Interpretabilidad de modelos de caja negra

David Solís Martín

Curso 2022/2023

1 Introducción

Los sistemas de IA incipientes, como los pequeños árboles de decisión o regresiones lineales, eran fácilmente interpretables, pero tenían capacidades limitadas. Sin embargo, durante los últimos años, el notable aumento en el rendimiento de los modelos predictivos (tanto para la clasificación como para la regresión) ha ido acompañado de un aumento en la complejidad del modelo. Esto ha sido a expensas de perder la capacidad de comprensión de las razones detrás de cada predicción en particular. Este tipo de modelo se conoce como modelos de caja *caja negra* [1] debido a la opacidad de su comportamiento. Esta falta de transparencia se convierte en un problema, especialmente cuando las predicciones de un modelo impactan diferentes dimensiones dentro del ámbito humano (como sucede en medicina, derecho, perfilado, conducción autónoma o defensa, entre otros) [2]. También es importante destacar que los modelos opacos son difíciles de depurar, a diferencia de los interpretables, que facilitan la detección de la fuente de sus errores/sesgos y la implementación de una solución [3].

La Inteligencia Artificial Explicable (XAI por sus siglas en inglés) aborda estos problemas proponiendo técnicas de aprendizaje automático (ML) que generan explicaciones de modelos opacos o crean modelos más transparentes. En particular existen técnicas que son aplicadas a posteriori (*post-hoc*) de ser entrenado el modelo. Las técnicas de explicabilidad post-hoc se pueden dividir en técnicas agnósticas al modelo y técnicas específica del modelo. Las técnicas de agnósticas al modelo abarcan aquellas que se pueden aplicar a cualquier modelo de ML, como LIME [5] o SHAP [6], por ejemplo. Mientras que las técnicas específicas del modelo están diseñadas para ciertos modelos. En este trabajo nos centraremos en estas dos técnicas agnósticas.

De la misma manera en que no existe un modelo mejor adaptado para resolver cualquier tarea de ML, no hay un método particular de XAI que proporcione la mejor explicación de cualquier modelo. Se puede encontrar una cantidad significativa de literatura dedicada a innovaciones en modelos interpretables y estrategias de explicabilidad. Sin embargo, cuantificar la corrección

de sus explicaciones sigue siendo un desafío. La mayoría de los esfuerzos de la investigación no tienen como objetivo comparar la calidad de las explicaciones proporcionadas por los métodos de XAI [4] [8], al menos de una manera cuantitativa que permita la comparación de diferentes métodos XAI objetivamente. Se pueden encontrar dos tipos de *indicadores* para la evaluación y comparación de explicaciones: cualitativos y cuantitativos. Los indicadores cuantitativos, que se usarán en este trabajo, están diseñados para medir las características deseables que cualquier método de XAI debería tener. Las métricas aproximan el nivel de logro de cada característica, lo que nos permite medirlas en cualquier método de XAI.

En las próximas secciones se dará una explicación del funcionamiento de los algoritmos LIME y SHAP así como de varias métricas interesantes para llevar a cabo el trabajo.

2 Vecindad de una muestra

Los métodos que se explicarán a continuación suponen que habrá cambios significativos en el rendimiento de un modelo *bien entrenado* si se alteran sus características relevantes. Para llevar a cabo dicho estudio, toman una muestra y realizan perturbaciones en uno o varios de sus atributos. A estas nuevas muestras perturbadas se les denomina vecindad de la muestra original.

Un método habitual de perturbar un atributo es introducir ruido, pero siempre teniendo en cuenta el rango de valores de dicho atributo. Por ejemplo, supongamos que tenemos un atributo que nos dice la altura de una persona. El rango de valores de este atributo puede ir desde 45 a 220 centímetros. Por tanto, para perturbar dicho atributo cuyo valor es v se podría generar un número aleatorio en el rango $[v - 45, 220 - v]$. De esta manera aseguramos no generar valores fuera del rango del atributo.

