Universidad de la República
Facultad de Ingeniería

Int. al Reconocimiento de Patrones

PROYECTO FINAL DE CURSO

DETECCIÓN DE CONSUMOS ANOMALOS

Integrantes

Federico Decia
Matías Di Martino
Juan I. Molinelli

Tutor: Ing. Alicia Fernandez

Montevideo - Uruguay
Diciembre 2010

Índice general

Índice general **2**

1. Introducción **3**

2. Pre-Procesamiento y Caracterización **5**
2.1. Pre-Procesamiento 5
2.2. Características 6
2.2.1. Descripción de las características 7

3. SVM **13**
3.1. Clasificador C-SVM 14
3.1.1. C-SVM: Soft Margin 14
3.1.2. Elección de los Parámetros 15
3.2. Evaluación del clasificador 19
3.3. One-Class SVM 22
3.3.1. SVM One Class : separando los datos del origen 23
3.3.2. Elección de los parámetros 24
3.3.3. Evaluación del clasificador 29

4. OPF **30**
4.1. Descripción del método 30
4.2. Aplicación del método 32
4.3. Resultados OPF sin aprendizaje 32
4.3.1. Conjunto 1 32
4.3.2. Conjunto 2 33
4.3.3. Conjunto 3 33
4.4. Resultados OPF con aprendizaje 34

5. Conclusiones **35**
5.1. Evaluación de los resultados 35
5.2. Trabajos a futuro 36

Bibliografía **40**

Capítulo 1

Introducción

Descripción del Problema

El uso irregular o fraudulento de la energía eléctrica representa un problema de gran magnitud provocando cuantiosas pérdidas a las empresas distribuidoras de muchos países o regiones. En el caso de Montevideo los balances de energía arrojan valores elevados de pérdidas totales. De todas estas, el trabajo se enfocará en las pérdidas por fraude por lo que no consideramos las pérdidas técnicas en la red de distribución y las pérdidas asociadas a las zonas carenciadas, las primeras por tener un origen ajeno al objeto de análisis, y las últimas por tratarse de pérdidas ya plenamente identificadas.

Las pérdidas por fraude representan un subconjunto minoritario de clientes en cuyos suministros existen irregularidades que no permiten el correcto registro del total de la energía consumida.Su detección es una tarea que requiere mucho trabajo ya que se debe estudiar caso por caso cada usuario para identificar posibles anomalías. Esto lo torna un problema difícil de combatir por el número elevado de clientes que se debe inspeccionar. Debido a esta importante característica del problema, se separa a los clientes en niveles según su patrón de consumo (fabricas de gran tamaño, fábricas de menor tamaño, supermercados, viviendas, etc.) y en base a esos niveles, la importancia que se le da a la detección del mencionado fraude; consumidores de mayores consumos (mayor nivel) reciben inspecciones con mayor regularidad. Este proyecto busca analizar el consumo de los usuarios en un determinado periodo de tiempo (estudiando distintas características de los mismos y del tipo de usuario que se trata) procurando clasificarlos como usuarios candidatos a ser fraudulentos o no. Los usuarios etiquetados como candidatos no necesariamente serán fraudulentos pero ameritarán una inspección. De esta forma se acota el conjunto de usuarios a los cuales se les debe efectuar una inspección y se busca que la detección de fraudulentos sea más eficiente. Se comienza trabajando con una herramienta desarrollada para detectar consumos anómalos de un conjunto acotado y muy particular de clientes (Autoservicios etiquetados) procurando desarrollar una herramienta más robusta capaz de detectar consumos anómalos en conjuntos de patrones más amplios.

La herramienta que se pretende desarrollar, podrá finalmente ser utilizada por los técnicos de UTE para facilitar y mejorar la clasificación de los consumos que actualmente se realiza de forma manual. Cabe señalar que no se pretende generar un algoritmo capaz de detectar de manera automática todos los consumos anómalos de manera precisa ni se pretende obtener una descripción cerrada de los atributos que presentan los consumos anómalos. Se pretende

desarrollar una herramienta que facilite, haga más eficiente y eficaz los esfuerzos realizados por el personal de UTE con el objetivo de detectar fraudes, generar una lista *más reducida* de consumidores a inspeccionar donde se concentren la *gran mayoría* de infractores. De este modo se lograra un mayor aprovechamiento de los recursos humanos y (se espera) un mejor porcentaje de clientes fraudulentos detectados.

En relación al párrafo anterior en el capítulo siguiente se mostrará que este problema tiene algunas características particulares que dificultan el diseño de una herramienta que de una solución categórica, sobre todo debido a como se generan las características. Si bien esto genera algunas dificultades adicionales y requieren de algunas consideraciones particulares que no son tenidas en cuenta en los abordajes típicos de problemas de reconocimiento de patrones, no representan una imposibilidad de llevar adelante el trabajo con buenos resultados [15][6].

Capítulo 2

Pre-Procesamiento y Caracterización

2.1. Pre-Procesamiento

Antes de empezar a desarrollar la herramienta para clasificar los patrones es muy importante analizar cual va a ser la entrada de nuestro sistema, los consumos y sus etiquetas. En nuestro caso contamos con 2 bases de datos, los primeros son los utilizados en el trabajo previo de Kosut y Alcetegaray [15] que corresponden a una porción muy especial de autoservicios (alrededor de 500).La idea de esta base fue probar la viabilidad de utilizar técnicas de reconocimiento de patrones para clasificar consumos, por lo que se intentó que los consumos que allí estuvieran fuesen los más aptos para lograr una buena performance. La segunda base de datos corresponde a una cantidad un poco mayor (alrededor de 1500) de consumos industriales de distinta índole, a la cual no se le realizó ninguna pre-selección previa.

Las bases de datos están compuestas en primer lugar por los consumos de los clientes tomados por UTE. Las medidas corresponden a la energía total consumida por cada cliente en el intervalo de un mes, comenzando en Octubre de 2004 y terminado en Setiembre de 2009. Las medidas son realizadas por las visitas de los técnicos de UTE a los hogares y esta forma de tomar los datos es la que introduce nuestro primer problema. El personal pasa por las casas una vez por mes, pero no cada 30 dias, por lo cual los datos tienen un cierto ruido. Por ejemplo si una persona consume exactamente lo mismo todos los meses, los consumos pueden no mostrar eso, dependiendo el día en que fueron tomados. Otro problema importante es que hay veces que la persona que va a la casa de los clientes no puede acceder a la medida, por ejemplo porque no se encuentra nadie en la misma. En estos casos la UTE estima el consumo a partir del histórico y luego se corrige cuando se logra acceder al consumo real. Estos 2 problemas hacen que los consumos de los usuarios puedan no representar de forma fidedigna el comportamiento de los mismos.

Aparte del consumo mensual de cada cliente la base de datos cuenta con una información que para nosotros es imprescindible, las etiquetas. En los problemas de reconocimiento de patrones son muy importantes ya que nos dicen a que clase (normal o anómalo en nuestro caso) pertenecen los consumos, lo cual vamos a usar para entrenar el clasificador y para evaluar su performance. Las etiquetas son datos externos al problema las cuales son suministradas por expertos en el tema. En nuestro caso fueron los ingenieros de UTE quienes miraron los consumos y los etiquetaron de manera manual, en base a experiencia acumulada y a criterios que se establecen en la empresa que caracterizan a un posible cliente

fraudulento. Por otro lado, también se dispone de la información de los consumos a los cuales se les constató una irregularidad mediante una inspección (esta puede ser una inspección debido a una etiqueta previa o simplemente una de rutina), por lo que a estos consumos también se los etiquetó como anómalos.

La forma descrita en el párrafo anterior de generar las etiquetas tiene 2 problemas fundamentales. El primero es que, a pesar de que los que realizaron el etiquetado son expertos en el tema, la decisión es bastante subjetiva y para nada contundente en algunos casos. También puede ser que el etiquetado haya sido realizado teniendo en cuenta factores más allá de los consumos. Esto sin duda puede confundir al clasificador a la hora de ser entrenado en las propiedades de cada clase. El segundo problema es que se tienen registros de las irregularidades detectadas pero eso no quiere decir que el resto no esté cometiendo faltas. Es seguro que algunos de los consumos de los que se asume por defecto son normales, podrían a priori ser consumos fraudulentos a los que simplemente aun no se ha inspeccionado o se los ha inspeccionado pero por las naturalezas del ilícito no se ha podido detectar el mismo durante la inspección. Los dos etiquetados tienen sus particularidades, partiremos de la hipótesis de que combinar las 2 formas es la mejor solución al problema descrito anteriormente.

2.2. Características

El éxito en todo proceso de reconocimiento de patrones, depende en buena medida, de qué tan bien se entienda el problema. Es vital intentar desarrollar el conocimiento y la habilidad que posee el experto, y que le permite clasificar los consumos al observarlos. Ubicarse en el trabajo de los expertos y entrar en el contexto en el que realizan la clasificación, tratar de entender qué variables entran en juego qué información es tenida en cuenta (de manera consciente e inconsciente) y cómo se combinan para desembocar en una decisión.

Es importante tratar de entender la globalidad del problema, y luego plasmar esta realidad en *características* medibles. Si se cuenta con baterías de métodos de extracción y selección de características, pero estos parten de la base de que ya se tiene una representación (en algún espacio) de los patrones. Es esta primera representación la que decimos que no cuenta con un proceso formal y un marco teórico bien establecido para su elaboración.

A lo largo de este capitulo, se describirán brevemente las características que se utilizaran. También se tratara de transmitir los motivos fundamentales que inspiraron la consideración de cada característica y que tipos de fraude pretenden contemplar.

En una primera aproximación al problema, se consulto como se había abordado la temática en otras partes del mundo, los primeros paper consultados en esta etapa fueron [9] [15] [1] y [6] (entre otros). No se encontró en la bibliografía consultada consenso desde el punto de vista de las características que se deben utilizar. Además, las características que se pueden utilizar, están atadas en buena medida a las metodologías de cada compañía eléctrica. En este caso de estudio, no se posee con los recursos disponibles para obtener algunas de las características utilizadas en los trabajos citados, motivo por el cual, se tuvieron que plantear características esencialmente nuevas, inspiradas en algunos casos en trabajos previos y en otros casos obtenidas de la información provista en las reuniones con los técnicos.

Notación: utilizaremos $C(i)$ para referirnos al valor de un cierto consumo en el mes i

2.2.1. Descripción de las características

Cociente entre los valores medios

Una de las formas en que se puede manifestar un consumo fraudulento, es mediante un cambio en el valor medio del consumo. Lógicamente cuando se efectúa un fraude, la intención es que el autor se vea beneficiado mediante una reducción en los pagos mensuales. Para lograr ese "*ahorro*" en las facturas, procurara por distintos medios, disminuir el valor medio de su consumo mensual.

Para detectar el cambio en el consumo promedio, es que su utiliza este vector de características, con la finalidad de observar el consumo medio a distintas escalas

Como primeras tres características, se utilizará la comparación de la media en los últimos 3, 6 y 12 meses con la media anterior del consumo. Es decir:

$$car1 = \frac{mean(C([1:n-4]))}{mean(C[n-3:n])} \tag{2.1}$$

$$car2 = \frac{mean(C([1:n-7]))}{mean(C[n-6:n])} \tag{2.2}$$

$$car2 = \frac{mean(C([1:n-13]))}{mean(C[n-12:n])} \tag{2.3}$$

donde n es el número de meses considerados en el consumo.

Norma de la diferencia entre el consumo esperado y el consumo real

Por medio de esta característica se pretende observar cambios en el comportamiento del usuario. La idea es comparar cada uno de los valores de consumo de cada mes del último año, con el valor de mismo mes del año anterior. La comparación se corrige con un factor correspondiente al cociente entre las medias para independizar las medidas del valor medio. El factor de corrección pretende eliminar variaciones debidas a cambios en la temperatura media en uno y otro año, que pueden trasladar verticalmente los consumos. Finalmente sumamos el error cometido durante los últimos 12 meses del consumo.

$$car4 = \sqrt{\sum_{i=n-11}^{i=n} \left(C(i) - \alpha C(i-12)\right)^2} \tag{2.4}$$

Donde n es el número total de meses que consideramos para los consumos y α corresponde al cociente entre el consumo medio del último año y el consumo medio en el penúltimo año.

Si bien se pierde algo de información al no considerar individualmente el error cometido mes a mes (como se hizo originalmente en el primer encare del problema) se disminuye significativamente el número de características.

