag{1.3}$$

A continuación se describen algunos de los métodos de selección de instancias más estudiados en la literatura:

- *Condensed Nearest Neighbor* (CNN) [Har68]
  Inicialmente el conjunto R se inicializa con una instancia cualquiera. Luego se itera sobre cada instancia $t \in T$; si t no es clasificada correctamente usando R, t se agrega a R. CNN reduce considerablemente el conjunto de datos, pero no asegura un conjunto consistente ni mínimo, pues depende del orden en el que son revisadas las instancias en T.

- *Edited Nearest Neighbor* (ENN) [Wil72]
  Comienza con $R = T$. Luego itera sobre las instancias en R; aquellas que no sean bien clasificadas usando R son eliminadas.

- *Repeated Edited Nearest Neighbor* (RENN) [Wil72]
  Aplica ENN al conjunto de datos R (inicialmente $R = T$) hasta que no ocurran cambios en R.

- *Reduced Nearest Neighbor* (RNN) [Gat72]
  RNN extiende a CNN, usandola como solución inicial $R = R_{CNN}$. Luego, itera sobre cada instancia $t \in R$: si todas las instancias en T son correctamente clasificadas usando $R \setminus \{t\}$, se elimina t de R. En caso contrario, se mantiene R y continua la iteración. La precisión de RNN puede mejorar respecto a CNN, pero es más costoso y su consistencia depende de la consistencia del conjunto resultante de CNN y del orden en que se iteren las instancias en R.

- *Decremental Reduction Optimization Procedure 1* (DROP1) [WM97]
  Comienza con una solución inicial $R = T$. Itera sobre cada instancia $t \in R$: si todos sus *asociados* en R son correctamente clasificados con $R \setminus \{t\}$, t se elimina de R. Reduce considerablemente el conjunto de datos inicial, pero obtiene baja precisión de clasificación, y el subconjunto resultante depende del orden en que se iteró sobre T.

- *Decremental Reduction Optimization Procedure 2* (DROP2) [WM97]
  Es una mejora sobre DROP1 en la cuál se elimina una instancia t cuando todos sus *asociados* en T son clasificadas correctamente usando $R \setminus \{t\}$. Además, DROP2 ordena las instancias con respecto a la distancia de su *enemigo* más cercano, en un intento de eliminar primero instancias centrales, y luego los puntos en los bordes de decisión.

- *Decremental Reduction Optimization Procedure 3* (DROP3) [WM97]
  Dado que el orden en que se iteran las instancias en DROP2 se ve alterado por puntos ruidosos, DROP3 filtra instancias ruidosas antes de ordenar el conjunto de entrenamiento.

1.3.1. Taxonomía

Debido a los numerosos enfoques existentes para aproximar soluciones al problema de *SI*, el trabajo de *García et al.* [GDCH12] describe un esquema taxonómico para caracterizar las diferentes estrategias que se han desarrollado en base al uso de clasificadores k-NN. Dicho esquema clasifica las heurísticas de acuerdo al tipo de selección, y al tipo de evaluación y dirección de la búsqueda (ver Figura 1.2).

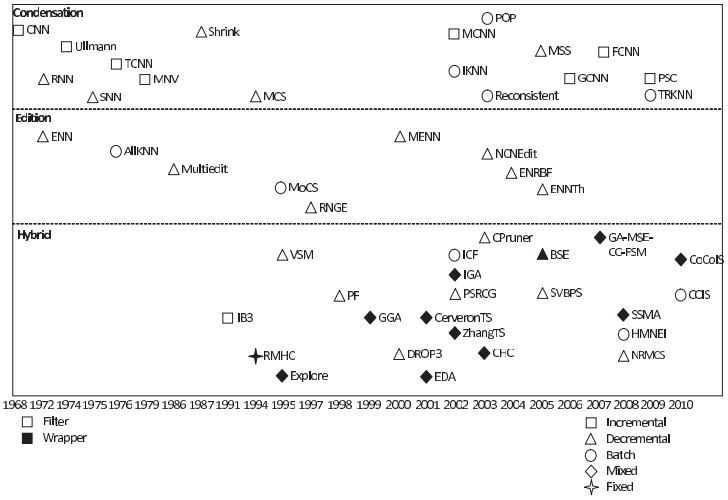

FIGURA 1.2: Taxonomía de algoritmos de aproximación para el problema de *SI* [GDCH12].

