Aplicación de Regresión con Vectores de Soporte en un Sistema Recomendador de Actividades Sociales

Julieta Ríos[1], Gonzalo Ulla[1] y Agustín Borello Gianni[1]

[1]Inteligencia Artificial, Departamento de Ingeniería en Sistemas de Información,
Universidad Tecnológica Nacional, Facultad Regional Córdoba,
Maestro M. López esq. Cruz Roja Argentina, Córdoba, Argentina
{julirios299, gonzaulla, agunstinborello21}@gmail.com

Resumen. Este trabajo desarrolla la aplicación de una técnica de aprendizaje automático, denominada Regresión con Vectores de Soporte en un Sistema Recomendador. Este último forma parte de una plataforma online con fines sociales que vincula organizaciones sin fines de lucro con voluntarios y con empresas. Dicha plataforma, denominada Helpo, brinda soporte a la planificación táctica de las actividades sociales gestionadas por organizaciones del tercer sector. Para ello, emplea el algoritmo SVR de la librería Scikit-learn, escrita en Python, en pos de recomendar a las organizaciones cuándo resulta conveniente realizar sus eventos y campañas. El algoritmo mencionado utiliza datos tanto de la organización y actividad a registrar, como del resto de entidades existentes en la plataforma, y recurre a técnicas complementarias como Feature Scaling y Grid Search. De esta forma, el presente trabajo responde a una necesidad social de forma innovadora: maximizando la probabilidad de que actividades sociales tengan éxito al sugerir cuándo ejecutarlas mediante un algoritmo de aprendizaje automático.

Palabras Clave: Regresión con Vectores de Soporte, SVR, Máquina de Soporte Vectorial, SVM, Inteligencia Artificial, Machine Learning, Aprendizaje Automático, actividades sociales, tercer sector.

1 Introducción

La inversión mundial en tecnologías de Inteligencia Artificial (IA) aumentó 59,1% en 2017 con respecto al año anterior, según un estudio de Accenture referido por La Nación [1]. Argentina no es la excepción: por lo menos desde 2014 se evidencian experiencias en el uso de IA que aplican estas tecnologías en instituciones del sector público, en empresas y en organizaciones del tercer sector [2]. Esto demuestra que el creciente interés por esta disciplina no es de naturaleza exclusivamente académica, sino también profesional y social.

A su vez, la cantidad de datos que manipulan actualmente las organizaciones de todo el mundo crece de manera exponencial. Analizar estos datos, caracterizados por su alta complejidad y diversidad, permite obtener información valiosa de ellos que coloca, a quien la posee, en una posición de ventaja competitiva.

Es de esperar que, en función de lo mencionado, la demanda de soluciones de software que apliquen IA crezca. En efecto, analistas predicen que prácticamente todos los nuevos productos y servicios de software implementarán tecnologías de Inteligencia Artificial para 2020 [3].

Precisamente, en este trabajo se describirá un caso de aplicación en el cual se ha implementado una técnica propia de dicha disciplina, basada en aprendizaje automático, como parte del desarrollo de un sistema web y móvil orientado a vincular organizaciones sin fines de lucro con voluntarios y con empresas. Este software, denominado "Helpo", adopta un modelo de plataforma online multilateral y busca propiciar la realización efectiva de actividades sociales relacionando los tres actores enumerados.

La formulación del problema abordado por el presente trabajo de investigación se establece de la siguiente manera: ¿cómo se puede aplicar una técnica de aprendizaje automático en una plataforma online con fines sociales?

Se espera, de la solución alcanzada por esta investigación, que se pueda responder al problema planteado, aplicando, de forma práctica, un algoritmo de aprendizaje automático. En pos de cumplir con este objetivo, se comenzará desarrollando el fundamento teórico necesario que sustenta todo este trabajo. Luego, se analizará el problema a abordar y se explicará la solución dada al mismo. Por último, se estudiará el comportamiento de esta solución y trabajos relacionados al tema de investigación, concluyendo al respecto.

