Algoritmos de Machine Learning para predecir la eficienica de un hospital completo

Nicolás Anabalón Romero.
Departamento Ingenieria Civil Industrial
Universidad de Concepción.

7 de julio de 2019

Resumen

En general los algoritmos de machine learning son idoneos para poder predecir o clasificar datos, en este estudio se puede ver como estos ayudan a entender el funcionamiento de un hospital completo a traves de la interpretación de la Overall Equipment Effectiveness (OEE), que representa que tan bien funciona el equipamiento en el hospital, así tambien se logro demostrar cuan eficiente son los tres algoritmos aplicados y como se comportan los datos con respecto a las prediciones con cada uno de ellos en sus partes de entrenamiento y test, considerando ademas el tipo de algoritmo y los mejores hyperparametros para cada uno.

1. Introducción

Muchas veces se ha estudiado el funcionamiebto de un hospital para poder mejorar y hacer mas eficientes sus procesos, pero pocas veces se a considerado el hospital completo con todas sus areas, por la complejidad que esto trae. Por esta razón la siguiente investigación toma en cuenta la simulación previamente hecha en un hospital, de donde se obtiene los datos para la implementacion de los algoritmos a continuacion tratados. La idea general del trabajo es implementar algunos algoritmos vistos en el estudio realizado por [1]De la Fuente. R. and Smith III. R.(2017), y la informacion general de cual es el funcionamiento del hospital completo ademas de sus implicacias en cada una de las areas se presentaron en el trabajo realizado previamente por [2]Smith III.R. and D. Roberts.S. (2014) y tambien en el estudio en solitario de [3]Smith III.R. (2014) donde se detalla aun mas cada area del hospital y se realiza un regresión para entender mejor las variables que influyen en el funcionamiento. Por otro lado el funcionamiento de los algoritmos de machine learning presentan un aprendizaje de los algoritmos a traves del entrenamiento con datos similares a los que se quiere predecir, donde si se logra un buen ajuste son herramientas muy utiles para tareas que antes solo se lograba con mucho tiempo y esfuerzo como lo son el reconocimiento facil, el analisis de big data o el mapeo de grandes areas de terreno, por esto tambien se utilizan para predecir respuestas esperadas de algun proceso sin incurrir en los altos costos de simulaciones o pruebas fisicas.

2. Hospital

El hospital a tratar como se mensiono anteriormente es el mismo analizado en los estudios de [3]Smith III.R. (2014), donde se puede dividir el hospital en 6 grandes areas, el departamento de emergencias donde llega la mayoria de los pacientes a traves de ambulancia o por sus propios medios, unidad quirúrgica que recibe a los pacientes que estaban calendarizados, salas de medicina al cual llegan los pacientes que manda el departamento de emergencia y casos especiales de admicion directa, salas de cirugia donde se trata a los pacientes que son enviados desde el departamento de emergencia y la unidad quirúrgica y dos servicios auxiliares, que pueden ser ocupados por diferentes areas del hospital, estos son el servicio de radiologia y el servicio de laboratorio, y se considera dos aspectos principales, estos son, el funcionamiento del

hospital con sus distintas areas y la comunidad (pacientes).Entre las variables a considerar se encuentran los ratios de llegada, los tiempos de espera para ser atendidos, cantidad de personas atendidas, cantidad de personas que abandonan el hospital sin ser atendidos, tiempos promedios de atencion, cantidad de camas por sala, y la variable que se quiere predecir es la eficiencia general del equipamiento (Overall Equipment Effectiveness), que se denominara OEE, por su nombre en ingles como se menciona en [3]Smith III.R. (2014), y nos da las metricas para medir el rendimiento de los departamento de emergencia.

3. Modelos

En esta sección se describe con mayor detalle cuales son los algoritmos de machine learning que se ocuparan para analizar los datos de una simulacion previa del hospital antes descrito. Los algoritmos vistos en el estudio en que esta basado este trabajo son cinco: Support Vector Regression, Multi-layer Perceptron, Random Forest Regression, Gradient Boosted Regression Tree, Gaussian Process, que se explican con mayor detalle en [1]De la Fuente. R. and Smith III. R.(2017). Para efecto de este estudio se analizan los 3 primero explicados a continuación.

3.1. Neuronal Network Multi-layer Perceptron (MLP)

Una red neuronal es un conjunto de capas compuestas por nodos, que intenta replicar el funcionaminto del cerebro, donde en cada una de las capas se agrega un nodo extra, para al final de 1 epoch se calcule el error cuadrado medio y a traves de backpropagation se ajusten los estimadores y asi obtener una mejor respuesta, tambien es importante recalcar que dentro de cada nodo se encuentra una funcion de activación y anterior a eso una correspondiente sumatoria de los parametros que vienen de la capa anterior, ya que esta red esta completamente conectada. Dentro de las funciónes de activacion mas utilizadas se encuentran relu o logistic, la cual de todas formas puede variar dependiendo de que datos se esten trabajando y que tipo de algoritmo se quiere implementar, ya que las redes neuronales sirven tanto como para clasificar y regresión como se utiliza en este estudio.

