UNIVERSIDAD CENTRAL MARTA ABREU DE LAS VILLAS
FACULTAD DE MATEMÁTICA - FÍSICA - COMPUTACIÓN
LICENCIATURA EN CIENCIAS DE LA COMPUTACIÓN

TRABAJO DE DIPLOMA

Algoritmos supervisados para la detección de ortólogos con manejo del desbalance

Autor: David Pérez García

Tutores: Msc. Deborah Galpert Cañizares

Lic. Reinier Millo Sánchez

Santa Clara

2013

"Año 55 de la Revolución"

Hago constar que el presente trabajo de diploma fue realizado en la Universidad Central "Marta Abreu" de Las Villas como parte de la culminación de estudios de la especialidad de Licenciatura en Ciencia de la Computación, autorizando a que el mismo sea utilizado por la Institución, para los fines que estime conveniente, tanto de forma parcial como total y que además no podrá ser presentado en eventos, ni publicados sin autorización de la Universidad.

Firma del Autor

Los abajo firmantes certificamos que el presente trabajo ha sido realizado según acuerdo de la dirección de nuestro centro y el mismo cumple con los requisitos que debe tener un trabajo de esta envergadura referido a la temática señalada.

Firma del Autor

Firma del Jefe de Departamento
donde se defiende el trabajo

Firma del Responsable de
Información Científico-Técnica

“El futuro pertenece a aquellos que creen en la belleza de sus sueños”

Eleanor Roosvelt

A mi querida familia

que tanto se lo merece.

A mis padres, a mis abuelos, a mi hermana, a Delfín, a Ida, a Julio, a toda mi familia por brindarme su apoyo y comprensión en todo momento.

A mis tutores Deborah Galpert y Reinier Millo por su ayuda y dedicación en el desarrollo de este trabajo.

A mis hermanos de la Universidad, quienes me acompañaron a lo largo de mi carrera, en especial a Raúl, Sergio, Oscar, el Make, el Piti. Mario, Carlos, Yaumara y Danay.

A mis amigos de toda la vida, Carlitos, Raidel, Rogelio, Jassiel, Manero, Pallí.

Al profesor Ramiro Vázquez por enseñarme la base de todo lo que aprendí en la carrera, y a Daniel Gálvez por sacarnos de tantos apuros en los momentos más difíciles.

A todos los profesores que han tenido que ver de una forma u otra en mi formación como profesional.

A Villa Clara por ser campeón este año.

A todos los que de un modo u otro han contribuido en el desarrollo de este trabajo.

RESUMEN

La presente tesis incluye dos enfoques para manejar el desbalance en la clasificación binaria de pares de genes ortólogos vista como problema supervisado. El estudio involucra los genomas de Saccharomyces Cerevisiae y Schizosaccharomyces Pombe con las clasificaciones de INPARANOID7.0 y GENEDB. Se conforman diferentes conjuntos de rasgos a partir de diferentes valores de los parámetros de alineamiento global. Los rasgos son: la puntuación del alineamiento local y global de proteínas, la comparación de la longitud, la pertenencia a bloques localmente colineales y la comparación de los perfiles físico-químicos de las proteínas.

En el primer enfoque, un clasificador de Regresión logística basado en mezcla genera un número predefinido de conjuntos de datos balanceados manteniendo la clase minoritaria y reemplazando con repetición en la clase mayoritaria. Los modelos seleccionados son evaluados en el conjunto de prueba y mezclados con el promedio o con el voto mayoritario. En el segundo enfoque, se propone un clasificador sensitivo al costo de bosque aleatorio “Random Forest” que considera la proporción de casos en la matriz de costo y utiliza un filtro de distribución supervisado o un método de reducción basado en los conjuntos aproximados.

Palabras claves: detección de ortólogos, medidas de similitud, clasificador basado en mezcla, clasificador sensitivo al costo, problema de desbalance, teoría de conjuntos aproximados

ABSTRACT

This thesis paper presents two approaches to manage imbalance in binary gene pair ortholog classification as a supervised problem. The study involves Saccharomyces Cerevisiae and Schizosaccharomyces Pombe genomes with INPARANOID7.0 and GENEDB classifications. From different alignment parameters we built feature sets including the score of local and global protein alignments, the sequence length comparison, the membership to locally collinear blocks and the comparison of physico-chemical protein profiles.

In the first approach, a logistic regression ensemble-based classifier randomly generates a predefined number of balanced datasets keeping the minority class and replacing with repetition in the majority class. Selected regression models are evaluated in the test dataset and merged with the average or the majority vote. In the second approach, a cost-sensitive Random Forest considers the proportion of cases in the cost matrix and uses a supervised spread subsample or a rough set reduction method.

Keywords: ortholog detection, similarity measures, ensemble-based classifier, cost-sensitive classifier, imbalance problem, rough set theory

TABLA DE CONTENIDOS

Tabla de Contenidos

TABLA DE FIGURAS

INTRODUCCIÓN

Durante la evolución de los genomas se producen cambios genéticos que pueden ser las mutaciones de bases de nucleótidos o cambios entre segmentos como pueden ser la duplicación, transferencia horizontal, inversión y transposición. En el proceso evolutivo también se conservan regiones genómicas y funciones de las proteínas. La comparación de genomas como rama de la Bioinformática ayuda al descubrimiento de funciones de proteínas en genomas desconocidos. Fundamentalmente, el estudio de la homología entre las secuencias resulta vital en el descubrimiento de conocimiento sobre especies poco estudiadas.

La homología de secuencias se refiere a las secuencias de dos o más proteínas que son similares entre sí, debido a que presentan un mismo origen evolutivo. La homología no es un criterio medible, las secuencias son homólogas o no lo son, aunque usualmente se asume que dos secuencias son homólogas si estas dos presentan un alto grado de similitud (Webber and Chris, 2004). Un alto grado de similitud entre dos secuencias puede estar dado simplemente al azar, como sucede en ocasiones con secuencias de poco tamaño (Mount, 2004).

Dentro de la homología de secuencias se distinguen dos tipos de homología: la ortología y la paralogía. Los genes ortólogos son genes homólogos de especies diferentes que evolucionaron a partir de un ancestro común en un proceso de especiación, mientras que los parálogos son resultado de la duplicación. Los genes ortólogos proveen información útil en estudios de taxonomía, estudios filogenéticos y estudios de las funciones conservadas de los genes entre los genomas. Es tarea de la detección de ortólogos distinguir los genes que son ortólogos a partir de los homólogos en cuanto a la similitud de la secuencia. Otros genes son ortólogos ya que sus productos proteicos preservan su función aunque no conservan la similitud de la secuencia.

Los algoritmos de detección de genes ortólogos se clasifican en algoritmos: basados en grafos, basados en árboles e híbridos. Específicamente, los basados en grafos realizan inicialmente el alineamiento de secuencias, fundamentalmente basado en *BLAST (Altschul*

et al., 1990) y seguidamente aplican heurísticas para agrupar los posibles ortólogos. Se mantiene vigente la necesidad de utilizar valores de los parámetros del alineamiento que mejoren su precisión y en general la precisión de la clasificación de los genes (Hagelsieb and Latimer, 2008, Edgar, 2009).

Diversas bases de datos han sido construidas a partir de la predicción basada en grafo como por ejemplo: NCBI KOG (Tatusov et al., 2003), INPARANOID (O'Brien et al., 2005), OrthoMCL-DB (Chen et al., 2006), y más recientemente INPARANOID 7.0 (Östlund et al., 2010) y (Linard et al., 2011). Algunas de las bases han sido curadas manualmente como es el caso de la base GeneDB (GeneDB, 2013).

Los algoritmos de clasificación consultados se presentan como algoritmos no supervisados aunque algunos realizan verificaciones con bases de funciones de proteínas o con la Ontología de genes GO (Ozer et al., 2004). En (Estopiñales, 2009, Towfic et al., 2009, Fernández, 2012, Galpert et al., 2012) se muestra una tendencia a incluir varios rasgos que caracterizan la relación entre pares de genes además del alineamiento y la longitud de la secuencia. Estos rasgos pueden ser las posiciones y orientación de los genes luego de los reordenamientos globales y las interacciones entre proteínas.

En la literatura consultada no aparecen soluciones supervisadas al problema de la detección de ortólogos por lo que parece importante intentar el uso de los métodos de aprendizaje basados en las clasificaciones existentes. Ante la diversidad de clasificaciones, en este trabajo se seleccionan las que aparecen en (Koch et al., 2012) de Inparanoid y GeneDB.

Entre las técnicas de clasificación supervisadas empleadas se encuentra la Regresión logística, el “Bagging”, el “Random Forest”. El problema del desbalance resulta un problema crucial en este tipo de clasificación por los efectos que provoca sobre la exactitud. Se consultan algunos enfoques para el manejo del desbalance como los basados en ensamblaje (Yang et al., 2013) y los basados en los conjuntos aproximados (Jinfu Liu, 2008), (Owna and Abrahamb, 2012). Estas técnicas pudieran mejorar la calidad de la clasificación obtenida en trabajos anteriores.

Antecedentes

En el Laboratorio Bioinformática del Centro de Estudios de Informática se han realizado investigaciones sobre el tema, como por ejemplo en (Sánchez, 2012) se presentan nuevas medidas de similitud entre genes y un algoritmo no supervisado basado en grafo. Por otra parte en (Brito, 2012) se muestra el uso de la simulación para estudiar los valores de los parámetros del alineamiento como la matriz de sustitución y las penalidades de "huecos".

En la primera de las tesis mencionadas se realiza la experimentación con los genomas de Saccharomyces Cerevisiae y Schizosaccharomyces Pombe de 5861 y 5006 genes respectivamente. Se obtienen resultados comparables con los de la base INPARANOID con 95.35% de clasificación correcta de pares de ortólogos usando la medida del alineamiento de secuencias y un 94.56% combinando el alineamiento de secuencias con la medida del perfil físico-químico para tamaño de ventana 3. En dicho trabajo se inicia el uso de la Regresión logística tomando como referencia la clasificación de INPARANOID 7.0 sobre muestras aleatorias balanceadas con reposición. Los resultados obtenidos muestran alto por ciento de clasificados correctos, principalmente en la clase mayoritaria pero un número alto de falsos positivos, tendencia que se encuentra en la literatura para problemas supervisados con alto grado de desbalance (Chawla et al., 2004). Nos proponemos en este trabajo aplicar técnicas supervisadas que eleven la precisión de la clasificación teniendo en cuenta el desbalance de los datos.

Planteamiento del problema

¿Cómo construir algoritmos de detección de ortólogos supervisados que eleven el porciento de clasificados correctos en la clase minoritaria sin elevar los falsos positivos en una base de casos de comparación de dos genomas de gran tamaño y muy desbalanceada?

Objetivo general

Diseñar, implementar y validar algoritmos de clasificación supervisada para la detección de ortólogos con manejo del desbalance.

Objetivos específicos

1. Realizar un estudio crítico de los algoritmos de clasificación supervisada y manejo del desbalance.
2. Diseñar e implementar algoritmos para el preprocesamiento de los datos.
3. Diseñar algoritmos de clasificación a partir de las implementaciones existentes en los paquetes de software de clasificación.
4. Validar los algoritmos.
5. Realizar la comparación de los resultados.

Tareas de investigación

1. Estudio de las herramientas de clasificación supervisada y de la estructura de los datos de clasificaciones existentes.
2. Estudio de técnicas de manejo del desbalance.
3. Diseño e implementación de algoritmos para la conformación de la base de casos y el preprocesamiento de los datos.
4. Diseño e implementación de algoritmos de clasificación supervisada con manejo del desbalance.
5. Validación y comparación de resultados.

Justificación

- La detección de genes ortólogos se mantiene como un problema abierto en la bioinformática por su importancia para conocer las funciones de proteínas no conocidas a partir de los datos de genomas conocidos.

- La implementación de algoritmos de clasificación de genes a partir de paquetes de alta confiabilidad resultaría de gran utilidad para aumentar la precisión en la clasificación.
- El desarrollo de este trabajo constituiría un paso de avance en los proyectos del grupo de Bioinformática de la UCLV y en particular de la tesis doctoral "Herramientas computacionales para la comparación de genomas" aprobada por el CITMA.

El trabajo tiene novedad, así como valor práctico y metodológico.

- **Novedad científica y aportes**: La principal novedad del trabajo radica en el planteamiento del problema de clasificación de ortólogos como problema supervisado, aprovechando el conocimiento subyacente en clasificaciones conocidas y el diseño e implementación de algoritmos para resolver este tipo de problema, teniendo en cuenta el desbalance entre las clases.
- **Valor práctico:** La implementación de los algoritmos utilizando los paquetes de software de clasificación induce un alto grado de confiabilidad y permite el uso de los algoritmos por parte de otros investigadores para su posterior mejora.
- **Valor metodológico:** El trabajo presenta los nuevos algoritmos con los pasos necesarios para su uso en la experimentación.

Distribución en capítulos

El capítulo I recoge la revisión bibliográfica relacionada con las medidas de similitud entre genes. Aborda aspectos relacionados con las soluciones no supervisadas consultadas en la literatura. Incluye el planteamiento del problema de la detección de ortólogos como problema supervisado y las soluciones consultadas para el problema del desbalance entre clases dando pie a su uso y modificación en este trabajo.

El capítulo II abarca el diseño e implementación de algoritmos supervisados para la detección de ortólogos con manejo del desbalance. Incluye los pasos a seguir para la implementación utilizando paquetes de clasificación como Weka y SPSS.

El capítulo III describe los experimentos realizados así como la discusión de los resultados.

CAPÍTULO 1. MARCO TEÓRICO

En este capítulo se presenta un resumen de la revisión bibliográfica relacionada con las medidas de similitud entre genes y las soluciones consultadas para el problema de detección de ortólogos como problema de aprendizaje no supervisado, específicamente, las basadas en grafos. Se abordan algunas técnicas supervisadas a utilizar en este trabajo así como soluciones consultadas para el problema del desbalance y la forma en que se propone utilizarlas en la detección de ortólogos.

1.1 Medidas de similitud entre genes

El alineamiento de secuencias es una forma de representar y comparar dos o más secuencias para resaltar las regiones de mayor similitud, que podrían indicar relaciones funcionales o evolutivas entre los genes o proteínas comparadas. Puede analizar cambios genéticos a pequeña escala como son la inserción, eliminación o sustitución de aminoácidos, mientras que cambios como: la inversión, duplicación o reordenamiento de aminoácidos, no pueden ser analizadas a través del alineamiento de secuencias (Kamvysselis, 2003). Por esta razón, en (Sánchez, 2012) se presentan cuatro medidas de similitud que tienen en cuenta diferentes rasgos relacionados con las secuencias comparadas: el alineamiento de las secuencias, la longitud de las secuencias, la pertenencia de las secuencias a los bloques *"Locally Colinear Blocks"* LCB y la información del perfil físico-químico de las proteínas. Todas estas medidas de similitud son trabajadas con valores normalizados, dividiendo cada valor por el máximo valor de similitud obtenido. El grado de similitud negativo es considerado como una similitud no significativa, y se asume como valor nulo el cero.

1.1.1 Medida basada en el alineamiento de secuencias

El alineamiento local de dos secuencias brinda la relación funcional entre las secuencias, mientras que el global brinda la relación estructural. Para tener en cuenta ambas relaciones se emplea la medida basada en la combinación de ambos tipos de alineamientos propuesta en (Sánchez, 2012).