2 Fundamento Teórico

El aprendizaje automático, conocido en inglés como Machine Learning [4], es una disciplina del ámbito de la Inteligencia Artificial cuya finalidad es desarrollar técnicas para permitir que los sistemas aprendan y resuelvan problemas cotidianos por sí mismos. Aprender, por un lado, implica aumentar el conocimiento y mejorar las capacidades y habilidades de actuación en un entorno, mediante la identificación de patrones de comportamiento en millones de datos. Hacerlo de forma automática, por otra parte, hace referencia a la mejora de los sistemas en forma autónoma a lo largo del tiempo.

En este contexto, un sistema que aprende de forma automática es un artefacto (o conjunto de algoritmos) que, para resolver problemas, toma decisiones basadas en la experiencia acumulada para mejorar su actuación [5].

Para que estos sistemas sean capaces de resolver problemas, deben hacerlo a partir de la selección y adaptación del conocimiento que van adquiriendo [6]. En la fase de selección, el sistema elige las características más relevantes de un objeto y las compara con otras conocidas (si existen) a través de algún método de cotejamiento. En la fase de adaptación, el sistema amolda su modelo de aquel objeto según el resultado del cotejamiento.

Según el tipo de selección y adaptación que un sistema realiza sobre la información disponible es factible identificar diversos paradigmas del aprendizaje automático [7]:

- Aprendizaje supervisado: el algoritmo produce una función que establece una correspondencia entre las entradas y las salidas que se desean del sistema. Los datos

de entrada se encuentran etiquetados, es decir, se conocen los resultados esperados, los cuales se utilizan para corregir las salidas del sistema.

- Aprendizaje no supervisado: en estos algoritmos, el proceso de modelado se lleva a cabo sobre un conjunto de datos sin etiquetar. Al no conocer los resultados deseados, se debe deducir y descubrir la estructura interna presente en los datos de entrada.
- Aprendizaje semi-supervisado: este tipo de algoritmo combina los dos anteriores, generando una función deseada o clasificador a partir de datos tanto etiquetados como no etiquetados.
- Aprendizaje por refuerzo: el algoritmo aprende observando aquello que lo rodea. Su entrada es la retroalimentación que obtiene del mundo exterior en respuesta a sus acciones.

Centrando la teoría en algoritmos de aprendizaje supervisado, se hallan, dentro de dicha clasificación, las Máquinas de Soporte Vectorial [8] (SVM, del inglés Support Vector Machines). Las SVMs buscan seleccionar un hiperplano de separación que equidista de los ejemplos (muestras o vectores de entrada) más cercanos de cada clase para conseguir lo que se denomina margen máximo a cada lado del hiperplano. Al momento de definir este último, sólo se consideran los ejemplos de entrenamiento de cada clase que caen justo en la frontera de dichos márgenes. Esos ejemplos de entrenamiento reciben el nombre de vectores de soporte. Así, un algoritmo de SVM construye un modelo capaz de predecir si un punto nuevo pertenece a una u otra clase.

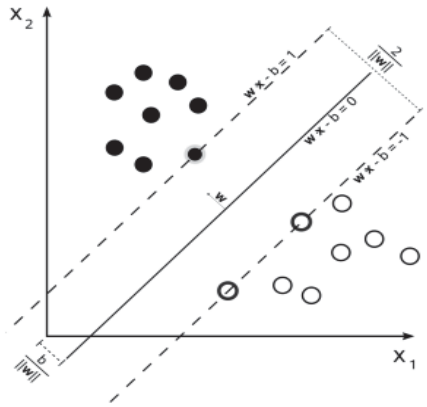

Fig. 1. Ejemplo de dos clases divididas por un hiperplano equidistante.

