

Estudo da Pandemia Covid19

1 Preâmbulo

A unidade curricular Laboratório/Projeto I é uma unidade que tem por objetivo consolidar e estender os conhecimentos de todas as unidades curriculares que funcionam no primeiro semestre. Esta unidade curricular inclui também um módulo de competências pessoais e métodos de trabalho que visam preparar os alunos para trabalhar eficazmente em equipas, melhorar a elaboração de relatórios e desenvolver competências de apresentação oral.

Assim, com o projeto descrito neste enunciado pretende-se que os alunos desenvolvam uma aplicação em linguagem *Java* (Horstmann, 2015) onde apliquem um processo básico de desenvolvimento de aplicações informáticas, valorizando todas as fases do ciclo de desenvolvimento, desde a análise e conceção aos testes de validação. Pretende-se também que os alunos elaborem um relatório que descreva a aplicação concebida, o processo de desenvolvimento e que apresentem e critiquem os resultados obtidos.

2 Objetivos

O projeto a desenvolver no corrente ano letivo é uma aplicação informática para a instituição *Monitorização da Saúde em Portugal* (MSP) ¹ que permita a esta instituição estudar e prever a evolução da pandemia Covid19.

A MSP regista dados diários sobre a evolução da pandemia em ficheiros CSV, registando dados acumulados e dados totais diários para os diferentes estados Covid19 em que um português se pode encontrar. Em concreto, regista o número total de não infetados, o número total de infetados, o número total de hospitalizados, o número total de internados em Unidades de Cuidados Intensivos (UCI) e o número total de óbitos que estão registados num determinado dia. Regista também o número acumulado de infetados, o número acumulado de hospitalizados, o número acumulado de internados em UCI e o número acumulado de mortos até um determinado dia em resultado da infeção Covid19. Estes dois tipos de dados Covid19 são sempre registados em ficheiros distintos (no moodle estão disponíveis ficheiros que são exemplos para cada um dos dois tipos de ficheiros/registos). Possuindo estes dados em bruto, e considerando a sua missão, a MSP pretende compreender fenómenos temporais/sazonais, fazer uma análise comparativa entre períodos temporais e prever a evolução da pandemia. Nas subsecções que se seguem serão identificados em detalhe os objetivos da MSP.

2.1 Compreensão dos Dados

Considerando a existência de dois tipos de registos de dados, os funcionários da MSP pretendem explorar os dados e obter um conhecimento aprofundado dos mesmos. Assim, os funcionários pretendem que a aplicação disponibilize duas funcionalidades que permitam:

- visualizar o número total de infetados, o número total de hospitalizados, o número total de internados em UCI e o número total de óbitos que estão registados num determinado dia em resultado da infeção Covid19;
- visualizar o número de novos casos, o número de novas hospitalizações, o número de novos internados em UCI, e o número de óbitos diários em resultado da infeção Covid19.

¹Empresa fictícia

Além da análise diária destes dados a aplicação deverá permitir fazer uma análise a diferentes resoluções temporais (análise diária, semanal ou mensal). Os funcionários pretendem especificar um intervalo de datas e uma determinada resolução temporal e a aplicação deve permitir visualizar a informação pretendida.

2.2 Análise Comparativa

A aplicação deve incluir também um conjunto de funcionalidades que permitam realizar uma análise comparativa dos números Covid19 para diferentes intervalos de tempo. Por exemplo, um funcionário da MSP poderá utilizar a aplicação para comparar os números do quarto trimestre de 2020 com os números do quarto trimestre de 2021. A este nível, a aplicação deve incluir duas funcionalidades que permitam:

- comparar o número total de infetados, o número total de hospitalizações, o número total de internados em UCI e o número total de óbitos que estão registados num determinado dia em resultado da infeção Covid19, para cada período temporal especificado;
- comparar o número de novos casos, o número de novas hospitalizações, o número de novos internados em UCI, e o número de óbitos diários em resultado da infeção Covid19, para cada período temporal especificado.

Qualquer que seja a funcionalidade escolhida, a aplicação deve sempre apresentar um vetor de diferenças que evidencie as diferenças entre os dois períodos considerados e deve apresentar a média e o desvio padrão para cada período considerado.

2.3 Previsão da Evolução da Pandemia

Para melhor antecipar a evolução da pandemia, a aplicação deve incluir uma funcionalidade que permita prever o número total de não infetados / número total de infetados / número total de hospitalizações / número total de internados em UCI e número total de mortos que estão registados/existem para um determinado dia especificado pelo utilizador da aplicação. Para fazer esta previsão será considerado apenas o histórico de dados relativo ao número total de não infetados / número total de infetados / número total de hospitalizações / número total de internados em UCI e número total de óbitos que estão registados/existem num determinado dia. I.e., o registo de números acumulados não será utilizado para efetuar qualquer tipo de previsão.

