

autoajusta durante la optimización; demuestran no solo que de los clasificadores implementados, la SVM no es únicamente una buena opción, sino la mejor.

Aunque, el clasificador merece mucho crédito, la solidez del preproceso y de la extracción de características, suponen un mejor desempeño, siempre que se obtengan resultados para el espacio de características, lo más discriminantes e invariantes posibles.

Por último, hay que tener en cuenta que no se tuvo acceso a todo el sistema de clasificación conforme a lo descrito en la implementación, por lo menos, en cuanto a lo que se refiere a la etapas de adquisición y segmentación, lo cual sugiere aspectos que a largo plazo mejorarían el desempeño del clasificador. Por diferentes motivos a veces se presentan interferencias entre canales que influyen bastante. El segmentador, aunque represente espacio extra en medios físicos, debería guardar un número constante de canales para una posible mejora en la extracción de características. El aprovechamiento de la no señal, en algunos canales sugeriría el diseño de reglas ya definidas, por lo menos en la clasificación manual, que definitivamente aportarían bastante a las cifras finales. Finalmente, lo que puede ser encontrado en artículos en cuanto a sismógrafos de 3 componentes deja ver un muy amplio abanico de posibilidades para conseguir incrementar el desempeño del sistema.

VII. AGRADECIMIENTOS

Los autores desean agradecer a Ingeominas seccional Manizales por toda la ayuda prestada para la realización de este artículo

VIII. BIBLIOGRAFÍA

- [1]. Richard O. Duda, Peter E. Hart, David G. Stork; Pattern Classification; Wiley-Interscience Publication; Second Edition.
- [2]. Samuel D. Stearns, Don R. Hush; Digital Signal Analysis; Prentice-Hall International; Second Edition.
- [3]. Chi Tsong Chen; One Dimensional Digital Signal Analysis; Marcel Dekker; 1979.
- [4]. Sophocles J. Orfanidis; Optimum Signal Processing: An Introduction; McGraw Hill Book Company; Second Edition; 1988.
- [5]. C. Valens; A Really Friendly Guide to Wavelets; perso.wanadoo.fr/polyvalens/clemens/download; 1999.
- [6]. Rehmi Post; A Gentle Introduction to Wavelets; University of Massachusetts, Amherst; web.media.mit.edu/~rehmi/wavelet/wavelet.html; 1995.
- [7]. Robi Polikar; Wavelet Tutorial; Iowa State University; www.public.iastate.edu/~rpolikar; 1999.
- [8]. Wendy L. Martinez, Angel R. Martinez; Computational Statistics Handbook with MatLab; Chapman & Hall/CRC; 2002.
- [9]. S. Lawrence Marple Jr.; Digital Spectral Analysis with applications; Prentice-Hall; 1987
- [10]. Simon Haykin; Neural Networks, A Comprehensive Foundation; Prentice-Hall; Second Edition; 1999.
- [11]. Martin T. Hagan, Mohammad B. Menhaj; Training Feedforward Networks with the Marquardt Algorithm; IEEE Transactions on Neural Networks; Vol. 5, No. 6; 1994.
- [12]. A.P. Paplinski; Advanced Learning Algorithms for Multilayer Perceptrons; NNets L-7; 1999.
- [13]. Christopher J.C. Burges; A tutorial on Support Vector Machines for Pattern Recognition; Data Mining and Knowledge Discovery; www.kernel-machines.org/papers; 1998.
- [14]. S.S. Keerthi, S.K. Shevade, C. Bhattacharyya, K.R.K. Murthy; A Fast Iterative Nearest Point Algorithm for Support Vector Machine Classifier Design; Intelligent Systems Lab, Dept. of Computer Science and Automation, Indian Institute of Science; guppy.mpe.nus.edu.sg/~mpessk; 1999.
- [15]. Olivier Bousquet, Sayan Mukherjee; Choosing Multiple Parameters for Support Vector Machines; AT&T Research Labs; www.connex.lip6.fr/~chapelle; 2001.