Las Máquinas de Vectores Soporte adaptadas para resolver problemas de regresión se conocen por el acrónimo SVR [10] (del inglés Support Vector Regression). Así, dado un conjunto de ejemplos de entrenamiento:

$$S = \{(x_1, y_1), \dots, (x_n, y_n)\}$$

en el que se asume que todos los valores y_i de todos los ejemplos de S se pueden ajustar (o cuasi-ajustar) mediante una función lineal, el objetivo de la tarea de regresión

es encontrar los parámetros $w = (w_1, \dots, w_n)$ que permitan definir dicha función lineal, de la forma:

$$f(x) = (w_1 x_1 + \cdots + w_d x_d) + b$$

En el caso de que los ejemplos no puedan ajustarse por una función lineal, se recurre a la metodología denominada Kernelización [9, 10]. Frente a este concepto, los ejemplos pertenecientes al espacio original de entradas se transforman en un nuevo espacio en el que sí es posible ajustar los ejemplos transformados mediante un regresor lineal. El tipo de transformación dependerá del tipo de función kernel utilizado, pudiendo ser Polinomial-homogénea, Perceptrón, Sigmoidea, Función de Base Radial (RBF), entre otras.

3 Descripción del problema resuelto

Helpo surge como una plataforma online orientada a vincular organizaciones sin fines de lucro con voluntarios y con empresas mediante un sistema web y móvil. Su objetivo es incrementar el flujo de recursos de estas organizaciones, brindando oportunidades para que voluntarios y empresas colaboren en actividades sociales, ya sea donando o participando de eventos y campañas. Así, Helpo da respuesta, de forma innovadora, a una necesidad vigente en la actualidad.

En este contexto, surge la necesidad de implementar algoritmos de Machine Learning que permitan a las organizaciones sin fines de lucro planificar sus actividades. Es decir, conocer en qué mes del año es conveniente organizar un evento o una campaña en base a datos analizados de actividades similares de todas las organizaciones previamente registradas en la plataforma. La sugerencia de organizar una actividad social en determinados meses incrementará la probabilidad de que la misma tenga éxito, lo que significa que la ONG satisfaga un mayor número de necesidades, tanto materiales como de recursos humanos.

4 Explicación de la solución

Para abordar la solución a la necesidad detectada, se hará uso de Scikit-learn [11], una librería de aprendizaje automático de Python, lenguaje de programación utilizado en el desarrollo del sistema Helpo.

El algoritmo SVR [12], presente en la librería, es el empleado en el desarrollo de la presente solución. Dicho algoritmo presenta una implementación del método de aprendizaje automático “Regresión con Vectores de Soporte” y, en base a lo explicado en el fundamento teórico anterior, produce un modelo que depende sólo de un subconjunto de ejemplos de entrenamiento, dado que la función empleada para construir el mismo no se preocupa por los puntos de entrenamiento que se encuentran más allá del margen.

El algoritmo SVR de la librería Scikit-learn toma un conjunto de parámetros de entrada necesarios para producir el modelo adecuadamente, entre los que se encuentran el tipo de kernel empleado -por defecto el kernel RBF o Función de Base Radial [13]-, el coeficiente gamma del kernel seleccionado y el parámetro de penalización C del término del error.

Previo a entrenar los vectores con el algoritmo seleccionado, se pre-procesan los datos haciendo "Feature Scaling" [14] de los mismos, es decir, escalando las características para ubicarlas entre un valor mínimo y máximo dado o para que el valor absoluto máximo de cada característica se ajuste al tamaño de la unidad.