3.2. Support Vector Regression (SVR)

Este algoritmo de aprendizaje supervisado presenta una buena estimación para los datos cuando el set de datos no es muy grande, en este caso se ocupa ya que se tienen solo 500 resultados de la simulación, por otro lado tambien es un buen algoritmo para predecir una tendecia ya que trabaja creando un hiperplano, el cual tiene dos bandas a sus costados para de esta forma considerar la mayor cantidad de datos posible, estas bandas tienen un valor denomindao maximo margen entre una y la otra, y los datos que quedan fuera de estas bandas se consideran el error asociado a los datos, donde la destancia de estos a la banda es denomidado epsilon.

3.3. Random Forest Regressor (RFR)

Este algoritmo esta basado en el algoritmo Random Forest, donde para este algoritmo se tienen muchos arboles que entreagan una respuesta, y calculando el promedio de todos los arboles se logra llegara a una respuesta, dado que este tipo de algoritmos son mejores para clasificación que para regresión existe la posibilidad de overfitting en los datos. Por otro lado es importante destacar que este tipo de algoritmos son bastante utiles para predecir casi cualquier tipo de set de datos, ya que trabaja muy bien con falta de datoso datos atipicos, dado que Random Foresr Regressor calcula el rango para los datos y asi poder dar una mejor presición.

4. Metodología

Para la preparacion de los datos y el preprocesamiento se realizo de la misma manera para los 3 tipos de algoritmos, se parte considerando uns set de datos compuesto por 25 caracteristicas para la matriz de diseño, y una caracteristica independiente a predecir (OEE),estos datos son en total 500 los cuales se dividieron en una parte de entrenamiento y una parte de test, la primera parte consta del 80 porciento de los datos y el test del sobrante, para poder asi validar los resultados obtenidos, estos datos se trabajaron gracias a las librerias pandas, sklearn y numpy, y luego se utilizo matplotlib para su interpretacion en graficos, los datos anteriormente divididos pasan a un proceso de standarizacion a traves de la funcion StandardScaler para suavizar su error y tener mejores resultados, esta estandarizacion solo se hace para la matriz de diseño.
Posteriormente se considera cada uno de los algoritmos, y se busca los mejores hyperparametros para cada uno de ellos a traves de la funcion GridSearchCV la cual es capaz de probar de forma iterativa el modelo con cada uno de los hyperparametros que se establece y ademas hacer el proceso de Cross Validation, de

esta forma entrega los mejores hyperparametros a utillizar y el mejor score encontrado, que representa el R cuadrado del modelo provado, estos hyperparametros a probar en cada modelo se presentan a continuación en las tablas 1,2 y 3:

MLP			
Hyperparametros			
hidden_layer_sizes	50	100	200
activation	relu	logistic	
solver	adam	sgd	

Figura 1: Hyperparametros a evaluar para MLP

SVR				
Hyperparametros				
gamma	scale	auto		
epsilon	0.1	0.2		
Kernel	rbf	linear	poly	sigmoid

Figura 2: Hyperparametros a evaluar para SVR

RFR			
Hyperparametros			
n_stimators	50	100	200
random_state	0	1	2
max_depth	2	3	

Figura 3: Hyperparametros a evaluar para RFR

ya con los hypeparametros a definir se tiene en cuanta que todos los algoritmos son diferentes por lo que la elección de que cambiar depende de cada uno, por esto por ejemplo en el caso de las redes neuronales es importante determinar cual va a ser el numero de capaz ocultas para la predición así tambien como la funcion de activacion, ya que determina la forma en que se transformaran los datos en cada nodo, por otro lado en support vector regression encontrar un buen epsilon y kernel es importante para determinar el algoritmo y finalmente en Random Forest Regressor tener el numero maximo de hojas ayuda a entender cual sera la profundidad del arbol, ya que si no se definiera, este seguiria buscando hasta lleagar al fondo de los datos.
Luego de obtener los mejores parametros que se explican en la sección de resultados, se procede a provar los algoritmos y obtener la correspondiente predición de cada modelo utilizando el set de entrenamiento y de test, por esto se realiza gracias al comando plt, graficos comparando las respuestas predichas y las esperadas (reales).

