75.06/95.58 Organización de Datos

Segundo Cuatrimestre de 2020

Trabajo Práctico 2: Enunciado

Objetivo

El objetivo del segundo trabajo práctico es resolver un problema de machine learning con los datos del TP1. Puntualmente, **se debe estimar la probabilidad de éxito para cada oportunidad de negocio**. El error de la solución (inversamente: su calidad) se calculará con la ecuación de **log-likelihood**; si tenemos un vector **y** binario con el éxito real de cada oportunidad y un vector **ŷ** con las probabilidades continuas, la función queda definida como:

$$\text{loss}(\mathbf{y}, \hat{\mathbf{y}}) = -\frac{1}{N}\sum_{i=1}^{N} y_i \log(\hat{y}_i) + (1 - y_i) \log(1 - \hat{y}_i)$$

Observamos que para cada elemento de la sumatoria el término que cuenta es el izquierdo si la oportunidad es exitosa y el derecho en caso contrario. La utilización del logaritmo suaviza la penalidad a la vez que tiene una derivada sencilla y facilita las cuentas. Hay un (-1) porque los logaritmos de números en el rango (0,1) son menores a cero y generalmente definimos la función objetivo como un error a minimizar. No es necesario que implementen la función ya que está incluida en los paquetes comunes de machine learning (por ejemplo en sklearn).

El trabajo práctico se desarrollará en el contexto de una competencia de Kaggle. Para esto deben tener dos conjuntos de datos:

- El de entrenamiento, sobre el cual ajustarán su/s modelo/s junto con sus hiperparámetros.
- El de test, sobre el cual realizarán predicciones y subirán los resultados a Kaggle. El score estará dado por la ecuación anterior.

Como mencionamos en clase, esperamos que resuelvan el problema de *target leakage* presente en el conjunto de datos.

Trabajo esperado

Se espera que **armen un pipeline completo** de machine learning experimentando en cada etapa. Esto quiere decir:

- Realizar *feature engineering*, encontrando codificaciones que sirvan, generando nuevos atributos, analizando cuáles atributos aportan y cuáles no.

- Probar modelos predictivos que crean que se encuadran al problema. Esto puede incluir realización de ensambles, también.

- Búsqueda de hiperparámetros óptimos de alguna manera automatizada.

Como sugerencia, recomendamos realizar una primera versión de cada etapa y que luego vayan en profundidad en cada una.

Además de los aspectos funcionales del código, también esperamos que **trabajen de manera científica**. Esto implica:

- Plantear preguntas o hipótesis y responderlas con experimentos y resultados. Las cosas no se hacen por que sí; debe haber una motivación basada en el conocimiento.

- Trabajo en equipo. Entendemos que la virtualidad y la asincronía complican la comunicación, pero en cualquier trabajo científico, en particular en los trabajos de aprendizaje automático, es fundamental la difusión de conocimiento entre las personas. Esto evita la redundancia de experimentos, genera un trabajo más cohesivo y hace que cada integrante tenga una visión más completa del problema.

- Realizar un poco de investigación del problema. En los apuntes de la materia, en las bibliotecas del lenguaje que utilizan, en internet. Esto no implica probar cosas a mansalva y sin criterio -- como mencionamos antes, debe haber una motivación o justificación.

- Interpretación de los resultados. Una vez que se realiza un experimento hay que considerar si funcionó como fue esperado o no y por qué. Los resultados negativos también son resultados. Es parte de la interpretación entender el problema en específico y encuadrar bien la solución.

- Presentación. Cualquier trabajo en ingeniería o ciencia requiere de una presentación a los colegas. Como pares de ustedes, esperamos que la comunicación que hagan del proceso y de los resultados se sustente en argumentos, datos y gráficos.

Competencia

En machine learning son comunes las competencias por Kaggle. La idea es que a medida que vayan avanzando con el trabajo y experimentando distintas cosas puedan ir subiendo sus predicciones a la plataforma. Esto les va a permitir ver su avance y lo hace más divertido. Para esto les pedimos que se registren en Kaggle y armen sus grupos ahí para hacer submits.

También nos permite entender cuánto trabajaron. Si hicieron pocos envíos y sólo durante la semana anterior a la entrega, no es un buen indicio.

El link a la competencia es https://www.kaggle.com/c/friofrio

Kaggle cuenta con un límite de 5 submits diarios, tener en cuenta esto y planificar acordemente las subidas.

Formato de entrega

La entrega se realiza en la primera semana de marzo. Tradicionalmente, se entrega un informe y luego tenemos una entrevista con cada grupo para charlar sobre el proceso, los resultados y el trabajo. Este cuatrimestre vamos a reemplazar la entrevista por un video corto de 10-15 minutos, aparte del informe.

Parte de la entrega también son los submits a Kaggle durante el trabajo. El puntaje en la competencia de Kaggle influye en la nota.

Informe

El informe se espera que sea detallado yendo a los experimentos individuales, resultados tabulados, visualizaciones y en general contando cómo fue el proceso de prueba e investigación que realizaron. Recuerden las cosas que esperamos del informe en la sección de Trabajo Esperado.

El informe debe contener un link al repositorio público para que podamos ver el trabajo que hicieron así como las contribuciones individuales.

Video

El video debe ser relativamente corto, de diez a quince minutos. No es necesaria una gran producción audiovisual; pueden grabarse presentando unas 3-4 transparencias por meet. Es importante que todos participen. Recuerden que cuando se presenta un trabajo grupal no hay un *yo*, sino un *nosotros*.

El objetivo del video es comunicarle a un/a colega suyo el trabajo que realizaron. Esto implica contar las cosas más importantes que hicieron a nivel *feature engineering*, construcción de modelos y/o ensambles y otras decisiones. Generalmente esto implica contar el mejor modelo, pero también puede haber otras conclusiones interesantes a contar aunque no fueron las que arrojaron el mejor resultado.