`PAC 4: Optimización

Presentación

Una empresa de distribución minorista quiere organizar su área geográfica de operación para dividirla en zonas relativamente compactas y con una cantidad de población similar. Los datos de las que disponen son un listado de municipios, con el número de habitantes y su posición geográfica (latitud y longitud). Dado que el número de agentes de venta es fijo, quieren dividir el área total en un número fijo de zonas.

Los criterios principales son que todas las zonas tengan una población similar y que la distancia entre los municipios de cada zona sea lo más reducida posible. La empresa pide una subdivisión de los municipios en zonas de operación con estas propiedades.

Competencias

En este enunciado se trabajan en un determinado grado las siguientes competencias general de máster:

- Capacidad para proyectar, calcular y diseñar productos, procesos e instalaciones en todos los ámbitos de la ingeniería en informática.
- Capacidad para el modelado matemático, cálculo y simulación en centros tecnológicos y de ingeniería de empresa, particularmente en tareas de investigación, desarrollo e innovación en todos los ámbitos relacionados con la ingeniería en informática
- Capacidad para la aplicación de los conocimientos adquiridos y de solucionar problemas en entornos nuevos o poco conocidos dentro de contextos más amplios y multidisciplinares, siendo capaces de integrar estos conocimientos.
- Poseer habilidades para el aprendizaje continuado, autodirigido y autónomo.
- Capacidad para modelar, diseñar, definir la arquitectura, implantar, gestionar, operar, administrar y mantener aplicaciones, redes, sistemas, servicios y contenidos informáticos.
- Capacidad para asegurar, gestionar, auditar y certificar la calidad de los desarrollos, procesos, sistemas, servicios, aplicaciones y productos informáticos.
- Las competencias específicas de esta asignatura que se trabajan son:
- Entender que es el aprendizaje automático en el contexto de la Inteligencia Artificial
- Distinguir entre los diferentes tipos y métodos de aprendizaje
- Aplicar las técnicas estudiadas en un caso concreto

Objetivos

En esta PAC se practicarán los conceptos del temario relacionados con optimización, en una vertiente práctica con un caso concreto.

Descripción de la PAC/práctica a realizar

Datos

La empresa que encarga este trabajo proporciona un fichero municipis.data con una línea por cada municipio en el área geográfica de operación. Por cada municipio hay cuatro columnas: el identificador de municipio (un número entero), el número de habitantes, la latitud y la longitud.

Actividad 1

Defina una estructura de datos adecuada para representar una solución a este problema. Escribir una función objetivo que mida la calidad de una solución teniendo en cuenta los criterios indicados por la empresa:

- Mismo número de habitantes por zona.
- Suma de las distancias entre todos los municipios de una zona (cuanto más pequeña sea, mejor).

Ambos criterios son igual de importantes. Tenga en cuenta la escala de los valores obtenidos para definir una función objetivo equilibrada entre ambos criterios. El número de zonas que se quieren definir es 10.

Representación de la solución

Como la solución que se pide es una manera de agrupar valores para satisfacer unos criterios, nos encontramos con un problema de agrupamiento (clustering), aunque en aplicarse algunas restricciones muy particulares no es posible aplicar los algoritmos de clustering habituales. Por lo tanto lo que se quiere encontrar es el agrupamiento que obtenga los mejores resultados según los criterios antes expuestos.

Por este motivo, una representación de la solución puede ser, simplemente, una lista con un elemento por cada municipio (es decir, 100), donde el elemento y tenga como valor el grupo al que pertenece el municipio y, es decir un número entre 0 y 9 (porque nos piden 10 grupos).

[0, 1, 4, 4, 1, 2, 4, 24, 5, 6, 2, 5, 7, 8, 1, 9, 8, 9, ...]

Además, con el fin de cargar los datos del fichero municipis.data y almacenarlas de forma práctica para su uso, se utilizará una estructura de datos de tipo diccionario de la forma:

{Municipio: (habitantes, latitud, longitud)}

La función para cargar el archivo de datos y el resto de código descrito en este apartado se encuentra el archivo apartat1.py.

Medida de los dos criterios de calidad

Se define una función para medir cada uno de los dos criterios de calidad respecto a una solución determinada. Estas funciones se utilizarán más adelante para calcular y ajustar la función objetivo del problema.

Función diferenciaHabitantes (): dada una solución y el diccionario con los datos de los municipios, calcula la desviación típica al número de habitantes de los grupos.

Función distanciaMaxima (): dada una solución y el diccionario con los datos de los municipios, calcula la distancia máxima entre los municipios de cada grupo y devuelve la media de estas distancias.

