

Facultad de ingeniería en Sistemas y ciencias de la Computación.

Bases de Datos II

Ing. Jorge Santos.

MINERIA DE DATOS

Nombre: Gabriel Adolfo Rodríguez Morales

Carné: 2890-10-5339

Algoritmo: Redes Neuronales

El algoritmo de red neuronal de Microsoft es una implementación de la popular arquitectura de red neuronal adaptable para el aprendizaje automático. El algoritmo prueba cada posible estado del atributo de entrada con cada posible estado del atributo de predicción, y calcula las probabilidades de cada combinación según los datos de aprendizaje. Puede usar estas probabilidades para tareas de clasificación o regresión, así como para predecir un resultado en función de algunos atributos de entrada. También se puede usar una red neuronal para el análisis de asociación.

Cuando se crea un modelo de minería de datos con el algoritmo de red neuronal de Microsoft, puede incluir varias salidas y el algoritmo creará varias redes. El número de redes incluidas en un modelo de minería de datos depende del número de estados (o valores de atributo) de las columnas de entrada, así como del número de columnas de predicción que usa el modelo de minería de datos y el número de estados de dichas columnas.

Casos de uso:

El algoritmo de red neuronal de Microsoft es útil para analizar datos de entrada complejos, como los datos de un proceso comercial o de producción, o problemas empresariales para los que hay una cantidad importante de datos de entrenamiento disponibles, pero en los que no es fácil derivar reglas mediante otros algoritmos.

Los casos sugeridos para utilizar el algoritmo de red neuronal de Microsoft son:

* Análisis de comercialización y promoción, como medir el éxito de una promoción por correo directo o una campaña publicitaria en la radio.
* Predecir los movimientos de las acciones, la fluctuación de la moneda u otra información financiera con gran número de cambios a partir de los datos históricos.
* Analizar los procesos industriales y de producción.
* Minería de texto.
* Cualquier modelo de predicción que analice relaciones complejas entre muchas entradas y relativamente pocas salidas.

Cómo funciona el algoritmo:

El algoritmo de red neuronal de Microsoft crea una red formada por hasta tres niveles de nodos (en ocasiones denominados *neuronas*). Estos niveles son el *nivel de entrada*, el *nivel oculto* y el *nivel de salida*.

**Nivel de entrada**: los nodos de entrada definen todos los valores de atributos de entrada para el modelo de minería de datos, así como sus probabilidades.

**Nivel oculto**: los nodos ocultos reciben entradas de los nodos de entrada y proporcionan salidas a los nodos de salida. El nivel oculto es donde se asignan pesos a las distintas probabilidades de las entradas. Un peso describe la relevancia o importancia de una entrada determinada para el nodo oculto. Cuanto mayor sea el peso asignado a una entrada, más importante será el valor de dicha entrada. Los pesos pueden ser negativos, lo que significa que la entrada puede desactivar, en lugar de activar, un resultado concreto.

**Nivel de salida**: los nodos de salida representan valores de atributo de predicción para el modelo de minería de datos.

Datos requeridos para los modelos de red neuronal:

El modelo de red neuronal debe contener una columna de clave, una o más columnas de entrada y una o más columnas de predicción.

Los modelos de minería de datos que usan el algoritmo de red neuronal de Microsoft están muy influenciados por los valores que se especifican en los parámetros disponibles para el algoritmo. Los parámetros definen cómo se muestrean los datos, cómo se distribuyen o cómo se espera que estén distribuidos en cada columna, y cuándo se invoca la selección de características para limitar los valores usados en el modelo final.

Ver un modelo de red neuronal:

Para trabajar con los datos y ver cómo el modelo pone en correlación las entradas y salidas, puede usar el **Visor de redes neuronales de Microsoft**. Con este visor personalizado, puede filtrar los atributos de entrada y sus valores, y ver gráficamente cómo afectan a las salidas. La información sobre herramientas del visor muestra la probabilidad y la mejora respecto al modelo predictivo asociados a cada par de valores de entrada y de salida.

La manera más fácil de explorar la estructura del modelo consiste en usar el **Visor de árbol de contenido genérico de Microsoft**. Este visor le permitirá ver las entradas, las salidas y las redes creadas por el modelo, así como hacer clic en cualquier nodo para expandirlo y ver las estadísticas relacionadas con los niveles de entrada, los niveles de salida y los niveles ocultos de los nodos.

Crear predicciones:

Una vez procesado el modelo, puede usar la red y los pesos almacenados dentro de cada nodo para realizar predicciones. Un modelo de red neuronal admite el análisis de regresión, de asociación y de clasificación. Por lo tanto, el significado de cada predicción puede ser diferente. También puede consultar el propio modelo, revisar las correlaciones encontradas y recuperar las estadísticas relacionadas.

