

PREDICCIÓN DE DESERCIÓN LABORAL UTILIZANDO ALGORITMOS GENÉTICOS Y REDES NEURONALES ARTIFICIALES

Gonzalo Reyes-Huertas

20120535@aloe.ulima.edu.pe

Universidad de Lima. Lima, Perú

Resumen

El objetivo del trabajo de investigación es desarrollar una herramienta que permita identificar la posible deserción de un empleado, entendiendo la deserción laboral como un problema internacional. El método propuesto consiste en un algoritmo genético que identifica las variables relevantes y mejora la arquitectura de una red neuronal artificial como solución. Las variables seleccionadas por la herramienta concordaban con las variables recopiladas de distintos estudios, descubriéndose que no todas eran consideradas en dichos estudios (e.g., distancia del hogar al trabajo y años totales trabajando). Asimismo, las variables y la arquitectura seleccionadas por la herramienta permitieron predecir la deserción laboral hasta un 88,92 % de exactitud.

Palabras clave: red neuronal artificial, algoritmo genético, rotación de personal, arquitectura de redes neuronales

Abstract

Predicting job abandonment through genetic algorithms and artificial neural networks

This research work aims to develop a tool to identify employees who might abandon their position, because job abandonment is considered an international problem. The proposed method consists of a genetic algorithm that allows identifying the significant variables and improving the architecture of an artificial neural network as a solution. The variables selected by the tool were similar to those collected from different studies but not all of them were considered in such studies (e.g., distance between home and workplace, and years of employment). Likewise, the variables and architecture selected by the tool allowed to predict job abandonment up to 88.92 % accuracy rate.

Keywords: artificial neural network, genetic algorithm, employee turnover, neural network architecture

1. Introducción

Se considera la deserción laboral como un problema internacional persistente, incluso a pesar de los avances y la implementación de sistemas de gratificación y retención de empleados. Según Fidalgo y Gouveia (2011), al año 2009, más del 50 % de empleados inmobiliarios en las organizaciones portuguesas se retiró antes de los 4 años de vinculación. En Estados Unidos, en el sector de TI, los empleados tienden a permanecer entre 3 y 5 años en la misma compañía, independientemente de la generación a la que pertenecen (Burnes, 2006). Igualmente, para las empresas de *outsourcing*, en India, la tasa de deserción laboral es alta; especialmente para los *call centers*, alcanzando tasas hasta de 40 % en algunas firmas (Anand, Saravanasudhan y Vijesh, 2012).

En el caso del Perú, es el país con la tercera tasa de rotación de personal más elevada de Latinoamérica. A 2017, el índice de rotación fue del 18 %, mientras que el de otros países oscilaba entre 5-10 % (“Perú es el tercer país con la mayor rotación de personal en Latam”, 2017). Adicionalmente, el promedio de permanencia, a 2014, fue de 3 años; ese mismo año la tasa de rotación en puestos de altos cargos llegó al 25 % (“Rotación laboral de ejecutivos en altos mandos alcanza el 25 %”, 2014). De acuerdo a Jack Welch (empresario estadounidense), citado en Flores (2014), el costo de rotar a un gerente equivale a 5 años de su remuneración. Este tipo de impacto financiero se refleja en la pérdida de 2 millones de dólares por rotación de personal en Arequipa (“Mal clima laboral genera pérdidas millonarias”, 2016). Estos estudios demuestran que en el Perú existe un problema de deserción laboral, el cual dificulta el cumplimiento de los objetivos que se mencionan a continuación.

Para el 2030, el Perú desea “lograr el empleo pleno y productivo y el trabajo decente para todas las mujeres y los hombres, incluidos los jóvenes [...]” (Naciones Unidas/CEPAL, 2016); así como también “reducir considerablemente la proporción de jóvenes que no están empleados [...]” para el 2020 (Naciones Unidas/CEPAL, 2016), objetivos que, de no cumplirse, significarían la existencia de jóvenes con desventajas profesionales y dificultad para estar empleados.

Este problema afecta tanto a los empleados como a las empresas. Las empresas tienen que invertir tiempo y dinero en la capacitación del nuevo empleado; además, sufren retraso en procesos o proyectos, dependiendo de la importancia del puesto; adicionalmente, se pierde el conocimiento tácito de los empleados retirados (Fidalgo y Gouveia, 2011). Desde el punto de vista del empleado, este tiene que adaptarse a la cultura organizacional de la empresa y aprender el conocimiento interno de la misma para llevar a cabo sus tareas.