Diferencia en los espectros

Otro de los aspectos importantes a la hora de distinguir entre consumos, puede ser como es el espectro de los mismos. Es decir, ¿qué componentes de frecuencia predominan?,

¿existe alguna periodicidad?, ¿se presentan cambios en el valor de continua?. Estas y otras preguntas se pueden responder simplemente mirando en el espacio de frecuencia los distintos consumos. Como quinta característica, se propone considerar la diferencia entre los coeficientes de Fourier de la curva de consumos del último año, con la curva de los años anteriores. Nuevamente se condensara la diferencia en el espectro de la señal en una única característica, en lugar de ver coeficiente a coeficiente las variaciones.

$$car5 = \|FFT(C_{actual}) - FFT(C_{medio})\| \tag{2.5}$$

C_{actual} representa el vector de consumos durante el último año, y C_{medio} el vector promedio del consumo en los años anteriores.

Diferencia en los coeficientes wavelet

Otra manera de ver el comportamiento de una señal, es a través de otra transformación llamada Wavelets. Al igual que la Transformada de Fourier (o su versión discreta DFT) la idea es representar a la señal en un espacio distinto al del tiempo en el cual se pueden resaltar, desde otro punto de vista distintas cualidades que en el espacio del tiempo quedan ocultas o pasan desapercibidas. Al igual que en otras transformaciones (como puede ser la TdF), la idea es proyectar la señal original en un espacio nuevo. En el caso de la transformada de wavelets, se utiliza una función $\psi(x)$ con determinadas características[a] llamada onda madre, y la base $\{\psi_{ij}(x)\}$ del espacio en que se va a proyectar se forma por medio de compresiones y dilataciones de la onda madre además de corrimientos $\mathcal{B} = \{\psi_{ij}(x) = \psi(i\,x - j) \text{ con } i, j \in \mathbb{Z}\}$. Esta transformación es muy usada en algunas de las referencias consultadas (por ejemplo [9]). La utilización de wavelets parece razonable por la buena capacidad que tiene este tipo de transformación para distinguir cambios abruptos y acotados en el tiempo. La separación de la señal en los coeficientes de ajuste y los coeficientes de detalle, permite obtener una visión del consumo a distintas escalas y detectar cambios bruscos.

Se utilizara la norma del vector diferencia en los coeficientes en lugar de considerar cada uno de los coeficientes individualmente. Nuevamente se apela a reducir la dimensión del espacio de características tratando de condensar la mayor cantidad de información posible.

$$\vec{car}6 = \|DWT\,(C_{actual}) - DWT\,(C_{medio})\| \tag{2.6}$$

C_{actual} representa el vector de consumos durante el último año, y C_{medio} el vector promedio del consumo en los años anteriores.

Por último cabe señalar, que se pueden utilizar distintas funciones madre $\psi(x)$ para definir la transformación, entre ellas se encuentras las ”Haar wavelets” , ”Daubenchies wavelets”, ”Biorthogonal wavelets”, ”Mexican hat wavelets” y ”Shannon wavelets”. Se utilizaran en particular las ”Harr wavelets” por ser las más simples desde el punto de vista conceptual, pero esto no representa ninguna limitación importante porque simplemente cambiando un parámetro se puede seleccionar otras bases sin mayores inconvenientes.

[a] para más detalle se puede consultar [5], [10]

Diferencia en el ajuste de un polinomio de grado N

Continuando con la caracterización de los consumos, se propone *aproximar*[b] los datos por un polinomio de grado N. Se pueden distinguir distintos casos en función de el valor que se elija para el grado del polinomio. A modo de ejemplo para $N = 1$ se estaría estudiando el valor medio del consumo mensual, para $N = 2$ se aproximan las curvas de consumo por una recta. Este último caso, es una de las características propuestas en [15] y que inspiro que se utilizara la aproximación por un polinomio (como generalización de ésta).

Se ajusta por medio de mínimos cuadrados, el polinomio de grado N (en este caso se usara $N = 4$) que mejor aproxima cada año de la curva de consumos. Luego, se obtiene la diferencia entre el promedio de los primeros años y el último.

$$car\{7, 8, 9, 10, 11\} = polyfit(C_{ultimo\ ao}) - \frac{1}{n-1}\sum_{i=1}^{n-1} polyfit(C_{ao\ i}) \tag{2.7}$$

Donde n es el número de años considerados

Distancia al consumo medio

Se parte de la base de que los consumos fraudulentos son muchos menos que los consumos normales. Además, los consumos fraudulentos, se puede asumir que no tienen una fuerte correlación pues los tipos de fraude que se pueden cometer son variados y se realizan en momentos independientes. Tomando dichas hipótesis, es razonable suponer que si se realiza el promedio de todos los consumos mes a mes, se obtendrá un consumo que representara en buena medida a un consumo *típico* o *normal*. Con esta idea en mente, parece razonable tomar como característica la distancia de cada consumo a dicho consumo medio, como una medida de que alejado o cercano esta cada consumo al consumo medio.

$$car12 = \left\| \vec{C} - \vec{C}_m \right\| \tag{2.8}$$

Donde $\vec{C}_m$ es el consumo promedio (mes a mes) entre todos los consumos de la base de datos.

Comparación de la varianza del consumo

La demanda de energía eléctrica por parte de los usuarios presenta mayores o menores fluctuaciones dependiendo de la naturaleza del consumidor pero en ninguno de los casos el consumo es constante. En una de las reuniones con los expertos de UTE se observo que, algunos de los clientes que adulteraban el contador o realizaban *puentes* en los bornes del mismo *ajustaban* el consumo que registraba el contador. El objetivo es obviamente disminuir el monto de las facturas. Para disimular el fraude por momentos los usufructuarios conectan el contador y de manera aproximada trataban de presentar consumos parejos a los largo de los meses. Se observo que este ajuste manual hace que las curvas de consumo presenten menores fluctuaciones que los consumos normales, en los cuales el cliente no tiene un mecanismo de control del consumo. Por esta razón consumos con varianzas muy bajas y valores muy parejos de consumo a lo largo de los meses pueden estar escondiendo fraudes del tipo descrito anteriormente.

[b] utilizando mínimos cuadrados

Inspirados en las observaciones anteriores se propone considerar dos características, en primer lugar el cociente entre la varianza que presenta el consumo promedio (año a año) con la varianza en el último período; en segundo lugar, la varianza promedio de **todos los consumos** con la varianza de cada consumo en el último año.

$$car13 = \frac{var\left(C_N,\right)}{var\left(\frac{1}{N-1}\sum_{i=1}^{N-1} C_i\right)} \tag{2.9}$$

Donde C_i representa el consumo durante el año i y N el número de años considerados.

$$car14 = \frac{var\left(C_N,\right)}{var\left(C_m\right)} \tag{2.10}$$

Donde C_N representa el consumo durante el último año y C_m el consumo promedio tomado entre todos los consumos.

Coeficientes de Fourier

Las siguientes 5 características, corresponden al modulo de los 5 primeros coeficientes de Fourier de **todo** el consumo. Los coeficientes de Fourier de todo el consumo fueron considerados en algunos trabajos previos con buenos resultados, razón por la cual incluyeran como características,

$$car\{15, 16, 17, 18, 19\} = \left\|FFT(C)_{\{1,2,3,4,5\}}\right\| \tag{2.11}$$

Pendiente de la recta que ajusta al consumo

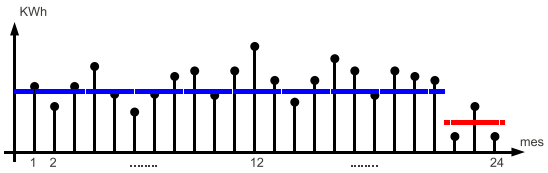
Como última característica, se incluye la pendiente de la recta que mejor ajusta la curva de consumos. Esta característica, resulta adecuada para obtener información de la tendencia del consumo y si el mismo presenta una caída sostenida. Además, fue considerada en trabajos anteriores [15] y parece adecuado incluirla en el conjunto final.

Resumen

Las características que se utilizaran se enumeran a continuación:

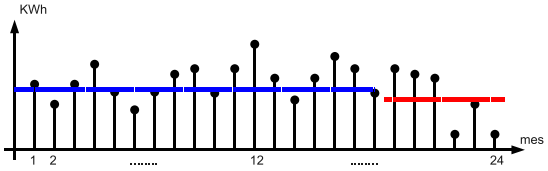
1. Cociente entre el valor medio en los últimos 3 meses y el pasado

$$car1 = \frac{mean(C([1:n-4]))}{mean(C[n-3:n])} \tag{2.12}$$

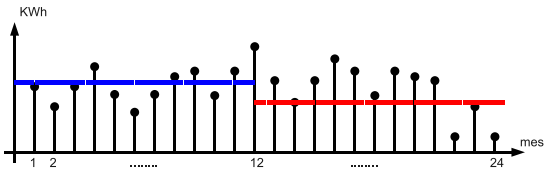


2. Cociente entre el valor medio en los últimos 6 meses y el pasado
3. Cociente entre el valor medio en los últimos 12 meses y el pasado
4. Norma de la diferencia entre el consumo esperado y el consumo real

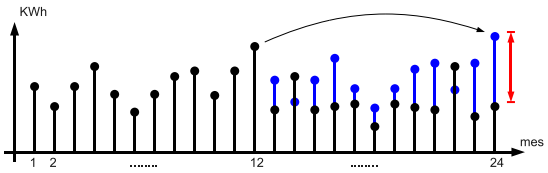
$$car2 = \frac{mean(C([1:n-7]))}{mean(C[n-6:n])} \tag{2.13}$$



$$car3 = \frac{mean(C([1:n-13]))}{mean(C[n-12:n])} \tag{2.14}$$



$$car4 = \sqrt{\sum_{i=n-11}^{i=n} \left(C(i) - \alpha C(i-12)\right)^2} \tag{2.15}$$



5. Diferencia en los espectros

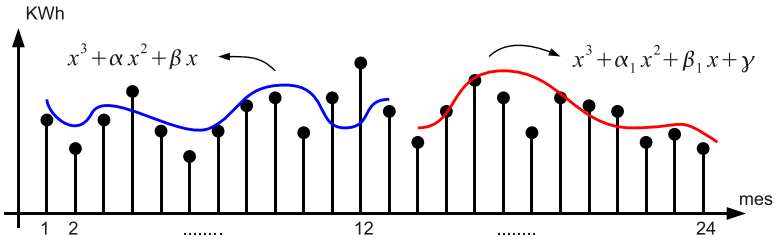
$$car5 = \|FFT(C_{actual}) - FFT(C_{medio})\| \tag{2.16}$$

6. Diferencia en los coeficientes wavelet

$$\vec{car}6 = \left\| DWT\left(C_N\right) - \frac{1}{N-1}\sum_{i=1}^{N-1} DWT\left(C_i\right) \right\| \tag{2.17}$$

7. Diferencia en el ajuste de un polinomio de grado n

$$car\{7,8,9,10,11\} = polyfit(C_{ao\ n}) - \frac{1}{n-1}\sum_{i=1}^{n-1} polyfit(C_{ao\ i}) \tag{2.18}$$

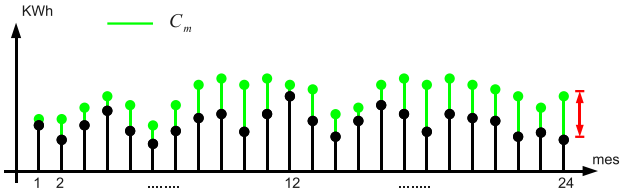


8. Distancia al consumo medio

9. Comparación de la varianza del consumo

$$car13 = \frac{var\left(C_N,\right)}{var\left(\frac{1}{N-1}\sum_{i=1}^{N-1} C_i\right)} \tag{2.20}$$

$$car12 = \left\| \vec{C} - \vec{C}_m \right\| \tag{2.19}$$



10. Varianza del consumo

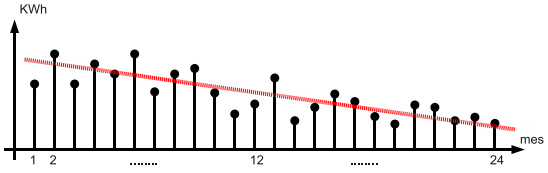
$$car14 = \frac{var\,(C_N,)}{var\,(C_m)} \tag{2.21}$$

11. Coeficientes de Fourier

$$car\{15, 16, 17, 18, 19\} = \left\| FFT(C)_{\{1,2,3,4,5\}} \right\| \tag{2.22}$$

12. Pendiente de la recta que ajusta al consumo

$$car20 = polyfit(C, 1) \tag{2.23}$$



Capítulo 3

SVM

SVM fue fue introducido por Vapnik en la década del 60 y desde la fecha a sido ampliamente utilizado y desarrollado. Hoy en día existen variadas aplicaciones que utilizan esta herramienta. El reconocimiento de patrones por medio de SVM ha demostrado ser uno método robusto y con excelentes despeños, utilizado tanto para reconocimiento facial [7] reconocimiento de textos [3] y en particular para la detección de fraude en energía eléctrica [6] y [15].

