

LSSVM APPLIED TO ESTIMATE THE ROTOR RESISTANCE OF INDUCTION MOTOR SQUIRREL CAGE

LSSVM APLICADA EN LA ESTIMACIÓN DE LA RESISTENCIA DE ROTOR EN MOTOR DE INDUCCIÓN JAULA DE ARDILLA

MSc(c). Pablo Santafé Gutiérrez, MSc. Jorge Luis Díaz R., PhD. Oscar Gualdrón G.

Universidad de Pamplona, Facultad de Ingenierías y Arquitectura,
Km 1., Vía a Bucaramanga, Pamplona, Norte de Santander, Colombia.
Tel.: +(57) (7) 568 5303, Ext. 164.
E-mail: {pablosantafe, jdiazcu, oscar.gualdrón}@unipamplona.edu.co.

Abstract: This paper deal with the estimation the rotor resistance present in the dynamics of an induction motor squirrel cage using LSSVM in decline. The problem is that the rotor resistance is difficult to measure what is required for estimating it. In addition to being affected by the increase in temperature, leading to the rotor time constant of the change and affect the engine parameters. We propose a model of the engine in Matlab Simulink, which is extracted the data, is preprocessed and applied the algorithm LSSVM in nonlinear regression with RBF kernel and gamma optimization is done by cross-validation. Performance measurement model used the square root of the average prediction error. Getting good results.

Keywords: Induction motor, vector control, rotor resistance, LSSVM, regression.

Resumen: En este artículo se estima la resistencia de rotor presente en la dinámica de un motor de inducción Jaula de Ardilla aplicando LSSVM en regresión. El problema es que la resistencia de rotor es difícil de medir por lo que se requiere estimarla. Además, de ser afectada por el incremento en la temperatura, conllevando a que la constante de tiempo del rotor cambie y afecte los parámetros del motor. Se plantea un modelo del motor, en Simulink de Matlab, del cual se extrae la data, se preprocesa y se aplica el algoritmo LSSVM en regresión con Kernel no lineal RBF y la optimización de gamma se hace por validación cruzada. La medición del desempeño del modelo se utilizó la raíz cuadrada del error medio de predicción. Obteniendo buenos resultados.

Palabras clave: Motor de inducción, control vectorial, resistencia de rotor, LSSVM, regresión.

1. INTRODUCCIÓN

Los motores de inducción Jaula de Ardilla (SCIM) (*Squirrel Cage Induction Motor*) son máquinas eléctricas simples, resistentes, diseñadas para potencias cuya eficiencia, en comparación con otras máquinas rotatorias, es relativamente alta y requiere poco mantenimiento (Aller, 2006).

Sin embargo, su control es complejo, ya que la relación par contra flujo y par contra velocidad son no lineales. Se han propuesto y adaptado técnicas para solucionar los problemas de control, como: Control por Campo Orientado FOC (*Field Oriented Control*) idea presentada por Hasse (1969) y Blaschke (1972), y tratada en este trabajo (Blaschke, 1972).

El parámetro que experimenta mayores cambios es la resistencia del rotor (R_r), esta es afectada con el incremento de la temperatura, la cual no puede ser medida directamente por lo que es necesario estimarla para garantizar la correcta alineación del sistema de referencia con el vector de flujo del rotor (Villasana, 2006), de lo contrario, se producirá un error en la referencia del par, lo que incrementa las pérdidas por calentamiento y disminuye la eficiencia de la máquina.

Las Máquinas de Soporte Vectorial (SVM) son algoritmos de aprendizaje supervisado, se usa sobre un conjunto de ejemplos para entrenamiento, tiene solución única; es robusto a ruido, múltiples dimensiones y redundancia en las dimensiones; usa modelos lineales para implementar clases con separaciones no lineales al transformar el espacio de entrada en un nuevo espacio (Rubio, 2010). Las Máquinas de Soporte Vectorial por Mínimos Cuadrados (*Least Squares Support Vector Machine* o LSSVM) están relacionadas con las redes de regularización y los procesos Gaussianos (Álvarez *et al.*, 2007), aunque con un enfoque mucho más basado en la teoría de optimización, obteniendo un planteamiento más práctico. Las LSSVM en regresión surgieron como alternativa a las máquinas de soporte vectorial en regresión SVR (Basak *et al.*, 2007), con algunas ventajas como resolución más sencilla, ya que desaparece el parámetro γ y se disminuyen a la mitad el número de multiplicadores de Lagrange solucionando los problemas de entrenamiento, eficiencia, y tamaño de los modelos (Rubio *et al.*, 2009).

2. MODELO MATEMÁTICO DEL MOTOR DE INDUCCIÓN JAULA DE ARDILLA

2.1 Resistencia de rotor

La dependencia de los parámetros del motor con la temperatura influye en el detrimento de su comportamiento dinámico, siendo la resistencia del rotor (R_r) el parámetro más afectado, con la desventaja de no poderla medir directamente.

