ONE CLASS SVM para la detección de fraudes en el uso de energía eléctrica

Diego Alcetegaray, Juan Pablo Kosut

1. Abstract

En este trabajo se presenta la aplicación de la técnica SVM de una clase para la detección de anomalías en los consumos eléctricos de almacenes y autoservicios.
Se analizan distintas funciones kernel y se utiliza la función kernel gaussiana para la implementación del algoritmo de clasificación.
Se analiza el desempeño del clasificador realizando un test sobre una muestra etiquetada mediante el análisis visual de los consumos. El mismo fue realizado por un grupo de técnicos dedicados a la tarea de análisis de consumos de suministros de UTE en Montevideo.

2. Introducción

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, ascendiendo las mismas a 20,9% de acuerdo al balance de Noviembre de 2008.

Para el presente trabajo interesa separar del total, 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 del análisis, y las últimas por tratarse de pérdidas ya plenamente identificadas.

Restando del total las anteriores, pueden estimarse las pérdidas por fraude en Montevideo en aproximadamente el 4% de la energía entrante. Tomando como hipótesis que en promedio los clientes fraudulentos pagan menos del 50% de la energía que consumen, se podría decir que hay menos del 8% de suministros con anomalías en el total de los 507.000 suministros que tiene UTE en Montevideo.

El problema entonces se convierte en la necesidad de detectar el subconjunto minoritario de clientes en cuyos suministros existen irregularidades que no permiten el correcto registro del total de la energía consumida

3. ONE CLASS SVM

El objetivo de fondo es el de construir un clasificador que separe los clientes fraudulentos de los no fraudulentos.
Se podría pensar que para tal fin sería necesario realizar una estimación de densidades. Sin embargo ésta técnica busca resolver un problema más simple, que consiste en capturar una región en donde se concentra la densidad de probabilidad, es decir

encontrar una función que separe la gran mayoría de los datos hacia un lado (suministros normales), identificando de esta forma las "novedades" (posibles suministros con irregularidades) como los datos que quedan del otro lado de la superficie de decisión. Esto tiene la ventaja de no resolver un problema más general del que en realidad se necesita, además de que será también aplicable a los casos en que la distribución no esté completamente definida, por ejemplo si la misma tiene componentes singulares

La forma funcional de la superficie de decisión esta dada por una serie kernel, en términos de los SV (vectores soporte). Los coeficientes de la serie se encuentran resolviendo un problema de programación cuadrática.
El algoritmo es una extensión natural del SVC (support vector classifier) para el caso de información no etiquetada.

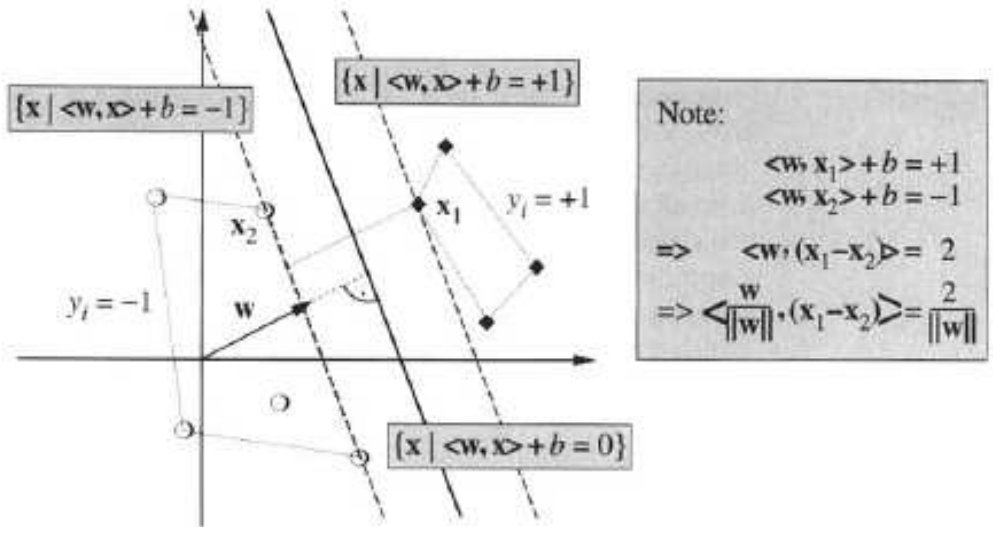
4. Support Vector Classifiers.

4.1. Clasificación binaria. Híperplano de separación.

Dada una muestra etiquetada, para la que se conoce la existencia de dos clases, se pretende encontrar el hiperplano óptimo de separación entre ambas. Si el mismo existe, se dice que las clases son linealmente separables.

$$\left\{x \in H \middle| \langle w, x\rangle + b = 0\right\}, \quad w \in H, b \in R$$

Forma canónica: Se obtiene escalando w y b de tal forma que $\left|\langle w, x_i\rangle + b\right| = 1$



El conjunto de muestras etiquetadas $(x_i, y_i) \in H \times \{\pm 1\}$ es separable si existe una función $f_{wb}(x_i) = \text{sgn}(\langle w, x_i\rangle + b)$ tal que $f_{wb}(x_i) = y_i \quad \forall i$

4.2. Margen de separación.

El margen geométrico de un punto $(x, y) \in H \times \{\pm 1\}$ es $\rho_{wb}(x, y) := y(\langle w, x \rangle + b) / \|w\|$

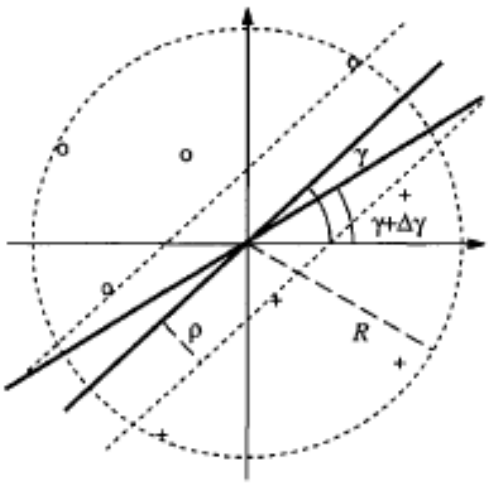
El margen geométrico de la muestra $(x_1, y_1)....(x_m, y_m)$ es $\rho_{wb} := \min(\rho_{wb}(x_i, y_i))$

Si se tiene un hiperplano en la forma canónica el margen es $1/\|w\|$

Es intuitivo que si es posible separar las muestras de entrenamiento con un margen amplio, entonces el clasificador funcionará bien con los datos de testeo.
Es posible demostrar esto en la teoría, introduciendo los conceptos de error de margen y capacidad de clasificación.

Teorema 1 (Cota de error): Dada una función de decisión $f(x) = \text{sgn}\langle w, x \rangle \; con \; \|w\| \leq \Lambda \; y \; \|x\| \leq R$ para algún $R, \Lambda > 0$. Siendo $\rho > 0$ y ν la fracción de muestras de entrenamiento con margen menor que $\rho/\|w\|$ (error de margen).

Para cualquier distribución P generadora de datos, con probabilidad de al menos $1 - \delta$ para una muestra de tamaño m, y para cualquier $\rho > 0, \delta \in (0,1)$ la probabilidad de que un patrón de testeo generado por la misma P quede mal clasificado tiene como cota superior $\nu + \sqrt{\frac{c}{m}(\frac{R^2\Lambda^2}{\rho^2}\ln^2 m + \ln(\frac{1}{\delta})}$



La cota indicada es la suma del error de margen de las muestras de entrenamiento ν, y un término de capacidad que para mantenerlo chico debe hacerse grande ρ pero que justamente va a implicar que crezca el error de margen.
Este término de capacidad tiene a cero cuando el número de muestras tiende a infinito.
Por tanto se desprende que debe buscarse un hiperplano alineado de tal forma que aún para valores grandes del margen existan pocos patrones de entrenamiento dentro del mismo.

4.3. Margen óptimo.

Buscamos el hiperplano con el máximo margen posible sujeto a que no existan errores de margen (partiendo de la base de que se trata de de una muestra linealmente separable)

$$\min_{w\in H, b\in R} \tau(w) = \frac{1}{2}\|w\|^2$$

$$sujeto\ a: y_i(\langle w, x_i\rangle + b) \geq 1$$

De acuerdo a la teoría de optimización, debe obtenerse a partir del problema primal planteado, el problema dual, dado que puede demostrarse que es más conveniente trabajar con este último obteniéndose la misma solución. Para esto se introduce el Lagrangeano.

$$L(w,b,\alpha) = \frac{1}{2}\|w\|^2 - \sum \alpha_i (y_i(\langle w, x_i\rangle + b) - 1)$$

El Lagrangeano debe maximizarse respecto de los multiplicadores α_i, y minimizarse respecto de $w\ y\ b$.