Actualmente, es uno de los métodos predominantes cuando se consultan las publicaciones más recientes relacionadas con la detección de fraude en energía eléctrica, por dicha razón era razonable (e indispensable) estudiar este tipo de técnicas para el caso particular que se pretende abordar. Las distintas variaciones y posibilidades que presenta SVM, dotan de esta una herramienta de mucha flexibilidad y versatilidad. Por supuesto que tiene un costo, una herramienta tan flexible y genérica presenta algunas dificultades pues resulta trabajoso encontrar la implementación que mejor se ajusta a cada problema concreto, además en general hay varias posibilidades de Kernel y cada una con variados parámetros a ajustar.

En este trabajo, abordaremos dos tipos de SVM, el primero sera C-SVM[a] y el segundo One Class SVM[b]. Ambos métodos consisten en encontrar el hiperplano óptimo, que separa a las clases con el mayor margen posible. La diferencia entre las dos técnicas consiste en que la primera considera que existen 2 clases en el problema y en base a las etiquetas de cada consumo busca el hiperplano que mejor separa las dos clases. Por otro lado, One Class SVM como su nombre lo indica, asume que solo tenemos presente una clase (los consumos normales) y lo que se pretenden identificar son outliers, es decir, consumos que no se ajustan a la clase pero que tomados en su conjunto no presentan necesariamente una estructura de clase. Este segundo método no utiliza ningún tipo de etiquetas (pues estrictamente no hay más que una sola clase).

En las secciones siguientes se estudiaran cada uno de los dos métodos, se hará un breve desarrollo teórico[c] de cada uno las distintas variaciones que presentan y los tipos de parámetros que entran en juego con sus respectivos significados. Por último se presentaran los resultados obtenidos en cada caso para el problema concreto que se esta abordando.

[a] este método fue utilizado en [6] con resultados prometedores

[b] utilizado en [15] para un conjunto muy particular de datos.

[c] para un desarrollo teórico más detallado de SVM se recomienda consultar [11] [14] y [2]

3.1. Clasificador C-SVM

3.1.1. C-SVM: Soft Margin

En 1995, Corinna Cortes y Vladimir Vapnik introdujeron la idea de un margen máximo que permita clasificar correctamente ciertas muestras. En caso de no existir un hiperplano tal que pueda separar las muestras en dos clases, el método de *Soft Margin* elegirá un hiperplano capaz de separar las muestras de la forma mas "limpia"posible, y al mismo tiempo maximizando la distancia a las muestras mejor separadas. Este método introduce una variable slack ζ_i que mide el grado de error de clasificación del dato x_i

El objetivo principal del algoritmo SVM binario (de dos clases) utilizado para clasificación es construir una función de decisión óptima, $f(x)$ que logre predecir de forma precisa a cual de las dos clases posibles pertenecen ciertos datos, minimizando el error de clasificación utilizando

$$f(x) = sgn(g(x)) \tag{3.1}$$

donde $g(x)$ es el límite de decisión entre las dos clases y es determinada a partir de los datos (muestras) de entrenamiento

$$X = x_1, x_2, \ ..., x_n, x_i \ \in \mathbb{R}^n, i = 1, 2, ..., n \tag{3.2}$$

donde cada dato de entrenamiento x_i tiene M características y pertenece a una de dos clases

$$Y = y_1, y_2, \ ..., y_n, y_i \ \in \{-1, +1\}, i = 1, 2, ..n. \tag{3.3}$$

El límite de decisión entre las dos clases es un hiperplano descrito por la ecuación

$$g(x) = \langle \omega, x \rangle \ + b \tag{3.4}$$

donde ω y b se obtienen de manera tal de clasificar correctamente los datos. Para lograr clasificar correctamente los datos debemos maximizar el margen de separación entre las dos clases. De acuerdo a [**?**], esto puede ser formulado como un problema de optimización de programación cuadrática

$$\Phi(\omega, \zeta_i) = \text{mín} \left\{ \frac{1}{2} \left\| \omega \right\|^2 + C \sum_{i=1}^{i=n} \zeta_i \right\} \tag{3.5}$$

sujeto a la condición de que todas las muestra de entrenamiento sean correctamente clasificadas (es decir que todas las muestras de entrenamiento se encuentren en el margen o fuera del mismo), esto es

$$y_i(\langle \omega, x \rangle \ + b) \geq 1 - \zeta_i, \ \ i = 1, 2, ..., n \tag{3.6}$$

donde ζ_i para $i = 1, 2, ..., n$ son variables no negativas. Al minimizar el primer término de 3.5 la complejidad del SVM se ve reducida, y al minimizar el segundo término se reduce el número de errores de entrenamiento. El parámetro C en 3.5 es un parámetro de regularización y se preselecciona para ser la compensación entre ambos términos de 3.5.

A partir de la formulación del problema dual resolvemos 3.5 sujeto a 3.6 y obtenemos que la expresión del limite de decision $g(x)$ está determinado únicamente por un subconjunto de los datos de entrenamiento

$$g(x) = \sum_{i=1}^{N_s} \alpha_i \ y_i \langle x, x_i \rangle + b \tag{3.7}$$

donde x es el vector de entrada, $\langle x, x_i \rangle$ es el producto interno, N_s es el número de vectores soportes y b es el sesgo.

En la mayoría de los casos no es posible determinar un limite de decisión lineal. En estos casos SVM mappea el vector de entrada x a un espacio de características de mayor dimension $\mathcal{H}$ dotado de producto interno [14] [**?**]. Esto se logra introduciendo una función de Kernel $k(.,.)$,

$$\langle x_i, x_j \rangle \rightarrow K(x_i, x_j) \tag{3.8}$$

La ventaja de esta transformación de las muestras de $\mathbb{R}^n$ a $\mathcal{H}$ mediante k consiste en introducir un paso no lineal en el proceso de construcción del clasificador, de este modo podemos obtener funciones de decisión no lineales y trabajar en espacios distintos del espacio original (en el cual la muestras pueden o no ser separables mediante un hiperplano). Teniendo en cuenta lo anterior es evidente la importancia de seleccionar el kernel que iremos a utilizar. Al introducir esta función de Kernel, el límite de decisión en 3.7 se transforma en

$$g(x) = \sum_{i=1}^{N_s} \alpha_i \, y_i K(x, x_i) \; + b \tag{3.9}$$

Cualquier función que satisfaga la condición de Mercer [13] puede ser utilizada como función de kernel. En general las funciones de kernel utilizadas en SVM se dividen en dos categorías: kernels basadas en distancia Euclidiana y kernels basadas en el producto interno Euclidiano [11]. Los Kernels son seleccionados basados en la estructura de los datos y el tipo de límites entre las clases.

En este trabajo y para este clasificador se decidió utilizar una función de kernel basada en la distancia Euclidiana, la función de kernel de base radial (RBF)

$$K(x_i, x_j) = e^{-\gamma \|x_i - x_j\|^2} \tag{3.10}$$

donde el parámetro γ controla el ancho de la función de kernel RBF. El kernel RBF induce un espacio de kernel de dimensión infinita en donde todos los vectores imágenes tienen la misma norma.

3.1.2. Elección de los Parámetros

Método de selección de los Parámetros Óptimos

La selección de los dos parámetros del modelo C-SVM es determinante para la precisión de clasificación. Por lo tanto, el modelo óptimo del clasificador se obtiene optimizando el parámetro del kernel RBF, γ, y el parámetro de penalización de error C. El primer paso para determinar los parámetros consiste en dividir la base de datos en dos, una de entrenamiento y otra de test. La base de entrenamiento es utilizada para probar distintos valores de C y γ y así obtener el modelo óptimo del clasificador.

En la practica, ambos parámetros (C y γ) se eligen utilizando validación cruzada. Con este propósito, se divide la base de entrenamiento en p partes iguales y se realizan p corridas de entrenamiento. En cada corrida se utilizan $p-1$ conjuntos para entrenar el clasificador (generar un modelo) y se utiliza el restante como un conjunto independiente de validación para optimizar los parámetros. En el caso mas simple, se eligen los parámetros que en promedio generan el mejor modelo según determinado criterio en las p corridas. Una vez determinados estos mejores parámetros, se entrena el clasificador con toda la base de entrenamiento. Esta metodología

presenta ciertos problemas. Primero, se utiliza la misma base para optimizar los parámetros y para entrenar el clasificador, esto puede llevar a overfitting. Segundo, la configuración óptima de parámetros para conjuntos de datos de tamaño m y $9/10m$ generalmente no coinciden. Por lo general, el conjunto de datos de menor tamaño requiere una mayor regularización. Esto puede significar un Kernel Gaussiano mas amplio, un menor grado polinomial, un menor C o un v mayor. Aun peor, es teóricamente posible que exista una fase de transición en la curva de aprendizaje entre conjuntos de diferente tamaño. Esto significa que la generalización de error como una función del tamaño del conjunto puede variar drásticamente entre $9/10m$ y m. A pesar de todo lo antes dicho, en general no se toman en cuenta esta consideraciones teóricas y se utilizan estos parámetros con excelentes resultados [11].

En este trabajo el procedimiento utilizado para determinar C_{opt} y γ_{opt} es el de validación cruzada y se lleva a cabo de la siguiente manera:

1. Se determinan los conjuntos $C = [C_1, C_2, ..., C_n]$ y $\gamma = [\gamma_1, \gamma_2, ..., \gamma_m]$.

2. Se eligen $C_i \in C$ y $\gamma_j \in \gamma$ y se divide la base de datos de entrenamiento en p partes iguales y se realizan p corridas de entrenamiento. Llamamos a cada base como B_i con $i = \{1, 2, ..., p\}$.

3. Se utiliza $B_{te} = B_1$ como base de test y $B_{tr} = B_2 \cup B_3 \cup ... \cup B_p$ como base de entrenamiento.

4. A partir de B_{tr}, C_i y γ_j se crea un modelo del clasificador. Como la relación entre las dos clases no es balanceada (hay una mayor cantidad de consumos normales que anómalos), al crear el modelo del clasificador C-SVM se utilizan pesos de clase definidos como la relación entre el número de muestras de entrenamiento sobre el número de muestras de cada clase.

5. Con este modelo se clasifican los vectores de la base B_{te} y se compara la etiqueta obtenida con la etiqueta de cada muestra. De esta comparación se obtiene el error según un determinado criterio de clasificación estimado para estos valores de C y γ que se llamamos $e_1(C_i, \gamma_j)$.

6. Se repite el procedimiento anterior considerando $B_{te} = B_2$ y la unión de las bases restantes como B_{tr} obteniendo $e_2(C_i, \gamma_j)$, luego $B_{te} = B_3$ y así sucesivamente hasta haber completado las p iteraciones.

7. Para el par de valores (C_i, γ_j) se cuenta con una estimación de errores del clasificador para cada validación cruzada. Se toma como error asociado a este par (C_i, γ_j), el valor promedio de los errores obtenidos en las validaciones cruzadas, $e(C_i, \gamma_j) = \frac{1}{p} \sum e_l(C_i, \gamma_j)$.

8. Este mecanismo se repite combinando todos los valores de los conjuntos C y γ

9. Finalmente, se elige como C_{opt} y γ_{opt} los valores con los cuales se obtiene el menor error promedio.

Entendiendo el procedimiento anterior se debe definir qué criterios de diseño se utilizarán para medir el error de clasificación, es decir traducir de alguna forma la importancia que se le da a la cantidad de consumos anómalos y normales mal clasificados. Estos criterios

tienen el objetivo de lograr que la clasificación sea sensible al costo de forma de minimizar el riesgo de mala clasificación.

Una posibilidad es utilizar costos en la toma de decisiones. De esta forma se puede mejorar la performance del clasificador al entrenarlo con una base de datos no balanceada. El modelo de costos se puede expresar en la forma de la matriz de costo, como se puede observar en la tabla 3.1.2, donde el costo de clasificar una muestra de la clase j como de la clase i corresponde a la entrada λ_{ij} de la matriz. Usualmente esta matriz se expresa en términos del costo promedio de los errores de clasificación. El objetivo en los problemas de clasificación sensibles al costo es minimizar el costo en los errores de clasificación, lo que puede obtenerse al elegir la clase con el mínimo riesgo condicional [4].

	Class i	Class j
Class i	0	λ_{ij}
Class j	λ_{ji}	0

Cuadro 3.1: Matriz de Costo

El criterio de evaluación mas común es el de la precisión definido como la cantidad de muestras mal clasificadas. Sin embargo este criterio no es un método adecuado para evaluar bases de datos no balanceadas (como sucede en nuestro caso) [4].

En la actualidad se han propuesto nuevos criterios de evaluación para bases de datos no balanceados. Algunos ejemplos son precisión (referido a la precision en alguna de las dos clases), recall, ROC, AUC (área debajo de la curva ROC), Valor F, MGM (máximo promedio geométrico de la precision en la clase de la mayoría y de la minoría), MS (suma máxima de precisión). Todas estos criterios pueden clasificarse en dos categorías: criterios basados directamente en la matriz de confusión y criterios basados en la precision de clases binarias o precision y recall directamente.

