CET
Centro de Estudios Tributarios
Universidad de Chile
www.cetuchile.cl

EVOLUCIÓN DE LA FISCALIZACIÓN DEL IVA Y LAS OPORTUNIDADES QUE DA LA FACTURA ELECTRÓNICA

Iván Beltrand Cruz

Ingeniero Civil Industrial, Universidad de Chile.
MBA, Universidad de Chile.
BinaryBag Limitada.

Gonzalo Vitta Fuentes

Ingeniero Civil en Computación, Universidad de Chile.
BinaryBag Limitada.

Daniel Zúñiga Correa

Analista Desarrollador de Aplicaciones de Software, DUOC.
Data Scientist.
BinaryBag Limitada.

1. INTRODUCCIÓN

La fiscalización del IVA se ve evolucionar y tomar características de servicio al contribuyente en la medida que la capacidad de procesamiento de la Administración Tributaria (AT) se robustece.

A mediados de la década del 2000, sin factura electrónica, el análisis se basó en declaraciones juradas o informativas de compras y ventas vs la declaración mensual de impuestos (cruce de IVA). Luego con la llegada de la Factura Electrónica se comenzó el rediseño del análisis del cumplimiento del IVA basándose en información de la Factura Electrónica, así mismo se diseñó el proceso de generar una propuesta de la declaración de IVA, lo que se materializó el 2016, una vez se implementó la obligatoriedad del uso de la Factura Electrónica. El SII de Chile, ha sido un ejemplo y pionero en estas materias, donde aprovechó su experiencia y capacidades ya desarrolladas en la propuesta de declaración del Impuesto a al Renta de Personas Naturales.

Se puede resumir la evasión en tres formas:

i. El abuso de la norma, por ejemplo, comprando bienes o servicios que no están asociados a la actividad del negocio necesariamente (se compra una camioneta a nombre de la empresa, pero se usa familiarmente y no en el negocio).
ii. La no emisión del comprobante al Consumidor Final, ya que se pierde la cadena del IVA en su totalidad.
iii. El uso de Facturas Falsas.

Al revisar la fiscalización del IVA en los países que han adoptado este impuesto, se puede apreciar como su fiscalización ha evolucionado de la mano del uso de las Tecnologías de la Información.

Es así como se observa desde el comienzo del nuevo siglo (y antes) el enfoque asociado al control del contribuyente, su identificación y el control sobre los documentos tributarios que debía emitir para respaldar la transacción con su cliente. Esto se hacía y se sigue haciendo con el control de la autorización de folios o secuencia por los distintos tipos de documentos de transacción (facturas a contribuyentes o a consumidor final). En este control primario, se busca entregar la cantidad adecuada de folios, y restringirlos en cantidad, según el comportamiento del cumplimiento tributario del emisor.

Avanzando en la capacidad tecnológica de recibir y procesar información, las Administraciones Tributarias (AT), comenzaron a solicitar información de las transacciones de los contribuyentes, partiendo de resúmenes anuales, semestrales y mensuales, hasta llegar al detalle de cada transacción con la Factura electrónica.

Así mismo, pasamos de procesos de fiscalización asociados a acciones presenciales para el control de los documentos emitidos, al cruce de los reportes periódicos de información para determinar brechas, definiendo así los programas de fiscalización a realizar en un periodo amplio de tiempo, teniendo muy en consideración los plazos de prescripción que cada AT debe respetar.

De esta forma, con la llegada de la factura electrónica, se pudo llegar a la propuesta de la declaración misma del IVA, ya que es posible establecer la carga tributaria teórica del contribuyente. Esto es un avance en la facilitación del cumplimiento tributario clave.

Sin embargo, a pesar de esta capacidad de anticipar las declaraciones, es interesante analizar el futuro de la fiscalización del IVA ante el uso inminente de herramientas de BIG Data y Machine Learning.

2. ¿QUÉ ES EL BIG DATA?

