Clasificación de Series de Tiempo Astronómicas

Muriel Pérez
201011755

11 de mayo de 2015

Índice general

1. Introducción **5**

2. El Problema del Aprendizaje **9**
2.1. El Clasificador de Bayes y Consistencia 11
2.2. Estimación del Error 14
2.3. Clasificadores 15
2.3.1. k Vecinos Más Cercanos 15
2.3.2. Máquinas de Soporte Vectorial 16
2.3.3. Árboles de Clasificación y Regresión 25
2.3.4. Bosques Aleatorios 30

3. Clasificación **33**
3.1. El conjunto de Datos 35
3.2. Características Seleccionadas 39
3.3. Clasificación 47
3.3.1. Árboles de Clasificación 47
3.3.2. Bosques Aleatorios 47
3.3.3. k Vecinos Más Cercanos 47
3.3.4. Máquinas de Soporte Vectorial 47

Capítulo 1

Introducción

Con los avances en técnicas de observación astronómica que han sucedido en los últimos años, hay grandes cantidades de datos disponibles. Por ejemplo se espera que el *VISTA Variables in the Via Lactea* (VVV) del *European Southern Observatory* (ESO) produzca del orden de 10^9 curvas de luz[1] de fuentes puntuales en el infrarojo cercano con hasta 100 observaciones en diferentes épocas de alta calidad. De la misma forma estudios como la misión Kepler de la *National Aeronautics and Space Administration* (NASA), cuyo objetivo principal es la detección de exoplanetas, tienen como subproductos gran cantidad de curvas de luz.

Para que estos datos sean útiles para la comunidad científica es necesario clasificarlos y extraer sus características. Aunque los métodos automáticos muchas veces no pueden igualar la inspección manual por parte de un experto, la cantidad de datos disponible hace que esta tarea no sea posible en corto tiempo y hace necesario utilizar técnicas de minería de datos. Este interés se manifiesta en proyectos como el *VVV Templates Project* que tiene como objetivo consolidar una base bien definida de curvas de luz de estrellas variables en el infrarojo cercano para ser utilzadas como referencia para la clasificación automática de curvas de luz.

Las curvas de luz no pueden ser analizadas con técnicas de análisis de series de tiempo porque, debido a limitaciones en el tiempo de observación, fallas técnicas, periodos de mantenimiento de los instrumentos utilizados y

[1] La curva de luz de una estrella es el resultado de medir su magnitud como función del tiempo. La magnitud de una estrella es el flujo de energía observado en una parte del espectro electromagnético (una banda), delimitada por un filtro, en escala logarítmica (ver el capítulo 4 de [16]).

la imposibilidad de abservar todas las regiones del cielo durante todo el año, las curvas de luz no constan del mismo número de observaciones y éstas no son hechas en intervalos regulares por lo que el tiempo durante el cual cada estrella no es observada es impredecible y algunas características importantes de las curvas de luz no son observadas.

Dependiendo de la serie de magnitudes observadas, una estrella puede ser clasificadas como variable o no variable; periódicas o no periódicas; y en diferentes clases de variabilidad estelar que depende de la morfología de su curva de luz. La forma de la curva de luz depende de las condiciones físicas de la estrella por lo que conocer a qué tipo de variabilidad pertenece cada estrella es de vital importancia para el estudio de las estrellas variables. A su vez, el estudio de las estrellas variables ha sido importante para el estudio de la evolución estelar, la determinación de distancias cósmicas y la búsqueda de exoplanetas, entre otras.

En estudios previos [12, 29, 24] se le ha asignado a cada curva de luz un vector, llamado vector de características, y, basado en él, se ha hecho la clasificación automática. Este proceso consiste en entrenar un clasificador basado en una muestra clasificada previamente, la muestra de entrenamiento, utilizando el vector de características escogido. La escogencia de el vector de características es crucial para el proceso de clasificación porque con él se debe poder clasificar cada curva de luz, es decir, debe logar que, en el espacio de características, las clases se superpongan lo menos posible. Para la conformación de este vector se han elegido coeficientes de Fourier de la curva de luz[12, 29, 24], que son calculados mediante métodos como el periodograma de Lomb-Scargle [30] o la minimización de la entropía de Shannon de la gráfica de la curva [9].

Esta elección de caracterísicas no es del todo conveniente porque requiere de gran poder computacional y limita el tipo de objetos que pueden ser clasificados. El cálculo del periodogramas como el de Lomb-Scargle para curvas de luz, y en general el de los métodos utilizados en la literatura, requiere de intentar una gran cantidad de periodos candidatos a ser el periodo de la curva de luz para luego elegir el mejor y de la inspección manual de las curvas de luz. Los periodos de los objetos observados varía entre desde unos pocos minutos y varios años por lo cual se requiere probar una gran cantidad de periodos. Por un lado este es un proceso es computacionalmente intensivo, lo que limita su uso en conjuntos grandes de curvas de luz; y por otro lado no es seguro que dé como resultado el periodo real de una curva de luz, por lo que a menudo éste debe ser revisado manualmente. Además el resultado de

la clasificación puede ser sensible a la calidad de las curvas de luz que sean elegidas como muestra de entrenamiento [12] y limita el estudio a fuentes periódicas.

En [25, 27], los autores notaron que algunas variables descriptivas de la serie de magnitudes de una curva de luz (como su sesgo o su curtosis) sirven para clasificar ciertos tipos de estrellas con clasificadores lineales. En este trabajo retomamos esa idea y construimos un vector de características basadas en variables tomadas de estadística descriptiva. El uso de este tipo de variables tiene las ventajas de que puede ser calculadas de manera rápida y da como resultado un vector de caracterísicas que sive para realizar clasificación con una taza de éxito alta. Para evaluar esta aproximación al problema utilizamos una parte del Catálogo de Estrellas Variables de la tercera fase del *Optical Gravitational Lensing Experiment* (OGLE III)[33, 38, 40, 36, 35, 34, 44, 43, 41, 21, 15, 22, 37, 42, 39] que contiene curvas de luz de estrellas previamente clasificadas en seis tipos de variabilidad estelar y curvas de luz de estrellas candidatas a ser clasificadas como Be (ver cuadro 3.1).

En este trabajo utilizamos k-vecinos más cercanos, árboles de clasificación, máquinas de soporte vectorial y bosques aleatorios para realizar la clasificación automática de las curvas de luz basada en nuestra elección de características. Asímismo, estimamos la probabilidad de que una nueva curva de luz sea clasificada correctamente por uno de estos clasificadores utilzando validación cruzada de 10 iteraciones. Estos clasificadores fueron elegidos porque son aproximaciones muy distintas al problema de clasificación, por su naturaleza no lineal y no paramétrica; y por el hecho de que han mostrado ser efectivos en gran cantidad de aplicaciones prácticas. Para todo el análisis utilzamos el paquete estadístico de fuente abierta R [23]. Para cada tarea utilizamos paquetes específicos que son referenciados a lo largo del documento.

Este documento está organizado de la siguiente forma. En el capítulo 3.1 damos una descripción del conjunto de datos utilizado en este trabajo. En el capítulo 3 abordamos el problema de clasificación de manera informal, presentamos y discutimos la elección de atributos y evaluamos el desempeño de los clasificadores mediante validación cruzada de 10 iteraciones. En los apéndices abordamos formalmente el problema de aprendizaje en general y damos una descripción de cada uno de los algorítmos utilizados en el trabajo.

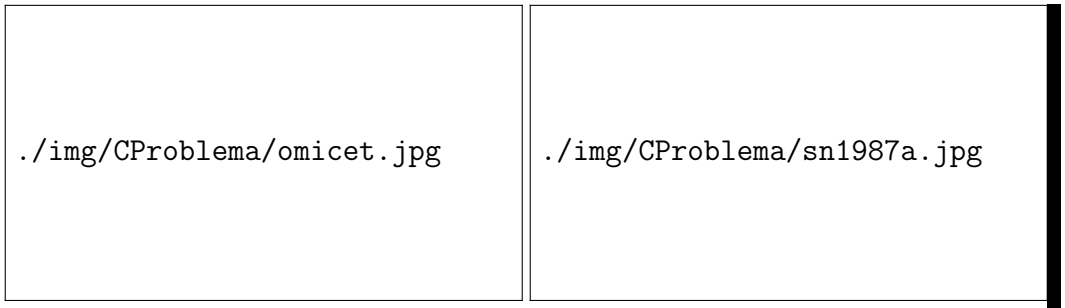
Capítulo 2

El Problema del Aprendizaje

Aquí diré cómo está organizado el capítulo.

Las personas reconocemos con facilidad las letras en manuscritos, las caras de otras personas, las palabras que alguien nos dice o el estado de la comida basado en su olor. La capacidad de agrupar los estímulos que recibimos en categorías, por ejemplo el olor de la comida en buen o mal estado,y la capacidad para actuar en respuesta a ellos ha sido de vital importancia para nuestra supervivencia . Por ello hemos desarrollado complejos sistemas para llevar a cabo estas tareas.

Con la popularización de computadores electrónicos, la construcción máquinas que aprendan de la experiencia ha sido objeto de estudio. La habilidad de crear estas máquinas tiene una importancia estratégica puesto que existen tareas que no pueden ser llevadas a cabo utilizando técnicas de porgramación clásicas porque no existe un modelo matemático para ellas. En el caso de la clasificaciónd ecurvas de luz, por la forma en que se hacen las observaciones y el hecho de que la identificación de una curva de luz se hace con base en su forma, es difícil hacer un modelo matemático que capture estas diferencias. A pesar de esto existe gran gantidad de ejemplos de curvas de luz disponibles, por lo que es natural preguntarse si se puede entrenar un computador para identificar estas diferencias de la misma forma en que una persona puede ser entrenada para reconocerlas. En la figura 2.1 se observan dos curvas de luz, una pulsante y una eruptiva, que pueden ser distinguidas utilizando únicamente esta información. La pregunta de si es posible entrenar un sistema basado en datos disponibles puede ser hecha para otras tareas, como el reconocimiento de textos en manuscritos, la detección e identificación de caras y objetos en imágenes o la identificación de genes en secuencias de ADN.



(a) Curva de luz de una estrella Mira (b) Curva de luz de la Supernova 1987A

Figura 2.1: Las estrellas pueden ser clasificadas en grupos basado en la forma de sus curvas de luz. Estas clasificación puede ser hecha con base en la forma de las curvas de luz, sin embargo es difícil crear un modelo matemático que capture estas diferencias. Imágenes tomadas de [8]

El reconocimiento de patrones es una disciplina científica cuya meta es la clasificación de objetos en clases. Existen situaciones en las cuales existe una gran cantidad de objetos previamente clasificados en clases predefinidas y la tarea es encontrar, o aproximar lo mejor posible, la dependencia funcional entre objetos y clases. Podemos precisar esto de la siguiente forma. Llamemos al espacio de los objetos que queremos clasificar Xy $\{1, \ldots, M\}$ es el conjunto de las posibles clases a las que pueden pertenecer los elementosd e X. En el caso de la clasificación de curvas de luz, X consta de todas las curvas de luz y $\{1, \ldots, M\}$ representa los posibles tipos de variabilidad estelar. Contamos con una muestra aleatoria de tamaño N, llamada muestra de entrenamiento, $\mathcal{L} = \{(x_i, j_i), \ldots, (x_N, j_N)\}$ con $x_l \in X$ y $j_l \in \{1, \ldots, M\}$, es decir, una muestra de X previamente clasificada. Nuestra tarea es entonces proponer una función $g : X \to \{1, \ldots, M\}$ a partir de la información contenida en $\mathcal{L}$ que representa nuestra predicción de la clase a la que pertenece cada elemento de X. La función g se llama clasificador y, para un elemento $x \in X$ cuya clase j es desconocida, el clasificador falla si $g(x) \neq j$.

El espacio X puede ser complejo o no estar matemáticamente bien definido, por lo cual con frecuencia se representan los objetos con vectores, llamados de características, en $\mathbb{R}^n$. Por ejemplo si queremos realizar detección de rostros, X consiste de todos los posibles rostros, por lo que es más conveniente representar cada rostro con un conjunto de números como la se-

paración de los ojos, el ángulo que forma las líneas que unen los ojos con la barbilla, etcétera; lo mismo sucede con las curvas de luz, por lo que representamos cada una con un vector. Estos vectores de características pueden, en principio, ser una combinación de variables continuas, discretas y categóricas, sin embargo esto no afecta en gran medida la teoría. Así las cosas, la elección de un clasificador puede ser una función $g : \mathbb{R}^n \to \{1, \ldots, M\}$.

Se debe utilizar un marco probabilístico para modelar la dependencia entre características y clases. Puede suceder que dos observaciones con un mismo vector de características pertenezcan a clases diferentes. Esto puede suceder en escenarios en los que pertenencia a una u otra clase no so sea completamente explicada por diferencias en los vectores de características, o porque la dependencia real entre características y clases sea no determinista. En este orden de ideas suponemos que existe una medida de probabilidad P sobre $\mathbb{R}^n \times \{1, \ldots, M\}$ tal que $P(\vec{x}, j)$ es la probabilidad de observar un vector de características $\vec{x} \in \mathbb{R}^n$ cuyo objeto representado pertenece a clase j. Así definimos la probabilidad de error del clasificador g, $P_e(g)$, como

$$P_e(g) = P(g(\vec{x}) \neq j). \tag{2.1}$$

Surge entonces la pregunta de qué tan bueno puede ser un clasificador. El mejor clasificador posible es llamado el clasificador de Bayes.

2.1. El Clasificador de Bayes y Consistencia

Decimos que un clasificador $g^* : \mathbb{R}^n \to \{1, \ldots, M\}$ es de Bayes si minimiza la probabilidad de error, es decir, que si g es otro clasificador entonces

$$P_e(g^*) \leq P_e(g). \tag{2.2}$$

Llamaremos P_e^* a $P_e(g^*)$.

