PROYECTO BIG DATA ALERTAQUA

Autores

Juan Carlos Sánchez Alcañiz

José Jesús Torregrosa García

Fecha de realización

27/06/2024

INDICE

1. INTRODUCCIÓN

La idea inicial era hacer un proyecto de detección de fraudes en el consumo de agua, pero al no poder conseguir ningún conjunto de datos sobre este tema, se eligieron otros datos de *El portal oficial de datos europeos* - https://data.europa.eu/ . En concreto, de su página web https://data.europa.eu/data/datasets/dat-163-en?locale=es se escogió el conjunto de datos **Calidad de la base de agua-Agua (archivos CSV).**

Pinchando en el enlace de descarga de ese conjunto de datos, se obtiene el fichero **Waterbase_v2018_1_WISE4_csv.zip.** Este archivo zip contiene el fichero usado para este trabajo: **DisaggregatedData.csv**. Fichero que contiene datos de toma de muestras de mediciones de agua en Europa con diversa información.

Los datos pertenecen a la *Agencia del entorno europeo* (*European Environment Agency* - *https://data.europa.eu/data/catalogues/eea?locale=es/)*

2. OBJETIVOS DEL PROYECTO

Los objetivos generales del proyecto son los siguientes:

- Practicar, conocer, investigar y probar lo aprendido en el curso recibido sobre Machine Learning e Inteligencia Artificial, en concreto en exploración de datos, transformación de datos, y construcción y validación de modelos con distintas técnicas de Inteligencia Artificial
- Trabajar en equipo, y conocer Colab para la realización de proyectos Big Data en equipos

Los objetivos concretos son, usando el conjunto de datos seleccionado **DisaggregatedData.csv**.

- Explorar los datos, y tratar los datos
- Elegir estrategias de análisis de datos para el conjunto de datos seleccionado
- Realizar el análisis de datos, usando distintos modelos de Inteligencia Artificial
- Sacar conclusiones de la ejecución de los diferentes modelos aplicados

3. EJECUCIÓN DEL PROYECTO

La ejecución del proyecto se describe a continuación: exploración, tratamiento, y estrategias de análisis de los datos

3.1. EXPLORACIÓN DE LOS DATOS

Descripción del proceso

Se realizan las siguientes tareas en Power BI

- Identificación de columnas que deberán de ser eliminadas más adelante en la siguiente fase porque no son necesarias para la casuística del problema
- Conteo de valores de una columna: haciendo conteos de los valores únicos para ver si el contenido de los campos coincide a priori con lo que dice su descripción.

Resultado

Se piensa en hacer un modelo de clasificación y un modelo de regresión. Se identifican las siguientes columnas que nos interesan del dataset:

- monitoringSiteIdentifier: Identificador internacional único del sitio de monitoreo en el que se obtuvieron los datos. Las dos primeras letras de este campo dicen el país donde se ha tomado la muestra. Los códigos de esas dos letras se pueden encontrar en: https://dd.eionet.europa.eu/vocabulary/common/countries
- parameterWaterBodyCategory: códigos que dicen el cuerpo del agua (rio, lago, costa, etc) http://dd.eionet.europa.eu/vocabulary/wise/WFDWaterBodyCategory/
- observedPropertyDeterminandCode: código del compuesto de la muestra tomada
- procedureAnalysedMedia: tipo de medio analizado
- resultUom: unidad de medida de la muestra
- phenomenonTimeSamplingDate: fecha en que se tomó la muestra
- resultObservedValue: valor observado de la muestra
- resultQualityObservedValueBelowLOQ: booleano, si la columna resultObservedValue está por debajo de un valor que sea cuantificable para esa sustancia (superior al detectable)
- procedureLOQValue: es la menor cantidad de compuesto que puedes detectar y medir en el agua con un 95% de certeza de que el valor medido es correcto y no una falsa detección o ruido de fondo
- procedureAnalyticalMethod: Código CEN/ISO del método analítico.
- parameterSampleDepth: Profundidad a la que se tomó la muestra, en metros por debajo de la superficie del agua
- resultObservationStatus: "A" registro confirmado como correcto, el reto de datos (L,M,N,O,W,X,Y) datos faltantes. Nos interesan los registros correctos
- metadata_observationStatus: Estado del registro con respecto a su confiabilidad. Es una columna metadata que para el análisis no nos va a servir, pero para un filtrado de los registros de datos confiables si (valor "A").