Según la dirección de la búsqueda

La dirección en la cuál procede la búsqueda de posibles subconjuntos R del conjunto inicial T puede darse de diferentes formas.

- *Incremental*: Comienzan con un conjunto R vacio, o con unas pocas instancias representativas de cada clase en el conjunto de datos. En cada iteración se añaden nuevas instancias a R. Estos métodos tienden a ser más rápidos, pero dependen del orden de iteración sobre las instancias, y pueden producir un elevado *overfitting*.

- *Decremental*: Inicialmente $R = T$, y en cada iteración se eliminan elementos de R. Tienen la ventaja de contar inicialmente con todo el conjunto de datos para poder tomar decisiones, aunque son más costosos y dependen del orden de revisión de las instancias.

- *Batch*: Estos métodos identifican aquellas instancias que cumplen cierto criterio, para luego eliminarlas/añadirlas todas como un conjunto. Presentan mayor complejidad.
- *Mixed*: Comienzan con una selección aleatoria o con la selección dada por otro método de *SI*, para posteriormente añadir o eliminar instancias. Generalmente presentan *overfitting*, además incrementar el tiempo de cómputo.
- *Fixed*: Se refiere a una subcategoría de las heurísticas *Mixed*, en la cuál se añaden y eliminan el mismo número de instancias; lo cuál no modifica el tamaño de la solución inicial.

Según el tipo de selección

Varian en los tipos de instancias que seleccionan: puntos borde, centrales o cualesquiera.

- *Condensation*: Seleccionan instancias cercanas a los bordes de decisión, también llamados puntos borde. Presentan mucha sencibilidad ante instancias ruidosas. En general estos métodos tienden a preservar la precisión para el conjunto de entrenamiento, pero afectan negativamente la generalización.
- *Edition*: Buscan eliminar puntos borde para mantener bordes de decisión más "suaves", por lo que presentan menor sencibilidad ante puntos ruidosos. Tienden a mejorar la generalización del conjunto T pero presentan un bajo porcentaje de reducción.
- *Hybrid*: Permiten la selección de puntos borde y centrales con el objetivo de conseguir el menor conjunto R que mantenga o aumente la precisión general del clasificador.

Según la evaluación de la búsqueda

Diferentes estrategias para la evaluación de soluciones intermedias.

- *Wrapper*: Utilizan el conjunto de datos completo sobre el clasificador k-NN para la evaluación de soluciones intermedias, aplicando el esquema de validación *leave-one-out*.

- *Filter*: Usan solo partes del conjunto de datos original para la evaluación de soluciones intermedias, y sin aplicar el esquema de validación *leave-one-out*. Implica menor tiempo evaluación, a costas de menor precisión.

1.3.2. Criterios de comparación

Para comparar métodos de *SI* se consideran una serie de criterios que pueden ser usados para evaluar las ventajas y desventajas de cada algoritmo. A continuación se describen los factores más relevantes:

- *Reducción*: El objetivo principal de métodos de *SI* es el de reducir número de instancias del conjunto de datos. Esto no solo disminuye el espacio necesario para almacenar los datos, sino que acelera el proceso de clasificación.
- *Precisión*: Un algoritmo exitoso debe reducir el conjunto de datos, afectando en la menor medida posible su capacidad de generalización.
- *Tiempo*: A pesar de que el proceso preprocesamiento y aprendizaje debe realizarse solo una vez, la complejidad de los algoritmos pueden volverlos poco prácticos para su uso sobre conjuntos de datos "grandes".
- *Tolerancia al ruido*: Algoritmos que seleccionen consistentemente instancias atípicas o inconsistentes ("ruido") presentan menor capacidad de generalización de los datos.

