AWAKELAB

Plan Formativo: Ciencia de Datos	Nivel de Dificultad
Módulo 5: Aprendizaje supervisado	Medio
Tema: Random forest	

Intención del aprendizaje o aprendizaje esperado:
• 6. Elaborar un modelo predictivo aplicando el algoritmo Random Forest para resolver un problema de clasificación utilizando lenguaje Python

Ejercicio planteado
El problema que abordaremos es predecir la temperatura máxima para mañana en una determinada ciudad usando un año de datos meteorológicos anteriores. Usted usará Seattle, WA, pero también puede buscar datos para otra ciudad usando la herramienta de datos climáticos en línea de la NOAA. Tenemos acceso a un año de temperaturas máximas históricas, las temperaturas de los dos días anteriores y una estimación de un amigo que siempre afirma saberlo todo sobre el clima. Este es un problema de **aprendizaje automático de regresión supervisado.** Durante el entrenamiento, para usar random forest debemos darle tanto las características como los objetivos, y debe aprender a asignar los datos a una predicción. Para esto usará los datos meteorológicos para Seattle, WA de 2016 utilizando la herramienta de datos climáticos en línea de la NOAA. En general, alrededor del 80 % del tiempo dedicado al análisis de datos se dedica a la limpieza y recuperación de datos, pero esta carga de trabajo se puede reducir encontrando fuentes de datos de alta calidad. La herramienta NOAA es

AWAKELAB

sorprendentemente fácil de usar y los datos de temperatura se pueden descargar como archivos csv limpios que se pueden analizar. El archivo de trabajo se llama `temps.csv` que encontrará cargado en la plataforma.

Usted posee las siguientes variables para trabajar:

- year: 2016 para todos los puntos de datos
- month: número del mes del año
- day: número para el día del año
- week: día de la semana como cadena de caracteres
- temp_2: temperatura máxima 2 días antes
- temp_1: temperatura máxima 1 día antes
- average: temperatura máxima promedio histórica
- actual: medición de temperatura máxima
- friend: la predicción de tu amigo, un número aleatorio entre 20 por debajo del promedio y 20 por encima del promedio

Realice lo siguiente:

1. Para identificar anomalías, obtenga un resumen estadístico, o dicho de otro modo, realice un análisis descriptivo de sus datos, comente.
2. Realice un análisis gráfico de las temperaturas máximas históricas, las temperaturas de los dos días anteriores y la estimación del amigo, con respecto a la fecha. Comente si hay anomalías.

Preparación de la data:

3. Recodifique los días de la semana, creando siete variables 1-0 asociados a los días de la semana. Le puede facilitar el proceso usando `get_dummies` de la librería Pandas.

	year	month	day	temp_2	temp_1	average	actual	friend	week_Fri	week_Mon	week_Sat	week_Sun	week_Thurs	week_Tues	week_Wed
0	2016	1	1	45	45	45.6	45	29	1	0	0	0	0	0	0
1	2016	1	2	44	45	45.7	44	61	0	0	1	0	0	0	0
2	2016	1	3	45	44	45.8	41	56	0	0	0	1	0	0	0
3	2016	1	4	44	41	45.9	40	53	0	1	0	0	0	0	0
4	2016	1	5	41	40	46.0	44	41	0	0	0	0	0	1	0

4. Ahora, separe los datos en predictores y variable respuesta. Su variable Y es la **temperatura máxima real** y los predictores son todas las

columnas que usa el modelo para hacer una predicción. Asegúrese de transformar en una matriz Numpy porque esa es la forma en que funciona el algoritmo.

5. Separe su base en entrenamiento y testeo en 75/25. Use una semilla (`random_state = 42`) para obtener los mismos valores.
6. Cree una línea de base, es decir, el error que obtendremos si simplemente predijéramos la temperatura máxima promedio para todos los días.

Ajuste del modelo:

7. Ajuste un modelo de regresión de bosque aleatorio con los datos de entrenamiento. Implemente 1000 árboles y use la misma semilla declarada en el punto 5.
8. Obtenga la predicción. Coméntalo.
9. Obtenga las métricas de precisión.
10. ¿Considera que el modelo es adecuado?

Caso
APRENDIZAJE DE MÁQUINA SUPERVISADO
Preguntas guía
Recursos Bibliográficos:
[1] Entendiendo Random Forest https://www.analyticsvidhya.com/blog/2021/06/understanding-random-forest/

AWAKELAB

[2] Random Forest

https://www.bigdata-insider.de/was-ist-random-forest-a-913937/

[3] Ejemplos en py

https://programmerclick.com/article/84431135962/