Similarity and distance for mixed data vs. data transformation in supervised classifiers: An empirical study.

Báez-Nieto Luis Á.[1][0000−0003−2388−2790]

Instituto Nacional de Astrofísica Óptica y Electrónica, Luis Enrique Erro 1, Sta María Tonanzintla, 72840 San Andrés Cholula, Pue. baez@inaoep.mx

Abstract. Muchas bases de datos contienen información mezclada; la cual, se requiere transformar para aplicar algún tipo de clasificador como máquinas de soporte vectorial (SVM), redes neuronales artificiales (ANN) o k vecinos cercanos (KNN). Dentro de las técnicas más utilizadas para la transformación de datos se encuentra la transformación categórica a númerica (*one hot encoding*) y transformación categórica a enteros.Para evitar la transformación de datos, se han explorado técnicas lógico combinatorias de patrones (Martinez Trinidad, 2001). AlVot (algorítmo de votación) es un ejemplo de estas técnicas.
En el presente trabajo se presenta un estudio empírico de clasificación supervisada usando diferentes clasificadores con diversas bases de datos, las cuales, en su mayoría contienen datos mezclados como atributos.

Keywords: Clasificación supervisada · Datos mezclados · AlVot.

1 *Introducción*

El tratamiento de atributos mezclados dentro de una base de datos es de vital importancia para clasificar nuevas instancias con rangos de error pequeños. Este hecho se presenta debido a que tener una transformación de algún atributo categórico a un numérico ordinal puede hacer que el modelo infiera cosas erroneas. Por ejemplo, si se tiene un atributo de color con tres posibles categorías, siendo estas rojo, azul y verde, al codificarlas a números enteros ordinales, se transforman ahora de rojo a uno, azul a dos y verde a tres; el modelo podría inferir que verde es mayor a rojo y azul. Esto, dependiendo de la problemática a resolver, podría no tener sentido alguno.

Otra forma para mitigar este problema es usando las diferentes técnicas de selección de características. Entre estas metodologías se encuentran técnicas estadísticas, como la correlación de Pearson o la prueba de hipótesis; basadas en modelos, como árboles de selección, modelos lineales o regularización; o mediante el uso de testores.

Aún hay otros enfoques que no se mencionan acá, amén de los métodos que se siguen desarrollando, como el trabajo de Duc Manh Doan [2], que propone una metodología nueva para la selección de atributos o Hesam Hasanpour [3], quien propone un modelo especializado para ello.

Este tipo de bases de datos se presentan en muchas áreas de estudio. Debido a esto, el uso de algoritmos especializados en datos mezclados como AlVot, pueden ser una buena opción para solventar esta problemática sin que se tenga que hacer una transformación previa a los datos. Tenga en cuenta que este trabajo no se enfoca en técnicas para datos perdidos o desbalance en las clases, aun así, son entre las bases de datos que se usaron en este estudio, se encuentran bases con estos problemas.

Los experimentos que se realizaron son los siguientes: con cada base de datos se clasificó mediante el modelo de AlVot, KNN y árboles de decisión. A su vez, cada modelo clasificó con la base de datos con la totalidad de atributos sin transformación alguna, otra sin atributos numéricos, otra más sin atributos categóricos, otra con atributos categóricos transformados y una más con todos los tipos de atributos con los categóricos transformados.

2 *Estado del arte*

En la literatura no se encuentra un estudio similar, sin embargo, sí se ha puntualizado este problema en algoritmos de aprendizaje no supervizado [4], donde la transformación de atributos impacta aún más la clasificación.

Es importante mencionar que en el estado del arte se encuentran variantes del modelo de AlVot, los cuales mejoran la precisión de la clasificación. Uno de ellos es el INC-AlVot [5], el cual usa las instancias nuevas como parte de su conocimiento, de esta forma, cada vez que clasifica, su precisión puede mejorar. El problema con este modelo propuesto es que es muy probable que las nuevas instancias no sean representativas, por lo que la mejora del modelo dependerá de estas instancias.

Otro trabajo importante que es pertinente mencionar, es el trabajo de Dalia Rodríguez-Salas y otros [6], donde proponen un modelo de votación por clase (AlVot BC). Este modelo se enfoca en construir sistemas de conjuntos de soporte por clase. De esta forma, cada conjunto de soporte provee información de pertenencia de una nueva instancia. La desventaja de este modelo se encuentra cuando se tiene una base de datos con un gran desbalance, ya que los votos de las clases minoritarias tendrán poco peso en la decisión final.