Capítulo 2

Algoritmos Evolutivos

Los algoritmos evolutivos (*EA*) son métodos estocásticos de búsqueda sobre espacios combinatorios, basados en el comportamiento evolutivo de las poblaciones. Comúnmente son aplicados en problemas con espacios de búsqueda poco conocidos, dado que permite explorar en amplitud el espacio de soluciones, sin olvidar la explotación de soluciones prometedoras.

Este es el caso del problema de *selección de instancias* (*IS*), donde un conjunto inicial de datos pequeño, con 100 instancias, tiene un espacio de 2^{100} soluciones factibles, y donde no tenemos información adicional sobre la distribución de los datos.

De los diferentes algoritmos evolutivos existentes, los autores propusieron la evaluación de los siguientes cuatro (4) modelos:

2.1. Generational Genetic Algorithm (GGA)

Se mantiene una *población* de *cromosomas* (conjunto de posibles soluciones), que evolucionan durante un número de *generaciones* (iteraciones).

El proceso de evolución consiste en: *i*) selección del conjunto de individuos con mayor *fitness* de la población, *ii*) proceso de recombinación entre pares de

individuos/cromosomas (llamados padres) usando operadores de cruce y mutación, que generan un par de nuevas soluciones (llamadas *descendencia*).

2.2. Steady-State Genetic Algorithm (SGA)

Utiliza un proceso similar a *GGA*. La diferencia radica en que en cada iteración/generación se producen solo 1 o 2 individuos nuevos.

En cada iteración se *i*) seleccionan dos individuos padres de la población actual, *ii*) se crea su descendencia mediante operadores de cruce y mutación, *iii*) se seleccionan el/los individos a reemplazar de la población siguiendo alguna estrategia de selección (menor *fitness*, más "viejo", aleatorio), y *iv*) se decide si se reemplazan dichos individuos con la nueva descendencia o no (reemplazo incondicional, o dependiente del *fitness* de los individuos).

2.3. CHC Adaptive Search Algorithm

Similarmente, este algorítmo mantiene una población de N cromosomas, donde en cada iteración: *i*) de la población de N individuos padres se genera una descendencia de N individuos, *ii*) de ambas poblaciones sobreviven los mejores N individuos para la siguiente generación.

CHC tiene otras dos particularidades. Por un lado, implementa un operador de recombinación llamado HUX, que intercambia la mitad de los bits que difieren entre los dos individos de forma aleatoria. Además *CHC* emplea la prevención de "incesto": antes de realizar el cruce usando HUX, calcula la *distancia de Hamming* entre ambos individuos padres; si dicha distancia es mayor a cierto umbral (inicialmente $L/4$, donde L es la longitud de los cromosomas), se realiza el cruce. En caso de no haberse generado ninguna descendencia durante una iteración, dicho umbral se disminuye en 1.

Nótese que durante este proceso no se aplica el operador de mutación. Cuando el umbral de prevención de incesto llega a cero, significa que la población convergió,

y se comienza un proceso de repoblación en el que se usa la mejor solución/cromosoma encontrado hasta el momento, modificando hasta 35 % de sus bits para generar los $N-1$ individuos restantes de la nueva población, y luego continuar el proceso evolutivo.