Además, es responsable de la variación de la constante de tiempo del rotor ($\tau_r = L_r/R_r$) afectando la dinámica de control y evitando la correcta predicción de flujo durante el transitorio, como consecuencia no se puede conocer la posición del flujo de rotor correctamente, el cual es usado para ubicar los ejes dq que rotan sincrónicamente. Se pierde el desacoplamiento de las componentes de flujo y par de la máquina (Vas, 1990), (Díaz, 2000) y (Sixto, 2004).

La resistencia de rotor a temperatura ambiente (20°C) es de $0.4\ \Omega$, y al entrar en operación el motor de inducción, puede alcanzar 155°C de temperatura, al ser un alambre de cobre el coeficiente térmico es de $\alpha = 0.00393^\circ\text{C}^{-1}$, Resultando al final una resistencia de $0.6122\ \Omega$.

$$R = 0.4 \left[1 + 0.00393(155^\circ - 20^\circ) \right] = 0.6122\ \Omega$$

2.2 Control indirecto

Este método no necesita medir o estimar el flujo, simplificando con esto el controlador y reduciendo los costos. Sin embargo el cálculo de la frecuencia del deslizamiento (w_{sl}) depende de la constante de tiempo del rotor (τ_r) la cual varía principalmente con la temperatura del rotor. Y si no se conoce su valor exacto los indicadores dinámicos del control por campo orientado se deterioran (Vas, 1990).

2.3 Ecuaciones del desacople

Las ecuaciones vectoriales del motor de inducción en un sistema de coordenadas que gira a velocidad del flujo del rotor, están dadas por las ecuaciones:

$$\mathbf{st}_s \frac{d\vec{i}_s}{dt} + \vec{i}_s = \frac{\vec{V}_s}{R_s} - jw_{mr} \mathbf{st}_s \cdot \vec{i}_s - (1-s) \mathbf{t}_r \cdot \frac{d\vec{i}_{mr}}{dt} - jw_{mr} (1-s) \mathbf{t}_s \cdot \vec{i}_{mr} \quad (1)$$

$$\mathbf{t}_r \frac{d\vec{i}_{mr}}{dt} + \vec{i}_{mr} = \vec{i}_s - j(w_{mr} - w_r) \mathbf{t}_r \cdot \vec{i}_{mr} \quad (2)$$

La fig. 1 muestra el diagrama vectorial, donde se representan las componentes de las magnitudes y ángulos del flujo del rotor y la corriente del estator, en un sistema de ejes fijo en el estator $\alpha\beta$, y otro sistema de ejes dq que gira a la velocidad del flujo magnético del rotor w_{mr} (Campo Orientado).

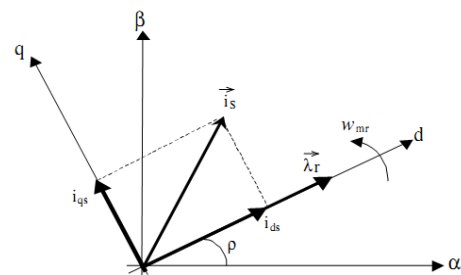


Fig. 1: Diagrama vectorial de los ejes dq .

Cuyas ecuaciones vienen dadas por (Vas, 1990):

$$V_{qs} = (R_s + L_s s) i_{qs} + w_{mr} (L_s + L_m^2/L_r) i_{ds} \quad (3)$$

$$V_{ds} = (R_s + L_s s) i_{ds} + w_{mr} L_s i_{qs} = R_s (\mathbf{st}_s s + 1) i_{qs} + w_{mr} L_s i_{ds} \quad (4)$$

Las anteriores ecuaciones representan el modelo del motor de inducción en campo orientado con la corriente como variable de entrada (Vas, 1990), (Díaz, 2000) y (Sixto, 2004).

3. MÁQUINA DE SOPORTE VECTORIAL POR MÍNIMOS CUADRADOS (LSSVM) EN REGRESIÓN

Las máquinas de soporte vectorial SVM (*Support Vector Machine*) son algoritmos de aprendizaje supervisado, se usa sobre un conjunto de ejemplos para entrenamiento, tiene solución única; es robusto a ruido, múltiples dimensiones y redundancia en las dimensiones; usa modelos lineales para implementar clases con separaciones no lineales al transformar el espacio de entrada en un nuevo espacio (Vapnik, 1963), (Vapnik, 1995).

