SELECCIÓN DE INSTANCIAS EN *BIG DATA*

Victor Martinez Santiago, Dr. Alejandro Rosales Pérez , Dr. Edgar Jiménez Peña
Centro de Investigación en Matemáticas, Unidad Monterrey

CIMAT

Introducción

"El objetivo de un método de selección de instancias es obtener un subconjunto $S \in T$ tal que S no contenga instancias superfluas y $Acc(S) \cong Acc(T)$ donde $Acc(X)$ es la exactitud de clasificación obtenida usando X como conjunto de entrenamiento" [1]"

Datos más fáciles de manipular

1. Mejorar tiempos de ejecución en clasificadores
2. Eliminar instancias ruidosas
3. Optimización en el almacenamiento de la información

FCNN

Algoritmo de Selección de instancias propuesto por Angulli [2]

Es probable que seleccione puntos cercanos al límite de decisión

El algoritmo finaliza cuando el conjunto T es clasificado correctamente por S.

Algoritmo 1 FCNN (Fast Condensed Nearest Neighbour)

Entrada: Conjunto de entrenamiento $T = \{(x_1, y_1), ..., (x_n, y_n)\}$
Salida: Conjunto consistente S de T

$S = \emptyset$
$S = centroides(T)$
Mientras $\Delta S \neq \emptyset$ **hacer:**
    $S = S \cup \Delta S$
    $\Delta S = \emptyset$
    Para cada $x \in S$ **hacer:**
        $\Delta S = \Delta S \cup \{rep(x, Voren(p, S, T))\}$
    Fin Para cada
Fin Mientras

Distancias

Una métrica debe cumplir con los siguientes criterios, donde $d(x, y)$ se refiere a la distancia entre dos objetos x e y

- $d(x, y) \geq 0$ No negativa
- $d(x, y) = d(y, x)$ Simetrica
- $d(x, z) \geq d(x, y) + d(y, z)$ Desigualdad del triangulo