Referencia técnica del algoritmo de red neuronal

La red neuronal de Microsoft usa una red de tipo *perceptrón multinivel*, que también se denomina *red de tipo regla delta de propagación hacia atrás*, compuesta por tres niveles de neuronas o *perceptrones*. Estos niveles son: un nivel de entrada, un nivel oculto opcional y un nivel de salida.

Esta documentación no abarca una discusión detallada sobre redes neuronales de tipo perceptrón multinivel. En este tema, se explica la implementación básica del algoritmo, incluido el método usado para normalizar los valores de entrada y de salida, y los métodos de selección de características usados para reducir la cardinalidad de los atributos. En este tema, se describen los parámetros y otros valores que se pueden usar para personalizar el comportamiento del algoritmo; además, se proporcionan vínculos a información adicional sobre cómo consultar el modelo.

Implementación del algoritmo

En una red neuronal de tipo perceptrón multinivel, cada neurona recibe una o más entradas y genera una o más salidas idénticas. Cada salida es una función no lineal simple de la suma de las entradas a la neurona. Las entradas pasan de los nodos del nivel de entrada a los nodos del nivel oculto y, a continuación, al nivel de salida; no existe ninguna conexión entre neuronas del mismo nivel. Si no se incluye ningún nivel oculto, tal y como pasa en un modelo de regresión logística, las entradas pasan directamente desde los nodos del nivel de entrada a los nodos del nivel de salida.

Existen tres tipos de neuronas en una red neuronal creada con el algoritmo de red neuronal de Microsoft:

**Neuronas de entrada**

Las neuronas de entrada proporcionan valores de atributo de entrada para el modelo de minería de datos. En el caso de los atributos de entrada discretos, las neuronas de entrada suelen representar un único estado del atributo de entrada. Esto incluye los valores ausentes, si los datos de entrenamiento contienen valores NULL para ese atributo. Un atributo de entrada discreto que tiene más de dos estados genera una neurona de entrada por cada estado y una neurona de entrada para un estado ausente, si existen valores NULL en los datos de entrenamiento. Un atributo de entrada continuo genera dos neuronas de entrada: una neurona para un estado ausente y otra neurona para el valor del propio atributo continuo. Las neuronas de entrada proporcionan entradas para una o más neuronas ocultas.

**Neuronas ocultas**

Las neuronas ocultas reciben entradas de las neuronas de entrada y proporcionan salidas a las neuronas de salida.

**Neuronas de salida**

Las neuronas de salida representan valores de atributo de predicción para el modelo de minería de datos. En el caso de los atributos de entrada discretos, una neurona de salida suele representar un único estado de predicción para un atributo de predicción, incluidos los valores que faltan. Por ejemplo, un atributo de predicción binario produce un nodo de salida que describe un estado ausente o existente, que indica si existe un valor para ese atributo. Una columna booleana utilizada como un atributo de predicción genera tres neuronas de salida: una neurona para un valor true, otra neurona para un valor false y una última neurona para un estado existente o ausente. Un atributo de predicción discreto que tiene más de dos estados genera una neurona de salida por cada estado y una neurona de salida para un estado ausente o existente. Las columnas de predicción continuas generan dos neuronas de salida: una neurona para un estado existente o ausente y otra neurona para el valor de la propia columna continua. Si se generan más de 500 neuronas de salida al revisar el conjunto de columnas de predicción, Analysis Services genera una red nueva en el modelo de minería de datos para representar las neuronas de salida adicionales.

Una neurona recibe la entrada de otras neuronas o de otros datos, dependiendo del nivel de la red en que se encuentra. Una neurona de entrada recibe entradas de los datos originales. Las neuronas ocultas y las neuronas de salida reciben entradas de la salida de otras neuronas de la red neuronal. Las entradas establecen relaciones entre neuronas; estas relaciones sirven como ruta de análisis para un conjunto específico de escenarios.

Cada entrada tiene un valor asignado denominado *peso*, que describe la relevancia o importancia de dicha entrada en la neurona oculta o en la neurona de salida. Cuanto mayor sea el peso asignado a una entrada, más importante o relevante será el valor de dicha entrada. Los pesos pueden ser negativos, lo cual implica que la entrada puede desactivar, en lugar de activar, una neurona específica. El valor de cada entrada se multiplica por el peso para poner de relieve la importancia de la entrada de una neurona específica. En el caso de pesos negativos, el efecto de multiplicar el valor por el peso es una pérdida de importancia.

