IA EXPLICABLE

3 y 6 de noviembre

GRÁFICOS DE DEPENDENCIA PARCIAL (PDP)

- Un Gráfico de Dependencia Parcial (PDP, Partial Dependence Plot) es una técnica de visualización utilizada para mostrar la relación entre una característica (o un conjunto de características) y el resultado predicho de un modelo de aprendizaje automático.
- Es útil para comprender modelos como Gradient Boosting, Random Forests o NNs, pero se puede aplicar a cualquier tipo de modelo.
- Su objetivo es mostrar el efecto de una única característica en la predicción de un modelo, manteniendo constantes todas las demás características. Por ejemplo, en un modelo de predicción del precio de una casa, podrías querer saber cómo afecta la superficie de la casa al precio, manteniendo constantes otros factores

GRÁFICOS DE DEPENDENCIA PARCIAL (PDP)

- Para una característica dada, el PDP variará sus valores a lo largo de su dominio.
- Para cada valor, hará predicciones del modelo usando los datos de entrenamiento, pero con la característica de interés establecida en ese valor para todas las instancias.
- Luego promedia estas predicciones para obtener una única predicción para ese valor de característica.
- Este proceso se repite para cada valor único de la característica o para una cuadrícula de valores que abarcan el dominio de la característica.

GRÁFICOS DE DEPENDENCIA PARCIAL (PDP)

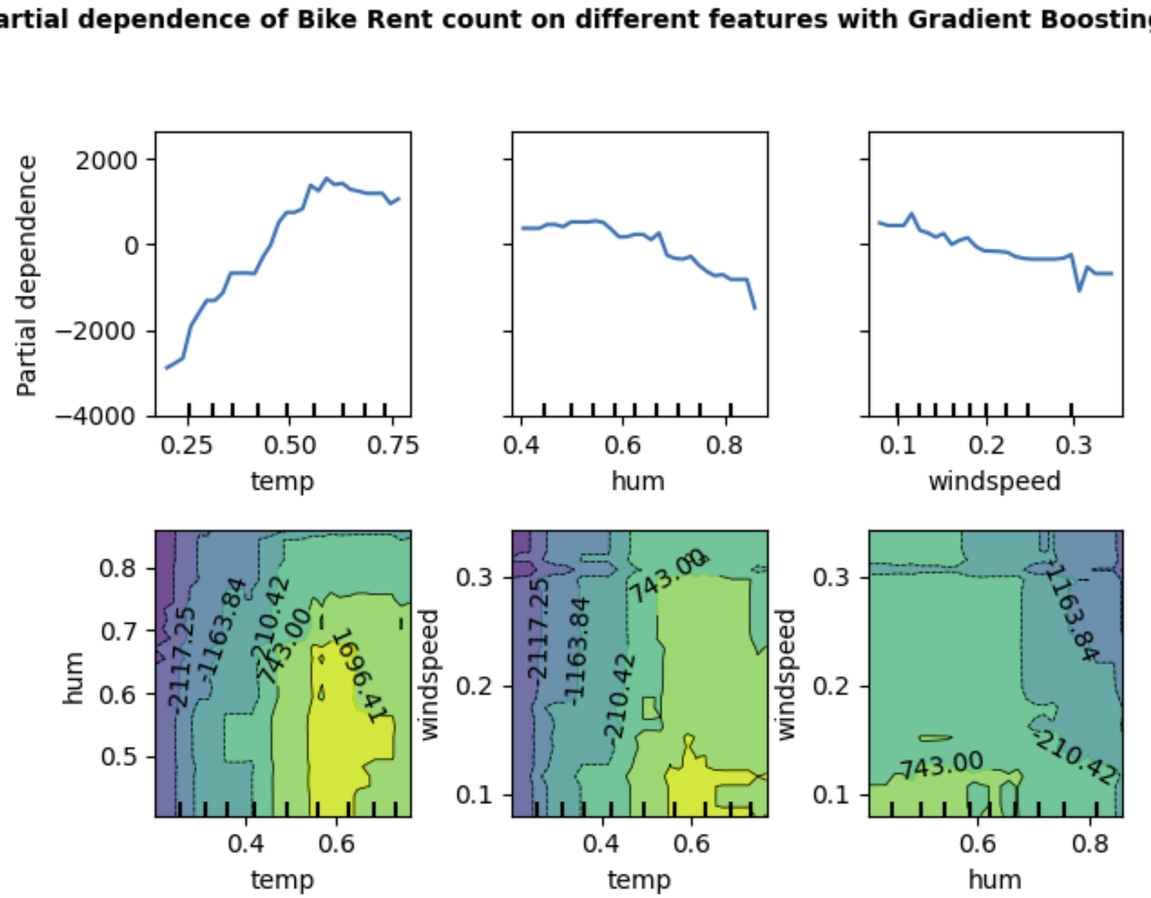
- El gráfico resultante tiene los valores de la característica en el eje x y el resultado promedio predicho en el eje y. Este gráfico puede ofrecer información sobre:
  - Direccionalidad: Si la relación es positiva o negativa.
  - Magnitud: Cuánto cambia la predicción a medida que cambia el valor de la característica.
  - No linealidad: Si el efecto es constante, aumenta o disminuye a lo largo de los valores de la característica.
  - Interacciones (cuando se considera más de una característica): Cómo las características combinadas interactúan para afectar la predicción.