Además, se utiliza una búsqueda de cuadrícula o "Grid Search" [15] para definir adecuadamente los hiper parámetros del algoritmo SVR, previamente mencionados. La búsqueda de cuadrícula genera exhaustivamente candidatos a partir de una cuadrícula de valores de parámetros especificados. Grid Search evalúa todas las combinaciones posibles de valores de parámetros y conserva la mejor combinación para, luego, emplear el algoritmo entrenador con los mejores parámetros.

```
y = pd.DataFrame()
y["pred"] = Mtrain["%Comp"]
training_data = Mtrain.drop(["%Comp"], axis=1)

rmse_error = make_scorer(mean_squared_error, greater_is_better=False)

parameters = {
    'C': [0.8, 0.9, 1],
    'epsilon': [0.04, 0.05, 0.06],
    'gamma': [0.001, 0.003, 0.005, 0.008]
}

svr = GridSearchCV(SVR(), cv=3, param_grid=parameters, scoring=rmse_error)

svr.fit(training_data, y)
print(svr.best_estimator_)
```

```
SVR(C=1, cache_size=200, coef0=0.0, degree=3, epsilon=0.06, gamma=0.008,
  kernel='rbf', max_iter=-1, shrinking=True, tol=0.001, verbose=False)
```

Fig. 2. Porción de código en la que se aplica el método "Grid Search", obteniendo los mejores valores posibles para C, épsilon y gamma, y se entrena el modelo.

A continuación, se entrena el modelo usando dos conjuntos -de entrenamiento y de prueba-. Para el presente trabajo, los conjuntos mencionados surgieron de datos consistentes y reales brindados por tres ONGs relevadas. Estas organizaciones proporcionaron información histórica acerca de sus actividades sociales y, a partir de ellas, el equipo de trabajo extrapoló los datos para generar nueva información, con la finalidad de lograr una base de datos suficiente y completa para entrenar y probar el modelo.

Finalmente, se evalúa el error del modelo con el estadígrafo RMSE [16] (del inglés Root Mean Square Error) que compara los valores predichos con los valores observados reales.

Las características que utilizará el algoritmo para ajustar el modelo son:

- Mes de las actividades previas registradas.
- Porcentaje promedio de completitud de necesidades materiales de las actividades previas registradas de la organización que desea planificar su nueva actividad.
- Porcentaje promedio de completitud de necesidades materiales de actividades previas registradas por otras organizaciones del mismo rubro.
- Porcentaje promedio de completitud de necesidades materiales de actividades previas registradas del mismo rubro que la nueva actividad que se desea planificar.
- Porcentaje promedio de completitud de necesidades de voluntarios de las actividades previas registradas de la organización que desea planificar su nueva actividad.

- Porcentaje promedio de completitud de necesidades de voluntarios de actividades previas registradas por otras organizaciones del mismo rubro.
- Porcentaje promedio de completitud de necesidades de voluntarios de actividades previas registradas del mismo rubro que la nueva actividad que se desea planificar.
- Suscripciones de la organización que desea planificar la nueva actividad.
- Suscripciones de todas las organizaciones del mismo rubro que la organización que desea planificar la nueva actividad.
- Promedio de visitas por actividad de la organización.
- Promedio de visitas por actividad de organizaciones del mismo rubro que la organización que desea planificar la actividad.
- Promedio de visitas por actividad de actividades del mismo rubro que la que se desea planificar.
- N variables booleanas que representan si la necesidad material se va a incluir en la actividad.
- N variables booleanas que representan si la necesidad de voluntario se va a incluir en la actividad.

Dando respuesta a la necesidad del proyecto Helpo, la salida del algoritmo, es decir, la predicción, es el porcentaje de completitud de las colaboraciones y participaciones respecto a las necesidades que se solicitarán en la nueva actividad. Para conseguir la sugerencia del mejor mes, se realizarán doce predicciones (con meses distintos) y aquel mes con el que se obtenga el mayor porcentaje de completitud será el recomendado.

5 Análisis del comportamiento de la solución

En pos de demostrar e ilustrar la solución alcanzada, se plantea a continuación un ejemplo particular.