Esta previsão deve considerar o histórico de dados mais recente disponível (i.e., deve considerar o dia mais próximo do dia que se pretende prever e para o qual temos dados de histórico) e a matriz de transições/Markov (ver Secção 2.5 de (Larson, 2012)) que consiste na representação matricial das probabilidades de transição entre estados Covid19 (conjunto de estados considerados: não infetado, infetado, hospitalizado, internamento em UCI e óbito), entre dias consecutivos. Num modelo de Markov, a probabilidade de um sistema estar num determinado estado i num tempo $t + 1$ apenas depende do estado em que estava no tempo t , i.e., $P(x^{t+1}|x^1, x^2, \dots, x^t) = P(x^{t+1}|x^t)$.

Considerando que $x^t = (x_{NaoInfetado}^t, x_{Infetado}^t, x_{Hospitalizado}^t, x_{InternadoUCI}^t, x_{Obito}^t)$ é um vetor que representa o número de não infetados, total de infetados, total de hospitalizados, total de internados em UCI ou total de óbitos num determinado dia t , e P (ver Tabela 1) representa a matriz de transições entre estados Covid19 em que p_{ij} representa a probabilidade de um membro da população portuguesa mudar do estado j para o estado i entre dois dias consecutivos, em que $\sum_{i=1}^5 p_{ij} = 1$. Por exemplo, no caso em que $p_{43} = 1$, então a probabilidade de um hospitalizado, num determinado dia, passar para a UCI no dia seguinte é 1. Dito de outra forma, se $p_{43} = 1$, então no dia seguinte todos os

doentes que foram hospitalizados no dia anterior agravam a sua situação e passam a estar internados na UCI no dia seguinte.

Tabela 1: Matriz de transições entre estados Covid 19

$$P = \begin{matrix} & \begin{matrix} naoInfetado & Infetado & Hospitalizado & InternadoUCI & Obito \end{matrix} \\ \begin{matrix} naoInfetado \\ Infetado \\ Hospitalizado \\ InternadoUCI \\ Obito \end{matrix} & \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} & p_{15} \\ p_{21} & p_{22} & p_{23} & p_{24} & p_{25} \\ p_{31} & p_{32} & p_{33} & p_{34} & p_{35} \\ p_{41} & p_{42} & p_{43} & p_{44} & p_{45} \\ p_{51} & p_{52} & p_{53} & p_{54} & p_{55} \end{pmatrix} \end{matrix}$$

Para estimar o número de não infetados / infetados / hospitalizados / internados em UCI e óbitos num determinado dia $t + k$ poderemos utilizar a equação:

$$\mathbf{x}^{t+k} = P^k \mathbf{x}^t.$$

Assim, a aplicação deve permitir ao utilizador obter uma estimativa do número de não infetados / infetados / hospitalizados / internados em UCI e óbitos para um determinado dia $t + k$ especificado pelo utilizador. O utilizador especifica o dia $t + k$ para o qual pretende obter a estimativa/previsão e a aplicação deve considerar uma matriz de transições/ Markov carregada (a partir de um ficheiro CSV) para apresentar o valor previsto. O utilizador pode identificar um dia $t + k$ que esteja incluído no histórico de dados carregado ou um dia posterior ao último dia carregado. Em qualquer dos casos, o vetor x^t a considerar para fazer a previsão deve ser o dia do histórico de dados mais próximo do dia a prever. Por exemplo, se o histórico de dados contém informação para o período compreendido entre o dia 20/10/2020 e o dia 01/11/2020 e o utilizador pretende fazer previsões para o dia 05/11/2020, então o dia t a considerar na fórmula é o dia 01/11/2020 e o dia $t + k$ corresponde ao dia 05/11/2020. Considerando este mesmo histórico de dados, se o utilizador pretender fazer uma previsão para o dia 25/10/2020, que faz parte do histórico de dados, então o dia t a considerar para fazer a previsão é o dia 24/10/2020, que é o dia mais próximo do dia 25/10/2020.

A MSP também pretende que seja incluída na aplicação uma funcionalidade que permita calcular o número esperado de dias que um indivíduo passa num estado antes de transitar para outro estado. Em particular, até morrer.