LIMITACIONES DE LOS GRÁFICOS DE DEPENDENCIA PARCIAL (PDP)

- Asume que el efecto de la característica es independiente de otras características. Esto no es correcto porque, en realidad, podría haber interacciones entre características.
- Trabajar con características categóricas de alta cardinalidad o conjuntos de datos de muy alta dimensión puede hacer que calcular los PDP sea computacionalmente costoso.
- La interpretación puede complicarse para características que están altamente correlacionadas con otras.

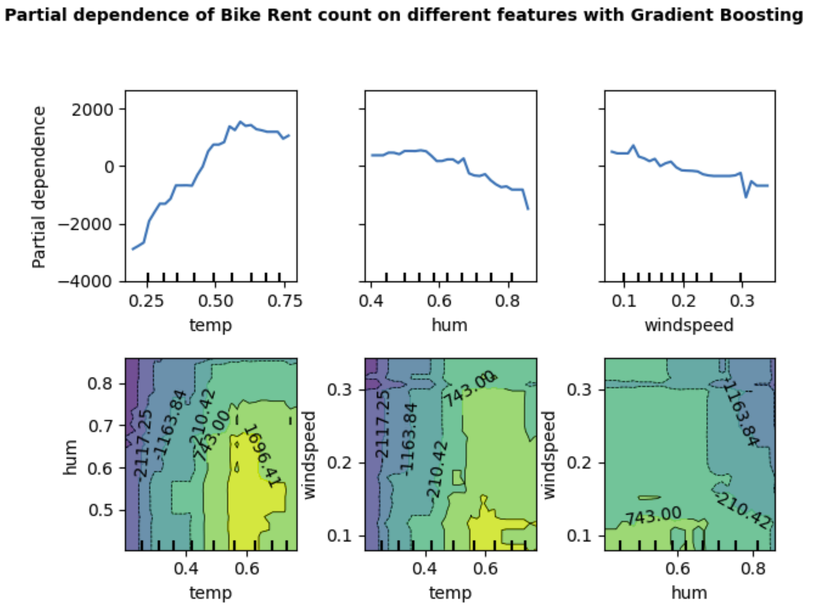
EJEMPLO PDP

- Ver notebook `interpretableAI.ipynb`



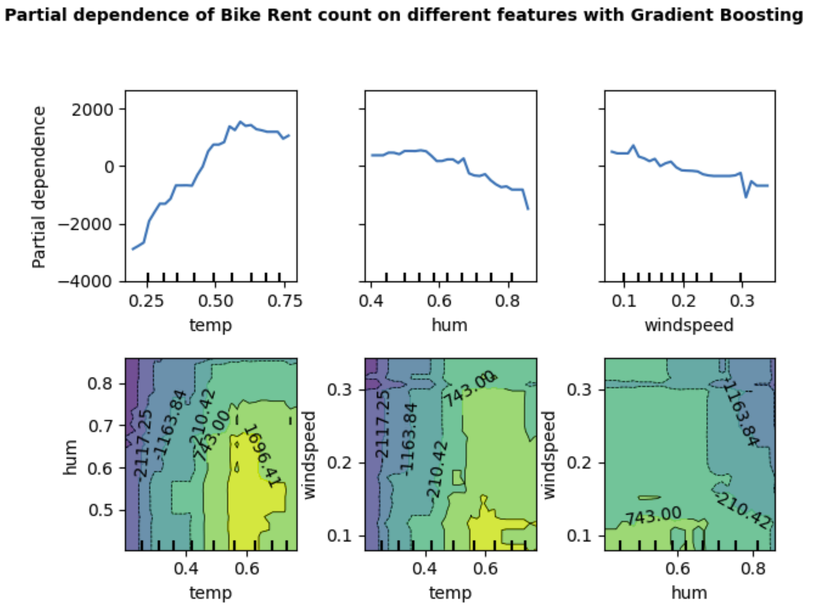
EJEMPLO PDP

- Interacción unidireccional: El eje y representa el número de bicicletas alquiladas, mientras que el eje x muestra cada característica.
- Entre las 3 características, podemos ver que la temperatura es la que más afecta a la variable objetivo. Cuando la temperatura es alta, se alquilan más bicicletas. Esto tiene sentido en la realidad, porque la gente suele ir en bici cuando hace calor. En invierno, no es ideal desplazarse en bicicleta. Del primer gráfico también se desprende que, cuando la temperatura alcanza un determinado nivel, el número de bicicletas alquiladas es relativamente constante.
- Por el contrario, cuando la humedad y la velocidad del viento aumentan (tal vez el tiempo sea lluvioso), el número de bicicletas alquiladas disminuye.