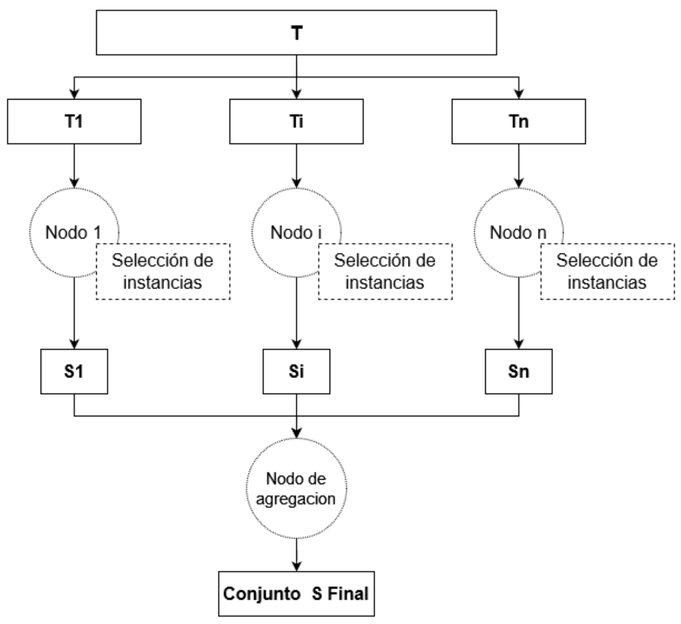
Distancia euclidiana

$$d(x, y) = \|x - y\| \tag{1}$$

Distancia en el espacio kernel

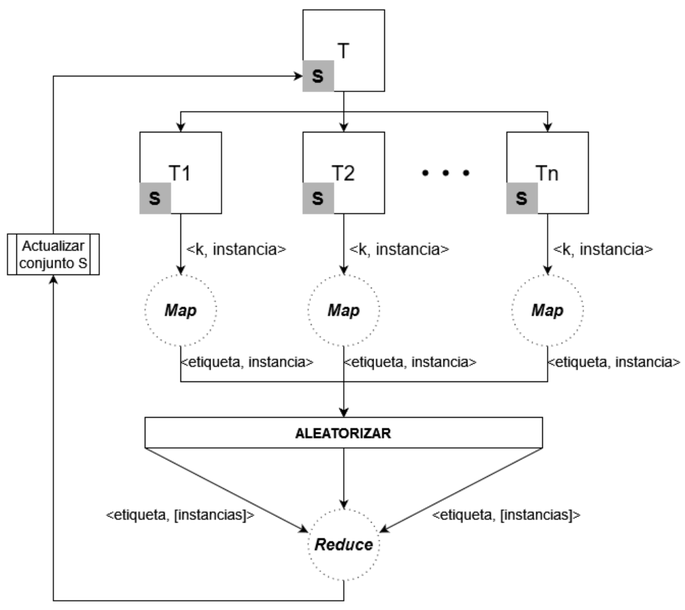
$$\begin{aligned} d^2(\phi(x), \phi(y)) &= \|\phi(x) - \phi(y)\|^2 \\ &= K(x, x) - 2K(x, y) + K(y, y) \end{aligned} \tag{2}$$

SI-MapReduce



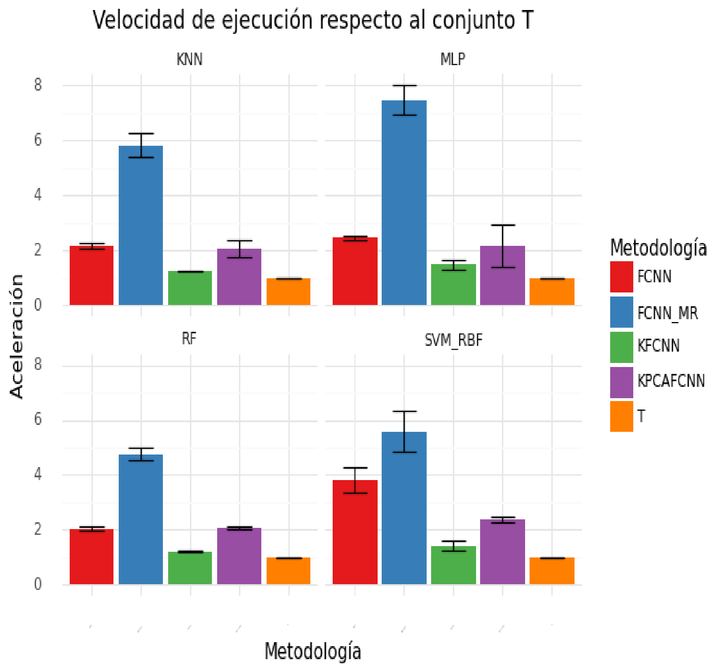
MRPR, Imagen Adaptada [3]

FCNN_MR



FCNN_MR, Imagen Adaptada [4]

Velocidad de Ejecución



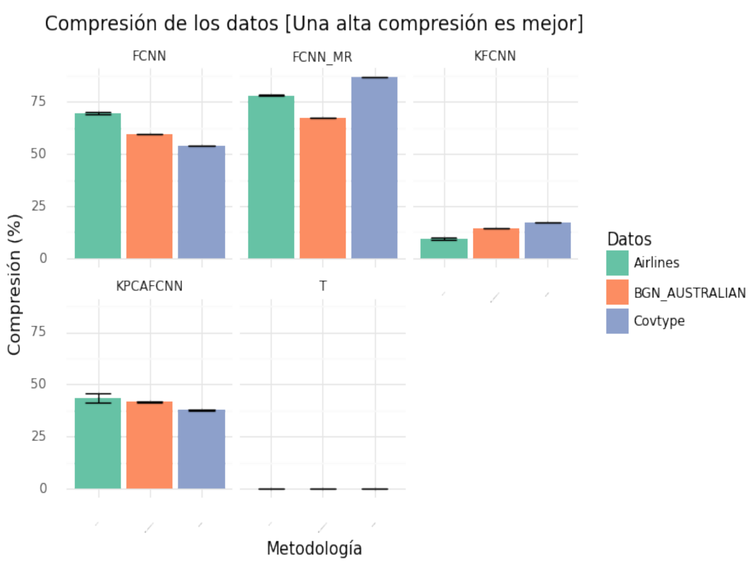
Desempeño del conjunto S

Tabla 1 : Evaluación de la Métrica F1-Macro para 3 conjuntos de datos utilizando 4 clasificadores

