FACULDADE DE CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE D COIMBRA

Metodologias Experimentais em Informática

Análise Exploratória de Dados e Regressão Linear

https://pixabay.com/illustrations/charts-tables-graph-statistics-6246450/

Exam Scheduling Problem

2021/2022

Mestrado em Engenharia Informática

PL4	Gabriel Fernandes	2018288117	gabrielf@student.dei.uc.pt
PL4	Miguel Rabuge	2018293728	rabuge@student.dei.uc.pt
PL4	Pedro Rodrigues	2018283166	pedror@student.dei.uc.pt

---

1. Introdução

Este projeto tem como objetivo a construção de uma experiência científica. Como tal, procuramos definir e planear experiências com vista a obter e analisar dados para retirar conclusões dos mesmos, utilizando métricas adequadas, com vista a formular e testar hipóteses estatísticas.

O âmbito do projeto consiste na análise de dois algoritmos, que implementam diferentes estratégias para escalonamento de exames, e cujo objetivo é determinar, dado **N** exames e a probabilidade **P** de existirem sobreposições dos mesmos, qual o número mínimo de *time slots* **M** para que não existam sobreposições, bem como o tempo **T** que os mesmos demoraram.

2. Definição do Problema e Identificação de Variáveis

O problema, apesar das várias questões possíveis, pode simplesmente ser definido pelas questões:

- *Existem relações entre as variáveis de entrada e o resultado dos algoritmos?*
- *De que forma o resultado dos algoritmos diferem para a mesma entrada?*

Relativamente a este problema identificamos as seguintes variáveis:

Variavel	Descrição	Tipo
N	Número de Exames	Independente
P	Probabilidade de ocorrência de sobreposições	Independente
M	Número mínimo de time slots necessários para que não existam sobreposições	Dependente
T	Tempo de execução do algoritmo (CPU time)	Dependente

3. Geração de Dados e Cenário Experimental

Com vista a gerar e organizar dados, foi criado um *Makefile* para esse efeito. Desta forma, correndo *make instances* é chamado o shell script *generate_instances.sh* que irá criar e popular um diretório "instances" com todas as combinações de **N**, **P** e da seed (**Si**) utilizada para o gerador aleatório da biblioteca random.py. Cada ficheiro (instância) tem como nome esp_**N**_**P**_**Si**.dat.
As instâncias acima geradas são posteriormente encaminhadas para os dois algoritmos (**Algorithm**) em estudo, *code1.c* e *code2.c*, em conjunto com a seed (**Ss**) para o gerador aleatório do algoritmo e o tempo máximo de execução deste (**Tmax**), através do Makefile, com *make runs* que chama o shell script run.sh, originando um ficheiro csv com as colunas: **N**, **P**, **Si**, **Ss**, **Algorithm**, **M**, **T**, **Tmax**.

Deste modo, obtemos um ficheiro de dados estruturado e preparado para a análise exploratória. Todos estes utilitários, dados e código utilizado para análise exploratória encontram-se em anexo a este documento.

4. Análise de Dados

Em primeiro lugar, com o objetivo de fazermos uma *overview* geral dos dados e de identificar possíveis correlações visualmente, começamos por representar os dados em dois scatter plots 3D, para code1.c e code2.c, com **T** em função de **N** e **P.**