Las LSSVM surgieron como alternativa a las SVR, solucionando algunos de sus problemas de eficiencia, de entrenamiento y de tamaño de los modelos. Las LSSVM presentan problemas, algunos generales a todos los métodos kernel y otros propios. Específicamente, el problema de optimización resultante tiene la mitad de parámetros que cuando se utilizan SVR, y el modelo se puede optimizar resolviendo un sistema de ecuaciones lineal en vez de un problema de optimización cuadrática. Las principales ventajas de las LSSVM para regresión con respecto a SVR son: la resolución más sencilla del problema (Wang, 2005), ya que desaparece el parámetro ϵ y se disminuyen a la mitad el número de multiplicadores de Lagrange. Además, soluciona problemas de eficiencia, de entrenamiento y de tamaño de los modelos. Los problemas comunes para SVM y LSSVM (Valdés, 2007) es que no existe ningún método para seleccionar la función *kernel* óptima para un problema concreto, en el estado del arte se usa de forma reiterativa el kernel Gaussiano o funciones de base radial RBF (Rojas *et al.*, 2000). La optimización de los parámetros es costosa computacionalmente dado que la dimensión del espacio de representación puede ser muy alta, y por último, los modelos generados pueden ser grandes (Jaramillo, 2007), (Huerta, 2005), (Maldonado, 2007), (Xu, 2005), (Seijas y Caralli, 2004).

Dado un conjunto de entrenamiento $\{x_i, y_i\}_{i=1}^n$ $x \in R^n$, $y \in R$. En el espacio de características el modelo dado por las máquinas de vectores de soporte toma la siguiente forma:

$$f(x) = \sum_{i=1}^n w_i f_i(x) + b \quad (5)$$

Donde el mapeo $f: X \rightarrow F$, mapea el espacio de entradas a algún espacio de características de mayor dimensión. En el método LSSVM se plantea el siguiente problema de minimización.

$$\min J(w, \mathbf{x}) = \frac{1}{2} \|w\|^2 + \frac{c}{2} \sum_{i=1}^l \mathbf{x}_i^2 \quad (6)$$

$$\text{Sujeto a: } y_i - \langle w \cdot \mathbf{f}(x_i) \rangle = \mathbf{x}_i \quad (7)$$

Dado el sistema de ecuaciones, el modelo utilizado por el método LSSVM se expresa así.

$$\hat{f}(x) = K(x, x_i) \mathbf{a} + b \quad (8)$$

Donde K representa al Kernel. Existen diferentes Kernels que según sea el caso se aplican: Kernel lineal, Kernel polinomial y kernel Gaussiano o función de base radial RBF, Kernel spline. La ventaja de los modelos SVR o LSSVM con kernel RBF son las precisiones obtenidas y el factor de regularización que adecuadamente ?jado garantiza buenas propiedades de generalización (Rubio *et al.*, 2008).

4. PROCESO EXPERIMENTAL

A continuación se detallan los pasos a realizar para obtener la estimación de la R_r .

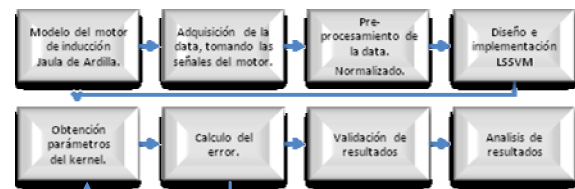


Fig. 2: Diagrama de bloques de la metodología de estimación de la R_r .

Los datos necesarios para entrenar LSSVM se obtienen de los resultados de las simulaciones del sistema formado por el motor y su control vectorial por método indirecto.

El modelo del motor de inducción Jaula de Ardilla controlado por voltaje, al cual se le hace una modificación en la R_r , con un valor inicial de 0.4 ? y empieza a variar de forma lineal hasta alcanzar un valor final de 0.6122 ? (Villasana, 2006), que equivale a una variación del 65.35 %. El modelo del motor se implementa en *Simulink*, consta de un bloque de control por voltaje; un bloque de motor, en el que se encuentra una submáscara donde se tiene el modelo del motor de CA en coordenadas de estator y las transformaciones de 3/2 y 2/3. Además, un botón para cagar los datos del motor, datos de parámetros del estator y rotor, datos de parámetros mutuos, datos de parámetros mecánicos, datos de referencia ganancia del deslizamiento, ganancia de adaptación, entre otros.

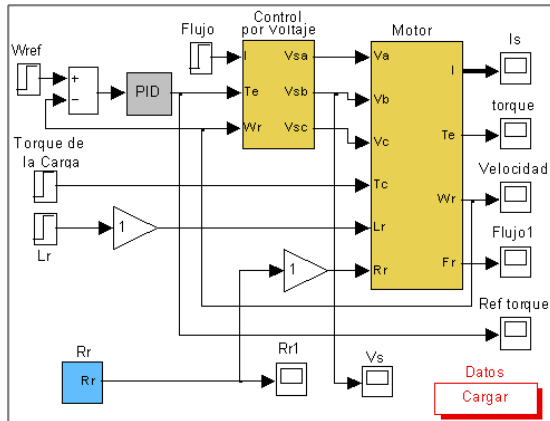


Fig. 3: Modelo del motor SCIM.

