



**Equipa\_8\_Ilha\_DKL**

1190986 \_ Raúl Coelho

1190434 \_ Bernardo Silva

1190780 \_ José Forno

1190447 \_ Bruno Silva

1190390 \_ André Pinto

Relatório do Projeto de LAPR1

Análise de Séries Temporais

Dezembro de 2019

**Docente(s)/Orientador(es)**

Ana Barata (ABT)

Carlos Ferreira (CGF)

Sandra Luna (SLU)

**Cliente**

Carlos Ferreira (CFG)

**Unidade Curricular**

Laboratório/Projeto I

**ÍNDICE**

[1 Introdução 1](#_Toc29220105)

[2 Metodologia do trabalho 2](#_Toc29220106)

[**2.1 EduScrum no desenvolvimento do Projeto** 2](#_Toc29220107)

[**2.2 Planeamento e distribuição de tarefas** 2](#_Toc29220108)

[**2.3 Reflexão crítica sobre a dinâmica do grupo** 2](#_Toc29220109)

[3 Análise de séries temporais 3](#_Toc29220110)

[**3.1 Sobre a análise de séries temporais** 3](#_Toc29220111)

[**3.2 Análise utilizando diferentes resoluções** 3](#_Toc29220112)

[**3.3 Filtragem/Suavização** 3](#_Toc29220113)

[**3.4 Previsão** 4](#_Toc29220114)

[4 Desenvolvimento e Implementação da Aplicação 5](#_Toc29220115)

[**4.1 Analise dos métodos do projeto** 5](#_Toc29220116)

[**4.2 Tempo de ordenação** 14](#_Toc29220117)

[**4.3 Testes unitários** 14](#_Toc29220118)

[5 Resultados 15](#_Toc29220119)

[**5.1 Análise dos resultados** 17](#_Toc29220120)

[6 Conclusão 19](#_Toc29220121)

[7 Referências 19](#_Toc29220122)

[ANEXOS i](#_Toc29220123)

[**ANEXO A\_Testes Unitários** ii](#_Toc29220124)

# **1 Introdução**

O presente relatório tem como objetivo apresentar uma aplicação que foi desenvolvida no âmbito da cadeira de LAPR1, bem como descrever o seu processo de desenvolvimento. Esta aplicação tem como objetivo analisar uma série temporal em diferentes resoluções, modelar uma série e prever valores futuros.

Para a realização deste projeto tivemos de adquirir conhecimentos básicos sobre a análise de séries temporais, como por exemplo técnicas de filtragem, suavização e previsão de séries, conhecimentos sobre a utilização do Gnuplot e do Java-Plot e ainda conhecimentos de programação na linguagem Java.

Neste relatório está presente toda a informação necessária para a compreensão da realização do nosso projeto desde a metodologia utilizada à análise de resultados. Esta informação está organizada em diversas etapas: um desenvolvimento onde apresentamos as técnicas e métodos estudados, a metodologia de trabalho, a implementação da aplicação e a análise de resultados; e uma conclusão onde é feita uma reflexão sobre o decorrer deste trabalho, bem como uma apresentação de aspetos a corrigir. Este relatório inclui também as referências aos artigos e sites visitados.

# **2 Metodologia do trabalho**

Esta secção inclui o método que utilizámos na delegação de tarefas e na relação e dinâmica entre o grupo.

## **2.1 EduScrum no desenvolvimento do Projeto**

EduScrum é uma metodologia de trabalho muito utilizada em trabalhos escolares e as suas metodologias práticas são o Sprint Planning, onde se faz o planeamento das atividades a ser desenvolvidas durante o período de 2 a 4 semanas, em que cada ciclo é um Sprint, o Sprint Review, que é a apresentação do que foi feito durante determinado Sprint, com discussão sobre acertos, erros e possíveis melhorias ao processo, com foco no produto produzido e o Product Backlog, onde listamos todas as tarefas do projeto.

No nosso grupo adotamos esta estratégia no desenvolvimento do projeto decidindo inicialmente quem seria o Scrum Master e de seguida começamos a distribuição de tarefas.

## **2.2 Planeamento e distribuição de tarefas**

Para começar criámos uma pasta no OneDrive para que pudéssemos colocar lá todos os ficheiros que achássemos úteis para o desenvolvimento do projeto. Criámos ainda uma board no trello que serviu para colocarmos o Product Backlog para sabermos como nos orientar e ainda mais tarde para identificarmos que tarefa estava atribuída a cada pessoa. Utilizamos o trello ainda para fazer o Log Book, registando as tarefas que fizemos nas aulas. Utilizamos também o SourceTree para fazer a partilha rápida do ficheiro do trabalho. Após a configuração destas três ferramentas começámos a distribuir tarefas, mas optámos por fazer cada tarefa num grupo de dois uma vez que achámos que o tempo disponibilizado seria suficiente para analisarmos bem o trabalho e procurarmos informação. Para quando desenvolvêssemos trabalho fora de aulas utilizámos ainda o Messenger para comunicarmos seja para tirar dúvidas ou para avisar os colegas sobre o que estivemos a alterar.