La principal dificultad, al momento de clasificar a un empleado como “futuro desertor”, es escoger correctamente las variables significativas que afectan la posibilidad de deserción y qué tanto la afectan. Algunos estudios, en el contexto peruano, coinciden en que los empleados se retiran por falta de motivación, mal clima laboral y falta de incentivos. Sin embargo, estos estudios suelen ser empíricos e individuales para cada variable.

En el presente trabajo de investigación se pretende conocer y seleccionar matemáticamente las principales variables que conllevan a los empleados a abandonar su trabajo, con el fin de poder desarrollar un modelo predictivo con arquitectura óptima que permita clasificar a los posibles empleados desertores.

2. Estado del arte

2.1 Identificación de variables

Según Fidalgo y Gouveia (2011), existen dos tipos de variables que contribuyen a la deserción laboral: aquellas que la empresa puede controlar, tales como salario, horario, línea de carrera, etc.; y las que no puede controlar, reubicación, problemas personales, etc. Los estudios revisados tratan sobre el primer tipo.

Es cierto que las razones por las que los empleados se retiran varían de empresa a empresa, y de región en región (Pawar y Chakravarthy, 2014); sin embargo, varios estudios (principalmente empíricos y cualitativos) con frecuencia consideran similares variables (ver tabla 1).

Tabla 1

Relación de variables consideradas en los diferentes estudios

Variables	Anand <i>et al.</i> (2012)	Ashok (2003)	Burnes (2006)	Firth <i>et al.</i> (2004)	Min (2007)	Pawar y Chakravarthy (2014)	Wei y Hei (2007)
Salario	X	X					
Ambiente laboral	X				X	X	
Línea de carrera	X		X			X	
Satisfacción laboral	X			X			X
Beneficios		X					X
Compromiso organizacional			X	X	X		X
Relación con superiores		X	X	X	X	X	
Stress laboral		X				X	
Estado civil						X	
Género						X	

Elaboración propia

Según Firth, Mellor, Moore y Loquet (2004), las responsabilidades de puestos con personal a cargo deben incluir el monitoreo de carga de trabajo, relación con subordinados y jefes (para reducir el estrés) y las fuentes de motivación intrínsecas y extrínsecas de los empleados. Adicionalmente, Ashok (2003) concuerda con que es responsabilidad de la empresa asegurar la satisfacción de sus empleados. También, autores como Min (2007), Pawar y Chakravarthy (2014), Wei y Hai (2007) enfatizan que la principal razón por la que los empleados se van no es el dinero. Por otro lado, las variables encontradas en los estudios del contexto peruano, en general, fueron (Flores, 2014; "Mal clima laboral genera pérdidas millonarias", 2016; "Rotación laboral de ejecutivos en altos mandos alcanza el 25 %", 2014; Sandoval, 2014): (mal) clima laboral, (ausencia de) beneficios laborales, (bajo) salario, búsqueda de mayor competitividad. Estas variables recolectadas de estudios en el Perú tienen coherencia con las variables descritas en la tabla 1.

2.2 Optimización de redes neuronales artificiales con un algoritmo genético

En esta sección se mencionan los estudios revisados para comprender la integración del algoritmo genético (AG) con la red neuronal artificial (RNA); en cada uno se describe su aporte distintivo.

2.2.1 Finding optimal neural network architecture using genetic algorithms

El objetivo de este artículo es encontrar la arquitectura óptima de una RNA para problemas particulares. Los pesos de las RNA fueron ajustados con *back-propagation* y el criterio de evaluación se basa en la exactitud de clasificación de las mismas. Fizelew *et al.* (2007) consideran 20 RNA como un buen número de población sobre la cual aplicar el AG. Para la selección de cromosomas se aplicó el método del torneo. El resultado fue que la RNA generada por el AG tenía mejor porcentaje para la clasificación de elementos previamente no vistos, que las RNA creadas empírica o aleatoriamente. Los autores concluyeron que el AG ayuda a generar la mejor RNA dependiendo del caso dado y el tipo de datos a utilizar.