4.1 Data

Se obtiene al simular el modelo del motor llevando los datos como un arreglo (*array*) desde *Simulink* al *Workspace* de *Matlab*. Una vez decididos los parámetros que deben ser extraídos del modelo del SCIM se forma una matriz de datos de m filas (medida) por n columnas (variables).

Esta matriz de datos tiene cinco entradas, dos de corriente de estator, dos de voltaje de estator, y velocidad del rotor. El vector objetivo o vector de salida $[R_r]$ que se usará en la implementación de la LSSVM en regresión.

$$\text{Matriz de datos} = \begin{bmatrix} i_{qs} & i_{ds} & V_{qs} & V_{ds} & w \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ i_{qsi} & i_{dsi} & V_{qsi} & V_{dsi} & w_i \end{bmatrix}$$

$$\text{Vector objetivo } y = R_r^T = [R_{r1} \quad R_{r2} \quad \dots \quad R_{ri}]$$

Donde i es el i -enésimo elemento de la matriz de entrada y del vector objetivo.

El principal problema es que las señales de entrada y el vector de salida están con valores de amplitud diferentes, lo que conlleva a que existan algunos inconvenientes en la generalización del parámetro de regularización. La solución fue normalizar entre cero y uno la matriz de entrada y el vector objetivo.

Se aplica la normalización cuando los parámetros que describen cada experiencia son de naturaleza (y de unidades) diferentes que describen cada experiencia son de naturaleza (y de unidades) diferentes (Durán, 2005).

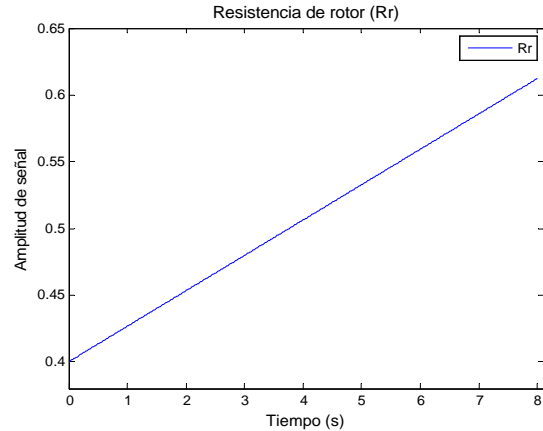


Fig. 4: Variación de R_r .

El aprendizaje en las LSSVM es supervisado, por lo que es necesaria una fase de entrenamiento para utilizarla posteriormente. La máquina debe aprender a mapear la relación entre el espacio de entrada y el de salida. Antes de aplicar la máquina se ha de tomar la data y dividirla, en tres conjuntos disjuntos para obtener los datos de entrenamiento, validación y prueba.

4.2 Normalización

El objetivo es dar la misma importancia numérica a todas las variables evitando problemas de saturación en la salida. En este método se obtiene el máximo y el mínimo de cada una de las variables, de manera que todas ellas tengan un valor comprendido entre 0 y 1, siendo este último su valor máximo normalizado. Los valores normalizados se almacenan en una matriz y en un vector de datos, esto con la ayuda de *Matlab*®. Para tal objetivo se usa la siguiente expresión:

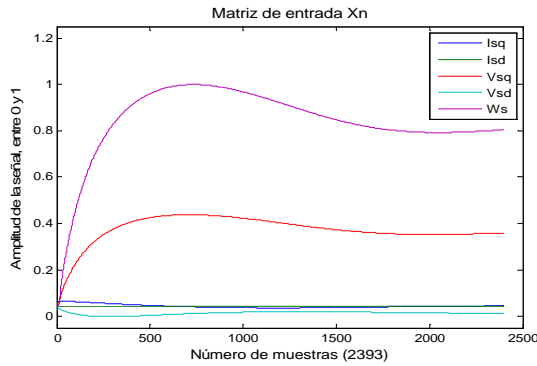
$$\tilde{X}_{ij} = \frac{X_{ij} - \min_j(X)}{\max_j(X) - \min_j(X)} \quad (9)$$

Donde X_{ij} es el valor original y $\max_j(X)$ y $\min_j(X)$ son el máximo y mínimo valor que toma el atributo j respectivamente.

4.3 Selección del Kernel.

Se selecciona el Kernel “Función de Base Radial Gaussiano” RBF (*Radial Basis Function Kernel*), que se fundamenta en las buenas propiedades y la respuesta presentada en el proceso de regresión al comparar el error RMSE con el del Kernel lineal y polinomial. Además, de ser el más implementado en la literatura.

$$K_s(X_i, X_j) = \exp\left(\frac{-\|X_i - X_j\|}{2s^2}\right) \quad (10)$$

Fig. 5: Variación de la R_r .