	Positivo	Negativo
Positivo	VP (Verdadero Positivo)	FN (Falso Negativo)
Negativo	FP (Falso Positivo)	VN (Verdadero Negativo)

Cuadro 3.2: Matriz de Confusion

La etiqueta de clase de la minoría es la positiva y la etiqueta de clase de la mayoría es la negativa. Como se puede ver en la tabla 3.1.2, VP y VN denotan el número de muestras positivas y negativas correctamente clasificadas, mientras que FN y FP denotan el numero de muestras positivas y negativas incorrectamente clasificadas.

A partir de la matriz de confusion podemos definir algunos de los criterios antes mencionados.

- $Precision_+ = \frac{VP}{VP+FP}$
- $Recall_+ = \frac{VP}{VP+FN}$
- $FPrate = \frac{FP}{FP+VN}$
- $VPrate = \frac{VP}{VP+FN}$

En este trabajo tomamos las ideas explicadas anteriormente y definimos dos criterios de diseño

1. Criterio 1: $Error = \alpha \frac{FP}{FP+VN} + (1-\alpha)\frac{FN}{VP+FN}$
   - $\alpha = \lambda_{ji}$ (costo de clasificar una muestra de la clase i como de la clase j)
   - $(1-\alpha) = \lambda_{ij}$ (costo de clasificar una muestra de la clase j como de la clase i)
   - $FP = \#$ Anómalos mal clasificados
   - $FN = \#$ Normales mal clasificados

   Al elegir un $\alpha > 0{,}5$ se resalta la mayor importancia de clasificar correctamente los consumos anómalos, comprometiendo la efectividad en la clasificación de los consumos normales. Se probaron distintos valores de α para determinar el valor óptimo. Tras ensayar el compromiso entre el número de VP contra FN se determinó que uno de los valores más convenientes para trabajar es $\alpha = 0{,}6$. Para valores superiores a este, se incrementa levemente el número de VP pagando un costo alto debido a un incremento elevado en el número de FN. En la sección 3.3 se presenta un análisis más detallado que confirma el valor seleccionado para α. En la página 25 se muestra una curva ROC para distintos valores de α.

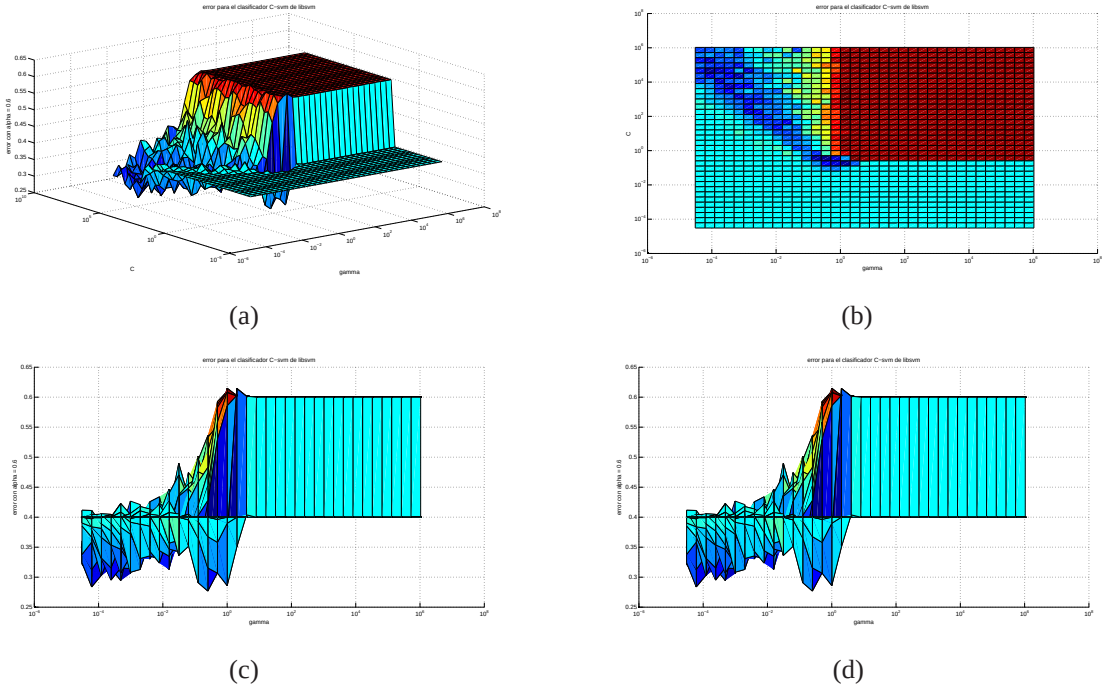
2. Criterio 2: $Error = FP$ Este criterio toma como medida del error la cantidad de consumos anómalos mal clasificados tendiendo en cuenta la densidad de consumos etiquetados como anómalos. Se define $d_0 = \frac{P}{N}$ y una densidad mínima $d_{min} = \beta \cdot d_0$ (variando β). Con estas dos densidades definidas se calcula la densidad de anómalos bien clasificados con respecto a todos los consumos clasificados como anómalos, $d = \frac{VP}{VP+FN}$ donde $VP = \#Anomalosbienclasificados$. Si esta densidad es mayor a la densidad mínima (d_{min}) entonces efectivamente puedo definir el error como la cantidad de consumos anómalos mal clasificados, de lo contrario no considero el error (a los efectos de programación y considerando que busco tener el menor error posible, defino el error como un numero muy grande).

Datos Obtenidos

Cuando se presento el algoritmo C-SVM se hizo hincapié en la importancia que tiene elegir correctamente los parámetros C y γ. A priori no conocemos cuales son los valores óptimos C_{opt} y γ_{opt} por lo que debemos realizar una búsqueda procurando determinar un rango acotado cercano a la solución. Como primer paso realizamos una búsqueda de los parámetros C y γ que minimizan el criterio de error en un rango muy amplio recorrido exponencialmente. En las figuras 3.1 y 3.2 podemos observar el error promedio obtenido para $C \in [2^{-15}, 2^{20}]$ y $\gamma \in [2^{-15}, 2^{20}]$ utilizando la base de datos de entrenamiento balanceada y no balanceada respectivamente. La decisión de explorar el error para los distintos valores de C y γ con la base de entrenamiento balanceada y no balanceada se debe a que en una primera instancia debemos verificar que utilizar pesos de clase para balancear la relación entre clases en el entrenamiento funciona correctamente. Esto se verifica al observar que en ambas figuras se aprecia un comportamiento similar en cuanto al error promedio de clasificación.

En las figuras 3.1 y 3.2 se pueden identificar dos zonas particulares donde el error promedio es constante al variar C y γ, en un caso el error es $0{,}6$ (zona roja) y en el otro

caso $0{,}4$ (zona celeste) . Estas zonas corresponden a los valores de C y γ que dejan a todas las muestras de un lado de la frontera (cuando todos los consumos son clasificados como normales se comete un error igual a α y cuando todos los consumos son clasificados como anómalos el error cometido será $(1-\alpha)$. Dejando de lado las zonas antes mencionadas, se observan variaciones del error cometido por el clasificador para distintas combinaciones de C y γ, en particular se puede reconocer una zona donde el error es mínimo ($C \in [0{,}001, 1]$ y $\gamma \in [0{,}001, 1]$). Esta zona es la que se estaba tratando de localizar, entorno a la misma haremos un recorrido lineal con pasos más finos para obtener los valores C_{opt} y γ_{opt}.



(a) (b)

(c) (d)

Figura 3.1: Error promedio utilizando $\alpha = 0{,}6$ y con la base de entrenamiento balanceada. La escala es logarítmica en C y γ

3.2. Evaluación del clasificador

Criterios para la evaluación

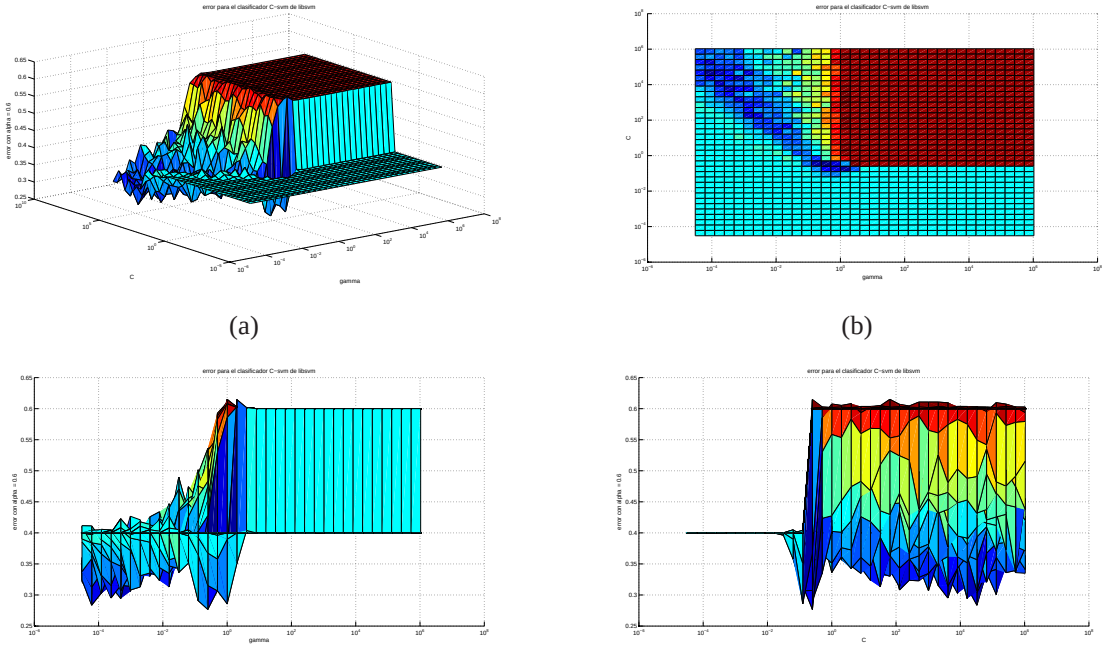
Además de los criterios de error introducidos en la página 18 estableceremos algunos criterios para la evaluación de los clasificadores. Estudiaremos como varían 3 cantidades (C_1, C_2 y C_3) para las distintas combinaciones de clasificadores, las mismas se definen a continuación:

$$C_1 = \frac{VP}{P} \tag{3.11}$$

$$C_2 = \frac{VP}{VP + FN} \tag{3.12}$$

$$C_3 = \left(\frac{VP}{P} + \frac{VN}{N}\right) \cdot \frac{50}{2} \tag{3.13}$$

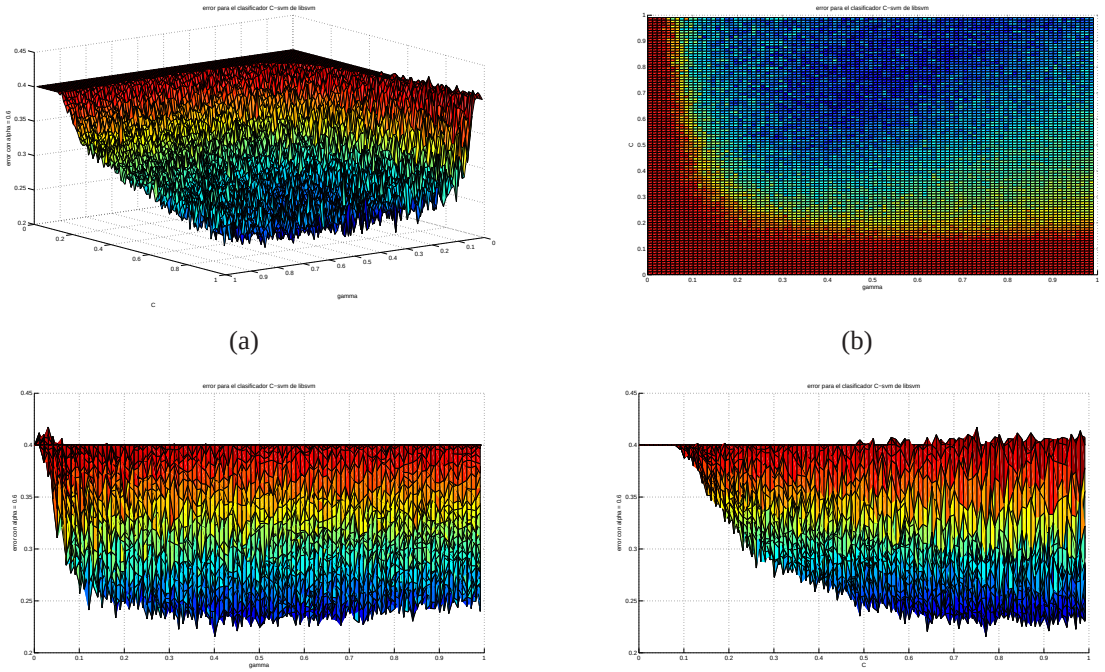
donde VP (verdaderos positivos) representa el número los anómalos bien clasificados, P el total de anómalos, FN (falsos negativos) los consumos normales mal clasificados, N es el número



(a) (b)

(c) (d)

Figura 3.2: Error promedio utilizando $\alpha = 0{,}6$ y con la base de entrenamiento balanceada. La escala es logarítmica en C y γ



(a) (b)

(c) (d)

Figura 3.3: Error promedio utilizando $\alpha = 0{,}6$ y con la base de entrenamiento balanceada. La escala es lineal en C y γ

de normales y VN (verdaderos negativos) el número de normales bien clasificados.