- Si se viola una restricción $((\langle w, x_i\rangle + b) - 1) < 0$, entonces α_i tiende a crecer para que aumente L.
- $w\ y\ b$ deben moverse para bajar L, entonces se mueven hasta satisfacer la igualdad de la restricción. $((\langle w, x_i\rangle + b) - 1) = 0$
- En los puntos en que se satisface la desigualdad $((\langle w, x_i\rangle + b) - 1) > 0$ de la restricción el valor del multiplicador que maximiza L es $\alpha_i = 0$

Lo anterior es una explicación intuitiva de cómo las variables tiran cada una para su lado buscando la solución de compromiso: máximo margen, sin violación de las restricciones.

(I) $\frac{\partial}{\partial b} L(w,b,\alpha) = 0 \rightarrow \sum_{I=1}^{m} \alpha_i y_i = 0$

(II) $\frac{\partial}{\partial w} L(w,b,\alpha) = 0 \rightarrow w = \sum_{I=1}^{m} \alpha_i y_i x_i$

Observar que el vector solución w se expresa en función de las muestras de entrenamiento, pero sólo de aquellas para las que $\alpha_i \neq 0$. Estas son llamadas vectores soporte (SV)

De acuerdo a las condiciones KKT de la teoría de optimización, los vectores para los cuales $\alpha_i > 0$ son los que cumplen con la igualdad de las restricciones

$((\langle w, x_i \rangle + b) - 1) = 0$, es decir los que se encuentran justo sobre el margen. El resto de las muestras de entrenamiento son irrelevantes para la solución.

De este hecho puede deducirse una cota superior para la probabilidad de clasificar erróneamente una muestra de testeo.
Proposición: El valor esperado de la cantidad de vectores soporte obtenidos durante el entrenamiento con una muestra de tamaño m, dividido m, es una cota superior del valor esperado de la probabilidad de error de testeo, de un clasificador entrenado con una muestra de tamaño m-1.

Volviendo al problema de optimización, sustituyendo (I) y (II) en el Lagrangeano se obtiene el problema dual:

$$\max_{\alpha \in R^m} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_y y_j \langle x_i, x_j \rangle$$

$$sujeto\ a: \quad \alpha_i \geq 0,\ y \sum_{i=1}^{m} \alpha_i y_i = 0$$

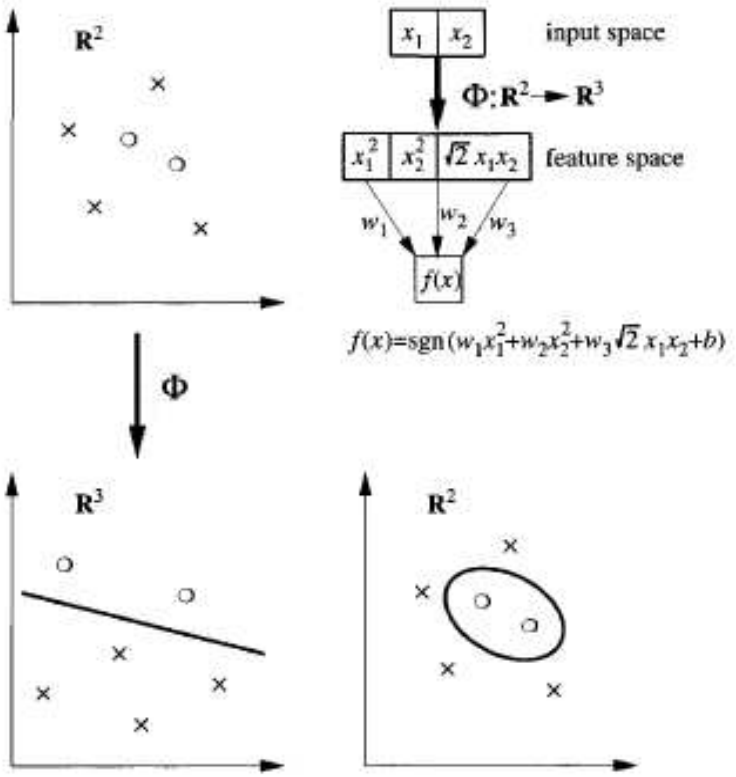
Sustituyendo para la función de decisión se obtiene.

$$f(x) = \text{sgn}(\sum_{i=1}^{m} \alpha_i y_i \langle x, x_i \rangle + b)$$

4.4. Clasificadores SVC no lineales

Con el objetivo de manejar superficies de decisión más generales, se usan kernels para transformar de manera no lineal los datos de entrada, en un espacio de características de mayor dimensión en donde se realiza la separación lineal.

Se utiliza una función de mapeo $\Phi : x_i \rightarrow x_i'$



La función de decisión es lineal en el espacio de características R^3 y representa una superficie más compleja en el espacio de muestras R^2.
Para que el algoritmo funcione es necesario que esté definido el producto escalar en el espacio de características, en cuyo caso basta con calcular en el problema de programación cuadrática el producto escalar $\langle \Phi(x), \Phi(x_i) \rangle$
Este complejo cálculo se simplifica significativamente utilizando un kernel tal que.

$$\langle \Phi(x), \Phi(x_i) \rangle = k(x, x_i)$$

De esta forma todo lo que se ha establecido para el caso lineal es válido para el caso no lineal usando un kernel apropiado.
El kernel puede visualizarse como una medida se similitud entre dos patrones.

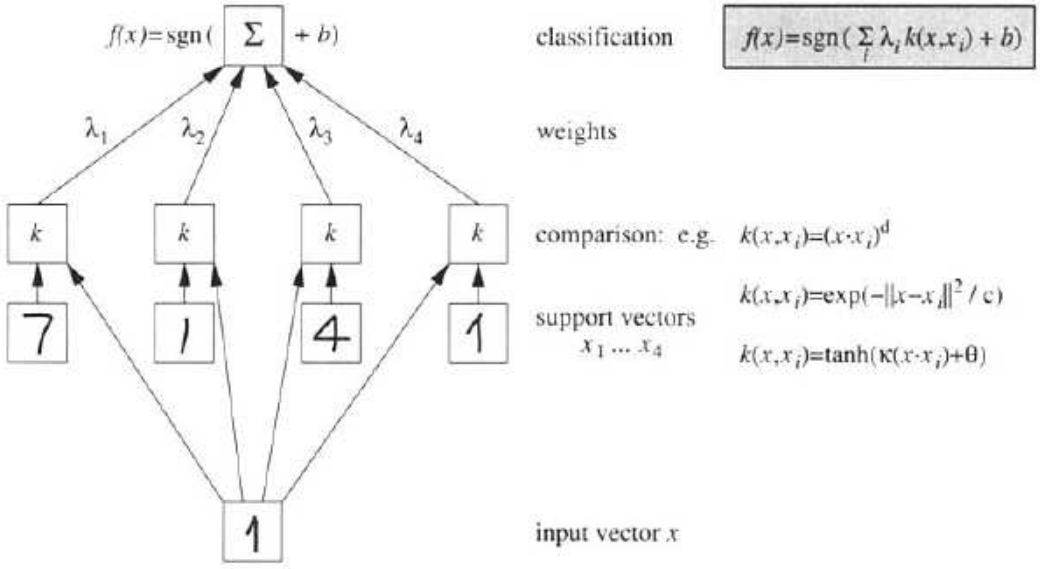


Figure 7.7 Architecture of SVMs. The kernel function k is chosen a priori; it determines the type of classifier (for instance, polynomial classifier, radial basis function classifier, or neural network). All other parameters (number of hidden units, weights, threshold b) are found during training, by solving a quadratic programming problem. The first layer weights x_i are a subset of the training set (the Support Vectors); the second layer weights $\lambda_i = y_i \alpha_i$ are computed from the Lagrange multipliers (cf. (7.25)).

El problema de programación cuadrática se transforma en:

$$\max_{\alpha \in R^m} W(\alpha) = \sum_{i=1}^{m} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{m} \alpha_i \alpha_j y_y y_j k(x_i, x_j)$$

$$sujeto\ a: \quad \alpha_i \geq 0,\ y \sum_{i=1}^{m} \alpha_i y_i = 0$$

Y la función de decisión:

$$f(x) = \text{sgn}(\sum_{i=1}^{m} \alpha_i y_i k(x, x_i) + b)$$

El valor de b se encuentra teniendo en cuenta que de las condiciones KKT $\sum_{i=1}^{m} y_i \alpha_i k(x_i, x_j) + b = y_j$ para $\alpha_j > 0$, es decir que se puede calcular por ejemplo promediando para todos los puntos con $\alpha_j > 0$.