El parámetro a sintonizar es $\text{sig}2(\mathbf{s}^2)$ que controla la forma del hiperplano de separación, y este puede optimizarse a partir de métodos de validación cruzada (*crossvalidation*). (Rubio, 2009).

4.4 Ajuste de parámetros

Se deben obtener los dos parámetros de sintonía: coeficiente de dispersión $\text{gam}(\mathbf{g})$ y parámetro de la función Kernel RBF, varianza, $\text{sig}2(\mathbf{s}^2)$. En la *Toolbox* LSSVMLab se pueden ajustar de dos formas, la primera es dando valores a estos mediante el método de prueba y error, y la segunda aplicando la función de sintonía *tune*, que emplea el parámetro de sintonía *leave-one-out crossvalidation* (CV).

4.5 Proceso de regresión

Este proceso se realizó con la ayuda de la *Toolbox* LSSVMLab v1.8 de Matlab, la cual tiene las siguientes propiedades: optimización de la rutina usada para encontrar la función mínima de validación cruzada mejorando la velocidad; Intervalo simultáneos de predicción y confianza. Etapas que se debe implementar, en la LSSVM.



Fig. 6. Comandos para obtener el modelo LSSVM

4.6 Cálculo del error

La raíz del error cuadrático medio RMSE (*Root Mean Square Error*) se usa para medir el error en regresión, que se obtiene de promediar los residuos al cuadrado, buscando que el valor sea lo más pequeño posible. Se usa debido a la ponderación de los errores grandes (Berenson *et al.*, 2001; Fogarty *et al.*, 1999).

$$RMSE = \sqrt{\frac{\sum_{i=1}^n (Y_{ref} - Y_{est.})^2}{n}} \quad (11)$$

Donde: Y_{ref} valores de referencia y $Y_{est.}$ valores estimados, n número de muestras. En la etapa de validación RMSEV (*root mean square error of validation*) y en la etapa de prueba RMSEP (*root mean square error of probe*), tienen la misma ecuación, la diferencia está en el momento de aplicación.

5. RESULTADOS

Los resultados establecidos están en función de las entradas aplicadas a la R_r , en un primer momento una variación lineal y en el segundo caso una variación escalonada parabólica, que servirá para conocer la generalización del modelo planteado.

En este punto se plantean dos formas para llegar al modelo final, la primera es dando valores a los parámetros de sintonía (prueba y error) y el segundo con la función de sintonía *tunelssvm*, en ambos casos se busca el menor error RMSE.

5.1 División de la data (R_r) con variación lineal

El aprendizaje en las LSSVM es supervisado, por lo que es necesaria una fase de entrenamiento para utilizarla posteriormente. La máquina debe aprender a mapear la relación entre el espacio de entrada y el de salida. Antes de aplicar la máquina se toma la data y se divide en tres conjuntos disjuntos para obtener los datos de entrenamiento, validación y prueba.

Tabla 1. Distribución de datos

Data	Matriz	Vector objetivo	Datos	Porcentaje
Entrenamiento	X_e	Y_e	599	25%
Validación	X_v	Y_v	225	9.4%
Prueba	X_p	Y_p	1794	65.6%

5.2 Entrenamiento del modelo

Se toma la matriz de entrada X_e y el vector objetivo Y_e para entrenar el modelo funcional. Los cuales constan de 599 datos y un porcentaje del 25% de la data normalizada. Este conjunto de datos se uso en los diferentes modelos implementados.

Con la data de entrenamiento se debe seleccionar la interfaz funcional o la orientada a objetos, paso seguido se buscan los parámetros de sintonía con la función *tunelssvm* o prueba de error, luego se simula el modelo con la función *simlssvm* y por último se visualizan los resultados con la función *plotlssvm*¹.

¹ LS-SVM labToolboxUser's Guide versión 1.8, apartado 3.1 "roadmapto LSSVM", pág. 17.

La señal estimada Y_t se obtiene al entrenar la máquina con $X_t = X_e$ para ver el error que puede presentar. En las figuras se muestran la comparación de la señal Y_e , línea punteada y de color roja, que corresponde al vector de salida de entrenamiento y la señal Y_t , línea continua y de color azul, que es la señal estimada. Se evidencia que el error se hace pequeño, tiende a cero, y servirá como referencia para entrenar con nuevos valores.

5.3 Evaluación del modelo con el conjunto de validación

En este punto la señal estimada Y_t se obtiene haciendo que $X_t = X_v$, con los parámetros de sintonía variando según la tabla 2, y con la función Kernel RBF, y por ultimo estableciendo el error porcentual y el RMSE. Posteriormente, se gráfica para visualizar las señales de la R_r obtenidas.

