

# Primeiro Trabalho de Inteligência Artificial

## 2019/2

Este trabalho consiste em realizar uma comparação experimental entre um conjunto pré-definido de metaheurísticas aplicadas ao problema da mochila. As metaheurísticas escolhidas são: Hill Climbing, Beam Search, Simulated Annealing, Genetic Algorithm e GRASP. O procedimento experimental será dividido em duas etapas: a primeira etapa consiste no ajuste de hiperparâmetros das metaheurísticas e a segunda etapa consiste na comparação das metaheurísticas considerando apenas os valores dos hiperparâmetros selecionados na primeira etapa. A metaheurística Hill Climbing, que não possui hiperparâmetros, só participará da segunda etapa da experimentação. Os problemas utilizados para a etapa de treino são necessariamente distintos dos problemas utilizados na etapa de teste.

### 1. Treino

Na primeira etapa da experimentação será necessário realizar uma busca em grade (grid search) considerando ao menos os seguintes valores de hiperparâmetros de cada metaheurística (atenção: pode até usar mais que esses, mas esses devem estar sempre presentes na avaliação):

Beam Search: [10, 25, 50, 100]

Simulated Annealing: {To: [500, 100, 50]; alfa: [0.95, 0.85, 0.7]; numIter (número de iterações): [350, 500]}

GRASP: {numIter (número de iterações): [50, 100, 200, 350, 500], numBest (numero de melhores elementos): [2, 5, 10, 15] }

Genetic Algorithm: {Tamanho da população: [10, 20, 30], taxa de crossover: [0.75, 0.85, 0.95], taxa de mutação: [0.10, 0.20, 0.30]}

O pseudo-código seguinte mostra os passos a serem realizados para o ajuste de hiperparâmetros das metaheurísticas:

#### Algoritmo de Treino

INÍCIO

PARA cada metaheurística que necessita de ajuste de hiperparâmetros FAÇA

PARA cada combinação de valores de hiperparâmetros da busca em grade FAÇA

PARA cada problema do conjunto de treino FAÇA

Executar metaheurística no problema até condição de parada

Armazenar resultado alcançado e tempo de execução

FIM-PARA

FIM-PARA

PARA cada problema do conjunto de treino FAÇA

Normalizar resultados alcançados pela metaheurística

FIM-PARA

PARA cada combinação de valores de hiperparâmetros FAÇA

```

        Obter média dos resultados normalizados
        SE média dos resultados normalizados é a maior ENTÃO
            Armazenar valor de hiperparâmetros para uso no teste
        FIM-SE
    FIM-PARA
    Apresentar valores dos hiperparâmetros selecionados para o teste
    Gerar boxplot dos 10 melhores resultados normalizados alcançados pela metaheurística
    Gerar boxplot dos tempos alcançados pela metaheurística
FIM-PARA
FIM

```

A condição de parada das metaheurísticas deve ser uma combinação entre algum critério definido na própria metaheurística e o tempo máximo de execução de 2 minutos.

## 2. Teste

A segunda etapa da experimentação consiste em avaliar as metaheurísticas no conjunto de problemas de teste com os valores de hiperparâmetros selecionados na etapa de treino.

O pseudo-código seguinte mostra os passos a serem realizados para avaliação e comparação das metaheurísticas:

### Algoritmo de Teste

```

INÍCIO
PARA cada metaheurística FAÇA
    PARA cada problema do conjunto de teste FAÇA
        Executar metaheurística no problema até condição de parada
        Armazenar resultado alcançado e tempo de execução
    FIM-PARA
    Obter média absoluta e desvio padrão das execuções
    Obter média e desvio padrão dos tempos de execução
FIM-PARA
PARA cada problema do conjunto de teste FAÇA
    Normalizar resultados alcançados pelas metaheurísticas
FIM-PARA
Obter média e desvio padrão dos resultados normalizados de cada metaheurística
Gerar tabela contendo média e desvio padrão absolutos e normalizados, e média e desvio
padrão dos tempos de execução de todas as metaheurísticas
PARA cada problema do conjunto de teste FAÇA
    Fazer ranqueamento das metaheurísticas segundo resultado absoluto
FIM-PARA
Obter média dos ranqueamentos das metaheurísticas segundo resultado absoluto
Apresentar as metaheurísticas em ordem crescente de média de ranqueamento