2.4. Population-Based Incremental Learning (PBIL)

Esta metaheurística consiste en mantener un vector de probabilidades V_p de tamaño L (número de instancias iniciales), donde $V_p[i]$ es la probabilidad de que la i-esima instancia pertenezca a la solución (i-esimo bit sea 1).
Inicialmente $V_p[i] = 0{,}5 \quad \forall i \in [1 \dots L]$. En cada iteración se sigue el siguiente proceso:

a) Se generan N cromosomas (secuencias de bits) basados en las probabilidades en V_p.

b) Se acerca V_p hacia la mejor solución generada S_{best}

$$V_p[i] = V_p[i] * (1 - LR) + S_{best}[i] * LR$$

Donde LR es la tasa de aprendizaje (*learning rate*).

c) Se aleja V_p de la peor solución generada S_{worst}
Si $S_{best}[i] <> S_{worst}[i]$

$$V_p[i] = V_p[i] * (1 - Negat\_LR) + S_{best}[i] * Negat\_LR$$

Donde $Negat\_LR$ es la tasa de aprendizaje negativa.

2.5. Particle Swarm Optimization (PSO)

PSO se inspira en el comportamiento de organismos biológicos, en particular, del vuelo de una bandada de aves. Cada ave o "partícula" (que representa una posible solución del espacio de búsqueda) tiene una velocidad asociada, y modifica

su vuelo en relación a su propia experiencia, y a la experiencia de sus "compañeras". Diferentes estudios muestran que *PSO* obtiene mejores resultados que los algoritmos genéticos (*GA*), y en menor tiempo de cómputo.

Inicialmente se obtienen P soluciones aleatorias, o partículas. Cada partícula i está representada por un posición en un espacio s-dimensional $\mathbf{x}_i = < x_{i1}, x_{i2}, \ldots, x_{is} >$. Luego, se realizan un número de iteraciones (`MAX_ITER`) en las que se actualiza la posición de cada partícula de acuerdo a su velocidad v_i:

$$\mathbf{x}_i = \mathbf{x}_i + v_i$$

$$v_i = wv_i + c_1 Rand()(p_i - \mathbf{x}_i) + c_2 Rand()(p_g - \mathbf{x}_i)$$

Donde c_1 y c_2 con constantes, $Rand()$ es una función aleatoria $[0, 1]$, p_i es la mejor solución encontrada por la partícula i (de acuerdo a una función de evaluación/*fitness* establecida), p_g es la mejor solución global, y w es el "peso de inercia" que establece la posible variabilidad de v_i. w disminuye cada iteración de acuerdo a la siguiente fórmula:

$$w = \frac{(w_{start} - w_{end})(\texttt{MAX\_ITER} - \texttt{Iter})}{\texttt{MAX\_ITER} + w_{end}}$$

Siendo `Iter` la iteración actual del algoritmo, y w_{start} y w_{end} valores predeterminados.

Capítulo 3

Punto de partida

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis

sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Capítulo 4

Evaluación Experimental

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis

sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Conclusiones y Recomendaciones

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet,

consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

1. First itemtext
2. Second itemtext
3. Last itemtext
4. First itemtext
5. Second itemtext

Bibliografía

[Ben75] Jon Louis Bentley. Multidimensional binary search trees used for associative searching. *Commun. ACM*, 18(9):509–517, September 1975.

[BL97] Avrim Blum and Pat Langley. Selection of relevant features and examples in machine learning. *Artif. Intell.*, 97(1-2):245–271, 1997.

[BT12] J. Bien and R. Tibshirani. Prototype selection for interpretable classification. *ArXiv e-prints*, February 2012.

[CH67] T. Cover and P. Hart. Nearest neighbor pattern classification. *IEEE Trans. Inf. Theor.*, 13(1):21–27, January 1967.

[DGKL94] Luc Devroye, Laszlo Gyorfi, Adam Krzyzak, and Gábor Lugosi. On the strong universal consistency of nearest neighbor regression function estimates. *The Annals of Statistics*, pages 1371–1385, 1994.

[FH51] E. Fix and J. L. Hodges. Discriminatory analysis, nonparametric discrimination: Consistency properties. *US Air Force School of Aviation Medicine*, Technical Report 4(3):477+, January 1951.