En el caso de que existan densidades condicionales f_j tales que para cada $A \subset \mathbb{R}^n$ medible se cumple

$$P(A|j) = \int_A f_j(\vec{x}) d\vec{x} \tag{2.3}$$

podemos dar una expresión explícita para el clasificador de Bayes. Para un

clasificador g podemos escribir

$$\begin{aligned} P_e(g) &= 1 - P(g(\vec{x}) = j) \\ &= 1 - \sum_{j=1}^{M} P(g(\vec{x}) = j|j)P(j) \\ &= 1 - \sum_{j=1}^{M} \left(\int_{\{g(\vec{x})=j\}} f_j(\vec{x}) d\vec{x} \right) P(j) \\ &= 1 - \int \sum_{j=1}^{M} \chi_{\{g(\vec{x})=j\}} f_j(\vec{x}) P(j) d\vec{x}. \end{aligned} \tag{2.4}$$

Donde $P(j)$ es la probabilidad *a priori* de encontrar un objeto de clase j y χ_A es la función indicadora del conjunto A. Ahora, para cada $\vec{x}$

$$\sum_{j=1}^{M} \chi_{\{g(\vec{x})=j\}} f_j(\vec{x}) P(j) \leq \max_j [f_j(\vec{x}) P(j)] \tag{2.5}$$

entonces

$$P_e(g) \geq 1 - \int \max_j [f_j(\vec{x}) P(j)] d\vec{x}. \tag{2.6}$$

Como la desigualdad 2.5 es igualdad cuando g le asigna a cada $\vec{x}$ la clase j para la cual $f_j(\vec{x})P(j)$ es máximo, podemos concluir que éste es el clasificador de Bayes, es decir,

$$g^*(\vec{x}) = \arg\max_{j \in \{1,\ldots,M\}} f_j(\vec{x}) P(j) \tag{2.7}$$

y

$$P_e^* = 1 - \int \max_j [f_j(\vec{x}) P(j)] d\vec{x}. \tag{2.8}$$

g^* es el estimador de máxima verosimilitud para j y le asigna a cada $\vec{x}$ la clase que hace que la observación $\vec{x}$ sea más probable.

El hecho de que este clasificador sea el mejor posible nos muestra la importancia de elegir un vector de características de forma tal que las clases, es decir, sus distribuciones marginales f_j se superpongan lo menos posible en el espacio de características. Aunque esto es intuitivamente obvio, esto nos da un argumento para afirmarlo. $f_j(\vec{x})P(j)$ puede ser interpretado como la probabilidad de que el vector $\vec{x}$ sea generado por la clase j. Si dos clases generan puntos en una región del espacio n-dimensional con probabilidad

parecida, al clasificar un punto en esa región se esperará una probabilidad de error alta. En el caso extremo en que se le asigna a cada objeto un vector constante, el clasificador 2.7 se reduce a escoger la clase que sea más probable *a priori*.

En realidad rara vez se conocen las distribuciones marginales f_j y frecuentemente no se conocen la probabilidades $P(j)$. Las probabilidades *a priori* pueden ser dadas por el analísta o estimadas a partir de los datos si la muestra es representativa de X. Algunos métodos intentan estimar estas distribuciones marginales con funciones $\tilde{f}_j$ y utilizar el clasificador

$$g(\vec{x}) = \underset{j\in\{1,\ldots,M\}}{\arg\max}\, \tilde{f}_j(\vec{x})P(j) \tag{2.9}$$

por ejemplo suponiendo alguna forma para las distribuciones f_j o utilizando métodos no paramétricos de estimación de densidades como estimación por núcleos. La estimación de densidades por núcleos ha probado ser útil porque se sabe que el estimador $\tilde{f}_j$ de la densidad f_j converge en media cuadrática, es decir,

$$\int \left(f(\vec{x}) - \tilde{f}(\vec{x})\right)^2 d\vec{x} \to 0 \text{ cuando } N \to \infty \tag{2.10}$$

sin embargo en dimensión d se ha demostrado que estos convergen a una velocidad de $O(N^{1/(4+d)})$por lo cual no son prácticos en dimensiones altas. revisar y cita

En la práctica, elegimos un clasificador entre una familia $\mathcal{H}$ de clasificadores posibles, llamados hipótesis, mediante un algorítmo de aprendizaje. Por ejemplo un árbol de clasificación es un árbol de decisón binario con funciones sencillas de decisión en cada nodo; existe un algorítmo para escoger un árbol entre todos los árboles binarios posibles posibles. Una manera natural de elegir este clasificador es minimizar la probabilidad de error empírica

$$\hat{P}_e(g) = \frac{1}{N}\sum_{i=1}^{N} \chi_{\{g(\vec{x}_i)\neq j_i\}}, \tag{2.11}$$

es decir, elegir el clasificador

$$g^*_{\mathcal{H},N} = \underset{g\in\mathcal{H}}{\arg\min}\, \hat{P}_e(g). \tag{2.12}$$

para el cual el error de clasificación es

$$P^*_{\mathcal{H},N} = \underset{g\in\mathcal{H}}{\min}\, \hat{P}_e(g). \tag{2.13}$$

Se dice que un clasificador elegido con el criterio 2.12 es consistente si $P^*_{\mathcal{H},N} \to P^*_e$ cuando $N \to \infty$. Surge entonces la pregunta de bajo qué condiciones un clasificador es consistente. Es posible demostrar que las regla de decisión de k vecinos más cercanos [13], bosques aleatorios [2] y máquinas de soporte vectorial [45] son constistentes. La idea utilizar la minimización del error empírico para elegir clasificadores fue desarrollada por Vapnik y Chervonenkis . Para saber más, es posble consultar [13].

(citas están en p187 del Devroye, PONER)

En aplicaciones no se conoce la probabilidad mínima de error P^*_e por lo que no es posible saber en términos absolutos qué tan bueno es un clasificador en un problema específico. Además de encontrar clasificadores que logren probabilidades tan bajas como sea posible, también es necesario implementar algorítmos de aprendizaje que sean eficientes y cuyos tiempos de ejecución no crezcan demasiado rápido con el tamaño de la muestra. Para juzgar un clasificador g es necesario estimar la su probabilidad de error $P_e(g)$.

2.2. Estimación del Error

Uno de los métodos más populares para estimar la probabilidad de clasificación correcta de la función de decisión entrenada por un algorítmo de aprendizaje es validación cruzada de v iteraciones. Se divide la muestra $\mathcal{L}$ en v muestras de prueba $\mathcal{L}_k$, $k = 1, \ldots, v$ con el mismo número de elementos (o lo más próximo posible) y se define la k-ésima muestra de entrenamiento como $\mathcal{L}^k = \mathcal{L} \setminus \mathcal{L}_k$. Utilizando cada una de las v muestras de entrenamiento $\mathcal{L}^k$ se puede entrenar una regla de decisión utilizando el algorítmo de aprendizaje en cuestión. Con ella se clasifican los elementos de la muestra de prueba $\mathcal{L}^k$ y se calcula N^k_{ij} el número de elementos de de la clase j clasificado como i. Sea $N_{ij} = \sum_k N^k_{ij}$ el número total de elmentos de la clase j clasificado como i. Es posible estimar la probabilidad de que un elemento de la clase j sea clasificado como i, $P^{VC}(g(\vec{x}) = i|j)$, con N_{ij}/N_j, donde N_j es el número de elementos pretenecientes a la clase j en la muestra $\mathcal{L}$. Intuitivamente, si la muestra es grande tendremos aproximadamente el mismo poder para clasificar con la muestra completa que con una fracción $\frac{v-1}{v}$ de ella, por lo cual P^{VC} será una buena aproximación a la probabilidad real de clasificación. El valor $v = 10$ es popular en la literatura aunque esta elección es un poco arbitraria. Una elección de v grande da estiamdos menos pesimistas de la probabildiad de error, sin embargo aumentar v aumenta el costo computacional del estimado, por lo que estos dos apectos deben ser balanceados.

A continuación exponemos algunos clasificadores.

2.3. Clasificadores

2.3.1. k Vecinos Más Cercanos

K vecinos más cercanos (knn por sus iniciales en inglés) fue propuesto por Fix y Hodges en [14], y luego republicado en [32]. Se basa en el principio de que los ejemplos de una misma clase se encuentran cerca y que es posible clasificar uno basado en la observación de la clase de sus vecinos más cercanos[1]. Dado un entero k fijo, esta regla le asigna a cada punto de $\mathbb{R}^n$ la clase a la que pertencen la mayoría de los k elementos más cercanos a $\vec{x}$ entre los elementos de la muestra $\{\vec{x}_1, \ldots, \vec{x}_N\}$, esto es,

$$g^{knn}(x) = j \text{ tal que } \sum_{i=1}^{N} w_{i,N}\chi_{\{j_i=j\}} > \sum_{i=1}^{N} w_{i,N}\chi_{\{j_i=k\}} \forall k \neq j \tag{2.14}$$

donde $w_{i,N} = 1/k$ si $\vec{x}_i$ está entre los k vecinos más cercanos de $\vec{x}$ y es 0 de lo contrario. $w_{i,N}$ es llamado peso. Es posible demostrar que knn es consistente en el caso en que $k/N \to 0$ cuando $N \to \infty$ para cualquier elección de pesos $w_{i,N}$ siempre y cuando no permita la clasificación en una clase con minoría numérica [13].

En primera aproximación, se puede implementar un algorítmo que clasifica un punto $\vec{x}$ en un tiempo $O(Nk)$, sin embargo es posible crear estructuras de datos que hacen esta búsqueda más eficiente. Por ejemplo un COVER TREE (traducir)[1] es una estructura de datos que ocupa $O(N)$ espacio, puede ser construida en un tiempo de $O(N \log N)$ y permite realizar búsquedas en tiempo $O(\log N)$ y ha mostrado aumentar la velocidad de las búsquedas entre uno y varios órdenes de magnitud con respecto a utilizar el algorítmo de fuerza bruta (Beygelzimer). Esta estructura se encuentra implementada en el paquete FNN (cite) para R.

Existen algunas variaciones de knn como knn empaquetado[2] [7, 4] y esquemas de pesos óptimos [28]. El proceso de empaquetado fue propuesto por [7, 4] y consiste en que, dado un punto $\vec{x}$, este es clasificado usando la regla de la mayoría entre n clasificadores de knn que utilizan submuestras de

[1] Dime con quién andas y te diré quién eres

[2] En inglés *bagged knn*. *Bagging* es una abreviación de *bootstrap aggregating*.

tamaño $m < N$ tomadas de la muestra original con reemplazo. Este procedimiento ha mostrado aumentar la precisión de los clasificadores [4]. Por otra parte, puede demostrarse que la elección de pesos dada en la ecuación 2.14 es asintóticamente óptima, sin embargo hay situaciones en las que una elección de pesos puede mejorar la precisión. Por ejemplo [28] dio pesos óptimos para el caso en que la dimensión de los datos es 4.

2.3.2. Máquinas de Soporte Vectorial

Las Máquinas de Soporte Vectorial (MSV) son sistemas de clasificación que utilizan con conjunto de hipótesis las funciones lineales en un espacio de dimensión alta. Las MSV como se conocen hoy en día fueron propuestas por Cortes y Vapnik en [10] basado en utilizar núcleos para para encontrar un plano que maximice la distancia a los puntos de la muestra, o equivalentemente, de margen maximal. Daremos una discusión sobre clasificadores lineales y clasificadores lineales de margen maximal, luego mencionamos los conceptos de optimización convexa que son la base de la utilización de núcleos para MSV y finalmente referenciamos el algorítmo de Minimización Secuencial Óptima. Seguimos el texto de Cristianini y Shawe-Taylor [11] en el que se puede encontrar una introducción completa a MSV y la teoría de optimización convexa necesaria para MSV. Para consultas sobre temas adicionales sobre optimización convexa, se puede consultar el libro de Boyd, Vandenberghe y Lieven [3].

Las MSV son clasificadores binarios, es decir, solo pueden distinguir entre dos clases. Para casos en los que se necesita clasificar más de dos clases, es posible implementar esquemas de votación que serán expuestos más adelante. Primero describimos los clasificadores lineales en $\mathbb{R}^n$ en casos en los que se puede escoger un plano que separe perfectamente las dos clases y luego extendemos estos métodos para casos en lo que esto no es posible. Supongamos que se tiene un problema de clasificación binario donde el conjunto de las clases bosibles es $\{-1, 1\}$. Consideremos clasificadores lineales de la forma

$$g(\vec{x}) = signo(\langle \vec{w}, \vec{x} \rangle + b). \tag{2.15}$$

Una muestra de entrenamiento de tamaño N $\mathcal{L} = \{(\vec{x}_1, j_1), \ldots, (\vec{x}_N, j_N)\}$ es linealmente separable si existe un plano definido por $\langle \vec{w}, \vec{x} \rangle + b = 0$ tal que

$$j_i \left(\langle \vec{w}, \vec{x}_i \rangle + b \right) > 0, i = 1, \ldots, N. \tag{2.16}$$

Nos referiremos al plano $\langle \vec{w}, \vec{x} \rangle + b = 0$ con la notación abreviada $(\vec{w}, b)$. $\vec{w}$ es el vector perpendicular al plano y la condición 2.16 corresponde a que los elementos de la clase 1 quedan ubicados en el semiespacio definido por $\langle \vec{w}, \vec{x} \rangle + b > 0$, mientras que los pertenecientes a la clase -1 quedan en el semiespacio $\langle \vec{w}, \vec{x} \rangle + b < 0$. Este plano puede no estar bien definido, en el sentido de que puede haber más de un plano separador para un conjunto de datos, por lo que es necesario definir una noción de lo que significa elegir "el mejor" plano.

Podemos elegir el plano maximizando la distancia entre el plano y los ejemplos, por lo que es necesario definir la noción de margen. Definimos el margen funcional γ_i de un ejemplo $(\vec{x}_i, j_i)$ con respecto al hiperplano $(\vec{w}, b)$ como

$$\gamma_i = j_i(\langle \vec{w}, \vec{x}_i \rangle + b). \tag{2.17}$$

Un ejemplo $(\vec{x}_i, j_i)$ es clasificado de manera correcta por el plano $(\vec{w}, b)$ si $\gamma_i > 0$ y nos referimos al mínimo de los márgenes funcionales de la muestra como el margen funcional de un plano. También podemos tomar el margen con respecto plano normalizado $(\frac{\vec{w}}{\|\vec{w}\|}, \frac{b}{\|\vec{w}\|})$ en cuyo caso el margen funcional mide la distancia euclidiana entre el punto $\vec{x}_i$ y el plano. En esta situación nos referiremos al margen como margen geométrico.