3 *Método*

En esta sección se presenta el algoritmo de votación (AlVot) que, como se mencionó, es un algorítmo especializado en datos mezclados. Tal especialización se debe a que usa métodos lógicos combinatorios para hacer la clasificación y no solo distancias. Además, usa el conocimiento de expertos para crear los pesos entre atributos e instancias de la base de datos.

Este algoritmo se compone de seis fases, las cuales, se explican a continuación. Antes de exponer estas fases, es conveniente introducir la siguiente notación.

- Ω, Sistema de conjunto de apoyo.
- n, Número de características para los objetos en la matriz de aprendizaje (LM)
- $R = \{x_1, x_2, ...x_n\}$, Conjunto de atributos que describen a los objetos
- $\Omega_k = \{x_{p_1}, x_{p_2}, ...x_{p_{s_k}}\}$, k-ésimo conjunto de apoyo donde $s_k \leq n$, $\Omega_k \subseteq R$ y $\Omega_k \in \Omega$
- X_j, Vector de descripción completa de el objeto O_j, de acuerdo a R
- X_j^k, Vector de descripción parcial de el objeto O_j, de acuerdo a R
- x_p^j, Valor de los atributos x_p in el objeto O_j
- m, Número de clases en LM
- $C = \{c_1, ..., c_m\}$, conjunto de etiquetas de clase
- y_j, Valor de la clase del objeto O_j, $y_j \in C$

Las fases se resumen como se describe a continuación

1. Sistema de conjuntos de apoyo. En esta primera fase se construye cada uno de los conjuntos de apoyo Ω_k. Estos conjuntos de apoyo deben poder discriminar cada objeto con su respectiva clase en LM. En una primera instancia, se usó el conjunto potencia con cardinal fijo 5. El uso de testores típicos es más recomendable o, en su defecto, el uso de cualquier otra selección de características o reducción de dimencionalidad.

2. Función de similaridad parcial. Esta función compara pares de objetos basados en los vectores de descripción parcial.

$$f(O_i, O_j) = \begin{cases} 1 & |O_i - O_j| \leq \mu \\ 0 & otherwise \end{cases} \tag{1}$$

Con $\mu = 0.23$

3. Función de evaluación parcial para conjuntos de soporte fijo al nivel de objeto. Cada conjunto de apoyo se usa para calcular las similaridades entre el objeto nuevo O_{new} y los objetos de la LM. Estos valores se consideran como los votos primarios. Cada voto se obtiene considerando solo las características de el

conjunto de soporte actual.

4. Función de evaluación parcial para un conjunto de soporte fijo al nivel de clase. La función suma los votos dados a un objeto nuevo O_{new} por los objetos de alguna clase, dado un conjunto de soporte particular.

$$\Gamma_{\Omega}^{j}(O_{new}) = \frac{1}{|n_j|} \sum_{t=1}^{n_j} \Gamma_k(X_t^k, X_{new}^k) \tag{2}$$

Donde n_j es el número de objetos en la clase c_j

5. Función de evaluación total para todos los conjuntos de apoyo al nivel de clase. Esta función suma los votos dados a un nuevo objeto O_{new} por los objetos de alguna clase, considerando todos los conjuntos de apoyo.

$$\Phi_j(O_{new}) = \frac{1}{|\Omega|} \sum_{\Omega_k \in \Omega} \Gamma_k^j(Onew) \tag{3}$$

6. Regla de decisión. Esta regla determina a qué clase pertenece O_{new}. En este caso, se tomó la regla por:

$$\Phi_i(O_{new}) > \Phi_j(O_{new}) \forall j = 1, ..., m, j \neq i \tag{4}$$

4 Experimentos y resultados

Se usó R como lenguaje de programación para la implementación del algoritmo de votación. De igual forma, se usaron las librerías de caret, rpart y class dentro del mismo lenguaje de programación para usar los clasificadores de k vecinos cercanos y árboles de decisión.

Las bases de datos que se usaron provienen de la UCI. Estas bases se han usado para probar nuevos modelos de clasificación y regresión. Para este estudio se usaron cinco bases de datos correspondientes a enfermedades del corazón (heart), tipos de animales en un zoologico (zoo), aprovación de seguro de viajes (trav), evaluación de enseñanza asistida (tae) y localización de sitios celulares de proteinas (prot).