Sean los genomas $G_1 = (X_1, X_2, \dots, X_n)$ y $G_2 = (Y_1, Y_2, \dots, Y_m)$, con n y m secuencias respectivamente y $swalign$ la función que calcula el alineamiento local entre un par de secuencias, entonces la medida de similitud basada en la puntuación del alineamiento local entre cada par de secuencias de ambos genomas se define como:

$$S_{loc}(X_i, Y_j) = \begin{cases} ca_{loc}(X_i, Y_j) & ca_{loc}(X_i, Y_j) > 0 \\ 0 & ca_{loc}(X_i, Y_j) \leq 0 \end{cases}$$
$$ca_{loc}(X_i, Y_j) = \frac{swalign(X_i, Y_j)}{máx(swalign(X_k, Y_l))}, \quad \forall k \in [1, n], \forall l \in [1, m] \tag{1}$$

De forma similar se define la similitud basada en la puntuación del alineamiento global entre cada par de secuencias, sustituyendo la función de alineamiento local $swalign$ por la función de alineamiento global $nwalign$:

$$S_{glob}(X_i, Y_j) = \begin{cases} ca_{glob}(X_i, Y_j) & ca_{glob}(X_i, Y_j) > 0 \\ 0 & ca_{glob}(X_i, Y_j) \leq 0 \end{cases}$$
$$ca_{glob}(X_i, Y_j) = \frac{nwalign(X_i, Y_j)}{máx(nwalign(X_k, Y_l))}, \quad \forall k \in [1, n], \forall l \in [1, m] \tag{2}$$

La medida de similitud S_1 basada en la puntuación del alineamiento se expresa como la media aritmética de la combinación de la ecuación (1)(2):

$$S_1(X_i, Y_j) = \frac{S_{loc}(X_i, Y_j) + S_{glob}(X_i, Y_j)}{2} \tag{3}$$

1.1.2 Medida basada en la longitud de las secuencias

La medida de similitud S_2 basada en la longitud de las secuencias es calculada a partir de la distancia de las diferencias renormalizadas (Duch, 2000), y queda expresada como:

$$S_2(X_i, Y_j) = 1 - \frac{\left|long(X_i) - long(Y_j)\right|}{máx\left(long(Z_k)\right) - mín(long(Z_k))} \quad Z = X_1, X_2, \dots, X_n, Y_1, Y_2, \dots, Y_m \quad \forall k \in [1, n+m] \tag{4}$$

Si una secuencia X_i de un genoma tiene una longitud muy pequeña y otra secuencia Y_j del otro genoma tiene una secuencia relativamente grande, entonces es probable que tengan un alto grado de similitud, dado por la puntuación del alineamiento local. Esto puede provocar que secuencias poco semejantes se clasifiquen incorrectamente. Por otra parte, la similitud basada en sus longitudes tiende a ser un valor pequeño, el cual sirve de contrapartida al alto grado de similitud obtenido por el alineamiento.

1.1.3 Medida basada en la pertenencia a los bloques LCB

Los bloques LCB de los genomas comparados se obtienen con la herramienta Mauve (Darling et al., 2004), la cual identifica regiones conservadas que aparentan no haber sido alteradas por los reordenamientos e inversiones dentro del genoma. Los bloques obtenidos representan regiones consideradas verdaderamente homólogas (Darling, Mau et al. 2004)), y genes que se encuentran en un mismo bloque son más propensos a ser otólogos que los que están en diferentes bloques.

En (Estopiñales, 2009) se define que una secuencia pertenece a un bloque LCB si esta tiene al menos un aminoácido dentro del bloque. La similitud basada en la pertenencia a los bloques LCB, se calcula a partir de la distancia entre los pares de secuencias teniendo en cuenta la pertenencia a los bloques LCB, por lo que la medida de similitud S_3 se define como:

$$S_3(X_i, Y_j) = 1 - dlcb(X_i, Y_j) \tag{5}$$

Sean los genomas $G_1 = (X_1, X_2, \dots, X_n)$ y $G_2 = (Y_1, Y_2, \dots, Y_m)$, $LCB[k, l]$ la matriz de la cantidad de aminoácidos de cada secuencia en cada bloque LCB, donde $LCB[k, 1 \dots n]$ es la cantidad de aminoácidos de las secuencias del genoma G_1 en el bloque k y $LCB[k, n +$

$1 \dots n+m]$ es la cantidad de aminoácidos de las secuencias del genoma G_2; entonces la distancia entre dos secuencias X_i y Y_j de los genomas G_1 y G_2 se define como:

$$dlcb(X_i, Y_j) = \frac{1}{P} \times \sum_{k=1}^{cantidadLCB} dlcb(k, X_i, Y_j)$$

$$dlcb(k, X_i, Y_j) = \begin{cases} 0, si\ máx(LCB[k,l]) = mín(LCB[k,l]) \\ \frac{|LCB[k,i] - LCB[k,n+j]|}{máx(LCB[k,l]) - mín(LCB[k,l])} \end{cases} \quad \forall l \in [1, n+m]; k = 1 \dots cantidadLCB \tag{6}$$

Donde P es la cantidad de bloques LCB donde una o ambas secuencias del par (X_i, Y_j) contiene al menos un aminoácido.

1.1.4 Medida basada en la información del perfil físico-químico de las proteínas

A diferencia de los métodos tradicionales, el análisis del perfil físico-químico propuesto en (Sánchez, 2012) no se basa en el alineamiento de la representación espectral de las secuencias [Carpio-Muñoz y Carbajal, 2002], si no, en el cálculo de la representación espectral de las secuencias a partir del alineamiento global de ambas secuencias, usando codificación predictiva lineal *"Linear Predictive Coding"* (LPC) (Deza, 2006).

Tomando como referencia el alineamiento global de dos secuencias, se buscan las regiones de correspondencia sin *"gaps"* y se sustituye cada aminoácido por su energía de contacto para luego calcular la relación entre las dos representaciones espectrales usando el coeficiente de correlación de Pearson. Una vez sustituidos cada uno de los aminoácidos por su energía de contacto para determinar la representación espectral de la región, se calculan las medias móviles para cada uno de los espectros, con un tamaño de ventana W. A partir de los nuevos espectros obtenidos se calcula el coeficiente de correlación de Pearson, donde la correlación es significativa si el valor de significación obtenido es menor que *0.05*, por lo que el grado de similitud de una región sin *"gaps"* se define como:

$$Corr_{Pearson}\ (MX, MY) = \begin{cases} corr(MX, MY) & , sig \leq 0.05 \\ 0 & , sig > 0.05 \end{cases} \tag{7}$$

Una vez calculada la similitud de cada una de las R regiones sin *"gaps"*, es necesario combinar estas similitudes para determinar la similitud global de las dos secuencias

comparadas. Las similitudes de cada una de las regiones son agregadas hallando la relación del grado de similitud en función de la longitud de la región, para la longitud del alineamiento sin *"gaps"*, por lo que la medida de similitud S_4 basada en la información del perfil físico-químico de las proteínas se define como:

$$S_4(X_i, Y_j) = \frac{\sum_{k=0}^{R} Corr_{Pearson}\,(MX_{ik}, MY_{jk}) \times long_k}{\sum_{k=0}^{R} long_k} \quad \textbf{(8)}$$

Donde $long_k$ representa la longitud de la región k.

1.2 Soluciones no supervisadas al problema de la detección de ortólogos

Los algoritmos de detección de genes ortólogos basados en grafos como COG (Tatusov et al., 2003), INPARANOID (O'Brien et al., 2005), OrthoMCL (Li et al., 2003), EGO (Lee et al., 2002) y la más reciente versión 7.0 de INPARANOID (Östlund et al., 2010) se basan en la similitud entre pares de secuencias para conformar el grafo *G(V,E,W)* de similitud bipartito completo y pesado entre pares de genes de dos genomas en comparación donde *V* es el conjunto de genes de ambos genomas, *E* es el conjunto de arcos y *W* es el conjunto de pesos de los arcos. El problema de clasificación no supervisado de detección de ortólogos se presenta en (Vashist et al., 2005) a partir de la función *F(H)* que denota una medida de proximidad entre elementos del subconjunto $H \subseteq V$ de genes. Un grupo de genes o grupo de ortólogos es un subconjunto H^* de *V* que maximiza *F*(*H*), es decir, $H^* = \max_{H \subseteq V} F(H)$. Definiendo la función de enlace *π(i,H)* que mide la similitud entre un gen $i \epsilon H$ y un subconjunto H entonces $F(H) = \min_{i \epsilon H} \pi(i, H), \forall i \epsilon H, \forall H \subseteq V$ es el valor *π(i,H)* del elemento menos similar en *H*. Entonces H^* contiene elementos tales que la similitud del elemento menos similar en *H* es máxima.

Los algoritmos basados en grafo inicialmente realizan la comparación par a par de las secuencias usando BLAST (Altschul et al., 1990) y luego hacen un filtrado de los datos, en el que se eliminan secuencias muy pequeñas y poco similares. La similitud entre pares de secuencias se define a partir de la puntuación obtenida en cada comparación o por su porciento de similitud. Por lo general las puntuaciones obtenidas al comparar los pares

(X,Y) y *(Y,X)* son asimétricas, por lo que sus valores de puntuación son promediados (Remm et al., 2001).

El grafo bipartido de similitud es construido de forma que las secuencias de ambos genomas representan nodos del grafo, y los nodos se enlazan por una arista cuyo peso está dado por las puntuaciones obtenidas en las comparaciones. Una vez conformado el grafo se aplica una definición operacional de ortología para determinar los posibles pares de genes ortólogos y parálogos, desechando los pares de genes menos semejantes. Las definiciones de ortología más usadas por los algoritmos son "Reciprocal Best Hits" RBH (Tatusov et al., 1997) y "Bidirectional Best Hits" BBH (Overbeek et al., 1999). A los posibles pares de genes ortólogos y parálogos se les aplica un algoritmo de agrupamiento sobre grafo para conformar grupos de pares de genes (Li et al., 2003), y a partir de estos determinar mediante diversos criterios los pares de genes ortólogos.

Algunos algoritmos como el BUS (Kamvysselis, 2003) aplican políticas de poda por umbral, y realizan una asignación de ortólogos a través de heurísticas. Otros como el OrthoMCL realizan el agrupamiento sobre el grafo bipartido con el algoritmo MCL (Van Dongen, 2000). Aún no existe un criterio óptimo para seleccionar una política de poda, un algoritmo de agrupamiento o una estrategia de asignación de ortólogos. En (Sánchez, 2012) se utiliza el agrupamiento MCL (Van Dongen, 2000) con el valor de inflación usado por el algoritmo OrthoMCL y se propone una nueva política de asignación de genes tratando de disminuir los falsos positivos en la clasificación.

1.3 Planteamiento del problema de detección de ortólogos supervisado

El principio del Aprendizaje Automático Supervisado en Aprendizaje Computarizado parte de una muestra de aprendizaje $L = \{(p_1, z_1), (p_2, z_2), \dots, (p_n, z_n)\}$ constituida por n realizaciones de un par de variables aleatorias *(P,Z)* donde P es un vector de entrada o variable explicativa y Z una variable de clase (o variable dependiente) que puede ser continua o discreta (si Z es continua se trata de un problema de regresión y si es discreta se trata de un problema de clasificación). En el caso de la detección de ortólogos P sería el vector de similitud de pares de genes y $Z=\{0,1\}$. Se necesita construir una función $f: P \rightarrow Z$

con la cual, dada una nuevo vector de entrada *P*, se pueda predecir con cierto grado de certeza la variable *Z= f(P).*

1.4 Técnicas de clasificación supervisada

1.4.1 Regresión logística

Los modelos de Regresión logística (Wasserman and Fickett, 1998) son modelos estadísticos en los que se desea conocer la relación entre una variable dependiente cualitativa, dicotómica (Regresión logística binaria o binomial) o con más de dos valores (Regresión logística multinomial) y una o más variables explicativas independientes, o covariables, ya sean cualitativas o cuantitativas. Los modelos re regresión se expresan de la siguiente forma: *Y= f (x1, x2, ...) + E*. Siendo la ecuación inicial del modelo de tipo exponencial, su transformación logarítmica (logit) permite su uso como una función lineal. Se construye entonces una función basada en el cálculo de la probabilidad de que la variable de interés adopte el valor del evento previamente definido, de la manera siguiente: *Y = ln (p / (1-p))* de forma que la nueva variable dependiente construida que se va a estimar sí puede tomar cualquier valor (no está restringida a un rango de valores) y se puede recurrir a los métodos de estimación de los modelos de regresión tradicionales para construir el modelo de Regresión logística.

Tras realizar una serie de transformaciones matemáticas se puede deducir que:

$p = \frac{1}{1+e^{-\text{modelo de regresión}}}$. En el modelo de Regresión logística se estima un modelo de regresión que en lugar de realizar estimaciones para la variable dependiente real, las realiza sobre la función de probabilidad asociada a ella, pudiendo entonces aplicar los métodos de estimación aplicables al modelo de regresión lineal, diferenciándose entonces ambos modelos únicamente en la interpretación de resultados.

Para entender en qué consiste un modelo de regresión, así como para interpretar correctamente los resultados, se deben relacionar dos conceptos: el coeficiente de correlación y el análisis de la varianza. Se puede demostrar que existe una relación entre el coeficiente de correlación (r) y el análisis de la varianza de la regresión, de tal forma que el cuadrado de r, llamado coeficiente de determinación, multiplicado por 100 se interpreta

como el porcentaje de la varianza de la variable dependiente que queda explicada por el modelo de regresión.

Ambos modelos de regresión permiten que las variables utilizadas para poder estimar el modelo (variables independientes) puedan ser de cualquier tipo, no existen restricciones sobre ellas. Únicamente se debe tener en cuenta una serie de consideraciones que se explican a continuación. Cuando la variable que se incluya en el modelo sea una variable continua, se introducirá la variable real o bien alguna transformación de ella (logaritmo, cuadrado, etc.), cuando sea necesario. Sin embargo, cuando la variable que se quiera introducir en el modelo sea una variable categórica de más de dos categorías, se necesitará recurrir a una serie de transformaciones para que los resultados obtenidos sobre la variable en cuestión sean correctos e interpretables. En dicho caso, no se podrá introducir la variable original en el modelo, sino que si la variable tiene n categorías deberán expresarse cada una de estas categorías mediante n-1 variables *dummy*. Una variable *dummy* es una variable construida artificialmente y que únicamente puede tomar los valores 0 o 1. Es posible expresar la misma información contenida en una variable de n categorías mediante la combinación de n-1 variables *dummy*. Adicionalmente estas variables permiten realizar todas las comparaciones necesarias respecto a las n categorías de la variable original en el modelo de regresión.

Los paquetes estadísticos realizan la transformación de la variables categóricas en las variables *dummy* necesarias automáticamente y no es necesario realizar todo el proceso manualmente, únicamente debe identificarse al programa cuál es el nombre de las variables que requieren este tipo de transformación. Es importante conocer este requisito al construir un modelo de regresión, principalmente al interpretar los resultados obtenidos, dado que en este tipo de variables, se obtiene un resultado global y otro para cada una de las variables *dummy*, debiendo conocerse qué comparación se realiza a través de cada una de las variables *dummy* que intervienen en el modelo para poder explicar los resultados correctamente.

La elección de las variables que deben introducirse en el modelo no debería suponer un problema, dado que al estar el objetivo del estudio concretamente definido, automáticamente las variables que son de mayor interés también quedan identificadas. Sin

embargo, no siempre queda todo tan evidente y se plantea encontrar "relaciones" posibles para evaluar una única respuesta claramente identificada. Puede ocurrir que aunque el objetivo de nuestro estudio esté bien definido, no se disponga de información previa o de indicio alguno que nos pueda indicar qué aspectos son los que suscitan mayor interés. Si a este hecho se le añade, el impulso de registrar más información de la que realmente se necesita (exceso de información), construir un modelo de regresión resulta un proceso laborioso y complicado. Por lo tanto, se recomienda desde el inicio elegir con esmero las variables indispensables para su análisis estadístico.

Resulta imprescindible como primer contacto con el modelo analizar la relación bivariante entre la característica de interés y el resto de variables registradas durante el estudio una a una. Como criterios para seleccionar aquellas variables a introducir en el modelo de regresión se podría aplicar los siguientes: en primera instancia y entre otros criterios igualmente válidos, introducir en el modelo aquellas variables que resultaron estadísticamente significativas en las comparaciones bivariantes realizadas previamente, o en un segundo plano debería considerarse la conveniencia de incluir en el modelo adicionalmente aquellas variables que se considere especialmente importantes o influyentes, como si se sospecha que a pesar de no haber resultado estadísticamente significativas, podrían modificar o intervenir en los resultados, otra serie de variables de las que se haya tenido conocimiento de su influencia a través de estudios previos.

Otro concepto importante que debe ser tenido en cuenta al construir un modelo de regresión es que pueden introducirse términos independientes únicos (una sola variable) y además las interacciones entre variables de cualquier orden, si pueden ser de interés o afectar a los resultados. Al introducir los términos de interacción en un modelo de regresión es importante para la correcta estimación del modelo respetar un orden jerárquico, es decir, siempre que se introduzca un término de interacción de orden superior (x•y•z), deben introducirse en el modelo los términos de interacción de orden inferior (x•y, x•z, y•z) y por supuesto los términos independientes de las variables que participan en la interacción (x, y, z). Si se introducen en un modelo de regresión términos de interacción y resultan estadísticamente significativos, no se podrán eliminar del modelo los términos de interacción de orden inferiores ni los términos independientes de las variables que

participan en la interacción para simplificarlo, deben mantenerse, aunque no resulten estadísticamente significativos.