- Este gráfico puede ayudarnos con 2 preguntas:
  - Cómo afecta cada característica a la variable objetivo.
  - En qué medida puede afectar cada característica a la variable objetivo.



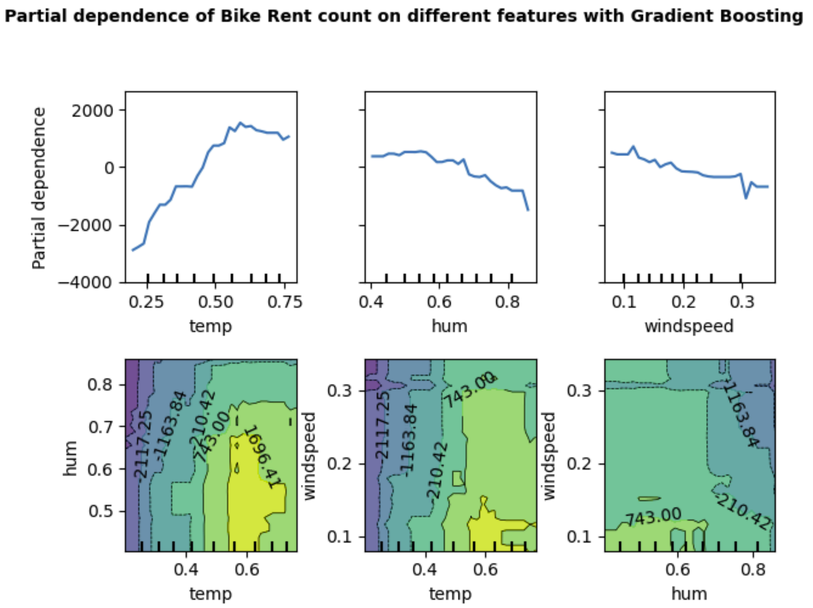
EJEMPLO PDP

- Interacción bidireccional: La segunda fila del gráfico es un mapa de calor que muestra cómo 2 características pueden influir en la variable objetivo.
- ("temp", "hum") nuestra cómo cambia la variable objetivo predicha a medida que varían la temperatura y la humedad. Este gráfico puede ayudarle a comprender cómo la combinación de temperatura y humedad influye en la variable objetivo. En el gráfico, cuanto más claro es el color, más bicicletas se alquilan. Concretamente, en el primer gráfico de interacción, el alquiler es alto cuando la temperatura es suficientemente cálida y la humedad es baja (sin lluvia).



EJEMPLO PDP

- Una de las suposiciones de PDP es que todas las características son independientes, por lo que podríamos mantener k - 1 características constantes y cambiar una característica cada vez con distintos valores para comprobar su efecto en el rendimiento del modelo.
- Sin embargo, esta suposición no es cierta, ya que los predictores suelen estar relacionados a cierto nivel. Por ejemplo, la temperatura puede estar relacionada con la humedad, por lo que si cambiamos la temperatura pero mantenemos la humedad, se distorsiona la esencia de los datos.