## **2.3 Reflexão crítica sobre a dinâmica do grupo**

A equipa demonstrou ser competente, organizada e cumpridora. Também apresentou um espírito de perseverança nos momentos em que havia qualquer tipo de erro ou falha no código do programa. Apesar de tudo houve alguns problemas visto que um dos representantes do grupo poucas vezes compareceu e não apresentou espírito de equipa. Na semana final do projeto um outro colega teve de ficar ausente devido a um problema de saúde atrasando ligeiramente o projeto.

# **3 Análise de séries temporais**

Esta secção tem de inclui o estudo sobre séries temporais com exemplos ilustrativos.

## **3.1 Sobre a análise de séries temporais**

Uma série temporal é uma sequência de observações ordenada cronologicamente que, em geral, são recolhidos em intervalos regulares. A análise de séries temporais pode ser aplicada a qualquer variável que muda ao longo do tempo e, de um modo geral, as observações mais próximas tem valores mais próximos que aqueles valores mais distantes.

A análise de séries temporais é de grande utilidade em vários domínios sendo que do seu processamento podem resultar ganhos significativos.

Para extrair conhecimento das séries temporais existe um conjunto alargado de métodos, técnicas e ferramentas que podem ser utilizados. Alguns destes métodos exigem conhecimentos avançados que são apenas adquiridos através de formação avançada. Outros métodos, como os que utilizamos neste projeto, são mais simples. Entre estes estão métodos e técnicas básicos que permitem analisar uma série temporal em diferentes resoluções, filtrar ou suavizar uma série e prever valores futuros utilizando modelos.

## **3.2 Análise utilizando diferentes resoluções**

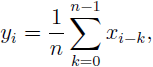
As observações que constituem uma série temporal podem ser recolhidas com elevada frequência, para permitir monitorizar e analisar com rigor ocorrências, ou com menor frequência, quando o estado de um sistema ou processo praticamente não é alterado ao longo do tempo e uma amostra recolhida com grande espaçamento é suficiente para analisar a evolução de um sistema ou processo.

Em geral, e atendendo à tecnologia disponível, recolhem-se dados com elevada frequência que depois são transformados para a resolução necessária ao objetivo e problema que se pretende resolver. Por exemplo, se pretendemos analisar o consumo de eletricidade mensal de uma casa e estamos a recolher dados que representam o consumo a cada hora, então teremos que somar o consumo de todos os dias e horas de um determinado mês para calcular o consumo total nesse mês.

## **3.3 Filtragem/Suavização**

Uma técnica relevante para analisar uma série temporal é a filtragem. Ao filtrar os dados podemos remover ruído e identificar tendências. Entre as técnicas de filtragem mais utilizadas e simples estão a Média Móvel Simples e a Média Móvel Exponencialmente Pesada (que são as técnicas utilizadas neste projeto).

A Média Móvel Simples é definida pela equação:



onde xi são os termos que representam a série original, yi é a série resultante da aplicação do filtro (da suavização) e n é a ordem da média móvel.

A Média Móvel Exponencialmente Pesada é definida pela equação:



onde xi são os termos que representam a série original yi é a série resultante da aplicação do filtro (da filtragem) e α é uma constante que toma valores no intervalo ]0,1]. Nas Figuras 1 e 2 são apresentadas uma série original e uma série filtrada que resulta da aplicação do modelo Média Móvel Exponencialmente Pesada, para dois valores distintos de α (0.05 e 0.5).