Columnas que no nos interesan y que serán candidatas a ser eliminadas en pasos posteriores:

- ~~monitoringSiteIdentifierScheme~~: Nomenclatura que sigue monitoringSiteIdentifier.
- ~~procedureAnalysedFraction~~: que porcentaje de la muestra fue analizada.
- ~~Remarks, metadata_versionId, metadata_beginLifeSpanVersion, metadata_statusCode, metadata_statements, UID~~: columnas de metadatos de los sistemas informáticos. No nos interesan

Se piensa en hacer cuatro de modelos:

- Un modelo que prediga el valor de resultObservedValue para el año 2018, y los datos anterior al 2018 que sirvan para un entrenamiento. El tipo de modelo será un Random Forest de Regresión
- Un modelo que prediga el valor de resultObservedValue para el año 2018, y los datos anterior al 2018 que sirvan para un entrenamiento. El tipo de modelo será una red neuronal
- Un modelo que clasifique el valor de resultQualityObservedValueBelowLOQ para el año 2018, y los datos anterior al 2018 que sirvan para un entrenamiento. El tipo de modelo será Random Forest de Clasificación
- Un modelo que clasifique el valor de resultQualityObservedValueBelowLOQ para el año 2018, y los datos anterior al 2018 que sirvan para un entrenamiento. El tipo de modelo será una red neuronal

3.2. TRATAMIENTO DE LOS DATOS

Se realizan los dos programas que siguen a continuación en Python

- **Programa_1_filtrado_ALERTAQUA.ipynb**

Descripción del proceso

- Limpieza de registros no válidos para la casuística del problema.
- Eliminar columnas(campos de la tabla) que no nos sirven para el análisis.
- Separación en dos subconjuntos del conjunto de datos filtrados: train(entrenamiento) y test(comprobación). De esa forma en futuros tratamientos de los datos podremos comprobar si hay valores en el train en y no en el test, y viceversa.

Resultado

- Nos quedamos con los registros metadata_observationStatus =”A”
- Nos quedamos con los registros procedureAnalyticalMethod distintos a”-” y no nulos. Para hacer una predicción o clasificación el método analítico tiene que estar definido. Se cree que la falta de este dato indica que el análisis de la sustancia sigue las normas que cada vez se exigen mas en la toma de muestras.
- parameterSampleDepth: se seleccionan los registros con valores no negativos. Se supone que el valor negativo es porque se recoge por encima de la superficie, sin embargo el modelo es para aguas dentro del elemento acuífero.
- metadata_statusCode: Los valores ‘experimental’ (pruebas de datos) no se recogen, son ficticios, se eliminan
-
- Después del filtrado, se eliminan estas columnas, como se dedujo en el punto EXPLORACIÓN DE LOS DATOS: 'monitoringSiteIdentifierScheme', ‘procedureAnalysedFraction', 'Remarks', 'metadata_versionId', 'metadata_beginLifeSpanVersion', 'metadata_statusCode','metadata_observationStatus','UID'
-
- Se exportan los datos a los ficheros intermedios:
-
    - fichero_filtrado_1.csv (copia de seguridad)
    - Entrenamiento_hasta_2017.csv (fichero de entrenamiento)
    - Test_2018.csv (fichero de predicción)

- **Programa 2_Simplificar_train_y_test.ipynb**

Descripción del proceso

- Columnas que deben de tener el mismo valor para todos los registros, 1º se filtran por ese valor solo los registros que nos serán útiles y 2º se elimina las columnas.
- Eliminar columnas que ya son necesarias

Resultado

- Se eliminan las columnas ProcedureAnalysedMedia (después de los filtrados tiene siempre valor ‘water’, que es lo que buscamos)
- Filtrar filas donde resultObservationStatus es igual a 'A' (registro confirmado como correcto). Una vez filtrado para aligerar el modelo se elimina la columna
- Los valores nulos de procedureLOQValue , se ponen a 0, por si diese problemas en el modelo
- Se filtran filas donde procedureLOQValue <= resultObservedValue (para no llevar a engaño las muestras que queremos predecir deben de ser correctas, puesto que buscamos la veracidad del dato en el entorno natural)
- La columna metadata_statements se tenía que haber eliminado anteriormente. Se elimina en este script
- Interesa el estudio realizado del campo resultUom, desde el punto de vista académico y profesional, en el train y en el test