GRAFICOS DE ESPERANZA CONDICIONAL

- Los gráficos de Esperanza Condicional Individual (ICE - Individual Conditional Expectation) son una técnica de visualización que se utiliza para entender cómo una característica en particular afecta las predicciones de un modelo de aprendizaje automático para casos individuales. Son una extensión o generalización de los Gráficos de Dependencia Parcial (PDP).
- Un PDP muestra el efecto promedio de una característica en las predicciones del modelo, mientras que un gráfico ICE muestra el efecto de una característica en la predicción para una instancia individual. Esto permite visualizar no solo el efecto promedio, sino también las variaciones individuales.

GRAFICOS DE ESPERANZA CONDICIONAL

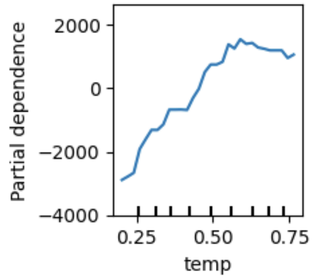
- Los gráficos de Esperanza Condicional Individual (ICE - Individual Conditional Expectation) son una técnica de visualización que se utiliza para entender cómo una característica en particular afecta las predicciones de un modelo de aprendizaje automático para casos individuales. Son una extensión o generalización de los Gráficos de Dependencia Parcial (PDP).
- Un PDP muestra el efecto promedio de una característica en las predicciones del modelo, mientras que un gráfico ICE muestra el efecto de una característica en la predicción para una instancia individual. Esto permite visualizar no solo el efecto promedio, sino también las variaciones individuales.

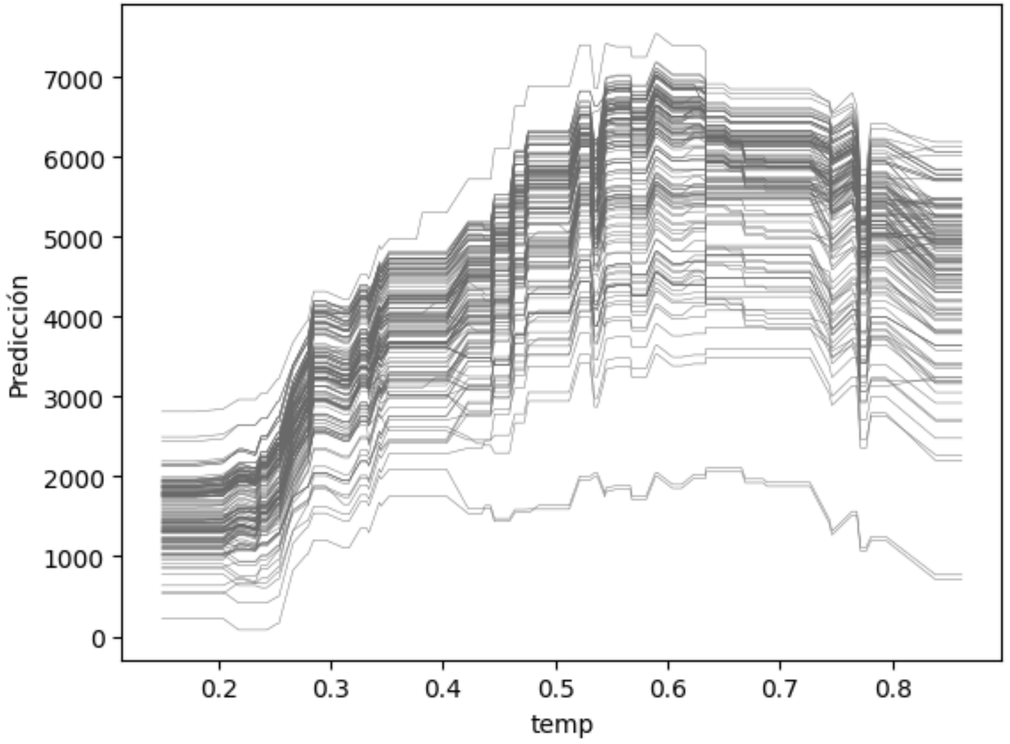
GRAFICOS DE ESPERANZA CONDICIONAL

- Se selecciona una instancia del conjunto de datos.
- Se varía la característica de interés a lo largo de su rango, mientras se mantienen constantes las otras características.
- Se predice el resultado del modelo para cada valor de la característica, creando una curva para esa instancia.
- Se repite el proceso para varias instancias, superponiendo todas las curvas en un mismo gráfico.