Existen varios métodos para construir el modelo de regresión, es decir, para seleccionar de entre todas las variables que se introducen en el modelo, cuáles son las que se necesitan para explicarlo. El modelo de regresión se puede construir utilizando las siguientes técnicas:

- Técnica de pasos hacia adelante *(Forward))*: consiste en ir introduciendo las variables en el modelo únicamente si cumplen una serie de condiciones hasta que no se pueda introducir ninguna más, hasta que ninguna cumpla la condición impuesta.
- Técnica de pasos hacia atrás *(Backward)*: se introducen en el modelo todas las variables y se van suprimiendo si cumplen una serie de condiciones definidas a priori hasta que no se pueden eliminar más, es decir ninguna variable cumpla la condición impuesta.
- Técnica por pasos *(Stepwise)*: combina los dos métodos anteriores, adelante y atrás introduciendo o eliminando variables del modelo si cumplen una serie de condiciones definidas a priori hasta que ninguna variable satisfaga ninguna de las condiciones expuestas de entrada o salida del modelo.
- Técnica de introducir todas las variables obligatoriamente *(Enter):* Esta última técnica de selección de variables para construir el modelo de regresión, produce que el proceso de selección de las variables sea manual, partiendo de un modelo inicial, en el que se obliga a que entren todas las variables seleccionadas, se va evaluando qué variable es la que menos participa en él y se elimina, volviendo a construir un nuevo modelo de regresión aplicando la misma técnica, pero excluyendo la variable seleccionada y aplicando el mismo proceso de selección. Este proceso se repite reiteradamente hasta que se considere que el modelo obtenido es el que mejor se ajusta a las condiciones impuestas y que no se puede eliminar ninguna variable más de las que los componen.

Para evaluar la adecuación de los modelos construidos, es conveniente comenzar a evaluar el modelo saturado, es decir el modelo que contiene todas las variables de interés a evaluar y todas las interacciones posibles. Progresivamente se van eliminando del modelo aquellos

términos no significativos, respetando el modelo jerárquico y comenzando por los términos de interacción superiores. Si un término de interacción es significativo, no podrán eliminarse del modelo los términos de interacción de grado inferior, ni los términos independientes de las variables que participan en la interacción. Las variables introducidas en el modelo se van eliminando progresivamente a cada nuevo modelo que se construye en base a los resultados obtenidos en el modelo anterior, y se van evaluando los nuevos modelos, de la manera que se explicará más adelante. Los coeficientes de las variables que permanecen en el modelo no varían de forma exagerada tras la eliminación de alguno de los términos del modelo, dado que si así sucediera, podría tratarse de un factor de confusión y por tanto debería mantenerse la variable en cuestión en el modelo, para permitir el ajuste del resto de variables y no obtener resultados artificiales.

1.4.2 Bagging

Bagging fue propuesta en (Breiman, 1994) para mejorar la clasificación combinando clasificaciones de conjuntos de entrenamiento generados aleatoriamente. La técnica de Bagging es un método que genera múltiples versiones de un predictor y usa estas para obtener un predictor agregado.

Los promedios de agregación sobre las versiones, cuando se realiza una predicción numérica, generan un voto cuando se predice una clase. Las versiones múltiples se forman al hacer réplicas de *bootstrap* del grupo de aprendizaje y usando éstas como nuevos conjuntos de aprendizaje. Las pruebas con conjuntos de datos reales y simulados usando clasificación y árboles de regresión, y subgrupos en regresión linear, muestran que con Bagging se tiene una ganancia de alta efectividad. El elemento vital es la estabilidad del método de predicción. Si al perturbar el conjunto de aprendizaje puede causar cambios significantes en el predictor construido, entonces el proceso de Bagging puede mejorar su efectividad.

Descripción de la técnica

Dado un conjunto de entrenamiento *D* de tamaño *n*, Bagging genera nuevos conjuntos *m* de entrenamiento D_i, de tamaño *n'*<*n*, tomando muestras de *D* uniformemente y con reemplazo. En el muestreo con reemplazo, es probable que algunos ejemplos estén repetidos en cada D_i. Si *n'*=*n*, entonces si *n* es grande, para el conjunto D_i, se espera tener la fracción $\left(1-\frac{1}{e}\right)(\approx 63.2\%)$ de los ejemplos únicos de *D*, el resto siendo duplicados. Este tipo de muestra es conocido como muestra de *bootstrap*. Los *m* modelos son ajustados usando las *m* muestras de *bootstrap* anteriores y promediando la salida, en el caso de la regresión, o votando, en caso de la clasificación.

1.4.3 Random Forest

Random Forest (Bosque Aleatorio) es una variante propuesta por (Breiman, 2001) y constituye una combinación de predictores de árboles de tal forma que cada árbol depende de los valores de un vector aleatorio muestreado independientemente y con la misma distribución para todos los árboles del bosque. La generalización de los errores converge hasta un límite en correspondencia con el crecimiento del número de árboles.

La generalización de un error en Random Forest depende de la fuerza de cada árbol perteneciente al bosque y de la correlación entre ellos. Utiliza la selección aleatoria de rasgos para separar el rendimiento de cada nodo. Los estimados internos de errores de monitoreo, fuerza, y correlación, son usados para mostrar la respuesta al incrementar el número de rasgos usados en la separación. Los estimados internos también son usados para medir la importancia de la variable. Estas ideas son también aplicables a la regresión.

Descripción de la técnica

Random Forest presenta la mejor combinación publicada de *decisión trees* y *Bagging*. El algoritmo genera múltiples árboles *(ℓ_1)* del conjunto de entrenamiento de un vector aleatorio *X*, muestreado independientemente con la misma distribución para cada árbol que forma parte del bosque. Como resultado, cada árbol genera un clasificador *h($x_1 \ell_1$)*. El voto mayoritario de todos los árboles determina la clase predicha. Random Forest genera una

marca estandarizada *z*, que indica la importancia de cada variable al final de la clasificación.

1.5 Soluciones para problemas supervisados muy desbalanceados

El problema del desbalance es reconocido como uno de los problemas más relevantes en Aprendizaje Computarizado. Se refiere a que los métodos de aprendizaje deben aprender a clasificar una clase minoritaria con mucho menos casos respecto a la mayoritaria. Al tratar de aprender a partir de datos desbalanceados, los métodos tradicionales tienden a producir un alto valor de exactitud para la clase mayoritaria y un bajo valor de exactitud para la clase minoritaria. (Chen et al., 2008), (Chawla et al., 2004). Este problema puede presentarse en aplicaciones como el diagnóstico médico, o de productos defectuosos, entre productos terminados, la identificación de causa de fallas eléctricas, la detección de intrusos en la red, el manejo de riesgos, el reconocimiento de textos, el reconocimiento de voz y en el campo de la Bioinformática, el reconocimiento de sitios de proteínas, y las aplicaciones de genómica funcional.

En el artículo (Chen et al., 2008) se mencionan dos grandes enfoques para tratar el problema del desbalance desde la perspectiva de los datos. El primero abarca (1) la reducción por muestreo ("under-sampling"), en la que se mantienen intacta la clase minoritaria mientras se reduce la clase mayoritaria por muestreo; (2) el sobre muestreo ("over-sampling"), en el que la clase minoritaria se sobre muestrea de modo que se logre una distribución por clase deseada en el conjunto de entrenamiento; (3) muestreo basado en agrupamiento, en el que los casos representativos son aleatoriamente muestreados a partir de los grupos. Además, en (Liu et al., 2008) se presentan un método pesado basado en los conjuntos aproximados. Sin embargo, estos métodos muestran algunas desventajas como el aumento de la carga computacional y el sobre entrenamiento debido a casos replicados en el caso del sobre muestreo. La reducción por muestreo no tiene en cuenta toda la información del conjunto de aprendizaje lo que conlleva una pérdida de información.

Por otra parte el segundo enfoque para tratar el desbalance está relacionado con los modelos de computación granular que utilizan el concepto de subatributos para describir los gránulos, colecciones de objetos ordenados por su similitud, adyacencia funcional o capacidad de distinguirse. En este enfoque cuando se manejan datos continuos aumenta la carga computacional debido a la generación de un gran número de subatributos.

En (Chawla et al., 2004) se mencionan enfoques de tratamiento del desbalance desde el punto de vista de los algoritmos. Algunas soluciones tratan de adaptar los umbrales de decisión para imponer sesgo en la clase minoritaria. Otras soluciones incluyen el ajuste de los costos de las clases, el ajuste de la estimación de probabilidad en el nivel de las hojas (cuando se trabaja con árboles de decisión), el ajuste del umbral de decisión o el aprendizaje basado en una clase en lugar de en dos o más. En el caso de los clasificadores sensitivos al costo se le asigna un costo superior de falla en la clasificación a la clase minoritaria y se trata de minimizar el costo total. Un ejemplo de este tipo de clasificadores es el presentado en (Chen et al., 2004) como "Weighted Random Forest" donde los autores asignan un peso superior a las fallas de clasificación de la clase minoritaria.
En las clasificaciones de dos clases la matriz de costo puede ser representada como C(+,-) que significa el costo de fallar en la clasificación de un caso positivo (par de ortólogos) como caso negativo y C(-,+) que penaliza el caso contrario. Un clasificador sensitivo al costo generalmente considera que C(+,-)>C(-,+) asignando un mayor costo a los falsos negativos. Sin embargo, en la detección de ortólogos es más conveniente considerar C(+,-)<C(-,+) penalizando más el riesgo de asignarle la clase ortólogo a un par cuando no lo es, es decir, se considera más incorrecta la posible asignación de una función a una proteína cuando no la tiene al clasificarla como ortóloga de otra. Con esta consideración le asignamos mayor costo a los falsos positivos.

Un tipo de clasificadores sensitivos al costo son aquellos insensitivos que se convierten en sensitivos y son conocidos como sensitivos al costo con meta aprendizaje. En esta categoría se encuentran los clasificadores que asignan peso a los casos, específicamente, que asignan mayor peso a los casos de la clase mayoritaria basándose en la distribución local.

Como otra solución al desbalance, en (Yang et al., 2013) se presenta un clasificador basado en mezcla que construye múltiples muestras balanceadas utilizadas para entrenar los clasificadores de base. Estos son aplicados entonces a una muestra desbalanceada de prueba. Las distribuciones de clasificación de cada muestra en el conjunto de prueba son normalizadas y combinadas para evaluar la medida de calidad con vistas a una selección de rasgos en un procedimiento de envoltura "wrapper" ávido.

1.6 Validación

1.6.1 Métodos de evaluación

Entre las técnicas de evaluación se encuentra la técnica "*holdout*" donde se particiona el conjunto de datos en conjunto de entrenamiento y conjunto de prueba (Le, 2011). Los parámetros del modelo se estiman en el conjunto de entrenamiento y las medidas de calidad se evalúan en el conjunto de prueba. Esto pudiera traer reducción de la información de ambos conjuntos. Por esta razón en varios de los experimentos de este trabajo se elige el conjunto de entrenamiento por distintas vías de acuerdo al algoritmo correspondiente y se prueban los resultados en el conjunto completo.

Otra de las técnicas de evaluación de clasificadores es la validación cruzada de *n* pliegues (Lachenbrush and Mickey, 1968) que utiliza todos los datos para el entrenamiento y la prueba dividiendo el conjunto de datos en n partes aproximadamente iguales y aplica el método "*holdout*" *n* veces. En cada una de las *n* corridas se utiliza un subconjunto de la muestra para la prueba y el resto para el aprendizaje. El funcionamiento general se mide promediando las medidas de validación.

1.6.2 Medidas de calidad

A partir de la matriz de confusión que aparece en la tabla 1.1 se pueden calcular las medidas que aparecen a continuación.

		Predicción	
		NEGATIVA	POSITIVA
Observada	Negativa	TN	FP
	Positiva	FN	TP

Tabla 1. 1 : Matriz de confusión

$$Accuracy = (TP + TN)/(TP + FP + TN + FN) \quad (9)$$

$$TPRate = Recall = TP/(TP + FN) \quad (10)$$

$$Precision = TP/TP + FP \quad (11)$$

$$F - Measure = (1 + \beta^2) * recall * precision/(\beta^2 * recall + precision) \quad (12)$$

La medida exactitud "*accuracy*" no resulta conveniente para medir la calidad ante un problema de desbalance (Padmaja et al., 2009).

Por otra parte, una **curva ROC** (acrónimo de **Receiver Operating Characteristic**, o Característica Operativa del Receptor) (Hanley, 1983) es una representación gráfica de la sensibilidad frente a (1 – especificidad) para un sistema clasificador binario según se varía el umbral de discriminación. Otra interpretación de este gráfico es la representación de la razón o ratio de verdaderos positivos (VPR = Razón de Verdaderos Positivos) frente a la razón o ratio de falsos positivos (FPR = Razón de Falsos Positivos) también según se varía el umbral de discriminación (valor a partir del cual decidimos que un caso es un positivo). ROC también puede significar Relative Operating Characteristic (Característica Operativa Relativa) porque es una comparación de dos características operativas (VPR y FPR) según cambiamos el umbral para la decisión.

El análisis de la curva ROC proporciona herramientas para seleccionar los modelos posiblemente óptimos y descartar modelos subóptimos independientemente de (y antes de especificar) el coste de la distribución de las dos clases sobre las que se decide. La curva ROC es también independiente de la distribución de las clases en la población, sin

embargo, en (Chawla et al., 2004) se plantea que las curvas ROC pueden estar afectadas por el desbalance.

1.7 Conclusiones parciales

Las medidas de similitud entre genes pueden ser consideradas como rasgos en los conjuntos de entrenamiento de la clasificación supervisada. Este tipo de clasificación puede incorporar al aprendizaje la información de clasificaciones conocidas ante la diversidad de clasificaciones que se obtienen con la técnica no supervisada. Las soluciones consultadas para el problema del desbalance pueden ser combinadas en nuevos algoritmos que utilicen la mezcla de modelos como los de Regresión logística y las técnicas de reducción de casos. La mezcla de modelos de Regresión logística se puede realizar a través de la evaluación de la función de probabilidad en el conjunto de prueba. Los algoritmos sensitivos al costo pueden ser combinados con técnicas de reducción para tratar de disminuir los falsos positivos obteniendo resultados aceptables de precisión.

CAPÍTULO 2. DISEÑO E IMPLEMENTACIÓN DE ALGORITMOS SUPERVISADOS

En este capítulo se expone la solución propuesta en dos algoritmos supervisados con manejo del desbalance. Inicialmente se aborda el paso de preprocesamiento de los datos y luego se describen los pasos de los algoritmos de clasificación así como su implementación utilizando los paquetes Weka y SPSS. En el diseño de los algoritmos se incluye el cálculo de la matriz de confusión y de los índices de calidad.

Weka es un conjunto de librerías JAVA para la extracción de conocimientos desde bases de datos, que contiene diversas herramientas para el preprocesamiento de los datos, selección de atributos y algoritmos de clasificación distribuidos en paquetes. Por su parte el SPSS (*Statistical Package for the Social Sciences)* es un instrumento de análisis multivariante de datos cuantitativos que está diseñado para el manejo de datos estadísticos, además es un paquete estadístico que posee gran capacidad para trabajar con bases de datos de gran tamaño.

2.1. Preprocesamiento de los datos

Los pares de genes que se utilizan para la realización de este estudio se obtienen a partir de los ficheros FASTA que contienen los genes de los genomas que se comparan. Se conforma un universo de pares $U = \{(X_i, Y_j)\}, \forall X_i \in V_1, \forall Y_j \in V_2$ donde V_1 y V_2 son los conjuntos de genes de los respectivos genomas. El conjunto de datos inicial se representa como un sistema de información (Komorowski and Polkowski, 1999) (*U*, *A*, *Z*) donde *A* es el conjunto de atributos y *Z* es el atributo de decisión binario. El conjunto de atributos *A* formado a partir del cálculo de las medidas de similitud entre pares de genes se define como $A = \{S_1^M(X,Y), S_2(X,Y), S_3(X,Y), S_4^M(X,Y)\}$, donde M representa uno de los modelos de alineamiento definidos a continuación utilizando valores recomendados de parámetros de alineamiento y penalización de "gaps".

M	Matriz de sustitución	Apertura de "gap"	Extensión de "gap"
1	BLOSUM50	15	8
2	BLOSUM62	8	7
3	BLOSUM62	12	6
4	PAM250	10	8

Tabla 2. 1: Modelos de alineamiento para la ejecución del algoritmo

La poda inicial por homología reduce el conjunto inicial manteniendo los pares cuyo porciento de similitud en el alineamiento global supera el 40% para todos los modelos de alineamiento.