Tabla 2. Resultados de entrenamiento y validación

PRUEBA	PARÁMETRO		ENTRENAMIENTO		VALIDACIÓN	
	gam	sig2	RMSE	ERROR	RMSE	ERROR
1	10	0.01	0.1096	8.4202	0.1095	8.4309
2	100	0.02	0.0232	1.2726	0.0229	1.2547
3	200	0.03	0.0119	0.6087	0.0117	0.6051
4	300	0.04	0.0078	0.3882	0.0083	0.3990
5	400	0.05	0.0058	0.2826	0.0070	0.3018
6	500	0.06	0.0047	0.2227	0.0065	0.2474
7	600	0.07	0.0039	0.1853	0.0063	0.2138
8	700	0.08	0.0034	0.1598	0.0062	0.1910
9	800	0.09	0.0031	0.1415	0.0062	0.1747
10	1000	0.1	0.0026	0.1182	0.0062	0.1538
11	300	0.01	0.0119	0.6434	0.0170	0.7182
12	500	0.01	0.0077	0.4072	0.0144	0.5050
13	10000	0.01	6.892e-4	0.0294	0.0121	0.1653
14	300	0.1	0.0061	0.2960	0.0070	0.3128

La figura 7 se muestra la señal de la R_r estimada y la señal de entrenamiento. Para la prueba 1, con los datos dados en la tabla 2.

En la figura 8 se visualiza el entrenamiento y prueba, respectivamente, de la máquina con la función de aproximación *tunelssvm*, la rutina de optimización *simplex* y la función de coste (*cost*) con validación cruzada mediante la función *leaveoneoutlssvm*.

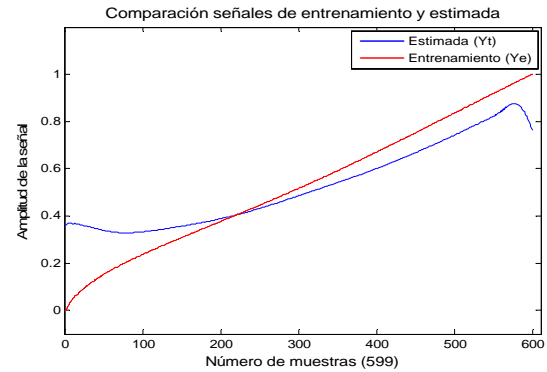


Fig. 7: Variación de R_r .

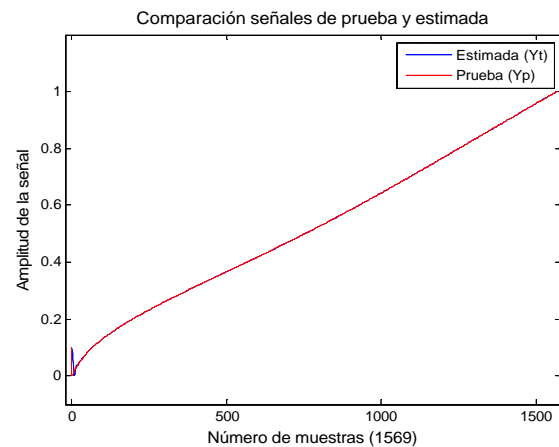


Fig. 8: Prueba del modelo de la R_r , con la función *tunelssvm* y *crossvalidation* optimizador de *sig2*.

5.4 Señal escalonada y parabólica, cambio de la forma de la R_r

La idea es cambiar la forma de la señal de la R_r en el modelo, de variación lineal a variación parabólica, y analizar los cambios que se presentaran al probar con la metodología planteada y la máquina anteriormente diseñada. Por lo cual se vario la R_r desde los límites establecidos para el caso de estudio anterior de 0.4 a 0.6122 Ω . El tiempo de muestreo va de un tiempo de corrida (*start time*) 0.0 hasta un tiempo de parada (*stop time*) de 10 segundos.

Tabla 3. Datos de entrenamiento, validación y prueba

Data de	Matriz	Vector objetivo	Datos	Porcentaje
Entrenamiento	X_e	Y_e	861	25%
Validación	X_v	Y_v	323	9.4%
Prueba	X_p	Y_p	2259	65.6%

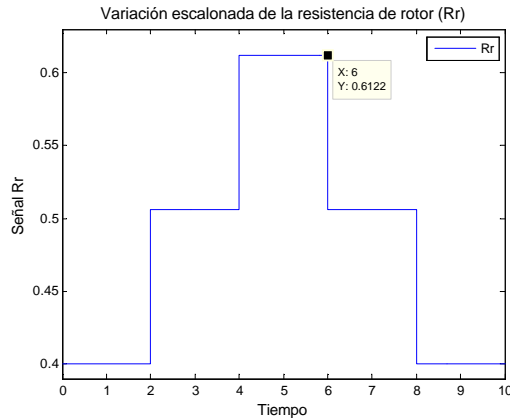


Fig. 9: Variación de la Resistencia de rotor R_r , de forma escalada y parabólica.