GRAFICOS DE ESPERANZA CONDICIONAL

- Un gráfico ICE típicamente tiene los valores de la característica en el eje x y el resultado predicho en el eje y. Cada curva representa una instancia individual. Así, en lugar de una sola línea como en el PDP, tendrás múltiples líneas en el gráfico ICE.
- A menudo, los gráficos ICE se visualizan junto con un PDP para mostrar tanto los efectos promedio como individuales. El PDP actúa como una línea de tendencia promedio a través del "bosque" de líneas ICE.





LIME

- LIME (Local Interpretable Model-agnostic Explanations) es un método para explicar las predicciones de cualquier modelo de aprendizaje automático de forma interpretable.
- La idea detrás de LIME consiste en aproximarse al modelo complejo usando un modelo más simple y interpretable en la vecindad de la instancia que se quiere explicar.

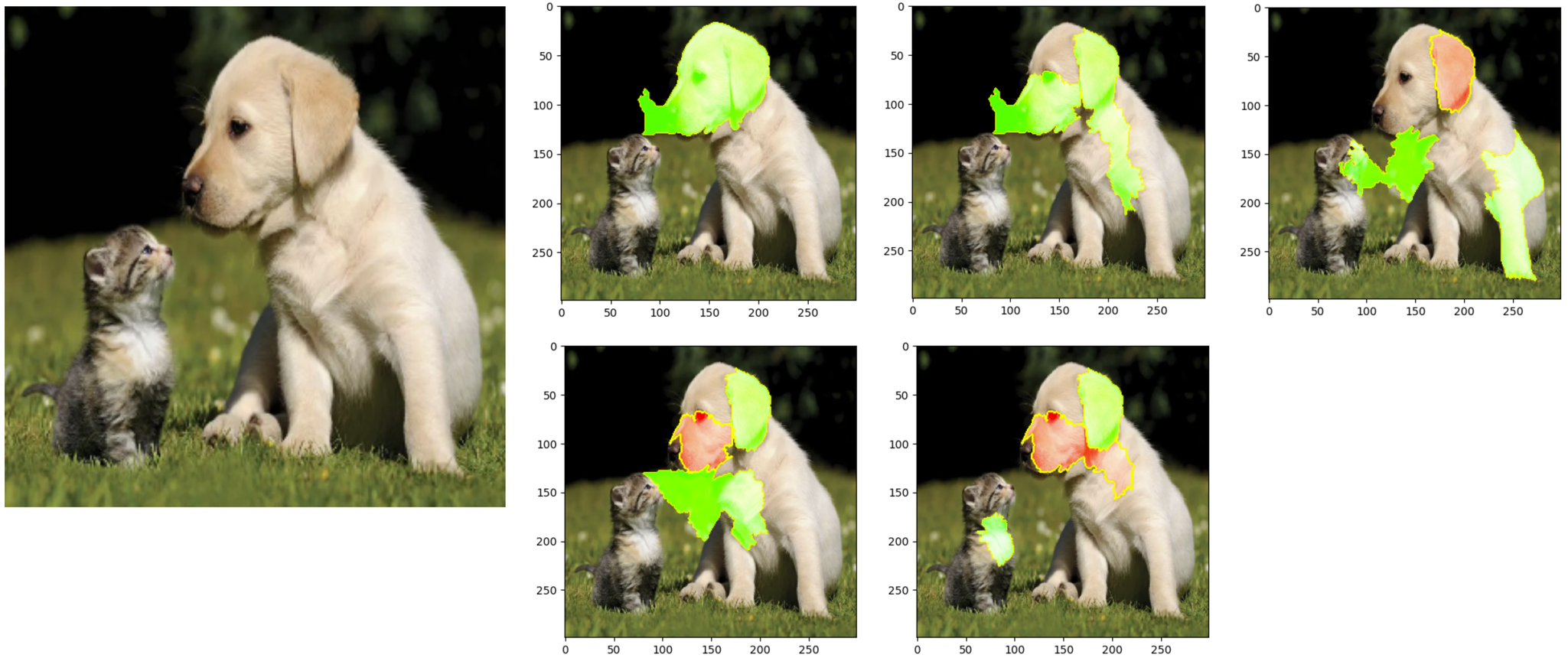
FUNCIONAMIENTO DE LIME

- Perturbación: Dada una instancia que queremos explicar, LIME la perturba creando un conjunto de muestras "similares" pero con características variadas.
- Predicción con el Modelo: LIME utiliza el modelo original para predecir los resultados de estas muestras perturbadas.
- Modelo Local: Se ajusta un modelo simple (por ejemplo, regresión lineal) usando las muestras perturbadas como entrada y las predicciones del modelo complejo como salida. La idea es que este modelo simple puede aproximar de manera aceptable al modelo complejo en la vecindad local de la instancia de interés.
