Selección de Instancias en *BigData*

Victor Martinez Santiago[a], Dr. Alejandro Rosales Pérez[b], Dr. Edgar Jiménez Peña[b]

[a] Alumno Maestría en Computo Estadístico, Centro de Investigación en Matemáticas
[b] Profesor Investigador, 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]

Nosotros abordaremos el problema pensando en la tarea de clasificación y la suposición de que trabajaremos con datos tabulados.

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

$$d(x, y) = \|x - y\|$$

Distancia en el espacio kernel

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

Reducción de Prototipos MapReduce

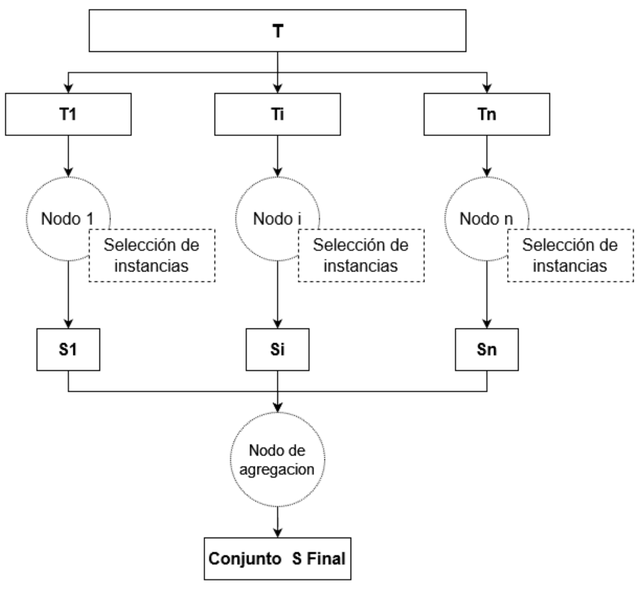


Figura adaptada de [3]

Motivación

Datos mas 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 limite 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 $T = \{(x_1, y_1), \ldots, (x_n, y_n)\}$
Salida: Conjunto consistente S de T

1: $S = \emptyset$; $S = centroides(T)$
2: **Mientras** $\Delta S \neq \emptyset$ **hacer:**
3: $S = S \cup \Delta S$
4: $\Delta S = \emptyset$
5: **Para cada** $x \in S$ **hacer:**
6: $\Delta S = \Delta S \cup \{rep(x, Voren(p, S, T))\}$
7: **Fin Para cada**
8: **Fin Mientras**

FCNN *MapReduce*

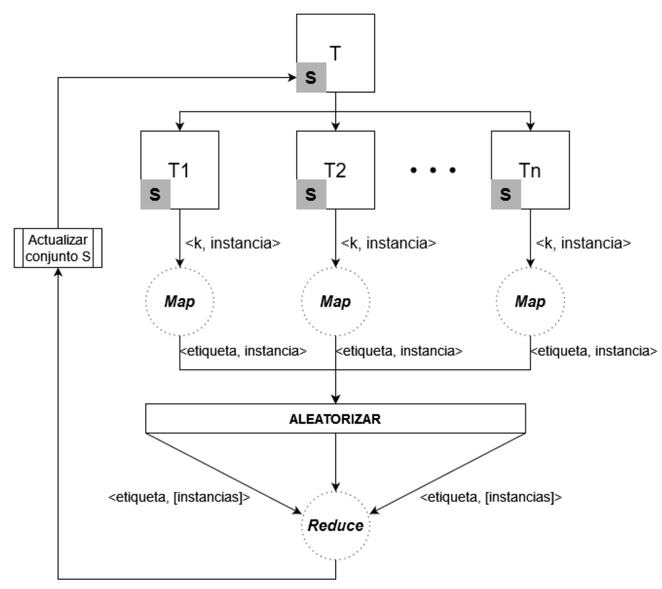
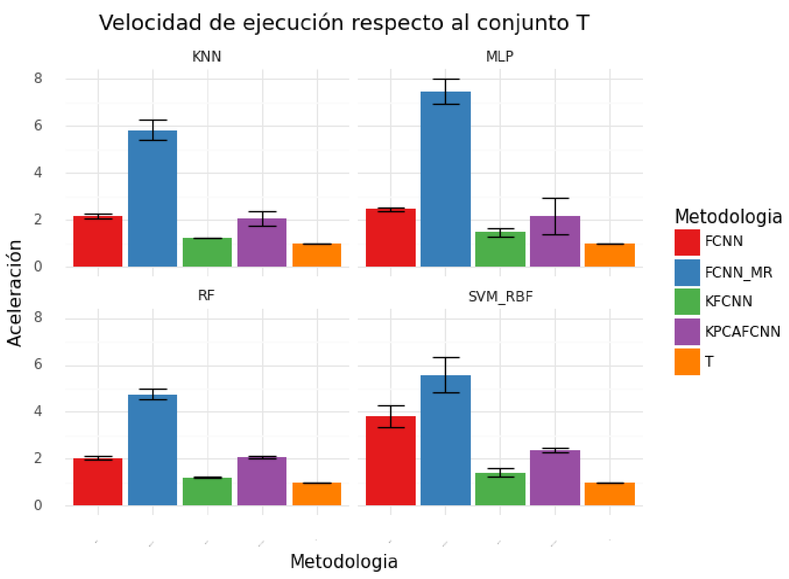