El tratamiento de la base de datos fue diferente para cada una de ellas, esto, debido a la naturaleza de los datos que poseen. El preprocesamiento que todas las bases tienen en común es la transformación de atributos tipo string a numérico

Table 1. Información general de las bases de datos

DB	A. numéricos	A. categóricos	No. de instancias	No. de atributos	No. Clases
Heart	6	7	303	13	2
Zoo	1	16	101	17	7
Trav	6	4	63326	10	2
tae	1	4	151	5	3
Prot	7	1	1484	8	10

y la normalización de los atributos numéricos. Esta normalización está definida por la siguiente fórmula.

$$x_n = \frac{x_i - min(x)}{max(x) - min(x)} \tag{5}$$

Para la parte de las pruebas se tomó en cuenta que cada clasificador debía seleccionar los mismos datos de entrenamiento y prueba (el porcentaje de datos de entrenamiento es del 95%), los cuales, se deben tomar de forma aleatoria por cada validación cruzada de las diez que se hicieron en total por clasificación, por lo que, como primer paso, se estableció una semilla con valor de 31 en cada experimento.

Cuando se hace la clasificación con datos categóricos transformados se usó la codificación one hot por su amplio uso en algoritmos de aprendizaje máquina

Ningún clasificador debe tener ventaja sobre otro, por lo que se usaron los valores por defecto de cada clasificador. En el caso de k vecinos cercanos, se estableció la k en 15 vecinos para todas las clasificaciones; con los árboles de decición no se tenía nungún parametro de regularización, salvo la selección del modo de operación; en el caso de AlVot, como ya se describió en la sección 3, se estableció el cardinal fijo en 5 para la creación de los conjuntos de apoyo, se eliminaron los pesos que proveen información de los atributos e instancias a el clasificador, las funciones de similaridad fueron las mismas para cada experimento, el umbral para la función de similaridad parcial se fijó en 0.23. Este valor de umbral μ se eligíó experimentalmente con la primera base de datos, tomando en cuenta que los valores numéricos fueron normalizados a una escala de cero a uno.

Ya que cada base de datos es diferente, antes de comenzar con la clasificación mediante AlVot, se realizan unas tareas previas. En el caso de la base de enfermedades del corazón, se cambia el valor de las clases; de 0 y 1 a 1 y 2. Este cambio se hace para que el algoritmo pueda almacenar los cambios en la matriz de confusión sin cambiar el código.

Para el caso de la base de datos de animales de zoológico, primero se elimina la primera fila, ya que esta corresponde a un identificador único para cada objeto.En el caso de las otras tres bases se sigue un procedimiento similar al planteado con estas dos bases de datos.

En la tabla 2 se presentan los resultados de la clasificación con AlVot, en la tabla 3 la clasificación mediante k vecinos cercanos y en la tabla 4 la clasificación con árboles de decisión.

Table 2. Presición de la clasificación con AlVot. A: Todos los atributos sin transformar B: Todos los datos con los atributos categóricos transformados C: Solo atributos categoricos D: Solo atributos numéricos E: Solo atributos categóricos transformados

DB	A	B	C	D	E
Heart	83.66%	73.66%	86.66%	68%	84%
Zoo	86%	84%	86%	NA	86%
Trav	74.4%	77.41%	74.96%	64.3%	73.65%
tae	48.21%	45%	44.81%	NA	45.53%
Prot	0%	NA	NA	0%	NA

Los resultados de los experimentos de la tabla 2 muestran que el algoritmo de votación tiene mejores resultados sin que haya una transformación involucrada en los atributos cuando se toma un cardinal fijo para la construcción del sistema de conjunto de apoyo. Este resultado se explica debido a que cuando se hace la transformación de los atributos categóricos a una codificación one hot, el número de atributos incrementa y, en consecuencia, el sistema de conjunto de apoyo. Por lo que los votos que van aportando son más variados.

Si observa, en el caso de la base de datos Prot, el algoritmo no fue capaz de clasificar con las configuraciones ya mencionadas. Esta base de datos en específico se es difícil hacer una clasificación (vea los resultados para esta misma base con los otros dos modelos). Aún así, esto no significa que no se puedan clasificar nuevas instancias con este algoritmo, solo hay que adaptarlo.

Note también que para el caso de la clasificación D en la segunda base de datos se encuentra un no aplica. Ese resultado se debe a la naturaleza de la base de datos. Esta base consta de datos categóricos y un numérico, el cual, es ordinal, por lo que una clasificación con un solo atributo no es posible. Las otras clasificaciones con no aplica tienen el mismo problema.