- Interpretación: Las características del modelo simple (como los coeficientes de la regresión lineal) proporcionan una explicación de cómo el modelo original toma decisiones para esa instancia particular (es una idea similar a la que ya vimos para los gráficos LLE, donde cada punto se explicaba con los coeficientes de una combinación de sus vecinos)

VISUALIZACIÓN DE LIME

- LIME suele presentar las explicaciones resaltando las características más importantes que contribuyen a una predicción.
  - Por ejemplo, en el caso de una imagen, podría resaltar las regiones que fueron decisivas para clasificarla en una categoría específica.
  - Para datos tabulares, podría indicar qué características y en qué magnitud influenciaron la predicción.

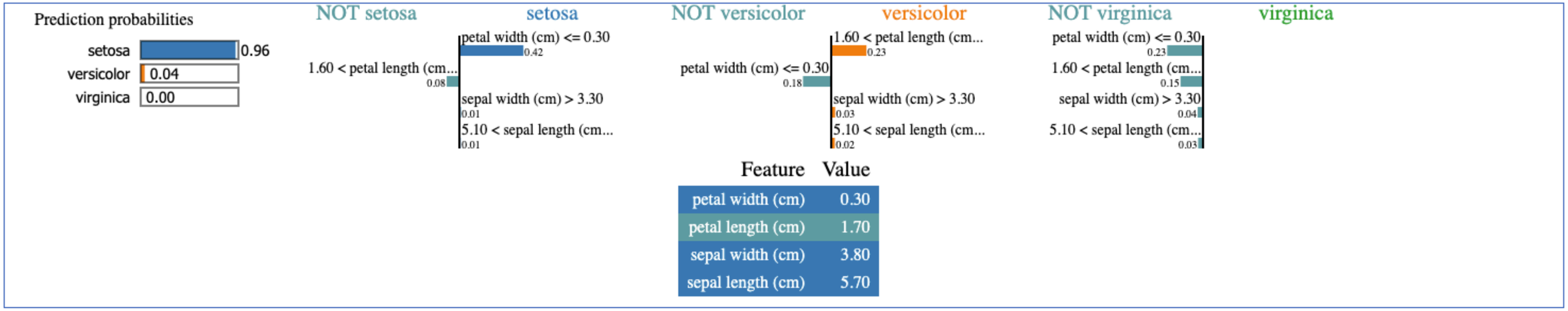
EJEMPLO LIME



- [208: 'Labrador retriever', 207: 'golden retriever', 216: 'clumber, clumber spaniel', 215: 'Brittany spaniel', 852: 'tennis ball']

LIME PARA DATOS TABULARES

- LIME también se emplea habitualmente para explicar datos tabulares en problemas de clasificación (ver ejemplo en notebook `interpretableAI.ipynb`)




```
exp.as_list()
```

✓ 0.0s

```
[('1.60 < petal length (cm) <= 4.30', 0.22850290607650411),
 ('petal width (cm) <= 0.30', -0.18307659758787295),
 ('sepal width (cm) > 3.30', 0.025595019670645244),
 ('5.10 < sepal length (cm) <= 5.80', 0.023349298398021526)]
```

SHAPLEY VALUES (SHAP)