C_1 pretende brindar una medida de la cantidad de consumos anómalos de la base de datos que se clasifican correctamente, a modo de ejemplo si $C_1 = 100\,\%$ quiere decir que nuestro sistema esta detectando el %100 de los consumos anómalos de la base, esto no quiere decir que no se hallan cometido errores ya que no tenemos información respecto a los FP que podemos tener, para obtener este información utilizamos C_2. Por medio de C_2 tenemos una idea de la densidad de consumos anómalos (verdaderos) dentro del conjunto de consumos **clasificados** como anómalos. A nodo de ejemplo, si tenemos un valor de $C_2 = 30\,\%$, implica del conjunto de consumos que el sistema clasifica como anómalos, el 30 % de los mismos es efectivamente anómalo. Si bien a priori valores bajos de C_2 podrían asociarse a malos desempeños del clasificador, esta medida debe analizarse de manera delicada pues, la densidad original de consumos anómalos en el universo de clientes es realmente baja (del orden del 4 %). Por dicha razón obtener una base más reducida de consumos a inspeccionar donde la densidad de consumos anómalos es del 40 % implica un incremento de 10 veces la densidad de consumos anómalos inicial. Finalmente C_3 es el error cometido considerando simultáneamente los consumos anómalos mal clasificados y los normales.

Datos Obtenidos

Una vez determinado el rango de valores para C_{opt} y γ_{opt}, se está en condiciones de evaluar el clasificador en distintas condiciones. En particular se evaluara cómo varia el desempeño del clasificador para las dos bases de datos disponibles a la fecha y las distintas opciones de características planteadas.

Descripción:
Datos:"cons-etiq-pru2" (idem [15])
Balanceo: si (50-50)
Características: DeCA (V2)
Clasificador: C-SVM
Parámetros del Clas: $\alpha = 0{,}6$

	Anómalos	Normales
Bien Clas	20	14
Mal Clas	0	6

$C_1 = 100\,\%$
$C_2 = 76{,}9\,\%$
$C_3 = 85\,\%$

Descripción:
Datos:"listado-con-fecha-V2"
Balanceo: si (50-50)
Características: DeCA (V2)
Clasificador: C-SVM
Parámetros del Clas: $\alpha = 0{,}6$

	Anómalos	Normales
Bien Clas	60	30
Mal Clas	6	36

$C_1 = 90{,}9\,\%$
$C_2 = 62{,}5\,\%$
$C_3 = 68{,}2\,\%$

Descripción:
Datos:”cons-etiq-pru2” (idem [15])
Balanceo: Si (50-50)
Características: utilizadas en [15]
Clasificador: C-SVM
Parámetros del Clas: $\alpha = 0{,}6$

	Anómalos	Normales
Bien Clas	20	19
Mal Clas	0	1

$C_1 = 100\,\%$
$C_2 = 95{,}2\,\%$
$C_3 = 97{,}5\,\%$

Descripción:
Datos: ”listado-con-fecha-V2”
Balanceo: Si (50-50)
Características: utilizadas en [15]
Clasificador: C-SVM
Parámetros del Clas: $\alpha = 0{,}6$

	Anómalos	Normales
Bien Clas	58	25
Mal Clas	8	41

$C_1 = 87{,}9\,\%$
$C_2 = 58{,}6\,\%$
$C_3 = 62{,}9\,\%$

3.3. One-Class SVM

Los algoritmos tradicionales de SVM son entrenados usando representantes de cada clase que se pretende separar, por ejemplo asumiendo que estamos en un problema donde se pretende distinguir entre dos clases, se busca un hiperplano que *separe* los representantes de cada clase. Luego clasificaremos las nuevas muestras en función de que lado ocupan del hiperplano.

En este método sin embargo, procuraremos extraer información asumiendo que solo tenemos una clase, es decir solo tenemos muestras de una misma clase. Este tipo de técnicas a resultado importante en varias aplicaciones, a modo de ejemplo, supongamos que queremos identificar un perfil de páginas web de interés para un usuario, en ese caso solo poseemos información de las páginas que visita pero carecemos de información para representar las páginas que no son de su interés.

Scholkopf (1999) fue uno de los pioneros en resolver esta problemática, propuso adaptar la metodología SVM al caso de una sola clase. Luego de transformar las muestras a un espacio de dimensión mayor mediante el uso de un Kernel, considera el origen como único representante de la segunda clase, luego se aplican los algoritmos tradicionales de SVM multiclase y se busca el hiper-plano que separa con mayor margen las muestras del origen.

En otras palabras, se desarrolló un algoritmo que dado un conjunto de muestras, devuelve una función f que toma valor $+1$ en una parte *pequeña* del espacio capturando la *mayoría* de los vectores de muestra y -1 fuera de esa región. La región donde $f(x) = -1$ es considerada la zona donde se encuentras los outliers o *anómalos*.

A continuación presentaremos un desarrollo formal de la metodología propuesta, luego se desarrollara como se implemento la misma en este caso particular y por último presentaremos los resultados obtenidos.

3.3.1. SVM One Class : separando los datos del origen

Supongamos que tenemos $x_1,\ x_2,\ ...\ x_l\ \in \mathbb{R}^n$ muestras de una clase C, por otro lado consideremos una función $\Phi : \mathbb{R}^n \to \mathcal{H}$ que mapea los datos x_i a un nuevo espacio $\mathcal{H}$ con la propiedad de que $\langle \Phi(x_i),\ \Phi(x_j) \rangle = k(x_i,\ x_j) \quad \forall\ x_i,\ x_j \in \mathbb{R}^n$. La propiedad anterior implica que no es necesario transformar cada muestra para realizar el producto interno en el espacio $\mathcal{H}$, es decir podemos expresar dicho producto interno en termino de las muestras en el espacio original (en este caso $\mathbb{R}^n$).

Con el objetivo de separar los datos (en $\mathcal{H}$) del origen, debemos resolver el siguiente problema:

$$\min_{\omega \in \mathcal{H},\, \zeta_i \in \mathbb{R},\, \rho \in \mathbb{R}} \frac{1}{2} \|\omega\|^2 + \frac{1}{\nu\, l} \sum_{i=1}^{i=l} \zeta_i - \rho \tag{3.14}$$

$$\text{sujeto a las restricciones } \langle \omega, \Phi(x_i) \rangle \geq \rho - \zeta_i, \quad \zeta_i \geq 0 \tag{3.15}$$

Donde $\nu \in (0, 1]$ es un parámetro que regula el compromiso entre cometer errores y obtener mayor separación de la frontera con el origen. Recordemos que el único representante de la segunda clase será el origen de modo que desearíamos que el hiperplano se encuentre lo más lejos posible del origen. Sin embargo por el otro lado cuanto más alejamos el hiperplano del origen, más muestras quedaran del lado incorrecto del hiperplano de modo que estaríamos cometiendo errores al clasificar algunas de las muestras x_i.

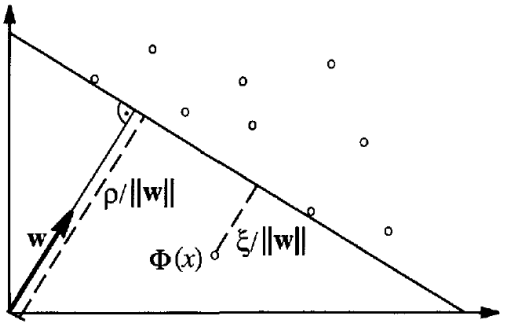


Figura 3.4: Ejemplo en el caso en que trabajamos con muestras en $\mathbb{R}^2$ para ilustrar el compromiso entre obtener un margen amplio y obtener pocos errores. Imagen tomada de [11, pag:232]

Una vez que resolvemos 3.14 sujeto a 3.15 obtenemos $f(x) = sign(\langle \Phi(x), \omega \rangle - \rho)$ que nos permitirá clasificar nuevas muestras (como anómalos si $f(x) = -1$ o consumos normales si $f(x) = 1$).

En este trabajo se utilizaran librerías que tienen implementados los algoritmos para calcular la solución óptima de 3.14 y 3.15, las mismas están disponibles para su descarga en `http://www.csie.ntu.edu.tw/~cjlin/libsvm`

3.3.2. Elección de los parámetros

Elección del kernel

En 3.3.1 se presentó en qué consiste el método de One Class SVM como herramienta para detectar muestras anómalas dada una base de datos con estructura de clase única, sin embargo aun queda camino por recorrer antes de tener un clasificador entrenado y listo para su evaluación y posterior uso. En primer lugar debemos determinar el kernel que deseamos utilizar. Tal como se describió en la sección anterior uno de los pasos fundamentales consiste en mapear las muestras originales a un espacio $\mathcal{H}$ dotado de producto interno [12] [6]. La ventaja de esta transformación de las muestras de $\mathbb{R}^n$ a $\mathcal{H}$ mediante Φ consiste en introducir un paso no lineal en el proceso de construcción del clasificador, de este modo podemos obtener funciones de decisión no lineales y trabajar en espacios distintos del espacio original (en el cual la muestras pueden o no ser separables mediante un hiperplano). Teniendo en cuenta lo anterior es evidente la importancia de seleccionar el kernel que iremos a utilizar dado por $k : \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R} \ : \ k(x, x_i) = \langle \Phi(x), \Phi(x_i) \rangle$.

Existen infinidad de kernels propuestos en la literatura (se puede consultar [12] si se desea un análisis profundo en cuanto a la elección de kernels), los más clásicos son:

- Kernels Polinómicos:
$$k(x, x_i) = \langle x, x_i \rangle^d \tag{3.16}$$
- Kernel Gaussiano:
$$k(x, x_i) = e^{-\frac{\|x - x_i\|^2}{\sigma}}; \quad \text{con } \sigma > 0 \tag{3.17}$$
- Kernel tanh:
$$k(x, x_i) = tanh(\kappa \langle x, x_i \rangle + \varphi); \quad \text{con } \ \kappa > 0 \text{ y } \theta \in \mathbb{R} \tag{3.18}$$

En este trabajo se utilizara el kernel Gaussiano pues es uno de los kernels más utilizados en la práctica y además ya fue utilizado en otros trabajos [15][6] de detección de fraude en energía eléctrica con buenos resultados.

Parámetros Óptimos

El paso siguiente luego de determinar el tipo de kernel que se va a utilizar, consiste en determinar los parámetros óptimos que definirán el modelo del clasificador. Estos parámetros dependen naturalmente del kernel que se utilice y del tipo de svm que se está implementando, en este caso particular se utilizara un kernel Gaussiano y SVM-One-Class. Los parámetros en juego en este son ν y σ presentados en (3.14) y 3.17 respectivamente.

Determinar el valor de estos parámetros de una manera adecuada es crucial a la hora de obtener buenos modelos para los clasificadores, en la figura 3.5 se ilustra como varia la frontera de decisión al variar σ. En la figura 3.4 se puede observar como afecta la variación de ν a la hora de determinar el margen y el número de vectores soporte. De que manera es alterado el desempeño de nuestro clasificador al variar simultáneamente ν y σ no es sencillo de predecir y será abordado en lo que sigue.