3 LIME (Local interpretable model-agnostic explanations)

LIME[5] es un método de XAI basado en un modelo subrogado. En XAI, los modelos subrogados se entrenan para aproximar las predicciones del modelo de caja negra. Estos modelos subrogados deben ser por tanto modelos de caja blanca, fáciles de interpretar. Lo habitual es usar modelos lineales dispersos (con regularización ridge). En el caso de LIME, el modelo subrogado se entrena para aproximar una predicción individual y las predicciones de su vecindad, obtenidas al perturbar la muestra individual estudiada. El modelo subrogado de LIME se entrena con una representación de las muestras originales $x : \epsilon : R^d$. La representación utiliza $x' \epsilon \{0, 1\}^{d'}$ para indicar la no-perturbación/perturbación de cada atributo original. Es decir, dada una muestra cuyos atributos son

Muestras

muestra	a1	a2	a3	a4
original	45	23.1	1	104
perturbacion 1	23	23.1	1	104
perturbacion 2	11	19.2	0	104
perturbacion 3	45	25.6	0	64
perturbacion 4	32	23.1	1	54
perturbacion 5	45	28.1	1	156

Representación

muestra	a1	a2	a3	a4
original	0	0	0	0
perturbacion 1	1	0	0	0
perturbacion 2	1	1	1	0
perturbacion 3	0	1	1	1
perturbacion 4	1	0	0	1
perturbacion 5	0	1	0	1

Figure 1: A la izquierda la muestra original y sus permutaciones con los valores reales. A la derecha su representación: 1 indica que ese atributo de esa muestra ha sido perturbado.

valores reales, generaremos una nueva muestra cuyos valores de sus atributos son únicamente 1 ó 0. Si un atributo ha sido perturbado, este tomará el valor 0 en su representación, en caso contrario, este tomará el valor 1 (ver figura 1).

Matemáticamente, las explicaciones obtenidas con LIME se pueden expresar como:

$$\xi(x) = \underset{g \,\epsilon\, G}{argmin}\, \mathcal{L}(f, g, \pi_x) + \Omega(g) \tag{1}$$

donde g es un modelo subrogado de la clase G perteneciente al grupo de modelos interpretables. El componente $\Omega(g)$ se usa como regularización para mantener la complejidad de g baja, ya que la alta complejidad se opone al concepto de interpretabilidad. El modelo que se está explicando se denota como f y $\mathcal{L}$ determina el rendimiento de g ajustando la localidad definida por π como una función de medición de proximidad con $\pi_x = \pi(x, \cdot)$. Es importante tener en cuenta, que cada muestra de entrenamiento (las representaciones de las muestras originales) se ponderan con la distancia entre la muestra perturbada y la muestra original. La distancia que se suele usar es la distancia coseno.

Como modelo subrogado, en este trabajo, se usará la regresión ridge. Una vez entrenado, la explicación será los pesos de la regresión. Así consideramos dichos pesos como la importancia que cada atributo tiene para la predicción estudiada.

El algoritmo para implementar LIME se detalla en el pseudocódigo 1.

Algorithm 1 LIME

N es el número de permutaciones a realizar
f es el modelo a explicar
$X' \leftarrow \{\}$ muestras perturbadas
$R \leftarrow \{\}$ representaciones
$W \leftarrow \{\}$ las distancias entre la muestra x y sus perturbaciones
for 1 to N **do**
    Selecciona k atributos aleatoriamente
    $x' \leftarrow$ una perturbación de x donde se perturban los k atributos anteriores.
    $w \leftarrow$ la distancia entre x y x'
    $r \leftarrow$ la representación de x' respecto a x
    $X' \leftarrow X' \cup x'$
    $R \leftarrow R \cup r$
    $W \leftarrow R \cup w$
end for
$Y' \leftarrow f(X')$ las predicciones de las perturbaciones
$G \leftarrow$ modelo ridge entrenado con R para predecir Y' y ponderando cada muestra con W
return los parámetros de G

4 SHAP (SHapley Additive exPlanations)

SHAP[6] es también un método para explicar predicciones individuales, como LIME. El método SHAP explica cada atributo calculando los valores de Shapley de la teoría de juegos coalicionales. El valor de Shapley se puede describir como la contribución promedia de cada jugador al resultado del juego. En SHAP los jugadores son los atributos y el resultado del juego es el valor de la predicción. Por tanto, los valores shapely representarán la contribución de cada atributo para generar la predicción. Como en LIME, se genera un vector representativo de cada muestra para representar si un atributo (o jugador) ha contribuido en la generación de la predicción (o resultado).