```

Obter média dos resultados normalizados de cada metaheurística  
Apresentar as metaheurísticas em ordem crescente de média dos resultados normalizados  
Gerar boxplot dos resultados normalizados alcançados pelas metaheurísticas  
Gerar boxplot dos tempos alcançados pelas metaheurísticas  
FIM

A condição de parada das metaheurísticas deve ser uma combinação entre algum critério definido na própria metaheurística e o tempo máximo de execução de 5 minutos.

O procedimento de ranqueamento consiste em ordenar as metaheurísticas colocando-as em ordem decrescente de média, isto é, a metaheurística com maior média tem ranqueamento 1, a segunda tem ranqueamento 2, e assim por diante. Em caso de empate entre duas ou mais metaheurísticas, deve-se atribuir um ranqueamento médio igual para todas as metaheurísticas empatadas. Por exemplo, se as metaheurísticas A e B empataram em terceiro lugar, soma-se o ranqueamento 3 e 4, totalizando 7 e divide-se seu valor por 2, totalizando 3,5. Este será o valor de ranqueamento de ambas metaheurísticas A e B.

### **3. Artigo**

Após a realização dos experimentos, um artigo descrevendo todo o processo experimental realizado deverá ser escrito em latex usando o software overleaf. O artigo deve ser estruturado contendo os seguintes componentes:

1. Título
2. Resumo
3. Seção 1. Introdução
4. Seção 2. Descrição do Problema da Mochila
  - a. Descrição geral do problema da mochila
  - b. Descrição das características particulares do problema específico utilizado nos experimentos
5. Seção 3. Descrição dos Métodos Utilizados
  - a. Seção 3.1 Hill Climbing
  - b. Seção 3.2 Beam Search
  - c. Seção 3.3 GRASP
  - d. Seção 3.4 Simulated Annealing
  - e. Seção 3.5 Genetic Algorithm
6. Seção 4. Descrição dos Experimentos Realizados
  - a. Seção 4.1 Treino
    - i. Descrição do Algoritmo de Treino
    - ii. Descrição dos Problemas de Treino
    - iii. Descrição dos valores de hiperparâmetros utilizados na busca em grade de cada metaheurística
    - iv. Apresentação dos bloxplots e dos valores selecionados dos hiperparâmetros de cada metaheurística

- v. Análise dos resultados alcançados
- b. Seção 4.2 Teste
  - i. Descrição do Algoritmo de Teste
  - ii. Descrição dos Problemas de Teste
  - iii. Apresentação da tabela contendo média e desvio padrão absolutos e normalizados, e média e desvio padrão dos tempos de execução de todas as metaheurísticas
  - iv. Apresentação de tabela contendo os ranqueamentos das metaheurísticas segundo resultados absolutos para cada problema de teste e a média dos ranqueamentos
  - v. Apresentação de tabela contendo os ranqueamentos das metaheurísticas segundo resultados normalizados para cada problema de teste e a média dos ranqueamentos
  - vi. Apresentação dos bloxplots dos resultados absolutos obtidos por cada metaheurística
  - vii. Apresentação dos bloxplots dos tempos de execução obtidos por cada metaheurística
  - viii. Análise dos resultados alcançados
- 7. Seção 5. Conclusões
  - a. Análise geral dos resultados
  - b. Contribuições do Trabalho
  - c. Melhorias e trabalhos futuros
- 8. Referências Bibliográficas

#### **4. Informações Complementares**

Os problemas a serem usados para treino e teste estão descritos nos apêndices A e B deste enunciado.