2.2.2 Neural architectures optimization and genetic algorithms

El artículo propone una formulación matemática para determinar la función de activación de una red neuronal, el número óptimo de capas ocultas y los pesos de sus nodos. El objetivo es resolver el problema de la optimización. La optimización del desempeño se obtiene minimizando el error (distancia) entre la salida y el valor real (Ettaouil y Ghanou, 2009). Para ello se hizo uso del AG. Entonces, se mapean las RNA a dos cromosomas: matriz de pesos y vector (u), que es un arreglo de variables de decisión. La población inicial se generó aleatoriamente; a cada cromosoma se le asignó un *fitness* que corresponde a la *performance* de la RNA correspondiente, y

se calcula con base en el porcentaje de exactitud obtenido. Para la selección de cromosomas se utilizó el método de la ruleta. Además, debido a que se trataba de dos cromosomas, el *crossover* estaba basado en 2 puntos (generalmente es 1). El operador de mutación permitió mantener la diversidad en los cromosomas y no converger en un máximo local.

2.2.3 Genetic algorithm for neural network architecture optimization

El artículo consultado (Idrissi, Ramchoun, Ghanou y Ettaouil, 2016) propone una formulación matemática multiobjetivo, a fin de determinar el número óptimo de capas ocultas, el número de neuronas en cada capa y el mejor valor de los pesos de sus respectivas salidas. Por lo general, la elección de la arquitectura de la RNA es hecha de forma empírica. El modelo propuesto por los autores fue una RNA *feed forward*, es decir, una RNA en la que sus nodos no interactúan con otros nodos de su misma capa, y *back propagation* para el cálculo de la gradiente. La tasa de aprendizaje empleado fue de 0,1. Asimismo, para el AG, se plantearon dos funciones objetivo: menor número posible de capas ocultas y minimizar el MSE de la RNA. Las restricciones que se consideraron fueron: debe existir al menos una capa oculta y si no se usa una capa oculta, no se usan sus nodos. Para poder calcular el *fitness* con base en ambas funciones objetivo, se escaló el valor resultante de cada una y se sumaron; tras lo cual se calcula su inversa.

3. Metodología

3.1 Procesamiento de datos

Primero, se normalizó la data: se filtraron las variables categóricas; de cada variable no categórica, se calculó la media y la desviación estándar. A cada valor se sustrajo la media y se dividió entre la desviación estándar. Con esto, la media de los datos era 0 y su desviación estándar era igual a 1. De esta forma, se redujo la diferencia entre la magnitud de las variables no-categóricas a considerar. En segundo lugar, se eliminaron las columnas/características que no aportaban a la solución. Algunos de los valores eran constantes para todos los registros o representaban lo mismo (ej., *daily rate*, *hourly rate*, *monthly rate* que representaban ingreso económico). Tercero, se aplicaron 2 tipos de codificación a las variables categóricas: para las variables que presentaban 2 categorías (ej., ¿trabaja sobretiempo?: sí o no), se toma una categoría como 0 y la otra como 1 (ej., no = 0, sí = 1). Para el caso de variables con más de 2 categorías (ej., departamento/área), se aplicó el *one hot encoding*, que consiste en crear una columna por cada categoría y, por cada registro (fila), se marca con 1 la categoría a la que pertenece y se pone 0 en las demás.

3.2 Propuesta de solución

Una vez aplicado el procesamiento de datos, estos pasaron por los modelos de aprendizaje elegidos del estado del arte, en este caso, redes neuronales artificiales (RNA) y algoritmo genético (AG). Para tener una mayor visibilidad de la herramienta de solución, en la figura 1 se muestra un gráfico de bloques con los módulos que se trabajaron.

Tomando en cuenta lo antes señalado, el programa se dividió en 3 módulos: el primero (explicado en la figura 1), dedicado a la lectura, normalización, codificación y partición de datos; el segundo, dedicado a entrenar, validar y probar la RNA; y el tercero, dedicado al AG, que invoca al módulo de la RNA cuando se desea convertir el cromosoma a RNA y ejecutar métodos exclusivos de la RNA (entrenamiento, validación, prueba, etc.).