4.5. Restricciones relajadas. (caso no separable)

Se pretende desarrollar un algoritmo que sea capaz de tolerar una cierta fracción de outliers, debido a que de no tenerse en cuenta, un patrón mal clasificado podría afectar radicalmente la definición del hiperplano obteniéndose una solución no adecuada.

Una forma de habilitar la posibilidad de que ciertas muestras violen las restricciones, es introducir las llamadas "slack variables" para producir restricciones relajadas

$$y_i(\langle x_i, w\rangle + b) \geq 1 - \varepsilon_i$$

De forma de no obtener la solución trivial en la que todos los ε_i son suficientemente grandes, es necesario penalizar su valor en la función objetivo.

- Clasificador C-SVC

$$\min_{w\in H, b\in R} \tau(w,\varepsilon) = \frac{1}{2}\|w\|^2 + \frac{C}{m}\sum_{i=1}^{m}\varepsilon_i$$
$$sujeto\ a: y_i(\langle w, x_i\rangle + b) \geq 1 - \varepsilon_i$$
$$\varepsilon_i \geq 0$$

Haciendo referencia al teorema del error de margen, considerando $\rho = 1$ se puede observar cada variable $\varepsilon_i > 0$ corresponde a un error de margen, por lo que el primer término de la cota del error de testeo crece directamente con $\frac{C}{m}\sum_{i=1}^{m}\varepsilon_i$, mientras que por otro lado el término de capacidad crece con $\|w\|$. Por tanto para un valor adecuado de la constante C, la optimización planteada minimiza aproximadamente el error de testeo.
De todas formas si los datos se solapan demasiado, no hay garantías de que el hiperplano funcione bien.

Debido a que no existen formas intuitivas de seleccionar la constante C, se propone una modificación que remplaza esta constante por otro parámetro ν que guarda relación directa con el error de margen y la cantidad de vectores soporte.

- Clasificador $\nu - \text{SVC}$

$$\min_{w\in H, b\in R} \tau(w,\varepsilon,\rho) = \frac{1}{2}\|w\|^2 - \nu\rho + \frac{1}{m}\sum_{i=1}^{m}\varepsilon_i$$
$$sujeto\ a: y_i(\langle w, x_i\rangle + b) \geq \rho - \varepsilon_i$$
$$\varepsilon_i \geq 0,\ \rho \geq 0$$

Además de aparecer ν sustituyendo a C, aparece una nueva variable ρ a ser optimizada.

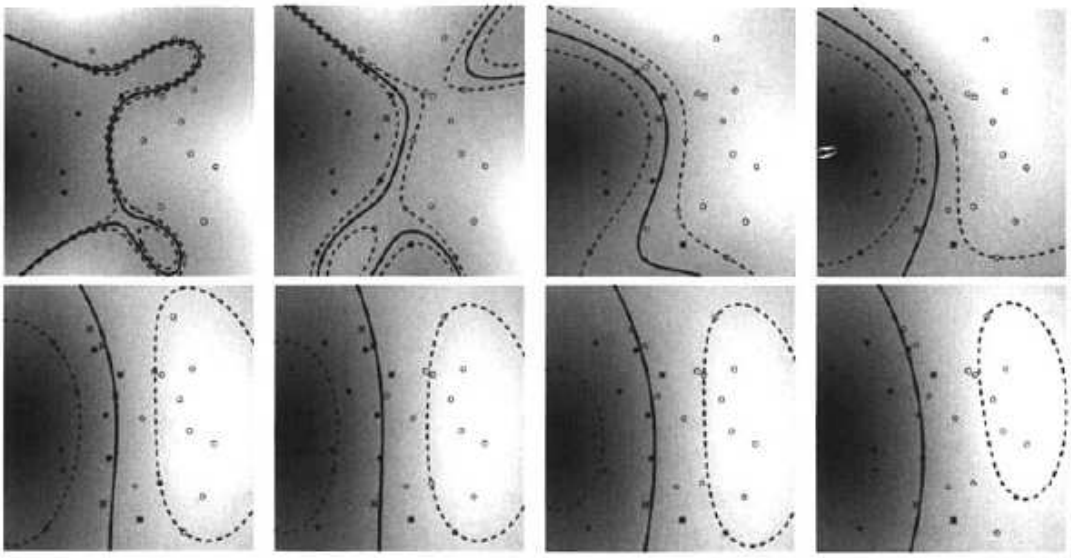
Para entender el rol de ρ observar que para $\varepsilon = 0$, la restricción implica que las dos clases están separadas por el margen $2\rho/\|w\|$, es decir que para maximizar el margen ahora hay que minimizar $\|w\|$ y maximizar ρ.

El significado de ν surge del siguiente resultado.

Proposición 1: Si se corre el algoritmo $\nu-$SVC sobre un conjunto de datos con el resultado de $\rho > 0$, entonces:

i) ν es una cota superior de la fracción de errores de margen (puntos que o son errores o están dentro del margen)
ii) ν es una cota inferior de la fracción de vectores soporte (SVs)

En la figura siguiente se observa un ejemplo de resolución variando el parámetro entre los valores $\nu = 0.1\, a\, \nu = 0.8$
Al aumentar el parámetro se habilita a que una mayor cantidad de puntos caigan dentro del margen, es decir se aumenta el margen.



ν	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8
fraction of errors	0.00	0.07	0.25	0.32	0.39	0.50	0.61	0.71
fraction of SVs	0.29	0.36	0.43	0.46	0.57	0.68	0.79	0.86
margin $\rho/\|\mathbf{w}\|$	0.005	0.018	0.115	0.156	0.364	0.419	0.461	0.546

Análogamente a los casos anteriores, considerando el Lagrangeano, e incorporando los kernel se obtiene el problema dual para el clasificador $\nu-$SVC

$$\max_{\alpha \in R^m} W(\alpha) = -\frac{1}{2}\sum_{i,j=1}^{m} \alpha_i \alpha_j y_i y_j k(x_i, x_j)$$

$$\textit{sujeto a}: \quad 0 \le \alpha_i \le \frac{1}{m}, \quad \sum_{i=1}^{m} \alpha_i y_i = 0, \quad \sum_{i=1}^{m} \alpha_i \ge \nu$$

Y la función de decisión:

$$f(x) = \operatorname{sgn}(\sum_{i=1}^{m} \alpha_i y_i k(x, x_i) + b)$$

Comparado con el C-SVC

- Hay una restricción adicional
- La función objetivo es homogénea en α
- Se demuestra que si $\nu - \text{SVC}$ arroja un resultado con $\rho > 0$, entonces C-SVC con $C = 1/\rho$ conduce a la misma función de decisión.

4.5.1. Experimentos, selección de parámetros

Se utiliza la base de dígitos manuscritos (USPS) para evaluar los clasificadores, y comparar el desempeño de los distintos tipos, obtenidos utilizando distintos kernels.
Se observa que el desempeño de las tres funciones kernel es muy similar, lo cual sugiere que la selección del kernel es menos importante que la selección adecuada de los parámetros de cada uno, es decir $d, c, y\ \Theta$ para los kernel polinómico, gaussiano y sigmoide respectivamente.

Table 7.3 Performance on the USPS set, for three different types of classifier, constructed with the Support Vector algorithm by choosing different functions k in (7.25) and (7.29). Error rates on the test set are given; and for each of the ten-class-classifiers, we also show the average number of Support Vectors of the ten two-class-classifiers. The normalization factor of 256 is tailored to the dimensionality of the data, which is 16×16.

polynomial: $k(x, x') = (\langle x, x' \rangle / 256)^d$

d	1	2	3	4	5	6	7
raw error/%	8.9	4.7	4.0	4.2	4.5	4.5	4.7
av. # of SVs	282	237	274	321	374	422	491

RBF: $k(x, x') = \exp\left(-\|x - x'\|^2 / (256\, c)\right)$

c	4.0	2.0	1.2	0.8	0.5	0.2	0.1
raw error/%	5.3	5.0	4.9	4.3	4.4	4.4	4.5
av. # of SVs	266	240	233	235	251	366	722

sigmoid: $k(x, x') = \tanh(2\langle x, x' \rangle / 256 + \Theta)$

$-\Theta$	0.8	0.9	1.0	1.1	1.2	1.3	1.4
raw error/%	6.3	4.8	4.1	4.3	4.3	4.4	4.8
av. # of SVs	206	242	254	267	278	289	296

La selección del parámetro por su parte determina fuertemente el comportamiento del clasificador.
En la siguiente figura se muestran de izquierda a derecha los resultados obtenidos al ir decreciendo el parámetro c del kernel gaussiano.