5. Resultados

En esta sección se dara a conocer cuales fueron los resultados obtenidos luego de aplicar GridSearchCV a los hyperparametros a probar y tambien se da a conocer cuales fueron los valores obtenidos a traves del score del R cuadrado ademas de la superpocición de graficas con respecto a la variable y predicha junto a la variable y real de los datos en el test, primero es importante destacar que el tiempo de proceso de los tres algoritmos es diferente, siendo en redes neuronales y random forest donde mas demoro el proceso en encontrar los hyperparametros debido a la complejidad de estos algoritmos sobre support vector regression, ademas se llego a valores de R cuadrado bastante altos que son los que se muestran a continuación en la tabla 4.

	MPL	SVR	RFR
R^2	0.841	0.965	0.996

Figura 4: Precición de cada modelo

Estos valores son los resultantes de cada algoritmo considerando la mejor combinacion de hyperparametros provados en GridSearchCV, los cuales tambien se muestran a continuacion en las tablas 5, 6 y 7.

MLP	
hidden_layer_sizes	200
activation	logistic
solver	adam

Figura 5: Mejores hyperparametros para MLP

En base a los resultados anteriormente se puede probar con cada uno de los modelos segun su nivel

SVR	
gamma	scale
epsilon	0.1
Kernel	linear

Figura 6: Mejores hyperparametros para SVR

RFR	
n_stimators	200
random_state	2
max_depth	3

Figura 7: Mejores hyperparametros para RFR

de significancia, en este caso con el indicador R cuadrado, este nos indica valores demasiado buenos, considerando Overfitting en los datos sobre todo en los modelos support vector regression y random forest regresor, por lo que es correcto decir que el algorito de redes neuronales entrega uan mejor respuesta, auqnue su R cuadrado sea menor, es probable que sea el unico modelo que no sufrio el proceso de Overfitting, dado ya por su backpropagation o su estructura de nodos.

Aun así se considera realizar la prueba de graficas para los tres modelos y asi poder evidenciar graficamente como se comporta cada predición con respecto a los datos reales que se quiere predecir, primero se puede apreciar los graficos de la red neuronal considerando la predición de los datos de y (OEE) de la muestra de entrenamiento, en el segundo y tercer grafico se puede ver la misma comparación entre los datos de la respuesta predicha con los datos de test del algoritmo SVR y Random Forest con los datos de la respuesta del test real.

En todos los graficos a presentar a continuación los puntos rojos representan la predición de cada caso y los puntos azules demuestran cuales son los datos reales de la base de datos en el test.

Como se puede ver en los graficos es claro el ajuste demasiado preciso del algoritmo RFR, debido a su R cuadrado tan alto, y como los otros dos algoritmos dan una respuesta mas real de como seria una predición.

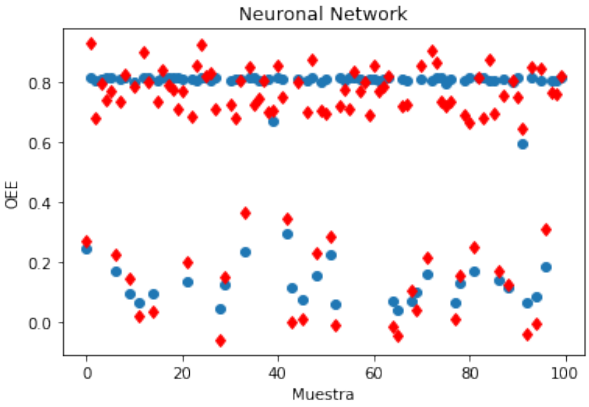

Figura 8: Datos de testeo y respuesta predicha por algoritmo MLP con los mejores parametros obtenidos

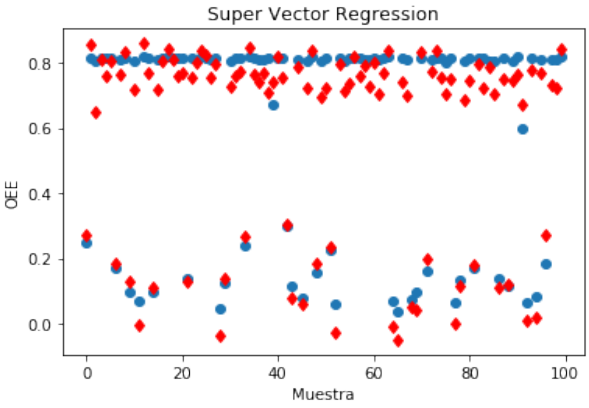

Figura 9: Datos de testeo y respuesta predicha por algoritmo SVR con los mejores parametros obtenidos