- Se exportan los datos a los ficheros intermedios:

    - Entrenamiento_hasta_2017_(simplificado).csv (fichero de entrenamiento)
    - Test_2018_(simplificado).csv (fichero de predicción)

3.3. ESTRATEGIAS DE ANÁLISIS DE LOS DATOS

Se realizan los siguientes programas en Python

- **Programa_3_Pre_modelos.ipynb**

Descripción del proceso

- Se calculan los valores únicos de cada columna para investigar, así como columnas categóricas con las que tuviésemos problemas computacionales mas adelante

- Es un script en Python cuyo objetivo es "ablandar los datos de las categorías" antes de que vengan los modelos vayan a crear variables dummy.
- Columnas que influyen en el modelo, con un valor único tanto en el fichero train como en el test, no se deben de eliminar por si el modelo se guarda y se reutiliza con otro conjunto de datos que tengan más valores para esa columna.

Resultado

- Se identifican las variables categóricas para convertirlas a dummy en un futuro:

    - *Análisis pre-dummy 1º:* Por falta de recursos computacionales en fases posteriores, y la complejidad de convertirla en variable dummy, se reduce el conjunto de datos. Se determina que para poder realizar la consecución del proyecto se acota el problema al país de Luxemburgo por los costes temporales. Para ello se investiga la complejidad del campo monitoringSiteIdentifier, en donde las dos primeras letras son las que identifican el país, y el resto la numeración del lugar. Se filtra por esos dos primeros caracteres

    - *Análisis pre-dummy 2º:* Analizamos la cantidad de posibles valores de parameterWaterBodyCategory. No debemos de eliminar el campo aunque tenga un solo valor (RW – Agua de río, River Water), este modelo si se guarda y se usa mas adelante con otros datos, podrían aparecer mas valores, y el modelo debe de estar preparado para ello.

    - *Análisis pre-dummy 3º:* Analizamos la cantidad de posibles valores de observedPropertyDeterminandCode, y vemos que tiene los valores adecuados para convertirla en dummy.

    - *Análisis pre-dummy 4º:* procedureAnalyticalMethod, se podrá convertir en variable dummy

- Se exportan los datos a los ficheros intermedios:

    - Entrenamiento_hasta_2017_(simplificado).csv (fichero de entrenamiento)
    - Test_2018_(simplificado).csv (fichero de predicción)

- Para la consecución del objetivo se eligen los siguientes modelos de regresión y clasificación:
-

- **Random_Forest_(CLASIFICACION):** Actuará sobre la columna resultQualityObservedValueBelowLOQ
- **Random_Forest_(REGRESION):):** Actuará sobre la columna resultObservedValue
- **Red_Neuronal_(REGRESION):** Actuará sobre la columna resultObservedValue
- **Red_Neuronal_(CLASIFICACIÓN):** Actuará sobre la columna resultQualityObservedValueBelowLOQ

Los veremos a continuación

3.3.1. CLASIFICACIÓN POR RANDOM FOREST

- **Programa 4_1_Random_Forest_(CLASIFICACION).ipynb**

Descripción del proceso

- Preparamos las variables categóricas en variables dummy permitiendo al multicolinealidad
- Se crea y entrena el modelo de Random_Forest para clasificación
- Se hacen predicciones sobre el conjunto de prueba y se calcula la precisión del modelo
- Se calcula la matriz de confusión y el informe de clasificación
- Se identifican las predicciones incorrectas

Resultado

- Parece que identifica mejor la clase 1 (verdadero) que la clase 0 (los falsos)

- La clase 0 (los falsos), aunque sea la minoritaria, si en algún momento se incrementa el modelo puede dejar de predecir correctamente. El recall indica que hay peso de arrastre provocado por los fallos de la clase 0(los falsos).