En el caso de la clasificación con k vecinos cercanos (tabla 3) , se observa el dominio que tiene el algoritmo cuando clasifica solo con atributos numéricos. Esta precisión va bajando en cuanto se toman en cuenta los atributos categóricos. También se observa que cuando clasifica con solo atributos categóricos, la precisión es la menor, además, esta precisión puede disminuir aún más cuando se le

Table 3. Presición de la clasificación con kNN. A: Todos los atributos sin transformar B: Todos los datos con los atributos categóricos transformados C: Solo atributos categoricos D: Solo atributos numéricos E: Solo atributos categóricos transformados

DB	A	B	C	D	E
Heart	97.33%	91.33%	96%	100%	86%
Zoo	62%	70%	64%	NA	68%
Trav	98.17%	98.49%	97.63%	100%	86%
tae	53.75%	85%	50%	NA	82.5%
Prot	38.91%	44.32%	NA	44.32%	NA

aplica una codificación one hot. Esto depende da la base de datos

Aunque kNN ha mostrado una precisión bastante alta para clasificaciones numéricas, se observa que es posible obtener una buena precisión de clasificación cuando se tiene un número de atributos categóricos mayoritarios (como en el caso de tae). Esto se debe a que, aunque el mayor número de atributos son categóricos, los datos que contiene son ordinales, por lo que una clasificación mediante kNN es bastante natural para el algoritmo.

Table 4. Presición de la clasificación con árboles de decisión. A: Todos los atributos sin transformar B: Todos los datos con los atributos categóricos transformados C: Solo atributos categoricos D: Solo atributos numéricos E: Solo atributos categóricos transformados

DB	A	B	C	D	E
Heart	83.33%	79.33%	84.66%	70.66%	84.66%
Zoo	50%	54%	54%	NA	54%
Trav	89.46%	90.12%	74.19%	87.74%	76.66%
tae	45%	46.25%	58.75%	NA	46.25%
Prot	25.13%	25.13%	NA	37.16%	NA

Con los resultados que se obtuvieron al clasificar con árboles de decisión (tabla 4) se presenta una precisión mayor cuando los atributos son categóricos para la mayoría de las bases de datos. Esta precisión aumenta cuando los atributos son transformados mediante la codificación one hot.

Para los casos en los que no se cumple este comportamiento es para las bases que tienen mayor número de instancias, en este caso, trav y prot. Al tener más números de instancias en la base de datos se obtiene mayor conocimiento de las clases, por lo que se construye un árbol más robusto.

Además, en el caso de prot, la mayoría de atributos es numérico por lo que se entiende el nivel bajo de precisión. Es interesante observar que al agregar el atributo categórico a el modelo, la precisión baja. Esto también sucede con kNN.

Por lo que en específico, este atributo aporta información irrelevante y dañina al clasificador.

En las tablas 5, 6 y 7 se muestra una comparación entre AlVot y kNN; Alvot y árboles de decisión (tree); y entre los tres clasificadores.

Por el nivel de precisión que se alcanza por cada modelo, la comparación entre cada modelo se ve altamente afectada por los resultados de kNN; sin embargo, la especialidad de cada modelo aún puede observarse (tablas de la 5 a la 7).

Table 5. Comparación entre AlVot y kNN. A: Todos los atributos sin transformar B: Todos los datos con los atributos categóricos transformados C: Solo atributos categoricos D: Solo atributos numéricos E: Solo atributos categóricos transformados

DB	A	B	C	D	E
Heart	kNN	kNN	kNN	kNN	kNN
Zoo	AlVot	AlVot	AlVot	NA	AlVot
Trav	kNN	kNN	kNN	kNN	kNN
tae	kNN	kNN	kNN	NA	kNN
Prot	kNN	kNN	NA	kNN	NA

Table 6. Comparación entre AlVot y árboles de decisión. A: Todos los atributos sin transformar B: Todos los datos con los atributos categóricos transformados C: Solo atributos categoricos D: Solo atributos numéricos E: Solo atributos categóricos transformados

DB	A	B	C	D	E
Heart	AlVot	Tree	AlVot	Tree	Tree
Zoo	AlVot	AlVot	AlVot	NA	AlVot
Trav	Tree	Tree	AlVot	Tree	Tree
tae	AlVot	Tree	Tree	NA	Tree
Prot	Tree	Tree	NA	Tree	NA

Dentro de la tabla 5 se muestra la comparación entre AlVot y kNN. Aunque kNN es superior en casi todos los casos, esto se debe a que la función de similaridad no es la óptima para los atributos mezclados en el caso de heart y trav, los cuales tienen atributos balanceados. En el caso de tae, al tener tan pocos atributos, el sistema de conjuntos de apoyo es muy reducido.

En la tabla 6 se hace la comparación entre AlVot y los árboles de decisión. En esta tabla se muestra la superioridad que tiene AlVot ante datos mesclados.