El concepto Big Data fue formalizado por Gartner en 2001 (y continúa siendo la definición de referencia): Big data son datos que contienen una mayor variedad y que se presentan en volúmenes crecientes y a una velocidad superior. Una definición más actual (2016) establece que "Big data representa los activos de información caracterizados por un volumen, velocidad y variedad tan altos que requieren una tecnología específica y métodos analíticos para su transformación en valor". Esto se conoce como "las tres V".

2.1. Las "tres V" de Big Data

i. Volumen: La creciente digitalización de procesos, genera una disponibilidad cada vez mayorde datos que procesar, en particular, grandes volúmenes de datos no estructurados de baja densidad. Puede tratarse de datos de valor desconocido, como feeds de datos de Twitter, flujos de clics de una página web o aplicación para móviles, equipo con sensores. Para algunas organizaciones, esto puede suponer terabytes de datos. Para otras, incluso cientos de petabytes.

ii. Velocidad: La velocidad es el ritmo al que se generan, reciben y (posiblemente) al que se utilizan los datos. Algunos productos inteligentes habilitados para Internet funcionan en tiempo real o prácticamente en tiempo real y requieren una evaluación y actuación en tiempo real.

iii. Variedad: La variedad hace referencia a los diversos tipos de datos disponibles. Los tipos de datos convencionales son estructurados y pueden organizarse en una base de datos relacional. Con el creciente incremento en la digitalización, los datos se presentan en nuevos tipos de datos no estructurados. Los tipos de datos no estructurados y semiestructurados, como el texto, audio o video, requieren un preprocesamiento adicional para poder obtener significado y habilitar los metadatos.

2.2. ¿Cómo se implementa?

El desarrollo de modelos de programación paralelizable y la creciente disponibilidad de clúster de almacenamiento y procesamiento además del desarrollo del commodity computing han permitido el desarrollo de frameworks de programación que proporcionan interfaces para la programación de clusters completos con Paralelismo de Datos y tolerancia a fallos. Los principales frameworks utilizados por la industria son:

i. **Apache Hadoop:** Es un framework de software, bajo licencia libre, para programar aplicaciones distribuidas que manejen grandes volúmenes de datos. Permite a las aplicaciones trabajar con miles de nodos en red y petabytes de datos. Hadoop se inspiró en los documentos de Google sobre MapReduce y Google File System (GFS).

ii. **Apache Spark:** Es un sistema de computación en clúster de propósito general y orientado a la velocidad. Proporciona APIs en Java, Scala, Python y R. También proporciona un motor optimizado que soporta la ejecución de grafos en general.

iii. **Apache Flink:** Es un framework de procesamiento de flujo de código abierto desarrollado por la Apache Software Foundation. El núcleo de Apache Flink es un motor de flujo de datos de transmisión distribuida escrito en Java y Scala que ejecuta programas de flujo de datos arbitrarios de forma paralela y canalizada.

3. APLICACIÓN DE BIG DATA

3.1 Detección de fraude

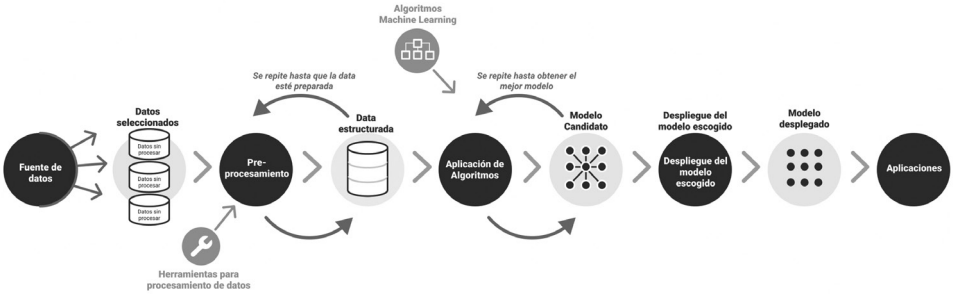
Las transacciones electrónicas realizadas mediante interfaces en línea poseen una estructura de funcionamiento, una serie de pasos y de elementos de seguridad que deben ser vigilados por las instituciones para garantizar que todos sus pasos cumplen con las normativas. Estas operaciones permiten estudiar millones de transacciones que están almacenadas en bases de datos y diferenciar las operaciones normales, de las operaciones riesgosas.