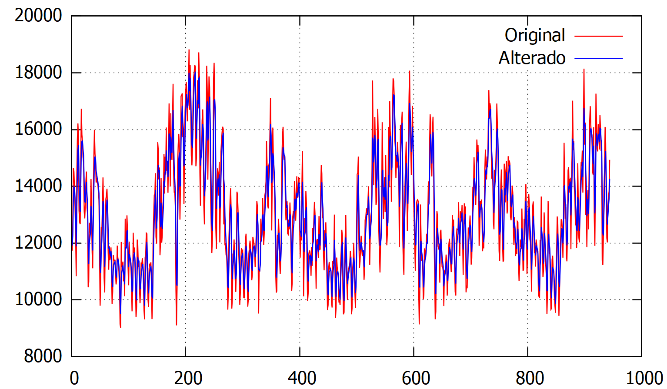
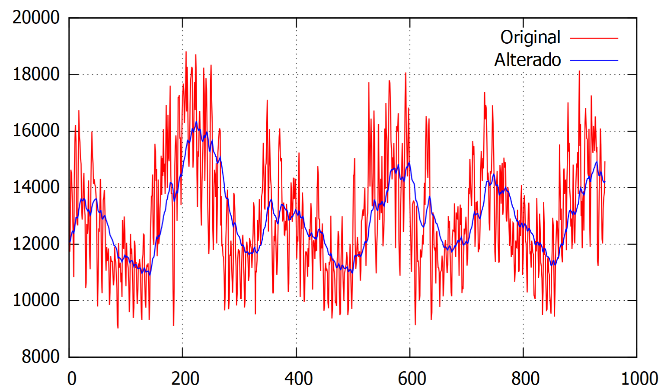


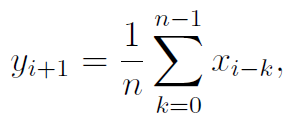
Figura 1: α de 0.05

Figura 2: α de 0.5

## **3.4 Previsão**

Para analisar uma série temporal também é possível encontrar um modelo matemático que capture o processo que gerou a série temporal e permita prever valores futuros da série utilizando o histórico de dados. Entre os modelos mais utilizados e simples estão os modelos de Média Móvel Simples e os modelos de Média Móvel Exponencialmente Pesada (que são as técnicas utilizadas neste projeto).

Para realizar uma previsão utilizando a Média Móvel Simples recorremos à equação:



onde xi são os termos que representam a série original, yi+1 representa uma previsão utilizando dados históricos e n é a ordem da média móvel.

No caso da Média Móvel Exponencialmente Pesada, para realizar uma previsão recorremos à equação:



onde xi são os termos que representam a série original, yi+1 representa uma previsão utilizando dados históricos e α é uma constante que toma valores no intervalo ]0; 1].

# **4** **Desenvolvimento e Implementação da Aplicação**

A nossa aplicação está dividida em módulos cada um feito de modo a atingir os objetivos especificados em cada alínea do trabalho.

É possivel correr o projeto de dois modos ,ambas através da linha de comandos do sistema operativo em utilização como por exemplo o Linux ,Mac OS ou Windows.

## **4.1 Analise dos métodos do projeto**

Agora procederemos á análise de cada um dos métodos presentes no projeto.

Numa primeira fase começamos por criar diversas estruturas de modo a guardar toda a informação presente no ficheiro CSV e que será fulcral no desenvolver de todo este projeto . Criámos uma variável global “MAX” com o valor de 30000, pois esta seria necessária ao longo de todo o programa deste modo tornando mais fácil a sua implementação.

**4.1.1 Modo não interativo**

Primeiramente temos a implementação do código do modo não interativo,em que será realizado o código todo automaticamente guardando a informação nos ficheiros .

O modo não interativo permite que o utilizador insira como parâmetros as informações necessárias para o funcionamento do programa do seguinte modo:

java -jar nome programa.jar - nome ts nome da serie temporal.csv -resolucao X -modelo M -tipoOrdenacao T -parModelo nAlpha -momentoPrevisao D, em que X, M, T, nAlpha e D são parâmetros que podem tomar os seguintes valores: X pode tomar os valores 11, 12, 13, 14, 2, 3 e 4 que representam respetivamente os periodos manhã, tarde, noite, madrugada, diário, mensal e anual; M pode tomar os valores 1 e 2 que representam respetivamente os modelos Média Móvel Simples e Média Móvel Exponencialmente Pesada; T pode tomar valores 1 ou 2, sendo que 1 representa ordenação crescente e 2 ordenação decrescente; nAlpha toma um valor numérico que corresponde ao parâmetro n ou parâmetro alpha, conforme o valor do parâmetro M; D representa um momento de previsão e o seu valor depende da resolução especificada.Neste modo,ao contrário do anterior , as informações de output não são mostradas no ecrã ,são guardadas em ficheiros . No caso dos gráficos em ficheiros PNG , as séries temporais analisadas nos gráficos em ficheiros CSV e o resto da informação em ficheiro de texto.

**4.1.1.1 Método ler\_ficheiro**

Numa primeira fase é feita a leitura do ficheiro o qual foi inserido como segundo parâmetro na linha de comandos e toda a informação neste contida é guardada num array bidimensional (ficheiro[][]).São também contados o número de linhas do ficheiro e guardados numa variavel (numeroDatas) que é retornada desse array;

---------------------------------------------------------//------------------------------------------------------------------

De seguida, será inserido por parâmetro o período a ser analisado (manhã , tarde ,noite , madrugada, diário, mensal ou anual,respetivamente com os números 11, 12, 13, 14, 2, 3 e 4).