Sendo que a matriz P contém quatro estados transientes e um estado absorvente, o estado óbito/morto. Este último estado é designado por absorvente porque não é possível regressar/transitar para qualquer outro estado depois de estar morto. Definindo a matriz Q por eliminação do estado absorvente (ver Tabela 2), e respetivas transições, da matriz P , e considerando I a matriz identidade, podemos definir a matriz:

$$N = \sum_{k=0}^{\infty} Q^k = (I - Q)^{-1}$$

Considerando a matriz N , o número esperado de dias até chegar ao estado óbito (i.e., até morrer)

Tabela 2: Matriz de transições entre estados Covid 19 sem estado absorvente

$$Q = \begin{matrix} & \begin{matrix} NaoInfetado & Infetado & Hospitalizado & InternadoUCI \end{matrix} \\ \begin{matrix} NaoInfetado \\ Infetado \\ Hospitalizado \\ InternadoUCI \end{matrix} & \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \\ p_{41} & p_{42} & p_{43} & p_{44} \end{pmatrix} \end{matrix}$$

quando o indivíduo começa num determinado estado transiente i , é dado pela entrada i do vetor ²:

$$M = \mathbb{1} \times N$$

onde $\mathbb{1}$ é um vetor linha de comprimento 4, cujas entradas tomam todas o valor 1.

A MSP não pretende que sejam utilizadas bibliotecas externas no desenvolvimento do projeto. O cálculo da matriz inversa deve ser realizado recorrendo à decomposição LU da matriz I-Q pelo método de Crout ³ (Press, Teukolsky, Vetterling, & Flannery, 1992).

2.4 Outros requisitos

- A aplicação deve ser desenvolvida em linguagem *Java* e estruturada e organizada em módulos. Será valorizada uma correta decomposição modular e o reaproveitamento de módulos. No final do trabalho deve resultar um UNICO projeto.
- A aplicação deve permitir carregar, a qualquer momento, um ficheiro CSV que regista, para um subconjunto de dias contíguos: o número total de não infetados, o número total de infetados, o número total de hospitalizados com a infeção Covid19, o número total de internados em Unidades de Cuidados Intensivos (UCI) e o número total de óbitos/mortos registados/existem num determinado dia. No moodle está disponível um ficheiro exemplo.
- A aplicação deve permitir carregar, a qualquer momento, um ficheiro CSV que regista, para um subconjunto de dias contíguos: o número de não infetados, o número acumulado de infetados, o número acumulado de hospitalizados com a infeção Covid19, o número acumulado de internados em Unidades de Cuidados Intensivos (UCI) e número acumulado de mortos em resultado da infeção. No moodle está disponível um ficheiro exemplo.
- Sempre que o utilizador pretender, a aplicação deve permitir carregar qualquer um dos ficheiros/registos. Num determinado momento, podemos ter: apenas o registo de dados acumulados carregado; apenas o registo do total casos (de não infetados, infetados, hospitalizados, internados em UCI, óbitos) num determinado dia carregado; ou os dois tipos de registos carregados.
- A aplicação deve permitir carregar, a qualquer momento, um ficheiro TXT contendo uma matriz de transições. No moodle está disponível um ficheiro exemplo.

²https://www.math.umd.edu/~immortal/MATH401/book/ch_absorbing_markov_chains.pdf

³https://webstor.srmist.edu.in/web_assets/srm_mainsite/files/2018/MatrixInversionandeigenvalue.pdf