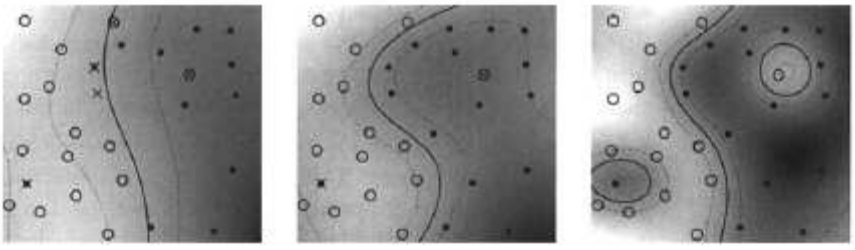


Figure 7.10 2D toy example of a binary classification problem solved using a soft margin SVC. In all cases, a Gaussian kernel (7.27) is used. From left to right, we decrease the kernel width. Note that for a large width, the decision boundary is almost linear, and the data set cannot be separated without error (see text). Solid lines represent decision boundaries; dotted lines depict the edge of the margin (where (7.34) becomes an equality with $\xi_i = 0$).

En la práctica el parámetro del kernel y el parámetro ν se eligen usando validación cruzada. Generalmente el clasificador se determina entrenando en el conjunto completo con los parámetros determinados en la validación cruzada.
Al hacer esto último debe tenerse en cuenta que puede producirse overfitting.
Otros elementos a tener en cuenta en la selección de parámetros son los siguientes:

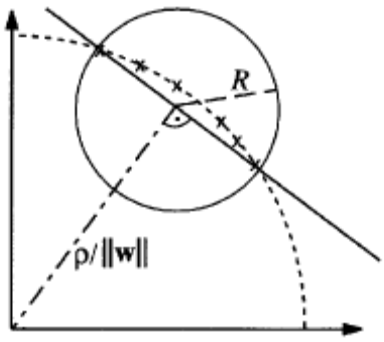
- Utilizar parámetros probados y que han funcionado bien en problemas similares, por ejemplo para el caso gaussiano intentar que $\left\|x_i - x_j\right\|^2 / c$ caiga en el mismo rango de valores
- Utilizar el hecho de que ν es una cota superior del error de margen, y éste está relacionado con el error de test por el Teorema 1, y entonces elegir el valor del parámetro en base al error de test esperado.

5. ONE CLASS SVM

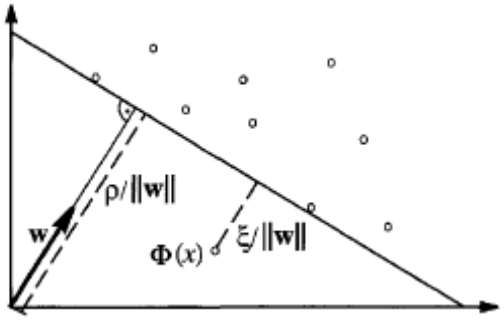
Volviendo al problema original de detección de irregularidades, y considerando la existencia de una sola clase, la de los suministros normales, se desea encontrar un algoritmo que determine una región lo mas pequeña posible en la que se encuentre concentrada la densidad de probabilidad de la clase.
Las muestras que caigan fuera de dicha región serán considerados suministros irregulares.
En base a la aplicación de SVM el algoritmo será capaz lograr este objetivo sin la necesidad de estimar la densidad completa de la distribución que genera los datos.

La estrategia, inspirada en los clasificadores mencionados anteriormente, es la de mapear los datos a un espacio de características por intermedio de un kernel, y allí separar las muestras del origen con máximo margen.

El conjunto de muestras se dice separable si existe $w / \langle w, x_i \rangle > 0 \; \forall i$
Debido a las propiedades del kernel gaussiano esto está asegurado ya que del hecho de que $k(x,x) = 1 \; \forall x$, $\left\|\Phi(x)\right\| = 1 \; \forall x$ y si además $k(x,x') > 0$, entonces todos los puntos del espacio de características están en el mismo octante.



Se pue de



demostrar que si el conjunto de datos es separable, entonces existe un único hiperplano que separa los datos del origen con máximo margen.

Para separar los datos del origen se plantea el siguiente problema de optimización.

$$\min_{w\in H,b\in R} \tau(w,\varepsilon,\rho) = \frac{1}{2}\|w\|^2 + \frac{1}{\nu m}\sum_{i=1}^{m}\varepsilon_i - \rho$$
$$sujeto\ a: \langle w, \Phi(x_i)\rangle \geq \rho - \varepsilon_i$$
$$\varepsilon_i \geq 0$$

El problema es similar al del clasificador $\nu-$SVC.

Las diferencias son:

- se trabaja con una muestra no etiquetada
- No aparece el parámetro de traslación b, ya que el margen se mide desde el origen
- Hay una restricción menos ya que no se requiere que $\rho \geq 0$

El parámetro ν se define en una marcada analogía con el clasificador $\nu-$SVC, y su significado es el mismo que en el caso anterior.
Proposición 2: Si se corre el algoritmo sobre un conjunto de datos con el resultado de $\rho \neq 0$, entonces:

iii) ν es una cota superior de la fracción de errores de margen (en este caso se trata de los outliers.
iv) ν es una cota inferior de la fracción de vectores soporte (SVs)

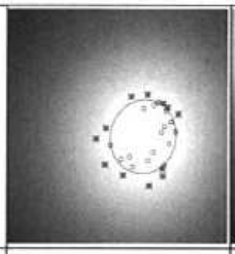
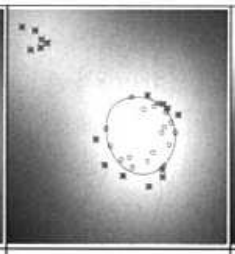
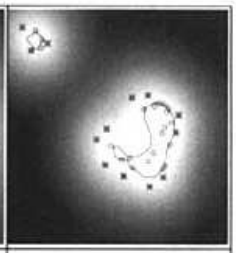
Considerando el Lagrangeano, e incorporando los kernel se obtiene el problema dual

$$\min_{\alpha\in R^m} W(\alpha) = \frac{1}{2}\sum_{i,j=1}^{m}\alpha_i\alpha_j k(x_i,x_j)$$
$$sujeto\ a: \quad 0 \leq \alpha_i \leq \frac{1}{\nu m}, \quad \sum_{i=1}^{m}\alpha_i = 1$$

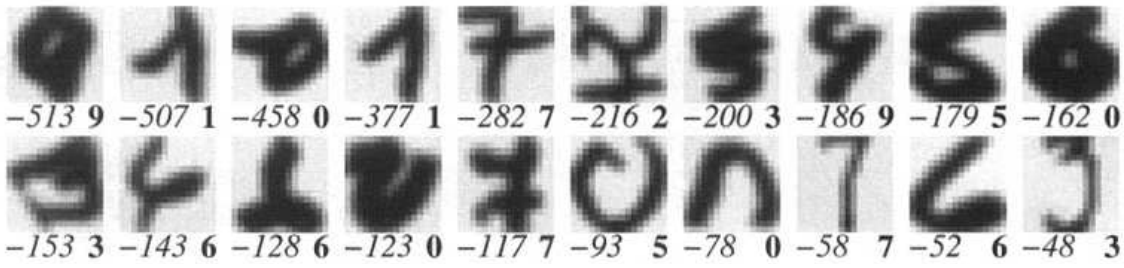
5.1. Experimentos, selección de parámetros

El desempeño del clasificador es afectado tanto por la selección de ν cómo por la selección del ancho del kernel.

- ν es una cota superior de la fracción de errores de margen. Valores chicos de este parámetro dan más peso a los outliers
- Cambiar el ancho del kernel implica que los datos sean analizados en una escala diferente, y en este caso determina que algunos outliers pasen a ser considerados como puntos importantes

				
ν, width c	0.5, 0.5	0.5, 0.5	0.1, 0.5	0.5, 0.1
frac. SVs/OLs	0.54, 0.43	0.59, 0.47	0.24, 0.03	0.65, 0.38
margin $\rho/\|\mathbf{w}\|$	0.84	0.70	0.62	0.48

Aplicado el algoritmo a los datos de USPS, se detectan los siguientes outliers ordenados de acuerdo al valor arrojado por la función de decisión. Valores negativos de mayor valor absoluto implican que la muestra está más alejada de la “normalidad”



6. APLICACIÓN A DETECCION DE CONSUMOS ANÓMALOS

6.1. Obtención de datos

De una base de datos de UTE se obtuvieron los consumos de aproximadamente 650 suministros pertenecientes a los rubros almacenes y autoservicios.
Estos se examinaron (en un período de 3 años) y se clasificaron mediante un análisis visual de los consumos, como normales (500 aprox.), y anómalos (150 aprox.).