Las herramientas de Big data permiten seleccionar los atributos que necesitamos dentro de los conjuntos de datos para estudiar a profundidad los elementos claves de un fraude, determinar características específicas y depurar los datos a estudiar.

Dentro de este tipo de análisis el proceso de minería de datos es fundamental debido a que ayuda a realizar el proceso de filtrado y selección de datos a estudiar. Con el resultado de este proceso junto al desarrollo de modelos basados en Machine Learning se puede estudiar relaciones complejas de datos y transformarlos en conocimiento. En particular, se pueden entrenar modelos que asignan una probabilidad de riesgo a una determinada operación (compra, venta, solicitud de crédito) basado en los millones de operaciones conocidas por la institución.

3.2 ¿Que es Machine Learning?[1]

El concepto Machine Learning se refiere a la detección automática de patrones significativos en la data, mediante complejos procesos computacionales, que pueden usarse para determinar si los datos de un conjunto pertenecen a una determinada categoría, o predevcir un número específico asociado a ellos.



Fuente: Elaboración propia

El diagrama explica el proceso, que se puede resumir en los siguientes pasos:

i. Obtención del conjunto de datos: Este paso considera la totalidad de datos con los que se cuente, sin distinción, por ejemplo, todas las facturas, las declaraciones mensuales y anuales y la información del catastro de contribuyentes

ii. Análisis y selección de los datos a utilizar: De la totalidad de los datos obtenidos, se determina cuales de ellos son relevantes para el entrenamiento de los modelos, determinando también si se construirán modelos predictivos o clasificatorios.

iii. Pre procesamiento y estructuración de los datos con significancia:Se realiza un proceso de limpieza y feature engineering[2] sobre la data seleccionada, determinando el o los vectores objetivo unidimensionales

1 Si es de su interés, se recomiendan las siguientes lecturas respecto del tema
https://ai.stanford.edu/people/nilsson/mlbook.html
https://web.stanford.edu/~hastie/ElemStatLearn/
https://www.deeplearningbook.org/
https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/machine-learning-modules

2 Análisis y caracterización de datos sin procesar utilizando técnicas de minería de datos.

(valores), la significancia estadística de los atributos asociados a ellos, y su correlación. En este paso también se recodifican los atributos y se prepara la data para la implementación de los modelos de Machine Learning, como, por ejemplo, describir y modelar casos conocidos de comportamiento, como un proveedor de facturas falsas, un sub declarante, contribuyentes calzados, etc.

iv. Aplicación de los algoritmos de Machine Learning, y obtención de las métricas para la determinación del mejor candidato.

Este paso se subdivide en varios puntos:

a) Elección de los modelos: se escogen modelos a construir, del tipo ya determinado previamente, en base la data a utilizar.

b) Construcción de los modelos: se construyen varias versiones de cada modelo escogido en el punto anterior, ajustados con distintos valores en sus hiperparámetros.[3]

c) División de la data en entrenamiento y validación.

d) Entrenamiento de los modelos construidos.

e) Obtención de las métricas correspondientes a cada modelo entrenado.

f) Comparación de las métricas y elección de los mejores modelos en base a dicha comparación. Es decir, comparar el caso teórico con los resultados.

v. Despliegue del modelo seleccionado en ambiente de producción: Se serializa el mejor modelo y despliega en el ambiente de producción, ajustando las aplicaciones correspondientes para su utilización. Este paso incluye la implementación de la automatización del proceso de limpieza y feature engineering en la data sobre la que actuará el modelo.

vi. Ajustes correctivos y reentrenamiento del modelo: Periódicamente se debe revisar la nueva data por si presenta cambios en su estructura, junto con realizar evaluaciones periódicas sobre el modelo. Tanto en el caso de observarse cambios en la estructura de la data, como en el que se observe una disminución en los valores de las métricas de las evaluaciones del modelo, se deberán construir nuevos modelos y contrastar sus métricas con las del modelo anterior.

