

# Projeto de Computação Evolucionária

## Manual de Utilizador

André Clemêncio e Joel Pires

22 de Maio de 2018

Este documento cumpre dois propósitos: listar e descrever os ficheiros e pastas que acompanham o projeto; explicar ao utilizador do projeto como pode executar o programa de forma a obter exatamente os mesmos resultados que nós obtivemos.

## 1 Ficheiros

O projeto é composto por cinco ficheiros de código python, um ficheiro de teste e três pastas que armazenam ficheiros de output.

### 1.1 Ficheiros de código Python

- **mainScript.py:** este é o *script* principal do projeto e é este que o inicia. É aqui que é carregado o ficheiro de testes *zi929.tsp* e que se definem os valores dos parâmetros do algoritmo evolucionário tais como: número de gerações, tamanho da população; probabilidade de mutação; probabilidade de crossover; argumento do método de seleção por torneio; argumento para o método de seleção de elitismo; número de *runs*, método de recombinação, método de mutação, método de seleção dos sobreviventes, função de *fitness*, tamanho do cromossoma e, por fim, método para medição da diversidade.
- **sea\_perm.py:** Este ficheiro contém as funções necessárias para representação e execução do algoritmo evolucionário, bem como escrita dos resultados de *fitness* e diversidade em ficheiros na pasta *testFiles*. Os ficheiros de output são ficheiros de texto do estilo *tsp\_<nome do método>\_<métrica>.txt*. O código é da autoria do Professor Ernesto Costa, apenas sofreu algumas adaptações.
- **stats.py:** É neste ficheiro que se dá a análise estatística dos dados. Fornece uma descrição estatística dos dados (guardado em *describeData.txt* na pasta *statisticFiles*), um diagrama de caixa (guardado em na pasta *images*) e oito histogramas - um histograma da distribuição dos dados e outro com a curva da distribuição normal sobreposta para cada um dos 4 métodos de seleção dos progenitores. Inclui também: teste de Komolgorov-Smirnov, teste de Shapiro-Wilk, teste de Levene, seis testes de Mann-Whitney e a correção de Bonferroni. Todas estas estatística são armazenadas num ficheiro de texto chamado *statisticalReport.txt* dentro da pasta *statisticalFiles*. Cada função possui uma breve descrição daquilo que faz.
- **tsp.py:** Este ficheiro contém as funções adequadas para resolver o problema propriamente dito do Caixeiro-Viajante em específico. O código é inteiramente da autoria do Professor Ernesto Costa.
- **utils.py:** É neste ficheiro que existem funções apropriadas para a visualização dos dados de *fitness* e diversidade dos indivíduos ao longo de várias gerações e ao longo de várias *runs*. O código é da autoria do Professor Ernesto Costa, apenas sofreu algumas adaptações.

### 1.2 Ficheiros de Testes

- **zi929.tsp:** Foi utilizado um ficheiro de teste com informação relativa a distâncias em quilómetros entre 259 cidades do Zimbabwe.

### 1.3 Pastas e Ficheiros de Output

- **images:** É nesta pasta que guardamos imagens relativas ao Diagrama de Caixas e Histogramas da distribuição dos dados

- **statisticFiles:** Aqui encontra-se: o ficheiro *describeData.txt* que apresenta uma descrição estatística básica dos quatro grupos de dados; *statisticalReport.txt* onde jazem os resultados de todos os testes estatísticos feitos durante a execução da função *statisticalReport()* do ficheiro *stats.py* e um excel (*results.ods*) onde se encontram todos os resultados dos testes compilados.
- **testFiles:** É nesta pasta que se encontram os resultados dos testes efetuados ao executar o projeto. Os ficheiros são do tipo: o *tsp\_<nome do método>\_<métrica>.txt*.

## 2 Execução

Para executar o projeto basta correr o ficheiro *mainScript.py*. Este ficheiro executará por defeito 30 runs para cada um dos quatro métodos de seleção dos progenitores. O projeto está munido das pastas e ficheiros suficientes para execução. Contudo, há determinadas *packages* das quais o projeto depende, a saber: *pandas*, *scipy*, *pylab*, *statsmodels*, *numpy* e *matplotlib*.