6.2. Selección de características

Se buscaron características que pudieran representar ciertos parámetros de interés en el patrón de consumos:

- desviación estándar del consumo sobre la media – da una idea de la variación porcentual del consumo.
- valor de la pendiente al aproximar el consumo por una recta –determina la tendencia del consumo.
- los primeros 3 coeficientes de fourier[1] (sin considerar la media) –determinan la estacionalidad del consumo.

6.3. Ajuste de parámetros

Una vez seleccionadas las características, se implementa una rutina en MATLAB que ajusta los valores de ν y c (parámetro del algoritmo y ancho del kernel) en la función objetivo del problema de optimización para una clase. Se busca seleccionar estos parámetros de manera de minimizar la cantidad de suministros anómalos mal clasificados[2].
Para tal propósito se utiliza validación cruzada en 5 grupos (dada la poca cantidad de muestras para hacerlo en 10 grupos).
En la resolución del problema de programación cuadrática se utiliza la función "svmoneclass" incluida en un toolbox proporcionado por los docentes del curso.
Esta función está específicamente diseñada para resolver el problema de una clase, y se ha probado que tiene un mejor desempeño con un tiempo de ejecución mucho menor que la función "quadprog" incluida en MATLAB.

La selección de parámetros está implementada en MATLAB a través de la rutina llamada train.m con un conjunto de muestras formado por 490 suministros etiquetados como normales y 30 suministros etiquetados como anómalos[3].

---

[1] Son 3 coeficientes complejos, es decir definen 6 características.
[2] Para ajustar sigma y un se tomaron como error sólo los errores de clasificación de anómalos ya que nuestro principal interés es el de detectar consumos anormales.
[3] Se tomaron sólo 30 anómalos de manera que la cantidad de estos sea pequeña comparada a la cantidad total de suministros a clasificar.

Los resultados obtenidos son los siguientes:

```
Validación número: 1

    sigma :0.003
    nu :0.092

    cant errores en train :18 de 416
    cant anomalos mal clasificados en train :0 de 24
    cant normales mal clasificados en train :18 de 392

    cant errores en test :10 de 104
    cant anomalos mal clasificados en test :1 de 6
    cant normales mal clasificados en test :9 de 98

Validación número: 2

    sigma :0.003
    nu :0.095

    cant errores en train :20 de 416
    cant anomalos mal clasificados en train :0 de 24
    cant normales mal clasificados en train :20 de 392

    cant errores en test :10 de 104
    cant anomalos mal clasificados en test :0 de 6
    cant normales mal clasificados en test :10 de 98

Validación número: 3

    sigma :0.003
    nu :0.087

    cant errores en train :19 de 416
    cant anomalos mal clasificados en train :0 de 24
    cant normales mal clasificados en train :19 de 392

    cant errores en test :11 de 104
    cant anomalos mal clasificados en test :0 de 6
    cant normales mal clasificados en test :11 de 98

Validación número: 4

    sigma :0.003
    nu :0.074

    cant errores en train :15 de 416
    cant anomalos mal clasificados en train :0 de 24
    cant normales mal clasificados en train :15 de 392

    cant errores en test :10 de 104
    cant anomalos mal clasificados en test :0 de 6
    cant normales mal clasificados en test :10 de 98
```

```
Validación número: 5

    sigma :0.004
    nu :0.102

    cant errores en train : 21 de 416
    cant anomalos mal clasificados en train :0 de 24
    cant normales mal clasificados en train :21 de 392

    cant errores en test :6 de 104
    cant anomalos mal clasificados en test :0 de 6
    cant normales mal clasificados en test :6 de 98
```

Resumen de los valores hallados:

Corrida n°	**1**	**2**	**3**	**4**	**5**	**Promedio**
$\sigma = c/1000$	0.003	0.003	0.003	0.003	0.004	0.003
ν	0.092	0.095	0.087	0.074	0.102	0.090
err. train anómalos	0	0	0	0	0	0
err. train normales	18	20	19	15	21	19
err. test anómalos	1	0	0	0	0	0
err. test normales	9	10	11	10	6	9

El valor de ν obtenido es bajo, lo cual implica que el margen y los errores de margen serán bajos.
El pequeño margen podría implicar que la capacidad del algoritmo de clasificar correctamente nuevas muestras no fuera del todo adecuada.

El valor de σ tiene que ver con la escala a la que se analizarán los datos por el kernel.

6.4. Validación de parámeros

Con los valores promedio de la tabla anterior se ejecuta la rutina test.m sobre el total de las muestras
Los resultados son los siguientes:

PRIMER TEST

```
cantidad total de suministros: 520
cantidad de suministros etiquetados como anómalos: 30
cantidad de suministros clasificados como anómalos: 51
cantidad de suministros etiquetados como normales mal clasificados: 22
cantidad de suministros etiquetados como anómalos mal clasificados: 1
cantidad de suministros mal clasificados: 23
```

Se analizan los suministros mal clasificados.

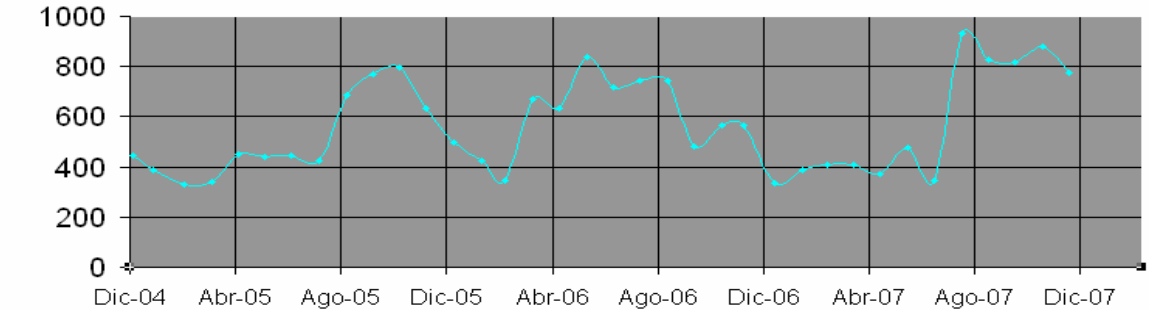
La función de decisión arroja un valor para cada suministro que es negativo si el mismo fue clasificado como anómalo y positivo si se clasificó como normal.

El valor absoluto de ese resultado permite ordenar los patrones en un ranking tal que el más alejado de la “normalidad” es el que tiene un resultado negativo de mayor valor absoluto

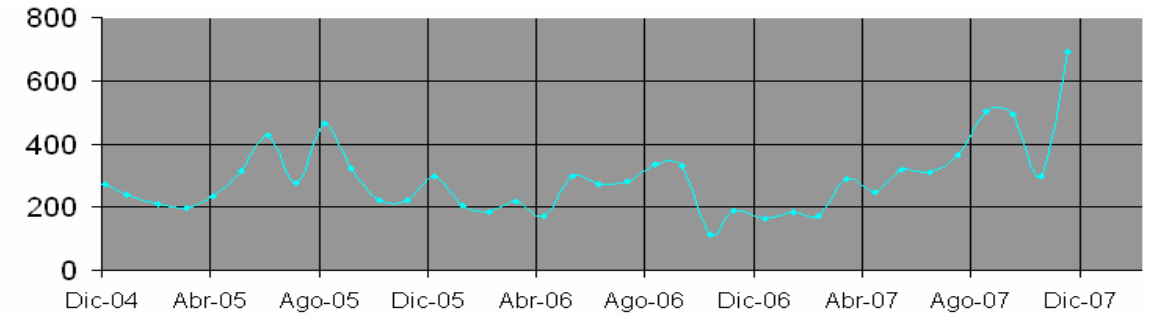
En base a esto, los suministros mal clasificados etiquetados como normales quedaron ubicados como sigue.

```
Columns 1 through 15

   14    15    22    25    27    28    31    32    36    37    38    39    40    42    43

Columns 16 through 30

   44    45    46    47    48    50    51    52    53    54    56    57    58    59    60
```

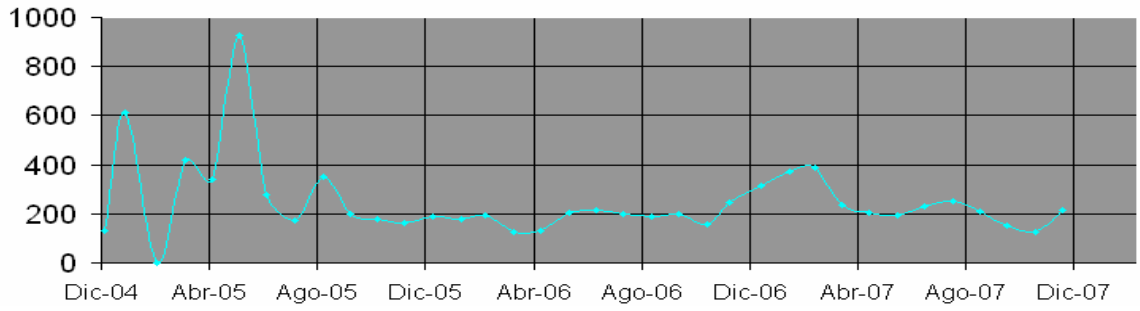
Al estudiar los suministros mal clasificados pudimos ver que habíamos cometido algunas equivocaciones en el etiquetado de los mismos.
A continuación veremos algunos de los suministros etiquetados como normales que se clasificaron como anómalos.



suministro n° 14 en la clasificación



suministro n° 15 en la clasificación.