3 Valores utilizados durante el entrenamiento.

Los algoritmos se dividen en dos grandes grupos: *modelos regresivos y modelos clasificatorios.*

i. Los modelos clasificatorios literalmente *obtienen la clasificación que le corresponde a un conjunto de datos,* como por ejemplo determinar si un correo es o no spam (en base a su contenido), si un cliente comprará o no un producto (en base a sus preferencias y comportamientos previos), si una persona tendrá o no cáncer (en base a su comportamiento, historial médico y datos familiares), etcétera.

ii. Los modelos regresivos *predicen un valor numérico,* como por ejemplo el precio que tendrá determinado producto, el rango de temperaturas de los próximos días, la cantidad de visitas que tendrá un sitio de internet, etcétera.

En ambos casos el conjunto de valores o categorías a estimar se denominan *vector objetivo,* y los datos utilizados para su estimación se denominan *atributos.*

Además, varios modelos regresivos tienen versiones clasificatorias que, en vez de calcular un valor, calculan la probabilidad de que el conjunto de datos pertenezca a una u otra categoría, contrastando el valor de dicha probabilidad con *puntajes de corte (o rangos de corte)* por categoría.

4. MODELOS CLASIFICATORIOS

Los modelos clasificatorios actualmente más usados son los siguientes:

i. Regresión Logística

Utilizamos la regresión lineal (ver más adelante), donde calculamos los coeficientes de la regresión mediante el *método de máxima verosimilitud*. Para los casos discretos, el estimador de máxima verosimilitud da como resultado un valor máximo para la probabilidad conjunta, o un valor que maximiza la probabilidad de la muestra.

La filosofía de la estimación de la máxima verosimilitud proviene de la noción de que *el estimador razonable de un parámetro que se basa en información muestral es el valor del parámetro que produce la mayor probabilidad de obtener la muestra*. Y en estos casos discretos, la verosimilitud es la probabilidad de observar de manera conjunta los valores en la muestra. Se presentará el siguiente modelo a estimar:

$$\log\left(\frac{\Pr(y)}{1-\Pr(y)}\right) = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \cdots + \beta_n * X_n + \varepsilon_i$$

ii. Bayes Ingenuo Multinomial

El algoritmo Bayes-Ingenuo pertenece a la familia de los clasificadores generativos, que son aquellos que aprenden características de un modelo a partir de la probabilidad conjunta entre atributos y un espacio finito de clases a asignar a observaciones mediante el Teorema de Bayes Ingenuo, que a su vez gravita alrededor de la probabilidad condicional (la probabilidad que un evento A ocurra condicional a la ocurrencia de otro evento B). En el contexto del Teorema de Bayes, decimos que la probabilidad a posteriori es proporcional a *la verosimilitud de la probabilidad a priori, ajustada por la evidencia*. De esta forma tendremos:

$$\Pr(posteriori) = \frac{\Pr(Verosimilitud) * \Pr\,(priori)}{\Pr\,(Evidencia)}$$

iii. Máquina de Soporte Vectorial (Support Vector Machine o SVM)

El algoritmo separa los datos de entrenamiento en dos grupos en base a un hiperplano N-dimensional construido con alguna función polinomial-homogénea, de perceptrón, radial Gaussiana o sigmoidea; de forma tal que cada dato nuevo será clasificado en uno de dichos grupos en base a su posición y cercanía al hiperplano divisor.

iv. Arboles de Decisión

Los árboles de decisión permiten predecir una clase asociada con una instancia específica recorriendo desde un nodo principal hasta un nodo terminal (u hoja).