La figura 2.1 contiene el diagrama de actividades del paso de preprocesamiento.

Figura 2. 1: Pasos para el preprocesamiento de datos

El manejo y preparación de los datos, la construcción y procesamiento de la base de casos a utilizar se implementó en *Java*, un lenguaje de programación de alto nivel, orientado a objetos, sencillo, robusto y seguro. Los programas de *Java* son compilados en un código intermedio, el cual posteriormente es interpretado por la máquina virtual, lo que permite la portabilidad del programa. Precisamente, la portabilidad permite que un programa escrito en *Java* se ejecute de forma similar en cualquier sistema operativo e independiente del hardware.

El estándar de *Java* dispone de un conjunto de clases que permiten el manejo de archivos binarios, texto plano y *XML*, manejo de excepciones, tipos de datos, algoritmos de ordenamiento y búsqueda, entre otros. También dispone de clases para el manejo de diferentes estructuras de datos como pueden ser: pilas, colas, listas, árboles y diccionarios.

2.2 Algoritmo basado en el ensamblaje de clasificadores de Regresión logística

Este algoritmo parte de que la base de casos con la que se desea trabajar tiene un nivel de desbalance muy alto. La figura 2.2 muestra la descripción del algoritmo basado en el ensamblaje de clasificadores de Regresión logística propuesto.

Primeramente, se debe seleccionar la clasificación que se tomará como referencia para aplicar el algoritmo. Seguidamente se procede a generar diez muestras aleatorias balanceadas a partir del conjunto de datos podado mediante una selección de casos en el SPSS. El filtro de propagación para lograr la distribución uniforme se halla calculando la proporción existente entre los valores positivos de la clase seleccionada y el total de casos. En la figura 2.3 se puede apreciar un ejemplo de la sintaxis que se aplica para generar una muestra aleatoria balanceada respecto a la clasificación GeneDB para los genomas *Saccharomyces cerevisiae* y *Schizosaccharomyces pombe*, donde el filtro de propagación es 0.00013. Una vez seleccionadas las muestras estas se dividen en los cuatro modelos de alineamiento para aplicarle los clasificadores de forma independiente a cada uno de ellos.

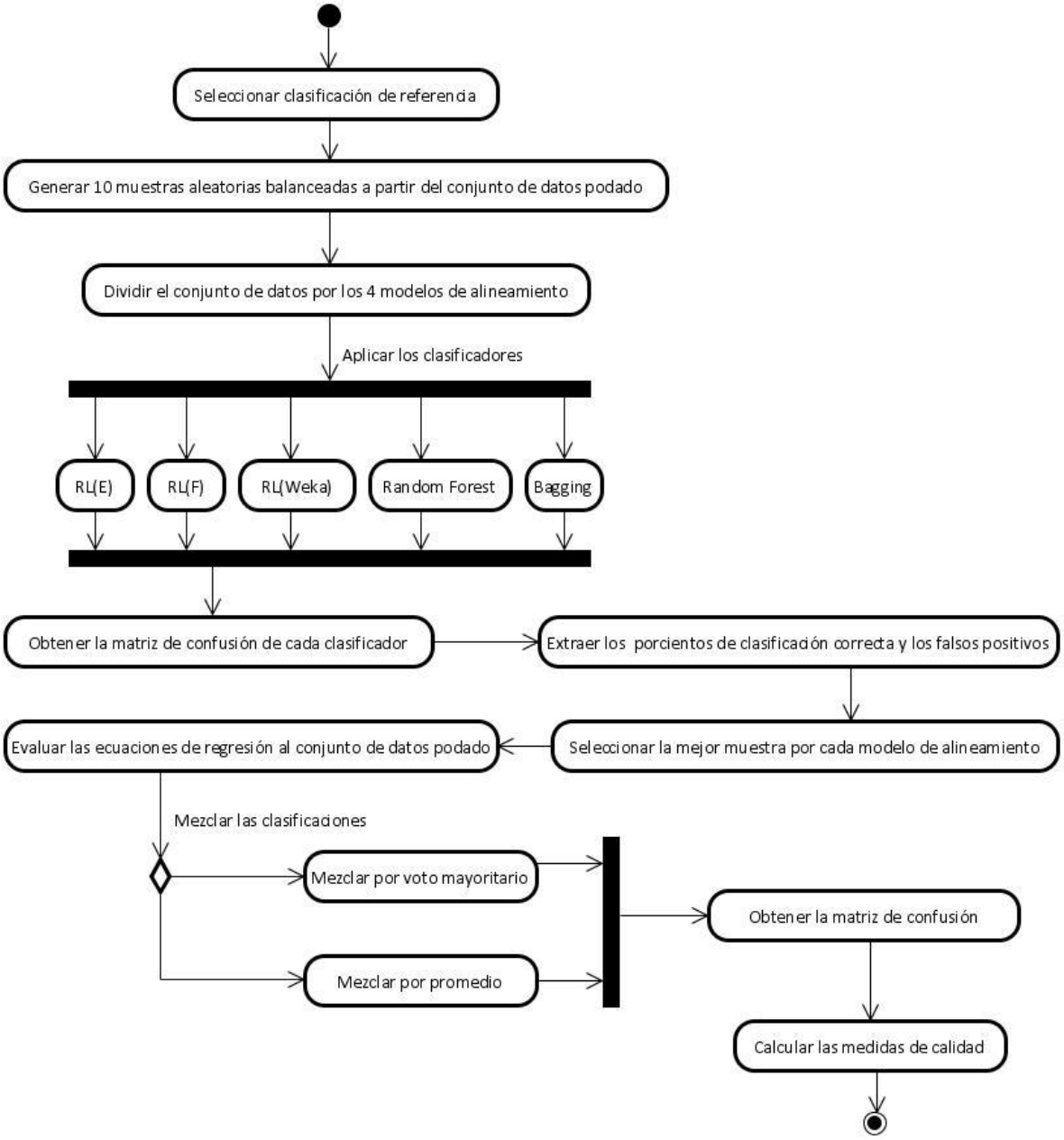


Figura 2. 2: Descripción del algoritmo basado en el ensamblaje de clasificadores de Regresión logística

```
USE ALL.
COMPUTE filter_$=((GeneDB=0)&(uniform(1)<=.00013))|(GeneDB=1).
VARIABLE LABEL filter_$ .
FORMAT filter_$ (f1.0).
FILTER BY filter_$.
EXECUTE .

DATASET COPY  prueba.
DATASET ACTIVATE  prueba.
FILTER OFF.
USE ALL.
SELECT IF  (NOT(filter_$=0)).
DATASET ACTIVATE  DataSet2.
EXECUTE.
```

Figura 2. 3: Selección de una muestra balanceada respecto a la clase GeneDB

2.2.1 Aplicación de los clasificadores

El segundo paso de este algoritmo consiste en aplicarle los clasificadores a las muestras generadas, que serán Regresión logística, Random Forest y Bagging. En el paquete estadístico SPSS se aplica el modelo de Regresión logística por los métodos de selección de variables Enter *(Introducir)* y Forward *(Pasos hacia delante)*, de este último se representa la sintaxis en la figura 2.4.

```
GET
  FILE='D:\Users\David\Tesis de David\Data\MuestrasOK\Interseccion\Blosum50\IntB50_01.sav'.
DATASET NAME IntB50_01 WINDOW=FRONT.
DATASET ACTIVATE IntB50_01.
LOGISTIC REGRESSION VARIABLES ClaseInterseccion
  /METHOD=FSTEP (COND) Longitud AlingBlos50 Blos50Vent3 Blos50Vent5 Blos50Vent7 LCB
  /PRINT=CORR
  /SAVE=PRED PGROUP
  /CRITERIA=PIN(0.05) POUT(0.10) ITERATE(20) CUT(0.5)
  /OUTFILE= MODEL ('D:\Users\David\Tesis de David\Data\MuestrasOK\Interseccion\Blosum50\XMLs(FORWARD)\RESULT1.xml').
SAVE OUTFILE='D:\Users\David\Tesis de David\Data\MuestrasOK\Interseccion\Blosum50\ResultadosRL(FORWARD)\IntB50_01.sav'
  /COMPRESSED.
DATASET CLOSE IntB50_01.
```

Figura 2. 4: Sintaxis para aplicar Regresión logística por el método forward a una muestra con el modelo Blosum50 para la clase Intersección

Por otra parte se utiliza la herramienta Weka para aplicar Bagging, Random Forest, y nuevamente Regresión logística, este último para comparar con los resultados obtenidos en el SPSS.

Weka posee una interfaz de experimentación con algoritmos de clasificación la cual permite seleccionar un clasificador *(botón [Choose])* y configurar sus parámetros *(pulsando sobre el nombre del algoritmo)*, además posibilita especificar el método de evaluación *(Test options)*, del cual en este caso utilizaremos la validación cruzada *(Cross Validation)*, que divide los datos disponibles en *k* grupos y realiza *k* tandas de entrenamiento-evaluación diferentes, usando *k-1* grupos para entrenar y el restante para la validación. Mediante esta vía se aplicará el clasificador Bagging usando coma base la Regresión logística, cuya implementación se encuentra en el apartado *META*, mientras en el apartado *TREES* se encuentra Random Forest. El clasificador de Regresión logística se encuentra dentro de las funciones de Weka.

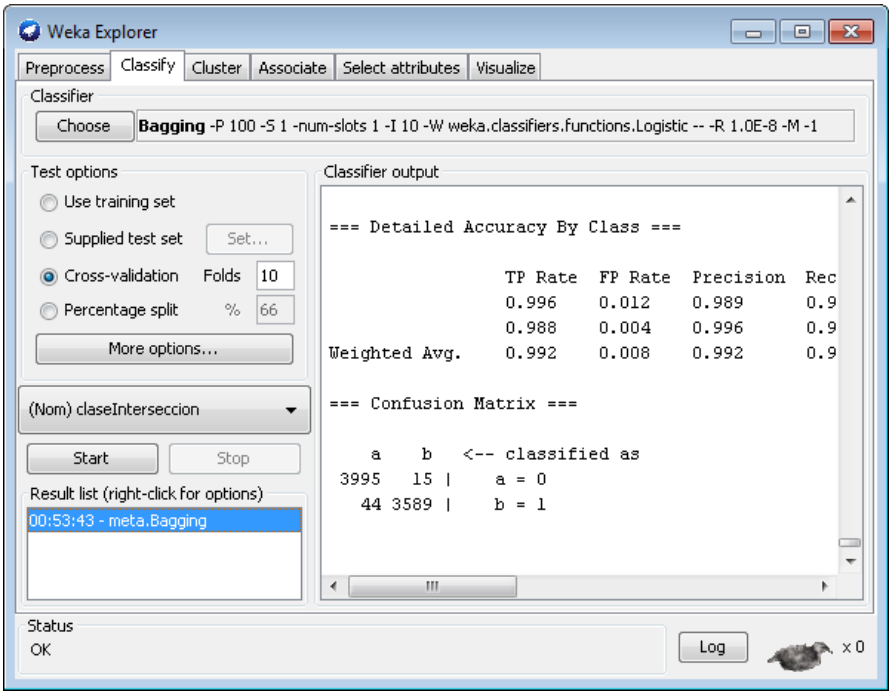


Figura 2. 5: Corrida del clasificador Bagging en Weka

2.2.2 Mezcla de resultados y validación

Al terminar de aplicar los clasificadores a las muestras se confecciona la matriz de confusión de cada una de las operaciones aplicadas para calcular las medidas de calidad, así como la cantidad de falsos positivos y falsos negativos. Los resultados obtenidos son comparados entre ellos para hacer una selección de la mejor muestra por cada modelo de alineamiento. Se selecciona la muestra que tenga los mejores porcientos de precisión y la medida F-measure.

Una vez hecha la selección de las mejores muestras se construye para cada una la ecuación de probabilidad (x) resultante de las regresiones logísticas aplicadas. Los coeficientes β_i de dicha ecuación se obtienen de la tabla que contiene las variables en la ecuación en la salida del SPSS. En el ejemplo de la figura 2.6 la tabla se obtiene al aplicar la Regresión logística por el método Enter, este caso particular representa la muestra 10 con el modelo Blosum50 para la clase Intersección, los coeficientes señalados son los que se toman para construir la ecuación de probabilidad.

Variables en la ecuación

		B	E.T.	Wald	gl	Sig.	Exp(B)
Paso 1[a]	Longitud	44,219	12,057	13,449	1	,000	1,600E+19
	AlingBlos50	1650,722	97,552	286,338	1	,000	.
	Blos50Vent3	10,066	3,148	10,225	1	,001	23531,954
	Blos50Vent5	1,313	3,435	,146	1	,702	3,718
	Blos50Vent7	-,815	2,595	,099	1	,754	,443
	LCB	1,083	1,004	1,165	1	,280	2,955
	Constante	-54,971	11,983	21,046	1	,000	,000

a. Variable(s) introducida(s) en el paso 1: Longitud, AlingBlos50, Blos50Vent3, Blos50Vent5, Blos50Vent7, LCB.

Figura 2. 6: Coeficientes que se toman para construir la ecuación de probabilidad

El siguiente paso de este algoritmo consiste en evaluar en el conjunto de datos podado las ecuaciones de probabilidad obtenidas, con el objetivo de lograr una nueva clasificación. Esta clasificación se logra otorgándole valor 1 *(ortólogo)* a los resultados de las evaluaciones que sean mayores que 0.5 y valor 0 *(no ortólogo)* en caso contrario. Este

procedimiento genera dieciséis nuevas variables: las ocho ecuaciones evaluadas de los cuatro modelos de alineamiento por las dos regresiones logísticas ejecutadas en el SPSS, es decir Enter y Forward, y las ocho nuevas clasificaciones calculadas.

La información obtenida por el procedimiento anterior requiere la mezcla de estos resultados, la cual puede hacerse mediante el promedio de los valores obtenidos por la evaluación de las ecuaciones o utilizando el voto mayoritario. El voto mayoritario clasifica como ortólogos a los pares que pertenecen a la clase 1 en más del 50% de las ocho nuevas clasificaciones y 0 a las que están por debajo de esa cifra. Ya realizada la mezcla se procede es obtener las tablas de contingencias. Una vez que se tienen el número de falsos positivos y negativos que resultan se procede a realizar las comparaciones de los resultados, usando como medida de calidad F-measure.

2.3 Algoritmo basado en una muestra con reducción de la clase mayoritaria

Este algoritmo consiste en aplicar los clasificadores estudiados en muestras conformadas por otros métodos de poda. La descripción de este algoritmo se muestra en la figura 2.7. Al igual que el algoritmo anterior, el primer paso es seleccionar la muestra que se tomará como referencia para luego dividir el conjunto de poda por los cuatro modelos de alineamientos en el SPSS para trabajar de forma independiente por cada modelo.

2.3.1. Filtro por proporción

Consiste en realizar un filtro en la base de casos mediante Weka, este se realiza mediante el método ***SpreadSubSample***, un filtro supervisado con el cual se puede hacer una poda de la base de casos cargada dada una proporción determinada, *distributionSpread*, en este caso se decidió hacer un poda con balance de uno a mil *(1:1000).* Esta vía permite aplicar los clasificadores a una muestra aún desbalanceada pero de menor tamaño, lo cual disminuye el tiempo de ejecución de las operaciones.