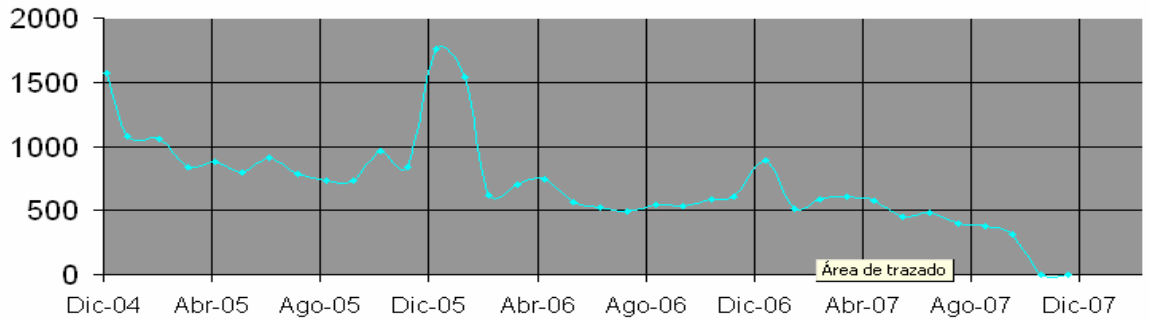
suministro n° 27 en la clasificación.

Los suministros 14 y 15 fueron clasificados como los más anómalos dentro de los que fueron etiquetados como normales.
En estos casos el algoritmo los clasificó bien, mientras que hubo un error en el etiquetado visual, debido a que se confundió la estacionalidad. La misma existe pero es justamente al revés de cómo debería ser, los consumos aumentan en invierno y bajan en verano
En el caso del suministro 27 simplemente la variación del consumo no es la adecuada, es decir que existe también un error en el etiquetado

Haciendo una segunda revisión de los consumos etiquetados como normales y clasificados como anómalos vimos que sólo 11 de los 52 presentaban consumos normales, es decir el método de clasificación nos mostró errores cometidos en el etiquetado.

El suministro anómalo etiquetado como normal es el numero 55

```
Columns 1 through 15

    1     2     3     4     5     6     7     8     9    10    11    12    13    16    17

Columns 16 through 30

   18    19    20    21    23    24    26    29    30    33    34    35    41    49    55
```



suministro n° 55 en la clasificación.

Vemos que este consumo es anómalo, pero que tiene ciertas características que lo hacen "normal", consumo mayor en invierno
El escalón hacia abajo que presenta en los últimos meses probablemente es muy pequeño como para que el método lo detecte. Probablemente al algoritmo le cuesta más identificar estos escalones cuando aparecen en un extremo del rango de fechas

6.5. Nuevo ajuste de parámetros

Luego de los errores de etiquetado evidenciados por el algorimo, se procedió a corregir las etiquetas y repetir el procedimiento
Se obtienen los siguientes resultados:

Corrida n°	1	2	3	4	5	Promedio
$\sigma = c/1000$	0.002	0.003	0.003	0.003	0.004	0.003
ν	0.147	0.188	0.195	0.191	0.182	0.181
err. train anómalos	0	0	0	0	0	0
err. train normales	12	24	28	25	22	22
err. test anómalos	0	0	0	0	0	0
err. test normales	11	9	8	8	8	9

Con los valores promedio de la tabla anterior se ejecuta la rutina test.m sobre el total de las muestras
Los resultados son los siguientes:

SEGUNDO TEST

```
cantidad total de suministros: 520
cantidad de suministros etiquetados como anómalos: 70
cantidad de suministros clasificados como anómalos: 98
cantidad de suministros etiquetados como normales mal clasificados: 28
cantidad de suministros etiquetados como anómalos mal clasificados: 0
cantidad de suministros mal clasificados: 28
```

Se clasificaron bien todos los suministros etiquetados como anómalos, lo cual es un resultados muy bueno, además vemos que se redujo de 520 suministros para revisar a 98 de los cuales 70 de ellos son anómalos.

TERCER TEST

Para ver si los parámetros encontrados no son particulares para estas muestras (es decir no sufrieron de overfitting) se repitió el test anterior pero cambiando los suministros anómalos por otros, obteniéndose los siguientes resultados:

```
cantidad total de suministros: 480
cantidad de suministros etiquetados como anómalos: 30
cantidad de suministros clasificados como anómalos: 89
cantidad de suministros etiquetados como normales mal clasificados: 59
cantidad de suministros etiquetados como anómalos mal clasificados: 0
cantidad de suministros mal clasificados: 59
```

```
1 - OK - suministro anómalo clasificado como anómalo
2 - OK - suministro anómalo clasificado como anómalo
3 - OK - suministro anómalo clasificado como anómalo
4 - OK - suministro anómalo clasificado como anómalo
5 - OK - suministro anómalo clasificado como anómalo
6 - OK - suministro anómalo clasificado como anómalo
7 - OK - suministro anómalo clasificado como anómalo
8 - OK - suministro anómalo clasificado como anómalo
9 - OK - suministro anómalo clasificado como anómalo
10 - OK - suministro anómalo clasificado como anómalo
11 - OK - suministro anómalo clasificado como anómalo
12 - OK - suministro anómalo clasificado como anómalo
13 - OK - suministro anómalo clasificado como anómalo
14 - OK - suministro anómalo clasificado como anómalo
15 - OK - suministro anómalo clasificado como anómalo
16 - OK - suministro anómalo clasificado como anómalo
17 - OK - suministro anómalo clasificado como anómalo
18 - OK - suministro anómalo clasificado como anómalo
19 - OK - suministro anómalo clasificado como anómalo
20 - OK - suministro anómalo clasificado como anómalo
21 - OK - suministro anómalo clasificado como anómalo
22 - OK - suministro anómalo clasificado como anómalo
23 - OK - suministro anómalo clasificado como anómalo
24 - OK - suministro anómalo clasificado como anómalo
25 - OK - suministro anómalo clasificado como anómalo
26 - OK - suministro anómalo clasificado como anómalo
27 - OK - suministro anómalo clasificado como anómalo
28 - ERROR - suministro normal clasificado como anómalo
29 - OK - suministro anómalo clasificado como anómalo
30 - ERROR - suministro normal clasificado como anómalo
31 - ERROR - suministro normal clasificado como anómalo
```