El objetivo de los árboles de decisión es encontrar una serie de reglas de división del espacio muestral que *optimicen la decisión local*. En cada nodo se busca resolver la siguiente función de regresión con constantes:

$$\hat{f}(x) = \sum_{m=1}^{S} c_m I((X_1, X_2) \in \mathbb{R}_m)$$

Donde $\mathbb{R}_m$ es la región $\mathbb{R}_m \ni (X_1, X_2)$, y c_m producto de la partición definida en el espacio de atributos , y es el promedio del *vector objetivo* en la región.

El objetivo del modelo es que, dado una partición del espacio de atributos, encontrar un *puntaje de corte* (denominado s) que asigne una nueva observación en una de las dos regiones:

$$\mathbb{R}_1(j, s) = \{X | X_j \leq s\} \text{ ó } \{X | X_j > s\}$$

v. Bosques Aleatorios (Random Forest)

Los bosques son métodos de ensamble paralelo con forma funcional que utilizan múltiples árboles de decisión cuyos valores individuales son considerados votos y, en consecuencia, la clasificación final dependerá del resultado de la votación.

vi. Potenciación Adaptativa (AdaBoost)

Es un modelo que obedece a la estrategia de ponderar la muestra de entrenamiento en cada iteración, buscando disminuir el error (la diferencia entre el valor esperado y el predicho). Tratándose de un modelo clasificatorio, tendremos un vector objetivo expresado como:

$$y \in Y = \{0,1\}$$

Dado un conjunto de atributos X, un clasificador específico $h(X)$ produce una predicción en una de las dos clases. Podemos obtener la tasa de error a lo largo de todos los clasificadores mediante:

$$\bar{\varepsilon_i} = \frac{\sum_{i=1}^{N} \prod(y_i \neq h(x_i))}{\mathrm{N}}$$

Y estimar la esperanza del error en el conjunto de clasificadores como

$$E_{xy} \left[\prod (y \neq h(X)) \right]$$

El algoritmo utiliza clasificadores débiles, en particular árboles de decisión configurados para actuar sobre una baja profundidad (pocos nodos) y que devuelven resultados con una probabilidad de acierto marginalmente mejor que la simple aleatoriedad, potenciando los resultados de dichos aciertos mediante un ponderador calculado en base al error.

vii. Potenciación del Gradiente (Gradient Boosting, o GBoosting)

El algoritmo también utiliza *clasificadores débiles,* pero potencia los resultados de cada iteración minimizando el error residual en base al gradiente de su función de pérdida:

$$f_0(x) = \min_{\gamma \in \mathbb{R}} \sum_{i=1}^{N} l(y_i, \emptyset(x_{i\gamma}))$$

Sin embargo, dicha potenciación es limitada para evitar un sobre ajuste del modelo, esto es, que el algoritmo prediga con gran exactitud sobre los datos usados para su entrenamiento, pero no le sea posible predecir de igual forma sobre nuevos datos no utilizados en su entrenamiento.

viii. Potenciación Extrema del Gradiente (Extreme Gradient boosting o XGBoost)

Consiste en una implementación de la potenciación del gradiente, pero con una regularización más eficiente del control del sobre ajuste del modelo, mejorando el uso de los árboles de decisión, a objeto de obtener mejores resultados sobre la predicción de datos no utilizados en su entrenamiento:

$$L^{(t)} = \sum_{i=1}^{N} l(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)) + \boldsymbol{\Omega}(f_t)$$

Además, XGBoost tiene la particularidad de que puede completar los datos faltantes gracias al uso de su función de pérdida.

5. MODELOS REGRESIVOS

Los modelos regresivos actualmente más usados son los siguientes:

i. Regresión Lineal

$$G_i = \beta_0 + \beta_1 * X_1 + \beta_2 * X_2 + \cdots + \beta_n * X_n + \varepsilon_i$$

Buscamos generar un modelo que explique la varianza del *vector objetivo* en función de la serie de atributos existentes en el conjunto de datos. El objetivo del algoritmo es generar estimaciones con respecto al intercepto (β_0, que representa el punto de partida de la función lineal) y de la pendiente (β_n, que representa la contribución de X en G_i cuando X cambia en una unidad).