La organización sin fines de lucro de fantasía “Mundo Feliz” desea planificar cuándo realizar el evento “Maratón solidaria”. Para ello, deberá ingresar en el sistema Helpo los siguientes datos sobre dicha actividad futura:

- Tipo de actividad: en este caso un “Evento”.
- Rubro de la actividad: en este ejemplo, “Deportivo”.
- Ubicación de la actividad: en este caso “Plaza de la Intendencia”.
- Necesidades materiales que solicitar: particularmente para este evento, “Ropa” y “Alimentos”.
- Funciones de voluntarios a cubrir: en el ejemplo, “Policías” y “Fotógrafos”.

Teniendo en cuenta los datos ingresados y demás datos registrados previamente en el sistema tanto de la organización “Mundo Feliz” como de otras organizaciones y sus actividades, Helpo le recomendará a esta organización el mes adecuado para llevar a cabo el evento “Maratón solidaria”. Para ello, se recurrirá al algoritmo SVR de la librería Scikit-learn tal como se explicó anteriormente.

Luego de que la organización sin fines de lucro ingrese los datos de la actividad que desea planificar, el sistema Helpo, en primer lugar, recuperará el modelo entrenado almacenado, previamente, en Amazon Simple Storage Service (Amazon S3) [17].

El modelo en cuestión fue almacenado luego de reiterados ciclos de selección y adaptación. Más precisamente y como se mencionó anteriormente, se utilizó el estadígrafo RMSE para evaluar los errores obtenidos en los conjuntos de entrenamiento y prueba y obtener el modelo que mejor ajustara a la realidad, es decir, el que proporcionara el mínimo error posible. El modelo resultante arrojó un RMSE en datos de entrenamiento igual a 0,0063 y un RMSE en datos de prueba igual a 0,0118.

```
training_error = mean_squared_error(Mtrain['%Comp'],
                                    svr.predict(Mtrain.loc[:, M.columns != '%Comp']))
test_error = mean_squared_error(Mtest['%Comp'],
                                    svr.predict(Mtest.loc[:, Mtest.columns != '%Comp']))
print('RSME en datos de entrenamiento: ' + str(training_error))
print('RSME en datos de prueba: ' + str(test_error))
```

```
RSME en datos de entrenamiento: 0.006286952394657705
RSME en datos de prueba: 0.01182923886669542
```

Fig. 3. Porción de código en la que se visualizan los errores obtenidos en los conjuntos empleados para ajustar el modelo.

Graficando el funcionamiento del modelo, tanto con el conjunto de entrenamiento como con el conjunto de prueba, se observa que el resultado es realmente óptimo. Los errores son relativamente bajos y se visualiza que las predicciones con el conjunto de prueba ajustan de forma muy cercana a la realidad.

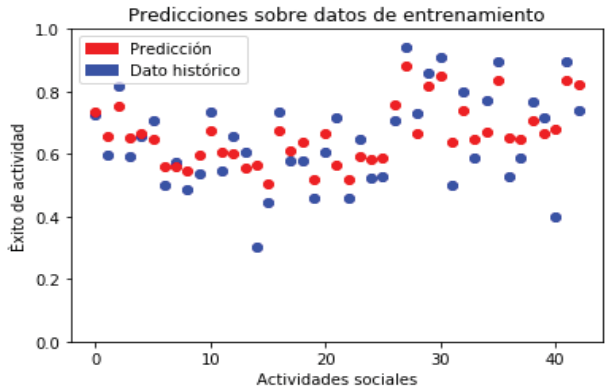

Fig. 4. Gráfico confeccionado con las predicciones sobre el conjunto de entrenamiento.