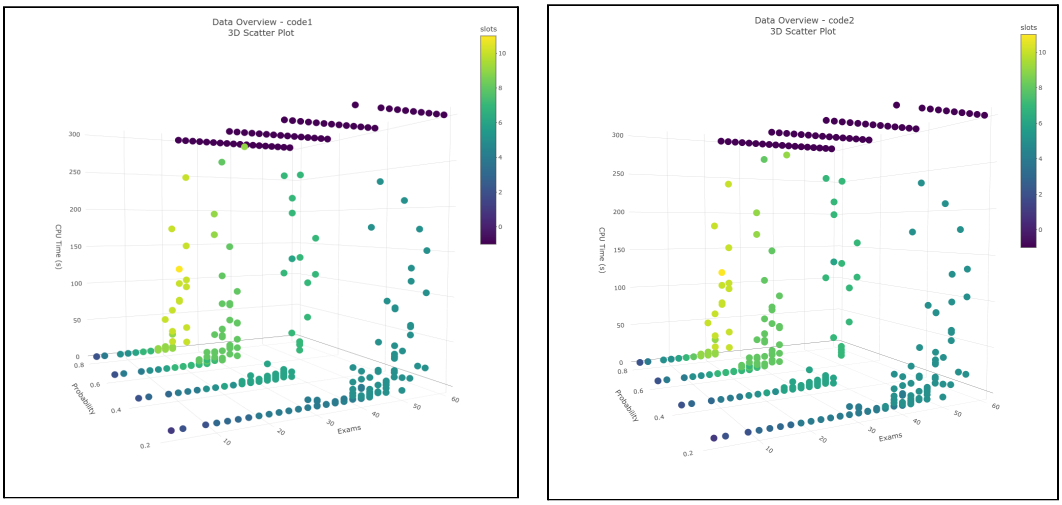


Figura 1 e 2: Scatter plots 3D dos dados obtidos para ambos os algoritmos (nos eixos encontram-se as variáveis: CPU Time (**T**), Probability **(P**), Exams (**N**); a cores o nº de slots (**M**) - (data2.csv)

A partir dos scatter plots, conseguimos ter a perceção de alguns padrões, nomeadamente:

- Os tempos (**T**) dos dois algoritmos são similares em relação a **N** e **P**
- Os tempos (**T**) começam a crescer mais cedo quando **P** aumenta
- Os tempos (**T**) aparentam seguir uma distribuição exponencial em relação a **N**.

- À medida que a probabilidade de sobreposições (**M**) aumenta, os tempos (T) tem tendência a tornar-se mais elevados mais cedo (para um menor número de exames **N**) e o número de slots necessários tende a aumentar.

Com o objetivo de detetar se os resultados obtidos por ambos os algoritmos seguem a mesma distribuição, utilizamos two sample (code1 vs code2) Q-Q plots com vista a averiguar se ambos os algoritmos seguem as mesmas distribuições:

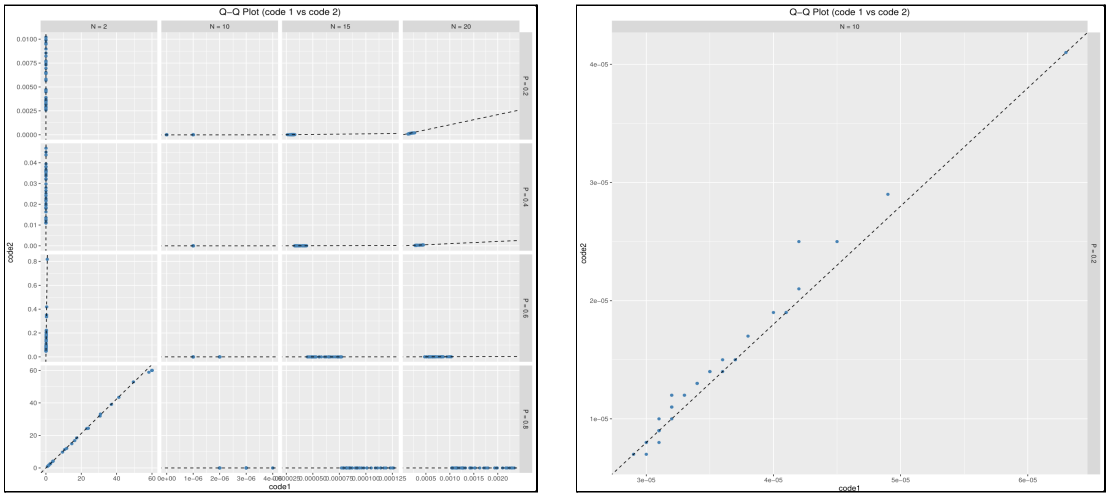
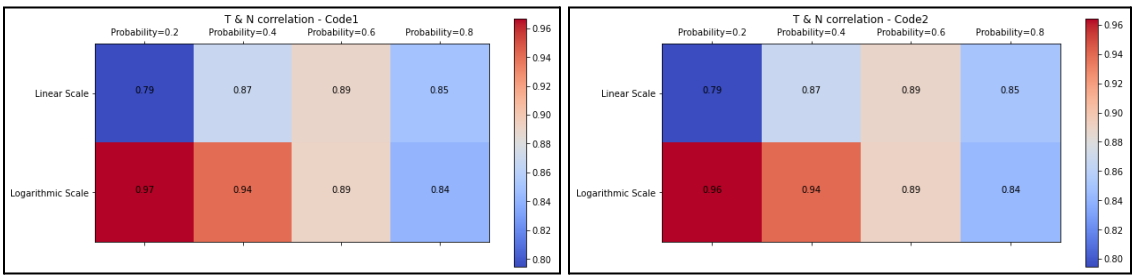


Figura 3 e 4: À direita, um exemplo de uma matriz de qqplots (code1 vs code2) para várias combinações (**N**) e (**P**). À esquerda uma das células da matriz (N = 10, P = 0.2), para mais fácil visualização. - (data1.csv)

Assim, pela observação dos resultados obtidos, conseguimos verificar que de uma forma geral ambos os algoritmos aparentam seguir a mesma distribuição, que assumimos ser exponencial. Importa salientar que embora os resultados sejam indicadores, numa fase seguinte verificaremos a veracidade desta hipótese.