Para determinar ν y σ realizamos el siguiente procedimiento (sugerido en [11]):

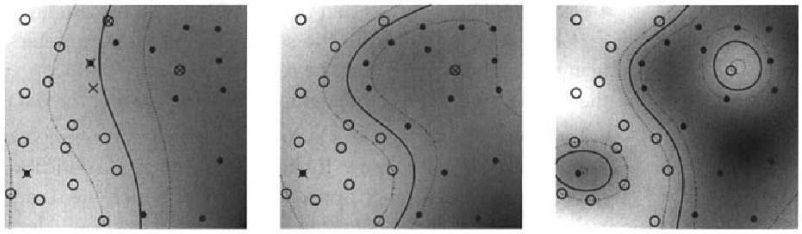


Figura 3.5: Figura a los efectos ilustrativos de como varía la frontera de decisión para un kernel Gaussiano al variar σ. De izquierda a derecha se decrementa σ. Se puede Observar que para valores elevados de σ la frontera es más lineal y los datos no pueden ser separados sin error, en el otro extremo para valores pequeños del ancho del kernel obtenemos una frontera más ajustada a los datos donde se obtiene un comportamiento no lineal y menos errores en la clasificación. Si reducimos mucho el ancho del kernel debemos tener precauciones de no obtener una curva muy sobre adaptada a los datos de entrenamiento. Imagen tomada de [11]

1. Dividimos la base de datos en p partes iguales, llamemos cada base como B_i con $i = \{1, 2, ..., p\}$
2. Tomamos $B_{te} = B_1$ como base de test y $B_{tr} = B_2 \cup B_3 \cup ... \cup B_p$ como base de entrenamiento.
3. Para cada $\nu \in U = \{\nu_1, \nu_2.....\nu_m\}$ y $\sigma \in S = \{\sigma_1, \sigma_2.....\sigma_l\}$ entrenamos el clasificador con la base B_{tr} y obtenemos una frontera de decisión f.
4. Con f clasificamos los vectores de la base B_{te} y comparamos la etiqueta obtenida con la etiqueta de cada muestra. De la comparación anterior obtenemos el error[d] estimado para ese valor de ν y σ, llamado $e_1(\nu, \sigma)$.
5. Repetimos el procedimiento anterior considerando $B_{te} = B_2$ y la unión de las bases restantes como B_{tr} obteniendo $e_2(\nu, \sigma)$, luego $B_{te} = B_3$ y así sucesivamente hasta haber completado las p iteraciones.
6. Contamos entonces con una estimación para el error del clasificador para cada $\nu \in U$ y $\sigma \in S$ en cada uno de los p casos $(e_1,\ e_2,\ ...\ ,\ e_p\ )$. Tomamos como error asociado a cada ν y σ el valor promedio de los errores obtenidos en las validaciones cruzadas, $e(\nu, \sigma) = \frac{1}{p} \sum e_i(\nu, \sigma)$.
7. Finalmente escogemos el valor de ν y σ con el que se obtiene menor error: $(\nu_{final},\ \sigma_{final}) = \text{argmín}_{\nu \in U,\ \sigma \in S} \{e(\nu, \sigma)\}$

Datos Obtenidos

Utilizaremos distintos criterios de error como se explico en la sección 3.1 (pág 18). Se comenzará introduciendo el primer criterio con $\alpha = 0{,}6$, este valor se elige tras analizar el compromiso que se obtiene entre el porcentaje de anómalos bien clasificados y el porcentaje

[d] existen distintas alternativas para evaluar el *error* cometido en la clasificación. En este trabajo se consideraron dos opciones para cuantificar el error cometido que fueron explicadas en la página 18 (sección 3.1)

de normales mal clasificados, es decir la curva *ROC* que analiza la evolución entre los VP (verdaderos positivos) y los FP (falsos positive). Dicha curva se muestra en la figura 3.6 donde se puede observar que para valores mayores que $\alpha = 0{,}6$ aumenta considerablemente el numero de normales mal clasificados si se desea aumentar el índice de anómalos mal clasificados apenas un $5\,\%$ o un $10\,\%$. También se puede observar otra alternativa de trabajo posible podría, si trabajamos con valores de $\alpha \approx 0{,}4$ tendremos un número reducido de falsos anómalos, por supuesto que esta elección tiene un costo, se perderá la capacidad de detectar algunos de lo consumos anómalos ya que para esos valores de alfa también se ve reducido el porcentaje de anómalos detectados.

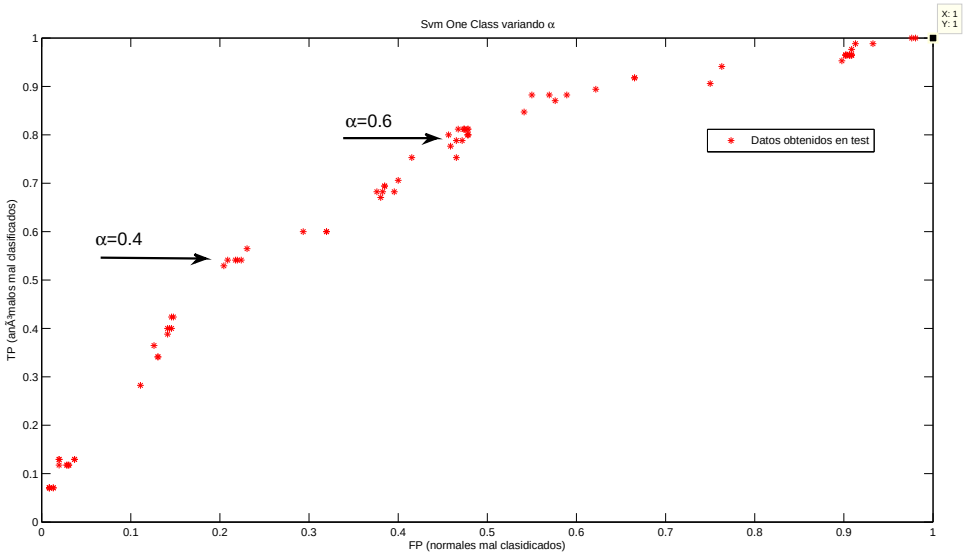
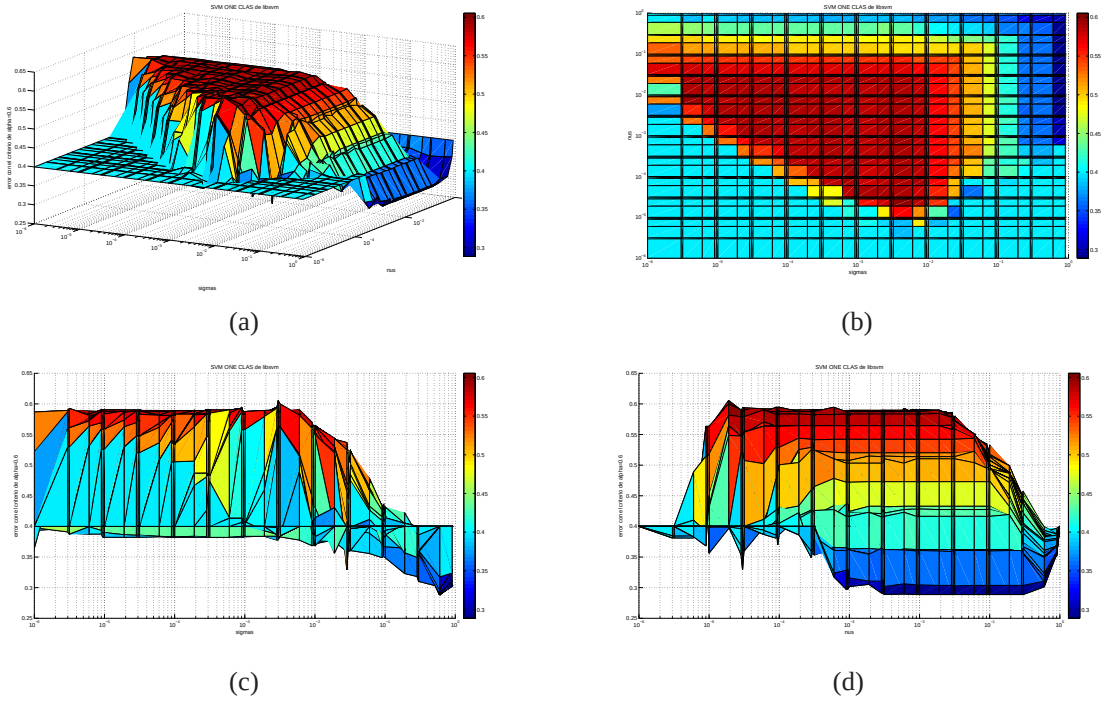


Figura 3.6: Curva ROC al variar $\alpha \in [0 : 0{,}01 : 1]$

Como a priori no se conoce el rango en el que se encuentran los ν y σ que minimizan el error, se realiza una exploración del error cometido al variar ν y σ en un rango bastante amplio recorrido exponencialmente. En la figura 3.7 se puede observar el error obtenido para $\nu \in [10^{-6}, 1]$ y $\sigma^{-1} \in [10^{-6}, 1]$, si se observa la figura se pueden identificar dos zonas particulares (al igual que sucedía para C-SVM[e]) donde el error es constante al variar ν y σ, en un caso el error es $0{,}6$ (zona roja) y en el otro caso $0{,}4$ (zona celeste) . Estas zonas corresponden a los valores de ν y σ que dejan a todas las muestras de un lado de la frontera (cuando todos los consumos son clasificados como normales se comete un error igual a α y cuando todos los consumos son clasificados como anómalos el error cometido será $(1 - \alpha)$). Sacando de lugar las zonas antes mencionadas, se puede observar como varía el error cometido al variar simultáneamente los parámetros del clasificador. Como se menciono anteriormente una correcta elección de los parámetros es crucial a la hora de obtener un buen modelo.

Si bien la información disponible en la figura 3.7 es conveniente para visualizar el comportamiento del clasificador en función de ν y σ, no será utilizada para determinar el valor de ν y σ óptimos. Se utilizara la información obtenida de dicha curva para establecer una zona $U \times S \ (\in \mathbb{R}^2)$ en la que se encuentran los mejores desempeños del clasificador y

[e] ver página **??**

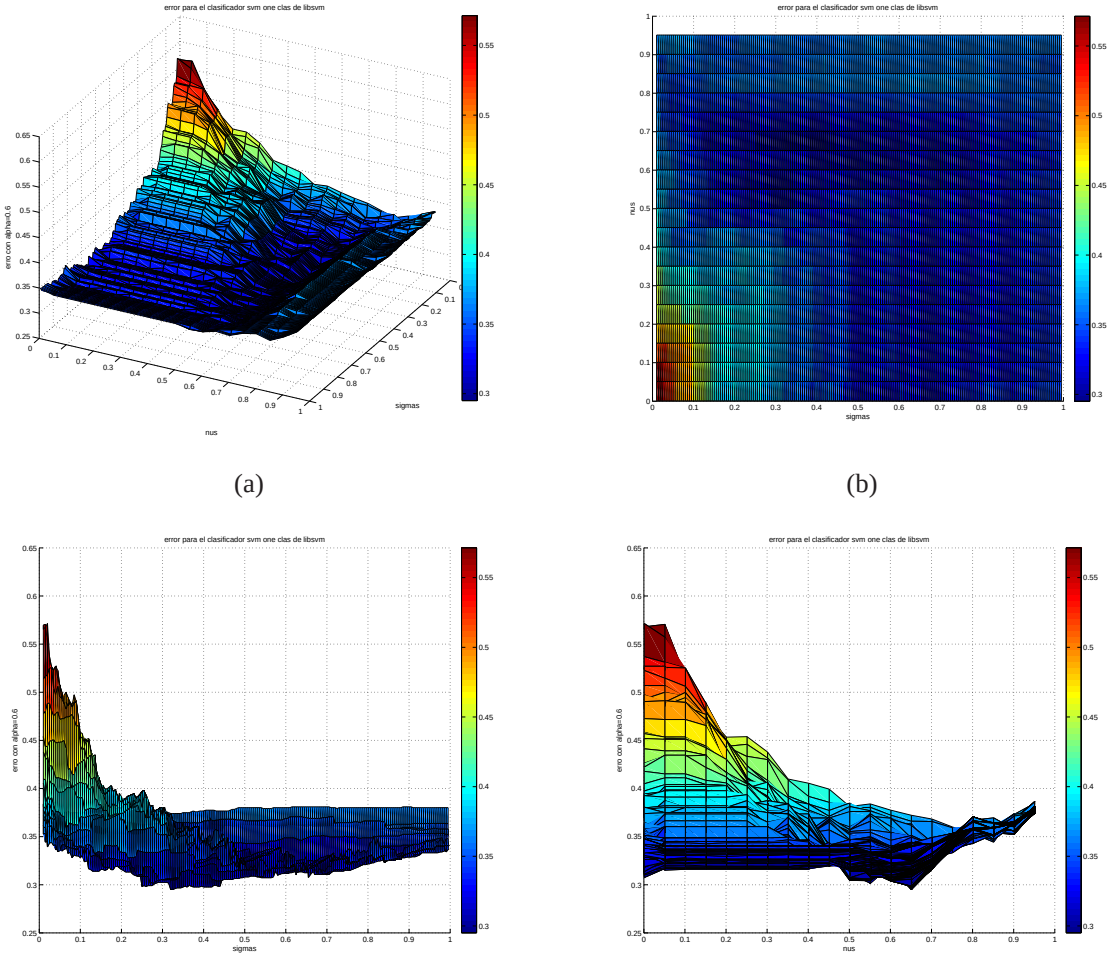


(a) (b)

(c) (d)

Figura 3.7: Error utilizando $\alpha = 0{,}6$. La escala es logarítmica en ν y σ^{-1}

se recorrerá dicha zona de manera lineal con pasos más pequeños para una idea más precisa del comportamiento del clasificador en la zona que presenta mejores resultados. Los resultados obtenidos al recorrer de manera más fina la zona del mínimo error se muestran en la figura 3.8. En este segundo gráfico la escala es lineal y se estudio el error del clasificador para valores de $\nu \in [0{,}001 : 0{,}05 : 0{,}999]$ y $\sigma^{-1} \in [0{,}01 : 0{,}005 : 0{,}999]$



(a) (b)

(c) (d)

Figura 3.8: Error utilizando $\alpha = 0{,}6$. La escala es lineal en ν y σ^{-1}
. Se explora de manera local la zona donde se obtienen mejores resultados en la figura 3.7

3.3.3. Evaluación del clasificador

Datos obtenidos

Luego de analizados los parámetros óptimos para determinar la frontera de decisión en cada caso, se dispone de las condiciones necesarias para entrenar y poner a prueba esta técnica de SVM con las bases de datos disponibles. Los resultados que se obtienen en distintas condiciones se muestran a continuación.