3.3.2. REGRESIÓN POR RANDOM FOREST

- **Programa_4_2_Random_Forest_(REGRESION).ipynb**

Descripción del proceso

- Preparamos las variables categóricas en variables dummy permitiendo al multicolinealidad
- Se crea y entrena el modelo de Random_Forest para regresión y predecir el valor de resultObservedValue
- Se hacen predicciones sobre el conjunto de prueba y se calcula la precisión del modelo
- Se calculan el error cuadrático medios y absoluto, con el coeficiente de determinación R^2
- Se calcula el MSE de Validación Cruzada, que permitirá seleccionar un posible mejor modelo usando GridSearch
- Se detectarán las columnas mas importantes a la hora de influir en la predicción de los datos

Resultado

- Es un modelo que tiene una alta predicción a los datos con los que ha trabajado, y se ha detectado unas columnas que determinan la predicción de la variable objetivo, que en este caso es resultObservedValue

3.3.3. REGRESIÓN POR RED NEURONAL

- **Programa_4_3_Red_Neuronal_(REGRESION).ipynb**

Descripción del proceso

- Preparamos las variables categóricas en variables dummy permitiendo al multicolinealidad
- Se crea y entrena la red neuronal una Red Neuronal Multicapa (Multilayer Perceptron, MLP)para regresión y predecir el valor de resultObservedValue. Se ha utilizado este tipo de red neuronal, porque es aprendizaje supervisado, y es una buena introducción a las redes neuronales

Resultado

- El modelo tiene unos resultados muy parecidos al del modelo anterior de Random Forest de Regresión, aunque por poco este le supera.
-

3.3.4. CLASIFICACIÓN POR RED NEURONAL

No se ha podido realizar por falta de tiempo

4. PROBLEMAS ENCONTRADOS

- Situación complicada por el trabajo laboral en el mes de Junio, junto con temas personales.
- Equipo de trabajo de solo dos personas para un tema tan extenso
- Complejidad del dataset en el filtrado para ser el 1º proyecto realizado
- Falta de experiencia, es el primer proyecto que se ha realizado de este tipo
- Encontrar el dataset para hacer el proyecto llevó 2 semanas. Reconstrucción del proyecto en base a las características del dataset (queríamos detección de fraudes y no fue posible)
- El tamaño del dataset para ser trabajado en Colab ha sido tan grande, que ha afectado en todas las partes del proyecto
- Falta de recursos de computación en Colab

5. RESULTADOS

Los resultados del proyecto se van a describir desde dos puntos de vista diferentes:

5.1. RESULTADOS EN FUNCIÓN DE LOS OBJETIVOS DEL PROYECTO

Se han cumplido (√) los siguientes objetivos descritos en el apartado **2. OBJETIVOS DEL PROYECTO.** Los objetivos generales y concretos conseguidos del proyecto son los siguientes:

- √ Practicar, conocer, investigar y probar lo aprendido en el curso recibido sobre Machine Learning e Inteligencia Artificial, en concreto en exploración de datos, transformación de datos, y construcción y validación de modelos con distintas técnicas de Inteligencia Artificial
- √ Trabajar en equipo, y conocer Colab para la realización de proyectos Big Data en equipos
- √ Explorar los datos, y tratar los datos
- √ Elegir estrategias de análisis de datos para el conjunto de datos seleccionado
- √ Realizar el análisis de datos, usando distintos modelos de Inteligencia Artificial
- √ Sacar conclusiones de la ejecución de los diferentes modelos aplicados

5.2. RESULTADOS EN FUNCIÓN DE LOS DATOS ANALIZADOS EN EL APARTADO 3. EJECUCIÓN DEL PROYECTO

Se han cumplido (√) los siguientes objetivos descritos en el apartado **3. EJECUCIÓN DEL PROYECTO.**

- √ CLASIFICACIÓN POR RANDOM FOREST
- √ REGRESIÓN POR RANDOM FOREST
- √ REGRESIÓN POR REDES NEURONALES

No se ha cumplido (X) el siguiente objetivo por falta de tiempo:

- X CLASIFICACIÓN POR REDES NEURONALES

Referente a la viabilidad de los modelos obtenidos, se tiene que recalcar lo siguiente:

- Los modelos son válidos para el subconjunto escogido (Luxemburgo)
- Podrían exportarse e ir probándose en un conjunto mayor en Europa
- Se debería de hacer también una selección de años, especialmente de los últimos

6. CONCLUSIONES

Este proyecto ha sido un aprendizaje incremental , y ha costado mucho enfocarlo en el tratamiento y exploración de los datos, porque ha sido complicado para ser el primer proyecto.

Se ha entendido las diferencias entre los distintos modelos, y los objetivos a cumplir en ellos, así como una buena introducción valiosa en el mundo del Big Data. Ha sido una experiencia gratamente enriqucedora