De um ponto de vista mais analítico, com o objetivo de compreender as correlações lineares entre **T** e **N,** bem como de **T** e **P**, apresentamos duas matrizes de correlações lineares, utilizando a correlação de Pearson tanto numa escala linear como numa escala logarítmica, fixando **P** e **N**, respetivamente. Com o objetivo de compreender as correlações em termos de rank (posições relativas das observações entre as duas variáveis), também apresentamos com os mesmos parâmetros as mesmas matrizes, utilizando a correlação de Spearman.



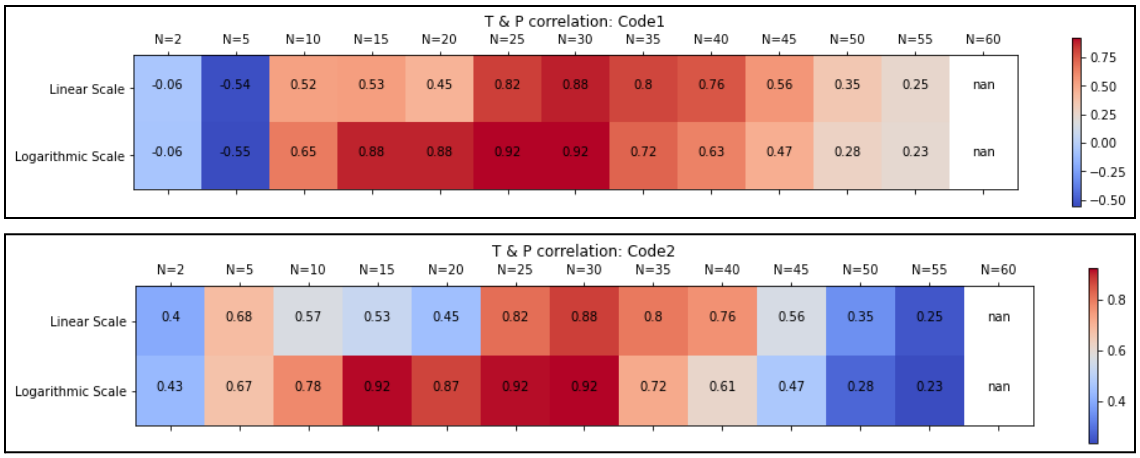


Figura 5 a 8: Correlações de Pearson em escalas linear e logarítmica, fixando número de exames (**N**) e probabilidade de sobreposição de exames (**P**) - (data1.csv)