Para este parâmetro serão executados os seguintes métodos de acordo com a informação inserida:

case "11":

passarZero(ana);

passarZero(copia);

k = separar2(ficheiro, numeroDatas, 6, 11, 2, ana);

naousem = separar2(ficheiro, numeroDatas, 6, 11, 2, copia);

for (int i = 0; i < k; i++) {

consumos[i] = ana[i][6];

}

fazgrafico(consumos, k, k, "Consumo geral", "nao");

break;

Exemplo de um caso para o parâmetro da resolução inserido em segundo lugar na linha de comandos.

**4.1.1.2 Método passarZero**

Este método serve para passar os valores dos arrays a 0.

**4.1.1.3 Método separar2**

Serve para separar a parte a analisar para outro array.

-----------------------------------------------------------//----------------------------------------------------------------

É utilizado outro array, o array de inteiros consumos[] para guardar os valores das energias.

**4.1.1.4 Método fazgrafico**

Este método tem como propósito a construção de um gráfico através da biblioteca JavaPlot do java.

No modo não interativo pretendemos guardar diretamente o gráfico num ficheiro PNG portanto não é necessário perguntar ao utilizador se pretende ou não guardar o mesmo,por isso implementamos o que será explicado de seguida ; Colocámos os parâmetros do seguinte modo

public static void fazgrafico(int[] analise, int numeroDatas, int k, String nome, String sim)

O ultimo parâmetro (String sim) permite-nos gravar diretamente para o ficheiro o gráfico realizado:

if (sim.equals("sim")) {

System.out.print("Deseja guardar o gráfico \"sim\" ou \"nao\"? ");

…….

} else {

String title = "Consumo de energia";

File file = new File("statistics\_" + title + ".png");

……..

}

O que nós passamos é a String “não”,(fazgrafico(consumos, k, k, "Consumo geral", "nao");),pelo que, será guardado diretamente no ficheiro sem perguntar ao utilizador o que não faria sentido visto que o modo é não interativo.

-------------------------------------------------------------//--------------------------------------------------------------

O que será inserido nos parâmetros em quarto lugar será o modelo (média móvel simples e média móvel exponencialmente pesada), porém o nosso código está organizado de outra forma pelo que procederei por analisar primeiro o método de ordenação crescente ou decrescente.

No caso de o parâmetro inserido ser 1 (ordem crescente) serão realizados apenas os dois métodos seguintes para ordenar, se for inserido o 0 (ordem decrescente) será ainda realizado o ponto **4.1.1.6.**

OBS: Infelizmente o ficheiro de código que mandámos ia com um erro neste módulo pelo que não está a funcionar a parte de ordenação decrescente.

**4.1.1.5 Método mergesort**

Este é o método de ordenação por nós escolhido por ser aquele que mais se adequava ao nosso projeto no modo como é executado e na velocidade de execução. Este módulo é chamado por si pois este vai dividindo a informação várias vezes. Dentro deste método é chamado outro método que o complementa e é necessário na ordenação dos valores do array.

**4.1.1.5.1 Método merge**

Complementa o método mergesort.

**4.1.1.6 Método inverter**

Serve para inverter a ordem da ordenação realizada de modo a termos uma ordem decrescente.

**4.1.1.7 Método fazgrafico2arr**

Este método serve também para a construção de um gráfico , a única diferença são os parâmetros passados . Tivemos que repetir o código mudando apenas os parâmetros que era necessário mudar.

**4.1.1.8 Método printwriter3**

Serve para guardar a informação em ficheiro CSV das séries temporais.

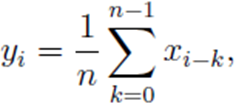
-----------------------------------------------------------//----------------------------------------------------------------

É feito de novo o método **4.1.1.3**.

No seguimento temos o quarto parâmetro em que será escolhido o modelo , média móvel simples ou média móvel exponencialmente pesada . No caso de inserirmos 1 será efetuada a média móvel simples , no caso de colocarmos 2 será feita a média móvel exponencialmente pesada.

**4.1.1.9 Método mediamovelsimplessem**

Aqui é calculada a média móvel simples pela fórmula:



É também onde é utilizado o valor inserido no penúltimo parâmetro neste caso o valor de n no cálculo da média.