Cada neurona tiene una función no lineal sencilla asignada denominada *función de activación* que describe la relevancia o importancia de una neurona específica para ese nivel de una red neuronal. Las neuronas ocultas usan una función *tangente hiperbólica* (tanh) para su función de activación, mientras que las neuronas de salida usan una función *sigmoidea* para la activación. Ambas son funciones no lineales continuas que permiten que la red neuronal modele relaciones no lineales entre neuronas de entrada y salida.

**Redes neuronales de entrenamiento**

Existen varios pasos implicados en el entrenamiento de un modelo de minería de datos que utiliza el algoritmo de red neuronal de Microsoft. Estos pasos están muy influenciados por los valores que se especifican en los parámetros de algoritmo.

En primer lugar, el algoritmo evalúa y extrae los datos de entrenamiento del origen de datos. Un porcentaje de los datos de entrenamiento, denominado *datos de exclusión*, se reserva para evaluar la precisión de la red. Durante el proceso de entrenamiento, la red se evalúa de forma inmediata después de cada iteración mediante los datos de entrenamiento. Cuando la precisión deja de aumentar, el proceso de entrenamiento se detiene.

Los valores de los parámetros *SAMPLE\_SIZE* y *HOLDOUT\_PERCENTAGE* se usan para determinar el número de casos de muestra de los datos de aprendizaje y el número de casos que se apartan en los datos de exclusión. El valor del parámetro *HOLDOUT\_SEED* se usa para determinar de forma aleatoria los casos individuales que se apartan para los datos de exclusión.

A continuación, el algoritmo determina el número y la complejidad de las redes que admite el modelo de minería de datos. Si el modelo contiene uno o más atributos que solamente se utilizan para la predicción, el algoritmo crea una única red que representa todos estos atributos. Si el modelo de minería de datos contiene uno o más atributos que se utilizan para la entrada y la predicción, el proveedor del algoritmo construye una red para cada atributo.

En el caso de los atributos de entrada y de predicción que tienen valores discretos, cada neurona de entrada o de salida representa respectivamente un único estado. En el caso de los atributos de entrada y de predicción que tienen valores continuos, cada neurona de entrada o de salida representa respectivamente el intervalo y la distribución de valores del atributo. El número máximo de estados admitidos en cada caso depende del valor del parámetro de algoritmo *MAXIMUM\_STATES*. Si el número de estados para un atributo específico supera el valor del parámetro de algoritmo *MAXIMUM\_STATES*, se eligen los estados más comunes o relevantes para dicho atributo hasta alcanzar el número máximo permitido y el resto de los estados se agrupan como valores ausentes para el análisis.

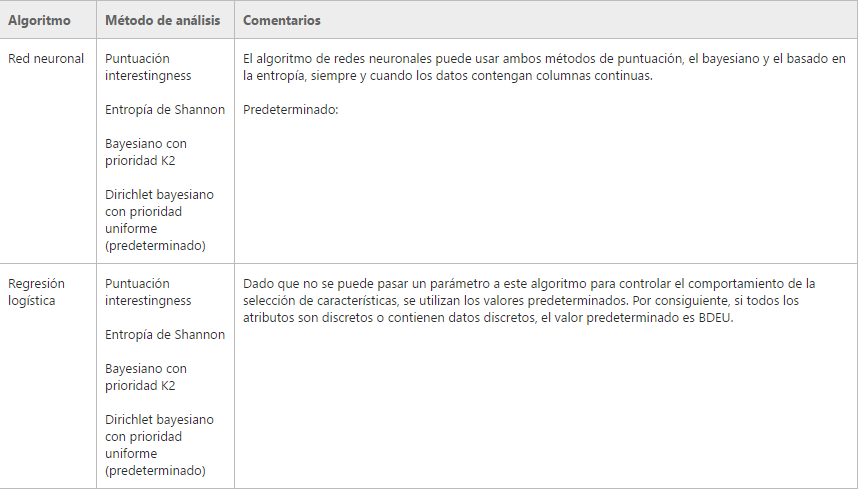
Después, el algoritmo usa el valor del parámetro *HIDDEN\_NODE\_RATIO* al determinar el número inicial de neuronas que se crearán para el nivel oculto. Puede establecer *HIDDEN\_NODE\_RATIO* en 0 para evitar la creación de un nivel oculto en las redes que genera el algoritmo para el modelo de minería de datos y tratar la red neuronal como una regresión logística.