Como $(\lambda\vec{w}, \lambda b)$ define el mismo plano que $(\vec{w}, b)$ para todo $\lambda \neq 0$, tenemos un grado de libertad para elegir el plano separador. Podemos llevar a cabo un truco para expresar el margen funcional del plano en términos de la norma de $\vec{w}$ y así traducir el problema de encontrar el plano a un problema de optimización. Por tener un grado de libertad, podemos maximizar el margen geométrico manteniende el margen funcional fijo e igual a 1. En este caso, si $\vec{w}$ es el vector que alcanza un margen funcional de 1 en el punto positivo$\vec{x}^+$ y -1 en el punto negativo $\vec{x}^-$, podemos calcular su margen geométrico de la siguiente forma. Como tener un margen funcional de ± 1 significa que $\langle \vec{w}, \vec{x}^{\pm} \rangle + b = \pm 1$ tenemos que el margen geométrico γ del plano cumple

$$\begin{aligned} \gamma &= \frac{1}{2}\left(\left\langle \frac{\vec{w}}{\|\vec{w}\|}, \vec{x}^+ \right\rangle - \left\langle \frac{\vec{w}}{\|\vec{w}\|}, \vec{x}^- \right\rangle \right) \\ &= \frac{1}{\|\vec{w}\|} \end{aligned} \tag{2.18}$$

por lo que encontrar el plano que maximize el margen geométrico es equiva-

lente a solucionar el problema de optimización

$$\begin{aligned} \underset{\vec{w},b}{\text{minimizar}} \quad & \langle \vec{w}, \vec{w} \rangle \\ \text{sujeto a} \quad & j_i(\langle \vec{w}, \vec{x_i} \rangle + b) \geq 1, \ i = 1, \ldots, N, \end{aligned} \tag{2.19}$$

donde la restricción hace que el margen funcional de cada $\vec{x}_i$ con respecto al plano sea mayor o igual que 1, porque elegimos el plano con margen 1. Así, podemos traducir el problema de encontrar el plano separador para una muestra linealmente separable con el problema de optimización convexa 2.19.

Hasta ahora solo hemos consierado situaciones en las que los datos $(\vec{x}_i, j_i)$ son linealmente separables. Para permitir clasificaciones erróneas podemos introducir variables de holgura $\xi_i > 0$ en el problema de optimización 2.19 para que los algunos ejemplos puedan ser clasificados erróneamente, es decir, $j_i(\langle \vec{w}, \vec{x_i} \rangle + b) \geq 1 - \xi_i$. También debemos introducir un costo asociado al vector de holgura $\vec{\xi} = (\xi_1, \ldots, \xi_N)$, por lo que la función objetivo será una combinación del costo y la función objetivo original. En el caso en que se utilice la norma 1 del vector $\vec{\xi}$, el problema de optimización 2.19 se convierte en

$$\begin{aligned} \underset{\vec{w},b,\vec{\xi}}{\text{minimizar}} \quad & \langle \vec{w}, \vec{w} \rangle + C \sum_i \xi_i, \\ \text{sujeto a} \quad & j_i(\langle \vec{w}, \vec{x_i} \rangle + b) \geq 1 - \xi_i, \\ & \xi_i > 0, \\ & i = 1, \ldots, N, \end{aligned} \tag{2.20}$$

y al elegir la norma 2

$$\begin{aligned} \underset{\vec{w},b,\vec{\xi}}{\text{minimizar}} \quad & \langle \vec{w}, \vec{w} \rangle + C \sum_i \xi_i^2, \\ \text{sujeto a} \quad & j_i(\langle \vec{w}, \vec{x_i} \rangle + b) \geq 1 - \xi_i, \\ & \xi_i > 0, \\ & i = 1, \ldots, N, \end{aligned} \tag{2.21}$$

que es equivalente al mismo problema tras eliminar la restricción $\xi_i > 0$, esto es,

$$\begin{aligned} \underset{\vec{w},b,\vec{\xi}}{\text{minimizar}} \quad & \langle \vec{w}, \vec{w} \rangle + C \sum_i \xi_i^2, \\ \text{sujeto a} \quad & j_i(\langle \vec{w}, \vec{x_i} \rangle + b) \geq 1 - \xi_i, \\ & i = 1, \ldots, N, \end{aligned} \tag{2.22}$$

donde C es un número que debe ser calibrado, utilizando estimaciones del error como validación cruzada. Elegimos la norma 2 como costo porque se pueden dar cotas para el errror de generalización, esto es, el error cometido al clasificar ejemplos que no están incluídos en la muestra de aprendizaje, basadas en la norma 2 del vector de holgura $\vec{\xi}$ (ver [11]), por lo que minimizar la norma 2 implica disminuir el error de generalización. El problema de optimización 2.19 es también un problema de optimización convexa.

La observación clave para la construcción del método de MSV es observar que el problema dual de 2.22 solo depende de el ptoducto interno entre los datos $\vec{x}_i, i = 1, \ldots, N$. Para ello es necesario definir el lagrangiano y el problema dual de un problema de optimización. Adicionalmente damos lasa condiciones de Karush-Kuhn-Tucker, que tienen como consecuencia el fenómeno que le da el nombre a MSV. Recordemos que para un problema de optimización de la forma

$$\begin{array}{ll} \underset{\vec{w}\in\mathcal{D}}{\text{minimizar}} & f_0(\vec{w}) \\ \text{sujeto a} & f_i(\vec{w}) \leq 0, i = 1, \ldots, m, \\ & h_i(\vec{w}) = 0, i = 1, \ldots, p, \end{array} \tag{2.23}$$

podemos definir su lagrangiano

$$L(\vec{w}, \vec{\alpha}, \vec{\beta}) = f_0(\vec{w}) + \sum_{i=1}^{m} \alpha_i f_i(\vec{w}) + \sum_{i=1}^{p} \beta_i h_i(\vec{w}) \tag{2.24}$$

y su función dual

$$W(\vec{\alpha}, \vec{\beta}) = \inf_{\vec{w}\in\mathcal{D}} L(\vec{w}, \vec{\alpha}, \vec{\beta}) \tag{2.25}$$

para la cual se cumple

$$W(\vec{\alpha}, \vec{\beta}) \leq L(\vec{w}, \vec{\alpha}, \vec{\beta}) \leq f_0(\vec{w}) \tag{2.26}$$

siempre que $\alpha_i \geq 0, i = 1, \ldots, m$. Como g es una cota inferior para f_0, podemos preguntar si maximizar g y minimizar f_0 son problemas equivalentes. Por lo anterior consideramos el problema dual de optimización

$$\begin{array}{ll} \underset{\vec{\alpha},\vec{\beta}}{\text{maximizar}} & W(\vec{\alpha}, \vec{\beta}) \\ \text{sujeto a} & \alpha_i \geq 0, i = 1, \ldots, m. \end{array} \tag{2.27}$$

En general no se tiene que maximizar W y minimizar f_0 sean problemas equivalentes. En el caso de que sí lo sean se dice que se hay dualidad fuerte.

Sin embargo, es posible probar que, para el caso en el que estamos interesandos, en el que el problema es de la forma 2.23 con $\mathcal{D}$ un dominio convexo[3], f_0 convexa[4] y $f_1, \ldots, f_m, h_0, \ldots, h_p$ funciones afines[5] entonces hay dualidad fuerte.

Si el problema de optimización 2.23 es convexo, es decir, si $f_0, \ldots, f_m$ son funciones convexas y $h_0, \ldots, h_p$ son afines, en el caso de que $f_0, \ldots, f_m$ son diferenciables es posible demostrar que para que $\vec{w}^*$ sea el punto óptimo de problema 2.23 es necesario y suficiente que exista $(\vec{\alpha}^*, \vec{\beta}^*)$ que cumplan las condiciones de Karush-Kuhn-Tucker (KKT)

$$f_i(\vec{w}^*) \leq 0, \quad i = 1, \ldots, m, \tag{2.28}$$

$$h_i(\vec{w}^*) = 0, \quad i = 1, \ldots, p, \tag{2.29}$$

$$\alpha_i^* \geq 0, \quad i = 1, \ldots, m, \tag{2.30}$$

$$\alpha_i^* f_i(\vec{w}^*) = 0, \quad i = 1, \ldots, m, \tag{2.31}$$

$$\nabla f_0(\vec{w}^*) + \sum_{i=1}^{m} \alpha_i^* \nabla f_i(\vec{w}^*) + \sum_{i=1}^{p} \beta_i^* \nabla h_i(\vec{w}^*) = 0. \tag{2.32}$$

Ahora retomamos el problema original. El lagrangiano del problema 2.22 es

$$L(\vec{w}, b, \vec{\xi}, \vec{\alpha}) = \frac{1}{2}\langle \vec{w}, \vec{w} \rangle + \frac{C}{2} \sum_{i}^{N} \xi_i^2 + \sum_{i=1}^{N} \alpha_i \left[\gamma_i (\langle \vec{w}, \vec{x}_i \rangle + b) - 1 + \xi \right]. \tag{2.33}$$

Para calcular la función dual W calculamos los puntos críticos de L

$$\frac{\partial L}{\partial \vec{w}} = \vec{w} - \sum_{i=1}^{N} j_i \alpha_i \vec{x}_i = \vec{0} \tag{2.34}$$

$$\frac{\partial L}{\partial \vec{\xi}} = C\vec{\xi} - \vec{\alpha} = \vec{0} \tag{2.35}$$

$$\frac{\partial L}{\partial b} = \sum_{i=1}^{N} j_i \alpha_i = 0 \tag{2.36}$$

[3]Un conjunto $\mathcal{D} \subseteq \mathbb{R}^n$ es convexo si para todos $d_1, d_2 \in \mathcal{D}$ se cumple $\theta d_1 + (1-\theta) d_2 \in \mathcal{D}$

[4]Una función $f(\vec{w})$ es convexa si $f(\theta \vec{w}_1 + (1-\theta)\vec{w}_2) \leq \theta f(\vec{x}_1) + (1-\theta) f(\vec{x}_2)$ para cada $\theta \in [0, 1]$

[5]Una función $h(\vec{w})$ es afín si es de la forma $h(\vec{w}) = A\vec{w} + \vec{v}$, las funciones afines son convexas

con lo que obtenemos

$$W(\vec{\alpha}) = \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,k=1}^{N} j_i j_k \alpha_i \alpha_k \left(\langle \vec{x}_i, \vec{x}_k \rangle + \frac{1}{C} \delta_{ik} \right). \tag{2.37}$$

por lo que lo que el problema de optimización 2.22 es equivalente (por haber dualidad fuerte) al problema de optimización

$$\begin{aligned} \underset{\vec{\alpha}}{\text{maximizar}} \quad & \sum_{i=1}^{N} \alpha_i - \frac{1}{2} \sum_{i,k=1}^{N} j_i j_k \alpha_i \alpha_k \left(\langle \vec{x}_i, \vec{x}_k \rangle + \frac{1}{C} \delta_{ik} \right) \\ \text{sujeto a} \quad & \sum_{i=1}^{N} j_i \alpha_i = 0, \\ & \alpha_i \geq 0, i = 1, \ldots, N, \end{aligned} \tag{2.38}$$

donde δ_{ik} es la función delta de Kronecker, $\delta_{ik} = 1$ si $i = k$ y 0 de lo contrario. Si α^* es la solución al problema 2.38, las condiciones 2.31, llamadas condiciones de complementariedad, que son en este caso

$$\alpha_i^* [j_i (\langle \vec{w}, \vec{x}_i \rangle + b) - 1 + \xi_i] = 0 \tag{2.39}$$

nos muestran que para los $\vec{x}_i$ que se encuentran en el margen o mal clasificados $\alpha_i \neq 0$ y de lo contrario $\alpha_i = 0$. Los vectores para los cuales $\alpha_i \neq 0$ son llamados vectores de soporte porque el plano encontrado está únicamente determinado por ellos. Por esto, se dice que la solución es dispersa. Esto se puede ver en la ecuación 2.34, que muestra que los vectores de soporte son los únicos que aportan al vector $\vec{w}$ que define el plano, esto es,

$$\vec{w} = \sum_{i=1}^{N} j_i \alpha_i^* \vec{x}_i = \sum_{i \in vs} j_i \alpha_i^* \vec{x}_i \tag{2.40}$$

donde $i \in vs$ hace referencia a los $\vec{x}_i$ que son vectores de soporte. b es escogido de tal manera que para los vectores de soporte su cumpla

$$j_i (\langle \vec{w}, \vec{x}_i \rangle + b) = 1 - \xi_i = 1 - \frac{\alpha_i^*}{C}, i \in vs \tag{2.41}$$

por las equaciones 2.39 y 2.35. Con lo anterior el clasificador 2.15 encontrado al solucionar el problema de optimización es

$$g(\vec{x}) = signo \left(\sum_{i \in vs} j_i \alpha_i^* \langle \vec{x}_i, \vec{x} \rangle + b \right) \tag{2.42}$$

cuyo margen geométrico $\gamma = \frac{1}{\|\vec{w}\|}$ se puede calcular con

$$\begin{aligned}
\langle \vec{w}, \vec{w} \rangle &= \sum_{i,k \in vs} j_i j_k \alpha_i \alpha_k \langle \vec{x}_i, \vec{x}_k \rangle \\
&= \sum_{i \in vs} j_i \alpha_i \sum_{k \in vs} j_k \alpha_k \langle \vec{x}_i, \vec{x}_k \rangle \\
&= \sum_{i \in vs} \alpha_i^* (1 - \xi_i^* - j_i b^*) \\
&= \sum_{i \in vs} \alpha_i^* - \sum_{i \in vs} \alpha_i^* \xi_i^* \\
&= \sum_{i \in vs} \alpha_i^* - \frac{1}{C} \langle \vec{\alpha}^*, \vec{\alpha}^* \rangle
\end{aligned} \tag{2.43}$$

por lo cual

$$\gamma = \left(\sum_{i \in vs} \alpha_i^* - \frac{1}{C} \langle \vec{\alpha}^*, \vec{\alpha}^* \rangle \right)^{-1/2}. \tag{2.44}$$

Ahora tenemos todos los ingredientes encesarios para introducir el uso de núcleos para MSV. La observación central de MSV es que el problema de optimización 2.38 solo depende de los datos $\vec{x}_1, \ldots, \vec{x}_N$ a través de la matriz $(\langle \vec{x}_i, \vec{x}_k \rangle)_{i,k=1}^N$, llamada matriz de Gram. Si podemos calcular la matriz de Gram para una representación de los datos en un espacio con producto interno de dimensión mayor, podemos modificar el problema de optimización 2.22 para encontrar un plano separador en ese espacio. Sea $\mathcal{H}$ un espacio de Hilbert con producto interno $\langle , \rangle_{\mathcal{H}}$, y $\phi : \mathbb{R}^n \to \mathcal{H}$ una función. Si logramos encontrar una función $K' : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ tal que $K'(\vec{x}, \vec{y}) = \langle \phi(\vec{x}), \phi(\vec{y}) \rangle_{\mathcal{H}}$ podemos resolver el problema de optimización equivalente a 2.38 para encontrar el plano de margen maximal en $\mathcal{H}$ usando una representación de los datos en $\mathcal{H}$ a través de ϕ, es decir, resolver

$$\begin{aligned}
\underset{\vec{\alpha}}{\text{maximizar}} \quad & \sum_{i=1}^N \alpha_i - \frac{1}{2} \sum_{i,k=1}^N j_i j_k \alpha_i \alpha_k K(\vec{x}_i, \vec{x}_k) \\
\text{sujeto a} \quad & \sum_{i=1}^N j_i \alpha_i = 0, \\
& \alpha_i \geq 0, i = 1, \ldots, n.
\end{aligned} \tag{2.45}$$

con $K(\vec{x},\vec{y}) = K'(\vec{x},\vec{y}) + \frac{1}{C}\delta(\vec{x},\vec{y})$, siendo δ la función delta de Dirac, $\delta(\vec{x},\vec{y}) = 1$ si $\vec{x} = \vec{y}$ y $\delta(\vec{x},\vec{y}) = 0$ de lo contrario. En analogía con 2.42, el clasificador obtenido al resolver el problema de optimización 2.45 es

$$g(\vec{x}) = signo\left(\sum_{i=1}^{N} j_i \alpha_i^* K\left(\vec{x}_i, \vec{x}\right) + b\right) \tag{2.46}$$

con b elegido como en 2.41 y margen geométrico 2.44.

