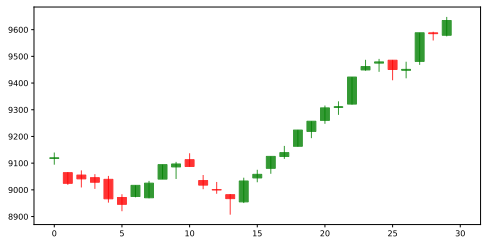
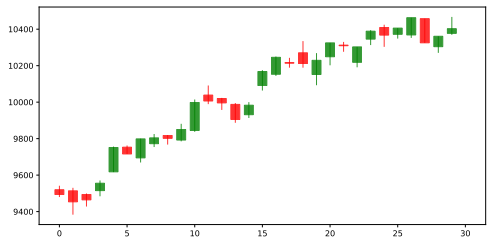
Objetivo

El objetivo de esta práctica es utilizar técnicas de aprendizaje automático para construir modelos que permitan reconocer patrones de análisis técnico dentro de una serie de precios. En su forma más sencilla estos modelos pueden plantearse como una tarea de clasificación binaria en la que el modelo es capaz de identificar:

1. Clase positiva: Existe un patrón en una ventana temporal. Ejemplo un doble suelo
2. Clase negativa: No existe el patrón de interés en la ventana temporal indicada



doble suelo (ejemplo positivo)



sin patrón (ejemplo negativo)

Para el desarrollo de la práctica planteamos 3 prototipos completos de dificultad incremental que permitirán al alumno ir evolucionando el modelo a la vez que toma contacto con los diferentes procesos de un workflow de minería de datos.

Materiales

- Series históricas del IBEX en formato pickle
- Un notebook de etiquetado de patrones, configurado para ventanas de 30 días

La práctica debe realizarse en notebooks de Python utilizando la librería *sklearn*. Se puede utilizar cualquier otra librería que se considere necesaria

Desarrollo

La práctica se compone de las siguientes partes:

Primera Iteración

En esta versión se requiere generar un modelo que clasifique un patrón de *doble suelo* y mostrar los resultados que muestren que dicho modelo es mejor que la aleatoriedad. Opcionalmente el alumno puede elegir un patrón técnico distinto que sea de su interés. Los pasos sugeridos son:

1. Etiquetar las ventanas temporales del IBEX como ejemplos positivos y negativos con el notebook de etiquetado. Se sugiere tener al menos unos 60 gráficos.
2. Extraer las series OHLC de las ventanas etiquetadas.

3. Generar características que se consideren interesantes para reconocer el patrón a partir de los datos de la ventana temporal (ver comentarios más adelante).

4. Entrenar un modelo de los vistos en clase haciendo una separación de los datos en train/test.

5. Reportar los resultados de evaluación que pueden incluirse en un notebook con el código para generarlos. La evaluación debe incluir:

   (a) Una matriz de confusión

   (b) Resultado de accuracy y AUROC

   (c) Una curva ROC

6. Reportar gráficamente la importancia de las características generadas, utilizando la importancia por permutación.

Sobre la generación de características se recomienda generar funciones sencillas que calculen valores que ayuden a distinguir unas series de otras, independientemente del patrón que se esté buscando. Como los ventanas tendrán distinta escala de precios, es necesario que las características estén normalizadas. Algunos ejemplos pueden ser:

- Número de días que pasan entre el máximo y el mínimo de la ventana
- Si ocurre primero el máximo o el mínimo
- Número de veces en la ventana que se llega a cierto umbral

El desarrollo de la práctica hasta esta iteración supone una nota máxima de 5 puntos.

Segunda Iteración

En esta parte se pide abordar el problema como una clasificación multiclase para que se puedan reconocer a la vez varias figuras técnicas. Un esquema sencillo sería tener por ejemplo doble suelo, doble techo y sin patrón. Las tareas sugeridas son:

1. Realizar los pasos del 1 al 3 de la primera iteración, pero teniendo las consideraciones requeridas para el problema multiclase.

2. Entrenar clasificadores utilizando 3 algoritmos diferentes, y elegir el mejor en función del accuracy estimado con una validación cruzada.

3. Reportar la evaluación del mejor modelo sobre un conjunto de test separado. Aquí se debe incluir precisión y sensibilidad, por clase y en sus versiones *micro-average* y *macro-average*.

4. Utilizar el modelo para identificar 5 patrones de cada clase en la serie histórica del EuroStoxx. Por ejemplo, se puede iterar sobre las ventanas temporales y elegir las de mayor confianza en la predicción de cada clase.

El desarrollo de la práctica hasta esta iteración supone una nota máxima de 8 puntos.

Tercera Iteración

En esta parte se abordará el problema desde un enfoque de aprendizaje activo. Dado que el etiquetado es un proceso manual y relativamente lento, planteamos utilizar el clasificador de la primera iteración, y re-entrenarlo utilizando ejemplos en los que no esté clara su frontera de decisión, pero que nosotros podemos dar la etiqueta real a modo de oráculo. Los pasos sugeridos son:

1. Entrenar un clasificador con el conjunto etiquetado en la primera iteración guardando un subconjunto para test.

2. Desarrollar un bucle que dada una ventana aleatoria prediga la probabilidad de ser doble suelo, almacenando aquellas ventanas con un umbral cercano a la frontera de decisión (ej. $[0.4, 0.6]$.

3. Etiquetar correctamente las ventanas almacenadas, en el propio bucle o como proceso separado

4. Re-entrenar el clasificador agregando los nuevos ejemplos a los originales y determinar la mejora en las métricas de evaluación

5. Repetir los pasos del 2 al 4 por lo menos 2 iteraciones para determinar si es posible seguir mejorando el clasificador.

En la generación de ventanas aleatorias de este caso se debe evitar que la seleccionada no esté en el conjunto de test, ni en sub-conjuntos previamente etiquetados. El desarrollo completo de la práctica supone una nota máxima de 10 puntos.

Entrega

La práctica debe realizarse de forma individual. Se puede entregar hasta el **día 30 de julio de 2022**. La entrega es un fichero empaquetado .zip por la plataforma de Aula Virtual. El fichero debe nombrarse con el nombre y los apellidos del alumno. El fichero entregado debe contener:

1. Los notebooks que generan los modelos y reportan los resultados de la práctica.

2. Las series de las ventanas etiquetadas manualmente, guardadas en formato CSV.

3. Una subcarpeta con las gráficas predichas para el EUROSTOXX (último apartado de la segunda parte).