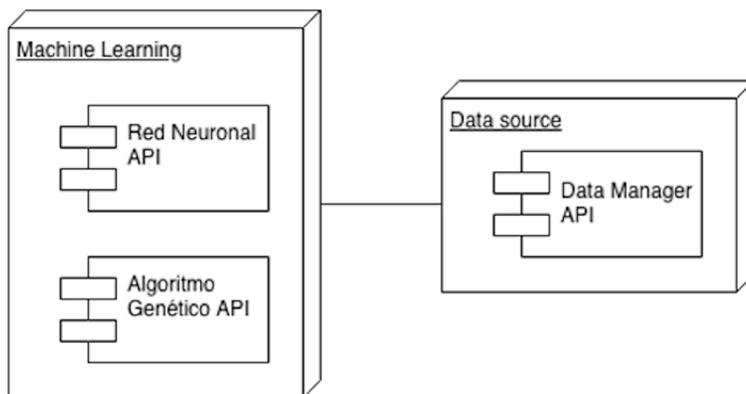


Figura 1. Representación en bloques de la solución

Elaboración propia

En la figura 2 se detalla la integración entre ambos modelos (AG y RNA). Inicialmente se debe tener una población de cromosomas, donde cada uno de ellos representa una RNA. Primero, se transforma cada cromosoma a la RNA que representa (considerando variables de entrada y capas ocultas), para entrenarla y validarla. Una vez entrenada y validada, se procede a hacer la predicción; luego, al ajuste del cromosoma se le asigna el valor de la exactitud resultante de la predicción. Los pasos mencionados se repiten por cada cromosoma en la población. Después de haber calculado el ajuste de todos los cromosomas de la población, se aplican los operadores genéticos escogidos (*crossover* y mutación) y se seleccionan los cromosomas de la siguiente generación.

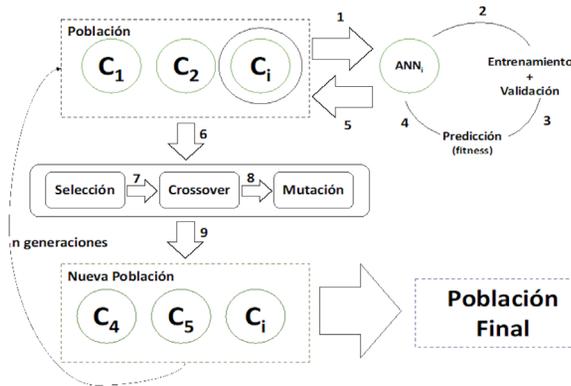


Figura 2. Representación gráfica de la propuesta de solución

Elaboración propia

El conjunto de pasos anteriores se itera n veces, hasta que el modelo ha convergido y se obtiene la población final, dentro de la cual estará el cromosoma que representa a la RNA con mayor exactitud.

Una ejecución significa haber iterado n veces el AG y haber obtenido la población final. Durante el desarrollo de la solución, se ejecutó más de una vez el programa.

3.3 Representación del cromosoma

Cada cromosoma de la población representa una RNA; dicha representación consiste en un vector (a) el cual a su vez estará compuesto por dos vectores, (b) y (c), donde:

$$b = (b_0, b_1, \dots, b_i, \dots, b_n) \quad (1)$$

b_i indica si la variable x_i se considera ($b_i = 1$) o no ($b_i = 0$)

$$c = (c_0, c_1, \dots, c_j, \dots, c_m) \quad (2)$$

donde el tamaño m del vector (c) representa la cantidad de capas ocultas y c_j es la cantidad de nodos en la capa oculta j .

Como se observa, los elementos del cromosoma corresponden a los elementos de la RNA, previamente, seleccionados y clasificados como variables. Debido a que los vectores (b) y (c) representan un cromosoma, ambos están sujetos a cambios de operadores genéticos en cada generación. La función objetivo del cromosoma se calcula con base en la exactitud de la red neuronal. El valor del ajuste varía entre 0 y 1, siendo 1 más favorable.

3.4 Restricciones del modelo

Después de iterar múltiples veces, se optó por mantener las 3 restricciones que se indican a continuación.

- Para las variables de entrada: $10 \leq n \leq 20$
donde n es la cantidad de variables de entradas que tiene la RNA.
- Las restricciones para las capas ocultas son: $1 \leq m \leq 3$
donde m es la cantidad de capas ocultas que tiene la RNA.
- Las restricciones para los nodos ocultos son: $5 \leq k_i \leq 10$
donde k_i es la cantidad de nodos en la i -ésima capa oculta.

3.5 Operadores genéticos utilizados

Como se indicó, los operadores genéticos a utilizar eran *crossover* y mutación. A continuación, se presenta la descripción y el pseudocódigo de sus implementaciones en el módulo del AG.

La función *crossover_single* intercambia los genes de dos cromosomas, con lo que se generan dos cromosomas hijos con genes combinados de los dos cromosomas originales.

Input: l, r : cromosomas padres

Output: cromosomas con genes intercambiados por el operador *crossover*, hijos de l y r

Variables:

$c1, c2$: cromosomas hijos de l y r

i : índice que indica el punto de intercambio de genes de los cromosomas

crossover_single(l,r):

```
i = random(min(l.size, r.size))
c1 = l[0:i] + r[i:r.size]
c2 = r[0:i] + l[i:l.size]
crossover_single = c1, c2
```

La función *mutate_single* altera el valor de algún gen del cromosoma, dependiendo de un porcentaje de probabilidad.