El proveedor de algoritmos evalúa iterativamente el peso de todas las entradas de la red simultáneamente, tomando el conjunto de datos de entrenamiento reservado anteriormente y comparando el valor real conocido de cada escenario de los datos de exclusión con la predicción de la red, en un proceso conocido como *aprendizaje por lotes*. Una vez que el algoritmo ha evaluado el conjunto completo de los datos de entrenamiento, revisa el valor predicho y real de cada neurona. El algoritmo calcula el grado de error, si lo hay, y ajusta los pesos asociados con las entradas de esa neurona, trabajando hacia atrás desde las neuronas de salida a las de entrada en un proceso conocido como *propagación hacia atrás*. A continuación, el algoritmo repite el proceso en todo el conjunto de datos de entrenamiento. Dado que el algoritmo puede admitir múltiples pesos y neuronas de salida, el algoritmo de gradiente conjugado se utiliza para guiar el proceso de entrenamiento en la asignación y evaluación de los pesos de las entradas. Esta documentación no abarca una discusión sobre el algoritmo de gradiente conjugado.

Selección de características:

Si el número de atributos de entrada es mayor que el valor del parámetro *MAXIMUM\_INPUT\_ATTRIBUTES* (o si el número de atributos de predicción es mayor que el valor del parámetro *MAXIMUM\_OUTPUT\_ATTRIBUTES*), se usa un algoritmo de selección de características para reducir la complejidad de las redes que se incluyen en el modelo de minería de datos. La selección de características reduce el número de atributos de entrada o de predicción a los más relevantes estadísticamente para el modelo.

Todos los algoritmos de minería de datos de Analysis Services usan automáticamente la selección de características para mejorar el análisis y reducir la carga de procesamiento. El método usado para la selección de características en los modelos de red neuronal depende del tipo de datos del atributo. Como referencia, en la tabla siguiente se muestran los métodos de selección de características usados para los modelos de red neuronal; además, se muestran los métodos de selección de características usados para el algoritmo de regresión logística, que está basado en el algoritmo de red neuronal.



Los parámetros de algoritmo que controlan la selección de características para un modelo de red neuronal son MAXIMUM\_INPUT\_ATTRIBUTES, MAXIMUM\_OUTPUT\_ATTRIBUTES y MAXIMUM\_STATES. También puede controlar el número de niveles ocultos mediante el establecimiento del parámetro HIDDEN\_NODE\_RATIO.

Métodos de puntuación:

La *puntuación* es un tipo de normalización que, en el contexto del entrenamiento de un modelo de red neuronal, hace referencia al proceso de convertir un valor, como una etiqueta de texto discreta, en un valor que se pueda comparar con otros tipos de entradas y que se pueda pesar en la red. Por ejemplo, si un atributo de entrada es Sexo y los valores posibles son Hombre y Mujer, y otro atributo de entrada es Ingresos, con un intervalo de valores variable, los valores para cada atributo no son comparables directamente y, por consiguiente, deben estar codificados a una escala común para que se puedan calcular los pesos. Puntuar es el proceso de normalizar tales entradas para los valores numéricos, específicamente, para un intervalo de probabilidades. Las funciones usadas para la normalización también ayudan a distribuir más uniformemente los valores de entrada en una escala uniforme para que los valores extremos no distorsionen los resultados del análisis.

Las salidas de la red neuronal también están codificadas. Si hay un único destino para la salida (es decir, la predicción), o varios destinos que se usan solo para la predicción, no para la entrada, el modelo crea una red única y es posible que no sea necesario normalizar los valores. Sin embargo, si se usan varios atributos para la entrada y la predicción, el modelo debe crear varias redes; por tanto, se deben normalizar todos los valores y, al salir de la red, las salidas deberán estar codificadas.

La codificación de las entradas se basa en la suma de cada valor discreto de los casos de entrenamiento y en la multiplicación de ese valor por su peso. Esto se denomina *suma ponderada*, que se pasa a la función de activación del nivel oculto. Para la codificación, se usa la puntuación-z:

**Valores discretos**

μ = p (la probabilidad anterior de un estado)

StdDev = sqrt(p(1-p))

**Valores continuos**

Valor presente= 1 - μ/σ

Ningún valor existente= -μ/σ

Una vez codificados los valores, se realiza una suma ponderada de las entradas, con los extremos de la red como pesos.

La codificación de las salidas usa la función sigmoidea, que tiene propiedades que la hacen muy útil para la predicción. Una de esas propiedades es que, sin tener en cuenta cómo se ajusta la escala de los valores originales, y sin tener en cuenta si los valores son negativos o positivos, la salida de esta función es siempre un valor entre 0 y 1, lo que resulta apropiado para la estimación de probabilidades. Otra propiedad útil es que la función sigmoidea tiene un efecto suavizador que hace que cuando los valores se alejan del punto de inflexión, la probabilidad del valor se aproxima lentamente a 0 o a 1.