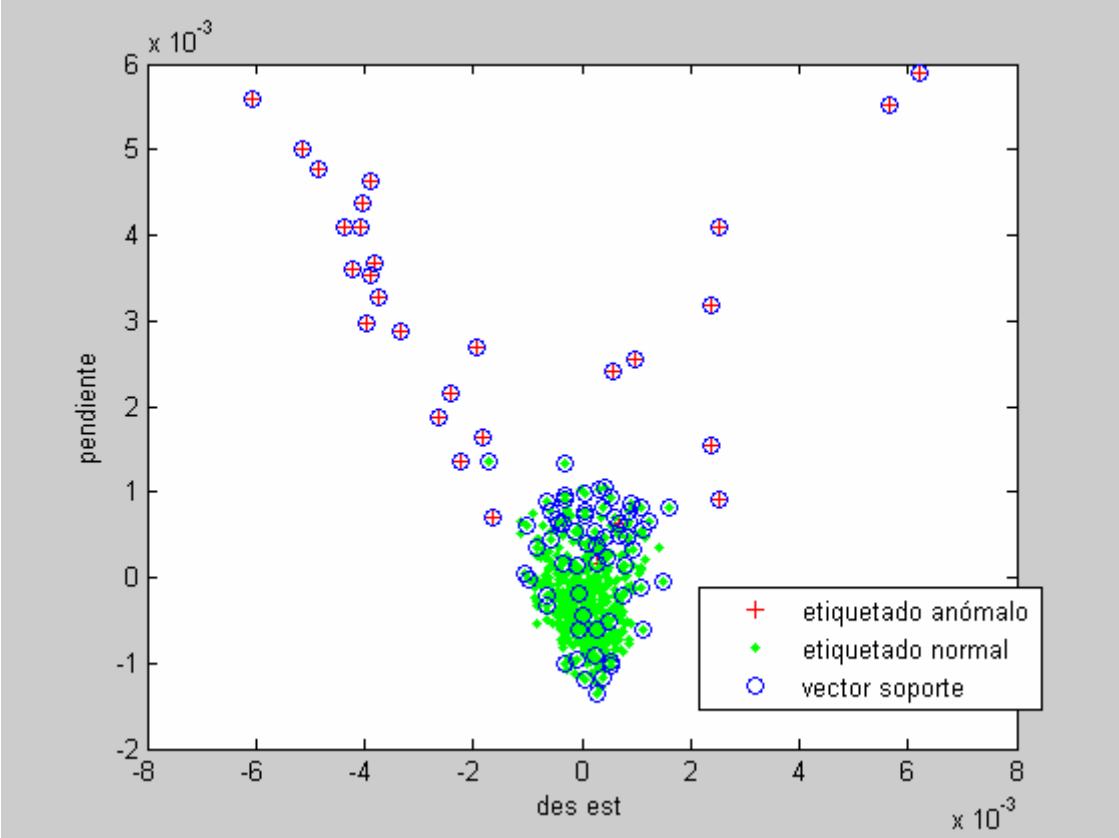
En este caso también fueron clasificados bien todos los suministros anómalos.
La cantidad de suministros a estudiar bajó de 480 a 89 de los cuales 30 de ellos son anómalos, es decir se pasó de un conjunto de estudio con una concentración de suministros sospechosos del 6% a otro con una concentración de 34%

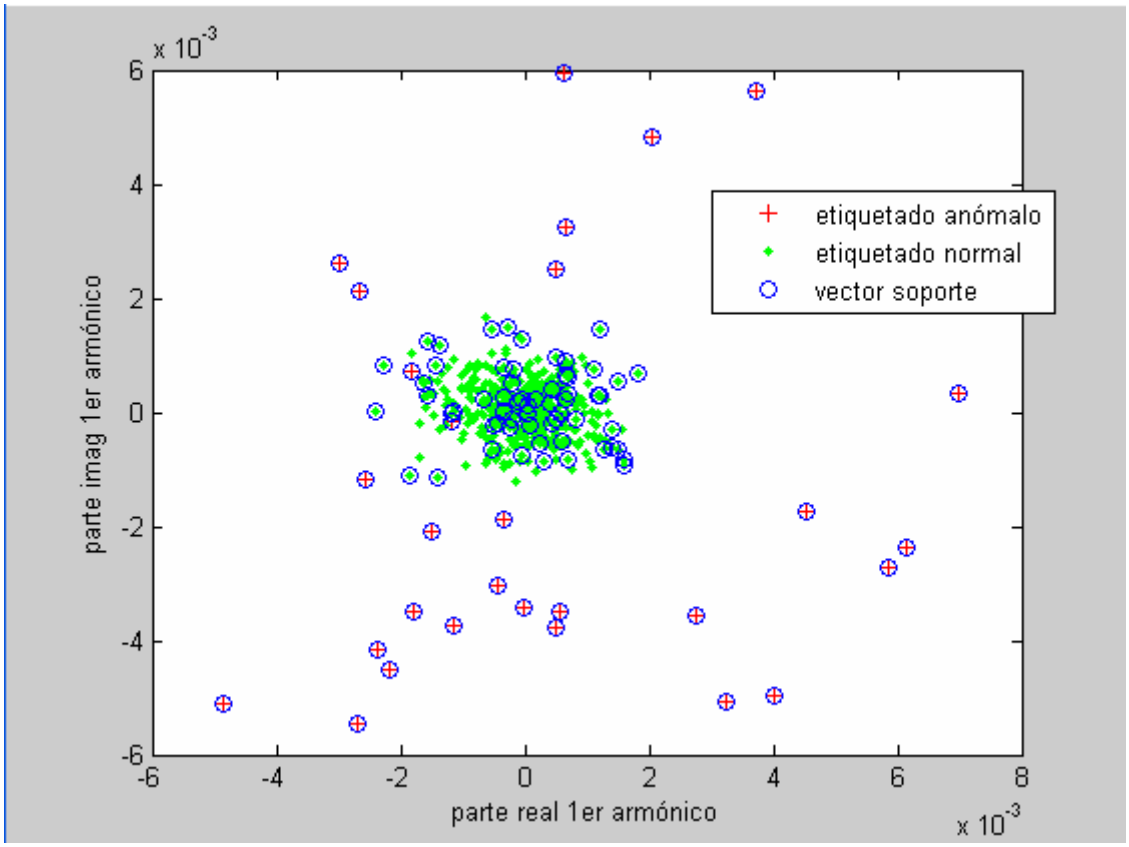
El algoritmo coloca los suministros en el orden adecuado.
Los suministros etiquetados como normales que luego fueron mal clasificados como anómalos son clasificados como los menos anómalos por parte del algoritmo.