**4.1.1.9.1 Método fazgrafico2simples**

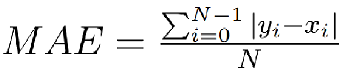
Método para fazer gráfico .Como são passados parâmetros diferentes tivemos que fazer outro método.

**4.1.1.9.2 Método printwriter2**

Outro dos métodos utilizados para fazer print da informação dos gráficos em ficheiro CSV.

**4.1.1.9.3 Método erro\_medio\_absoluto**

Método utilizado para cálculo do erro médio absoluto.Pela fórmula :



em que xi são os termos que representam a série original, yi é a série resultante da aplicação do filtro e N é o número de observações da série.

**4.1.1.10 Método media\_movel\_exponencialmente\_pesadasem**

Onde é feito o cálculo da média móvel exponencialmente pesada pela fórmula:



É também onde é utilizado o valor inserido no penúltimo parâmetro neste caso o valor de alpha no cálculo da média .

**4.1.1.10.1 Método fazgrafico2**

Método para fazer gráfico .Como são passados parâmetros diferentes tivemos que fazer outro método.

São também executados os métodos **4.1.1.9.2**;**4.1.1.9.3**.

-----------------------------------------------------//----------------------------------------------------------------------

O último parâmetro a ser inserido é o momento da previsão e o seu valor depende da resolução especificada.

Se a resolução for mensal ,ou seja , se for inserido o numero 3 no terceiro parâmetro então será realizado o método cortarmes com o valor inserido no último parâmetro.

**4.1.1.11 Método cortarmes**

Serve para separar a informação inserida que virá toda junta com o seguinte aspeto 12052017 por exemplo e dividi-la em dois.

----------------------------------------------------------//-----------------------------------------------------------------

Se a resolução for da manhã,tarde,noite,madrugada ou diário ou seja, se o numero inserido for respetivamente 11,12 ,13,14 ou 2 será realizado o método cortardia com o valor inserido no último parâmetro.

**4.1.1.12 Método cortardia**

Do mesmo modo do método **4.1.1.11** serve também para separar a informação só que desta vez em três.

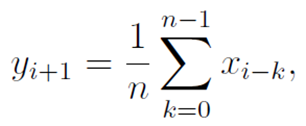
---------------------------------------------------------//------------------------------------------------------------------

Por último se esta for uma resolução anual não será preciso dividir informação pois apenas será inserido apenas o ano .

Com o penúltimo parâmetro inserido na linha de comandos é também determinado o modelo de previsão a ser efetuado. Se for 1 media movel simples e 2 média móvel exponencialmente pesada.

**4.1.1.12 Método mediamovelsimplesprevisaosem**

Este método é utilizado para fazer a previsão com a média móvel simples com a seguinte fórmula:



É realizado o método **4.1.1.9.2.**

**4.1.1.13 Método media\_exponencialmente\_pesada\_previsaosem**

Este método é utilizado para fazer a previsão com a média móvel exponencialmente pesada com a seguinte fórmula:



É realizado o método **4.1.1.9.2.**

-------------------------------------------------------//--------------------------------------------------------------------

OBS : Houve uma falha na implementação do trabalho pois no modo interativo faltou-nos colocar a parte do cálculo do número de observações e o histograma consequente .

**4.1.2 Modo interativo**

No método interativo a aplicação é chamada da linha de comandos utilizando o comando: java -jar nome programa.jar -nome ts nome da serie temporal.csv. Neste modo de apresentação todos os dados são apresentados na linha de comandos à exceção dos gráficos que são apresentados através do JavaPlot (http://javaplot.panayotis.com/index.html) que consiste num interface para a aplicação do gnuplot.

Este modo envolve um ciclo em que primeiro é inserida a parte a analisar e de seguida é inserida a opção a realizar ,sendo sempre preciso inserir a parte a analisar antes de realizar qualquer uma das opções, este ciclo acaba quando se insere o inteiro 0 ou a String “sair”.

Primeiramente será lido o ficheiro através do método **4.1.1.1** e será retornado o número de linhas do mesmo.

De seguida aparecerá ao utilizador o menu seguinte:

Introduza qual parte pretende analisar

11-Manhã

12-Tarde

13-Noite

14-Madrugada

2-Diário

3-Mensal

4-Anual

5-Escolher outro ficheiro

0-Fechar o programa

O utilizador escolherá uma das opções .

Se for inserido 0 o programa será fechado.

Nas opções 11,12,13,14,2,3,4 o que acontecerá será muito parecido:

case "11":

passarZero(ana);

passarZero(copia);

k = separar2(ficheiro, numeroDatas, 6, 11, 2, ana);

naousem = separar2(ficheiro, numeroDatas, 6, 11, 2, copia);

break;

Exemplo do caso 11.