Existe un teorema, debido a Mercer [18], que da las condiciones necesarias para que $K(\vec{x},\vec{y}) = \langle \phi(\vec{x}), \phi(\vec{y}) \rangle_{\mathcal{H}}$ para algún ϕ, lo cual es computacionalmente eficiente puesto que evita la necesidad de conocer explícitamente el espacio $\mathcal{H}$ y la función ϕ. Tal función K es llamada núcleo. Para que $K(\vec{x},\vec{y})$ defina un producto interno es claro que debe ser definida positiva y simétrica. En el caso de dimensión finita, si K es una matriz simétrica semidefinida positiva, entonces podemos escribir K de la forma

$$K = \sum_i \lambda_i v_i v_i^T \tag{2.47}$$

por lo que, si escribimos $\vec{x} = (x_1, \ldots, x_n)$ y $\vec{y} = (y_1, \ldots, y_n)$ en la base normalizada de autovectores de K, K define un producto interno $\langle \cdot, \cdot \rangle_K$ dado por

$$K(\vec{x},\vec{y}) = x^T K y = \sum_{i,j} \lambda_i x^T v_i v_i^T y = \sum_i \lambda_i x_i y_i = \langle x, y \rangle_K \tag{2.48}$$

siendo $\vec{v_i}$ los autovectores de K y $\lambda_i \geq 0$ sus autovalores. Análogamente, si λ_i y ϕ_i $i = 1, 2, \ldots$ son los autovalores y las autofunciones normalizadas del problema

$$\int K(\vec{x},\vec{z})\phi(\vec{z})d\vec{z} = \lambda\phi(\vec{x}), \tag{2.49}$$

el teorema de Mercer da condiciones para que podamos escribir

$$K(\vec{x},\vec{z}) = \sum_{i=1}^{\infty} \lambda_i \phi_i(\vec{x})\phi_i(\vec{z}) = \langle \phi(\vec{x}), \phi(\vec{y}) \rangle_{\mathcal{H}} \tag{2.50}$$

para $\phi(\vec{x}) = (\phi_1(\vec{x}), \phi_2(\vec{x}), \ldots)$ y el producto interno definido por $\langle \psi, \varphi \rangle_{\mathcal{H}} = \sum_{i=1}^{\infty} \lambda_i \psi_i \varphi_i$. Sin ir en más detalle en el análisis, citamos el teorema.

Teorema 1 (Mercer)**.** *Sea X un subconjunto compacto de $\mathbb{R}^n$. Suponga que K es una función simétrica y continua tal que el operador integral $T_K : L_2(X) \to L_2(X)$*

$$(T_K f)(\cdot) = \int_X K(\cdot, \vec{x}) f(\vec{x}) d\vec{x}$$

es positivo, esto es,

$$\int_{X \times X} K(\vec{x}, \vec{z}) f(\vec{x}) f(\vec{x}) d\vec{x} d\vec{z} \geq 0$$

para todo $f \in L_2(X)$. Entonces podemos expandir $K(\vec{x}, \vec{z})$ en una serie iniformemente convergente en $X \times X$ en términos de las autofunciones $\phi_i \in L_2(X)$, normalizadas de tal forma que $\|\phi_i\|_{L_2} = 1$ y autovalores positivos $\lambda_i \geq 0$

$$K(\vec{x}, \vec{z}) = \sum_{i=1}^{\infty} \lambda_i \phi_i(\vec{x}) \phi_i(\vec{z}).$$

Con las condiciones anteriores, podemos encontrar el núcleo K, para encontrar el plano de margen maximal en el $\mathcal{H}$ resolviendo el problema de optimización 2.45 sin siquiera conocer explícitamente la función de representación ϕ ni $\mathcal{H}$. Si K_1, y K_2 son núcleos es posible demostrar que cualquier núcleo definifo por un polinomio de dos variables $K(\vec{x}, \vec{z}) = p(K_1(\vec{x}, \vec{z}), K_2(\vec{x}, \vec{z}))$ también es un núcleo, así como $K(\vec{x}, \vec{z}) = \exp(K_1(\vec{x}, \vec{z}))$. Un núcleo popular es el núcleo radial

$$K(x, y) = e^{-\gamma \|\vec{x} - \vec{y}\|^2}, \tag{2.51}$$

que ha sido utilizado en gran variedad de aplicaciones [11]. Al utilizar el núcleo radial se debe calibrar tanto como el costo C y el ancho del núcleo γ para minimizar la tasa clasificación errónea utilizando, por ejemplo, validación cruzada.

En el caso de clasificación en $M > 2$ categorías se pueden implementar esquemas de votación. Una posible aproximación, llamada uno-contra-uno[6], es entrenar $\frac{M(M-1)}{2}$ clasificadores que discriminen entre cada par de clases. Se escoge entonces la clase que haya sido decidida por la mayoría de clasificadores y en caso de empate se elige la que tenga índice menor. Otra aproximación, llamada uno-contra-resto[7] es entrenar M clasificadores que discriminen entre

[6] En inglés *one-against-one*

[7] En inglés *one-against-rest*

cada clase y su complemento. Se elige la clase en que de la clasificación con el mayor margen.

El problema 2.45 es un problema de optimización convexa. Estos problemas han sido estudiados extensamente y existen formas eficientes de encontrar las soluciones que pueden ser aplicadas directamente. El principal obstáculo para utilizar estas aproximaciones es que el espacio necesario para almacenar la matriz de Gram del problema crece cuadráticamente con el tamaño de la muestra dado que, en general, no es una matriz dispersa. Sin embargo se pueden explotar algunas características del problema, como el hecho de que la solución es dispersa para crear algorítmos eficientes.

Optimización Secuencial Minimal (OSM) es un algorítmo utilizado para resolver el problema de optimización 2.45. Se basa en el el requerimiento de que $\sum_i \alpha_i j_i = 0$ en todas las iteraciones del algorítmo, por lo que se deben actualizar a lo sumo dos α_i en cada paso. En cada paso OSM utiliza eurísticas para eligir dos α_i, α_k para optimizar y utiliza el hecho de que este subproblema puede ser resuelto analíticamente, lo que elimina la necesidad de realizar operaciones con matrices. Este algorítmo logra reducciones en el tiempo de ejecución de varios órdenes de magnitud con respecto a algorítmos como el del gradiente. Existen implementaciones disponibles, como la librería libsvm escrita en C++ por Chih-Chung Chang and Chih-Jen Lin para la cual existe una interfaz para R en el paquete e1071 [19]. Para clasificación en múltiples clases, libsvm utiliza el esquema de votación uno-contra-uno porque, basado en el análisis de (comparison, chin-wei) los resultados con ambas aproximaciones son comparables y el método uno-contra-uno toma menos tiempo de entrenamiento.

(cita libsvm)

2.3.3. Árboles de Clasificación y Regresión

Árboles de Clasificación y Regresión (CART, por sus siglas en inglés) es un método de clasificación que utiliza el conjunto de los árboles de decisión binarios como conjunto de hipótesis. Este método fue propuesto por Breiman, Friedman, Olshen y Stone a lo largo de varios trabajos, que luego fueron condesados en [6]. La metodología consiste en construir a partir de la muestra de entrenamiento árboles binarios que a cada nodo terminal le asignan una clase, que es el resultado de la clasificación. Explicaremos la forma en que se construyen estos árboles. Una implementación libre se encuentra en el paquete rpart [47] para R.

El proceso de construcción de la regla de decisión consiste de dos pasos:

primero construir un árbol lo suficientemente grande y luego podarlo para obtener un árbol con buenas propiedades de generalización. Inicialmente se cuenta con toda la muestra de aprendizaje; la muestra se divide en dos grupos utilizando un criterio de impureza (que será definido más adelante) en un grupo que pertenece a un subconjunto del espacio de características y otro que no. Nos referimos a estas divisiones como preguntas. Se hacen nuevas preguntas hasta que no se pueda mejorar la pureza de las divisiones o hasta que cada nodo tenga un número mínimo de datos. Las divisiones de los datos pueden ser representadas con un árbol binario, lo que brinda una interpretación sencilla del clasificador. Subsecuentemente se poda el arbol resultante porque con frecuencia este es muy complejo y se sobreajusta a la muestra de aprendizaje. Una observación sin clasificar puede ser asignada a uno de los nodos terminales del árbol utilizando las preguntas que resultaron de la construcción del arbol y el resultado de su clasificación es la clase que minimice el costo de clasificación estimado con la muestra de entrenamiento para ese nodo.

Las divisiones hechas en cada nodo son preguntas hechas sobre una variable y para construir un arbol de decisión de manera algorítmica se necesita un conjunto de pregutas predeterminadas. Por esta razón se necesita que la muestra tenga una estructura estándar. Una muestra de aprendisaje $\mathcal{L} = \{(\vec{x}_1, j_1), \ldots, (\vec{x}_M, j_M)\}$ tiene estructura estándar si todos los vectores de características $\vec{x}$ tienen la forma $\vec{x} = (x_1, \ldots, x_n)$, es decir, tienen dimensionalidad fija y los x_i están ordenados de tal forma que corresponden a la misma característica. Cada vector de características puede consistir de variables continuas o categóricas. En cada nodo se hace una pregunta y las preguntas estándar son de la forma $x_k \leq c$ en el caso de que x_k sea continua y $x_m \in S$ en el caso que x_m sea categórica, donde S es un subconjunto de los posibles valores que la variable x_m puede tomar. Las preguntas posibles pueden hacerse más complejas para incluír combinaciones lineales de características contínuas o combinaciones booleanas entre variables categóricas, sin embargo esto aumenta considerablemente el número de preguntas posibles para una muestra, lo cual hace que buscar exhaustivamente sobre todas las preguntas sea poco práctico computacionalmente.

Antes de dar el criterio para dividir un nodo en dos nodos hijos, tenemos que definir una medida de impureza. En el nodo t hay una proporción $p(i|t), i = 1, \ldots, M$ de datos de cada una de las clases. Definimos una medida de impureza $i(t)$ como una función positiva $\phi = \phi(p(1|t), \ldots, p(M|t))$ que cumple dos propiedades:

- ϕ alcanza su máximo en $\phi(\frac{1}{M}, \ldots, \frac{1}{M})$, es decir, la impireza es máxima cuando hay igual proporción elementos de cada clase en el nodo t,
- $\phi(1, 0, \ldots, 0) = \phi(0, 1, \ldots, 0) = \cdots = \phi(0, 0, \ldots, 1) = 0$ que significa que la impureza es 0 cuando todos los elementos en el nodo pertenecen a una única clase (el nodo es puro).

Dos medidas de impureza son la entropía de Shannon

$$-\sum_{i=1}^{M} p(i|t) \log_2 p(i|t), \tag{2.52}$$

y el coeficiente de Gini

$$\sum_{i=1}^{M} \sum_{j=1, j\neq i}^{M} p(i|t)p(j|t). \tag{2.53}$$

Se ha observado que el arbol resultante es más bien insensible a la escogencia de medida de impureza [6].

Una forma de hacer las divisiones en cada nodo es maximizar el cambio de impureza de cada partición dada una elección de medida de impureza. Una división s corresponde a una pregunta estándar. Si se dividen los datos que se encuentran en el nodo t en dos nodos, t_I y t_D en proporciones p_I y p_D, el cambio en la impureza es

$$\Delta i = i(t) - p_I i(t_I) - p_D i(t_D). \tag{2.54}$$

Para encontrar la división que maximiza el cambio de impureza es necesario hacer una búsqueda exhasustiva sobre todas las preguntas que se pueden hacer en el nodo en cuestión.

Para describir el proceso de poda de CART necesitamos estimados del error de clasificación y, más generalmente, de costo de clasificación errónea. Nos referimos al conjunto de nodos terminales de un árbol T como $\widetilde{T}$ y a la proporción de la muestra de entrenamiento en un nodo terminal t como $p(t)$. La regla de decisión en cada nodo terminal t es asignar la clase $j(t)$ a la que pertenezca la mayor parte de los datos en el nodo, es decir, $j(t) = \arg\max_{j\in\{1,\ldots,M\}} p(j|t)$. La estimación por resubstitución de la probabilidad de clasificación erróneas en un nodo t es, con esta regla,

$$r(t) = 1 - \max_{j\in\{1,\ldots,M\}} p(j|t) = 1 - p(j(t)|t) \tag{2.55}$$

y denotamos

$$R(t) = r(t)p(t). \tag{2.56}$$

La estimación por resubstitución del error cometido por el árbol T es

$$R(T) = \sum_{t \in \tilde{T}} R(t). \tag{2.57}$$

Podemos introducir un costo $C(i|j)$ asociado a la clasificación incorrecta de un elemento de clase i como uno de clase j. $C(i|j)$ debe ser una función no negativa tal que $C(i|i) = 0$ para cada i. la estimación por resubstitución del costo de clasificación en el nodo t es

$$r(t) = \min_j \sum_i C(j|i)p(j|t). \tag{2.58}$$

En ese caso la regla de decisión $j(t)$ en un nodo terminal t es asignar la clase que minimice el costo esperado de clasificación errónea, esto es, $j(t) = \arg\min_{j \in \{1,\ldots,M\}} \sum_i C(j|i)p(i|t)$. El estimado por resubstitución para el costo esperado de clasificación errónea para el árbol T es

$$R(T) = \sum_{t \in \tilde{T}} r(t)p(t) = \sum_{t \in \tilde{T}} R(t). \tag{2.59}$$

Cuando $C(i|j) = 1 - \delta_{ij}$ la regla de decisión que asigna a un nodo terminal la clase que presente el menor costo esperado es equivalente a la regla que le asigna la de menor probabilidad de error, por lo que la formulación en términos de funciones de costo es más general.