Table 7. Comparación entre AlVot, kNN y árboles de decisión. A: Todos los atributos sin transformar B: Todos los datos con los atributos categóricos transformados C: Solo atributos categoricos D: Solo atributos numéricos E: Solo atributos categóricos transformados

DB	A	B	C	D	E
Heart	kNN	kNN	kNN	kNN	kNN
Zoo	AlVot	AlVot	AlVot	NA	AlVot
Trav	kNN	kNN	kNN	kNN	kNN
tae	kNN	kNN	Tree	NA	kNN
Prot	kNN	kNN	NA	kNN	NA

Incluso es mejor cuando se tiene solo numéricos. La precisión de los árboles de decisión solo es mayor cuando se hace una codificación one hot.

5 Conclusiones

Con este estudio se muestra el nivel de especialidad que tiene cada clasificador ante un determinado tipo de atributo. Por lo que, el elegir un clasificador u otro depende de la base de datos que se tenga.

Para ser más específicos, el uso de AlVot para clasificar bases de datos con datos mezclados es una de las mejores opciones, aun así, el generar el modelo requiere un conocimiento a priori de la base de datos, así como también se recomienda el conocimiento de un experto cuando sea posible. De esta forma, se pueden proponer las funciones de semejanza que den los mejores resultados, así como información para la relevancia de cada atributo e instancia en la base de datos. El tener un conjunto de apoyo robusto mejora bastante la precisión, aunque se corre el riesgo de sobre ajustar el modelo.

Se mostró que kNN tiene los mejores resultados ante atributos numéricos. Para el caso de árboles de decisión se observa que en general, no muestra una ventaja ante los otros clasificadores.

La selección de instancias representativas dentro de las bases de datos mejoran bastante la clasificación final del clasificador.

6 Trabajo futuro

Aunque se intentó hacer una clasificación con diferentes modelos con sus configuraciones predeterminadas para no generar ventaja de un modelo ante otro, se observó que AlVot tuvo más desventaja ante los demás clasificadores. Se llegó a esta conclusión debido a la función de similaridad (de lo cual ya se habló en los resultados) y al hecho de que el sistema de conjunto de apoyo formado por

un órden fijo en 7, subió en un 10% aproximadamente la presición de la clasificación, pero esto, genera un sistema de conjunto de apoyo aún mayor, por lo que el tiempo de procesamiento sube bastante.

Por estos hechos es que se propone hacer la comparativa con AlVot usando testores para la generación del sistema de conjuntos de apoyo y una función de similaridad distinta para cada base de datos.

También, para reducir el tiempo de ejecución en la clasificación (aunque con el uso de testores este tiempo se reduce bastante), se propone incorporar estrategias de divide y conquista en el algoritmo o, en su defecto, usar cómputo paralelo.

Para generar una visión más amplia de la comparativa se propone usar un número mayor de bases de datos y clasificadores, tales como máquinas de soporte vectorial y random forest.

References

1. Martinez-Trinidad, J. F., Guzmán-Arenas, A.: The logical combinatorial approach to pattern recognition, an overview through selected works. Pattern Recognition, **34**(4), 741–751 (2001)
2. D. M. Doan, D. H. Jeong and S. Ji, "Designing a Feature Selection Technique for Analyzing Mixed Data," 2020 10th Annual Computing and Communication Workshop and Conference (CCWC), Las Vegas, NV, USA, 2020, pp. 0046-0052.
3. H. Hasanpoura, R. G. Meibodia, K. Navia, S. Asadi.: Dealing with mixed data types in the obsessive-compulsive disorder using ensemble classification. Neurology, Psychiatry and Brain Research, **32**, 77–84 (2019)
4. A. Ahmad and S. S. Khan, "Survey of State-of-the-Art Mixed Data Clustering Algorithms," in IEEE Access, vol. 7, pp. 31883-31902, 2019.
5. Uriel. E. Franco, G. S. Díaz.: The incremental voting algorithm INC-ALVOT for supervised classification. Rev. Fac. Ing. Univ. Antioquia **50**, 195 – 204 (2009)
6. D. Rodríguez-Salas, M. S. Lazo-Cortés, R. A. Mollineda, J. A. Olvera-López, Jorge de la Calleja, and Antonio Benitez.: Voting Algorithms Model with a Support Sets System by Class. Nature-Inspired Computation and Machine Learning, 13th Mexican International Conference on Artificial Intelligence, MICAI 2014 Tuxtla Gutiérrez, Mexico, November 16–22, 2014 Proceedings, Part II, pp 128–139. Springer, México (2014)