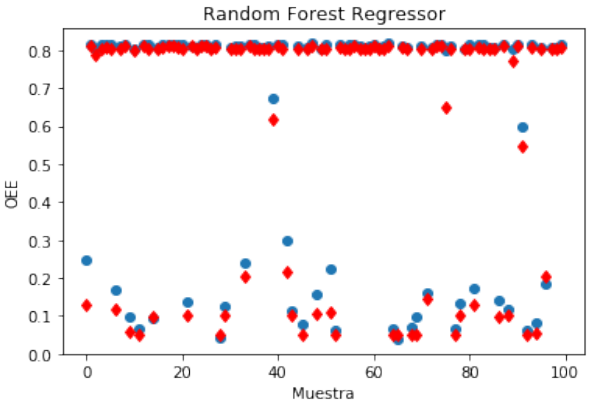

Figura 10: Datos de testeo y respuesta predicha por algoritmo RFR con los mejores parametros obtenidos

6. Discusión

Los datos obtenidos a traves del estudio de los algoritmos para predecir la eficiencia del equipamiento del hospital completo presenta datos no tan favorables, ya que la relacion de los R cuadrado da demasiodo alta para estos modelos considerandose en estado de Overfitting sobre todo en el caso del algoritmo Random Forest Regressor, por esto se puede decir que la mejor opción para este tipo de datos es aplicar la red neuronal ya que sus valores de predición nos indica que su ajuste se compara a casos reales, donde las presiciónes de otros datos tambien estan dentro de un rango aceptable para una solución. Como se pudo ver tambien en resultados mostrados en [1]De la Fuente. R. and Smith III. R.(2017), las redes neuronales tambien se sobre pusieron como la mejor opción a la hora de tomar la decición de que metodo es mejor para ajustar el modelo, solo superado por algoritmos gausianos que no se trataron en este trabajo.
Finalmente se deja como material a tratar el hecho del sobre ajuste y los datos que tienen una moda, siendo muy similares todas las prediciones.

7. Conclusión

Como conclusión dado todos los argumentos de cual de los modelos es mejor, llegando a que este es la red neuronal y descartando los modelos de support vector regression y random forest regressor, debido a su sobre ajuste, se puede decir que los algoritmos de machine learning si son buenos para predecir de buena manera datos como los estudiados en este trabajo, pero la elección de los hyperparametros y estandarización de los datos es fundamental para que se logre un buen ajuste, ya que como se puedo evidenciar, pequeños errores puede traer problemas que sesguen la respuesta dada por los algoritmos.
Por otro lado en base a las diferentes iteraciones realizadas por el algoritmo gridsearchcv entrega los mejores parametros a demas de un score con el mejor R cuadrado, pero este solo es valido si los parametros ingresados son de verdad relevantes con el estudio, ya que cambiar parametros que no cambien la eficacia del modelo no traera mas que tiempo de proceso desperdiciado.
Aun asi se considera un buen estimador para dar los parametros que se quiere imponer al modelo, dado esto tambien existen otro tipo de procesos que ayudan a encontrar estos hyperparametros de una manera mas eficiente, como lo son HyperoptEstimator de hpsklearn y RandomizedSearchCV de sklearn. Se propone mejorar los resultados antes entregados aplicando uno de estos dos ultimos algoritmos de de elección y asi desminuir el sesgo de colocar solo los paramtero que se creen importantes.
Otro aspecto importante a considerar son los datos ocupados,en este caso los datos a predecir que corresponden al OEE, estos datos como se puede ver en los graficos 8,9 y 10 como los puntos azules, de estos graficos es facil reconocer que la gran mayoria de los datos tienen un patron establecido, es decir la gran mayoria de ellos correspone al valor de 0.8, esto tambien puede causar el sobre ajuste de los datos, ya que como el algoritmo aprende que con la mayoria de las combianciones el resultado es similar, es facil decir que sei se le entregan nuevis datos, los resultados sean este número mas probable y alguna variación como se ve en los puntos entre 0 y 0.2.
Finalmente teniendo en cuenta que los algoritmos de machine learning son de aprendizaje supervizado, es claro que una mayor cantidad de datos provee un mejor ajuste de los modelos, como en este caso los datos son 500 no se concidera un número como para dar los mejores resultados, así para obtener los mejores resultados de estos algoritmos lomideal seria entrenarlos con una cantidad de datos mucho mayor, y de esta forma obtener mejores prediciones.

Referencias

[1] De la Fuente. R. and Smith III. R., Metamodeling a System Dynamics Model: A Contemporary Comparison of Methods, Proceedings of the 2017 Winter Simulation Conference (2017)

[2] Smith III.R. and D. Roberts.S., A Simulation Approach to Exploring Whole Hospital System Operational Performance and Efficiencies, Proceedings of the 2014 Industrial and Systems Engineering Research Conference (2014)

[3] Smith III.R.,Sensitivity Analysis for a Whole Hospital System Dynamics Model, Proceedings of the 2014 Winter Simulation Conference (2014)