Input: c: cromosoma a mutar

Variables: gene: gen del cromosoma

Constantes:

MIN: valor mínimo posible para el gen del cromosoma

MAX: valor máximo posible para el gen del cromosoma

MUTATION_RATE: probabilidad de que el gen sea mutado

```
mutate_single(c) :
    foreach gene in c do
        if random(0,1) < MUTATION_RATE then
            gene = random(MIN, MAX)
```

Dado que el cromosoma está compuesto por dos vectores: (b) y (c), ambos operadores se aplican a cada vector. Es decir, tanto las variables de entrada como los nodos y capas ocultas están sujetos a posibles modificaciones por el AG en cada generación.

3.6 Método de selección

El método de selección de cromosomas implementado fue el del torneo. Con este método, se seleccionaban los cromosomas que pasarían a la siguiente generación. A continuación, se detalla su uso y pseudocódigo.

La función *select_tournament* selecciona aleatoriamente cromosomas de la población, y devuelve el cromosoma con el ajuste más alto.

Input: p: población de cromosomas

Output: cromosoma con mayor ajuste (precisión)

Variables:

new_p: arreglo de cromosomas seleccionados

i: contador de cromosomas seleccionados

Constantes: TOURNAMENT_SELECTION_SIZE: tamaño máximo de cromosomas seleccionados

```

select_tournament(p):

    new_p = []

    i = 0

    while i != TOURNAMENT_SELECTION_SIZE do

        c = p[random(p.size)]

        new_pop.append(c)

        i = i + 1

    select_tournament = new_p.sorted()[0]

```

3.7 Configuración de la red neuronal

Para las RNA, se utilizó la función sigmoidea como función de activación, empleada en la investigación de Islam, Baharudin, Raza y Nallagownden (2014). De igual manera, el algoritmo de *back-propagation* para el ajuste de pesos, que fue el algoritmo validado en las investigaciones de Ettaouil y Ghanou (2009), Fiszlewe *et al.* (2007), Idrissi *et al.* (2016) e Islam *et al.* (2014).

Según investigaciones revisadas (Fiszlewe *et al.*, 2007; Echegaray y Barrios, 2015), se utilizó el método de validación cruzada para la validación de RNA, por lo que fue el método que se tuvo en cuenta para validar a la RNA durante su entrenamiento, de tal manera que, si estuviera en riesgo de subajuste o sobreajuste, tenga una menor precisión. Esto se debe a que al aplicar *cross-validation* se calcula la precisión promedio de K ejecuciones ($K = 10$ en este caso) con data particionada de diferente manera en cada iteración.

3.8 Configuración del algoritmo genético

Como ya se ha mencionado, los operadores genéticos del GA son *crossover* y mutación. El método de selección, torneo. El tamaño inicial de la población escogido fue de 25, como en el estudio de Echegaray y Barrios (2015).

4. Resultados

Una vez aplicado el procesamiento de datos explicado al inicio de la sección anterior, se obtuvieron las variables de entrada sobre las cuales se llevó a cabo el análisis (ver tabla 2).

Tabla 2
Variables de entrada luego del procesamiento de datos

Id	Variables
X00	Age
X01	DistanceFromHome
X02	Education
X03	EnvironmentSatisfaction
X04	Gender
X05	JobInvolvement
X06	JobLevel
X07	JobSatisfaction
X08	MonthlyIncome
X09	NumCompaniesWorked
X10	OverTime
X11	PercentSalaryHike
X12	PerformanceRating
X13	RelationshipSatisfaction
X14	StockOptionLevel
X15	TotalWorkingYears
X16	TrainingTimesLastYear
X17	WorkLifeBalance
X18	YearsAtCompany
X19	YearsInCurrentRole
X20	YearsSinceLastPromotion
X21	YearsWithCurrManager
X22	Business Travel = Non-Travel
X23	Business Travel = Travel_Frequently
X24	Business Travel = Travel_Rarely
X25	Department = Human Resources
X26	Department = Research & Development
X27	Department = Sales
X28	EducationField = Human Resources
X29	EducationField = Life Sciences
X30	EducationField = Marketing
X31	EducationField= Medical
X32	EducacionField = Other
X33	EducationField = Technical Degree
X34	JobRole = Healthcare Representative

(continúa)

(continuación)

