

La práctica se deberá realizar en equipos de 2 personas.

Dataset

1. Dataset para regresión polinomial: `polynomial.txt`. Tiene un atributo (`age`) y una variable de salida (`length`).

Uso de EMO Project

1. Como requerimientos previos se necesita una computadora con sistema operativo tipo UNIX (Linux, MacOSX) y tener instalado lo siguiente:
 - a. gcc 4.1.2
 - b. MPI 3.0, MPICH 2.2 (optional)
 - c. GNU Make 3.81
2. Descomprimir el archivo `EMO_Project1.36.zip`.
3. Ingresar en el directorio `EMO_Project1.36/` utilizando una consola.
4. Escribir el comando `make clean`.
5. Una vez que haya terminado el comando anterior, ejecutar el comando `make`.
6. Una vez compilada la aplicación entrar en el directorio `EMO_Project1.36/demo/`.
7. Para ejecutar un algoritmo evolutivo multi-objetivo, hacerlo como sigue:
`./emo_moea MOEAD input/Param_03D.cfg DTLZ1 5`
8. Una vez terminada la ejecución, en el directorio `EMO_Project1.36/demo/output/` se encontrarán los archivos de salida. De estos archivos de salida, todos aquellos con extensión `.pos` corresponderán con los atributos mientras que los archivos con extensión `.pof` serán las variables de salidad.
9. Para construir un dataset a partir de los archivos de salida, juntaremos en un solo archivo todos los archivos `.pos`. Asegúrate de quitar de cada archivo la cabecera que lleva puesta (se encuentra en la primera línea de cada archivo y tiene una apariencia: `# N M`). Al juntar todos los archivos `.pos` tendremos el dataset completo de atributos. Ahora hay que unir todos los archivos `.pof` en un único archivo como se hizo anteriormente. Este archivo único contendrá todas las etiquetas de las instancias que se encuentran en el archivo `.pos` unificado.

Procedimiento para regresión lineal multi-salida

1. En EMO_Project ejecutar la siguiente línea:
`./emo_moea MOEAD input/Param_03D.cfg DTLZ1 100`. Esto generará 7 atributos y 3 variables de salida.
2. Generar el dataset como se indica en la sección anterior.
3. Generar el training set y test set de forma aleatoria con una proporción 80%-20%, respectivamente.
4. Investigar el uso de la regresión lineal multi-salida en `Scikit-Learn`. En la presentación de la clase pasada (12/04/21) se dio un fragmento de código que será el punto de partida.
5. Entrenar el modelo.
6. Evaluar el modelo empleando una validación simple.
7. Calcular la precisión del modelo y reportarla.
8. En una gráfica en \mathbb{R}^3 , presentar las variables de salida del test set y los valores predichos por el modelo (usar color rojo para las variables de salida del test set y color azul para los valores predichos). En caso de que no se aprecien correctamente las diferencias entre los puntos, presentar las proyecciones sobre los planos $x - y$ y $y - x$, tomando en cuenta que el espacio \mathbb{R}^3 está dado por los ejes x, y, z .

Procedimiento para regresión polinomial

1. Emplear el dataset `polynomial.txt`.
2. Para este dataset se ocupará el método de evaluación por Bootstrapping.
3. Investigar el uso de la regresión polinomial en `Scikit-Learn`. En la presentación de la clase pasada (12/04/21) se dio un fragmento de código que será el punto de partida.
4. Entrenar el modelo.
5. Evaluar el modelo empleando una validación simple.
6. Calcular la precisión del modelo y reportarla.
7. Gráficar el dataset completo y utilizar el modelo para predecir los valores en todos los puntos discretos que toma el atributo `age`. De esta manera, se podrá comparar el dataset contra la gráfica que predice el modelo de regresión polinomial. Usar color rojo para el dataset y color azul para los valores predichos.

Reporte de la práctica

Emplear el formato de práctica dado por el profesor y seguir las instrucciones mostradas. El archivo que se subirá a *Google Classroom* deberá estar estrictamente en formato PDF y deberá ser nombrado como `report.pdf`.

Usar el lenguaje `Python` para desarrollar la práctica. **Únicamente será aceptado este lenguaje para la generación de los programas.** Además, es forzoso el uso de `Scikit-learn`. Entregar el programa con extensión `.py` debidamente comentado. Entregar un

archivo `README.txt` donde se exponga cómo ejecutar el programa (indicar los parámetros en caso de necesitarlos) y un ejemplo para cómo ejecutar el programa y producir así los resultados reportados.

Entrega global

Tanto el reporte y el programa deberán ser empaquetados en un archivo `.ZIP` y nombrarlo: `practice2.zip`. **Cualquier falta a las instrucciones pedidas implicará la anulación de la práctica para todos los integrantes del equipo.**