

# Tarea 2 - Introducción a la Ciencia de Datos

2024

Esta tarea es la continuación de la Tarea 1, por lo que se utilizarán los mismos datos y puede reutilizar cualquier parte del código.

En el repositorio intro-cd bajo la carpeta Tarea 2, se encuentra el notebook de referencia sobre el que deberá trabajar.

La entrega se debe dejar disponible en el mismo repositorio de la Tarea 1. Los archivos a evaluar deben estar en la branch principal (main). En dicha rama no debe haber commits posteriores a la fecha de entrega estipulada. Los archivos que deben estar presentes en el repositorio son:

- Un informe en formato PDF incluyendo todos los resultados relevantes, que será el trabajo a evaluar.
- Todo el código que haya sido implementado (al menos un notebook y posibles scripts adicionales), pero estos sólo serán revisados detalladamente en caso de que existan dudas referentes a la implementación. De todas maneras se evaluará la documentación de los notebooks, en el sentido que éstos deben ser comprendidos por otra persona, que por ejemplo, quisiera continuar o volver a correr los notebooks con los mismos datos.

Agregar un archivo [README.md](#) al repositorio, con indicaciones básicas, por ejemplo, indicando cuál es el informe de cada tarea, notebooks o scripts utilizados para responder las preguntas y en caso de haber más de uno, indicar para qué se usó cada uno.

Recuerde que el propósito de la tarea es poner en práctica algunos conceptos clave de aprendizaje automático, y no necesariamente profundizar en técnicas específicas de procesamiento de lenguaje natural. El foco debe estar en entender el proceso y la interpretación de los resultados a nivel conceptual, y no necesariamente en mejorar las métricas indefinidamente. Si los resultados no parecen buenos, la idea es generar hipótesis y buscar y formular posibles razones.

## Parte 1: Dataset y representación numérica de texto

Para esta parte, se utilizará como referencia la sección [Extracting features from text files](#) de la documentación oficial de scikit-learn.

1. En el notebook de la tarea, se crea un Dataset reducido de sólo 3 personajes. Se espera que utilice su propia versión de la función `clean_text()` de la Tarea 1. Parta los datos para generar un conjunto de test del 30% del total, utilizando muestreo estratificado.
  - Sugerencia: utilice el parámetro `stratify` de la función `train_test_split` de scikit-learn, fijando también `random_state` para obtener resultados reproducibles.
2. Genere una visualización que permita verificar que el balance de párrafos de cada personaje es similar en train y test.

3. Transforme el texto del conjunto de entrenamiento a la representación numérica (features) de conteo de palabras o bag of words. Explique brevemente cómo funciona esta técnica y muestre un ejemplo. En particular explique el tamaño de la matriz resultante, y la razón por la que es una sparse matrix.
  - Sugerencia: puede ser útil imaginar qué sucedería con la memoria RAM requerida si no estuviéramos trabajando con un conjunto de datos tan reducido.
4. Explique brevemente qué es un n-grama. Obtenga la representación numérica Term Frequency - Inverse Document Frequency. Explique brevemente en qué consiste esta transformación adicional.
5. Muestre en un mapa el conjunto de entrenamiento, utilizando las dos primeras componentes PCA sobre los vectores de tf-idf. Analice los resultados y compare qué sucede si utiliza el filtrado de `stop_words` para idioma inglés, el parámetro `use_idf=True` y `ngram_range=(1,2)`. Opcionalmente, también puede analizar qué sucede si no elimina los signos de puntuación.
  - ¿Se pueden separar los personajes utilizando sólo 2 componentes principales?
  - Haga una visualización que permita entender cómo varía la varianza explicada a medida que se agregan componentes (e.g: hasta 10 componentes).

## Parte 2: Entrenamiento y Evaluación de Modelos

1. Entrene el modelo Multinomial Naive Bayes, luego utilícelo para predecir sobre el conjunto de test, y reporte el valor de accuracy y la matriz de confusión. Reporte el valor de precision y recall para cada personaje. Explique cómo se relacionan estos valores con la matriz anterior.
  - ¿Qué problemas puede tener el hecho de mirar solamente el valor de accuracy?
  - Considere qué sucedería con esta métrica si el desbalance de datos fuera aún mayor entre personajes.
  - Sugerencia: utilice el método `from_predictions` de `ConfusionMatrixDisplay` para realizar la matriz.
2. Explique cómo funciona la técnica de validación cruzada o cross-validation. Interprete y complete el código de ejemplo para la búsqueda de hiper-parámetros. Genere una visualización que permita comparar las métricas (e.g: accuracy) de los distintos modelos entrenados, viendo el valor promedio y variabilidad de las mismas en todos los splits (e.g: en un gráfico de violín).
3. Elija el mejor modelo (mejores parámetros) y vuelva a entrenar sobre todo el conjunto de entrenamiento disponible (sin quitar datos para validación). Reporte el valor final de las métricas y la matriz de confusión. Discuta las limitaciones de utilizar un modelo basado en bag-of-words o tf-idf en cuanto al análisis de texto.
4. Evalúe al menos un modelo más (dentro de scikit-learn) aparte de Multinomial Naive Bayes para clasificar el texto utilizando las mismas features de texto. Explique brevemente cómo funciona y compare los resultados con el anterior.
5. Evalúe el problema cambiando al menos un personaje. En particular, observe el (des)balance de datos y los problemas que pueda generar, así como cualquier indicio que pueda ver en el mapeo previo con PCA. Puede ser útil comentar acerca de técnicas como sobre-muestreo y submuestreo, no es necesario implementarlo.
6. Busque información sobre al menos una técnica alternativa de extraer features de texto. Explique brevemente cómo funciona y qué tipo de diferencias esperaría en los resultados. No se espera que implemente nada en esta parte.

7. (Opcional) Entrenar el modelo de fasttext y comparar las predicciones en términos de accuracy y de la matriz de confusión, así como cualquier otra métrica que considere relevante. Mencione posibles ventajas y desventajas de utilizar este modelo.
  - [FastText](#)