La simulación da como resultado un total de 3441 muestras, que serán las que se tomen para normalizar y hacer el análisis, y estimación de la R_r . La figura muestra la amplitud de la señal R_r , contra el número de muestras.

Tabla 4. Resultados de la simulación.

Parámetro	Entrenamiento	Validación	Prueba
RMSE	0.1351	0.1350	0.1339
error	6.9482	6.9370	8.4941
Tiempo de enlace (s)	0.154235	0.053821	0.309182

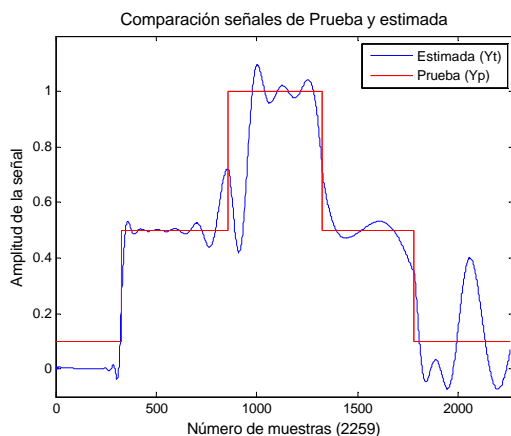


Fig. 10: Validación, $gam = 1000$ y $sig2 = 0.1$

Tabla 5. Resultados de la simulación. $sig2=1.7351$

Parámetro	Entrenamiento	Validación	Prueba
RMSE	0.0328	0.0244	0.0679
error	0.9658	0.8832	4.2836
Tiempo de enlace (s)	0.113311	0.055814	0.315347

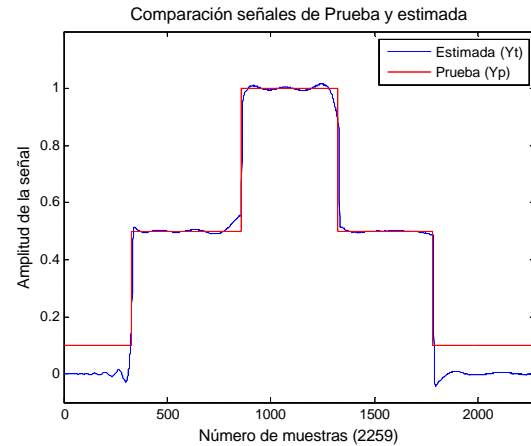


Fig. 11: Prueba, $gam = 1000$ y $sig2 = 0.1$

6. CONCLUSIONES

Se implementa el algoritmo *Support Vector Machines* por mínimos cuadrados (LSSVM) en regresión para la estimación de la resistencia de rotor en un motor de inducción Jaula de Ardilla, con variación lineal de la R_r , mediante la aplicación de varias técnicas que ofrece la *toolbox lssvmlab*, como son: interfaz funcional, interfaz orientada a objetos, por inferencia bayesiana y por intervalos de confianza y predicción, cuyo objetivo era obtener la estimación de los parámetros de sintonía ($gamma$ y $sigma2$) y establecer el modelo final.

Se documenta los métodos existentes aplicados en la estimación de la resistencia de rotor, entre los que se encuentra los algoritmos de adaptación, mínimos cuadrados recursivos (RLS), modelos de referencia, Observador de Luenberger extendido (ELO), Filtro de Kalman Extendido (EKF), máquinas de vectores de soporte vectorial (SVM) y red neuronal artificial (RNA). La mayoría de estos métodos fueron desarrollados en tesis de doctorado. El común denominador es la gran limitante en el coste computacional, por lo que aun se está investigando en la aplicación de algoritmos que requieran menor coste computacional, que sean fáciles de implementar y generalizar, ya que cada motor tiene sus características propias.

Se plantea el modelo matemático y la adquisición de la data del modelo implementado en *Simulink*® de Matlab. La data tomada constaba de cinco entradas y una salida que era la resistencia de rotor, la que se analizó para una señal linealmente variante y otra de variación parabólica escalonada, a esta última se le aplicó el modelo final obteniendo buenos resultados.

Los resultados fueron buenos tanto para la señal de variación lineal como para la señal de variación parabólica escalonada. Los parámetros optimizados con la interfaz orientada a objetos sirvieron para establecer el comportamiento ante la variación parabólica de la resistencia de rotor, la cual ofrece buenos resultados en comparación a la obtenida con la interfaz funcional. La limitante es el alto coste computacional para obtener el modelo, debido a que requiere de varias iteraciones y se recomienda aplicar en caso debidamente necesario.