Descripción:
Datos:"cons-etiq-pru2" (idem [15])
Balanceo: no
Características: utilizadas en [15]
Clasificador: SVM One Class
Parámetros del Clas: $\alpha = 0{,}5$

	Anómalos	Normales
Bien Clas	24	220
Mal Clas	2	5

$C_1 = 92{,}3\,\%$
$C_2 = 82{,}8\,\%$
$C_3 = 95{,}0\,\%$

Descripción:
Datos:"listado-Con-Fecha-V2"
Balanceo: no
Características: DeCA (V2)
Clasificador: SVM One Class
Parámetros del Clas: $\alpha = 0{,}6$

	Anómalos	Normales
Bien Clas	73	228
Mal Clas	9	223

$C_1 = 89{,}0\,\%$
$C_2 = 24{,}7\,\%$
$C_3 = 69{,}8\,\%$

Descripción:
Datos:"listado-Con-Fecha-V2"
Balanceo: no
Características: utilizadas en [15]
Clasificador: SVM One Class
Parámetros del Clas: $\alpha = 0{,}6$

	Anómalos	Normales
Bien Clas	69	287
Mal Clas	13	164

$C_1 = 84{,}1\,\%$
$C_2 = 29{,}6\,\%$
$C_3 = 73{,}9\,\%$

Descripción:
Datos:"cons-etiq-pru2" (idem [15])
Balanceo: no
Características: DeCA (V2)
Clasificador: SVM One Class
Parámetros del Clas: $\alpha = 0{,}6$

	Anómalos	Normales
Bien Clas	25	135
Mal Clas	0	90

$C_1 = 100\,\%$
$C_2 = 21{,}7\,\%$
$C_3 = 80\,\%$

Capítulo 4

OPF

Uno de los papers reseñados en la bibliografía acerca de la técnica de reconocimiento de patrones aplicada a la detección de fraudes en el consumo de energía eléctrica fue ”A New Approach for Nontechnical Losses Detection Based on Optimum-Path Forest” de Caio César Oba Ramos et al. En el mismo se detalla un clasificador llamado Optimum-Path Forest (OPF en adelante) con el cual se obtienen muy buenos resultados en un problema de similares características al nuestro.
La técnica de OPF surge como una alternativa, al igual que SVM, para los problemas donde las clases no son linealmente separables. Como ya se vió, SVM asume una separabilidad en una dimensión mayor a la que tienen los datos, pero las operaciones que lleva a cabo el mismo para lograr esto, son muy costosas en cuanto a tiempo, sobre todo cuando el conjunto de entrenamiento es grande. OPF intenta solucionar el problema de clases solapadas mediante un método computacionalmente mas eficiente

4.1. Descripción del método

A continuación se realizará una breve descripción cualitativa del método, un análisis más fondo del mismo se puede encontrar en [8].

Lo primero a realizar, al igual que en los métodos anteriores, es dividir a los datos en el conjunto de entrenamiento y el conjunto de test, a quien llamaremos Z_1 y Z_3 respectivamente. Con los vectores de Z_1 se crea el clasificador y con Z_3 se mide su performance, manteniéndose la información de a que clase pertenecen los elementos de test, ocultas hasta el final.

La técnica de OPF se encuentra dentro de los clasificadores llamados supervisados, es decir, para entrenarse, usa las etiquetas de los consumos.

El paper presenta como ejemplo el máximo de la función de costo entre nodos que componen cada camino. El algoritmo de entrenamiento lo que hace es minimizar la función de costo de camino para cualesquiera elementos del conjunto, dejando solo los arcos que componen algún camino óptimo entre 2 elementos. A esto se le da el nombre de minimum-spanning tree (MST). Luego de que este grafo está completado se procede a elegir los llamados ”prototipos”, es decir, los elementos mas representativos de cada clase, de tal manera que los elementos a clasificar sean asociados a la clase del prototipo cuya función de costo de conexión sea menor. El paper en cuestión, sostiene que los prototipos óptimos son los elementos adyacentes que

En la fase de entrenamiento lo que se intenta es crear un grafo donde los nodos son los vectores pertenecientes a Z_1. En un principio se conectan todos los nodos entre sí mediante arcos lográndose una red en donde cada elemento de Z_1 está conectado a todos los elementos del conjunto (ver figura 4.1-a). A continuación se elige una función de costo para estos arcos como puede ser la distancia euclidiana en el espacio de características entre ambos nodos. Se trata de hacer predominar los arcos que minimizan este costo; los arcos que nos van a servir son los de elementos adyacentes. Luego se crea una nueva función de costo a la que llamaremos de conexión, asignándole un costo a cada camino posible entre 2 elementos.

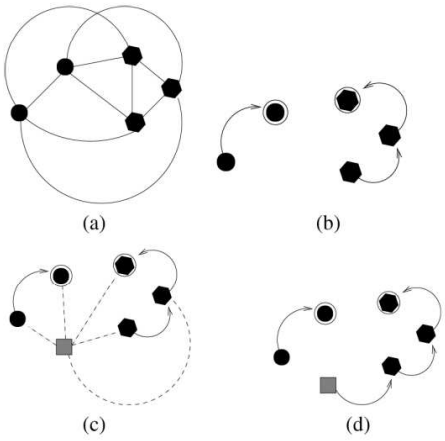


Figura 4.1:

tienen distintas etiquetas, es decir, los elementos que se encuentran en la frontera de las clases (figura 4.1-b) . Una vez que se eliminan los arcos que unen estos elementos se completa la etapa de entrenamiento A la hora de clasificar un nuevo elemento (figura 4.1-c), se lo conecta con todos elementos de Z_1 y se calcula el costo de conexión de este elemento con cada prototipo del grafo. Se encuentra el mínimo de estos costos, que corresponde al camino óptimo, y se le etiqueta con la clase de este prototipo con el que forma el camino óptimo (figura 4.1-d). De esta manera se clasifican todos los elementos de Z_3. La performance del clasificador se desprenderá de comparar las etiquetas previas con la salida del clasificador.

El paper también propone una mejora a lo dicho anteriormente que también fue implementada. Ésta trata de que el clasificador “aprenda” cuales son los elementos más significativos de cada clase y entrenar con ellos. Lo que se hace es al principio, dividir la base de datos en 3 conjuntos Z_1, Z_2 y Z_3, de entrenamiento, aprendizaje y test respectivamente. El clasificador se entrena con los elementos de Z_1 de la misma manera que la descrita previamente, logrando así el MST y los prototipos. La diferencia es que antes de pasar al test, se clasifican los elementos de Z_2, y los elementos mal clasificados de este conjunto se intercambian aleatoriamente por elementos de Z_1. Esto con la justificación de que los elementos mal clasificados son los elementos con más información de las clases. Se realiza esta iteración una determinada cantidad de veces y se elige el conjunto Z_1 que origina menor error en los elementos de Z_2.

Es importante destacar en este método que el entrenamiento y la clasificación se hace de forma automática, libre de cualquier parámetro. Lo único que se debe elegir es en que relación se reparten los elementos dentro de Z_1, Z_2 y Z_3. La ventaja de esto, es que a diferencia de SVM no hay que hacer validaciones cruzadas dentro del conjunto de entrenamiento ni probar variando los parámetros para descubrir cuales son los óptimos. Por otro lado tiene una desventaja, es más difícil enfocar el clasificador a un problema de clases desbalanceadas y con distintos costos de error, por ejemplo, minimizar los anómalos mal clasificados.

4.2. Aplicación del método

Para aplicar el clasificador OPF a nuestro problema se usó la librería libOPF mencionada en [8], de la página del instituto de computación de la Universidad estatal de Campinas. Es una librería en lenguaje C lista para compilar en un equipo con LINUX. Los archivos compilados se ejecutaron desde Matlab.

Al igual que para los otros clasificadores se utiliza el preprocesamiento y las características implementados para los consumos, logrando los vectores de características y las etiquetas que le ingresaremos al programa.

4.3. Resultados OPF sin aprendizaje

La primera prueba que se realizó con este clasificador fue algo bastante amplio, intentando descifrar su comportamiento ante las distintas bases de datos, las distintas características y los distintos parámetros con los que puede dividir la base para entrenar y testear. Con cada base y cada características se hicieron 9 pruebas variando el porcentaje de la base que se usa para entrenar (f_train), si se balancea, y con que porcentaje se balancea (p_bal). Para cada una de éstas, se entrenó y testeó 10 veces dividiendo la base previamente, para tener un promedio de cada uno de los indicadores explicados en 3.2 Vale la pena explicar que el balanceo puede ser de dos tipos, balanceando en train y en test, o solo balanceando en train. A continuación se muestran los parámetros con que se realizaron cada prueba utilizando el clasificador OPF sin la opción de aprendizaje.

Prueba	1	2	3	4	5	6	7	8	9
f_train	0.5	0.6	0.5	0.5	0.5	0.6	0.6	0.6	0.6
Balanceo	No	No	Todo	Todo	Todo	Todo	Todo	Train	Train
p_bal	0.5	0.5	0.5	0.4	0.6	0.5	0.6	0.5	0.6

4.3.1. Conjunto 1

En primer lugar se quiso comprobar la performance con lo utilizado en el proyecto anterior, es decir, los datos y las características de [15]. En la siguiente tabla se muestran el valor de los indicadores.

	1	2	3	4	5	6	7	8	9
C_1	34	37	57	55	55	57	61	54	58
C_2	79	80	91	91	91	89	94	32	32
C_3	67	68	76	75	74	75	78	74	76

4.3.2. Conjunto 2

La segunda prueba fue con la base de datos más amplia, “consumos con fecha” pero con las mismas características que la prueba anterior. Los resultados son estos.

	1	2	3	4	5	6	7	8	9
C_1	37	40	64	61	69	65	67	65	68
C_2	40	42	72	67	73	72	74	16	15
C_3	64	65	70	69	69	70	69	69	69

4.3.3. Conjunto 3

Por último realizamos la prueba con la base de datos ”consumos con fecha” y con las características explicadas en la sección de características del presente trabajo.

	1	2	3	4	5	6	7	8	9
C_1	34	35	62	60	67	63	65	64	67
C_2	38	36	69	65	72	70	71	15	15
C_3	62	62	68	66	67	68	66	67	68

Analizando los resultados anteriores podemos concluir lo siguiente:

- Los mejores resultados se dan para el caso de la base de datos utilizada en el proyecto anterior, lo cual es coherente ya que esta base es un conjunto muy acotado y particular de consumos, los cuales reúnen mejor las características que se supone definen mejor a cada clase
- De las pruebas con la base de datos amplia el mejor resultado se obtienen con las características del proyecto anterior. Puede ser por 2 motivos, el primero es que aquellas características logren describir mejor las diferencias en la clases, sin embargo esto no es coherente con los resultados en los anteriores clasificadores. El otro motivo es que tener muchas características, sobre todo algunas que pueden ser poco discriminatorias de las clases, en vez de ayudar, pueden confundir al clasificador, sobre todo a uno donde las distancias a vecinos es lo determinante. La solución de esto es implementar un método de selección o extracción para intentar tener la caracterización necesaria, mas concentrada en menos dimensiones.
- Resultados de las pruebas 1 y 2 son notoriamente inferiores a los de el resto de las pruebas, sobretodo a lo que respecta al criterio C_1 y C_2, lo cual demuestra que al balancear las clases se obtienen mejores resultados. Esto también era previsible ya que por como es el clasificador, si a la hora de entrenar hay muchos más elementos de una clase que de otra, a la hora de clasificar un nuevo elemento es más probable que el camino óptimo sea hacia un prototipo de la clase mayoritaria. Sobre todo esto es problemático cuando los elementos que más nos interesan detectar son los que se encuentran en minoría

- En las pruebas 8 y 9 cae significativamente el indicador C_2, lo cual es razonable ya que en esta prueba solo se balancea en train, mientras que en test se clasifican todos los consumos, siendo de esta manera, mucha más cantidad de normales que anómalos. Recordemos que el C_2 mide la relación entre los anómalos bien clasificados entre los clasificados como anómalos, si aumenta la cantidad de normales que clasificamos, también aumenta cuantos de estos son mal clasificados, disminuyendo el cociente anterior.