Figura adaptada de [4]

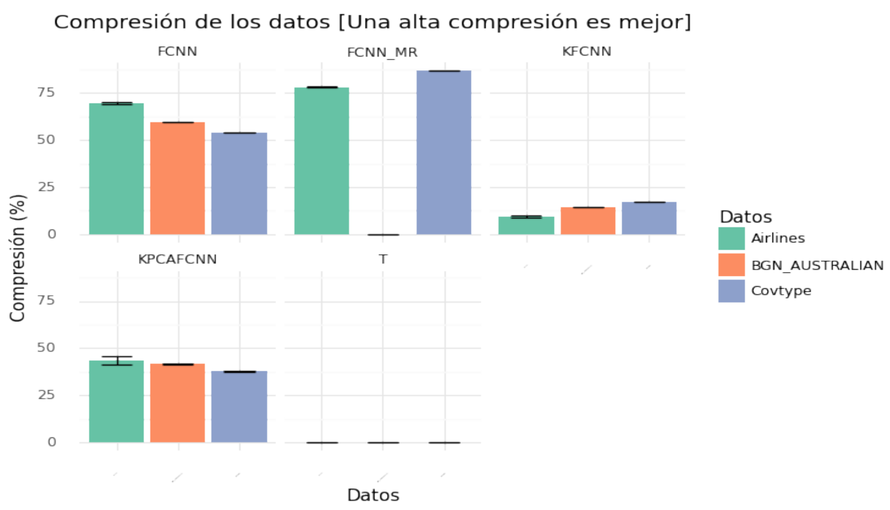
Velocidad de Ejecución



Rendimiento del conjunto S

Datos	Metodologia	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				
	FCNN	0.78106	0.85676		0.86257
	KFCNN	**0.83234**	**0.86448**		**0.86819**
	KPCAFCNN	0.82489	0.86412		0.86243
	T	0.83203	0.86480		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				
	T	**0.94792**	0.89891	**0.92573**	0.76175

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 linea 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 del 10%.
- ...

Bibliografia

[1] J. Olvera-López, J. Carrasco-Ochoa, J. F. Martínez-Trinidad u. a., "A review of instance selection methods," *Artif. Intell. Rev.*, Jg. 34, S. 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, S. 25–32.
[3] I. Triguero, D. Peralta, J. Bacardit u. a., "MRPR: A MapReduce solution for prototype reduction in big data classification," *Neurocomputing*, Jg. 150, S. 331–345, 2015, ISSN: 0925-2312.
[4] L. Si, J. Yu, S. Li u. a., "FCNN-MR: A Parallel Instance Selection Method Based on Fast Condensed Nearest Neighbor Rule," *Journal of information and communication convergence engineering*, Jg. 11, S. 855–861, 2017.