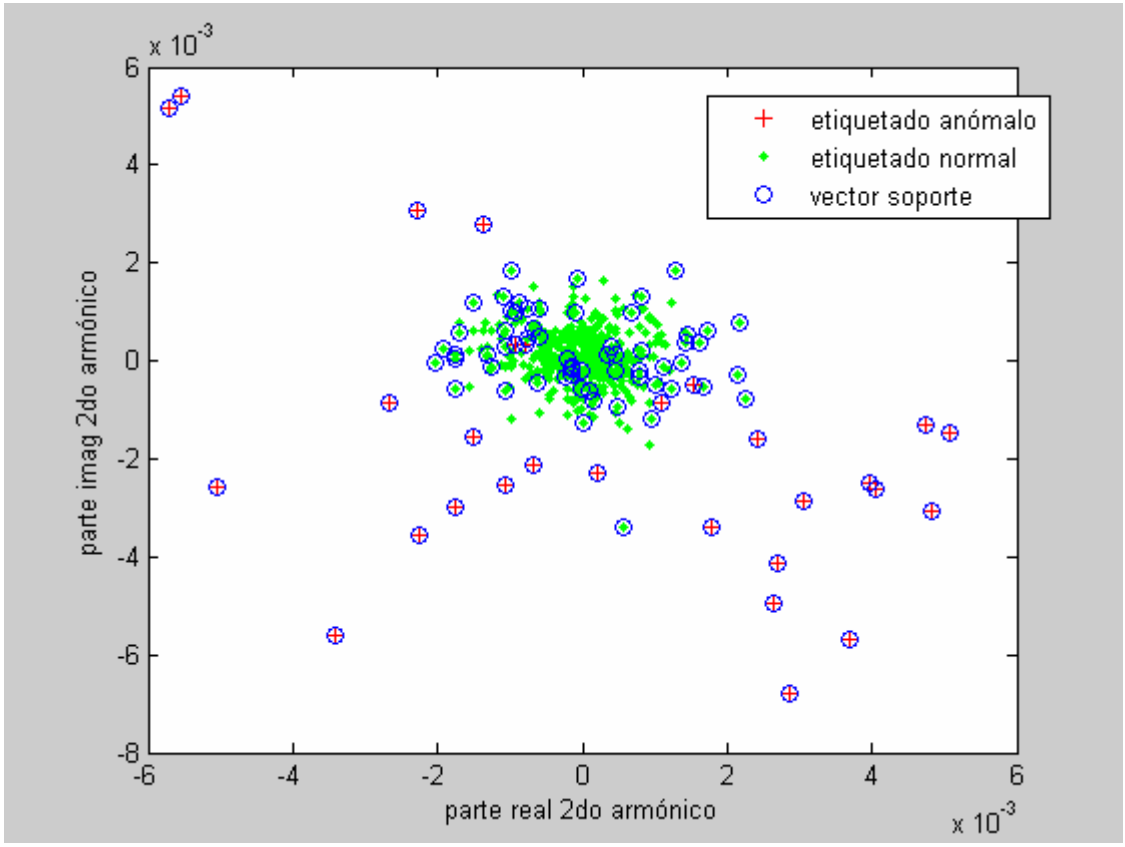
7. ANALISIS DE CARACTERÍSTICAS ELEGIDAS

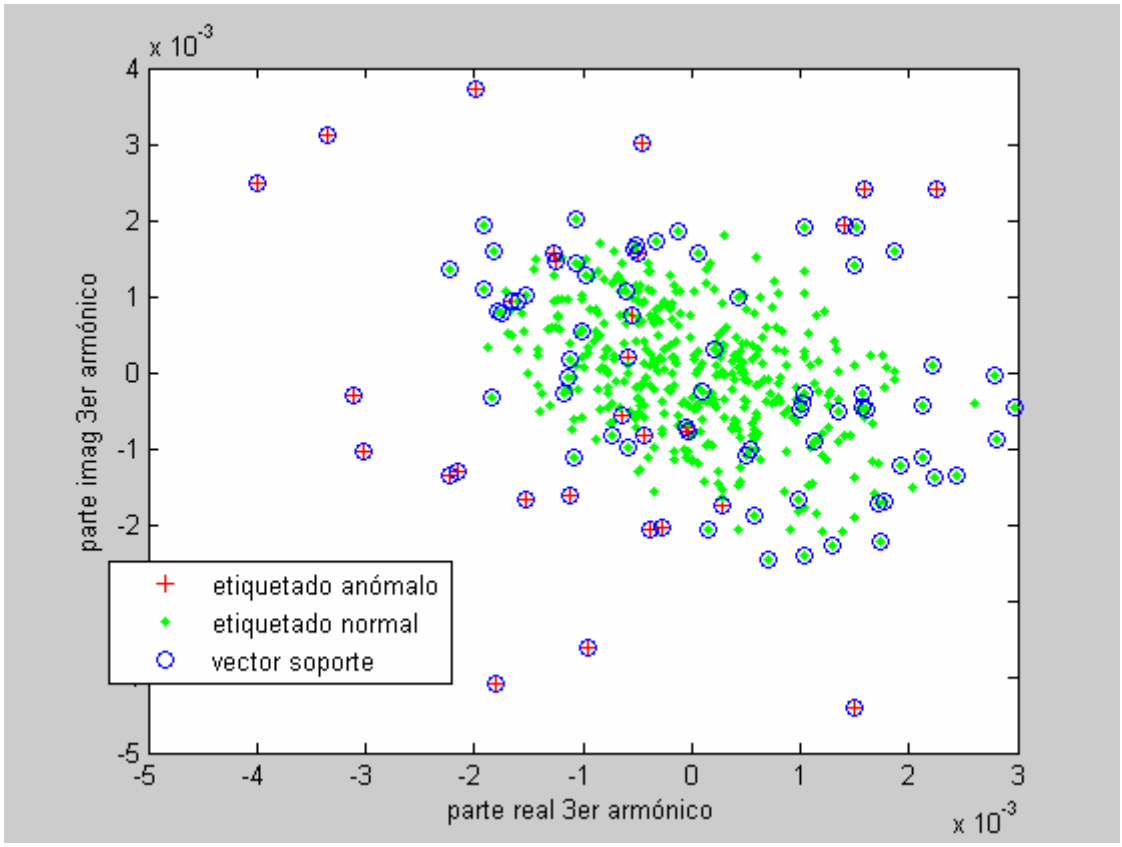
Se observa que las características elegidas son adecuadas para el objetivo de encontrar una región del espacio en donde se concentra la densidad de las muestras.

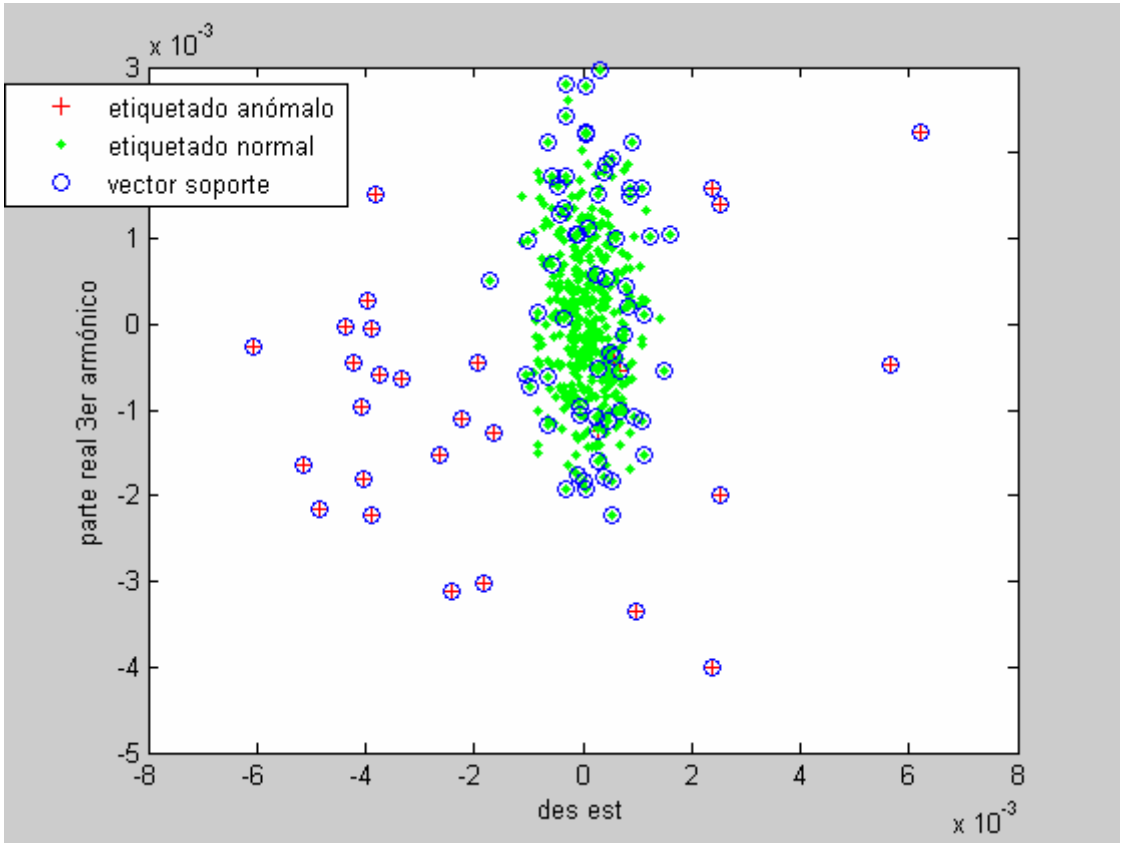
En el espacio de muestras no es posible encontrar funciones discriminantes lineales, lo cual sí fue posible al mapear los datos en el espacio de características mediante la función kernel.




x 10^-3
6
4
2
0
-2
-4
-6
-6
-4
-2
0
2
4
6
8
x 10^-3
parte imag 1er armónico
parte real 1er armónico
etiquetado anómalo
etiquetado normal
vector soporte


x 10^-3
6
4
2
0
-2
-4
-6
-8
-6
-4
-2
0
2
4
6
x 10^-3
parte imag 2do armónico
parte real 2do armónico
etiquetado anómalo
etiquetado normal
vector soporte


x 10^-3
parte imag 3er armónico
parte real 3er armónico
x 10^-3
etiquetado anómalo
etiquetado normal
vector soporte


x 10^-3
parte real 3er armónico
des est
x 10^-3
etiquetado anómalo
etiquetado normal
vector soporte

8. CONCLUSIONES

Se estudió y se aplicó el método de detección de anomalías utilizando SVM de una clase a un conjunto de suministros eléctricos obteniéndose muy buenos resultados, como ser, errores de clasificación del 0% para consumos anómalos y menores que el 13% para consumos normales.

Se utilizó con éxito un toolbox de MATLAB[4] para resolver este tipo de problemas, el cual es mucho más eficiente en tiempo de ejecución que la rutina que MATLAB tiene implementada para dichos propósitos.

Se da un paso en la automatización del análisis de consumos para detectar suministros sospechosos, logrando obtener grupos mas reducidos de suministros para analizar manualmente, con mayores concentraciones de consumos anómalos.

9. TRABAJOS FUTUROS

Es necesario analizar la posibilidad de implementar esta técnica para el análisis de suministros residenciales
En este segmento UTE tiene 450.000 suministros, por lo que la utilidad de una herramienta de este tipo queda claramente evidenciada por el hecho de que el análisis visual por parte de un gestor humano es totalmente imposible

Para la implementación en el segmento de los residenciales será necesario trabajar en la búsqueda de las características necesarias para que el algoritmo tenga un desempeño adecuado

[4] Este toolbox fue proporcionado por los docentes.