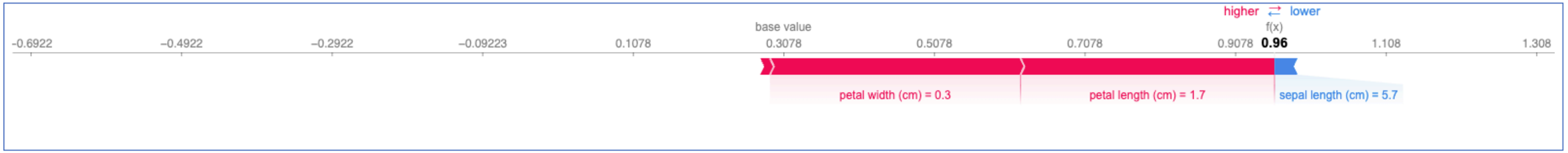
- Los Valores Shapley provienen de la teoría de juegos cooperativos y se utilizan para determinar cómo distribuir un "pago" entre diferentes jugadores en función de su contribución al juego completo.
- En el contexto de la interpretabilidad del modelo de aprendizaje automático, los "pagos" son las predicciones del modelo, y los "jugadores" son las características (o variables) de entrada del modelo. Los Valores Shapley proporcionan una forma de "repartir" la predicción entre las características basándose en su importancia.
- El cálculo de los Valores Shapley se basa en considerar todas las posibles combinaciones de características y determinar cuánto contribuye cada característica a la diferencia entre la predicción del modelo con y sin esa característica.

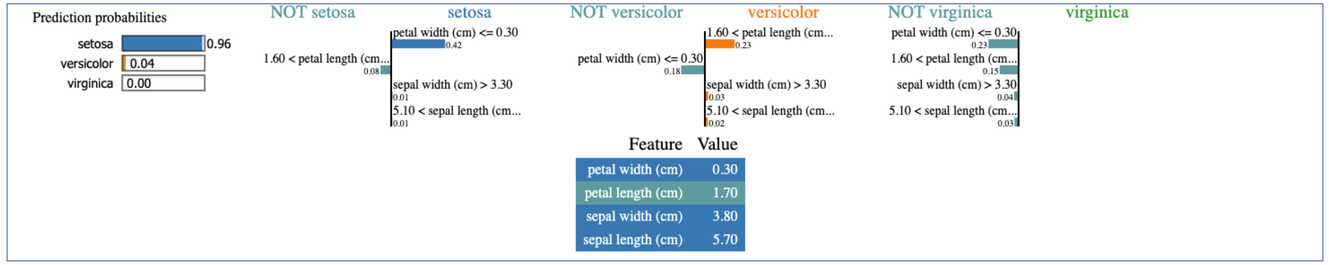
PROPIEDADES DE LOS SHAPLEY VALUES (SHAP)

- Eficiencia: La suma de los Valores Shapley de todas las características es igual a la diferencia entre la predicción del modelo para esa instancia y el valor base (que suele ser el promedio de todas las predicciones del modelo).
- Simetría: Si dos características contribuyen de igual manera, entonces tendrán el mismo Valor Shapley.
- Aditividad: Si se combina el "pago" de dos juegos diferentes, entonces el Valor Shapley de un jugador (en este caso, una característica) en el juego combinado es la suma de los Valores Shapley en los juegos individuales.
- Nulidad: Si una característica no cambia la predicción, su Valor Shapley es cero.

EJEMPLO SHAP

- Ver notebook `interpretableAI.ipynb`
- En el mismo problema iris, la importancia de cada una de las variables en la clasificación de la instancia anterior como "setosa" se muestra en la figura superior (observa que la lista de variables relevantes no coincide con la indicada por LIME, figura inferior):





DIFERENCIAS SHAP/LIME

- Las explicaciones son aproximaciones y pueden variar según la técnica, los hiperparámetros y las características del modelo y los datos
- LIME aproxima localmente el modelo. Genera perturbaciones (muestras) alrededor de la instancia de interés, obtiene las predicciones del modelo para estas perturbaciones y luego entrena un modelo interpretable (como una regresión lineal) en este conjunto perturbado para aproximarse a las predicciones del modelo original en esa región local.
- SHAP, por el contrario, utiliza los valores de Shapley (que son una distribución equitativa de las contribuciones basadas en la teoría de juegos cooperativos) para distribuir equitativamente la contribución de cada característica a la predicción.