- [16]. S.S. Keerthi; Efficient Tuning of SVM Hyperparameters Using Radius/Margin Bound and Iterative Algorithms; National University of Singapore; guppy.mpe.nus.edu.sg/~mpesk/npam; 2001.
- [17]. Jhon C. Platt; Sequential Minimal Optimization: A Fast Algorithm for Training Support Vector Machines; Microsoft Research; www.research.microsoft.com/~jplatt; 1998.
- [18]. Bernhard Schölkopf, Alex Smola; Learning with Kernels: Support Vector Machines, Regularization, Optimization and Beyond; MIT Press; 2002.
- [19]. Signal Processing Toolbox, For Use with MatLab; The MatWorks Inc.; User's Guide, Version 4; Second Edition; 1998.
- [20]. Wavelet Toolbox, For Use with MatLab; The MatWorks Inc.; User's Guide, Version 2; Second Edition; 2000.
- [21]. Neural Network Toolbox, For Use with MatLab; The MatWorks Inc.; User's Guide, Version 3; Fifth Edition; 1998.
- [22]. William H. Press, Saul A. Teukolsky, William T. Vetterling, Brian P. Flannery, Michael Metcalf; Numerical Recipes in Fortran 90, The Art of Parallel Scientific Computing; Cambridge University Press; Second Edition; 1997.

IX. APÉNDICES

A. Simetría de la función de autocorrelación

Teniendo en cuenta las características de la formulación de la función de autocorrelación, se define una medida de simetría que, en términos generales, muestra que tan parecidas son las secuencias comparando los retardos positivos y negativos.

$$\hat{r}_+[x] = \frac{1}{N} \sum_{n=0}^{N-m-1} r[n+m]x^*[n] \text{ para } 0 \leq m \leq N-1$$

$$\hat{r}_-[x] = \frac{1}{N} \sum_{n=0}^{N-|m|-1} r[n+|m|]x^*[n] \text{ para } -(N-1) \leq m \leq 0$$

Donde $\hat{r}_+[x]$ y $\hat{r}_-[x]$ son los retardos positivos y negativos, respectivamente, de la función de autocorrelación de la señal x . Sin tener en cuenta el retardo en 0, que es el mismo para ambas secuencias, cada una de longitud $N-1$. Por lo tanto, si cada secuencia se toma desde $1, \dots, N-1$, se obtiene lo siguiente:

$$S\hat{r}[x] = \hat{r}_+[x](n) - \hat{r}_-[x](N-n)$$

De manera, que $S\hat{r}[x]$ es la medida de simetría de la función de autocorrelación de la señal $x \quad \forall n=1, \dots, N-1$.

B. Medida del error.

Se van a obtener dos medidas del error, una sobre el grupo de validación como es usual y otra sobre las observaciones de la clase que se está discriminando, de la siguiente manera:

N_v Número total de vectores de validación. Todos los del tipo A y todos los del tipo B.

N_c Número total de vectores pertenecientes a una clase determinada. Solo los del tipo A.

N_{mc} Número de malas clasificaciones en las cuales el vector observado pertenece a la clase que se está tratando. Es decir, para el tipo A, el clasificador etiqueta una muestra de tipo A como B.

N_{nmc} Número de malas clasificaciones en las cuales el vector observado no pertenece a la clase que se está tratando. De la misma forma, el clasificador etiqueta la muestra B como A.

$e_{\%}$ Porcentaje de error total.



$e_{\%c}$ Porcentaje de error para cada clase.

$$e_{\%} = 100 \frac{N_{mc}}{N_c}$$
$$e_{\%c} = 100 \frac{N_{mc} + N_{nmc}}{N_v}$$

Esto se hace porque son muchas clases no muy proporcionadas, de manera que el valor del error total no es concluyente.