Ahora podemos describir el proceso para podar un arbol de decisión. Decimos que T_2 es subarbol podado de T_1 si T_2 se obtiene al quitarle ramas al árbol T_1, esto lo denotamos por $T_1 > T_2$. La metodología consiste en construir una secuencia $T_1, \ldots, T_k$ de árboles a partir del arbol construido anteriormente T_{max} con nodo inicial t_1 que cumpla $T_{max} = T_1 > T_2 > \ldots > T_k = \{t_1\}$ para luego elegir el mejor árbol entre ellos. Si $|\widetilde{T}|$ es el número de nodos terminales de un árbol T, definimos la complejidad del arbol como $|\widetilde{T}|$ y la función de costo-complejidad para el arbol T como

$$R_\alpha(T) = R(T) + \alpha|\tilde{T}|. \tag{2.60}$$

Minimizar $R_\alpha(T)$ significa encontrar un compromiso entre la complejidad del arbol y su costo estimado de clasificación errónea. Por ejemplo, para

un arbol T que tenga tantos nodos terminales como datos en la muestra de entrenamiento el estimado por resubstitución del error de clasificación errónea es cero, sin embargo tiene la mayor complejidad posible para esa muestra de entrenamiento. Para un nodo t del arbol T definimos

$$R_\alpha(\{t\}) = R(\{t\}) + \alpha \tag{2.61}$$

y, si T_t es el arbol de los descendientes de t, se tiene que mientras se cumpla que

$$R_\alpha(T_t) < R_\alpha(t) \tag{2.62}$$

el arbol de descendientes de t tiene un menor costo-complejidad que el nodo t, por lo que es preferible mantener la rama que se desprende de t. Para cada nodo t hay entonces α en el que la desigualdad anterior deja de ser cierta y está dado por

$$\alpha_t = \frac{R(\{t\}) - R(T_t)}{|\tilde{T}| - 1}. \tag{2.63}$$

Este valor de α_t es una medida de qué tan fuerte es el nodo t como enlace a la rama T_t en el sentido de que si es mayor para un nodo t_1 que para un nodo t_2, a medida que aumenta el valor de α, es preferible mantener la rama T_{t_1} que la rama T_{t_2}. Así pues se construye la secuencia $T_1, \ldots, T_k$ de árboles quitándo sucesivamente la rama más débil, esto es, el arbol T_{i+1} se obtiene a partir de T_i quitándole la rama para la cual α_t es mínimo entre los nodos $t \notin \widetilde{T}$. A cada arbol T_i de la sucesión construida le corresponde un valor α_i. Por la forma en que fueron construidos los T_i, la sucesión α_i es creciente, es decir, $\alpha_1 > \ldots > \alpha_k$. Es posible demostrar que cada uno de los árboles T_i es el árbol que minimiza el la función de costo-complejidad $R_{\alpha_i}(T)$ y que, si $I_1 = (0, \alpha_1], I_2 = (\alpha 2, \alpha_3], \ldots, I_k = (\alpha_{k-1}, \infty)$, entonces a cualquier valor de $\alpha \in I_i$ le corresponde el mismo árbol que minimiza $R_\alpha(T)$, T_i.

Una vez construida una secuencia de árboles $T_{max} = T_1 > T_2 > \ldots > T_k = \{t_1\}$ a partir de un árbol T_{max} con nodo inicial t_1 se debe elegir entre uno de ellos para usar como clasificador. Para escoger el mejor árbol se utiliza validación cruzada. Para cada I_i se define $\beta_1 = 0, \beta_2 = \sqrt{\alpha_1 \alpha_2}, \ldots, \beta_{k-1} = \sqrt{\alpha_{k-2}\alpha_{k-1}}, \beta_k = \infty$, que pueden ser pensados como puntos representativos de cada intervalo I_i. Denotamos con T_{β_i} al arbol que minimiza la función costo-complejidad R_{β_i}. La muestra de entrenamiento $\mathcal{L}$ se divide en v partes iguales $\mathcal{L}_i, i = 1, \ldots, v$ y con cada $\mathcal{L}^k = \mathcal{L} \setminus \mathcal{L}_k$ se encuentran los árboles T_{β_i} y los estimados $R_{\beta_i}(T_{\beta_i})$. Luego se suma sobre los $\mathcal{L}^k$ para obtener el estimado por validación cruzada de la función de costo-complejidad para

cada β_i. Finalmente se elije el valor β_i que minimiza estas estimaciones del error, que llamamos β, y se elije como arbol podado al arbol T_β construido con la totalidad de los datos.

Es posible dar una medida de la importancia de cada variable. Para un nodo t del arbol T, podemos calcular la mejor división que una variable x_m puede dar, digamos s_m^*. Si se hiciera la división s_m^*, la impureza de los nodos hijos de t cambiaría en $\Delta I(s_m^*, t)$. Se define la medida de importancia de x_m como

$$M(x_m) = \sum_{t \in T} \Delta I(s_m^*, t). \tag{2.64}$$

Los árboles de clasificación pueden ser utilizados para encontrar estructuras en los datos, sin embargo son sensibles al ruido. Dado que estos clasificadores pueden ser representados como un arbol y es posible dar una medida de importancia de las características usadas en su construcción, lo árboles de clasificación con frecuencia son útiles para inferir información sobre los procesos que generan los datos. A pesar de esto, la forma de los árboles es sensible al ruido, puesto que si en un nodo t dos variables pueden lograr disminuciones parecidas en la pureza al tenerlas en cuenta para la siguiente división, la presencia de ruido puede hacer que esta elección sea esencialmente aleatoria. Por esta razón la estructura del clasificador puede variar apreciablemente con diferentes muestras de un proceso ruidoso. Para lidiar con estas variaciones existen otros métodos de aprendizaje que buscan reducir con esta varianza utilizando remuestreo o selección aleatoria de variables, como es el caso de bosques aleatorios.

2.3.4. Bosques Aleatorios

Este método fue propuesto por Breiman en [5]. El método consiste en construir un conjunto de árboles con la metodología de CART cuyas decisiones estén poco correlacionadas. Cada arbol por sí solo es un clasificador débil, sin embargo al tomar la decisión de la mayoría de los arboles entrenados se obtiene un clasificador que suele ser mejor que el obtenido con CART. La consistencia de este método fue demostrada por Biau, Devroye y Lugosi en [2] y utilizamos la implementación incluida en el paquete randomForest [17] para R [23].

La metodología se basa en el empaquetamiento y la selección aleatoria de características. Se construyen árboles pequeños (típicamente de 3-5 niveles). Para generar cada arbol se toma una submuestra con reemplazo de la muestra

de entrenamiento $\mathcal{L} = \{(\vec{x}_1, j_1), \ldots, (\vec{x}_N, j_N)\}$ y para realizar las divisiones se escoge subconjunto aleatorio de tamaño m de las características para realizar cada división del arbol. Despues de generar una número predeterminado de árboles, se toma la decisión de la mayoría de ellos al clasificar un nuevo dato.

Con bosques aleatorios es posible dar medidas de la importancia de las variables en la clasificación. Existen dos medidas de importancia de las variables. La primera es simplemente la reducción promedio de la impureza al usar la variables en cuestión en cada uno de los árboles. La segunda es se basa en el argumento de que, cuando se permutan una característica x_i, la precisión del clasificador disminuye. Si una característica está fuertemente asociada a la clasificación, es de esperar que la diferencia entre la precisión del clasificador al usar la variable permutada junto con las demás y usarlas sin modificar sea alta. La precisión se estima utilizando estimados *out of bag* que consisten en utilizar la proporción de clasificaciones correctas para cada uno de los datos $\vec{x}_i$ de la muestra de aprendizaje hecha por los árboles construidos sin utilizar $\vec{x}_i$. Aunque esta medida de importancia ha mostrado ser útil para descubrir la importancia real de las características, ha mostrado tener un sesgo hacia características correlacionadas, es decir, dadas dos características que estén correlacionadas, su importancia será mayor [46].

Capítulo 3

Clasificación

Conociendo la curva de luz de un objeto podemos clasificarlo según su tipo de variabilidad estelar, sin embargo esta relación no puede ser programada en un computador de manera sencilla. Cuando decimos que a cada curva de luz le corresponde un tipo de variabilidad estelar queremos decir que existe una relación funcional entre curvas de luz y tipos de variabilidad estelar. Esta relación funcional puede, en general, no ser determinísta por lo que es necesario un marco probabilístico. La función objetivo será entonces la que minimice un costo asociado a la clasificación errónea de los objetos. El objetivo general del aprendizaje supervisado es aproximar esta función utilizando la experiencia previa. Esta experiencia previa es, en este caso, nuestro conjunto de datos (ver cuadro 3.1) y la estimación de esta función, o regla de decisión, es encontrada mediante un algorítmo de aprendizaje. Un algorítmo de aprendizaje escoge una regla de decisión de un conjunto de reglas, llamado conjunto de hipótesis. Por ejemplo la metodología de árboles de decisión utiliza como conjunto de hipótesis el conjunto de todos los árboles de decisión binarios.

Cada curva de luz en nuestro conjunto de datos es una tríada que consta de una sucesión de mediciones de magnitud, una suseción de fechas y un tipo de variabilidad estelar. Como cada curva de luz tiene un número de mediciones diferentes que están repartidas en diferentes intervalos de tiempo, esto dificulta la implementación de algorítmos para entrenar una regla de desisión. Para hacer frente a esto, le asignamos a cada curva de luz un vector de dimensionalidad fija, llamado vector de características. Esta asignación se hace con la intención de clasificar las curvas de luz con base en sus vectores de características, por lo que deben capturar las diferencias entre diferentes clases.

Este vector puede ser, en principio, una combinación de variables categóricas y numéricas; en este trabajo le asignamos únicamente variables numéricas. La función de decisión divide el espacio de caracterísitcas en regiones tales que a cada elemento del espacio de características le asigna una clase (un tipo de variabilidad estelar) basado en qué región se encuentra. Así, para clasificar una curva cuyo tipo de variabilidad es desconicido, calculamos su vector de caracterísicas y le asignamos la clase de variabilidad dada por la regla de decisión previamente entrenada. Por lo tanto la elección de características es crucial puesto que si los vectores de características de diferentes clases se superponen, no será posible entrenar una regla de decisión que distinga entre las clases superpuestas.

Subsecuentemente llamaremos $g : \mathbb{R}^n \to \{VLP, \ldots, BeEC\}$ al clasificador en cuestión que le asigna a cada vector de características un tipo de variabilidad (ver cuadro 3.2). Representamos cada curva de luz con una pareja $(\vec{x}, i)$, siendo $\vec{x} \in \mathbb{R}$ su vector de características e $i \in \{VLP, \ldots, BeEC\}$ la clase a la que pertenece. La regla de decisión se equivoca si $g(x) \neq i$. Suponemos que existe una distribución de probabilidad $P(\vec{x}, i)$ que representa la probabilidad de observar el vector de características $\vec{x}$ con el tipo de variabilidad i y buscamos una regla que minimice la probabilidad de error $P(g(x) \neq i)$. Como no conocemos el valor real de la probabilidad de error, debemos estimarla a partir de los datos.

Para estimar la probabilidad de que la función de decisión entrenada por un algorítmo de aprendizaje con la totalidad de los datos se equivoque al realizar futuras clasificaciones, utilizamos validación cruzada de v iteraciones. Este procedimiento constiste en dividir la muestra $\mathcal{L}$ en v muestras de prueba $\mathcal{L}_k$, $k = 1, \ldots, v$ con el mismo número de elementos (o lo más próximo posible) y definimos la k-ésima muestra de entrenamiento como $\mathcal{L}^k = \mathcal{L} \setminus \mathcal{L}_k$. Utilizando cada una de las v muestras de entrenamiento $\mathcal{L}^k$ entrenamos una regla de decisión utilizando el algorítmo de aprendizaje en cuestión, con ella clasificamos los elementos de la muestra de prueba $\mathcal{L}^k$ y calculamos N_{ij}^k el número de elementos de de la clase j clasificado como i. Definimos $N_{ij} = \sum_k N_{ij}^k$ el número total de elementos de la clase j clasificado como i. Estimamos la probabilidad de que un elemento de la clase j sea clasificado como i, $P^{VC}(g(\vec{x}) = i|j)$, con N_{ij}/N_j, donde N_j es el número de elementos pretenecientes a la clase j en la muestra $\mathcal{L}$. Intuitivamente, si la muestra es grande tendremos aproximadamente el mismo poder para clasificar con la muestra completa que con una fracción $\frac{v-1}{v}$ de ella, por lo cual P^{VC} será una buena aproximación a la probabilidad real de clasifica-

ción. Tomamos $v = 10$ siguiendo la popularidad de este valor en la literatura. La estimación de la probabilidad de que un elmento cualquiera sea clasificado correctamente, llamada precisión, será $\sum_i P^{VC}(g(\vec{x} =)i|i)P(i)$. $P(i)$ es la probabilidad *a priori* de encontrar un objeto del tipo de variabilidad i. Como nuestra muestra no es representativa de las poblaciones de estrellas observadas y no existen estudios al respecto en la literatura para todos los tipos de variabilidad, tomamos $P(i)$ uniforme, es decir, $P(i) = 1/7$ para cada i (hay 7 tipos de variabilidad estelar en la muestra).

Con frecuencia es necesario ajustar algún parámetro para un algorítmo de aprendizaje. Este es el caso de Máquinas de Soporte Vectorial, donde es necesario ajustar el parámetro de costo y, si se utiliza un núcleo gaussiano, el ancho γ de éste. Utilizamos la maximización de la precisión como criterio para elegir los parámetros óptimos. Adicionalmente analizamos para cada clase la sensibilidad $p(g(\vec{x}) = i|i)$ (tasa de verdaderos positivos), la especificidad $P(g(\vec{x}) \neq i|i^c)$ (tasa de verdaderos negativos), el poder de predicción positiva $p(i|g(\vec{x}) = i)$ (probabilidad de que una vez clasificado, la clasifición sea correccta) y el poder de predicción negativa $p(i^c|g(\vec{x}) \neq i)$. El poder de predicción positiva juega un papel importante en este análisis puesto que, dada una nueva base de datos cuya clasificación no se conoce, si aplicamos el clasificador entrenado con nuestra muestra, esta es la estimación de la probabilidad de que esa clasificación sea correcta, lo cual corresponde a las situaciones reales que se encontrarán una vez se hagan públicos nuevas curvas de luz de estrellas variables sin clasificar.