- A aplicação deve permitir guardar em ficheiro CSV os resultados apresentados na consola.
- A aplicação deve ter um interface simples e intuitivo que permita ao utilizador interagir com a aplicação de forma rápida e minimizando os erros. A interface da aplicação será em modo de texto e deverá incluir um conjunto de menus que permita aceder às funcionalidades de forma rápida. A interface da aplicação também deve permitir parametrizar cada uma das funcionalidades selecionadas, de acordo com o que é apresentado nos objetivos do projeto.
- A aplicação pode ser executada em modo interativo ou sem interação por parte do utilizador.
 - No modo interativo, a aplicação deverá ser chamada da linha de comandos utilizando o comando: *java -jar nome_programa.jar*. Neste modo todos os parâmetros necessários são solicitados em tempo de execução ao utilizador.
 - No modo não interativo, o utilizador especifica todos os parâmetros necessários à execução da aplicação na linha de comando, definindo: o ficheiro contendo o registo do total de casos Covid19 que existem num determinado dia, o ficheiro contendo o registo acumulado de casos Covid19, o ficheiro contendo a matriz de transições, a resolução temporal a utilizar na análise descritiva dos dados (diária, semanal ou mensal), o intervalo de datas a considerar, o dia a prever e o nome do ficheiro de saída (um ficheiro txt) onde é guardada toda a informação que seria apresentada na consola se as funcionalidades fossem executadas. Neste modo o comando terá a seguinte sintaxe:
java -jar nome_programa.jar -r X -di DD-MM-AAAA -df DD-MM-AAAA -di1 DD-MM-AAAA -df1 DD-MM-AAAA -di2 DD-MM-AAAA -df2 DD-MM-AAAA -T DD-MM-AAAA registoNumeroTotalCovid19.csv registoNumerosAcumuladosCovid19.csv matrizTransicao.txt nome_ficheiro_saida.txt. Neste caso:
 - o valor associado ao parâmetro r (X) identifica a resolução temporal. X toma os valores 0, 1 e 2, respetivamente, para resolução diária, semanal e mensal.
 - as datas associadas aos parâmetros di e df (DD-MM-AAAA) representam, respetivamente, a data de início e fim do intervalo a considerar para visualizar os dados. No modo não interativo, o mesmo intervalo de datas será considerado para análise de novos casos e casos ativos.
 - as datas associadas aos parâmetros di1, df1, di2 e df2 (DD-MM-AAAA) representam as datas a considerar para comparar dados de dois períodos. As datas associadas aos parâmetros di1 e df1 são definidas para representar um período e as datas associadas aos parâmetros di2 e df2 são definidas para representar o segundo período a considerar. No modo não interativo, os dois períodos definidos serão considerados tanto para a análise de novos casos como para a análise do total de casos (total de não infetados, infetados, hospitalizados, internados em UCI e óbitos) registados/existente num determinado dia.
 - a data associada ao parâmetro T (DD-MM-AAAA) representa o dia para o qual o utilizador quer obter previsões para o total de casos (total de não infetados, infetados, hospitalizados, internados em UCI e óbitos) registados/que existem num determinado dia.
 - neste modo, a funcionalidade que permite calcular o número médio de dias até à morte é sempre executada e não requer que seja especificado qualquer parâmetro.
 - caso o utilizador não pretenda fazer previsões, nem estimar o número médio de dias até à morte, o comando deverá ser definido sem que seja identificado o ficheiro que regista o número de casos Covid19 total / que existem num determinado dia, sem identificar a matriz de transição e sem identificar o parâmetro T. Neste caso o comando

a executar seria `java -jar nome_programa.jar -r X -di DD-MM-AAAA -df DD-MM-AAAA -di1 DD-MM-AAAA -df1 DD-MM-AAAA -di2 DD-MM-AAAA -df2 DD-MM-AAAA registoNumerosAcumuladosCovid19.csv nome_ficheiro_saida.txt`.

- caso o utilizador apenas pretenda fazer previsões e estimar o número médio de dias até à morte, o comando deverá ser executado sem que seja identificada a resolução temporal, sem identificar o ficheiro que regista o número acumulado de casos Covid19 e sem identificar o intervalo de datas. Neste caso o comando a executar seria `java -jar nome_programa.jar -T DD-MM-AAAA registoNumerosTotalCovid19.csv matrizTransicao.csv nome_ficheiro_saida.txt`.

- Todos os métodos desenvolvidos terão, obrigatoriamente, de estar associados a testes unitários. Por exemplo, se o aluno criar o método `find_max_coeficiente(matriz)` para determinar o maior coeficiente da `matriz`, também deve criar o método `test_find_max_coeficiente(matrix, maxCoeficienteEsperado)` (ver Algoritmo 1), em que `maxCoeficienteEsperado` é o valor do maior coeficiente da matriz. Estes testes são extremamente úteis para determinar se os métodos estão de acordo com a sua especificação e se a edição destes não alterou a sua funcionalidade.

```

Bool test_find_max_coeficiente(matriz, maxCoeficienteEsperado)
{

    maxCoeficiente = find_max_coeficiente(matriz);

    if(maxCoeficienteEsperado==maxCoeficiente)
        return True;
    else
        return False;

}

```

Algoritmo 1: Exemplo de métodos de teste unitários

3 Trabalho a Desenvolver

O trabalho a realizar até ao dia 23 de Janeiro de 2022 consiste no:

- Estudo de cadeias de Markov (ver Secção 2.5 de (Larson, 2012)).
- Estudo da decomposição LU e método de Crout ⁴ (Press et al., 1992).
- Desenvolver uma aplicação informática que permita estudar a pandemia Covi19, dando resposta aos objetivos da MSP.
- Elaborar um relatório onde: é feita uma breve introdução às cadeias de Markov, à decomposição LU, ao método de Crout e ao cálculo da matriz inversa utilizando a decomposição LU; é descrita a metodologia de trabalho que utilizaram para desenvolver a aplicação; é descrita