Personalizar el algoritmo de red neuronal

El algoritmo de red neuronal de Microsoft admite varios parámetros que afectan al comportamiento, al rendimiento y a la precisión del modelo de minería de datos resultante. También puede modificar la forma en la que el modelo procesa los datos; para ello, puede establecer marcas de modelado en las columnas o marcas de distribución que especifiquen cómo se deben procesar los valores dentro de la columna.

Establecer parámetros del algoritmo:

En la tabla siguiente, se describen los parámetros que se pueden usar con el algoritmo de red neuronal de Microsoft.

HIDDEN\_NODE\_RATIO  
Especifica la proporción de neuronas ocultas por neuronas de entrada y de salida. La siguiente fórmula determina el número inicial de neuronas de la capa oculta:

HIDDEN\_NODE\_RATIO \* SQRT(Total de neuronas de entrada \* Total de neuronas de salida)

El valor predeterminado es 4,0.

HOLDOUT\_PERCENTAGE  
Especifica el porcentaje de casos de los datos de entrenamiento que se han usado para calcular el error de datos de exclusión, que se usa como parte de los criterios de detención durante el entrenamiento del modelo de minería de datos.

El valor predeterminado es 30.

HOLDOUT\_SEED  
Especifica un número que se usa para inicializar el generador pseudoaleatorio cuando el algoritmo determina aleatoriamente los datos de exclusión. Si este parámetro se establece en 0, el algoritmo genera la inicialización basada en el nombre del modelo de minería de datos para garantizar que el contenido del modelo permanece intacto al volver a realizar el proceso.

El valor predeterminado es 0.

MAXIMUM\_INPUT\_ATTRIBUTES  
Determina el número máximo de atributos de entrada que se pueden proporcionar al algoritmo antes de emplear la selección de características. La función de selección de atributos de entrada se deshabilita cuando este valor se establece en 0.

El valor predeterminado es 255.

MAXIMUM\_OUTPUT\_ATTRIBUTES  
Determina el número máximo de atributos de salida que se pueden proporcionar al algoritmo antes de emplear la selección de características. La característica de selección de atributos de salida se deshabilita cuando este valor se establece en 0.

El valor predeterminado es 255.

MAXIMUM\_STATES  
Especifica el número máximo de estados discretos por atributo que admite el algoritmo. Si en número de estados de un atributo específico es mayor que el número especificado para este parámetro, el algoritmo utiliza los estados más frecuentes de este atributo y trata al resto como estados que faltan.

El valor predeterminado es 100.

SAMPLE\_SIZE  
Especifica el número de casos que se van a usar para realizar el entrenamiento del modelo. El algoritmo utiliza el valor menor entre este número o el porcentaje del total de escenarios que no están incluidos en los datos de exclusión, según se especifica en el parámetro HOLDOUT\_PERCENTAGE.

En otras palabras, si HOLDOUT\_PERCENTAGE se establece en 30, el algoritmo utilizará el valor de este parámetro o un valor igual al 70 por ciento del número total de casos, según cuál sea menor.

El valor predeterminado es 10000.

Marcas de distribución:

El algoritmo de red neuronal de Microsoft admite las siguientes marcas de distribución. Las marcas solo se usan como sugerencias para el modelo; si el algoritmo detecta una distribución diferente, usará la distribución encontrada, no la proporcionada en la sugerencia.

Normal  
Indica que los valores de la columna se deben tratar como si representasen la distribución normal o gaussiana.

Uniforme  
Indica que los valores de la columna se deben tratar como si estuviesen distribuidos uniformemente; es decir, la probabilidad de cualquier valor es más o menos la misma y depende del número total de valores.

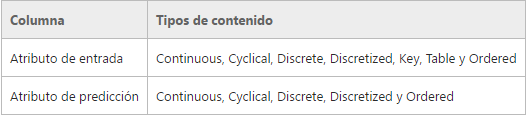
Logarítmica normal  
Indica que los valores de la columna tienen que tratarse como si estuviesen distribuidos según la curva *logarítmica normal*, lo que significa que el logaritmo de los valores se distribuye normalmente.

Requisitos

Un modelo de red neuronal debe contener por lo menos una columna de entrada y una columna de salida.

Columnas de entrada y de predicción:

El algoritmo de red neuronal de Microsoft admite las columnas de entrada y de predicción específicas que se enumeran en la tabla siguiente.