3.1. El conjunto de Datos

Los datos utilizados en este trabajo provienen de la tercera fase del *Optical Gravitational Lensing Experiment* (OGLE-III). OGLE es un proyecto de larga duración cuyo objetivo principal es la búsqueda de materia oscura mediante el aprovechamiento de lentes gravitacionales. La tercera fase del protecto comenzó en 2001 y hace uso de un telescopio de 1,3m de diámetro localizado en el observatorio de Las Campanas, Chile[48]. Uno de los principales resultados de OGLE-III es la reducción y publicación [49] de las curvas de luz de objetos en el bulbo de la Galaxia, la Gran Nube de Magallanes y la Pequeña Nube de Magallanes. En este trabajo utilizamos las curvas de luz de 431653 objetos del catálogo de estrellas variables de OGLE-III de seis tipos de variabilidad (ver tabla 3.1) al cual se puede acceder en la página del

fuente de los datos de las Be

proyecto [1] y 475 curvas de luz de estrellas candidatas a ser clasificadas como Be.

Las curvas de luz tomadas del catálogo de estrellas variables de OGLE-III se encuentran clasificadas por tipo de variabilidad estelar en un proceso que que involucró, en una etapa, la inspección manual de las curvas de luz (ver referencias en la tabla 3.1) por lo cual tomaremos esta clasificación como verdadera. En este trabajo utilizamos únicamente las curvas de luz registradas en la banda I [2] a pesar de que también se encuentra disponible información adicional sobre las curvas de luz como sus periodos y algunos coeficientes de Fourier (ver referencias en la tabla 3.1). Esta elección se debe a que el cálculo de estas cantidades es computacionalmente intensivo, no siempre se encuentram disponibles datos en diferentes bandas y proponemos hacer la clasificación utilizando variables tomadas de estadística descriptiva.

Agrupamos los 432128 objetos disponibles en siete clases de variabilidad estelar (ver tabla3.2). Esta elección de clases puede ser refinada puesto que en cada una de estas clases existen subclases. Por ejemplo entre las Cefeidas se puede distinguir entre aquellas que pulsan en su modo fundamental, en su primer sobretono (segundo armónico) o en su segundo sobretono (tercer armónico) (ver figura 3.1). Sin embargo conocer a qué clase de variabilidad estelar pertence un objeto facilita considerablemente su clasificación en subclases y análisis subsecuentes.

En el Catálogo de Estrellas Variables de OGLE-III, cada curva de luz está disponible en un archivo que contiene tres columnas con los valores de magnitud, fecha juliana [3] en la que fue tomada cada medida y error en la medida de la magnitud. El número de medidas para cada objeto y la separación temporal varía ampliamente. La separación mínima dos mediciones en toda la muestra es de 0.00147d, la máxima es 2156.9d y en promedio están separadas por 5.1d; por su parte el número promedio de observaciones por objeto es 759; el máximo, 5173; y el mínimo, 11. El 75 % de los objetos cuenta con más de 386 observaciones. Para todos los objetos estas observaciones están repartidas en los años en que estuvo activo OGLE-III. En la figura 3.2 se puede observar una curva de luz del catálogo de estrellas variables de

número de años que estuvo activo OGLE

[1] http://ogle.astrouw.edu.pl/

[2] Los objetos observados emiten radiación en una parte amplia del espectro electromagnético. Los telescopios utilizan filtros para recoger solo la radiación emitida por estos objetos en ciertas partes del espectro electromagnético. El filtro I (infrarojo) tiene un ancho de banda de 149nm y una longitud de onda efectiva de 797nm (ver [16])

[3] La fecha Juliana es el tiempo medido en días desde el 1 de enero de 4713 a. C.

Tipo de variabilidad y origen	Número de Objetos
RR Lyrae - BG[33]	16836
RR Lyrae - PNM [38]	2475
RR Lyrae - GNM [40]	24906
Cefeidas - BG [36]	32
Cefeidas - PNM [35]	4630
Cefeidas - GNM [34]	3361
Variables de Largo Periodo - BG [44]	232406
Variables de Largo Periodo - PNM [43]	19384
Variables de Largo Periodo - GNM [41]	91995
Sistema Binario Eclipsante - PNM [21]	6138
Sistema Binario Eclipsante - GNM [15]	26121
δ-Scuti - Nube Mayor de Magallanes[22]	2786
Cefeidas Tipo II - BG [37]	335
Cefeidas Tipo II - PNM [42]	43
Cefeidas Tiplo II - GNM [39]	197
Cadidata a Be - Vía Láctea (cita!)	475

Cuadro 3.1: Conjunto de datos utilizados. BG hace referencia al Bulbo Galáctico; PNM, a la Pequeña Nube de Magallanes y GNM, a la Gran Nube de Magallanes.

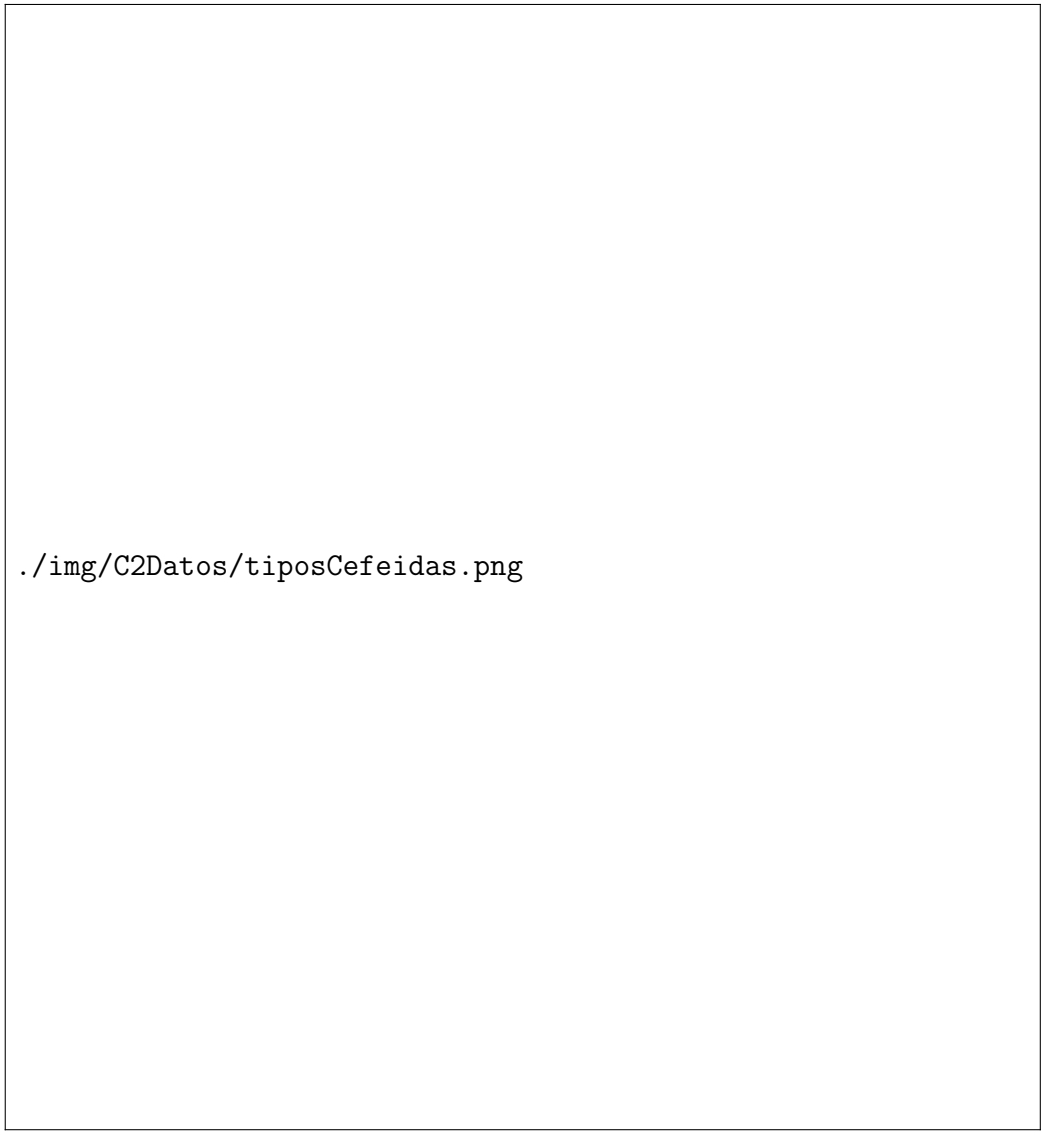


Figura 3.1: Curvas de luz ilustrativas de Cefeidas en modo fundamental (izquierda), primer sobretono (mitad), segundo sobretono (derecha). Los números pequeños a la derecha de cada recuadro muestran los periodos redondeados de las curvas de luz presentadas en los recuadros. Tomado de [36]

Tipo de Variabilidad	Cantidad
Variables de Largo Periodo (VLP)	343782
RR Lyrae (RRLyr)	44217
Cefeida (Cef)	8004
Sistema Binario Eclipsante (SBE)	32259
δ-Scuti (δSct)	2788
Cefeida Tipo II (CefT2)	603
Candidata a Be (BeEC)	475
Total	432128

Cuadro 3.2: Cantidad de datos por tipo de variabilidad

OGLE-III.

3.2. Características Seleccionadas

Para una curva de luz denotamos con $(m_i)_{1\leq i\leq n}$, $(t_i)_{1\leq i\leq n}$ y j a su serie de magnitudes, tiempos y tipo de variabilidad respectivamente.

Idealmente, el vector de características debe ser fácil de cálcular y debe capturar las diferencias entre los tipos de variabilidad estelar, sin embargo en estudios previos [12, 29, 24] los parámetros utilizados no son calculables de manera rápida y sin intervención humana. En la literatura [12, 29, 24] se han utilizado coeficientes de Fourier para este propósito. Suponiendo que los pares (t_i, m_i) provienen de una versión corrupta de la magnitud verdadera, es decir, $m(t) = y(t) + \epsilon$ donde $m(t)$ es la magnitud observada, $y(t)$ es la magnitud verdadera, ϵ es una variable aleatoria que modela el ruido, y $m_i = m(t_i), y_i = y(t_i)$, en [12] los autores encuentran estimadores de mínimos cuadrados para los parámetros a_{ls}, f_l y b_{ij} del el modelo

$$\tilde{y}(t) = \sum_{l=1}^{3}\sum_{s=1}^{4}(a_{ls}\,\mathrm{sen}\,2\pi f_l st + b_{ls}\cos 2\pi f_l st) + b_0.$$

Luego los autores utilizan estos coeficientes para dar una descripción de $y(t)$ que es independiente de traslaciones temporales. Lo importante no es entrear en los detalles de esta elección de parámetros sino resaltar que la búsqueda de estos es computacionalmente intensiva. Los autores de [12] utilizan el periodograma de Lomb-Scargle [30] con el cual se obtiene una potencia para

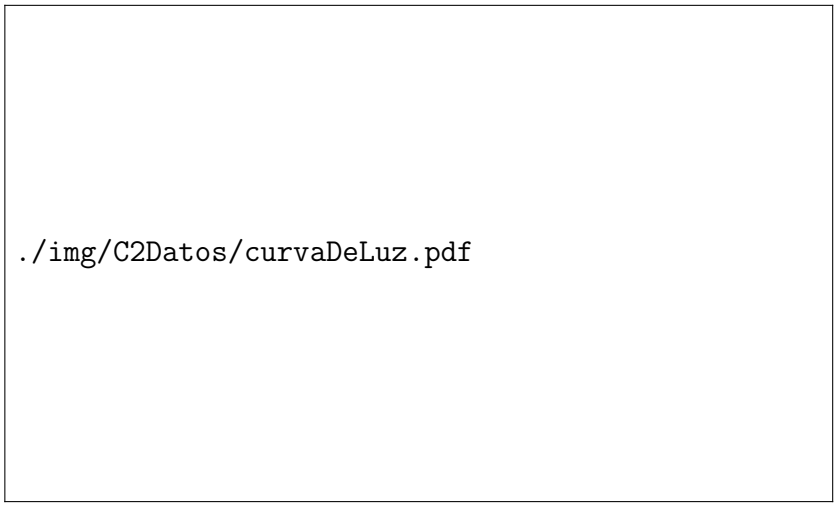


Figura 3.2: Curva de luz de OGLE-LMC-CEP-2515 del catálogo OGLE-III. Los periodos en los que no hay mediciones corresponden a los momentos del año en los que la zona en la que se encuentra el objeto no puede ser observada debido a la posición relativa entre el Sol y la Tierra.

cada periodo posible. Predefinir los periodos posibles es un reto si no se tiene más información que la curva de luz de un objeto. Por ejemplo para clasificar las curvas de luz de Cefeidas Clásicas para el catálogo de OGLE-III [34] los autores probaron frecuencias entre 0.0 y 24.0 ciclos por día en aumentos de frecuencias de 0.0001 para 32 millones de objetos, para lo cual utilizaron supercomputadores del *Centre for Mathematical and Computational Modeling* de la Universidad de Varsovia, siguido de un análisis que llevó a la inspección manual de decenas de miles de curvas de luz. En este trabajo, basado en los hallazgos de [25] y [27], proponemos utilizar en lugar de estos coeficientes, variables descriptivas de la serie de magnitudes que pueden ser calculadas en tiempos abrumadoramente menores, con menos poder computacional y sin intervención manual.

