TP2 - Machine Learning I

Cátedra Argerich

Ir al inicio

TP2 - Machine Learning I

Dataset a usar

El dataset a usar es el extraído de los experimentos realizados en el paper Mushroom data creation, curation, and simulation to support classification tasks sobre la predicción de si un hongo es comestible o no. El dataset se encuentra disponible en UCI Machine Learning Repository (*Link alternativo directo al zip del dataset*) (utilizar secondary data). En el paper citado y en el sitio para descargar el dataset se explica qué es cada feature. El objetivo del TP es explicar y predecir la variable `class`, que vale `p`, si el hongo es venenoso (poisonous), o `e`, si el hongo es comestible (edible).

Parte I - Análisis Exploratorio (5 puntos)

Deberán realizar 5 visualizaciones **interesantes** que **ayuden a explicar el target** haciendo almenos un plot de cada uno de los siguientes tipos:

- Bar plot (o stacked bar plot o variaciones)
- Violin plot
- Box plot
- Heatmap

Parte II - Machine Learning Baseline (5 puntos)

Vamos a construir un modelo muy sencillo para saber qué es lo peor que podemos hacer, en general esta es una tarea muy importante que queremos que repitan en sus proyectos de machine learning. ¿Por qué?

- Navaja de Ockam: “Cuando se ofrecen dos o más explicaciones de un fenómeno, es preferible la explicación completa más simple; es decir, no deben multiplicarse las entidades sin necesidad.” ¿Para qué desarrollar un modelo super complejo si capaz es peor o casi igual que uno muy sencillo?
- Nos sirve para saber si estamos usando bien los modelos más complejos, si su score nos da peor al baseline probablemente se deba a un error de código.

- Nos sirve para rápidamente saber que tan complejo es un problema.
- Los modelos simples son fáciles de entender.

Utilice **todas las columnas del dataset** (exceptuando columnas que no tenga sentido usar para predecir) con algún encoding donde sea necesario para entrenar una regresión logística, utilizando búsqueda de hiperparametros y garantizando la reproducibilidad de los resultados cuando el notebook corriera varias veces. Conteste las preguntas:

- ¿Cuál es el mejor score de validación obtenido? (¿Cómo conviene obtener el dataset para validar?)
- Al predecir con este modelo para test, ¿Cúal es el score obtenido? (guardar el csv con predicciones para entregarlo después)
- ¿Qué features son los más importantes para predecir con el mejor modelo? Graficar.

Parte III - Random Forest (5 puntos)

Segun el paper con un clasificador basado en Random Forest deberiamos lograr un AUC de 1. Entrenar un Random Forest con búsqueda de hiperparametros que logre un AUC de 1 (¿cómo conviene elegir los datos de validación respecto de los de train?). El modelo debe cumplir las siguientes condiciones:

- Deben utilizar AUC-ROC como métrica de validación.
- Deben medirse solo en validación, **no contra test!!!**
- Deben ser reproducibles (correr el notebook varias veces no afecta al resultado).
- Deben tener un score en validación igual a 1.

Parte IV - Machine Learning (5 puntos)

Entrenar un nuevo modelo (que no sea Random Forest ni el utilizado para el baseline) con búsqueda de hiperparametros (¿cómo conviene elegir los datos de validación respecto de los de train?). El modelo debe cumplir las siguientes condiciones:

- Deben utilizar AUC-ROC como métrica de validación.
- Deben medirse solo en validación, **no contra test!!!**
- Deben ser reproducibles (correr el notebook varias veces no afecta al resultado).
- Deben tener un score en validación superior a 0,9.
- Para el feature engineering debe utilizarse imputación de nulos, mean encoding y one hot encoding al menos una vez cada uno.
- Deben utilizar al menos 40 features (contando cómo features columnas con números, pueden venir varios de la misma variable).
- Deberán contestar la siguientes preguntas:
    - ¿Cuál es el score en test? (guardar el csv con predicciones para entregarlo después)
    - ¿Por qué cree que logro/no logro el mismo valor de AUC que con Random Forest?

Puntos extra (un punto c/u)

Estas consignas suman puntos extra por fuera de los necesarios para aprobar el TP, mientras más consignas extra realicen más puntos consiguen y menos va a depender su aprobación de que los puntos de arriba estén bien:

- Entrenar una red neuronal con Keras que sea reproducible, usando al menos 40 features y un score en validación superior a 0,9. Debe ser un modelo por separado a los propuestos, no necesita búsqueda de hiper parámetros ni cumplir otra condición. ¿Cúal es su score en validación y en test?
- Graficar la importancia de features para el Random Forest de la parte III. ¿Qué tanto se parece a los features importantes de la parte II?
- Ensamble los modelos de la parte III y IV en uno solo. ¿Cúal es su score en validación y en test?
- Utilizando los árboles creados por el Random Forest y la importancia de los features, cree un árbol de decisión simple para que una persona normal pueda identificar si un hongo es comestible o no. Qué nivel de error posee este árbol al intentar clasificar un set de datos de testing?