[FI93] Usama M. Fayyad and Keki B. Irani. Multi-interval discretization of continuous-valued attributes for classification learning. In Ruzena Bajcsy, editor, *IJCAI*, pages 1022–1029. Morgan Kaufmann, 1993.

[Gat72] Geoffrey W. Gates. The reduced nearest neighbor rule (corresp.). *IEEE Transactions on Information Theory*, 18(3):431–433, 1972.

[GDCH12] Salvador Garcia, Joaquin Derrac, Jose Cano, and Francisco Herrera. Prototype selection for nearest neighbor classification: Taxonomy and empirical study. *IEEE Trans. Pattern Anal. Mach. Intell.*, 34(3):417–435, March 2012.

[GK14] Lee-Ad Gottlieb and Aryeh Kontorovich. Near-optimal sample compression for nearest neighbors. *CoRR*, abs/1404.3368, 2014.

[GKK13] Lee-Ad Gottlieb, Aryeh Kontorovich, and Robert Krauthgamer. Efficient classification for metric data. *CoRR*, abs/1306.2547, 2013.

[Har68] P. Hart. The condensed nearest neighbor rule (corresp.). *IEEE Trans. Inf. Theor.*, 14(3):515–516, September 1968.

[KL04] Robert Krauthgamer and James R. Lee. Navigating nets: simple algorithms for proximity search. In J. Ian Munro, editor, *SODA*, pages 798–807. SIAM, 2004.

[KS04] Mohammad Kolahdouzan and Cyrus Shahabi. Voronoi-based k nearest neighbor search for spatial network databases. In *Proceedings of the Thirtieth International Conference on Very Large Data Bases - Volume 30*, VLDB '04, pages 840–851. VLDB Endowment, 2004.

[LHTD02] Huan Liu, Farhad Hussain, Chew Lim Tan, and Manoranjan Dash. Discretization: An enabling technique. *Data Min. Knowl. Discov.*, 6(4):393–423, October 2002.

[LM98] Huan Liu and Hiroshi Motoda. *Feature Extraction, Construction and Selection: A Data Mining Perspective*. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[LM02] Huan Liu and Hiroshi Motoda. On issues of instance selection. *Data Min. Knowl. Discov.*, 6(2):115–130, April 2002.

[SSBD14] S. Shalev-Shwartz and S. Ben-David. *Understanding Machine Learning: From Theory to Algorithms*. Cambridge University Press, 2014.

[Tou02] Godfried T. Toussaint. Open problems in geometric methods for instance-based learning. In Jin Akiyama and Mikio Kano, editors, *JCDCG*, volume 2866 of *Lecture Notes in Computer Science*, pages 273–283. Springer, 2002.

[Vor08] Georges Voronoï. Nouvelles applications des paramètres continus à la théorie des formes quadratiques. deuxième mémoire. recherches sur les parallélloèdres primitifs. *Journal für die reine und angewandte Mathematik*, 134:198–287, 1908.

[Wil72] DR Wilson. Asymptotic properties of nearest neighbor rules using edited data. *Institute of Electrical and Electronic Engineers Transactions on Systems, Man and Cybernetics*, 2:408–421, 1972.

[Wil91] Gordon Wilfong. Nearest neighbor problems. In *Proceedings of the Seventh Annual Symposium on Computational Geometry*, SCG '91, pages 224–233, New York, NY, USA, 1991. ACM.

[WM97] D. Randall Wilson and Tony R. Martinez. Instance pruning techniques. In *Proceedings of the Fourteenth International Conference on Machine Learning*, ICML '97, pages 403–411, San Francisco, CA, USA, 1997. Morgan Kaufmann Publishers Inc.

[Zuk10] A. V. Zukhba. Np-completeness of the problem of prototype selection in the nearest neighbor method. *Pattern Recognit. Image Anal.*, 20(4):484–494, December 2010.