Os gráficos bloxplot requeridos no treino e no teste devem ser gerados usando função específica do pacote seaborn (ver instruções de instalação e uso no apêndice C deste enunciado).

O apêndice D deste enunciado apresenta instruções de instalação e uso do overleaf para a escrita do artigo.

#### **5. Condições de Entrega**

O trabalho deve ser feito individualmente e submetido por e-mail até as 23:59 horas do dia 28/10/2019 para o endereço [fvarejao@gmail.com](mailto:fvarejao@gmail.com) com o subject IA\_TRABALHO\_1\_NomedoAluno\_SobrenomedoAluno. O e-mail deve conter um arquivo .pdf com o artigo produzido no trabalho e .zip contendo todos os arquivos com código fonte em python utilizados no trabalho. Tanto arquivo pdf quanto zip devem possuir o mesmo nome do subject do e-mail enviado. Note que a data limite já leva em conta um dia adicional de tolerância para o caso de problemas de submissão via rede. Isso significa que o aluno deve

submeter seu trabalho até no máximo um dia antes da data limite. Se o aluno resolver submeter o trabalho na data limite, estará fazendo isso assumindo o risco do trabalho ser cadastrado no sistema após o prazo. Em caso de recebimento do trabalho após a data limite, o trabalho não será avaliado e a nota será ZERO. Aluno que receber zero por este motivo e vier pedir para o professor considerar o trabalho estará cometendo um ato de DESRESPEITO ao professor e estará sujeito a perda adicional de pontos na média.

### **Observação importante**

**Caso haja algum erro neste documento, serão publicadas novas versões e divulgadas erratas em sala de aula. É responsabilidade do aluno manter-se informado, freqüentando as aulas ou acompanhando as novidades na página da disciplina na Internet.**

### ***Apêndice A. Problemas de Treino***

Tamanho da Mochila  $t$

Lista de Itens  $vt$ :  $[(v_i, t_i)]$

P1:

$t = 19$

$vt = [(1,3),(4,6),(5,7)]$

P3:

$t = 58$

$vt = [(1,3),(4,6),(5,7),(3,4)]$

P4:

$t = 58$

$vt = [(1,3),(4,6),(5,7),(3,4),(8,10),(4,8),(3,5),(6,9)]$

P6:

$t = 58$

$vt = [(1,3),(4,6),(5,7),(3,4),(8,10),(4,8),(3,5),(6,9),(2,1)]$

P8:

$t = 120$

$vt = [(1,2),(2,3),(4,5),(5,10),(14,15),(15,20),(24,25),(29,30),(50,50)]$

P9:

$t = 120$

$vt = [(1,2),(2,3),(3,5),(7,10),(10,15),(13,20),(24,25),(29,30),(50,50)]$

P11:

$t = 120$

$vt = [(24,25),(29,30),(50,50)]$

P14:

t = 138

vt = [(1,3),(4,6),(5,7),(3,4), (2,6), (2,3), (6,8), (1,2), (2,3), (3,5), 7,10), (10,15), (13,20), (24,25), (29,30), (50,50)]

P17:

t = 13890000

vt = [(1,3),(4,6),(5,7),(3,4), (2,6), (2,3), (6,8), (1,2),(3,5),(7,10),(10,15),(13,20),(24,25),(29,37)]

P20:

t = 45678901

vt = [(1,3),(4,6),(5,7),(3,4),(2,6),(1,2),(3,5),(7,10),(10,15),(13,20),(15,20)]

### ***Apêndice B. Problemas de Teste***

P2:

t = 192

vt = [(1,3),(4,6),(5,7)]

P5:

t = 287

vt = [(1,2),(2,3),(3,4),(4,5),(5,6),(6,7),(7,8),(8,9),(9,10)]

P7:

t = 120

vt = [(1,2),(2,3),(4,5),(5,10),(14,15),(13,20),(24,25),(29,30),(50,50)]