X35	JobRole = Resources
X36	JobRole = Laboratory Technician
X37	JobRole = Manager
X38	JobRole = Manufacturing Director
X39	JobRole = Research Director
X40	JobRole = Research Scientist
X41	JobRole = Sales Executive
X42	JobRole = Sales Representative
X43	MaritalStatus = Divorced
X44	MaritalStatus = Married
X45	MaritalStatus = Single

Elaboración propia. *Dataset* obtenido de <https://www.kaggle.com/pavansubhasht/ibm-hr-analytics-attrition-dataset>

4.1 Selección de variables

Para reducir la variabilidad de las capas ocultas de la RNA, se propuso utilizar solo una capa oculta, con cantidad de nodos igual a:

$$\text{Nodos} = 1/2 (1 + \sum_{i=1}^n v_i)$$

donde, v es el vector que representa las variables a utilizar y n , la longitud del mismo vector.

Entonces, la cantidad de nodos en la capa oculta es igual a la mitad de la suma de la cantidad de nodos en la capa de entrada más la cantidad de nodos en la capa de salida (1). Con esto, se logra reducir la influencia de las capas ocultas en el valor de la exactitud.

El programa se ejecutó 6 veces, lográndose una precisión máxima de 87,2 % (ver tabla 3). Se intentó obtener mayor precisión creando las RNA que consideren las variables utilizadas más frecuentemente en las ejecuciones E1 hasta E6, manteniendo los nodos y capas ocultas constantes.

Tabla 3
Precisión obtenida por AG en 6 ejecuciones (E1...E6)

	E1 (Gen 567)	E2 (Gen 407)	E3 (Gen 442)	E4 (Gen 446)	E5 (Gen 342)	E6 (Gen 376)
Precisión	0,866	0,870	0,872	0,865	0,865	0,864

Elaboración propia

Luego se generaron 3 nuevas RNA, la primera considero las variables que tuvieron una frecuencia (F) mayor o igual a 5; la segunda, mayor o igual a 4; la tercera, mayor o igual a 3 (ver tabla 4).

Tabla 4
Comparación de RNA propuestas en base a la frecuencia de las variables

	E(F>=5)	E(F>=4)	E(F>=3)
Promedio de precisión	0,87458	0,86836	0,868213
Desviación estándar	0,005138	0,00565	0,004236

Elaboración propia

Después de haber corrido 10 veces el módulo de RNA del programa, se obtuvo una nueva relación de variables, luego de comparar las redes neuronales propuestas. Las variables de entrada con una frecuencia de aparición mayor o igual a 5 (F>=5) tuvieron la mayor exactitud (ver tablas 5 y 6).

Tabla 5
Variables que inicialmente aparecieron en 5 o más ejecuciones

Rótulo	Variable
X00	Edad
X01	Distancia (en km) desde la casa al trabajo
X03	Satisfacción con el ambiente laboral
X04	Género
X05	Participación laboral
X07	Satisfacción laboral
X10	Sobretiempo
X18	Cantidad de años en la compañía

(continúa)

(continuación)

X20	Cantidad de años desde último ascenso
X23	Trabajo requiere que viaje frecuentemente
X30	Área de marketing
X40	Rol de investigador

Elaboración propia

Tabla 6
Agrupación de variables recopiladas y seleccionadas

	Variables recopiladas	Variables seleccionadas
V01	Salario	-
V02	Ambiente laboral	X03
V03	(Falta) línea de carrera	X20
V04	Satisfacción laboral	X07
V05	Beneficios	X23
V06	Compromiso organizacional	X05, X18
V07	Relación con superiores y compañeros	X03, X05, X07
V08	Estrés laboral	X10
V09	Estado civil	-
V10	Género	X04

Elaboración propia

Adicionalmente, se obtuvieron variables que fueron descartadas o no se encontraron, explícitamente, en el estado del arte (ej., beneficios no monetarios [X10] de la compañía y frecuencia de viaje por trabajo [X23]), pero que fueron seleccionadas por la herramienta:

X00: edad

X01: distancia (en km) desde la casa al trabajo

X23: trabajo requiere que viaje frecuentemente

X30: área de *marketing*

X40: rol de investigador

Por otro lado, las variables recopiladas que el modelo no seleccionó fueron:

V01: salario

V09: estado civil

En la investigación cualitativa de Burnes (2006) se concluyó que en el sector de las tecnologías de la información no influía tanto la edad, ya que había más similitudes que diferencias entre los empleados de la generación X y de la generación no-X. Sin embargo, los datos utilizados son multiárea; según la herramienta, la edad sí es una variable significativa.