Premio kahoot (un punto)

Utilizamos el promedio del puntaje normalizado de cada kahoot/parcialito para armar un podio. El podio se modificara a medida participen en los Kahoots. Los 5 primeros reciben un punto extra.

Criterio de corrección

Se necesita un 60% (12/20) de los puntos para aprobar. Los puntos extra permiten sumar por dentro de los 20 (uno se puede sacar hasta 25 pero se sigue aprobando con 12 y el 20 representa un 10).

Criterio de reentrega

Se podrá reentregar el TP si el puntaje es >=8 y están **todos los puntos desarrollados**. La reentrega consiste en hacer un punto extra y corregir todos los puntos donde tuvieran menos de la mitad de los puntos.

Se aprueba la reentrega si todos los puntos reentregados tienen al menos la mitad de los puntos. En caso de luego aprobar la instancia de reentrega, la nota es siempre 4.

Parte I

Cada visualización vale un punto y debe cumplir con las siguientes condiciones:

- Debe explicarse por si misma, sin necesidad de texto aclaratorio.
- Debe tener rótulos en los ejes que corresponda y en el título.
- Debe mostrar una relación o algo con la variable pedida que sea claro e interesante.
- El uso de color debe ser intencional, elegido por ustedes, no por la librería.
- La visualización debe ser legible (por ejemplo, un bar plot de 40 barras es ilegible)

Parte II

Vamos a corregir los siguientes puntos (no pueden restar más de 5 en total):

- Utiliza mal los datos de validación ya sea para obtener el resultado o para buscar hiper parámetros (-4 puntos), ejemplos: calcular el score con otras labels, calcular el AUC-ROC usando la predicción binaria y no la probabilidad, el set de validación se usa para elegir los parámetros pero también está dentro del entrenamiento de cada modelo, el set de validación se usa filtrando información a los encodings, validación no está tomado de forma correcta, etc.
- El modelo no está bien hecho (-5 puntos), ejemplo: entrenan con las labels o datos cambiados para algunas filas
- No es capaz de predecir para test o no lo hace correctamente (-5 puntos)
- No es reproducible (-2 puntos)
- No obtiene bien los features más importantes (-2 puntos)
- La predicción en test da menos de 0.5 (-2 puntos)
- La predicción para test tiene errores (-1 punto)
- No utiliza todas las columnas del dataset (-1 punto)

Parte III

- Cada condición no cumplida (o mal hecha) resta 1 punto.
- Feature engineering inapropiado (-2 puntos)
- No buscan para todos los hiperparametros importantes (-2 puntos)
- Resultado menor a 1 en validación (-5 puntos)

(a medida se acumulan estos pueden hacer que el modelo valga 0, pero nunca negativo)

Parte IV

- Cada condición no cumplida (o mal hecha) resta 1 punto.
- Feature engineering inapropiado para el modelo elegido (-2 puntos), ejemplos: features que no están normalizadas para una red neuronal, features sin ninguna consideración de escalas para un KNN, etc.
- No buscan para todos los hiperparametros importantes (-2 puntos)
- Resultado menor a 0.9 en validación (-5 puntos)

- Predicción para test no está bien hecha (-3 puntos)

(a medida se acumulan estos pueden hacer que el modelo valga 0, pero nunca negativo)

Detalles y recomendaciones

- Para consultas conceptuales sobre machine learning o preguntas de consigna pueden consultar en el canal de slack #consultas-tp2.
- Para consultas de código con su corrector o algún ayudante por privado.
- No recomendamos usar de forma directa y sin modificaciones modelos entrenados por otros para usarlos, esto solo les puede jugar en contra porque es imposible que cumplan las condiciones pedidas. Es probable que esos modelos estén orientados a conseguir buenos resultados (cosa que encima no evaluamos) y que tengan algún error conceptual.
- Recomendamos trabajar durante todo el TP en solo 4 notebooks: Uno de visualizaciones, otro para la regresión logística y uno para cada modelo (parte III y IV). Les recomendamos desarrollarlos de forma prolija y mostrar de forma ordenada cada uno de los resultados y pasos, con títulos y comentarios donde corresponda.
- El TP pide solo 5 visus y 3 modelos con condiciones muy claras, tengan esa consideración a medida avanzan para chequear que cumplen todo.
- El TP **no pide ni evalúa más que lo que dice**, si bien ser original y tener un buen score suma en términos de trabajo y aprendizaje para ustedes, sean inteligentes respecto a los modelos y features que eligen para trabajar para garantizar que pueden terminar. Ya van a tener tiempo de ser originales en el TP3...
- Particularmente **este TP es muy difícil empezarlo al final**, en cuotas se vuelve mucho más sencillo, les recomendamos empezar por las visus que no necesitan teoría nueva. Sabemos que muchos de ustedes vienen haciendo algunos tps la última semana, la experiencia de cuatrimestres anteriores nos dice que **con este no se puede hacer eso**, son demasiados conceptos a entender y muchas formas de hacerlo mal, no es solo una consigna a cumplir. No es que el TP sea más largo, sino que se vuelve más corto mientras más temprano lo empiecen.
- Todos los puntos deben estar desarrollados (exceptuando por supuesto los extra).