Datos	Metodología	KNN ($K = 3$)	MLP	SVM ($Kernel = RBF$)	RF
Airlines	FCNN_MR	0.88638	0.92891	0.93011	0.88130
	FCNN	0.88993	0.93087	0.93171	0.88390
	KFCNN	0.90571	0.93886	0.93410	0.89009
	KPCAFCNN	0.89291	0.92940	0.92012	0.88556
	T	**0.90737**	**0.93906**	**0.93478**	**0.89060**
BGN Australian	FCNN_MR	0.79384	0.85210	0.85558	0.86048
	FCNN	0.78106	0.85676	0.85871	0.86257
	KFCNN	**0.83234**	0.86448	0.85909	**0.86819**
	KPCAFCNN	0.82489	0.86412	0.85881	0.86243
	T	0.83203	**0.86480**	**0.85981**	0.85796
CovType	FCNN_MR	0.91101	0.85590	0.87814	0.76216
	FCNN	0.93067	0.88871	0.91909	0.76672
	KFCNN	0.94382	**0.90009**	0.92374	0.76814
	KPCAFCNN	0.93771	0.89333	0.92097	**0.76970**
	T	**0.94792**	0.89891	**0.92573**	0.76175
FraudChallengue*	FCNN_MR	0.71496	0.78282	0.62630	0.75245
	FCNN	0.68190	0.78277	0.64840	0.75634
	KFCNN	0.68199	0.78277	0.649703	0.75789
	KPCAFCNN	0.6856	**0.78347**	**0.68615**	**0.76578**
	T	**0.72698**	0.76954	0.55169	0.65220

*Datos desequilibrados

Compresión de los datos



Conclusiones

- La mayor compresión y aceleración se obtiene utilizando el algoritmo $FCNN\_MR$, y la perdida de rendimiento no supera las 5 centésimas de la línea base en nuestra métrica de interés.
- La metodología $FCNN$ muestra resultados ligeramente mejores a $FCNN\_MR$ en la métrica de interés, no obtiene la mayor compresión y en contraparte el tiempo de ejecución es menor al empleado por $FCNN\_MR$.
- $KFCNN$ muestra el mejor rendimiento, la velocidad de ejecución supera a $FCNN\_MR$ pero no logra una gran aceleración y la compresión mínima es de alrededor 10%.

Referencias

[1] J. Olvera-López, J. Carrasco-Ochoa, J. F. Martínez-Trinidad, and J. Kittler, "A review of instance selection methods," *Artif. Intell. Rev.*, vol. 34, pp. 133–143, Aug. 2010. DOI: 10.1007/s10462-010-9165-y.

[2] F. Angiulli, "Fast condensed nearest neighbor rule," in *Proceedings of the 22nd international conference on Machine learning*, 2005, pp. 25–32.

[3] I. Triguero, D. Peralta, J. Bacardit, S. García, and F. Herrera, "Mrpr: A mapreduce solution for prototype reduction in big data classification," *Neurocomputing*, vol. 150, pp. 331–345, 2015, ISSN: 0925-2312. DOI: https://doi.org/10.1016/j.neucom.2014.04.078.

[4] L. Si, J. Yu, S. Li, *et al.*, "Fcnn-mr: A parallel instance selection method based on fast condensed nearest neighbor rule," *Journal of information and communication convergence engineering*, vol. 11, pp. 855–861, 2017.