Serão feitos os métodos **4.1.1.2** e **4.1.1.3**.

Na opção 5 o utilizador poderá escolher outro ficheiro para ser analisado , porém por falha nossa não colocámos a possibilidade de o utilizador a seguir a escolher esta opção poder escolher qual parte pretende analisar , apenas após o ciclo ser feito uma vez será possível escolher a parte a analisar.

Será de novo utilizado o método **4.1.1.2** de modo a passar a zero os valores do ficheiro inicial e da copia.

De seguida será lido o novo ficheiro através do método **4.1.1.1**.

Após escolher a parte a ser analisada aparecerá o seguinte menu:

O que pretende fazer?

1-Visualizar gráfico de consumos

2-Visualizar as distribuições de observações

3-Ordenar valores

4-Efetuar uma Filtragem

5-Efetuar uma Previsão

6-Escolher outro ficheiro

0-Fechar o programa

Na primeira opção é possível visualizar o gráfico de consumos.Será guardada a informação dos consumos dentro de um array de inteiros consumos[].A seguir será feito o método **4.1.1.4** que é o método que permite a realização do gráfico , porém desta vez será perguntado ao utilizador se quer ou não guardar o gráfico num ficheiro PNG e a respetiva série temporal num ficheiro CSV, pelo que, ao contrário do modo não interativo é passado como parâmetro a String “sim” (explicação em **4.1.1.4**).

No caso da segunda opção poderá ser visualizada a distribuição de observações tanto em forma de intervalo como através de um histograma abordado no seguinte método.

**4.1.2.1 Método fazhistograma**

Método que serve para a realização de um histograma ou gráfico de barras.

---------------------------------------------------------------//------------------------------------------------------------

No caso 3 será para escolher um tipo de ordenação ,crescente ou decrescente.

Poderá inserir 1 para a crescente e 0 para a decrescente.

Ordem crescente serão realizados os métodos pela seguinte ordem:**4.1.1.5**;**4.1.1.7**;**4.1.1.8**

Ordem decrescente serão realizados os métodos pela seguinte ordem:**4.1.1.5**;**4.1.1.6**;

**4.1.1.7**;**4.1.1.8**

Na opção 4 poderá ser efetuada a filtragem dos dados.

Será realizado o seguinte método:

**4.1.2.2 Método submenu**

Apresentará a seguinte informação na linha de comandos:

Qual modo pretende usar?

Média Móvel Simples-1

Média Móvel Exponencialmente Pesada-2

---------------------------------------------------------------//------------------------------------------------------------

Ao escolhermos 1 será utilizada a média móvel simples.

**4.1.2.3 Método mediamovelsimples**

Este método também fará a média móvel simples com a mesma fórmula do **4.1.1.9.**

Dentro deste serão chamados os seguintes métodos: **4.1.1.9.1**;**4.1.1.9.2**;**4.1.1.9.3**.

---------------------------------------------------------//------------------------------------------------------------------

Ao escolhermos 2 será utilizada a média móvel exponencialmente pesada.

**4.1.2.4 Método media\_movel\_exponencialmente\_pesada**

Este método também fará a média móvel exponencialmente pesada com a mesma fórmula do **4.1.1.10**.

Dentro deste serão chamados os seguintes métodos : **4.1.1.10.1**; **4.1.1.9.2**;**4.1.1.9.3.**

----------------------------------------------------------//-----------------------------------------------------------------

No caso de ser inserido 5 será feita uma previsão .

Primeiro aparecerá o **4.1.2.2** de modo a ser possível escolher se será feita uma previsão através da média móvel simples ou exponencialmente pesada .

No caso de ser escolhido o 1 , ou seja , a média móvel simples será efetuado o método mediamovelsimplesprevisao.

**4.1.2.5 Método mediamovelsimplesprevisao**

Este método é utilizado para fazer a previsão com a média móvel simples.

É realizado o método **4.1.1.9.2.**

------------------------------------------------------//---------------------------------------------------------------------

No caso de ser escolhido 2 , será executado o método seguinte.

**4.1.2.6 Método media\_exponencialmente\_pesada\_previsao**

Este método é utilizado para fazer a previsão com a média móvel exponencialmente pesada.

É realizado o método **4.1.1.9.2.**

---------------------------------------------------//------------------------------------------------------------------------

OBS:Não era suposto aparecer a opção “Escolher outro ficheiro” no segundo menu pois esta quando escolhida não tem qualquer output.

Se em qualquer dos modos os argumentos forem mal inseridos a seguinte mensagem aparecerá no ecrã "Argumentos introduzidos de maneira errada".