Bajo este punto de vista, las magnitudes son vistas como una variable aleatoria y las cantidades de la tabla 3.3 son variables descriptivas de su densidad. En la figura 3.3 se observa una curva de luz y la densidad estimada de sus magnitudes. Al utilizar la distribución de la serie de magnitudes se asume que el número de observaciones es lo suficientemente grande, que estas son hechas en intervalos que evitan el aliasing y que son hechas durante al menos un periodo del objeto observado. Es de esperar que las curvas que tienen for-

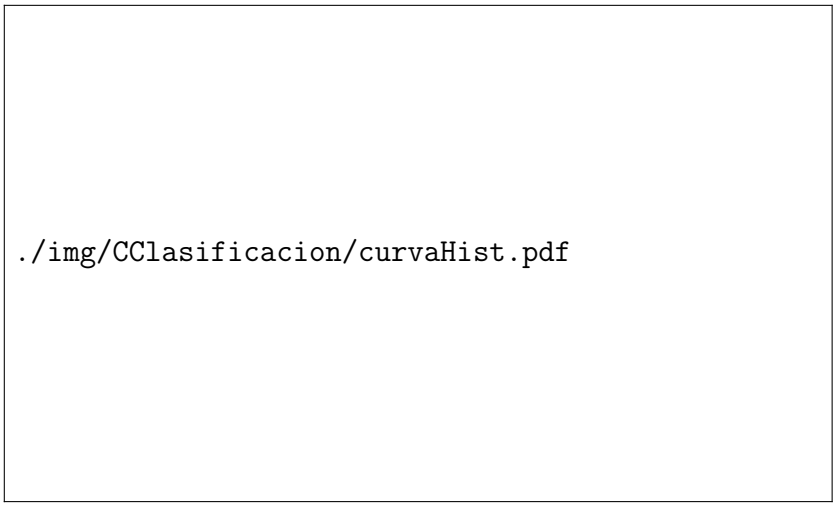


Figura 3.3: Curva de luz OGLE-LMC-CEP-0503 y densidad estimada de las magnitudes.

mas similares, es decir, que pertenecen al mismo tipo de variabilidad estelar, tengan densidades de magnitudes similares y que, por ende, los parámetros descriptivos utilizados también sean similares.

Las caracterísitcas utilizadas se encuentran en la tabla 3.3. La media, la desviación estándar, el sesgo,la curtosis, el rango y la variación cuadrática son variables bien conocidas de estadística descriptiva y damos sus descripciones más adelente. El valor Abbe $\mathcal{A}$ y el valor Abbe promedio $\bar{\mathcal{A}}_t$ fueron propuestos para el estudio de fenómenos transientes en [20] y su definición e interpretación son dadas má adelante. La entropía de Rényi es una medida de la incertidumbre propuesta por [26] que generaliza la entropía de Shannon [31] y que le da más peso a los valores más probables de m_i. A continuación discutimos cada una de estas variables.

La media μ y la desviación estándar σ (ver cuadro 3.3) y el rango son variables descriptivas bien conocidas. En este caso la media es el valor al rededor del cual la serie de magnitudes oscila y tanto como la desviación estándar como el rango son una medida de la amplitud de estas oscilaciones. . La figura 3.4 muestra la densidad de cada una de las clases en el plano μ-σ. Aunque las diferentes clases se superponen en este plano, hay pares de clases que pueden ser distinguidas como δSct y RRLyr.

(Dar argumentos astronómicos)

Basado en el trabajo de [25] utilizamos el sesgo y la curtosis como ca-

Cantidad	Fórmula
Media	$\mu = \frac{1}{n}\sum_i m_i$
Desviación estándar	$\sigma = \sqrt{\frac{1}{n}\sum_i (m_i - \mu)^2}$
Sesgo	$\frac{1}{n}\sum_i \left(\frac{m_i - \mu}{\sigma}\right)^3$
Curtosis	$\frac{1}{n}\sum_i \left(\frac{m_i - \mu}{\sigma}\right)^4$
Rango	$\text{máx}_i\, m_i - \text{mín}_i\, m_i$
Variación cuadrática	$\frac{1}{n}\sum_i (m_i - m_{i-1})^2$
Valor Abbe [20]	$\mathcal{A} = \frac{n}{2(n-1)} \frac{\sum_i (m_i - m_{i-1})^2}{\sum_i (m_i - \mu)^2}$
Abbe promedio [20]	$\bar{\mathcal{A}}_t$
Entropía de Shannon [31]	$\sum_i -p_i \log_2 p_i$
Entropía de Rényi[26]	$\frac{1}{1-\alpha} \log_2 \sum_x p_i^\alpha$

Cuadro 3.3: Variables utilizadas como características para la calsificación automática.

racterísticas. El sesgo es el tercer momento central estandarizado [4] y es una medida de de la asimetría de una distribución. Una distribución es simétrica si su sesgo es 0, su cola izquierda es más larga si su sesgo es positivo y su cola derecha es más larga si su sesgo es negativo. Por su parte la curtosis es el cuarto momento central estndarizado. Es una medida de qué tan concentrada está la distribución al rededor de la media. La curtosis de una distribución normal es 3 y con frecuencia se estudia una cantidad llamada exceso de curtosis que es el resultado de restarle 3 a la curtosis. Los autores de [25] encontraron que algunos tipos de variabilidad estelar podían ser distinguidos utilizando clasificadores lineales en el plano sesgo-curtosis.

Dado que la estimación de el sesgo y la curtosis con las fórmulas del cuadro 3.3 requiere de calcular las potencias $(\mu - m_i)^3$ y $(\mu - m_i)^4$, son propensas a dar estimaciones erróneas en el caso de que existan datos atípicos. Calculamos también la l-curtosis y el l-sesgo[5] y reemplazando el sesgo y la curtosis por

[4] El k-ésimo momento centrado de una variable aleatoria X (o de su distribución) es $\mu_k = E[(X - \mu)^k]$, siendo μ su media. Su k-ésimo momento central estandarizado es $\frac{\mu_k}{\sigma^k}$, siendo σ la desviación estándar.

[5] Los l-momentos son combinaciones lineales de los estadísticos de orden. Son robustos, toman valores entre 0 y 1, y la interpretación de sus valores es análoga a la de los momentos. De la misma manera en que se define el sesgo y la curtosis muestral, es posible definir la l-curtosis y el l-sesgo. Fuero propuestos en (cita l-momentos) y su cálculo fue realizado utilizando el paquete lmoments (cita paquete l-moments) para R

Figura 3.4

estas cantidades no encontramos dierencias importantes en el poder para clasificar de los clasificadores que utlizamos. Esto puede ser un indicador de que no existe una proporción grande de datos atípicos en las curvas de luz. Por su simplicidad utilizamos el sesgo y la curtosis.

El valor Abbe fue propuesto por [20] para la detección de estrellas que muestran fenómenos transientes. El valor Abe es el cociente de dos cantidades. Por un lado si la medición $m_i = m(t_i)$ es una versión corrupta de la magnitud verdadera $y(t)$ y $m(t) = y(t) + \epsilon$ con ϵ una variable aleatoria tal que $E(\epsilon) = 0$, es posible demostrar que

$$\frac{1}{2(n-1)} \sum_i (m_i - m_{i-1})^2 \tag{3.1}$$

es un estimador consistente de la varianza de los residuos $m(t) - y(t) = \epsilon$, es decir,

$$\frac{1}{2(n-1)} \sum_i (m_i - m_{i-1})^2 \to Var(\epsilon) \tag{3.2}$$

casi seguramente cuando el número de observaciones tiende a infinito y $y(t)$ es Lypschitz continua y ϵ ... . $y(t)$ puede ser visto como un modelo que

mirar condiciones para que esta convergencia se de

será ajustado a los datos por lo que si tomamos el modelo y $Var(\epsilon)$ como la varianza residual con respecto a este modelo, estimar la varianza residual utilizando 3.2 tiene la virtud de que no es necesario proponer una forma para $y(t)$ para calcularlo. En el caso de que $y(t) = \mu$,

$$\frac{1}{n}\sum_i (m_i - \mu)^2 \tag{3.3}$$

no es más que la estimación de la varianza residual con respecto al modelo que toma a $m(t)$ como una versión ruidosa de una función constante. En este orden de ideas el valor Abbe

$$\mathcal{A} = \frac{n}{2(n-1)} \frac{\sum_i (m_i - m_{i-1})^2}{\sum_i (m_i - \mu)^2} \tag{3.4}$$

es una medida de qué tan lejos está $y(t)$ de ser constante basado en la información disponible en las observaciones m_i. Para el caso en que $y(t)$ es constante es de esperar que $\mathcal{A} \approx 1$, mientras que de lo contrario el se espera que 3.2 sea menor que 3.5 y, por lo tanto $\mathcal{A} < 1$. Adicionalmente incluímos el estimador de la variación cuadrática

$$\frac{1}{n}\sum_i (m_i - m_{i-1})^2. \tag{3.5}$$

como una caracterísitica adicional motivado por 3.2.

El valor Abbe $\mathcal{A}$ puede ser calculado en subintervalos de tiempo y puede dar cuenta de la escala de tiempo en la cual una curva de luz varía. Consideremos un valor fijo Δt y para cada instante t_i podemos definir $S_i = \{k | t_k \in (t_i - \Delta t/2, t_i + \Delta t/2)\}$. Con esto para cada t_i podemos calcular el valor Abbe utilizando exclusivamente $\{m_k\}_{k \in S_i}$, que son las mediciones hechas en el intervalo de tiempo $(t_i - \Delta t/2, t_i + \Delta t/2)$. Llamemos a cada uno de estos valores calculados $\mathcal{A}_{t,i}$ y definamos

$$\bar{\mathcal{A}}_t = \frac{1}{n}\sum_{i=1}^{n} \mathcal{A}_{t,i} \tag{3.6}$$

donde el subíndice t hace referencia a la escogencia de Δt y n es el número de observaciones disponibles para la curva de luz en cuestión. Si una curva de luz varía en escalas de tiempo mayores que Δt, es de esperar que $\bar{\mathcal{A}}_t \approx 1$ y $\bar{\mathcal{A}}_t < 1$ de lo contrario. La escala de tiempo a la cual varían las curvas

cambia ampliamente entre clases de variabilidad estelar. Esta puede ser de unos pocos minutos para estrellas δ-Scuti a unos cuantos años para variables de largo periodo. Una cruva de luz puede ser representada como un punto en los planos$\mathcal{A} - \bar{\mathcal{A}}_t$ y su posición depende de la relación entre la escala a la que cambia la magnitud y Δt. Como la escala de tiempo a la que cambian las magnitudes de las curvas de luz está relacionada con el tipo de variabilidad estelar, es de esperar que estas variables puedan ser utilizadas para realizar clasificación. Para cubrir las diferentes escalas de tiempo en las cuales cambian las magnitudes de las curvas de luz elegimos los valores $\Delta t =, 10d, 20d, 50d, 100d, 200d, 500d, 750d.$.

pensar en plots aquí

La entropía de Shannon [31] mide qué tan sorpresiva es una realización de una variable aleatoria y puede ser estimada eficientemente. La entropía es un funcional de la densidad de probabilidad, es convexa, alcanza su máximo cuando la variable aleatoria se distribuye uniformemente y es mínima cuando uno la variable aleatoria toma uno de sus valores con probabilidad 1. Shannon demostró que bajo ciertas condiciones es el único funcional de la densidad de probabilidad que puede dar cuenta de la sorpresa que causa una realización de la variable aleatoria [31][6]. Intuitivamente la entropía de una variable que toma pocos valores con gran probabilidad será menor que la de una variable que toma muchos valores con probabilidades parecidas, por ejemplo es de esperar que la entropía la serie de magnitudes de curva de luz como sea mayor que la de una curva como . Para una variable aleatoria continua M se define como

figura con poca entropía

figura con mucha entropia

$$H(M) = -\int f_M(m) \log_2 f_M(m) dm \tag{3.7}$$

donde f_M es la densidad de la variable M, sin embargo esta definición tiene propiedades inusuales, pues depende de las coordenadas en que se calcule. Con frecuencia la entropía de Shannon se estima discretizando la variable M. Se realiza una partición de los posibles valores de M en n_p partes y se estima la probabilidad p_k de que la variable tome valores en la k-ésima partición con

[6]En el contexto de comunicaciones, Shannon demostró que si M representa los posibles símbolos que produce una fuente de mensajes, la entropía de M medida en bits/símbolo es la longitud mínima promedio de un código que represente los mensajes producidos por M, es decir, la entropía $H(M)$ es una cota inferior para la compresibilidad del resultado de M

el la fracción de los datos que se encuentra esa partición y se estima

$$H(M) \approx -\sum_{k=1}^{n_p} p_i \log_2 p_i. \tag{3.8}$$

A esta cantidad nos referimos como entropía de la partición, $H(M, n_p)$. Para realizar estas estimaciones utilizamos $n_p = 10$ particiones regulares entre el máximo m_i y el mínimo m_i.

La entropía de Rényi [26] es otra medida de la sorpresa causada por una realización de una variable aleatoria que satisface un conjunto de axiomas diferente a la entropía de Shannon [26]. Las probabilidades p_k se definen de la misma manera como la probabilidad de pertenecer a una de las n_p particiones y la entropía de Renyí de una partición se definida como

$$H_\alpha(M, n_p) = \frac{1}{1-\alpha} \log_2 \sum_x p_i^\alpha. \tag{3.9}$$

H_α tiene propiedades similares a la entropía de shannon: para $\alpha > 1$ alcanza su máximo cuando M se distribuye uniformemente y su mínimo cuando M toma un valor con probabilidad 1. Cuando $\alpha \to 1$, la entropía de Rényi tiende a la entropía de Shannon; cuando $\alpha \to 0$ a la función constante 1 y cuando $\alpha \to \infty$,

$$H_\infty(M, n_p) = -\log_2 \max_i p_i. \tag{3.10}$$

Así, a medida que aumenta α, H_α está determinada por los valores más probables. Para explorar la posibilidad de utilziar adicionalmente esta caracterísitca para realizar clasificación, utilizamos $\alpha = 2, 5, 10, \infty$. .

pensar en gráficas

Se pueden separar las características en dos grupos: las que dependen de la escala en la que se mide la magnitud y las que no. La media, la desviación estándar y el rango dependen de la escala de las variaciones de la magnitud de las curvas. Por ejemplo las curvas de luz de Cefeidas y de Cefeidas tipo 2 tienen formas muy aprecidas, sin embargo se diferencian en que las Cefeidas son más brillantes, por lo que se espera que las variables que dependen de la escala sirvan para discriminar entre estas dos. Por su parte el sesgo, la curtosis, los valores Abbe y las entropías de Shannon y Rényi dependen únicamente de la forma de la de la denisidad de magnitudes. Como es de esperar que la forma de la distribución de magnitudes de una curva de luz esté relacionada con su morfología, creemos que estos parámetros pueden

ayudar a discriminar entre curvas de luz que tienen formas diferentes aunque tomen valores de magnitud similares.