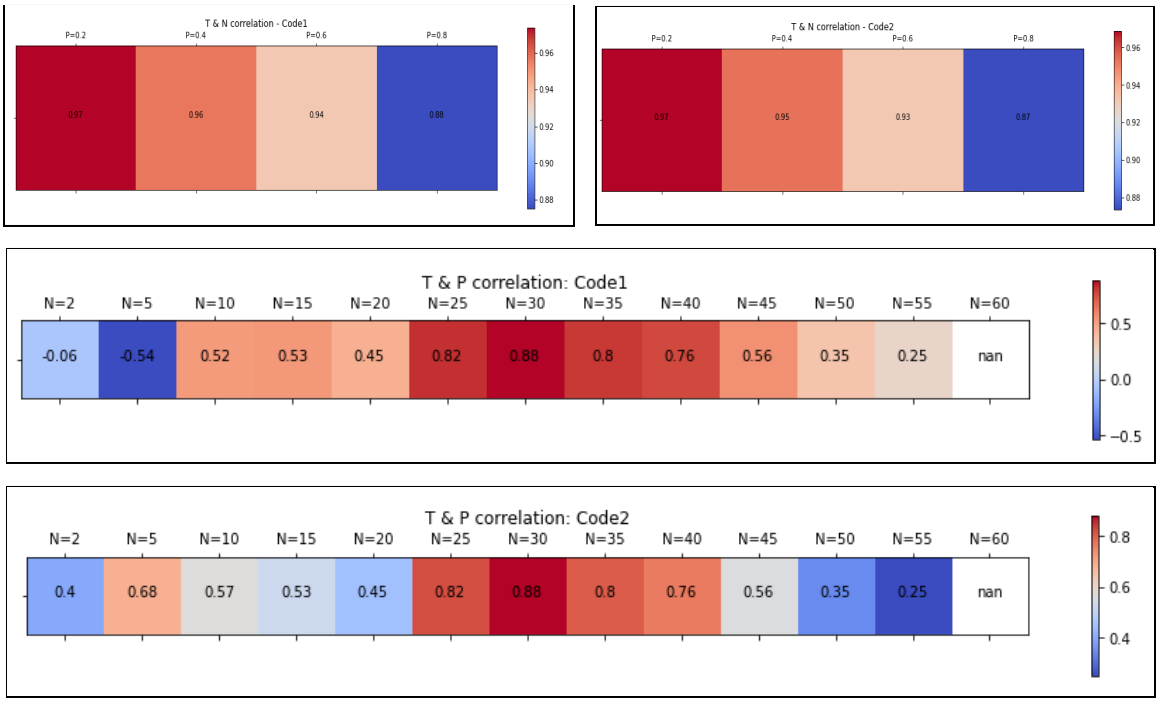


Figura 9 a 12: Correlações de Spearman, fixando número de exames (**N**) e probabilidade de sobreposição dos exames (**P**). - (data1.csv)

Deste modo, verifica-se que a correlação de Pearson, quando aplicada aos dados transformados por um logaritmo, tende a aumentar em módulo, o que confirma a nossa intuição de que os tempos (**T**) seguem uma distribuição exponencial em função de **N**. Os valores obtidos para as correlação de spearman também suportam a esta hipótese, visto que são elevados, sendo isto indicativo de que os dados obtidos por ambos os algoritmos apresentam o mesmo ordenamento relativo. O *nan* presente na figura, deve-se a todos os pontos serem do tipo (**N:** 60, **T:** 60), levando a que não seja possível obter uma correlação.

Tendo em conta que as correlações de Pearson se apresentaram bastante relevantes, especialmente numa escala logarítmica, faz sentido apresentar e estudar regressões lineares:

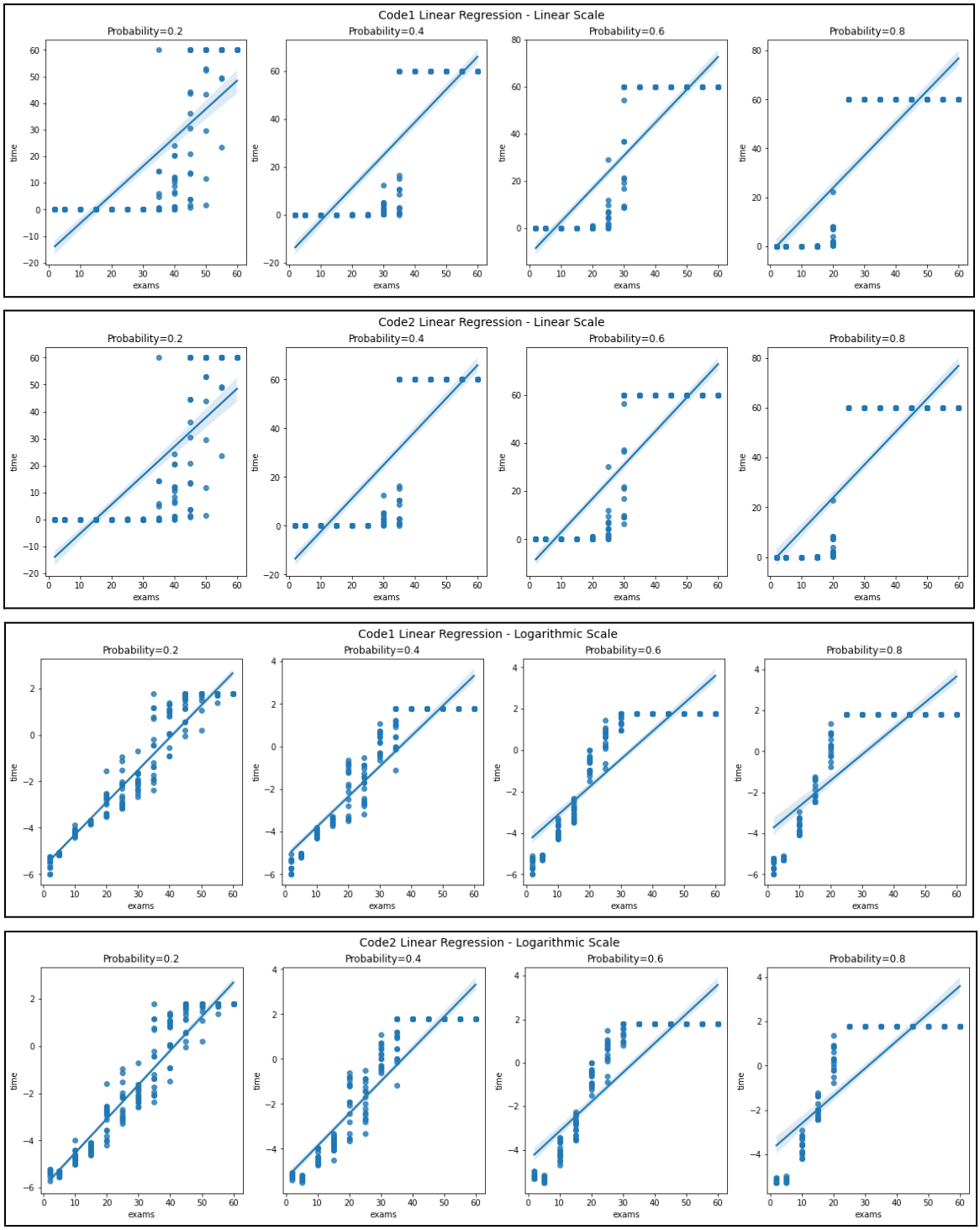
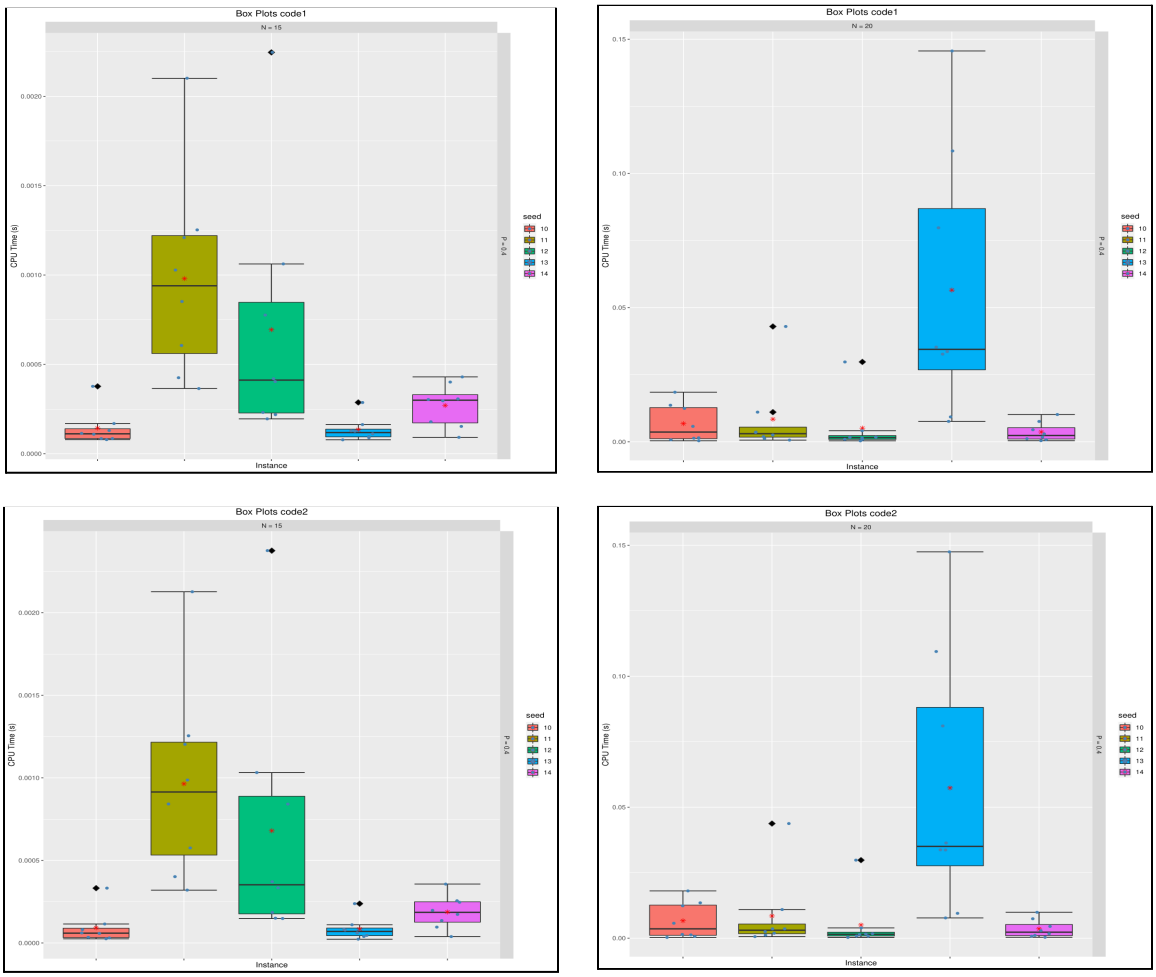