A

La función objetivo para la optimización debe tener en cuenta los dos criterios indicados de manera que ambos tengan un peso similar. Esto en principio es complicado porque no es fácil determinar el rango de valores que pueden tomar las dos medidas (mismo número de habitantes por zona y suma de las distancias entre los municipios).

Aunque se podrían establecer unos mínimos y máximos teóricos (mínimo de cero y máximos para el caso de que hubiera un grupo con todos los municipios y los otros huecos), posiblemente estos límites no serán útiles porque pertenecen a casos muy atípicos que en la práctica no suelen aparecer.

Es más adecuado estudiar cuáles son los valores de estos parámetros con agrupamientos "típicos" y entonces aplicar algún tipo de corrección a partir de estos valores típicos.

¿Qué podemos considerar un agrupamiento típico? Pues es posible generar soluciones aleatorias y usarlas para tener una idea del valor típico que pueden tomar estos valores.

Si este proceso se repite con un número suficiente de soluciones y se calcula el valor medio se puede obtener una estimación razonablemente buena de los valores típicos de estas funciones.

Al código del apartado 1 se estiman estos valores con 1.000 soluciones de prueba, y se obtienen los siguientes valores medios:

Diferencia de habitantes: 4.891,17

Distancia máxima: 87,77

Entonces cuando la función objetivo mida la diferencia de habitantes o la distancia media de una solución, dividirá cada valor obtenido por la correspondiente estimación media, para que ambos criterios de calidad tengan un peso similar al evaluar una posible solución. El resultado final de la

función objetivo será la suma de estos dos valores corregidos, y se desea que el método de optimización minimice esta función objetivo.

$$objetivo(s) = \frac{diferenciaHabitantes(s)}{estimaciónDifHabitantes} + \frac{distanciaMaxima(s)}{estimaciónDistanciaMáxima}$$

Tenga en cuenta que, posiblemente, no será posible encontrar una solución por la cual la función objetivo valga cero, para que no sea posible anular ninguno de los dos criterios de calidad. El objetivo de la optimización debe ser, en todo caso, intentar encontrar una solución con un valor tan bajo como sea posible por la funcionaria objetivo.

Actividad 2

Encuentre una solución a este problema utilizando algoritmos genéticos. Define qué formato tendrá cada individuo. Pruebe con diferentes números de individuos (10, 20, 50, 100) y diferente número de iteraciones (50, 100, 200). Analizar los resultados en función de estos parámetros.

Tomando como base el código 5.2 de los materiales (algoritmos genéticos), y haciendo algunas modificaciones, se puede resolver este apartado. Como función objetivo se utilizará la función definida en el apartado anterior. El código de este apartado se puede encontrar en el archivo apartat.py.

Las modificaciones necesarias al código base de algoritmos genéticos son las siguientes.

Función cruza (): se debe eliminar la llamada a ajustaCapital ().

Función muta (): se debe eliminar la llamada a ajustaCapital (). Ahora las mutaciones son cambios de grupo de uno o más municipios.

Función evoluciona (): recibe más parámetros (municipios, estimaciones medias criterios calidad). Sustituye la llamada a rendimiento () para la llamada objetivo () con los parámetros adecuados. Ahora la función objetivo se debe minimizar (en el ejemplo se debía maximizar), por eso la ordenación de los individuos debe ser descendente.

Los valores obtenidos por el algoritmo genético para diferente número de iteraciones son los siguientes:

Tenga en cuenta que, posiblemente, no será posible encontrar una solución por la cual la función objetivo valga cero, para que no sea posible anular ninguno de los dos criterios de calidad. El objetivo de la optimización debe ser, en todo caso, intentar encontrar una solución con un valor tan bajo como sea posible por la función objetivo.

Població / Iteracions	50	100	200	300	500
10	1,1264	1,0226	0,9302	0,9211	**0,9121**
20	1,1463	1,1077	0,9935	0,9468	0,9574
50	1,0059	1,0124	0,9643	0,9803	0,9266
100	0,9962	1,0571	0,9682	0,9478	0,9303

Como puede observarse, aunque en general ejecutar un mayor número de iteraciones da mejores resultados, a veces con más iteraciones los resultados son peores. Esto es debido al azar inherente a todo el proceso, donde puede que con menos iteraciones o una población más reducida se encuentre una solución mejor.

En general a partir de 500 iteraciones los resultados no mejoraban o puede ser empeoraban, por este motivo no se han mostrado. En general se ve una mejora clara de los resultados hasta 200 iteraciones, pero más adelante ya no.

De hecho se puede comprobar que el mejor resultado, en este caso, se obtiene con la población más pequeña (10 individuos). Esto puede estar causado por la influencia del elitismo: como en este código el mejor individuo pasa directamente a la siguiente generación, la importancia de esta técnica es mayor en poblaciones pequeñas. Se podría probar, pues, con un elitismo más grande (un 10% de la población, por ejemplo) y comparar el efecto.