Para calcular la contribución marginal es necesario considerar todas las combinaciones posibles de presencia o no de cada atributo. Sin embargo, considerar todas las posibles combinaciones es inviable en la mayoría de los casos. La variante Kernel SHAP lleva a cabo una aproximación para el cálculo de las contribuciones teniendo solamente en cuenta la presencia de unas pocos atributos o la ausencia de unos pocos atributos. La intuición detrás de esto es que aislar atributos proporciona más información sobre su contribución a la predicción. Por tanto, el procedimiento será considerar aquellas coaliciones con solamente un 1, es decir, se perturban todos los atributos excepto uno de ellos. De la misma forma, se consideran las coaliciones con solamente un 0, es decir, solamente se perturba un atributo. Después se pasará a las combinaciones de dos 1's o dos 0's solamente. Como en LIME, las representaciones están ponderadas, pero en este caso esta ponderación va en función del número de 1's y 0's presentes:

$$\pi_x(r) = \frac{(M-1)}{\binom{M}{|r|}(M-|r|)} \tag{2}$$

donde M es el número de atributos totales de la muestra estudiada, r el vector de coalición (representativo) y $\pi_x(r)$ es la ponderación (o peso) asociado a dicho vector. Según la ecuación anterior, las coaliciones pequeñas (con pocos 1's) y las coaliciones grandes (es decir, con muchos 1's) reciben ponderaciones más grandes.

El algoritmo para implementar Kernel SHAP de describe en el pseudocódigo 2. Como se puede observar, el algoritmo es prácticamente el mismo que el de LIME. Únicamente cambia el modo de crear las representaciones y el modo de ponderarlas.

Algorithm 2 Kernel SHAP

NC es el número máximo de atributos a perturbar
$A \leftarrow$ el conjunto de los atributos de x
f es el modelo a explicar
$X' \leftarrow \{\}$ muestras perturbadas
$R \leftarrow \{\}$ representaciones
$W \leftarrow \{\}$ ponderaciones de las representaciones.
for $nc \in \{1..NC\}$ **do**
  for $A' \in \binom{A}{nc}$ **do**
    $x' \leftarrow$ una perturbación de x donde se perturban los atributos en A'.
    $r \leftarrow$ la representación de x' respecto a x
    $w \leftarrow$ ponderación de r según la ecuación 2
    $X' \leftarrow X' \cup x'$
    $R \leftarrow R \cup r$
    $W \leftarrow R \cup w$
  end for
end for
$Y' \leftarrow f(X')$ las predicciones de las perturbaciones
$G \leftarrow$ modelo ridge entrenado con R para predecir Y' y ponderando cada muestra con W
return los parámetros de G

5 Métricas

A continuación, se pasa a definir un conjunto de métricas para utilizar en este trabajo. Estas métricas pretenden medir el nivel de cumplimiento de ciertas características deseables que cada método de XAI debería cumplir:

- *Identidad*: El principio de identidad establece que objetos idénticos deben recibir explicaciones idénticas. Esto estima el nivel de no determinismo intrínseco en el método.

$$\forall a, b \ (d(x_a, x_b) = 0 \implies d(\epsilon_a, \epsilon_b) = 0) \tag{3}$$

donde x son muestras, d es una función de distancia y ϵ son vectores de explicación (que explican la predicción de cada muestra).

- *Separabilidad*: Objetos no idénticos no pueden tener explicaciones idénticas.

$$\forall a, b : (d(x_a, x_b) \neq 0 \implies d(\epsilon_a, \epsilon_b) > 0) \tag{4}$$

Si una característica no es necesaria para la predicción, entonces dos muestras que difieren solo en esa característica tendrán la misma predicción. En este escenario, el método de explicación podría proporcionar la misma explicación, aunque las muestras sean diferentes. Por simplificación, esta métrica supone que cada característica tiene un nivel mínimo de importancia, positivo o negativo, en las predicciones.

- *Estabilidad*: Objetos similares deben tener explicaciones similares. Esto se basa en la idea de que un método de explicación solo debe devolver explicaciones similares para objetos ligeramente diferentes. La correlación de Spearman ρ se utiliza para definir esto:

$$\rho(\{d(x_i, x_0), d(x_i, x_1), ..., d(x_i, x_n)\}, \{d(\epsilon_i, \epsilon_0), d(\epsilon_i, \epsilon_1), ..., d(\epsilon_i, \epsilon_n)\}) \underset{\forall i}{=} \rho_i > 0 \tag{5}$$

- *Selectividad*: La eliminación de variables relevantes debe afectar negativamente a la predicción. Para calcular la selectividad, las características se ordenan de la más a la menos relevante. Una por una, las características se eliminan, estableciéndolas en cero, por ejemplo, y se obtienen los errores residuales para obtener el área bajo la curva (AUC).