P10:

t = 1240

vt = [(1,2),(2,3),(3,5),(7,10),(10,15),(13,20),(24,25),(29,30),(50,50)]

P12:

t = 104

vt = [(25,26),(29,30),(49,50)]

P13:

t = 138

vt = [(1,3),(4,6),(5,7),(3,4), (2,6), (2,3), (6,8)]

#P15

t = 13890

vt = [(1,3),(4,6),(5,7),(3,4), (2,6), (2,3), (6,8), (1,2), (2,3), (3,5), (7,10), (10,15), (13,20), (24,25), (29,30), (50,50)]

P16:

t = 13890

vt = [(1,3),(4,6),(5,7),(3,4), (2,6), (2,3), (6,8), (1,2),(3,5),(7,10),(10,15),(13,20),(24,25),(29,37)]

P18:

t = 190000

vt = [(1,3),(4,6),(5,7)]

P19:

t = 4567

vt = [(1,3),(4,6),(5,7),(3,4),(2,6),(1,2),(3,5),(7,10),(10,15),(13,20),(15,20)]

### ***Apêndice C. Boxplots usando seaborn***

Instruções de instalação para python3:

```
sudo apt install python3 python3-pip
```

```
pip3 install seaborn
```

```
# arquivo bloxplot.py
```

```
# library & dataset
```

```
import seaborn as sns
```

```
Instruções de instalaçãoimport matplotlib.pyplot as plt
```

```
def example1():
```

```
    mydata=[1,2,3,4,5,6,12]
```

```
    sns.boxplot(y=mydata) # Also accepts numpy arrays
```

```
    plt.show()
```

```
def example2():
```

```
    df = sns.load_dataset('iris')
```

```
    #returns a DataFrame object. This dataset has 150 examples.
```

```
    #print(df)
```

```
    # Make boxplot for each group
```

```
    sns.boxplot( data=df.loc[:, :] )
```

```
    # loc[:, :] means all lines and all columns
```

```
    plt.show()
```

```
example1()
```