Actividad 3

Encuentre una solución a este problema utilizando otro método de optimización de los materiales de la asignatura. Justifique su elección. Pruebe con diferentes configuraciones del método, en su caso.

Por este apartado se ha decidido probar con la recocido simulado para poder comparar un método muy costoso en tiempo de ejecución (algoritmos genéticos) con otro con un coste muy bajo. De esta manera se puede estudiar si el hecho de probar muchas soluciones a los algoritmos genéticos suponen una ventaja real o no.

Se deben hacer algunas modificaciones al código (5.1), como sustituir las llamadas a la función tempsMitjaEspera () para llamadas a la función objetivo (), y reescribir la función generaVei () para adaptarla a este problema. En general no son necesarios muchos cambios. El código correspondiente se encuentra en el archivo apartat3.py.

En la tabla siguiente se pueden ver los resultados para diferente número de iteraciones y diferentes valores de tolerancia.

Tolerància / Iteracions	100	500	1000	2000	3000
1.0	1,5444	1,8051	1,6318	1,3313	1,4524
0.5	1,8135	1,4995	1,8031	**1,2030**	1,5101
0.1	1,6979	1,5801	1,5375	1,7696	1,3947

Como se puede ver, a este problema no vale la pena ir más allá de 2000 iteraciones. El valor de la tolerancia no afecta significativamente los resultados, siendo 0.5 el que funciona mejor en este caso.

Actividad 4

Comparar los resultados de las actividades 2 y 3. Justificar las diferencias.

Como se puede ver en las tablas de resultados de los apartados anteriores, los algoritmos genéticos dan resultados bastante mejores que la recocido simulado para este problema. En principio se podría pensar que la gran ventaja es que los algoritmo genéticos usan muchas soluciones a la vez, y lleva parte de razón, pero también se ha visto que no se gana nada aumentando mucho el número de individuos.

En cuanto a las mutaciones, de hecho la recocido simulado aplica un tipo de mutaciones en su solución. Por esta razón seguramente no sea ésta la diferencia fundamental.

Hay que pensar, pues, que puede ser el cruce entre diferentes soluciones sea la clave para obtener los mejores resultados de los algoritmos genéticos, porque es el elemento diferencial más claro. El cruce no funciona bien en todos los problemas, porque a veces lleva a soluciones que no tienen sentido, pero en este caso parece que una solución cruzada de otros dos puede tener sentido y por lo tanto puede ser una herramienta útil para encontrar mejores soluciones.

Recursos

Este PAC requiere de los siguientes recursos:

Básicos: Fichero de datos adjuntos al enunciado

Complementarios: Manual de teoría de la asignatura. En especial los archivos de código del tema 5.

Criterios de valoración

Los ejercicios tendrán la siguiente valoración asociada:

Actividad 1: 3 puntos

Actividad 2: 2.5 puntos

Actividad 3: 2.5 puntos

Actividad 4: 2 puntos

Razonar la respuesta en todos los ejercicios. Las respuestas sin justificación no recibirán puntuación.

Formato y fecha de entrega

La PEC debe entregarse antes del **próximo 6 de Junio** (antes de las 24h).

La solución a entregar consiste en un informe en formato PDF (formato libre) más los archivos de código (*. Py) que usó para resolver la prueba. Estos archivos deben comprimir en un archivo ZIP.

Adjuntar el fichero a un mensaje en el apartado de **Entrega y Registro de AC (RAC)**. El nombre del archivo debe ser ApellidosNombre_IA_PEC2 con la extensión. zip.

Para dudas y aclaraciones sobre el enunciado, diríjase al consultor responsable de su aula.

Nota: **Propiedad intelectual**

A menudo es inevitable, al producir una obra multimedia, hacer uso de recursos creados por terceras personas. Es por tanto comprensible hacerlo en el marco de una práctica de los estudios del Máster de Informática, siempre que esto se documente claramente y no suponga plagio en la práctica.

Por lo tanto, al presentar una práctica que haga uso de recursos ajenos, se presentará junto con ella un documento en el que se detallen todos ellos, especificando el nombre de cada recurso, su autor, el lugar donde se obtuvo y el su estatus legal: si la obra está protegida por copyright o se acoge a alguna otra licencia de uso (Creative Commons, licencia GNU, GPL ...). El estudiante deberá asegurarse de que la licencia que sea no impide específicamente su uso en el marco de la práctica. En caso de no encontrar la información correspondiente deberá asumir que la obra está protegida por copyright.

Deberán, además, adjuntar los archivos originales cuando las obras utilizadas sean digitales, y su código fuente si corresponde.