Tras calcular estas cantidades obtuvimos un vector de características de 20 componentes, por lo que surge la pregunta de si algunas de estas variables son redundantes. Para decidir si algunas variables son redundantes o no podemos analizar la matriz de correlaciones y Análisis de Componentes Principales (ACP). En la figura mostramos la matriz de correlaciones de Spearman [7]. Se ve que las medidas de entropía utilizadas están positivamente correlacionadas entre sí, de la misma forma que los valores Abbe promedio. Por otro lado la curtosis y las medidas de entropía están correlacionadas negativamente.

figura con la correlación de Spearman

Tras realizar una descomposición en varlores singulares de los datos, vemos que se puede

La correlación de Spearman entre dos variables aleatorias es una medida de correlación que varía entre -1 y 1. Toma como valor 1 si las variables aleatorias consideradas están relacionandas por un

3.3. Clasificación

3.3.1. Árboles de Clasificación

3.3.2. Bosques Aleatorios

3.3.3. k Vecinos Más Cercanos

3.3.4. Máquinas de Soporte Vectorial

[7] La correlación de Spearman es una medida de correlación que toma valores -1 y 1 cuando dos variables están relacionadas por una función monótona creciente y decreciente respectivamente y valores intermedios en las demás situaciones. Para una dos muestras $\{x_1, \ldots, x_n\}$ y $\{y_1, \ldots, y_n\}$ la correlación de Spearman es $\frac{6\sum_i (r_i - s_i)}{n(n^2-1)}$ donde r_i y s_i son el lugar que ocupan x_i y y_i al ordenar las muestras $\{x_1, \ldots, x_n\}$ y $\{y_1, \ldots, y_n\}$. r_i y s_i son llamados rangos.

Cuadro 3.4: Matriz de confusión para CART

	becand	cep	dcst	ebs	lpv	rrlyr	t2cep
becand	470	12	6	1039	8906	47	18
cep	0	6210	12	18	320	3098	92
dcst	0	35	2511	5538	91	1615	1
ebs	0	1	148	21346	346	64	1
lpv	5	107	24	1762	304810	100	3
rrlyr	0	648	82	552	13880	34313	92
t2cep	0	991	5	2004	15429	4980	396

Cuadro 3.5: Tasas de clasificación estimadas por validación cruzada de 10 iteraciones

	becand	cep	dcst	ebs	lpv	rrlyr	t2cep
becand	0.99	0.00	0.00	0.03	0.03	0.00	0.03
cep	0.00	0.78	0.00	0.00	0.00	0.07	0.15
dcst	0.00	0.00	0.90	0.17	0.00	0.04	0.00
ebs	0.00	0.00	0.05	0.66	0.00	0.00	0.00
lpv	0.01	0.01	0.01	0.05	0.89	0.00	0.00
rrlyr	0.00	0.08	0.03	0.02	0.04	0.78	0.15
t2cep	0.00	0.12	0.00	0.06	0.04	0.11	0.66

Cuadro 3.6: k = 1

	becand	cep	dcst	ebs	lpv	rrlyr	t2cep
becand	381	1	0	50	41	0	0
cep	1	6595	4	172	54	986	87
dcst	0	7	2064	340	13	151	0
ebs	51	184	536	30129	573	668	29
lpv	42	187	19	944	342740	371	158
rrlyr	0	971	165	595	308	41911	217
t2cep	0	59	0	29	53	130	112

Cuadro 3.7: k=1, validación cruzada de 10 iteraciones

	becand	cep	dcst	ebs	lpv	rrlyr	t2cep
becand	0.80	0.00	0.00	0.00	0.00	0.00	0.00
cep	0.00	0.82	0.00	0.01	0.00	0.02	0.14
dcst	0.00	0.00	0.74	0.01	0.00	0.00	0.00
ebs	0.11	0.02	0.19	0.93	0.00	0.02	0.05
lpv	0.09	0.02	0.01	0.03	1.00	0.01	0.26
rrlyr	0.00	0.12	0.06	0.02	0.00	0.95	0.36
t2cep	0.00	0.01	0.00	0.00	0.00	0.00	0.19

Cuadro 3.8: $\gamma = 0{,}1$, costo = 16

	becand	cep	dcst	ebs	lpv	rrlyr	t2cep
becand	297	0	2	26	16	0	0
cep	0	4622	0	55	85	788	69
dcst	0	0	1683	163	4	47	0
ebs	89	48	865	29439	198	563	5
lpv	89	490	22	1380	342823	1113	255
rrlyr	0	2844	216	1196	656	41706	274
t2cep	0	0	0	0	0	0	0

Cuadro 3.9: $\gamma = 0{,}1$, costo = 16, validación cruzada de 10 iteraciones

	becand	cep	dcst	ebs	lpv	rrlyr	t2cep
becand	0.63	0.00	0.00	0.00	0.00	0.00	0.00
cep	0.00	0.58	0.00	0.00	0.00	0.02	0.11
dcst	0.00	0.00	0.60	0.01	0.00	0.00	0.00
ebs	0.19	0.01	0.31	0.91	0.00	0.01	0.01
lpv	0.19	0.06	0.01	0.04	1.00	0.03	0.42
rrlyr	0.00	0.36	0.08	0.04	0.00	0.94	0.45
t2cep	0.00	0.00	0.00	0.00	0.00	0.00	0.00

Bibliografía

[1] Alina Beygelzimer, Sham Kakade, and John Langford. Cover trees for nearest neighbor. In *Proceedings of the 23rd international conference on Machine learning*, pages 97–104. ACM, 2006.

[2] Gérard Biau, Luc Devroye, and Gábor Lugosi. Consistency of random forests and other averaging classifiers. *The Journal of Machine Learning Research*, 9:2015–2033, 2008.

[3] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[4] Leo Breiman. Bagging predictors. *Machine learning*, 24(2):123–140, 1996.

[5] Leo Breiman. Random forests. *Machine learning*, 45(1):5–32, 2001.

[6] Leo Breiman, Jerome Friedman, Charles J. Stone, and R. A. Olshen. *Classification and Regression Trees*. Chapman and Hall/CRC, New York, N.Y., 1 edition edition, January 1984.

[7] Leo Breiman and others. Heuristics of instability and stabilization in model selection. *The annals of statistics*, 24(6):2350–2383, 1996.

[8] BSJ. Types of Variables, June 2012.

[9] Pablo M. Cincotta, Mariano Mendez, and Josue A. Nuñez. Astronomical time series analysis. I. A search for periodicity using information entropy. *The Astrophysical Journal*, 449:231, 1995.

[10] Corinna Cortes and Vladimir Vapnik. Support-vector networks. *Machine learning*, 20(3):273–297, 1995.

[11] Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[12] Jonas Debosscher, L. M. Sarro, Conny Aerts, J. Cuypers, Bart Vandenbussche, R. Garrido, and E. Solano. Automated supervised classification of variable stars-I. Methodology. *Astronomy & Astrophysics*, 475(3):1159–1183, 2007.

[13] Luc Devroye. *A probabilistic theory of pattern recognition*, volume 31. Springer Science & Business Media, 1996.

[14] Evelyn Fix and Joseph L. Hodges Jr. Discriminatory analysis-nonparametric discrimination: consistency properties. Technical report, DTIC Document, 1951.

[15] D. Graczyk, I. Soszyński, R. Poleski, G. Pietrzyński, A. Udalski, M. K. Szymański, M. Kubiak, Ł. Wyrzykowski, and K. Ulaczyk. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. XII. Eclipsing Binary Stars in the Large Magellanic Cloud. *Acta Astronomica*, 61:103–122, June 2011.

[16] Hannu Karttunen, Pekka Kröger, Heikki Oja, Markku Poutanen, and Karl Johan Donner, editors. *Fundamental Astronomy*. Springer, Berlin ; New York, 5th edition edition, August 2007.

[17] Andy Liaw and Matthew Wiener. Classification and regression by randomforest. *R News*, 2(3):18–22, 2002.

[18] James Mercer. Functions of positive and negative type, and their connection with the theory of integral equations. *Philosophical transactions of the royal society of London. Series A, containing papers of a mathematical or physical character*, pages 415–446, 1909.

[19] David Meyer, Evgenia Dimitriadou, Kurt Hornik, Andreas Weingessel, and Friedrich Leisch. *e1071: Misc Functions of the Department of Statistics (e1071), TU Wien*, 2014. R package version 1.6-4.

[20] N. Mowlavi. Searching transients in large-scale surveys. A method based on the Abbe value. *Astronomy and Astrophysics*, 568:78, 2014.

[21] M. Pawlak, D. Graczyk, I. Soszyński, P. Pietrukowicz, R. Poleski, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, S. Kozłowski, and J. Skowron. Eclipsing Binary Stars in the OGLE-III Fields of the Small Magellanic Cloud. *Acta Astronomica*, 63:323–338, September 2013.

[22] R. Poleski, I. Soszyński, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, O. Szewczyk, and K. Ulaczyk. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. VI. Delta Scuti Stars in the Large Magellanic Cloud. *Acta Astronomica*, 60:1–16, March 2010.

[23] R Core Team. *R: A Language and Environment for Statistical Computing*. R Foundation for Statistical Computing, Vienna, Austria, 2014.

[24] Joseph W. Richards, Dan L. Starr, Nathaniel R. Butler, Joshua S. Bloom, John M. Brewer, Arien Crellin-Quick, Justin Higgins, Rachel Kennedy, and Maxime Rischard. On Machine-learned Classification of Variable Stars with Sparse and Noisy Time-series Data. *The Astrophysical Journal*, 733(1):10, May 2011.

[25] Bayron Stevenson Rodríguez Feliciano and José Alejandro García Varela. *Análisis estadístico en poblaciones de estrellas variables*. Tesis (Físico). Universidad de los Andes. Bogotá : Uniandes, 2012., 2012.

[26] Alfréd Rényi and others. On Measures of Entropy and Information. In *Proceedings of the Fourth Berkeley Symposium on Mathematical Statistics and Probability, Volume 1: Contributions to the Theory of Statistics*. The Regents of the University of California, 1961.

[27] B. E. Sabogal, A. García-Varela, and R. E. Mennickent. Search for Southern Galactic Be Star Candidates. *Publications of the Astronomical Society of the Pacific*, 126:219–226, 2014.

[28] Richard J. Samworth. Optimal weighted nearest neighbour classifiers. *The Annals of Statistics*, 40(5):2733–2763, October 2012.

[29] L. M. Sarro, Jonas Debosscher, M. López, and Conny Aerts. Automated supervised classification of variable stars-II. Application to the OGLE database. *Astronomy & Astrophysics*, 494(2):739–768, 2009.

[30] Jeffrey D. Scargle. Studies in astronomical time series analysis. II-Statistical aspects of spectral analysis of unevenly spaced data. *The Astrophysical Journal*, 263:835–853, 1982.

[31] C.E. Shannon. A mathematical theory of communication. *Bell System Technical Journal, The*, 27(3):379–423, July 1948.

[32] Bernard W. Silverman and M. Christopher Jones. E. Fix and JL Hodges (1951): An important contribution to nonparametric discriminant analysis and density estimation: Commentary on Fix and Hodges (1951). *International Statistical Review/Revue Internationale de Statistique*, pages 233–238, 1989.

[33] I. Soszyński, W. A. Dziembowski, A. Udalski, R. Poleski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, S. Kozłowski, and P. Pietrukowicz. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. XI. RR Lyrae Stars in the Galactic Bulge. *Acta Astronomica*, 61:1–23, March 2011.

[34] I. Soszyński, R. Poleski, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, L. Wyrzykowski, O. Szewczyk, and K. Ulaczyk. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. I. Classical Cepheids in the Large Magellanic Cloud. *Acta Astronomica*, 58:163–185, September 2008.

[35] I. Soszyński, R. Poleski, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, O. Szewczyk, and K. Ulaczyk. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. VII. Classical Cepheids in the Small Magellanic Cloud. *Acta Astronomica*, 60:17–39, March 2010.

[36] I. Soszyński, A. Udalski, P. Pietrukowicz, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, R. Poleski, and S. Kozłowski. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. XIV. Classical and TypeII Cepheids in the Galactic Bulge. *Acta Astronomica*, 61:285–301, December 2011.

[37] I. Soszyński, A. Udalski, P. Pietrukowicz, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, R. Poleski, and

S. Kozłowski. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. Type II Cepheids in the Galactic Bulge - Supplement. *Acta Astronomica*, 63:37–40, March 2013.

[38] I. Soszyński, A. Udalski, M. K. Szymański, J. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, and R. Poleski. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. IX. RR Lyr Stars in the Small Magellanic Cloud. *Acta Astronomica*, 60:165–178, September 2010.

[39] I. Soszyński, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, O. Szewczyk, K. Ulaczyk, and R. Poleski. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. II.Type II Cepheids and Anomalous Cepheids in the Large Magellanic Cloud. *Acta Astronomica*, 58:293, December 2008.

[40] I. Soszyński, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, O. Szewczyk, K. Ulaczyk, and R. Poleski. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. III. RR Lyrae Stars in the Large Magellanic Cloud. *Acta Astronomica*, 59:1–18, March 2009.

[41] I. Soszyński, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, O. Szewczyk, K. Ulaczyk, and R. Poleski. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. IV. Long-Period Variables in the Large Magellanic Cloud. *Acta Astronomica*, 59:239–253, September 2009.

[42] I. Soszyński, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, and R. Poleski. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. VIII. Type II Cepheids in the Small Magellanic Cloud. *Acta Astronomica*, 60:91–107, June 2010.

[43] I. Soszyński, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, R. Poleski, S. Kozłowski, and P. Pietrukowicz. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. XIII. Long-Period Variables in the Small Magellanic Cloud. *Acta Astronomica*, 61:217–230, September 2011.

[44] I. Soszyński, A. Udalski, M. K. Szymański, M. Kubiak, G. Pietrzyński, Ł. Wyrzykowski, K. Ulaczyk, R. Poleski, S. Kozłowski, P. Pietrukowicz, and J. Skowron. The Optical Gravitational Lensing Experiment. The OGLE-III Catalog of Variable Stars. XV. Long-Period Variables in the Galactic Bulge. *Acta Astronomica*, 63:21–36, March 2013.

[45] Ingo Steinwart. Consistency of support vector machines and other regularized kernel classifiers. *Information Theory, IEEE Transactions on*, 51(1):128–142, 2005.

[46] Carolin Strobl, Anne-Laure Boulesteix, Thomas Kneib, Thomas Augustin, and Achim Zeileis. Conditional variable importance for random forests. *BMC bioinformatics*, 9(1):307, 2008.

[47] Terry Therneau, Beth Atkinson, and Brian Ripley. *rpart: Recursive Partitioning and Regression Trees*, 2014. R package version 4.1-8.

[48] A. Udalski. The Optical Gravitational Lensing Experiment. Real Time Data Analysis Systems in the OGLE-III Survey. *Acta Astron.*, 53(astro-ph/0401123):291, 2004.

[49] A. Udalski, M. K. Szymanski, I. Soszynski, and R. Poleski. The Optical Gravitational Lensing Experiment. Final Reductions of the OGLE-III Data. *Acta Astronomica*, 58:69–87, 2008.