Contenido del modelo de minería de datos para los modelos de red neuronal (Analysis Services)

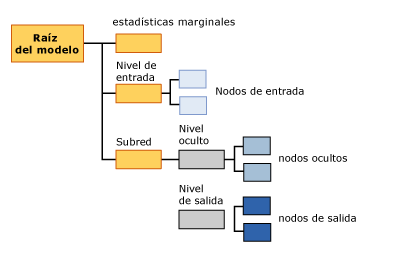
En este tema se describe el contenido del modelo de minería de datos específico de los modelos que utilizan el algoritmo de red neuronal de Microsoft. Para obtener una explicación de cómo interpretar las estadísticas y la estructura compartidas por todos los tipos de modelos, así como las definiciones generales de términos relacionados con el contenido del modelo de minería de datos.

Descripción de la estructura de un modelo de red neuronal:

Cada modelo de red neuronal tiene un nodo primario único que representa el modelo y sus metadatos, y un nodo de estadísticas marginal (NODE\_TYPE = 24) que proporciona estadísticas descriptivas sobre los atributos de entrada. El nodo de estadísticas marginal es útil porque resume información sobre las entradas, de modo que no necesite consultar los datos de los nodos individuales.

Debajo de estos dos nodos hay al menos dos nodos más y podría haber muchos más, en función de cuántos atributos de predicción tenga el modelo.

* El primer nodo (NODE\_TYPE = 18) siempre representa el nodo superior del nivel de entrada. Bajo este nodo superior, puede encontrar los nodos de entrada (NODE\_TYPE = 21) que contienen los atributos de entrada reales y sus valores.
* Cada nodo sucesivo contiene una *subred* diferente (NODE\_TYPE = 17). Cada subred siempre contiene un nivel oculto (NODE\_TYPE = 19) y un nivel de salida (NODE\_TYPE = 20) para esa subred.



La información del nivel de entrada es simple: el nodo superior de cada nivel de entrada (NODE\_TYPE = 18) actúa como un organizador para una colección de nodos de entrada (NODE\_TYPE = 21). El contenido de los nodos de entrada se describe en la tabla siguiente.

Cada subred (NODE\_TYPE = 17) representa el análisis de la influencia del nivel de entrada en un atributo de predicción determinado. Si hay varias salidas de predicción, hay varias subredes. El nivel oculto de cada subred contiene varios nodos ocultos (NODE\_TYPE = 22) que contienen los detalles sobre los pesos de cada transición que finaliza en ese nodo oculto en especial.

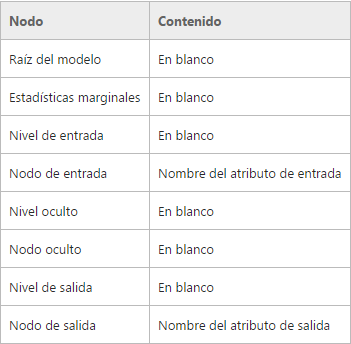
El nivel de salida (NODE\_TYPE = 20) contiene los nodos de salida (NODE\_TYPE = 23) que contiene cada uno valores distintos del atributo de predicción. Si el atributo de predicción es un tipo de datos numéricos continuo, solo hay un nodo de salida para el atributo.

Contenido de un modelo de red neuronal:

MODEL\_CATALOG  
Nombre de la base de datos en la que se almacena el modelo.

MODEL\_NAME  
Nombre del modelo.

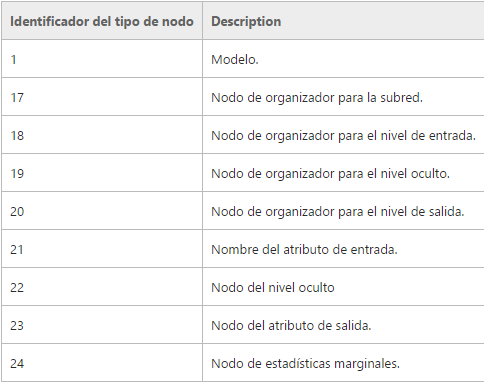
ATTRIBUTE\_NAME  
Nombres de los atributos que corresponden a este nodo.



NODE\_NAME  
Nombre del nodo. Esta columna contiene el mismo valor que NODE\_UNIQUE\_NAME.