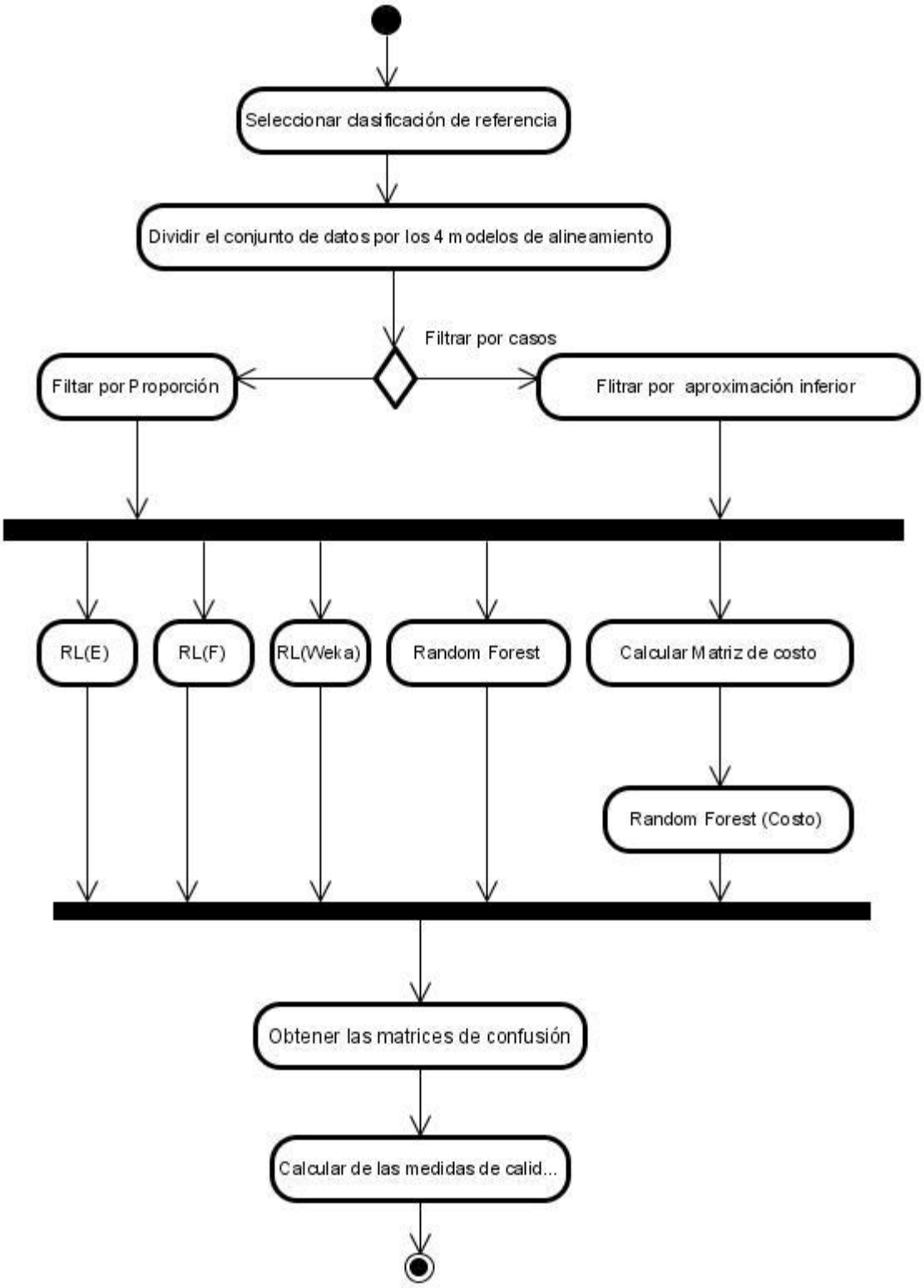


Figura 2. 7: Descripción del algoritmo basado en una muestra con reducción de la clase mayoritaria

2.3.2. Filtro por aproximación

En esta sección se muestra el uso de la teoría de los conjuntos aproximados RST (Pawlak, 1982) para construir un filtro. Se define el sistema de información (Komorowski and Polkowski, 1999) (U, A, Z) donde $U = \{(X_i, Y_j)\}, \forall X_i \in V_1, \forall Y_j \in V_2$ es el universo de pares de genes formados a partir de los conjuntos de genes $V_1 \subseteq G_1$ y $V_2 \subseteq G_2$ de genes no podados del grafo bipartido completo pesado *G(V,E,W)* donde $V = V_1 \cup V_2$, E es el conjunto de arcos que une genes de V_1 con genes de V_2 y el peso de cada arista $w_{ij} = S(X_i, Y_j)$ siendo

$$S(X_i, Y_j) = \frac{1}{4} \times \sum_{k=1}^{4} S_k(X_i, Y_j) \qquad (13)$$
$$i = 1 \dots n; j = 1 \dots m$$

la función de similitud global entre los genes X_i, Y_j calculada a partir de las medidas de similitud $S_1, \dots, S_4$. El conjunto de atributos o rasgos $A = \{S(X_i, Y_j)\}$ está definido por la medida de similitud empleada para ponderar el grafo bipartido y Z es el conjunto de conceptos que se desean evaluar. Los conceptos pueden definirse a partir de las clases resultantes de un proceso de clasificación. El problema que abarca este trabajo es un problema de clasificación por lo que $Z = \{Z^1, Z^0\}$, donde Z^1 es el concepto de los objetos clasificados como ortólogos y Z^0 el concepto de los clasificados como no ortólogos. Cualquier subconjunto Z^i del universo U se puede expresar en términos de estos bloques de forma exacta o aproximada.

Es posible definir una función de similitud entre pares de genes que pertenecen al universo U a partir de la función de similitud entre genes (13)

$$S_p\big((x_a, y_b), (x_c, y_d)\big) = \frac{mín(S(x_a, y_b), S(x_c, y_d))}{máx(S(x_a, y_b), S(x_c, y_d))} \tag{14}$$

Utilizando la extensión de RST que aparece en (Slowinski and Vanderpooten, 1997) se puede definir una relación de similitud R'_B extendida de R_B sin que se requiera la transitividad, ni la simetría, siendo el único requerimiento la reflexividad.

$$R'_B((x_a, y_b)) = \{(x_c, y_d): (x_a = x_c \vee y_b = y_d) \wedge S_p\big((x_a, y_b), (x_c, y_d)\big) \geq \xi\} \tag{15}$$

En esta relación ξ es un valor de umbral para la similitud entre los pares de genes del universo. Este valor de umbral se define como la media de los valores máximos de las similitudes entre cualquier par de objetos de U. (García, 1999)

$$\xi = \frac{1}{n}\sum_{i=1}^{n} \underset{\substack{j=1..n \\ i \neq j}}{máx}\{S_p(O_i, O_j)\}, O_i, O_j \in U, n = |U| \tag{16}$$

La aproximación R-inferior contiene todos los objetos del concepto o clase que se relacionan solamente con los pares del propio concepto, donde Z^0 es el concepto de los pares no ortólogos y Z^1 el concepto de los pares ortólogos.

$$R'_*(Z^i) = \{z \in Z^i: R'_B(z) \subseteq Z^i\} \tag{17}$$

Finalmente, se realiza la poda manteniendo la clase minoritaria y pares de la clase mayoritaria que pertenecen al conjunto de aproximación R-inferior en Z^0.

2.3.3. Aplicación de clasificadores

Para este algoritmo se aplican los clasificadores ya usados: Regresión logística y Random Forest, incluyendo esta vez el Random Forest con costo. Este último requiere del uso de un clasificador sensitivo al costo de Weka, ***CostSensitivityClassifier***, ubicado en el apartado *META* el cual permite calcular una matriz de costo, para aplicárselo posteriormente a un clasificador que en este caso será Random Forest. El valor C(+,-) correspondiente al costo

de los falsos positivos se propone como el cociente de la división entre el total de casos y la cantidad de falsos positivos obtenidos al aplicar Random Forest sin costo. De igual forma para C(-,+) se propone el cociente de la división entre el total y el número de falsos negativos.

Para el modelo de Regresión logística se usarán los métodos de selección ya propuestos: Enter y Forward.

Para el análisis de los resultados obtenidos a partir de las matrices de confusión de cada clasificador aplicado se extraen los valores de la medida F-measure. Con estos datos se realizan las comparaciones concluyentes.

2.4. Conclusiones parciales

En este capítulo se propusieron vías de solución para mejorar las medidas de calidad y disminuir el número de falsos positivos a través de dos algoritmos diseñados: Algoritmo1: Algoritmo basado en el ensamblaje de clasificadores de Regresión logística y Algoritmo2: Algoritmo basado en una muestra con reducción de la clase mayoritaria. Con la ayuda del paquete estadístico SPSS, se aplicaron modelos de Regresión logística para las clasificaciones, además se utilizó el método del pesaje de casos para el Algoritmo1. A través de la herramienta Weka y el SPSS es posible aplicar los filtros y clasificadores supervisados propuestos.

CAPÍTULO 3. EXPERIMENTOS Y RESULTADOS

En este capítulo se muestran los experimentos realizados con datos reales de comparación de dos genomas y se discuten los resultados de la aplicación de los algoritmos expuestos en el capítulo 2.

3.1 Conformación de conjuntos de datos

Para la conformación de los pares de genes se parte de los ficheros FASTA, que contienen una descripción de los genes de cada genoma seleccionado. En la tabla 3.1 se muestra la cantidad de genes que se emplearon para este estudio por cada genoma.

Saccharomyce scerevisiae	5861 genes
Schizosaccharomyce spombe	5006 genes

Tabla 3. 1: Genomas que se emplearon para la conformación de los datos

El conjunto de datos resultante contiene los pares de genes formados de la combinación de todos los genes del genoma *Saccharomyces cerevisiae* con cada uno de los genes del *Schizosaccharomyces pombe*. A estos pares de genes se les calcula la similitud basada en la longitud de secuencias, LCB, los cuatro modelos de alineamiento y la información del perfil físico químico (ver figura 3.1). La diferencia entre los modelos de alineamiento se encuentra en los parámetros empleados para el cálculo de los alineamientos locales y globales (Sánchez, 2012). Se toman como referencia las clasificaciones: Inparanoid, GeneDB y además, la unión de las dos clasificaciones y la intersección de estas dos clasificaciones. En la figura 3.1 se muestra la estructura de la base de casos conformada.

Al obtener el conjunto de datos original la base de casos resulta tener un gran tamaño, al presentar un total de 29,340,166 casos que representan pares de genes y mostrando un alto nivel de desbalance. Con el objetivo de disminuir el tamaño de la base a emplear en la realización de los experimentos se hizo una poda por homología en la cual se extrajeron los pares de genes que presentaron menos del 40% de similitud. Mediante esta poda se

consiguió disminuir el tamaño de la base a 8,095,907 pares de genes, menos de un tercio de la cantidad original. En la tabla 3.2 se muestra el desbalance presentado por ambas muestras.

Gen1	Gen2	Longitud	4 Modelos de alineamientos	LCB	Inparanoid	GeneDB	Unión	Intersección

Figura 3. 1 Estructura de la base de casos

Clasificación	Conjunto de datos original		Conjunto de datos podado (Base_8)	
	No Ortólogos	Ortólogos	No Ortólogos	Ortólogos
INPARANOID7.0	29335077	5089	8091041	4866
GENEDB	29336337	3829	8092183	3724
Clase UNIÓN	29349785	5208	8090950	4957
Clase INTERSECCIÓN	29336456	3710	8092274	3633

Tabla 3. 2: Desbalance de ambas bases

3.2 Aplicación de clasificadores a Base_8

Inicialmente se aplicaron los clasificadores de Regresión Logística y Random Forest a la Base_8, para posteriormente poder comparar sus resultados con los obtenidos al aplicar los algoritmos diseñados. Para estos experimentos se toma como referencia la intersección de la clasificación de Inparanoid y GeneDB.

3.2.1 Regresión logística

El primer clasificador que se aplicó fue la Regresión Logística, experimento que se realizó por los dos métodos de selección explicados en el capítulo anterior: Enter y Forward. Cada clasificador se aplicó independiente por cada modelo de alineamiento.

Al aplicarse estos clasificadores se obtuvo la relación existente entre los falsos positivos y falsos negativos. Como se muestra en la figura 3.2 los modelos de Blosum50 y Blosum62

tienen la mejor relación de falsos positivos, no siendo así en la relación de falsos negativos, como se puede ver en la figura 3.3.

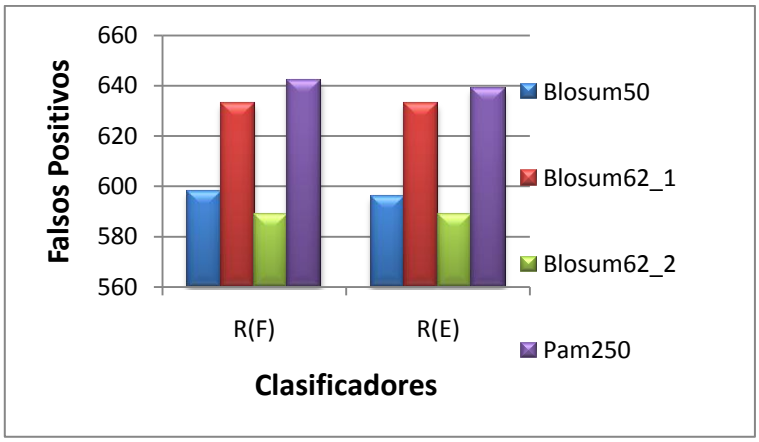


Figura 3. 2: Número de falsos positivos respecto a la clase Intersección

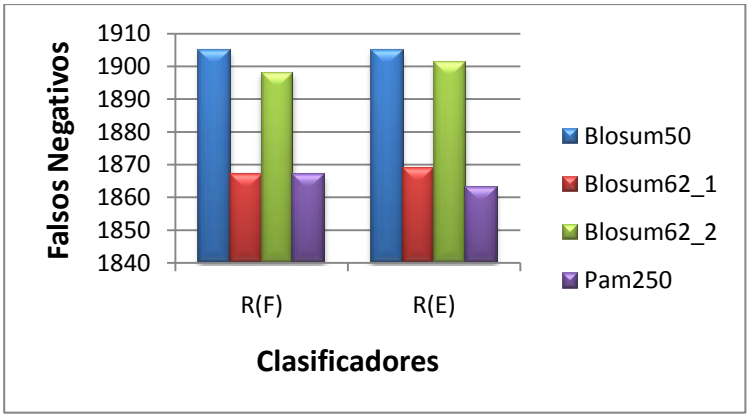


Figura 3. 3: Número de falsos negativos respecto a la clase Intersección

Para evaluar la clasificación se calculó la medida F-measure de las clasificaciones (ver figura 3.4). Como se puede ver los mejores resultados de esta medida se producen para los modelos Blosum62_1 y Pam250 en el método Enter, existiendo diferencia significativa entre los dos métodos empleados para la Regresión Logística.

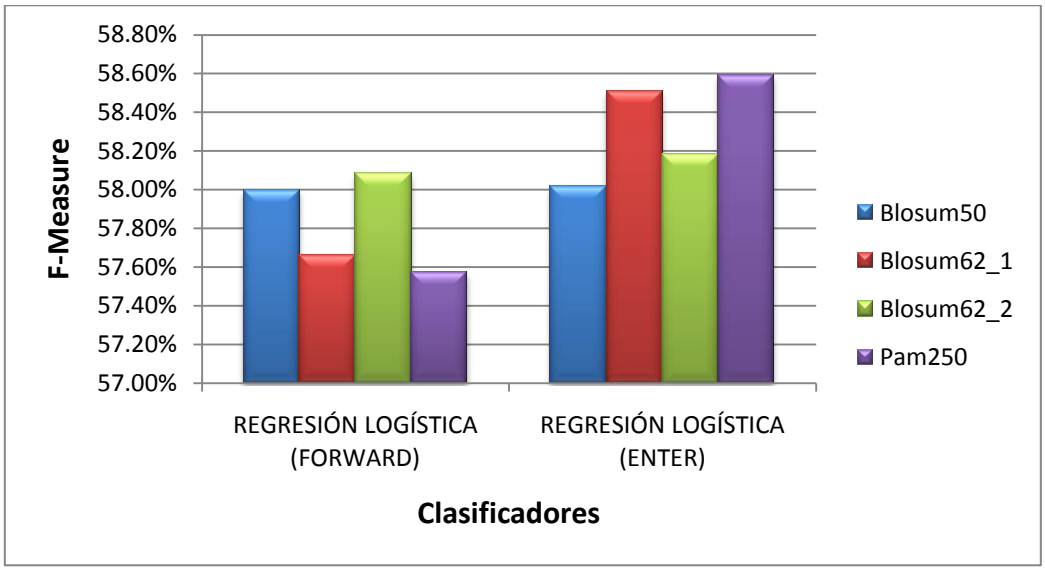


Figura 3. 4: F-Measure de la Regresión logística para la Intersección

3.2.2 Random Forest de WEKA

Usando la herramienta Weka se le aplicó a la Base_8 el clasificador Random Forest, con el cual se obtuvo un resultado relativamente superior en la Regresión Logística, pues el valor de la precisión obtenido en este caso fue superior. Los resultados de este experimento se muestran en el Anexo 1.

Tratando de mejorar aún más el resultado de Random Forest, se usó un clasificador sensitivo al costo, ***CostSensitivityClassifier***, que permite utilizar una matriz de costo. La matriz se calculó por la proporción existente entre el total de casos y los falsos positivos y negativos alcanzados por el propio clasificador sin costo, en este caso Random Forest, (ver tabla 3.3).