ii. Máquina de Soporte Vectorial regresiva

Consiste en una aplicación específica de la máquina de soporte vectorial clasificatoria, en la que se asocian en forma no lineal los datos de entrenamiento a un espacio dimensional superior, sobre el cual se aplica una regresión lineal.

iii. Árboles de Decisión regresivos

La versión regresiva del algoritmo tiene el mismo funcionamiento de la clasificatoria, con la salvedad que termina asignado el valor de la subdivisión final de la muestra.

iv. Bosques Aleatorios regresivos

En su versión regresiva, la votación es reemplazada por el promedio de los valores obtenidos por cada árbol.

v. Potenciaciones regresivas Adaptativa, del Gradiente, y Extrema del Gradiente

Las versiones regresivas de estos modelos solo calculan el valor esperado, mejorando la predicción en cada iteración.

6. CREACIÓN DEL MODELO DE MACHINE LEARNING

Podemos construir modelos que utilicen una serie de datos asociados a una factura y el comportamiento de los emisores y receptores de dicha factura, para determinar si debe o no ser clasificada como fraudulenta, o si se trata de un contribuyentes subdeclarantes.

Utilizaremos las siguientes métricas de evaluación y comparación del rendimiento de estos modelos:

i. *Accuracy:* Mide la exactitud de las predicciones. Se calcula dividiendo el número de predicciones correctas por el total de predicciones realizadas:

(Verdaderos Positivos + Verdaderos Negativos) / (Total Positivos + Total Negativos)

Pierde fuerza cuando en la data de entrenamiento las categorías del vector objetivo no están distribuidas en forma balanceada.

ii. *Precisión:* Mide la precisión de las predicciones. Se calcula dividiendo los resultados positivos verdaderos sobre la suma de los resultados verdaderos positivos y falsos positivos:

Verdaderos Positivos / (Verdaderos Positivos + Falsos Positivos)

iii. *Recall:* Mide el porcentaje de resultados correctamente predichos por categoría. Se calcula dividiendo los resultados positivos verdaderos sobre la suma de verdaderos positivos y falsos negativos:

Verdaderos Positivos / (Verdaderos Positivos + Falsos Negativos)

iv. *Fl-score:* Determina una medida armónica entre Precisión y Recall. Se calcula multiplicando por dos la precisión y el Recall, lo que se divide por la suma entre precisión y Recall:

((Precisión * Recall) * 2) / (Precisión + Recall)

v. Área Bajo la Curva *Receiver Operating Characteristic (AUC):* Muestra la tasa de verdaderos positivos contrastada con la tasa de falsos positivos. Permite visualizar gráficamente cuan mejor es el modelo para predecir resultados positivos correctos, sobre un modelo con probabilidad uniforme del cincuenta por ciento de predecir un resultado para 2 categorías.

En base a estas métricas buscaremos el modelo con el mejor y más balanceado f1-score, y la mayor AUC posible, lo serializaremos y exportaremos para su posterior uso.

6.1. Casos de uso: detección de facturas fraudulentas

Considerando que las facturas falsas constituyen una de las principales formas de defraudación del erario, el uso del Machine Learning y reglas cobre el Big Data es inminente e imprescindible. Es así como se deben utilizar modelos clasificatorios, y en base a los datos asociados a las facturas y el comportamiento de los emisores y receptores de dichas facturas podemos construir y entrenar modelos para predecir si una factura es o no fraudulenta. Complementando dichos datos con información adicional de los contribuyentes, como por ejemplo sus declaraciones de renta, podemos construir modelos que predigan si las declaraciones son o no fraudulentas.

7. CONCLUSIONES

La experiencia ha demostrado que no existe un modelo o solución óptima y única para todos los tipos de problemas, inclusive para dos problemas idénticos, pero con distinto conjunto de datos, razón por la que se hace necesario construir diversos modelos para posteriormente evaluar sus resultados y escoger de entre ellos el más adecuado. Para este caso concreto y por tratarse de un problema de clasificación (fraudulenta o no fraudulenta), los modelos a utilizar y comparar son los siguientes:

Tanto la Regresión Logística como los modelos clasificatorios de Bosques Aleatorios y las tres Potenciaciones, presentan la ventaja de que es posible obtener de ellos los valores regresivos contrastados con los *puntajes de corte* que finalmente determinan la clasificación realizada.