Las variables “área de marketing” (X30) y “rol de investigador” (X40) son variables que tampoco se encontraron en el estado del arte. Parecen ser más específicas al caso dado, por lo que su interpretación sería: la deserción laboral también varía por área y rol del empleado.

4.2 Selección de capas y nodos ocultos

Debido a que se encontraron las variables que maximizan la exactitud del modelo, se constató que ellas se mantuvieron constantes (ver tabla 7).

Tabla 7

Precisión lograda por AG en 6 ejecuciones (E1...E6) manteniendo las variables constantes

	E1	E2	E3	E4	E5	E6
Capas y nodos ocultos	(6)	(6, 3)	(12)	(7)	(5, 3, 2)	(5, 3, 1)
Máxima precisión	0,8734	0,8800	0,8746	0,8780	0,8680	0,0865

Elaboración propia

La tabla 7 indica las mejores configuraciones obtenidas por el AG. La interpretación de la nomenclatura de capas y nodos es la misma que se explicó previamente.

Luego se obtuvo la precisión promedio de las RNA, utilizando las configuraciones antes señaladas y ejecutando el módulo de RNA 10 veces (igual al procedimiento empleado previamente).

Tabla 8
Comparación de RNA en base a sus capas y nodos ocultos

Ejecución	C1 (6)	C2 (6, 3)	C3 (12)	C4 (7)	C5 (5, 3, 2)	C6 (5, 3, 1)
1	0,8826	0,8724	0,8742	0,8749	0,8708	0,8461
2	0,8707	0,8699	0,8691	0,8733	0,8571	0,8648
3	0,8724	0,8597	0,8750	0,8775	0,8648	0,8546
4	0,8775	0,8657	0,8681	0,8751	0,8682	0,8486
5	0,8690	0,8734	0,8572	0,8776	0,8657	0,8427
6	0,8758	0,8640	0,8699	0,8750	0,8520	0,8537
7	0,8792	0,8623	0,8759	0,8693	0,8571	0,8622
8	0,8699	0,8716	0,8665	0,8699	0,8682	0,8470
9	0,8844	0,8682	0,8818	0,8892	0,8562	0,8529
10	0,8699	0,8656	0,8674	0,8750	0,8563	0,8529
Promedio de precisión	0,8751	0,8672	0,8705	0,8756	0,8616	0,8525

Elaboración propia

De la tabla 8 se puede ver que la mejor configuración fue la C4, que cuenta con 7 nodos en su única capa oculta, ya que obtuvo la precisión promedio mayor y una precisión máxima de 88,92 %.

Es cierto que no se percibe mucha mejora entre esta configuración (C4) y la hallada anteriormente (C1). Esto se debe a que la propuesta señalada previamente de utilizar una capa oculta y nodos igual a la mitad de variables seleccionadas, coincidió con ser la segunda mejor solución posible. Si se hubiera escogido una configuración de tres capas ocultas para esa sección, se hubiera tenido una precisión de 85 % como en C6. Debido a que la RNA tuvo doce variables de entrada, no fue necesario tener tantas capas ocultas como inicialmente se pensó. Tal vez, si en otro escenario se tuvieran más variables (como las 46 iniciales), sería más factible utilizar configuraciones de tres capas ocultas como C5 y C6.

5. Conclusiones

Se logró identificar y contrastar las variables recopiladas de estudios empíricos pasados con las variables seleccionadas por la herramienta propuesta. La mayoría de variables revisadas también fueron consideradas y son coherentes con las seleccionadas por el modelo, lo cual da más veracidad al mismo. Además, se consideraron nuevas variables que no se habían visto en la literatura revisada (ej., distancia de la casa al trabajo).

Adicionalmente, se logró establecer restricciones razonables para las potenciales arquitecturas de las redes neuronales que mejoraban la precisión (métrica seleccionada). Entonces, se logró crear una herramienta útil y de calidad que permite identificar la deserción de un empleado hasta un 88,92 % de precisión.