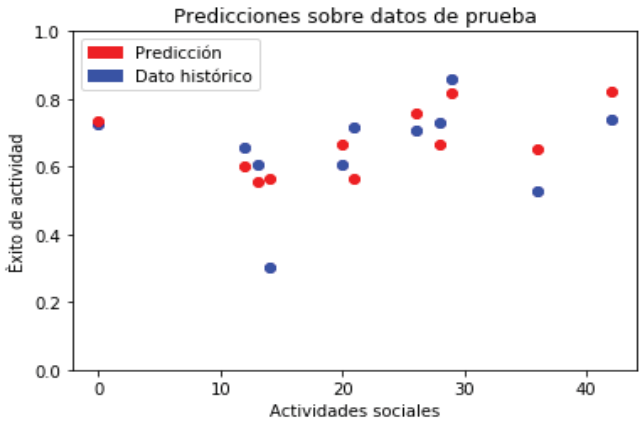

Fig. 5. Gráfico confeccionado con las predicciones sobre el conjunto de prueba.

Retomando el caso planteado, una vez levantado el modelo se procede a evaluar y calcular el porcentaje de completitud por mes de las colaboraciones y participaciones respecto a las necesidades que se solicitan en la actividad "Maratón Solidaria". Realizando doce predicciones (con distintos meses), se recomienda el mes que obtenga el mayor porcentaje de completitud.

Finalmente, si la organización en cuestión opta por registrar su evento en el mes recomendado por Helpo, esto maximizará la probabilidad de cubrir sus necesidades materiales y de recursos humanos.

6 Trabajos relacionados

En primer lugar, en cuanto al sistema Helpo como producto de software, cabe destacar que no existen, a nivel local, plataformas con características similares. Tomando la totalidad de la República Argentina, es posible identificar soluciones con objetivos compartidos, pero ninguna con foco en Córdoba ni con los mismos actores y procesos de negocio involucrados.

Con respecto a líneas de investigación que aborden temáticas vinculadas a Inteligencia Artificial, Machine Learning y SVR, se han podido relevar trabajos de carácter eminentemente teórico oriundos tanto de Argentina como de la región latinoamericana [8, 9]. Sin embargo, no se han hallado aplicaciones prácticas relacionadas a estas disciplinas.

Ampliando el alcance de búsqueda a un nivel mundial, existen trabajos que denotan aplicaciones de estos conceptos con fines sociales -por ejemplo, para predecir desastres naturales [18]- o con fines de lucro -pronosticar series de tiempo en el rubro financiero [19] o estimar tiempos de viaje [20]-.

Por último, en relación a la utilización de IA y aprendizaje automático en organizaciones del tercer sector, es oportuno señalar que, si bien Argentina es el país de Latinoamérica con las startups de Inteligencia Artificial que más ingresos generan [21],

estos emprendimientos aún no arriban a dicho sector, lo cual representa una clara oportunidad.

7 Conclusión y Trabajos Futuros

Es cierto que, a nivel local, disciplinas y técnicas como Inteligencia Artificial y aprendizaje automático aún se encuentran en una fase de desarrollo incipiente, a pesar de estar en boga últimamente. Aun así, resulta imposible negar el elevado crecimiento que estos campos de las ciencias de la computación enfrentan, inclusive en Argentina.

El caso particular de aplicación detallado en este trabajo articula conocimientos teóricos propios de la disciplina de Inteligencia Artificial con una solución de software orientada a reunir organizaciones sin fines de lucro, voluntarios y empresas. De esta forma, se ha respondido a la pregunta de investigación propuesta inicialmente, detallando cómo aplicar una técnica de aprendizaje automático en una plataforma online con fines sociales. Para ello, se ha empleado el algoritmo SVR de la librería Scikit-learn, escrita en el lenguaje de programación Python.

El valor de negocio que aporta la utilización del algoritmo mencionado radica en recomendar cuándo resulta conveniente registrar una actividad social. Gracias a este trabajo Helpo ahora responde a una necesidad adicional de sus usuarios: planificación táctica a medida.

Más allá del ámbito académico en el cual fue concebida esta solución, la misma responde a una necesidad social de forma innovadora: aplicando algoritmos de aprendizaje automático para maximizar la eficacia de actividades orientadas al bien común.