No obstante, la existencia de los modelos y la capacidad de procesamiento, su uso para la detección de facturas con un perfil de alto riesgo de ser falsas deberá ser implementado combinando la experiencia del personal de la administración tributaria y la de expertos en ciencia de datos. Lo anterior debido a que sin la vista experta el modelo no podría ser entrenado para buscar los patrones deseados. Y luego, la acción de control asociada al tratamiento del riesgo dependerá de las capacidades propias de la organización, como el capital humano, las competencias técnicas, disponibilidad de la información.

Hace 15 años existían restricciones presupuestarias para abordar este tema, el almacenamiento y el Hardware era costosos, sin embargo, hoy esto es absolutamente accesible.

Los datos que las AT tienen, y que son cruciales en el uso de los modelos y su entrenamiento, son:

Identificación, socios, sociedades, aportes de capital, declaraciones de impuestos indirectos y directos, facturas, notas de crédito y débito, documentos de transporte.

Con lo anterior se perfilan:

i. Relaciones de cliente proveedor, su frecuencia y magnitud.
ii. Relaciones de Socios y sociedades y las manifestaciones de riqueza.
iii. Relación de ingresos y egresos (por contribuyente y en cada relación)
iv. Impacto en resultados mensuales / anuales de facturas de riesgo

Según lo anterior, con una arquitectura adecuada, se podrán complementar los análisis con la información de las declaraciones de impuestos mensuales y últimos pagos de estos, para definir si una factura de proveedor es de riesgo de ser falsa, lo que se puede determinar a los segundos de que esta sea recibida por la AT. Por ejemplo, identificar una factura de monto relevante para el comprador y cuyo proveedor es no habitual.

El siguiente desafío, será sin duda, la explotación de los datos contenidos en los detalles de cada factura, donde no solo se podría aproximar un control de inventario (en empresas productivas, si no que también un análisis de servicios prestados, seguimiento de ordenes de compra, plazos de pago, etc. Todos factores relevantes al momento de definir un perfil de riesgo de una operación falsa,

Las Administraciones Tributarias se han destacado por ser vanguardistas en el Big Data del sector público. Sin duda, serán, de igual forma, las más innovadoras en el

uso del Machine Learning para el análisis del comportamiento de sus contribuyentes, donde la aplicación directa del análisis de datos de compra y venta (cruce de IVA[4]) será solo una parte del análisis general del Cumplimiento Tributario.

8. BIBLIOGRAFÍA

8.1 Referencias Machine Learning

(1) Explicación general de Algoritmos de Machine Learning:

Murphy, K. 2012. Machine Learning: A Probabilistic Perspective. Cambridge, MA: MIT Press.

Gollapudi, S. 2016. Practical Machine Learning

Kubat, M. 2017. An Introduction to Machine Learning, Second Edition. Springer

(2) Regresión Logística y Lineal:

Geknab, A., & HIll, J. 2006. Data analysis using regression and multilevel/ hierarchical Models. Analytical Methods for Social Research, New York: Cambridge University Press.

Kuhn, M., & Johnson, K. 2013. Applied predictive modeling (vol. 26). New York: Springer.

(3) Bayes Ingenuo Multinomial:

Ng, A; Jordan, M. 2001. On Discriminative vs. Generative classifier: A comparison of logistic regression and naïve Bayes.

Hand, D. J. & Yu, K. 2010. Idiot's Bayes: Not so Stupid After all?. International Statistical Review.

Stone, K. 2013. Bayes Rule: A tutorial introduction to Bayesian Analysis. Sebtel Press. Ch1: An Introduction to Bayesian Rule.

