

Reprodução da Análise Quantitativa de um Experimento

Diogo C. T. Batista¹, Elissandra G. Pereira¹

¹Universidade Federal do Paraná (UFPR)
Curitiba – Paraná – Brasil

diogocezar@ufpr.br, egpereira@inf.ufpr.br

Resumo. [completar]

1. Identificação do Artigo

[completar]

Artigo de [Jin 2013]

2. Reconhecimento dos Testes Estatísticos do Experimento

[completar]

3. Reprodução dos Testes

Para simular os resultados qualitativos do experimento analisado foi empregada a linguagem *Python*.

Como o artigo não fornece todas respostas do questionário, gerou-se valores aleatórios com base na média, desvio padrão e número de respostas obtidas, tanto para os testes com *Structure design guidelines*, *Selective-attention design guidelines* e o grupo de controle. Com isso, calculou-se o *valor-p* e o comparamos com os resultado apresentados pelo do autor do artigo. Utilizou-se a estratégia de executar o algoritmo 100 vezes e retirou-se a média de todos os *valor-p* encontrados, para assim podermos termos uma análise melhor para com o resultado descrito no artigo.

4. Algoritmo Criado

Para a elaboração do algoritmo utilizou-se a linguagem *Python*. Em conjunto com as bibliotecas *numpy* e *scipy*

O Código 1 mostra a definição de uma função genérica que obtém informações de um grupo qualquer e um grupo controle. São informadas as variáveis:

- `agent_average`: média do grupo agente;
- `agent_sigma`: desvio padrão do grupo agente;
- `agent_num`: quantidade de amostras do grupo agente;
- `control_average`: média do grupo de controle;
- `control_sigma`: desvio padrão do grupo de controle;
- `control_num`: quantidade de amostras do grupo de controle;
- `start`: menor resposta;
- `end`: maior resposta;
- `iterations`: quantas iterações serão realizadas para gerar a média;

```

1 import numpy as np
2 import scipy.stats as stats
3
4 def generate(agent_average, agent_sigma, agent_num, control_average, control_sigma,
              control_num, start, end, iterations):
5     values = []
6     for x in range(iterations):
7         agent_dist = stats.truncnorm((start - agent_average) / agent_sigma, (end -
              agent_average) / agent_sigma, loc=agent_average, scale=agent_sigma)
8         agent_values = agent_dist.rvs(agent_num)
9         control_dist = stats.truncnorm((start - control_average) / control_sigma, (end -
              control_average) / control_sigma, loc=control_average, scale=control_sigma)
10        control_values = control_dist.rvs(control_num)
11        u_statistic, p_val = stats.mannwhitneyu(agent_values, control_values)
12        values.append(p_val)
13    return round(np.mean(values), 20)

```

Código 1. Definição da Função Genérica

Nessa função, obtém-se cada um dos parâmetros, e se gera em um range de *iterations* valores aleatórios que representem a média e o desvio padrão, considerando o grupo de controle nos cálculos.

Na sequência, o Código 2 mostra as definições da variáveis de acordo com o trabalho de [Jin 2013].

```

1 if __name__ == "__main__":
2     structure = {
3         "control": {
4             "average": 2.30,
5             "sigma": 1.69,
6             "n": 35,
7         },
8         "structure_design": {
9             "average": 5.07,
10            "sigma": 2.38,
11            "n": 35,
12            "start": 0,
13            "end": 12
14        },
15        "selective_attention": {
16            "average": 3.31,
17            "sigma": 2.25,
18            "n": 35,
19            "start": 0,
20            "end": 12
21        },
22    }
23    comprehension = {
24        "control": {
25            "average": 10.46,
26            "sigma": 2.90,
27            "n": 35,
28        },
29        "structure_design": {
30            "average": 11.94,
31            "sigma": 3.00,
32            "n": 35,
33            "start": 0,
34            "end": 20
35        },
36        "selective_attention": {
37            "average": 11.28,
38            "sigma": 3.31,
39            "n": 35,
40            "start": 0,
41            "end": 20

```

```

42     },
43 }
44 result = {
45     "structure": {
46         "structure_design": generate(
47             structure["structure_design"]["average"],
48             structure["structure_design"]["sigma"],
49             structure["structure_design"]["n"],
50             structure["control"]["average"],
51             structure["control"]["sigma"],
52             structure["control"]["n"],
53             structure["structure_design"]["start"],
54             structure["structure_design"]["end"],
55             100
56         ),
57         "selective_attention": generate(
58             structure["selective_attention"]["average"],
59             structure["selective_attention"]["sigma"],
60             structure["selective_attention"]["n"],
61             structure["control"]["average"],
62             structure["control"]["sigma"],
63             structure["control"]["n"],
64             structure["selective_attention"]["start"],
65             structure["selective_attention"]["end"],
66             100
67         )
68     },
69     "comprehension": {
70         "structure_design": generate(
71             comprehension["structure_design"]["average"],
72             comprehension["structure_design"]["sigma"],
73             comprehension["structure_design"]["n"],
74             comprehension["control"]["average"],
75             comprehension["control"]["sigma"],
76             comprehension["control"]["n"],
77             comprehension["structure_design"]["start"],
78             comprehension["structure_design"]["end"],
79             100
80         ),
81         "selective_attention": generate(
82             comprehension["selective_attention"]["average"],
83             comprehension["selective_attention"]["sigma"],
84             comprehension["selective_attention"]["n"],
85             comprehension["control"]["average"],
86             comprehension["control"]["sigma"],
87             comprehension["control"]["n"],
88             comprehension["selective_attention"]["start"],
89             comprehension["selective_attention"]["end"],
90             100
91         )
92     }
93 }
94 print(result)

```

Código 2. Definições das Variáveis

5. Resultados

Após a execução do algoritmo, obteve-se resultados disponíveis no Código 3.

```

1  {
2  'structure': {
3      'structure_design': 0.0001331037517740879,
4      'selective_attention': 0.050186023592034855
5  },
6  'comprehension': {
7      'structure_design': 0.07122011277566724,
8      'selective_attention': 0.17129878841709156
9  }

```

Código 3. Resultados Obtidos

Pode-se notar que os resultados obtidos possuem equivalência com os apresentados em [Jin 2013].

Experimento	Tipo	pvalue (autor)	pvalue (gerado)
Structure Design	Structure	.000	.000
Selective Attention	Structure	.012	.050
Structure Design	Comprehension	.000	.071
Selective Attention	Comprehension	.010	0.17

Tabela 1. Estratégias de representação

6. Conclusão

Apesar das limitações criadas pela ausência dos dados brutos, o resultado da reprodução conseguiu replicar de forma satisfatória os mesmos resultados obtidos pelo trabalho de [Jin 2013], mesmo quando houve uma certa diferença entre os valores obtidos.

O exercício permitiu criarmos de forma satisfatória os valores quantitativos de resultado de um experimento controlado.

Referências

Jin, S.-H. (2013). Visual design guidelines for improving learning from dynamic and interactive digital text. *Computers & Education*, 63:248 – 258.