0.0	**4658.0**
12830.0	0.0

Tabla 3. 3: Matriz de Costo clase Intersección

Los resultados alcanzados tras esta aplicación mostraron una ligera reducción en el número de falsos positivos y por tanto un aumento en el valor de precisión (ver figura 3.5). Se puede decir entonces que el clasificador Random Forest con costo fue el que mejor resultados reflejó para la Base_8.

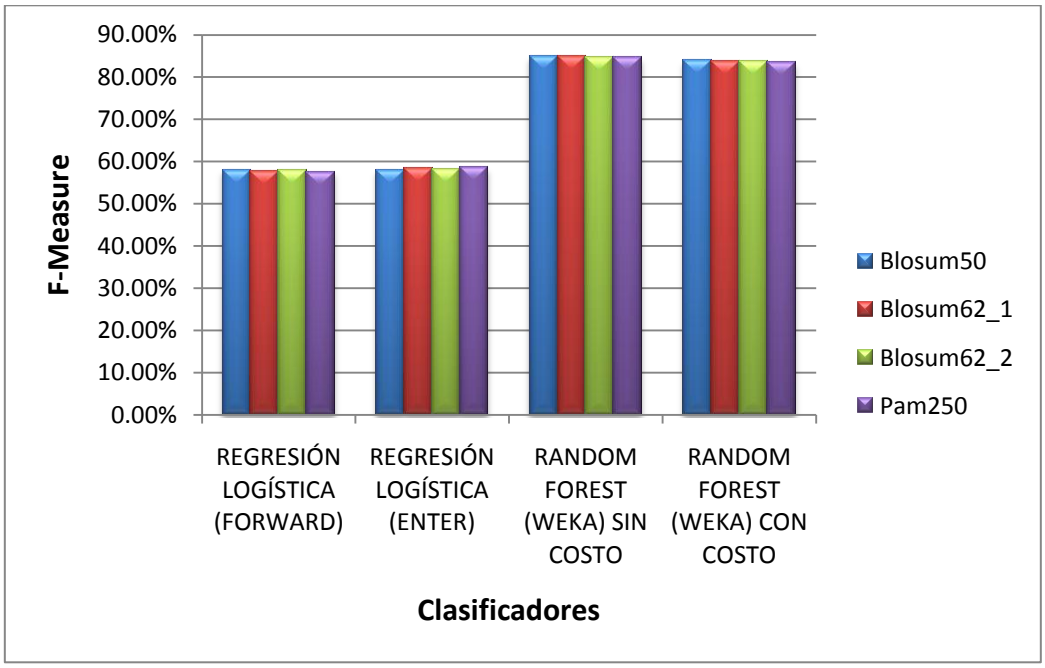


Figura 3. 5: Resultados de los experimentos a la Base_8

3.3 Algoritmo basado en el ensamblaje de clasificadores de regresión logística

En este epígrafe se muestran los resultados obtenidos en el algoritmo basado en el ensamblaje de clasificadores de regresión logística para la detección de genes ortólogos con manejo del desbalance.

Inicialmente se aplicó un filtro en el SPSS para generar diez muestras aleatorias balanceadas que son empleadas en los cuatro modelos de alineamiento para realizar los experimentos con los clasificadores mencionados en el capítulo anterior. En la tabla 3.4 se muestra el balance de las clases obtenido en cada muestra para cada una de las clasificaciones de referencia y en la tabla 3.5 sus identificaciones para este estudio, donde *X* representa el número que les corresponde.

Muestra	Inparanoid		GeneDB		Intersección		Unión	
	0	1	0	1	0	1	0	1
1	4798	4866	3752	3724	4010	3633	4859	4957
2	4818	4866	3769	3724	4013	3633	4897	4957
3	4771	4866	3744	3724	3878	3633	4947	4957
4	4906	4866	3693	3724	3961	3633	4864	4957
5	4782	4866	3692	3724	4098	3633	4866	4957
6	4875	4866	3704	3724	4018	3633	4981	4957
7	4825	4866	3813	3724	3948	3633	4853	4957
8	4758	4866	3740	3724	3914	3633	4950	4957
9	4990	4866	3601	3724	3952	3633	4906	4957
10	4897	4866	3653	3724	3998	3633	4825	4957

Tabla 3. 4: Relación de las muestras aleatorias balanceadas generadas

Muestra	Descripción
InpB50_X	**Clase Inparanoid con Blosum50**
InpB62_1_X	**Clase Inparanoid con Blosum62_1**
InpB62_2_X	**Clase Inparanoid con Blosum62_2**
InpP250_X	**Clase Inparanoid con Pam250**
GenB50_X	**Clase GeneDB con Blosum50**
GenB62_1_X	**Clase GeneDB con Blosum62_1**
GenB62_2_X	**Clase GeneDB con Blosum62_2**
GenP250_X	**Clase GeneDB con Pam250**
UniB50_X	**Clase Unión con Blosum50**
UniB62_1_X	**Clase Unión con Blosum62_1**
UniB62_2_X	**Clase Unión con Blosum62_2**
UniP250_X	**Clase Unión con Pam250**
IntB50_X	**Clase Intersección con Blosum50**
IntB62_1_X	**Clase Intersección con Blosum62_1**
IntB62_2_X	**Clase Intersección con Blosum62_2**
IntP250_X	**Clase Intersección con Pam250**

Tabla 3. 5: Identificación de las muestras generadas

A estas muestras se le aplicaron los modelos de Regresión Logística, con los dos métodos de selección, Enter y Forward. También se le aplicaron los clasificadores Random Forest y Bagging. En el Anexo 2 se exhiben los resultados de estos clasificadores agrupados por clase en cada uno de los modelos de alineamiento.

En la figura 3.6 se muestran los resultados de los clasificadores para las muestras en el modelo Blosum50 tomando la clasificación de la intersección. Como se puede observar la mejor muestra para los clasificadores es la muestra 1, teniendo un mejor comportamiento en el clasificador Random Forest.

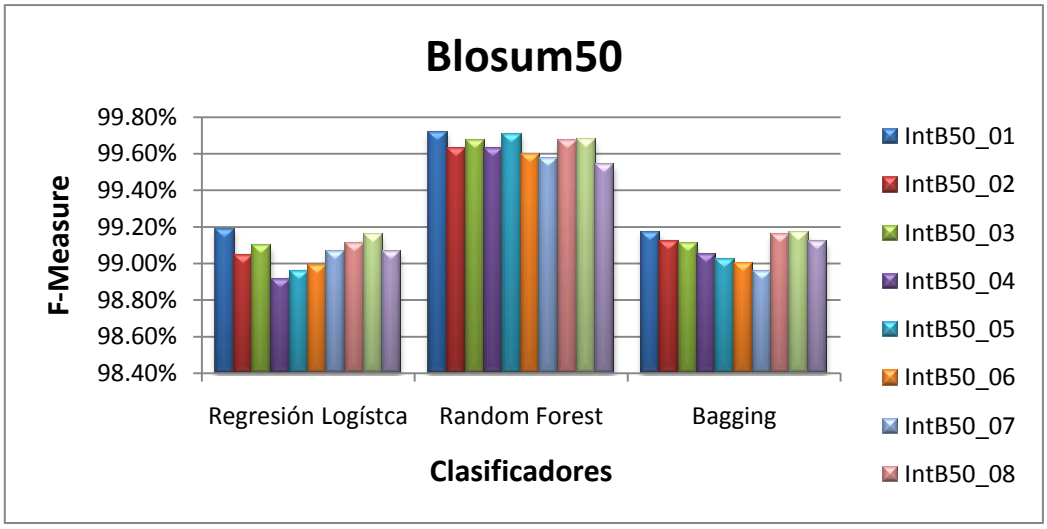


Figura 3. 6: Valores de F-measure de los clasificadores aplicados a las muestras

Aunque los resultados obtenidos de la aplicación de los clasificadores son muy buenos, las muestras resultan ser muy pequeñas en comparación con la Base_8, razón por la cual los resultados que reflejan los clasificadores no pueden ser tomados como referencia para llegar a conclusiones.

Se hizo una selección de la mejor muestra por los cuatro modelos de alineamiento, atendiendo a los mejores valores de precisión y F-measure. Como ya se mencionó anteriormente la muestra con mejores resultados fue la 1 (ver tabla 3.6).

Una vez que se tienen las muestras de referencia se extrajeron los coeficientes alcanzados por los modelos de regresión aplicados para conformar las ecuaciones de probabilidad las cuales fueron evaluadas en la Base_8, estas se pueden observar en la tabla 3.7.

Int 01			R.L. (E)		R. L. (F)		R. L. (Weka)		Random F.		Bagging	
	0	***1***	***Prec.***	***F-M.***	***Prec.***	***F-M.***	***Prec.***	***F-M.***	***Prec.***	***F-M.***	***Prec.***	***F-M.***
Blosum50	4798	4866	99.84	0.992	99.87	0.993	98.87	0.992	99.73	0.997	98.43	0.992
Blosum62_1	4798	4866	99.86	0.993	99.86	0.992	98.57	0.992	99.58	0.997	98.55	0.993
Blosum62_2	4798	4866	99.81	0.991	99.81	0.991	97.83	0.99	99.7	0.998	97.76	0.99
Pam250	4798	4866	99.81	0.994	99.79	0.994	98.76	0.994	99.48	0.997	98.8	0.994

Tabla 3. 6: Resultados de la mejor muestra

IntB50_01				IntB62_2_01			
ENTER		**FORWARD**		**ENTER**		**FORWARD**	
Constant	-49.066	**Constant**	-51.533	**Constant**	-54.980	**Constant**	-55.403
Longitud	35.037	**Longitud**	38.106	**Longitud**	40.583	**Longitud**	41.504
AlingBlos50	2363.312	**AlingBlos50**	2340.500	**AlignBlos622**	2427.993	**AlignBlos622**	2421.950
Blos50Vent3	7.192	**Blos50Vent3**	10.668	**Blos622Vent3**	7.913	**Blos622Vent3**	13.029
Blos50Vent5	1.256			**Blos622Vent5**	2.982		
Blos50Vent7	3.295			**Blos622Vent7**	3.152		
LCB	1.419			**LCB**	0.629		
IntB62_1_01				**IntP250_01**			
ENTER		**FORWARD**		**ENTER**		**FORWARD**	
Constant	-38.231	**Constant**	-39.126	**Constant**	-50.130	**Constant**	-51.653
Longitud	24.635	**Longitud**	25.461	**Longitud**	35.489	**Longitud**	37.187
AlignBlos621	1999.844	**AlignBlos621**	2005.193	**AlignPam250**	1682.132	**AlignPam250**	1669.375
Blos621Vent3	5.817	**Blos621Vent3**	6.988	**Pam250Vent3**	12.165	**Pam250Vent3**	14.360
Blos621Vent5	2.096	**Blos621Vent7**	5.387	**Pam250Vent5**	3.511		
Blos621Vent7	4.008			**Pam250Vent7**	-1.412		
LCB	0.934			**LCB**	1.441		

Tabla 3. 7: Coeficientes para el cálculo de la ecuación de probabilidad

Ya evaluadas las funciones se procede a la mezcla de los resultados, que como ya se abordó en el capítulo anterior puede ser por promedio o mediante la realización de un voto mayoritario. Una vez obtenidas las funciones se procede a extraer las tablas de contingencias para ver los resultados.

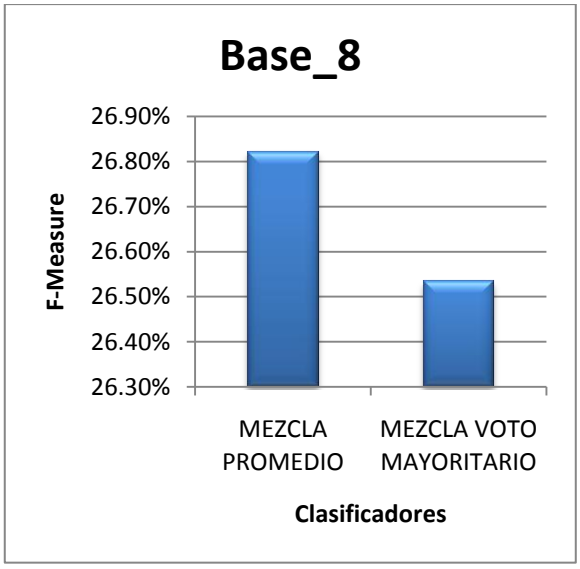


Figura 3. 7: Comparación de las mezclas

3.4 Algoritmo basado en una muestra con reducción de la clase mayoritaria

Este algoritmo consiste en aplicar los clasificadores a dos tipos de muestras conformadas a través de filtros para disminuir el tamaño de la base a utilizar, las cuales se describen en los siguientes subepígrafes.

3.4.1 Filtro por proporción

Para realizar esta poda proporcionada se aplicó el filtro ***SpreadSubSample*** de la herramienta Weka con una proporción 1:1000, con lo cual, tomando como referencia la clase Intersección, se obtiene una muestra con un desbalance de un par ortólogo por cada mil no ortólogos (ver tabla 3.8).

Base_3.6		
Total	**Clase 0**	**Clase 1**
3636633	3633000	3633

Tabla 3. 8: Desbalance de la Base_3.6

Aplicación de clasificadores

A la Base_3.6 se aplicaron los clasificadores propuestos, Regresión logística por los métodos Enter y Forward, además de Random Forest, todos para la clasificación Inparanoid. Además se empleó una variante de este último clasificador sensitivo al costo en Weka, cuya matriz de costo se muestra en la figura 3.8. Cada clasificador se aplicó igualmente a cada modelo de alineamiento.

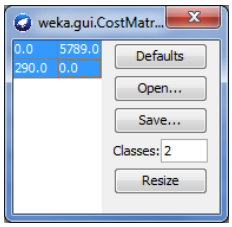


Figura 3. 8: Matriz de costo

Los resultados se aprecian en el Anexo 3. En la gráfica de la figura 3.9 se muestra la comparación entre los porcientos de precisión de los clasificadores aplicados.

Según consta en las gráficas anteriores los porcientos de F-measure que se obtienen son superiores para el clasificador Random Forest sin costo, con una precisión de más del 88%, el mejor resultado obtenido hasta el momento.

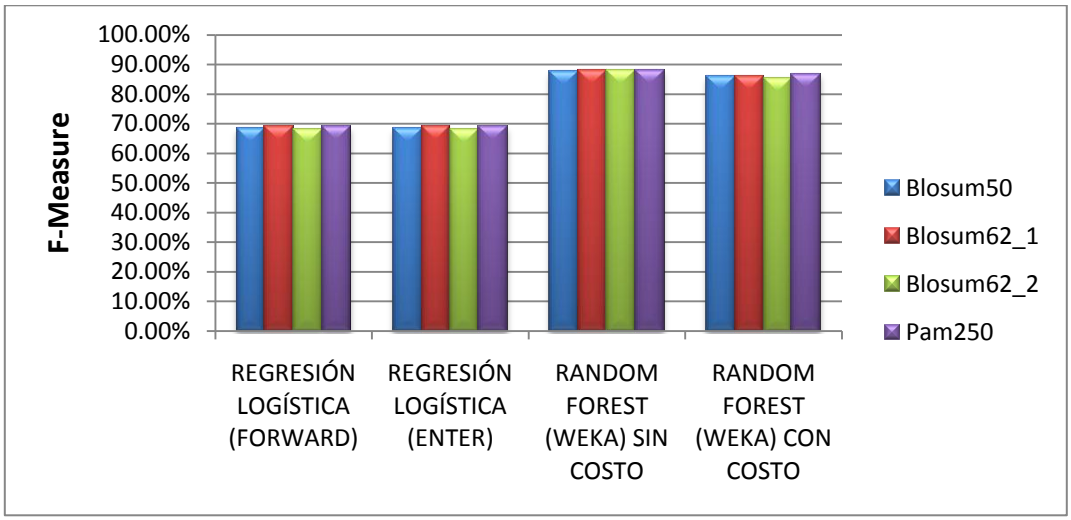


Figura 3. 9: Comparación de los clasificadores de la Base_3.6

3.4.2 Filtro por aproximación

El cálculo del filtro se hizo según el procedimiento explicado en la sección 2.3.2. El conjunto de datos resultante se muestra en la tabla 3.9.

Base_RST		
Total	**Clase 0**	**Clase 1**
20088	16455	3633

Tabla 3. 9: Desbalance de la BaseRST

Aplicación de clasificadores

Al igual que en el Filtro por Proporción, a la BaseRST se aplicaron los clasificadores, Regresión logística por los métodos Enter y Forward, Random Forest y la variante de este clasificador sensitivo al costo.

Los resultados en este experimento mostraron una precisión inferior en comparación con los alcanzados con la Base_3.6 como se aprecia en el Anexo 4. En la figura 3.10 se muestran los resultados obtenidos por los clasificadores respecto a la clase Intersección.

