**Puntos:**

1. Introducción, objetivo y explicación.
2. Análisis exploratorio.
3. Importación de librerías y archivo.
4. Inspección de tabla y limpieza, tipos de datos.
5. Análisis de columnas y valores, imputación, eliminación.
6. Gráficos, agrupaciones descriptivas.
7. Reportes (nuevo, librería especial Sweetviz).
8. Feature Engineering (si es necesario).
9. Over Sampling o UnderSampling. Preguntar a Tomas cuál usar
10. Dummificación.
11. Clustering (ni idea)
12. Métrica a usar/mejorar para el proyecto. Justificación.
13. Modelo KNN.
14. Modelo Naive Bayes (separar categóricas y numéricas, tienen tratamiento diferente).
15. Modelo Regresión Logística.
16. Modelo Árbol de decisión simple.
17. Modelo de ensamble.
18. Bagging
19. RandomForest
20. Boosting

* Gradient Boost
* ADA Boost
* XG Boost
* LightGBM

1. GridSearch en cada modelo.
2. Selección del mejor modelo.
3. Pipeline + GridSearch.
4. Feature importance.
5. PCA o TSN?
6. APIS y modelo real en BBDD.

Observaciones generales:

1. Comentar el código: qué se hace en cada bloque
2. Posible conclusión de lo obtenido.

Elementos importantes:

* En problemas de Churn siempre estará **desequilibrado** en favor de los casos que no abandonan.
* Las **importancias de las características** se pueden extraer de nuestros modelos para comprobar qué características han contribuido más para que el modelo decida si un cliente va a abandonar o no. Podemos caer en la **maldición de la dimensionalidad**.
* Elegir la(s) métrica(s) que realmente refleja el aporte de valor del modelo al negocio es clave y tiene que estar claro antes de empezar el diseño del conjunto de datos y el modelo. ¡Las métricas son bastante buenas! Un acierto (accuracy) del 85%! ¡Fantástico! O… bueno, pensemos un poco antes. ¿Es realmente el accuracy la métrica que queremos observar? ¿Qué quiere detectar nuestro negocio con éste modelo?

Esa es la pregunta que debemos hacernos a la hora de evaluar un modelo. En este caso, la detección de aquellos casos que SÍ abandonan, queremos ver **Precisión** **(¿cuándo predice positivos, que porcentaje clasifica correctamente? Significa que quiero tener un bajo número de falsos positivos**) De esta forma estaremos haciendo uso correcto de los recursos ($) para retener al cliente. También nos importa el recall, así que podríamos usar **score F1**. Parece que algunas de las métricas que habitualmente se usan para evaluar modelos (accuracy) de éste tipo pueden ser engañosas.