Figura 13 a 16: Regressões lineares em escalas linear e logarítmica, fixando número de exames (**N**) e probabilidade de sobreposição de exames (**P**). - (data1.csv)

Depois de visualizadas as regressões, podemos verificar que as mesmas se adequam melhor aos dados quando a escala é logarítmica. Importa referir que os tempos obtidos foram majorados por *max_time* = 60s, o que causa a distorção dos dados observados e, consequentemente, um viés na reta de regressão linear, bem como na respetiva correlação de Pearson.

Seguidamente, para observar como os algoritmos se comportam, mostramos os diferentes tempos de execução (**T**) obtidos por estes, com diferentes seeds de execução (**Ss**), para várias instâncias, através de 2 box plots, para code1.c e code2.c, respetivamente.



Figuras 17 a 20: Exemplos de box plots com distribuição dos tempos obtidos, com múltiplas execuções com diferentes sedes, para cada instância. Na primeira linha temo resultados para code1 (N = 15, 20 e P = 0.4). Na segunda linha temos os resultados para code2 (N = 15, 20 e P = 0.4). - (data1.csv)

Os box plots apresentados foram realizados numa tentativa de perceber como a dispersão dos tempos (**T**) é influenciada pelas seeds (**Si** e **Ss**) das instâncias e dos algoritmos utilizados. Foram elaborados mais plots, mas por questões de visualização/espaço apenas apresentamos estes. Embora a quantidade de dados não seja significativa, conseguimos perceber que de uma forma geral, as variações

temporais resultantes são sugestivas de que as seeds (**Si** e **Ss**) não são um fator diferenciador na qualidade dos resultados obtidos, embora seja algo que iremos validar recorrendo a testes de hipóteses em metas seguintes.

5. Conclusões

De uma forma geral fomos obtendo vários dados resultantes da análise exploratória que nos permitiram responder às questões que nos propusemos a responder, nomeadamente:

- *Existem relações entre as variáveis de entrada e o resultado dos algoritmos?*
  - Observamos que mediante diversas combinações dos parâmetros de entrada conseguimos obter resultados bastante diferentes. Em primeiro lugar, o aumento do número de exames (**N**) reflete-se exponencialmente no tempo (**T**) necessário para resolver o problema. Em segundo lugar, a probabilidade (**P**) também tem impacto, na medida em que afeta o número de sobreposições de exames resultando no algoritmo necessitar de um maior tempo, mesmo para menos exames. Por fim, as seeds (**Si** e **Ss**) utilizadas parecem não ter grande influência nos resultados obtidos, embora seja algo que necessite de mais estudo.
- *De que forma o resultado dos algoritmos diferem para a mesma entrada?*
  - Ambos apresentam comportamentos similares, visto que, pela nossa análise, estes seguem a mesma distribuição.

Histórico de Versões:

- **V0.1.0 -** Primeira versão deste documento, entregue na plataforma inforestud@nte a 08-11-2021 23:11
- **V0.2.0 -** Versão do documento com as modificações propostas pelo professor Alexandre Jesus após sua apreciação do documento apresentado.
  - Melhoria na apresentação e identificação das variáveis
  - Modificação dos scatter plots 3D apresentados por forma a incluir mais dados, nomeadamente para um número de exames superior a 40.
  - Modificação dos two sample Q-Q plots apresentados por forma a corrigir um erro associado à sua construção e ao posicionamento da qqline.
  - Adição de matrizes com correlação de spearman por forma a complementar as correlações já apresentadas (sugestão do professor)
  - Adição de legendas aos gráficos e alterações mínimas ao texto que acompanha os mesmos. (identificação dos ficheiros de dados utilizados).