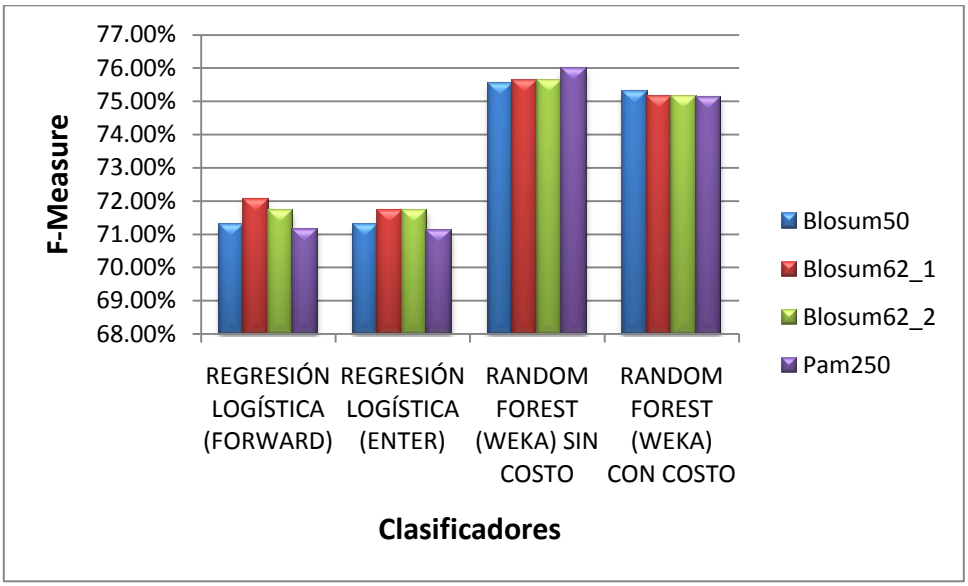


Figura 3. 10: Resultados de los clasificadores a la Base_RST.

En este caso el clasificador que obtuvo la mejor precisión fue Random Forest, con poco más de un 78%.

3.5 Comparación de resultados

Una vez obtenidos los resultados de los algoritmos diseñados, se hizo una comparación entre los mejores resultados de cada conjunto de datos: Base_3.6, BaseRST, Base_8 y el resultado de la mezcla promedio del Algoritmo1. Se hizo una selección del experimento que mejor resultado alcanzó por el conjunto (ver figura 3.11).

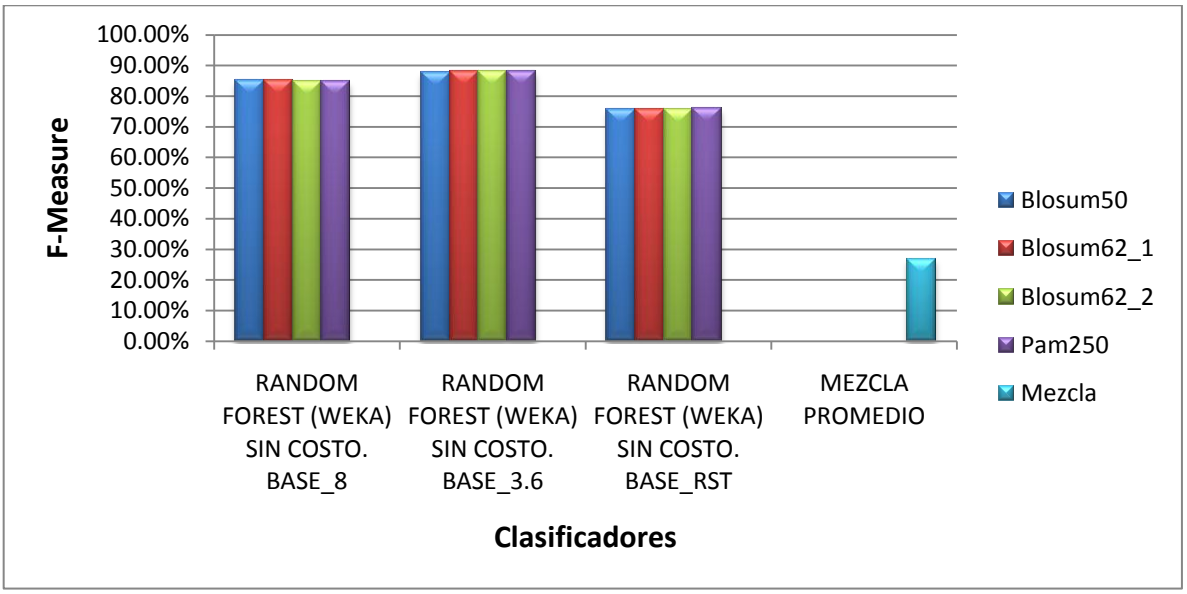


Figura 3. 11: Comparación de resultados de los clasificadores

Tras analizar los porcientos de F_measure se puede concluir que el Algoritmo2 basado en una muestra con reducción de la clase mayoritaria refleja los mejores resultados, específicamente en la muestra con proporción con el clasificador Random Forest sin costo.

3.6 Conclusiones Parciales

Los resultados obtenidos en la base podada corroboran los efectos del desbalance para los clasificadores de Regresión logística y Random Forest. La mezcla de modelos del Algoritmo1 debe ser mejorada debido a que los clasificadores de base evaluados en la muestra de prueba producen bajos niveles de precisión. El Algoritmo2 muestra los mejores resultados para los clasificadores Random Forest, que a su vez son los mejores resultados con relación a los clasificadores aplicados a la Base_8, incluyendo los sensitivos al costo.

CONCLUSIONES

1. El problema de la detección de ortólogos puede ser representado como un problema de clasificación supervisado utilizando los datos de clasificaciones conocidas y las posibles soluciones requieren el tratamiento del marcado desbalance en conjuntos de datos de gran tamaño.
2. Los enfoques consultados de manejo de desbalance desde la perspectiva de los datos y de los algoritmos pueden ser combinados para conformar nuevas propuestas de solución.
3. El preprocesamiento inicial de los datos es necesario para reducir el tamaño de los conjuntos de datos sin perder información relevante para la clasificación.
4. Tanto el Weka como el SPSS resultaron paquetes de gran utilidad para la implementación de los algoritmos propuestos.
5. La validación de los algoritmos mediante el cálculo de la medida F-measure permitió comparar las propuestas realizadas en este trabajo.
6. El algoritmo basado en una muestra con reducción de la clase mayoritaria muestra los mejores resultados para los clasificadores Random Forest, que a su vez son los mejores resultados con relación a los clasificadores aplicados a la Base_8, incluyendo los sensitivos al costo.

RECOMENDACIONES

1. Proponer nuevas variantes de filtros y de tratamiento de los pesos.
2. Diseñar otros filtros a partir de los conjuntos aproximados.
3. Proponer nuevas variantes para el cálculo de los costos para algoritmos sensitivos al costo.
4. Proponer nuevas medidas de calidad y sus variantes pesadas a partir de las distintas clasificaciones que se recopilen.
5. Comparar los resultados de los algoritmos supervisados con los de algoritmos no supervisados.
6. Diseñar nuevas estrategias de mezcla de clasificadores.
7. Incluir en las comparaciones los resultados de las clasificaciones utilizando el conjunto de datos pesados.

REFERENCIAS BIBLIOGRÁFICAS

ALTSCHUL, S. F., GISH, W., MILLER, W., MYERS, W. & LIPMAN, D. J. 1990. Basic local alignment search tool. *Journal Molecular Biology,* 215**,** 403-410.

BREIMAN, L. 1994. Bagging predictors. *Technical Report*

BREIMAN, L. 2001. Random Forest. *Technical Report*

BRITO, C. C. 2012. *Estudio de simulación en la detección de genes ortólogos.* Diploma, Universidad Central "Marta Abreu" de Las Villas.

CHAWLA, N. V., JAPKOWICZ, N. & LCZ, A. K. 2004. Editorial: Special Issue on Learning from Imbalanced Data Sets. *Sigkdd Explorations,* 6**,** 1-6.

CHEN, C., LIAW, A. & BREIMAN, L. 2004. Using Random Forest to Learn Imbalanced Data.

CHEN, F., MACKEY, A. J., JR., C. J. S. & ROOS, D. S. 2006. OrthoMCL-DB: querying a comprehensive multi-species collection of ortholog groups. *Nucleic Acids Research,* 34**,** 363-368.

CHEN, M.-C., CHEN, L.-S., HSU, C.-C. & ZENG, W.-R. 2008. An information granulation based data mining approach for classifying imbalanced data. *Information Sciences,* 178**,** 3214–3227.

DARLING, A. C. E., MAU, B. & BLATTNER, F. R. 2004. MAUVE: Multiple Alignment of Conserved Genomic Sequence With Rearrangements. *Genome Res.,* 14**,** 1394-1403

DEZA, E. 2006. *Dictionary of Distances*, Elsevier.

DUCH, W. 2000. Similarity-based methods: a general framework for classification, approximation and association. *Control and Cybernetics,* 29**,** 1-30.

EDGAR, R. C. 2009. Optimizing substitution matrix choice and gap parameters for sequence alignment. *BMC Bioinformatics,* 10.

ESTOPIÑALES, M. 2009. *Herramientas Computacionales para la Comparación de Genomas.* Trabajo de diploma, Universidad Central "Marta Abreu" de las Villas.

FERNÁNDEZ, M. A. 2012. *Herramientas computacionales para la comparación de genomas y detección de genes ortólogos con un enfoque de grafo bipartido.* Tesis de Maestría, Universidad Central "Marta Abreu" de las Villas.

GALPERT, D., FERNÁNDEZ, M. A., COMPANIONI, C. & MILLO, R. 2012. A local-global gene comparison for ortholog detection in two closely related eukaryotes species. *Investigación de Operaciones*.

GARCÍA, M. M. 1999. Monografía de reconocimiento de patrones. Santa Clara: Universidad Central "Marta Abreu" de Las Villas.

GENEDB. 2013. Available: http://old.genedb.org/genedb/pombe/index.jsp [Accessed 2013].

HAGELSIEB, G. M. & LATIMER, K. 2008. Choosing BLAST options for better detection of orthologs as reciprocal best hits. *Bioinformatics,* 24**,** 319-324.

HANLEY, J. B. M. 1983. A method of comparing the areas under receiver operating characteristic curves derived from the same cases. *Radiology,* 48**,** 839–843.

JINFU LIU, Q. H., DAREN YU 2008. A comparative study on rough set based class imbalance learning. *Knowledge-Based Systems,* 21**,** 753-763.

KAMVYSSELIS, M. K. 2003. *Computational comparative genomics: genes, regulation, evolution.* Doctor of Philosophy in Computer Science, Massachusetts Institute of Technology

KOCH, E. N., COSTANZO, M., BELLAY, J., DESHPANDE, R., CHATFIELD-REED, K., CHUA, G., D'URSO, G., ANDREWS, B. J., BOONE, C. & MYERS, C. L. 2012. Conserved rules govern genetic interaction degree across species. *Genome Biology,* 13.

KOMOROWSKI, J., Z. & POLKOWSKI, P. A. L. 1999. Rough sets: a tutorial, in Rough-Fuzzy Hybridization: A New Trend in Decision Making. *In:* EDITED BY S. K. PAL Y A. SKOWRON, S., SPRINGER-VERLANG, 1999 (ed.).

LACHENBRUSH, P. & MICKEY, M. 1968. Estimating error rates in discriminant analysis. *Technometrics,* 10**,** 167-178.

LE, G. H. N. 2011. *Machine Learning with Informative Samples for Large and Imbalanced Datasets.* Doctor of Philosophy, University of Wollongong.

LEE, Y., SULTANA, R., PERTEA, G. & CHO, J. 2002. Cross-referencing eukaryotic genomes: TIGR Orthologous Gene Alignments (TOGA). *Genome Research,* 12**,** 493-502.

LI, L., STOECKERT, C. J. & ROOS, D. S. 2003. OrthoMCL: Identification of Ortholog Groups for Eukaryotic Genomes. *Genome Research,* 13**,** 2178–2189.

LINARD, B., THOMPSON, J. D., POCH, O. & LECOMPTE, O. 2011. OrthoInspector: comprehensive orthology analysis and visual exploration. *BMC Bioinformatics,* 12**,** 1471-2105.

LIU, J., HU, Q. & YU, D. 2008. A weighted rough set based method developed for class imbalance learning. *Information Sciences,* 178**,** 1235–1256.

MOUNT, D. W. 2004. *Bioinformatics Sequence and Genome Analysis*, CSHL Press.

O'BRIEN, K. P., REMM, M. & SONNHAMMER, E. L. L. 2005. Inparanoid: a comprehensive database of eukaryotic orthologs. *Nucleic Acids Research,* 33**,** D476-D480.

ÖSTLUND, G., SCHMITT, T., FORSLUND, K., KÖSTLER, T., MESSINA, D. N., ROOPRA, S., FRINGS, O. & SONNHAMMER, E. L. L. 2010. InParanoid 7: new

algorithms and tools for eukaryotic orthology analysis. *Nucleic Acids Research,* 38**,** D196–D203.

OVERBEEK, R., FONSTEIN, M., D'SOUZA, M., PUSCH, G. D. & MALTSEV, N. Year. The use of gene clusters to infer functional coupling. *In:* Proceedings of the National Academy of Sciences of the United States of America, 1999. 2896–2901.

OWNA, H. S. & ABRAHAMB, A. 2012. A new weighted rough set framework based classification for Egyptian NeoNatal Jaundice. *Applied Soft Computing* 12**,** 999–1005.

OZER, H. G., C.J., ZHANG, F. & Y., B. 2004. Clustering of Eukaryotic Orthologs Based on Sequence and Domain Similarities Using the Markov Graph-Flow Algorithm. *Advances in Bioinformatics and its Applications*.

PADMAJA, T. M., KRISHNA, P. R. & BAPI, R. S. 2009. Majority Filter-based Minority Prediction (MFMP): An Approach for Unbalanced Datasets.

PAWLAK, Z. 1982. Rough sets. *International Journal of Computer and Information Sciences,* 11**,** 341-356.

REMM, M., STORM, C. E. V. & SONNHAMMER, E. L. L. 2001. Automatic Clustering of Orthologs and In-paralogs from Pairwise Species Comparisons. *Jorunal Molecular Biology,* 314**,** 1041-1052.

SÁNCHEZ, R. M. 2012. *Aplicación de medidas de similitud y algoritmos de agrupamiento a la detección de genes ortólogos.* Diploma, Universidad Central "Marta Abreu" de Las Villas.

SLOWINSKI, R. & VANDERPOOTEN, D. 1997. Similarity relation as a basis for rough approximations. *In:* WANG, E. B. P. P. (ed.).

TATUSOV, R. L., FEDOROVA, N. D., JACKSON, J. D., JACOBS, A. R., KIRYUTIN, B. & KOONIN, E. V. 2003. The COG database: an updated version includes eukaryotes. *BMC Bioinformatics,* 4**,** 1-14.

TATUSOV, R. L., KOONIN, E. V. & LIPMAN, D. J. 1997. A genomic perspective on protein families. *Science,* 278.

TOWFIC, D., GREENLEE, M. H. W. & HONAVAR, V. 2009. Detection of Gene Orthology Based On Protein-Protein Interaction Networks. *IEEE International Conference on Bioinformatics and Biomedicine.* Washington DC.

VAN DONGEN, S. M. 2000. *Graph clustering by flow simulation.*

VASHIST, A., KULIKOWSKI, C. & MUCHNIK, I. 2005. Screening for Ortholog Clusters Using Multipartite Graph Clustering by Quasi-Concave Set Function Optimization. *D. ´Slezak et al. (Eds.): RSFDGrC 2005, LNAI Springer-Verlag Berlin Heidelberg,* 3642**,** 409–419.

WEBBER, C. A. P. & CHRIS, P. 2004. Genes and Homology. *Current Biology,* 14.

YANG, P., LIU, W., ZHOU, B. B., CHAWLA, S. & Y.ZOMAYA, A. 2013. Ensemble-based wrapper methods for feature selection and class imbalance learning.

ANEXOS

Anexo 1: Resultados obtenidos de los clasificadores aplicados a la Base_8 tomando como referencia la clasificación Intersección.