- *Coherencia*: Se calcula la diferencia entre el error de predicción p_e^i sobre la señal original y el error de predicción e_e^i de una nueva señal donde se eliminan las características no importantes.

$$\alpha_i = \left| p_e^i - e_e^i \right| \tag{6}$$

donde α_i es la coherencia de una muestra.

- *Completitud*: Evalúa el porcentaje de error de explicación con respecto al error de predicción.

$$\gamma_i = \frac{e_e^i}{p_e^i} \tag{7}$$

- *Congruencia*: La desviación estándar de la coherencia proporciona el proxy de congruencia. Esta métrica ayuda a capturar la variabilidad de la coherencia.

$$\delta = \sqrt{\frac{\sum((\alpha_i - \overline{\alpha})^2}{N}} \tag{8}$$

donde $\overline{\alpha}$ es la coherencia promedio sobre un conjunto de N muestras:

$$\overline{\alpha} = \frac{\sum \alpha_i}{N} \tag{9}$$

6 Desarrollo del trabajo

El trabajo consistirá en las siguientes tareas:

1. Implementación de un método XAI (LIME o SHAP).

2. Implementación de las métricas expuestas en la sección anterior. Se recomienda el uso de librerías como numpy y scipy.

3. Selección de 2 conjuntos de datos. El tipo de dato con el que se deberá trabajar dependerán de la evaluación (junio o julio). Ver secciones 6.1 y 6.2.

4. Entrenamiento de dos modelos de caja negra (random forest, xgboost, redes neuronales, ...) con cada conjunto de datos (total 4 modelos). El modelo deberá estar bien entrenado, es decir, no deberá estar sobreajustado ni infra-ajustado. Se valorará por tanto las técnicas de validación y optimización de modelo usadas.

5. Medición de las métricas sobre los 4 modelos entrenados para el método XAI implementado. Para llevar a cabo las explicaciones se deberán seleccionar al menos 256 muestras de los conjuntos de datos que no hayan sido usadas para entrenar los modelos.

6. Presentación de resultados y conclusiones.

6.1 Evaluación de junio (1ª convocatoria)

En la evaluación de junio se centrará en el desarrollo y la experimentación con conjuntos de datos tabulares. Se deberán seleccionar 2 conjuntos de datos de la web UCI[1]. La selección de conjuntos de datos es libre, pero deberán ser de tamaño medio: al menos deberá tener 10 atributos y 5000 muestras (filas).

[1] Conjuntos de datos de la UCI

Figure 2: Superpíxeles extraídos con SLIC

6.2 Evaluación de julio y noviembre (2ª convocatoria y convocatoria extraordinaria)

En la evaluación de julio se centrará el desarrollo y la experimentación con conjuntos de datos de imágenes en escala de grises. En este caso, para entrenar los modelos se deberán aplanar las imágenes para entrenar el clasificador. Para la aplicación de los métodos XAI no es viable usar como atributo cada pixel, ya que la dimensionalidad es enorme y se obtendría un menor nivel de explicabilidad. Con este tipo de datos se deben extraer de las imágenes lo que se conoce como super-píxel (ver figura 2).

Estos super-pixeles harán las funciones de los atributos en el caso tabular. La perturbación en este caso, consistirá en hacer totalmente blanco o negro el super-pixel seleccionado. Para la extracción de los super-pixeles se permitirá el uso de código disponible en internet. Como por ejemplo el que se puede encontrar en este blog.

En este caso se deberán seleccionar 2 conjuntos de datos entre los siguientes:

- MNIST.
- Fashion-MNIST
- not-MNIST
- Sign Language MNIST. En este caso, las imágenes se deberán convertir primero a escala de grises.

7 Criterios de evaluación

El trabajo será evaluado siguiendo los siguientes criterios:

- Corrección, claridad y comentarios en la implementación del método XAI (1 punto).
- Corrección, claridad y comentarios en la implementación de las métricas (1 punto a dividir entre las métricas).
- Corrección, claridad y comentarios en el entrenamiento de los modelos. (0.6 puntos)
- Medición de las métricas (0.6 puntos).
- Memoria, presentación de resultados y conclusiones (0.8 puntos)

El trabajo deberá ser defendido. En esta defensa se valorará el conocimiento adquirido, la claridad en la presentación y defensa y la veracidad de la realización del trabajo por parte del alumno. Durante la defensa se solicitará:

- la aplicación del método de aplicabilidad implementado con una muestra y un modelo.
- la aplicación de las métricas implementadas para un modelo.