Referencias

- Anand, V. V., Saravanasudhan, R., y Vijesh, R. (2012). Employee attrition: A pragmatic study with reference to BPO industry. *Management*, 2(9), pp. 769-775. Recuperado de <https://ieeexplore.ieee.org/document/6215943/>
- Ashok, J. (2003). A study on linkage between organizational practices and employee attrition. *International Journal of Knowledge Management & Practice*, 1(1), pp. 30-34.
- Burnes, P. T. (2006). Voluntary employee turnover: Why IT professionals leave. *IT Professional*, 8(3), pp. 46-48. doi:10.1109/MITP.2006.78
- Echegaray Calderón, O. A., y Barrios Aranibar, D. (2015). Optimal selection of factors using Genetic Algorithms and Neural Networks for the prediction of students' academic performance. *Latin America Congress on Computational Intelligence (LA-CCI)*, pp. 1-6. doi:10.1109/LA-CCI.2015.7435976
- Ettaouil M., y Ghanou, Y. (2009). Neural architectures optimization and genetic algorithms. *WSEAS Transactions on Computers*, 8(3), pp. 526-537. Recuperado de <http://www.wseas.us/e-library/transactions/computers/2009/31-699.pdf>
- Fidalgo F., y Gouveia, L. B. (2012). Turnover impact into organizational knowledge management. The case of employee turnover in Portuguese Real Estate. *6th Iberian Conference on Information Systems and Technologies (CISTI 2011)*, pp. 1-6.
- Firth, L., Mellor, D. J., Moore, K. A., y Loquet, C. (2004). How can managers reduce employee intention to quit? *Journal of Managerial Psychology*, 19(2), pp. 170-187. doi:10.1108/02683940410526127
- Fiszelew, A., Britos, P., Ochoa, A., Merlino, H., Fernandez, E., y García-Martínez, R. (2007) Finding optimal neural network architecture using genetic algorithms. *Advances in Computer Science and Engineering Research in Computer Science*, 27, pp. 15-24. Recuperado de <http://www.iidia.com.ar/rgm/articulos/RCSJ-27-15-24.pdf>
- Flores, C. (22 de octubre de 2014). Perú tiene alta tasa de rotación de personal. *Diario Correo*. Recuperado de <https://diariocorreo.pe/economia/peru-tiene-alta-tasa-de-rotacion-de-personal-9770/>

- Idrissi, M. A. J., Ramchoun, H., Ghanou, Y., y Ettaouil, M. (2016). Genetic algorithm for neural network architecture optimization. *IEEE 3rd International Conference on Logistics Operations Management (GOL)*, pp. 1-4. doi:10.1109/GOL.2016.7731699
- Islam, B., Baharudin, Z., Raza, M. Q., y Nallagownden, P. (2014). Optimization of neural network architecture using genetic algorithm for load forecasting. *IEEE 5th International Conference on Intelligent and Advanced Systems (ICIAS)*, pp. 1-6. doi:10.1109/ICIAS.2014.6869528
- Mal clima laboral genera pérdidas millonarias a las empresas (3 de mayo de 2016). *La República*. Recuperado de <https://larepublica.pe/sociedad/936033-mal-clima-laboral-genera-perdidas-millonarias-a-las-empresas>
- Naciones Unidas/CEPAL (2016). Agenda 2030 y los Objetivos de Desarrollo Sostenible: una oportunidad para América Latina y el Caribe. doi:10.1017/CBO9781107415324.004
- Pawar, A., y Chakravarthy, V. (2014). Factors influencing employee turnover in fusion healthcare organization. *International Journal of Management Research and Reviews*, 4(9), pp- 834-846.
- Perú es el tercer país con la mayor rotación de personal en Latam (24 de octubre de 2017). *El Portal del Capital Humano*. Recuperado de <http://www.infocapitalhumano.pe/recursos-humanos/noticias-y-movidas/peru-es-el-tercer-pais-con-la-mayor-rotacion-de-personal-en-latam/>
- Rotación laboral de ejecutivos en altos mandos alcanza el 25 %. (7 de julio de 2014). *El Comercio*. Recuperado de <https://elcomercio.pe/economia/ejecutivos/rotacion-laboral-ejecutivos-altos-mandos-alcanza-25-346784>
- Sandoval, M. (27 de agosto de 2014). El Perú tiene el más alto índice de rotación laboral en Latinoamérica. *El Portal del Capital Humano*. Recuperado de <http://www.infocapitalhumano.pe/recursos-humanos/noticias-y-movidas/el-peru-tiene-el-mas-alto-indice-de-rotacion-laboral-en-latinoamerica/>
- Wei, X., y Hai, L. (2007). Chinese employee's turnover intentions in relation to organizational identification, work values, job satisfactions in service industry. *IEEE International Conference on Wireless Communications, Networking and Mobile Computing*, pp. 4435-4438. doi:10.1109/WICOM.2007.1095