DIFERENCIAS SHAP/LIME

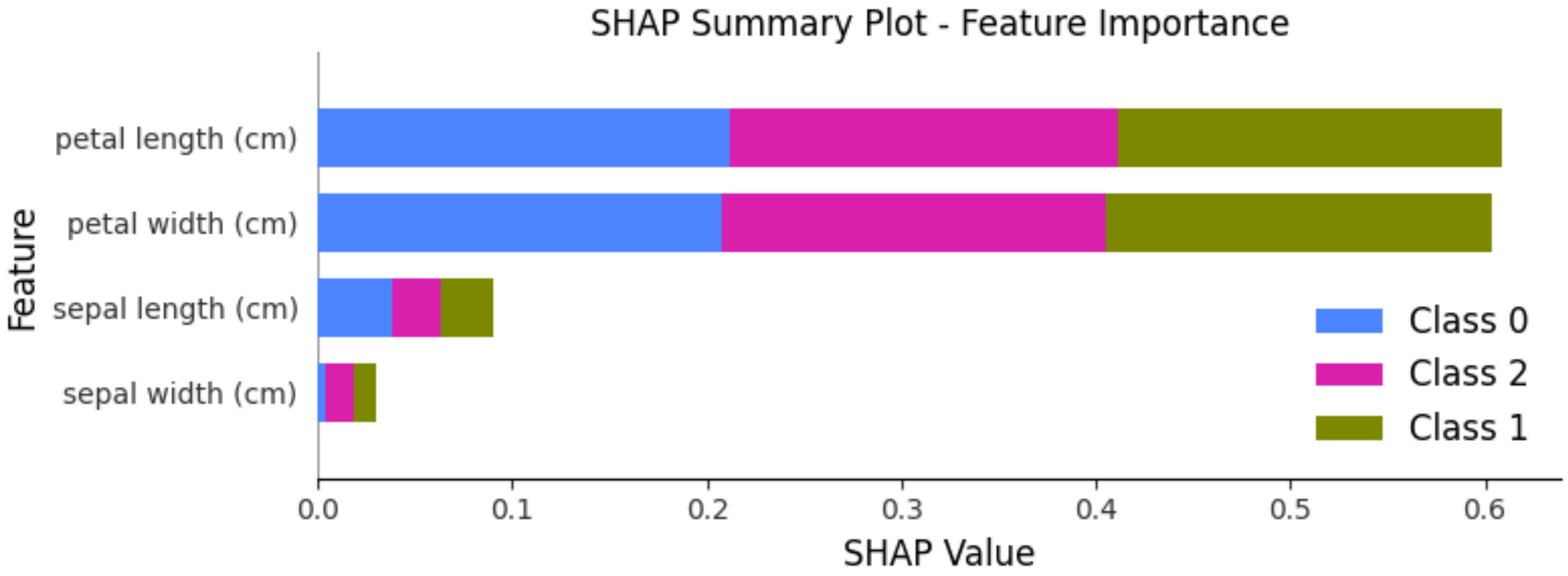
- LIME utiliza perturbaciones aleatorias de la instancia de interés para obtener un conjunto de datos local y aprender el modelo explicativo. Estas perturbaciones pueden variar en diferentes ejecuciones, lo que puede llevar a variabilidad en las explicaciones.
- SHAP considera todas las posibles combinaciones de características, por lo que es determinístico en ese aspecto.

DIFERENCIAS SHAP/LIME

- LIME utiliza perturbaciones aleatorias de la instancia de interés para obtener un conjunto de datos local y aprender el modelo explicativo. Estas perturbaciones pueden variar en diferentes ejecuciones, lo que puede llevar a variabilidad en las explicaciones.
- SHAP considera todas las posibles combinaciones de características, por lo que es determinístico en ese aspecto.
- Las explicaciones basadas en SHAP pueden considerar interacciones de mayor orden entre características, especialmente cuando se utiliza con modelos basados en árboles. LIME, en su enfoque estándar con modelos lineales, puede no captar estas interacciones de la misma manera.

SHAP SUMMARY PLOT

- La importancia de cada variable en relación con cada clase en todo el conjunto de datos puede resumirse con SHAP de la forma siguiente:



GRÁFICOS SHAP DE CASCADA

- Una representación alternativa a la importancia de cada variable en la decisión tomada para una instancia concreta es el gráfico de cascada (waterfall) (en la parte inferior se muestra la misma información con la representación vista antes)