La nota obtenida anteriormente será ponderada en función del resultado de la defensa.

8 Documentación a entregar

Para que el trabajo sea evaluado se deberá entregar el código y una memoria. La memoria deberá contener una explicación de los métodos aplicados su estado del arte, así como los resultados y conclusiones.

La memoria deberá redactarse siguiendo un formato de artículo científico. En la página web de la asignatura se pueden encontrar plantillas donde se sugiere una estructura general. Estas plantillas siguen el formato de los IEEE conference proceedings, cuyo sitio web guía para autores ofrece información más detallada. El documento entregado deberá estar en formato PDF

Se recomienda que el código sea subido a algún repositorio de código fuente como github, gitlab o bitbucket. Se deberá incluir en la memoria el enlace a dicho repositorio.

9 Honestidad académica

Cualquier plagio, compartición de código o uso de material que no sea original y del que no se cite convenientemente la fuente, significará automáticamente la calificación de cero en la asignatura para todos los alumnos involucrados. Por tanto, a estos alumnos no se les conserva, ni para la actual ni para futuras convocatorias, ninguna nota que hubiesen obtenido hasta el momento. Todo ello sin perjuicio de las correspondientes medidas disciplinarias que se pudieran tomar.

References

[1] Pomerleau, Dean A. **Neural networks for intelligent vehicles**. *Proceedings of IEEE Conf. on Intelligent Vehicles*, 19–24

[2] Goodman, Bryce and Flaxman, Seth. **European Union regulations on algorithmic decision-making and a "right to explanation**. *AI magazine* **2017**, *38*, 50–57

[3] *Gilpin, L.H.; Bau, D.; Yuan, B.Z.; Bajwa, A.; Specter, M.; Kagal, L.* **Explaining explanations: An overview of interpretability of machine learning.**, *2018 IEEE 5th International Conference on data science and advanced analytics (DSAA)*, **2018**; pp 80–89

[4] *Carvalho, Diogo V and Pereira, Eduardo M and Cardoso, Jaime S* **Machine learning interpretability: A survey on methods and metrics**, *Electronics*, **2019**, *8(8)*, 832

[5] Ribeiro, Marco Tulio and Singh, Sameer and Guestrin, Carlos. **" Why should i trust you?" Explaining the predictions of any classifier**. *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining* **2016**, 1135–1144

[6] Lundberg, Scott M and Lee, Su-In. **A unified approach to interpreting model predictions**. *Advances in neural information processing systems* **2017**, *30*

[7] *Schlegel, Udo and Arnout, Hiba and El-Assady, Mennatallah and Oelke, Daniela and Keim, Daniel A* **Towards a rigorous evaluation of xai methods on time series**, *2019 IEEE/CVF International Conference on Computer Vision Workshop (ICCVW)*, **2019**; 4197–4201

[8] *Vollert, Simon and Atzmueller, Martin and Theissler, Andreas* **Interpretable Machine Learning: A brief survey from the predictive maintenance perspective**, *2021 26th IEEE international conference on emerging technologies and factory automation (ETFA)*, **2021**, pp 01–08

[9] *Samek, Wojciech and Wiegand, Thomas and Müller, Klaus-Robert* **Explainable artificial intelligence: Understanding, visualizing and interpreting deep learning models**, *arXiv preprint arXiv:1708.08296*, , **2017**

[10] *Doshi-Velez, F.; Kim, B.* **Towards a rigorous science of interpretable machine learning**. *arXiv 2017, arXiv:1702.08608*

[11] *Honegger, M.* **Shedding Light on Black Box Machine Learning Algorithms: Development of an Axiomatic Framework to Assess the Quality of Methods that Explain Individual Predictions**. *arXiv 2018, arXiv:1808.05054*

[12] *Silva, Wilson and Fernandes, Kelwin and Cardoso, Maria J and Cardoso, Jaime S* **Towards complementary explanations using deep neural networks**. Understanding and Interpreting Machine Learning in Medical Image Computing Applications, **2018**; pp. 133–140.

[13] *Shapley LS.* **A value for n-person games**, **1953**