- Aunque en el resto de las pruebas los resultados son similares, la mejor performance general aparece en las pruebas 5 y 7. Esto se explica a que en éstas el porcentaje de anómalos en el balanceo es del 60 %, por lo tanto, al ser mayoritarios los anómalos en entrenamiento, a la hora de clasificar se va a cometer menos errores en éstos.

4.4. Resultados OPF con aprendizaje

Las mismas pruebas que se realizaron en la sección anterior con el clasificador OPF sin aprendizaje se volvieron a realizar agregándole la parte de aprendizaje. A modo de comparación con los resultados expresados anteriormente se muestran los valores de los indicadores para el caso de la base "consumos con fecha" y las características desarrolladas en este proyecto (conjunto 3)

	1	2	3	4	5	6	7	8	9
C_1	36	35	61	60	64	63	64	66	67
C_2	35	35	67	62	70	66	71	14	14
C_3	62	61	65	65	66	65	66	67	67

Analizando estos resultados, y comparándolos con los del mismo conjunto en la sección anterior, nos damos cuenta que en realidad el algoritmo de aprendizaje no mejora la performance del clasificador. La única diferencia, que no se desprende directamente de los datos anteriores, pero que sí percibimos es que aunque el promedio sea similar a el caso sin aprendizaje, la dispersión de los resultados es mucho mayor dependiendo del conjunto de entrenamiento. Por este motivo es que decidimos no usar el algoritmo de aprendizaje para lo que va a ser el clasificador final.

Capítulo 5

Conclusiones

En este trabajo se propone un avance en materia de generar una herramienta capaz de brindar apoyo a la hora de detectar fraude en consumidores de energía eléctrica. Si bien existen algunas investigaciones que tratan esta temática [9][6], se pretende abordar un tema actual del cual no se encontró a nivel mundial un desarrollo fuerte y consolidado. Por otra parte en nuestro país tampoco se cuenta con mucho camino recorrido en materia de reconocimiento de fraude por medio de algoritmos de reconocimiento de patrones, si bien ya se cuenta con un trabajo previo realizado por los propios técnicos de UTE [15]. Los métodos que se presentaron en este trabajo ya han sido aplicados en algunas de las referencias, sin embargo, no se encontró ningún trabajo que analizara todos los métodos de manera combinada. Además en esta oportunidad se esta desarrollando una aplicación que pretende ajustarse a las necesidades particulares de la división de detección de fraude de UTE, más allá de hacer un análisis de los métodos existentes en la literatura y una evaluación de los mismos.

En lo que concierne a esta instancia en particular, enmarcados en el trabajo final de la materia ”Int. Al Reconocimiento de Patrones”. Este documento pretende transmitir el manejo adquirido de algunas herramientas y algoritmos de reconocimiento de patrones, así como también de la teoría general. Se abordo un problema de punta a punta, desde la propuesta de las características, hasta la búsqueda de conocimientos previo, pasando por el estudio de las bibliotecas disponibles en la web y cómo implementar, combinar y adaptar las mismas a los datos con los que se cuenta. Paralelamente se intentó en el transcurso del tiempo no perder contacto con el cliente final de este proyecto, los técnicos de UTE quienes en última instancia pondrán a prueba las herramientas que se están desarrollando.

5.1. Evaluación de los resultados

Se implementaron 3 tipos de clasificadores: C-SVM, One-Class-SMV y OPF, presentados en la secciones 3.1, 3.3 y 4.1 respectivamente. En esta etapa se pretende realizar una breve comparación de los desempeños de los mismos en las distintas condiciones en que se han ido ensayando. Cabe señalar que a los efectos finales de este proyecto, no será tan importante cuanto se destaca un clasificador con respecto a los demás pues se pretende utilizar de manera combinada el veredicto de cada uno para obtener una etiqueta final. Por esta razón, se espera que el desempeño final (luego de la combinación de los clasificadores) sea superior incluso al

desempeño del mejor de los clasificadores (considerados individualmente).

El desempeño de los clasificadores será comparado en término de los indicadores introducidos en 3.2.

Si observamos en primer lugar los datos obtenidos para C-SVM, podemos notar, que el clasificador arroja excelentes resultados[a] para la base de datos ”cons-etiq-pru2”, esta base de consumos fue la utilizada en [15] y recordemos que consistía en un conjunto reducido de consumos con características particulares y bien identificadas por tratarse de autoservicios. Sin importar las características que se utilizan (las propuestas en este trabajo o las propuestas en [15]) el resultado para la base de datos antes mencionada es muy bueno, en particular el mejor resultado se obtiene para las características propuestas en [15].

¿Qué pasa si ampliamos el universo de trabajo y evaluamos con la base de dato ”Listado-con-fechaV2”?

Si trabajamos con la base de datos más genérica, los resultados son naturalmente peores. En este caso se obtienen mejores resultados para las características planteadas en este trabajo comparado con la performance que se obtiene con las características propuestas en [15], de todas maneras la diferencia en el desempeño del clasificador con uno o el otro set de características no es significativo. Lo que sí es significativo es el cambio del desempeño al utilizar la base de datos más general. Si bien se mantiene un buen índice de consumos anómalos detectados (primera prioridad de este trabajo), el decremento mayor en la performance se ve reflejado en el incremento del numero de consumos normales mal clasificados.

En segundo lugar, si se analizan los resultados obtenidos para One Class SVM[b], se puede notar una fuerte correlación con los resultados obtenidos para C-SVM. Al igual que en el caso anterior, los cambios más drásticos en la performance se dan al cambiar la base de datos. Mientras que para las distintas características planteadas el método parece comportarse de manera más o menos uniforme.

En tercer lugar, analizamos los resultados obtenidos para el clasificador OPF en las variantes con y sin aprendizaje, los resultados numéricos se muestran en la página 32. Al igual que para los clasificadores anteriores, no se encuentran diferencias significativas al cambiar el set de características, sin embargo en este caso se pueden observar algunas particularidades de este método. Si bien este método es menos preciso a la hora de detectar consumos anómalos, comete menos errores que los métodos anteriores a la hora de identificar consumos normales.

5.2. Trabajos a futuro

Este trabajo se enmarca en un proyecto más general en el cual se seguirá avanzando y estudiando muchas cuestiones que aquí se dejaron planteadas. A continuación se enumeran algunas de las futuras mejoras que se pretende implementar en los futuros meses de trabajo.

[a] consultar la página 21 para ver valores numéricos
[b] consultar la página 29

1. Ampliar la base de datos de estudio.
   Actualmente se dispone de una tercer base de datos para profundizar los análisis y estudiar el desempeño de las técnicas propuestas en conjuntos más generales. Esta base de datos, adquirida recientemente gracias a la colaboración del personal de UTE, consta de unos 37000 consumos algunos de los cuales corresponden a clientes fraudulentos que se les ha constatado irregularidades durante inspecciones. Si bien esta base de datos ofrece algunas posibilidades y permite validar muchos resultados, presenta algunos inconvenientes que deberán ser sorteados. Por ejemplo, la base se encuentra en un formato bastante distinto al que se esta utilizando, es necesario realizar las adaptaciones y conversiones necesarias. En segundo lugar, no se cuenta con etiquetas explicitas, es decir, se tiene certeza de que aquellos consumos a los cuales se les detecto irregularidades corresponden a consumos fraudulentos pero no hay ninguna seguridad de que los consumos a los que no se inspecciono sean normales. Se establecerán criterios para evaluar la performance de los clasificadores, teniendo en cuenta las consideraciones antes mencionadas y para obtener estimaciones razonables del desempeño del clasificador a pesar de no contar con etiquetas para todas las muestras de la base.

2. Realizar inspecciones para validar los resultados obtenidos.
   Teniendo en mente las dificultades planteadas en el item anterior, se estableció de común acuerdo con el personal de UTE la realización de aproximadamente 500 inspecciones a clientes seleccionados por los algoritmos finales. Se pretende comparar el índice de fraudes detectados utilizando la nueva herramienta que se esta diseñando para compararlo con los índices de detección obtenidos mediante inspecciones de rutina.

3. Incluir nuevas características.
   Para la nueva base de datos que se pretende abordar, se dispone de información adicional como puede ser el tipo de contador con el que cuenta el cliente[c] y la potencia contratada. Se pretende incorporar dicha información para ampliar el rango de la características planteadas en este trabajo.

4. Métodos de extracción y selección de características.
   Actualmente se esta trabajando con un conjunto de 20 características y se pretende en el corto plazo incorporar algunas características adicionales. Trabajar en un espacio de características muy elevado tiene aparejado costos en el tiempo de procesamiento. Además, características inservibles reducen la performance de los clasificadores y en algunos casos pueden empeorar el desempeño de manera concluyente, por ejemplo en técnicas propuestas como OPF. Teniendo en cuenta lo anterior será necesario incorporar una etapa de selección y extracción de características que reduzca la dimensionalidad del problema así como también elimine características ruidosas. Ya se ha implementado parcialmente algún método de selección pero aun se encuentra en las primeras etapas de prueba razón por la cual no se decidió incluirlo en este trabajo.

[c] digital o analógico

5. Nuevas alternativas para seleccionar las muestras que se descartan a la hora de balancear la base. Algunos de los algoritmos propuestos en este trabajo requieren ser entrenados con bases de datos balanceadas para un mejor desempeño. Actualmente dicho balanceo se realiza simplemente descartando parte de los vectores normales de la base de datos sin un criterio especifico. Algunas de las referencias que se citan en este trabajo (como [4]) implementan métodos de sub-muestreo de la base que mejoran el rendimiento para algunos de los clasificadores estudiados. Por dichas razones se contemplara la posibilidad de incluir nuevos criterios a la hora de obtener bases de datos balanceadas y se estudiara en cada caso las respectivas incidencias que esto ocasiona en el error de los clasificadores.

6. Combinación de clasificadores.
   Como última linea de trabajo, se pretende combinar la etiqueta que arrojan los clasificadores individualmente para obtener un veredicto unificado en cuando a la naturaleza de cada consumo. De este modo se combinaran los clasificadores para obtener una mejora de los desempeños particulares. Existen distintas alternativas a la hora de combinar clasificadores algunas de las cuales serán objetos de estudio durante los próximos meses de trabajo.

7. Estudiar nuevos criterios de evaluación para los clasificadores.

Bibliografía

[1] R. Tanscheit C. Muniz, M. Vellasco and K. Figueiredo. Ifsa-eusflat 2009 a neuro-fuzzy system for fraud detection in electricity distribution.

[2] N. Cristianini and J. Shawe-Taylor. *An Introduction to Support Vector Machines*. Cambridge, MA: Cambridge Univ, 2000.

[3] S. Dumais. Using svms for text categorization. *IEEE Intell. Syst. Mag., Support Vector Machines, vol. 13, no. 4, pp. 21-23*, 1998.

[4] Xinjian Guo and Guangtong Zhou. On the class imbalance problem. *IIE - Computer Society*, 1:192, 2008.

[5] Stephane Mallat. *A Wavelet Tour of Signal Processing, Second Edition (Wavelet Analysis & Its Applications)*. Academic Press, 1999.

[6] Jawad Nagi and Malik Mohamad. Nontechnical loss detection for metered customers in power utility using support vector machines. *IEEE TRANSACTIONS ON POWER DELIVERY, VOL. 25, NO. 2*, 2010.

[7] E. Osuna. Applying svms to face detection. *IEEE Intell. Syst. Mag., Support Vector Machines, vol. 13, no. 4, pp. 23-26*, 1998.

[8] Joao Paulo Papa and Alexandre Xavier Falcao. Optimum-path forest: A novel and powerful framework for supervised graph-based pattern recognition techniques. *Institute of Computing University of Campinas*, 2010.

[9] Harry Tagaris Rong Jiang and Andrei Laschusz. Wavelets based feature extraction and multiple cassifiers for electricity fraud detection.

[10] Ana María Clara Ruedin. Introducción a las wavelets.

[11] Bernhard Scholkopf and Alexander J. Smola. *Learning with Kernels*. The MIT Press, London, 2. edition, 2002.

[12] Ingo Steinwart and Andreas Christmann. *Support Vector Machines*. Springer, USA, 1. edition, 2006.

[13] V. N. Vapnik. *The Nature of Statistic Learning Theory*. New York: Springer, 1995.

[14] V. N. Vapnik. *Statistical Learning Theory*. New York: Wiley, 1998.

[15] Diego Alcetegaray y Juan Pablo Kosut. One class svm para la detección de fraudes en el uso de energía eléctrica. *Trabajo Final Curso de Reconocimiento de Patrones, Dictado por el IIE- Facultad de Ingeniería- UdelaR*, 2008.