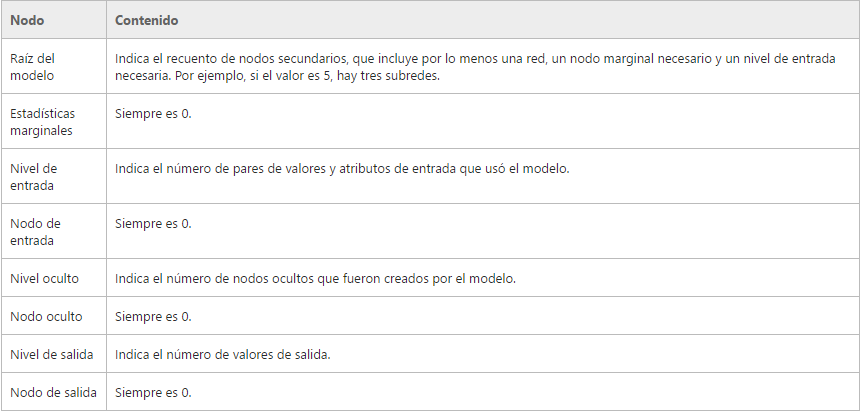
NODE\_UNIQUE\_NAME  
Nombre único del nodo.

NODE\_TYPE  
Un modelo de red neuronal genera únicamente los tipos de nodos siguientes:



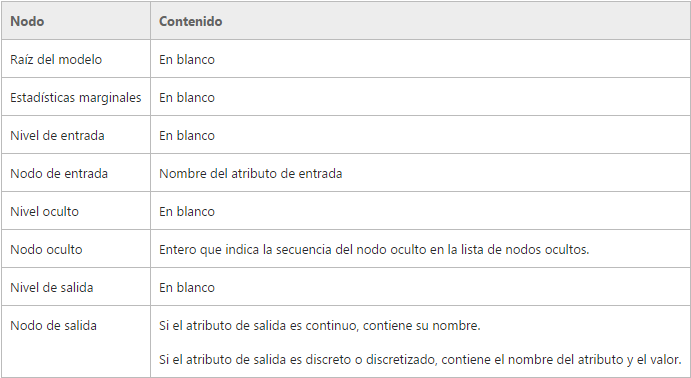
NODE\_CAPTION  
Etiqueta o título asociado al nodo. En los modelos de red neuronal, siempre está en blanco.

CHILDREN\_CARDINALITY  
Cálculo del número de elementos secundarios que tiene el nodo.

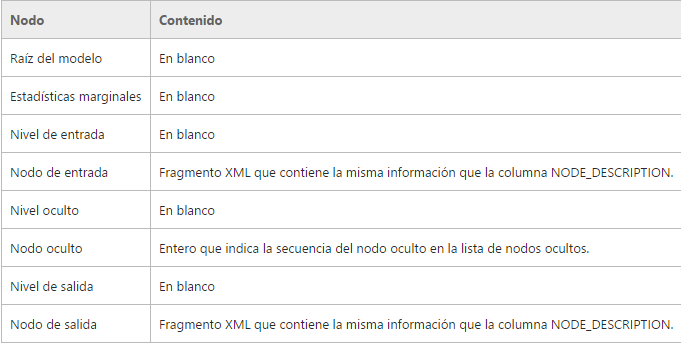


PARENT\_UNIQUE\_NAME  
Nombre único del nodo primario del nodo. Se devuelve NULL para todos los nodos del nivel raíz.

NODE\_DESCRIPTION  
Descripción fácil de comprender del nodo.



NODE\_RULE  
Descripción XML de la regla que está incrustada en el nodo.



MARGINAL\_RULE  
En los modelos de red neuronal, siempre está en blanco.

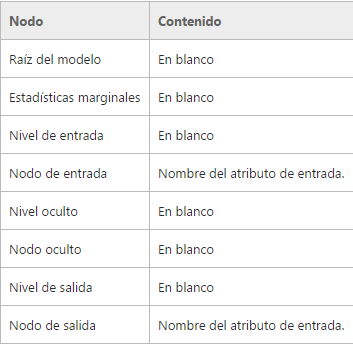
NODE\_PROBABILITY  
Probabilidad asociada a este nodo. En los modelos de red neuronal, siempre es 0.

MARGINAL\_PROBABILITY  
Probabilidad de alcanzar el nodo desde el nodo primario. En los modelos de red neuronal, siempre es 0.

NODE\_DISTRIBUTION  
Tabla anidada que contiene la información estadística para el nodo.

NODE\_SUPPORT  
En los modelos de red neuronal, siempre es 0.

MSOLAP\_MODEL\_COLUMN

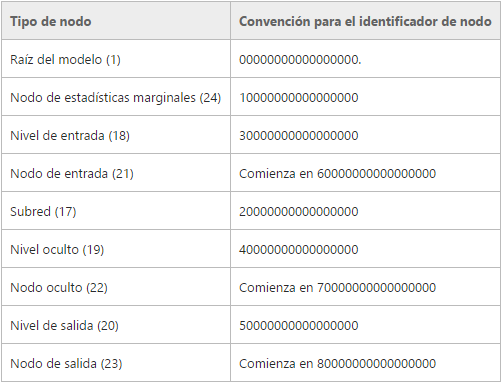


MSOLAP\_NODE\_SCORE  
En un modelo de red neuronal, siempre es 0.