Referencias

1. Goldschmidt, O.: Innovación made in Argentina: la inteligencia artificial pide pista en el mercado local, *La Nación*, https://www.lanacion.com.ar/2114594-innovacion-made-in-argentina-la-inteligencia-artificial-pide-pista-en-el-mercado-local, último acceso: 1/05/2019.
2. Ingrassia, V.: Avanza el uso de inteligencia artificial en el sector público, privado y ONG argentinas, *Infobae*, https://www.infobae.com/tendencias/innovacion/2018/04/01/avanza-el-uso-de-inteligencia-artificial-en-el-sector-publico-privado-y-ong-argentinas/, último acceso: 1/05/2019.
3. Moore, S.: Gartner says AI technologies will be in almost every new software product by 2020, Gartner, https://www.gartner.com/en/newsroom/press-releases/2017-07-18-gartner-says-ai-technologies-will-be-in-almost-every-new-software-product-by-2020, último acceso: 1/05/2019.
4. González, A.: ¿Qué es Machine Learning?, *CleverData*, https://cleverdata.io/que-es-machine-learning-big-data/, último acceso: 1/05/2019.
5. Aggarwal, C.: Data Classification: Algorithms and Applications, *CRC Press*, (2015).
6. Moreno, A., Armengol, E., Béjar, J., Belanche, L., Cortés, U., Gavaldá, R., Gimeno, J.M., López, B., Martín, M., Sánchez, M.: Aprendizaje Automático, Edicions UPC, (1994).
7. Russo, C.: Tratamiento masivo de datos utilizando técnicas de Machine Learning, *WICC, Entre Ríos* (2016).

8. Betancourt, G.: Las máquinas de soporte vectorial (SVMs), *Scientia et Technica*, Año XI, No 27, (2005).
9. Carmona Suárez, E.: Tutorial sobre Máquinas de Vectores Soporte (SVM), *Dpto. de Inteligencia Artificial, ETS de Ingeniería Informática, UNED,* (2014), http://www.ia.uned.es/~ejcarmona/publicaciones/[2013-Carmona]%20SVM.pdf, último acceso: 3/05/2019.
10. Kernelización: https://es.wikipedia.org/wiki/M%C3%A1quinas_de_vectores_de_soporte#SVR._Regresi%C3%B3n, último acceso: 3/05/2019.
11. Muller, A., Guido, S.: Introduction to Machine Learning with Python, *O'Reilly*, (2016).
12. AlgoritmoSVR: http://scikit-learn.org/stable/modules/generated/sklearn.svm.SVR.html#sklearn.svm.SVR, último acceso: 3/05/2019.
13. Kernel RBF: https://en.wikipedia.org/wiki/Radial_basis_function_kernel, último acceso: 3/05/2019.
14. Feature Scaling: http://scikit-learn.org/stable/modules/preprocessing.html, último acceso: 3/05/2019.
15. Grid Search: http://scikit-learn.org/stable/modules/grid_search.html, último acceso: 3/05/2019.
16. RMSE: https://en.wikipedia.org/wiki/Root-mean-square_deviation, último acceso: 3/05/2019.
17. Amazon Simple Storage Service: https://aws.amazon.com/es/s3/, último acceso: 3/05/2019.
18. Tien Bui, D.: Spatial prediction of rainfall-induced landslides for the Lao Cai area, Springer, Berlin Heidelberg (2016).
19. Tay, F., Cao, L.: Application of support vector machines in financial time series forecasting, *Department of Mechanical Engineering, National University of Singapore* (2001).
20. Wu, C.: Travel-time prediction with support vector regression, *IEEE Transactions on Intelligent Transportation Systems,* Volume: 5, Issue: 4, pp. 276-281, (2004).
21. Argentina, el país con las startups de Inteligencia Artificial más grandes en la región, Ámbito Biz, http://www.ambito.com/935630-argentina-el-pais-con-las-startups-de-inteligencia-artificial-mas-grandes-en-la-region, último acceso: 3/05/2019.