Se concluye que entre mayor sea el coeficiente de dispersión gamma (γ) con respecto a la varianza sigma2 (σ^2) el error RMSE tiende a hacer más pequeño.

REFERENCIAS

- Aller, J. M. (2006). *Máquinas Eléctricas Rotativas: Introducción a la Teoría General*, Editorial Eequinoccio, Universidad Simón Bolívar, Caracas, Venezuela.
- Álvarez, M.; Henao, R. y Duque, E. (2007). *Clasificación de Eventos Sísmicos Empleando Procesos Gaussianos*. Scientia Et Technica, vol. XIII, núm. 35, agosto, 2007, pp. 145-150.
- Basak, D.; Pal, S. and Chandra Patranabis, D. (2007). "Support Vector Regression", Neural Information Processing – Letters and Reviews, Vol. 11, No. 10, India.
- Blaschke, F. (1972). "The principle of field orientation as applied to the new transvector closed loop control for rotating machines", Siemens Review, Vol 30, no. 5, pp. 217-220.
- Díaz, J. L. (2000). *Control por Campo Orientado del Motor de Inducción con adaptación de los parámetros por modelo de referencia*, Tesis de Maestría, UCLV, Santa Clara, Cuba.
- Durán Acevedo, C. M. (2005). *Diseño y optimización de los subsistemas de un sistema de olfato electrónico para aplicaciones agroalimentarias e industriales*, Universidad Rovira i Virgili, Cataluña, España, pp. 59-60.
- Huerta, P. F. y González, J. (2005). *Estimación de la Resistencia del Rotor en el Control Vectorial Indirecto del Motor de Inducción utilizando una Red Neuronal Artificial*, México, D. F.
- Jaramillo Garzón, J. A. (2007). *Metodología de optimización de los parámetros de control de un algoritmo genético*, Manizales Colombia.
- Maldonado Alarcón, S. (2007). *Utilización de Support Vector Machines no Lineal y Selección de Atributos para Credit Scoring*, Chile.
- Resendiz Trejo, J. A. (2006). *Las máquinas de vectores de soporte para identificación en línea*, México, D.F., Sep.
- Rojas, I.; Pomares, H.; González, J.; Bernier, J.; Ros, E.; Pelayo, F. y Prieto, A. (2000). "Analysis of the functional block involved in the design of radial basis function networks". Neural Processing Letters, 12:1–17.
- Rubio Flores, G. (2010). *Modelos Avanzados de Inteligencia Computacional para Aproximación Funcional y Predicción de Series Temporales en Arquitecturas Paralelas*, España.
- Rubio, G.; Guillen, A.; Herrera, L. J.; Pomares, H. y Rojas, I. (2008). "Use of specific to problem kernel functions for time series modeling". In ESTSP'08: Proceedings of the European Symposium on Time Series Prediction, pp. 177–186.
- Rubio, G.; Pomares, H.; Rojas, I.; Herrera, L. J., y Guillén, A. (2009). "Efficient optimization of the parameters of LSSVM for regression versus cross-validation error". ICANN, pp. 406–415.
- Seijas Fossi, C. y Caralli D'Ambrosio, A. (2004). "Uso de las Máquinas de Vectores de Soporte para la Estimación del Potencial de Acción Celular", Ingeniería UC, abril, vol. 11, número 001, Universidad de Carabobo, Valencia, Venezuela, pp. 56-61, 2004
- Sixto Berrocal, J. (2004). *Control Vectorial del Motor de Inducción en Bajas Velocidades*, Instituto Politécnico Nacional, México, D.F.
- Valdés, G. D. (2007). *Prógnosis Del Precio Del Oro Mediante Least Square Support Vector Machine (LSSVM)*, Univ. de las Américas.
- Vapnik, V. (1995). *The Nature of Statistical Learning Theory*, Springer-Verlag, New York.
- Vapnik, V. and Lerner, A. (1963). "Pattern recognition using generalized portrait method", Automation and Remote Control, 24.
- Vas, P. (1990). *Vector control of AC machines*: Clarendon Press Oxford.
- Villasana, S.; Caralli, A.; Seijas, C.; Villanueva, C.; Saenz L. y Arteaga, F. (2006). "Un novedoso método para estimar la resistencia rotórica del motor de inducción usando máquinas de vectores de soporte", Rev. Ingeniería UC. Vol. 13, No. 1, Venezuela.
- Wang, H. and Hu, D. (2005). *Comparison of SVM and LSSVM for regression*, Shanghai, Jiao Tong University, IEEE Press.
- Xu, R. R., Bian, G. X and Chen T. L. (2005). "Discussion about nonlinear time series prediction using least squares support vector machine". Communications in Theoretical Physics, 43:1056–1060.