MSOLAP\_NODE\_SHORT\_CAPTION  
En los modelos de red neuronal, siempre está en blanco.

Usar nombres de nodo e identificadores

La denominación de los nodos de un modelo de red neuronal proporciona información adicional sobre el tipo de nodo, para facilitar relacionar el nivel oculto con el nivel de entrada y el nivel de salida con el nivel oculto. En la tabla siguiente se muestra la convención para los identificadores que están asignados a los nodos de cada nivel.



Puede determinar qué atributos de entrada se relacionan con un nodo de nivel oculto concreto viendo la tabla NODE\_DISTRIBUTION en el nodo oculto (NODE\_TYPE = 22). Cada fila de la tabla NODE\_DISTRIBUTION contiene el identificador de un nodo de atributo de entrada.

De igual forma, puede determinar qué niveles ocultos se relacionan con un atributo de salida viendo la tabla NODE\_DISTRIBUTION en el nodo de salida (NODE\_TYPE = 23). Cada fila de la tabla NODE\_DISTRIBUTION contiene el identificador de un nodo del nivel oculto, junto con el coeficiente relacionado.

Interpretar la información de la tabla NODE\_DISTRIBUTION

La tabla NODE\_DISTRIBUTION puede estar vacía en algunos nodos. Sin embargo, para los nodos de entrada, los nodos del nivel oculto y los nodos de salida, la tabla NODE\_DISTRIBUTION almacena información importante e interesante sobre el modelo. Como ayuda para interpretar esta información, la tabla NODE\_DISTRIBUTION contiene una columna VALUETYPE para cada fila que indica si el valor de la columna ATTRIBUTE\_VALUE es Discreto (4), Discretizado (5) o Continuo (3).

Nodos de entrada:

El nivel de entrada contiene un nodo para cada valor del atributo que se utilizó en el modelo.

**Atributo discreto:** el nodo de entrada solo almacena el nombre del atributo y su valor en las columnas ATTRIBUTE\_NAME y ATTRIBUTE\_VALUE. Por ejemplo, si [Work Shift] es la columna, se crea un nodo independiente para cada valor de esa columna que se utilizó en el modelo, como AM y PM. La tabla NODE\_DISTRIBUTION para cada nodo muestra solo el valor actual del atributo.

**Atributo numérico de datos discretos:** el nodo de entrada almacena el nombre del atributo y el valor, que puede ser un intervalo o un valor concreto. Las expresiones, como '77.4 - 87.4' o ' < 64.0', representan todos los valores para el valor de [Time Per Issue]. La tabla NODE\_DISTRIBUTION para cada nodo muestra solo el valor actual del atributo.

**Atributo continuo:** el nodo de entrada almacena el valor medio del atributo. La tabla NODE\_DISTRIBUTION para cada nodo muestra solo el valor actual del atributo.

Nodos del nivel oculto:

El nivel oculto contiene un número variable de nodos. En cada uno, la tabla NODE\_DISTRIBUTION contiene las asignaciones del nivel oculto a los nodos del nivel de entrada. La columna ATTRIBUTE\_NAME contiene un identificador de nodo que corresponde a un nodo del nivel de entrada. La columna ATTRIBUTE\_VALUE contiene el peso asociado a esa combinación de nodo de entrada y nodo de nivel oculto. La última fila de la tabla contiene un coeficiente que representa el peso de ese nodo oculto en el nivel oculto.

Nodos de salida

El nivel de salida contiene un nodo de salida para cada valor de salida que se utilizó en el modelo. En cada nodo, la tabla NODE\_DISTRIBUTION contiene las asignaciones del nivel de salida a los nodos del nivel oculto. La columna ATTRIBUTE\_NAME contiene un identificador de nodo que corresponde a un nodo del nivel oculto. La columna ATTRIBUTE\_VALUE contiene el peso asociado a esa combinación de nodo de salida y nodo de nivel oculto.

La tabla NODE\_DISTRIBUTION tiene la información adicional siguiente, en función de si el tipo del atributo es:

**Atributo discreto:** las dos filas finales de la tabla NODE\_DISTRIBUTION contienen un coeficiente para el nodo en conjunto y el valor actual del atributo.

**Un atributo numérico de datos discretos:** idéntico a los atributos discretos, solo que el valor del atributo es un intervalo de valores.

**Atributo continuo:** las dos filas finales de la tabla NODE\_DISTRIBUTION contienen la media del atributo, el coeficiente para el nodo en conjunto y la varianza del coeficiente.