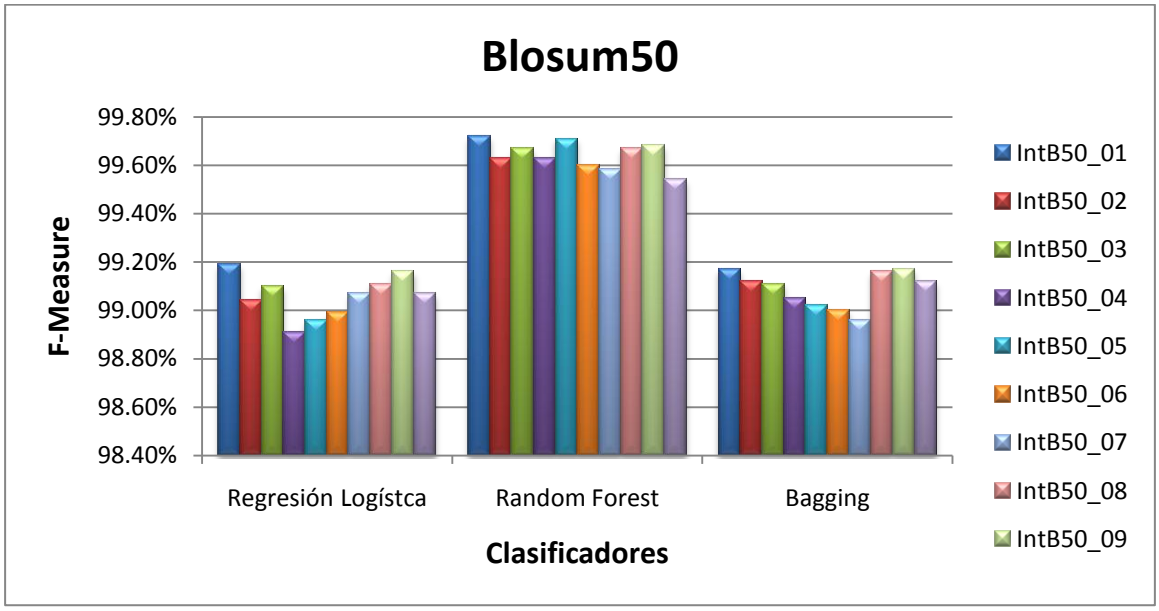
REGRESIÓN LOGÍSTICA (FORWARD)									
8095907	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Precisión**	**Recall**	**F-Measure**
Blosum50	3633	8092274	8091676	598	1728	1905	0.743	0.476	0.580
Blosum62_1	3633	8092274	8091641	633	1728	1905	0.732	0.476	0.577
Blosum62_2	3633	8092274	8091685	589	1728	1905	0.746	0.476	0.581
Pam250	3633	8092274	8091632	642	1728	1905	0.729	0.476	0.576

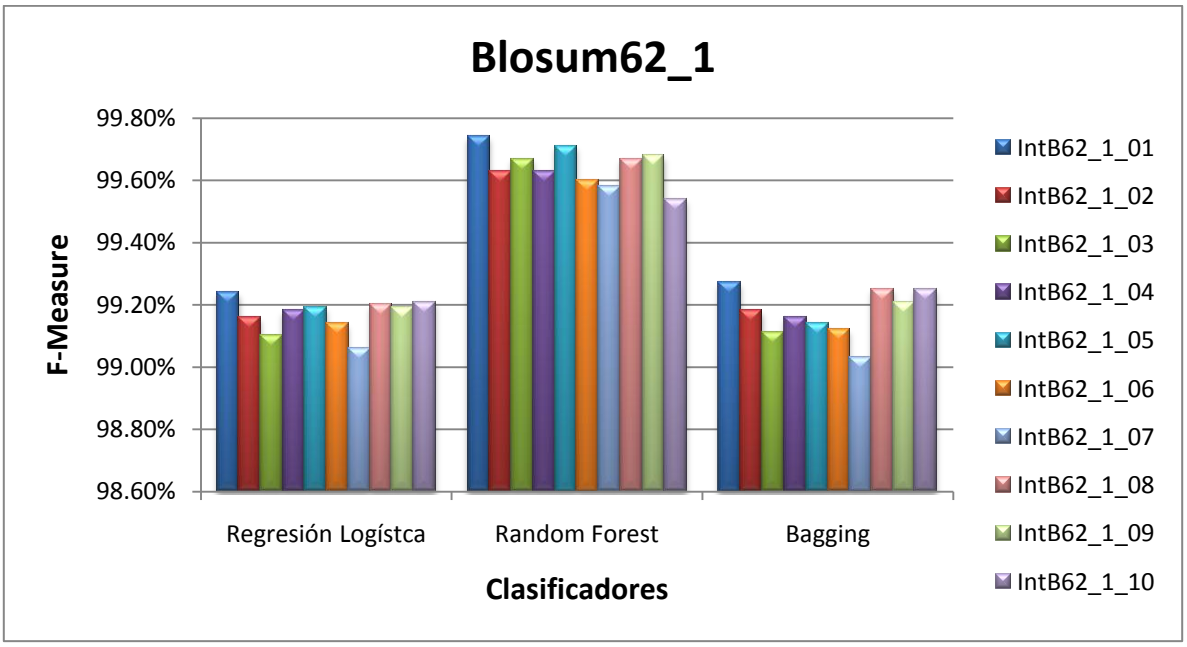
REGRESIÓN LOGÍSTICA (ENTER)									
8095907	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Precisión**	**Recall**	**F-Measure**
Blosum50	3633	8092274	8091678	596	1728	1905	0.744	0.476	0.580
Blosum62_1	3633	8092274	8091641	633	1764	1869	0.736	0.486	0.585
Blosum62_2	3633	8092274	8091685	589	1732	1901	0.746	0.477	0.582
Pam250	3633	8092274	8091635	639	1770	1863	0.735	0.487	0.586

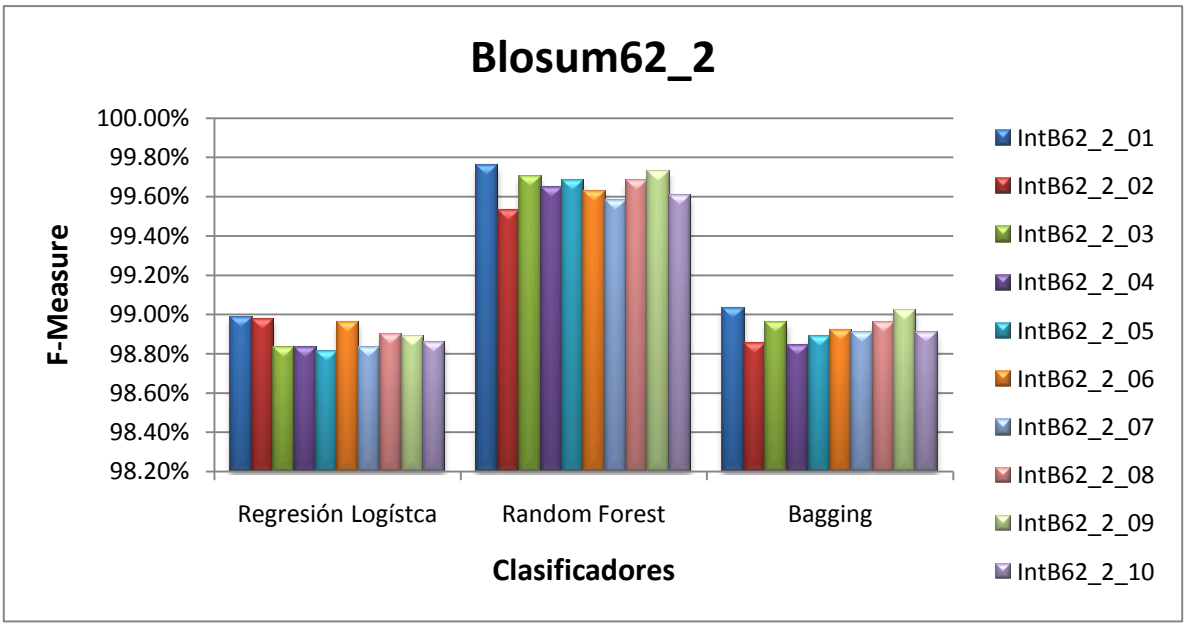
RANDOM FOREST (WEKA) SIN COSTO									
8095907	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Precisión**	**Recall**	**F-Measure**
Blosum50	3633	8092274	8091405	869	3326	307	0.793	0.915	0.850
Blosum62_1	3633	8092274	8091410	864	3316	317	0.793	0.913	0.849
Blosum62_2	3633	8092274	8091413	861	3299	334	0.793	0.908	0.847
Pam250	3633	8092274	8091434	840	3291	342	0.797	0.906	0.848

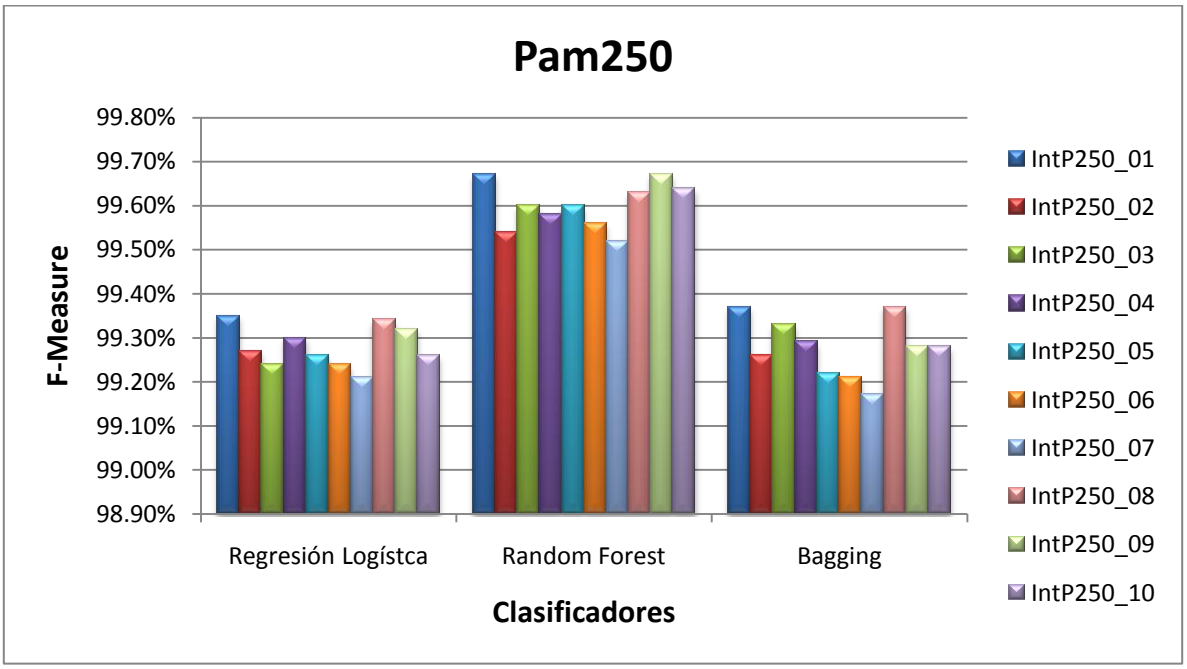
RANDOM FOREST (WEKA) CON COSTO									
8095907	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Precisión**	**Recall**	**F-Measure**
Blosum50	3633	8092274	8091553	721	3141	492	0.813	0.865	0.838
Blosum62_1	3633	8092274	8091568	706	3117	516	0.815	0.858	0.836
Blosum62_2	3633	8092274	8091552	722	3128	505	0.812	0.861	0.836
Pam250	3633	8092274	8091570	704	3100	533	0.815	0.853	0.834

Anexo 2: Resultados obtenidos de los clasificadores aplicados a las muestras balanceadas.






Blosum62_2
100.00%
99.80%
99.60%
99.40%
99.20%
99.00%
98.80%
98.60%
98.40%
98.20%
F-Measure
Regresión Logístca
Random Forest
Bagging
Clasificadores
IntB62_2_01
IntB62_2_02
IntB62_2_03
IntB62_2_04
IntB62_2_05
IntB62_2_06
IntB62_2_07
IntB62_2_08
IntB62_2_09
IntB62_2_10


Pam250
99.80%
99.70%
99.60%
99.50%
99.40%
99.30%
99.20%
99.10%
99.00%
98.90%
F-Measure
Regresión Logístca
Random Forest
Bagging
Clasificadores
IntP250_01
IntP250_02
IntP250_03
IntP250_04
IntP250_05
IntP250_06
IntP250_07
IntP250_08
IntP250_09
IntP250_10

Anexo 3: Resultados obtenidos de los clasificadores aplicados a la Base_3.6 tomando como referencia la clasificación Intersección.

REGRESIÓN LOGÍSTICA (FORWARD)									
3636633	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**% Prec.**	**Recall**	**F-Measure**
Blosum50	3633	3633000	3632553	447	2117	1516	0.826	0.583	0.683
Blosum62_1	3633	3633000	3632541	459	2165	1468	0.825	0.596	0.692
Blosum62_2	3633	3633000	3632562	438	2098	1535	0.827	0.577	0.680
Pam250	3633	3633000	3632533	467	2169	1464	0.823	0.597	0.692

REGRESIÓN LOGÍSTICA (ENTER)									
3636633	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**% Prec.**	**Recall**	**F-Measure**
Blosum50	3633	3633000	3632554	446	2117	1516	0.826	0.583	0.683
Blosum62_1	3633	3633000	3632541	459	2166	1467	0.825	0.596	0.692
Blosum62_2	3633	3633000	3632561	439	2097	1536	0.827	0.577	0.680
Pam250	3633	3633000	3632533	467	2169	1464	0.823	0.597	0.692

RANDOM FOREST (WEKA) SIN COSTO									
3636633	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**% Prec.**	**Recall**	**F-Measure**
Blosum50	3633	3633000	3632384	616	3319	314	0.843	0.914	0.877
Blosum62_1	3633	3633000	3632416	584	3315	318	0.850	0.912	0.880
Blosum62_2	3633	3633000	3632402	598	3329	304	0.848	0.916	0.881
Pam250	3633	3633000	3632438	562	3308	325	0.855	0.911	0.882

RANDOM FOREST (WEKA) CON COSTO									
3636633	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**% Prec.**	**Recall**	**F-Measure**
Blosum50	3633	3633000	3632617	383	3031	602	0.888	0.834	0.860
Blosum62_1	3633	3633000	3632638	362	3015	618	0.893	0.830	0.860
Blosum62_2	3633	3633000	3632621	379	2999	634	0.888	0.825	0.856
Pam250	3633	3633000	3632664	336	3039	594	0.900	0.836	0.867

Anexo 4: Resultados obtenidos de los clasificadores aplicados a la Base_RST tomando como referencia la clasificación Intersección.

REGRESIÓN LOGÍSTICA (FORWARD)									
20088	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Prec.**	**Recall**	**F-Measure**
Blosum50	3633	16455	15779	676	2386	1247	0.779	0.657	0.713
Blosum62_1	3633	16455	15772	683	2429	1204	0.781	0.669	0.720
Blosum62_2	3633	16455	15788	667	2404	1229	0.783	0.662	0.717
Pam250	3633	16455	15769	686	2384	1249	0.777	0.656	0.711

REGRESIÓN LOGÍSTICA (ENTER)									
20088	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Prec.**	**Recall**	**F-Measure**
Blosum50	3633	16455	15779	676	2387	1246	0.779	0.657	0.713
Blosum62_1	3633	16455	15769	686	2432	1201	0.780	0.669	0.720
Blosum62_2	3633	16455	15788	667	2404	1229	0.783	0.662	0.717
Pam250	3633	16455	15770	685	2382	1251	0.777	0.656	0.711

RANDOM FOREST (WEKA) SIN COSTO									
20088	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Prec.**	**Recall**	**F-Measure**
Blosum50	3633	16455	15616	839	2714	919	0.764	0.747	0.755
Blosum62_1	3633	16455	15626	829	2706	927	0.765	0.745	0.755
Blosum62_2	3633	16455	15604	851	2727	906	0.762	0.751	0.756
Pam250	3633	16455	15662	793	2712	921	0.774	0.746	0.760

RANDOM FOREST (WEKA) CON COSTO									
20088	**Ortólogos**	**No Ortólogos**	**TN**	**FP**	**TP**	**FN**	**Prec.**	**Recall**	**F-Measure**
Blosum50	3633	16455	15677	778	2664	969	0.774	0.733	0.753
Blosum62_1	3633	16455	15672	783	2636	997	0.771	0.726	0.748
Blosum62_2	3633	16455	15625	830	2687	946	0.764	0.740	0.752
Pam250	3633	16455	15697	758	2641	992	0.777	0.727	0.751