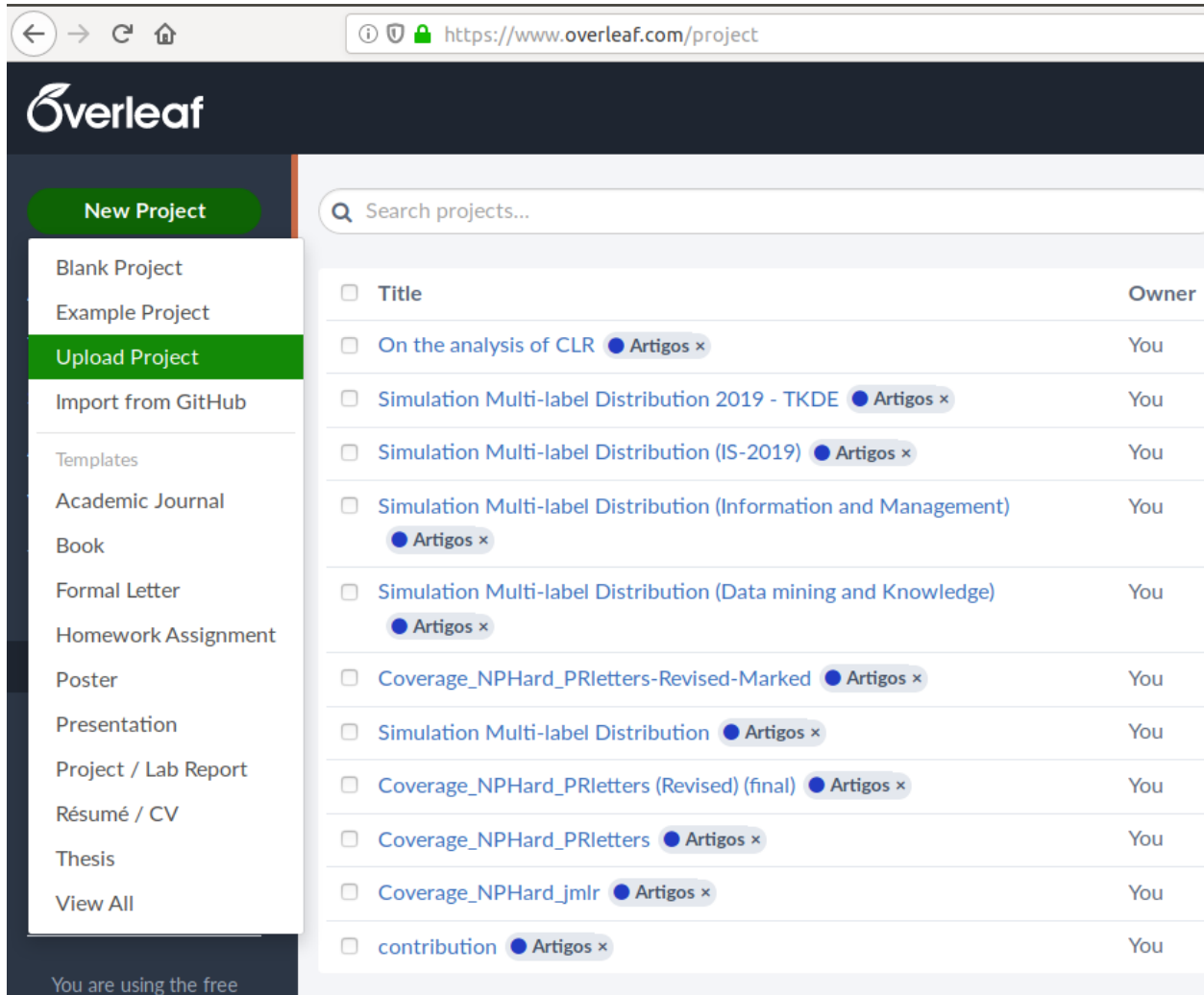
```
example2()
```

Uso em python3:

```
python3 boxplot.py
```

## Apêndice D. Artigo em Latex usando Overleaf

Juntamente com este enunciado foi disponibilizado um arquivo zip com o template de latex para confecção do artigo. O primeiro passo a ser feito é criar uma conta pessoal no Overleaf (<https://www.overleaf.com/register>). Uma vez criada sua conta, deve-se entrar nela. Para incluir o template no overleaf, basta apenas selecionar "New Project>Upload Project" e selecionar o arquivo zip, como mostrado na figura abaixo. Não é necessário descompactar, faça o upload do zip direto. Lembrar de renomear o artigo após o upload do arquivo.



The screenshot shows the Overleaf website interface. At the top, there is a navigation bar with the Overleaf logo and a search bar. Below the navigation bar, there is a sidebar with a 'New Project' button. A dropdown menu is open from this button, showing options: 'Blank Project', 'Example Project', 'Upload Project' (highlighted in green), and 'Import from GitHub'. Below these options, there is a 'Templates' section with various project types: 'Academic Journal', 'Book', 'Formal Letter', 'Homework Assignment', 'Poster', 'Presentation', 'Project / Lab Report', 'Résumé / CV', 'Thesis', and 'View All'. At the bottom of the sidebar, it says 'You are using the free'. The main content area shows a list of projects with columns for 'Title', 'Owner', and 'Actions'. The projects listed are:

Title	Owner
On the analysis of CLR	You
Simulation Multi-label Distribution 2019 - TKDE	You
Simulation Multi-label Distribution (IS-2019)	You
Simulation Multi-label Distribution (Information and Management)	You
Simulation Multi-label Distribution (Data mining and Knowledge)	You
Coverage_NPHard_PRIletters-Revised-Marked	You
Simulation Multi-label Distribution	You
Coverage_NPHard_PRIletters (Revised) (final)	You
Coverage_NPHard_PRIletters	You
Coverage_NPHard_jmlr	You
contribution	You