(4) Máquinas de Soporte Vectorial:

Cortes, C.; Vapnik, V. 1995, Support-Vector Networks. Machine Learning 10. pp: 273-297.

4 Si se declaró lo que se vendió/compró y si coincide con lo que dicen los terceros involucrados.

Hsu, Chih-Wei & Chih-Jen Lin. 2002. A comparison of methods for multiclass support vector classification. 1-16.

Hastie, T.; Tibshirani, R.; Friedman, J. 2008. The Elements of Statistical Learning. Data Mining, Inference, and Prediction. New York: Springer.

Murphy, K. 2012. Machine Learning: A probabilistic Perspective. Cambridge, MA: MIT Press. Ch14: Kernels. 14.4: The kernel Trick. 14.5: Support Vector Machines.

(5) Árboles de decisión:

Breiman, L.; Friedman J. H.; Olshen, R. A.; & Stone, C. I. 1984. Classification and regression trees. Belmont, California: Wadsworth Press.

Hastie, T.; Tibshirani, R.; Friedman, J. 2008. The Elements of Statistical Learning. Data Mining, Inference, and Prediction. New York: Springer.

Murphy, K. 2012. Machine Learning: A probabilistic Perspective. Cambridge, MA: MIT Press.

Shalev-Shwartx, S. & Ben-David, S. 2014. Understanding Machine Learning: From Theory to Algorithms, Ch13: Decision Trees.

(6) Bosques Aleatorios:

Breiman, Leo. 2001. Random Forest.

Hastie, T.; Tibshirani, R.; Friedman, J. 2008. The Elements of Statistical Learning. Data Mining, Inference, and Prediction. New York: Springer.

Zhou, 2012. Ensembles Methods, Foundations and Algorithms.

(7) Potenciación Adaptativa, Potenciación del Gradiente, Potenciación Extrema del Gradiente:

Wolpert, D. & MacReady. 1997. No free lunch theorems for optimization. IEEE transactions on evolutionary computation 1 (1), 67-82

Mason, L.; Baxter, J.; Bartlett, P. L.; Frean, Marcus. 1999. Boosting Algorithms as Gradient Descent.

Hastie, T.; Tibshirani, R.; Friedman, J. 2008. The Elements of Statistical Learning. Data Mining, Inference, and Prediction. New York: Springer.

Murphy, K. 2012. Machine Learning: A probabilistic Perspective.

Cambridge, MA: MIT Press. Ch16: Adaptative basis function models.

Schapire, R. E. / Freud, Y. 2012. Boosting: Foundations and algorithms. MIT press.

Zhou, 2012. Ensembles Methods, Foundations and Algorithms.

Chen, T. and Guestrin, C. 2016. Xgboost: A scalable tree boosting system. In Proceedings of the 22Nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, pages 785-794, New York, NY, USA. ACM.

(8) Métricas de desempeño de los modelos:

Murphy, K. 2012. Machine Learning: A probabilistic Perspective. Cambridge, MA: MIT Press. Ch3 (Validación Cruzada) y Ch11 (Métricas de desempeño en clasificación).

Raschka, S. 2018. Model Evaluation, Model Selection, and Algorithm Selection in Machine Learning. University of Wisconsin-Madison, Department of Statistics

8.2 Referencias Big Data

(1) Doug Laney, "3D Data Management: Controlling Data Volume, Velocity, and Variety"", Gartner, file No. 949. 6 February 2001, http://blogs.gartner.com/doug-laney/files/2012/01/ad949-3D-Data-Management-Controlling-Data-Volume-Velocity- and-Variety.pdf

(2) "5C Architecture, Introduced by IMS Center for Cyber-Physical Systems in Manufacturing". Imscenter.net. Archivado desde el original el 27 de mayo de 2016.

(3) "Apache Hadoop, open-source software for reliable, scalable, distributed computing", https://hadoop.apache.org/

(4) "Apache Spark, Lightning-fast unified analytics engine", https://spark.apache.org/

(5) "Apache Flink, Stateful Computations over Data Streams", https://flink.apache.org/