**4.1.3 Outros métodos de ordenação**

Deixámos no nosso código os dois outros métodos de ordenação, o bubblesort e o insertionsort de modo a podermos verificar através de um outro ficheiro a rapidez de cada um.Concluímos que o mergesort era o mais rápido e por isso optámos por utilizá-lo.

**4.1.4 Diagrama de Interligação de módulos**

No nosso código os módulos interligados, como foi referido anteriormente, são os seguintes:

mediamovelsimples(4.1.2.3) / mediamovelsimplessem(4.1.1.9)

media\_movel\_exponencialmente\_pesada(4.1.2.4) / media\_movel\_exponencialmente\_pesadasem(4.1.1.13)

mergesort(4.1.1.5)

mergesort(4.1.1.5)

fazgrafico2simples(4.1.1.9.1)

mergesort(4.1.1.5)

printwriter2(4.1.1.9.2)

merge(4.1.1.5.1)

erro\_medio\_absoluto(4.1.1.9.3)

mediamovelsimplesprevisao(4.1.2.3) / mediamovelsimplesprevisaosem(4.1.1.9)

media\_exponencialmente\_pesada\_previsao(4.1.2.6) / media\_exponencialmente\_pesada\_previsaosem(4.1.1.10)

printwriter2(4.1.1.9.2)

Os outros módulos são todos independentes embora pudéssemos ter otimizado consideravelmente mais se tivéssemos optado por começar a desenvolver o código pela parte não interativa uma vez que alguns módulos podiam ser unificados.

## **4.2 Tempo de ordenação**

Com o ficheiro “tempo.java” é possível analisar o tempo de qualquer um dos métodos de ordenação.

## **4.3 Testes unitários**

O ficheiro de testes permite o teste da maioria dos métodos .

@Test

public void testLer\_ficheiro() throws Exception {

System.out.println("ler\_ficheiro");

int[][] ficheiro = new int[30000][7];

String args = "DAYTON.csv";

int expResult = 22680;

int result = hehe.ler\_ficheiro(ficheiro,args);

assertEquals(expResult, result);

}

Exemplo de teste unitário.

# **5 Resultados**

Nesta secção vamos mostrar um dos resultados obtidos usando o ficheiro disponibilizado pelo docente.

Como exemplo vamos usar uma análise mensal. Quando optámos pelo gráfico que apenas dispõe os valores do ficheiro, obtemos o seguinte gráfico:

Uma imagem com captura de ecrã

Descrição gerada automaticamente

Quando optámos pelo histograma com o número de observações obtemos o seguinte histograma:

Uma imagem com captura de ecrã, texto

Descrição gerada automaticamente

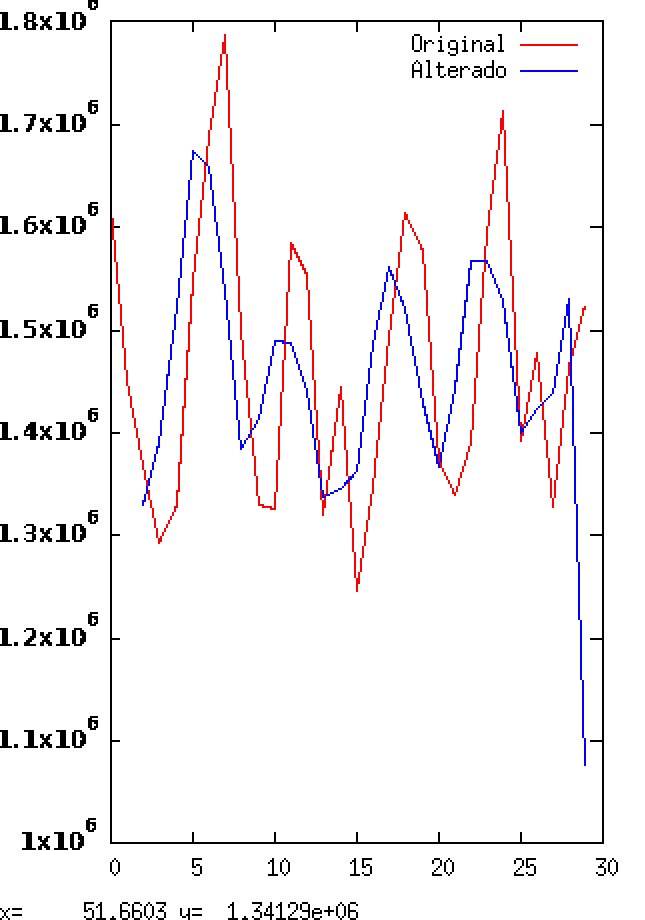
Quando optámos pela ordenação podemos obter dois resultados, o crescente e o decrescente, e o resultado obtido é os seguintes gráficos respetivamente:

Uma imagem com texto, mapa

Descrição gerada automaticamente Uma imagem com texto, mapa

Descrição gerada automaticamente

Quando optámos por efetuar uma filtragem podemos optar por dois métodos, a da média móvel simples e a da média móvel exponencialmente pesada, onde temos ainda de introduzir os valores de N ou α, neste caso utilizámos para N o valor 3 e para α o valor 0.3. Os gráficos obtidos foram os seguintes, sendo o primeiro o da média móvel simples e o segundo o da média móvel exponencialmente pesada:

Uma imagem com texto

Descrição gerada automaticamente

Para além dos gráficos obtemos também o erro médio absoluto que para a média móvel simples obtivemos 207857 e para a média móvel exponencialmente pesada obtivemos 122589.

Quando optámos por efetuar uma previsão podemos optar por dois métodos, a da média móvel simples e a da média móvel exponencialmente pesada, onde temos ainda de introduzir os valores de N ou α, neste caso utilizámos para N o valor 3 e para α o valor 0.3. Para a previsão temos ainda que introduzir o mês, visto que estamos a fazer de forma mensal que pretendemos prever, neste caso escolhemos o mês 4 do ano 2017. Os valores que obtivemos para a média móvel simples foi 1440178 e para a média móvel exponencialmente pesada foi 1449903.

## **5.1 Análise dos resultados**

Analisando agora os resultados obtidos para o gráfico que dispõe os valores pode-se reparar a queda no último mês uma vez que este se encontra incompleto.

Tendo em conta o histograma podemos reparar que apenas um mês se encontra a mais de vinte por cento abaixo da média tal como apenas um mês se encontra a mais de vinte por cento acima da média.

Quanto à ordenação podemos ver que está correta em ambos os casos.

Em relação às filtragens podemos logo notar num erro visto que quando utilizada a média móvel simples o gráfico não vai até ao último mês contrariamente ao que deveria. Após uma análise conseguimos resolver o problema e verificar que o gráfico que deveríamos ter obtido era o seguinte:

Uma imagem com captura de ecrã

Descrição gerada automaticamente

Verificando as previsões podemos reparar que ambas estão muito próximas e sendo o valor real obtido nesse mês de aproximadamente 1245830 podemos verificar que a média móvel simples deu um valor mais próximo ao da realidade pelo que o consideramos o melhor método dos dois. Este valor disperso do original dá-se porque existe uma quebra como podemos verificar no gráfico no ponto 15, ponto este atribuído ao mês 4 de 2017.

# **6 Conclusão**

Este trabalho realizado revelou-se bastante fulcral e importante, na medida em que, adquirimos novos conhecimentos que muito provavelmente nos vão ser bastante úteis no futuro e permitiu-nos aplicar os conhecimentos teóricos que havíamos anteriormente apreendido.

Além dos conhecimentos teóricos já referidos, também aprendemos a trabalhar como um grupo, devido à metodologia de trabalho utilizada (EduScrum).

Ao início, o projeto revelou-se trabalhoso e bastante difícil de executar, mas à medida que fomos investigando conseguimos ultrapassar as nossas dificuldades e progredir na realização do mesmo.

O trabalho desenvolvido pelo grupo foi bastante positivo, mas podíamos melhorar alguns aspetos para o futuro. Por exemplo, se, inicialmente, tivéssemos analisado melhor o enunciado e percebêssemos o que tínhamos que fazer, teríamos conseguido realizar o trabalho de forma mais eficaz. Ou se tivéssemos começado o trabalho pela parte não interativa uma vez que assim podíamos ter reduzido a quantidade de módulos.

**7 Referências**

Scrum Roles Demystified (2019, Dezembro). Retrieved from:<https://www.scrumalliance.org/agile-resources/scrum-roles-demystified>

Thomas Williams & Colin Kelley (2019, Dezembro). Gnuplot 5.2. Retrieved from: <http://www.gnuplot.info/docs_5.2/Gnuplot_5.2.pdf>

# **ANEXOS**

**ANEXOS**

## **ANEXO A\_Testes Unitários**

**Uma imagem com captura de ecrã

Descrição gerada automaticamente**

Uma imagem com captura de ecrã

Descrição gerada automaticamente

Uma imagem com captura de ecrã

Descrição gerada automaticamente

Uma imagem com captura de ecrã

Descrição gerada automaticamente

Uma imagem com captura de ecrã

Descrição gerada automaticamente

Uma imagem com captura de ecrã

Descrição gerada automaticamente