⁴https://webstor.srmist.edu.in/web_assets/srm_mainsite/files/2018/MatrixInversionandeigenvalue.pdf

a implementação da aplicação; e é feita uma análise dos resultados. A descrição da implementação da aplicação deve incluir um diagrama que identifique claramente os módulos e suas dependências. A apresentação dos conceitos de Álgebra Linear deve incluir exemplos ilustrativos.

$$A = \begin{pmatrix} 0.9995 & 0.03 & 0.002 & 0.001 & 0 \\ 0.0005 & 0.96 & 0.004 & 0.015 & 0 \\ 0 & 0.007 & 0.98 & 0.02 & 0 \\ 0 & 0.003 & 0.01 & 0.95 & 0 \\ 0 & 0 & 0.004 & 0.014 & 1 \end{pmatrix}$$

$$B = \begin{pmatrix} 0.9995 & 0.049 & 0.005 & 0.001 & 0 \\ 0.0005 & 0.95 & 0.005 & 0.018 & 0 \\ 0 & 0.0009 & 0.98 & 0.02 & 0 \\ 0 & 0.0001 & 0.0085 & 0.947 & 0 \\ 0 & 0 & 0.0015 & 0.014 & 1 \end{pmatrix}$$

- No relatório também devem explorar um caso de estudo. Neste caso de estudo as equipas devem analisar a evolução da pandemia no período compreendido entre 01-10-2021 e 31-12-2021. Neste estudo devem considerar os registos Covid19 disponíveis no moodle, dois modelos de previsão, as matrizes de transição A e B, para prever, para os 7 primeiros dias de Janeiro de 2022, o número total de não infetados, o número total de infetados, o número total de hospitalizados, o número total de internados em Unidades de Cuidados Intensivos (UCI) e o número total de óbitos registados / que existem num determinado dia. A população inicial a considerar para prever os sete primeiros dias de Janeiro de 2022 é a população do último dia de Dezembro de 2021. A qualidade de cada modelo (um associado à matriz A e o outro associado à matriz B) deverá ser avaliada através do cálculo da soma do valor absoluto da diferença entre as previsões realizadas por cada modelo e os valores observados. Supondo que $x_{previsaoInternadosUCI}$ e $x_{observadosInternadosUCI}$ são vetores que representam, respetivamente, as previsões e os valores observados para o período de 7 dias a considerar, então o erro do modelo ao prever, por exemplo o número de internados em UCI, seria obtido através da seguinte fórmula: $Erro_{InternadosUCI} = \sum_{t=1}^7 |x_{observadosInternadosUCI}^t - x_{previsaoInternadosUCI}^t|$. No moodle estão disponíveis dois ficheiros contendo todos os dados necessários ao estudo, um ficheiro registando o acumulado de casos Covid19 e outro ficheiro registando o total de casos (total de não infetados, infetados, hospitalizados, internados em UCI e óbitos) que existem num determinado dia para o período em estudo. Os dados para o período compreendido entre 01-01-2022 e 07-01-2022 estão disponíveis apenas para avaliar a qualidade de cada modelo.

Todos os resultados apresentados devem ser obtidos recorrendo à aplicação desenvolvida pelo grupo. Apenas o cálculo do erro das estimativas/previsões deve ser efetuado sem utilizar a

aplicação. Ao analisar os dados devem considerar que estamos perante uma dinâmica da população em que a natalidade, migrações, mortalidade em resultado de causas não COVID19 não tem qualquer impacto na população a considerar no estudo. Os alunos devem discutir os resultados obtidos.

4 Método de Trabalho

- Todos os alunos devem utilizar a metodologia de trabalho definida no *eduScrum* (Delhij & Solingen, 2013). Cada um dos grupos deve escolher dois *Scrum Masters*, que serão responsáveis por gerir a execução de tarefas. O primeiro *Scrum Master* é responsável pela gestão de tarefas desde o início do projeto e até ao dia 12 de Janeiro às 24h00m, enquanto o segundo *Scrum Master* é responsável por gerir as tarefas entre o dia 13 de Janeiro às 00h00m e a data de submissão final do projeto. Para atingir os objetivos, o grupo deve utilizar a ferramenta *Trello* e registar as tarefas do projeto, a atribuição de tarefas, o estado de cada tarefa e as tarefas concluídas.
- A aplicação será desenvolvida utilizando o sistema de controle de versões *Git* e o *Bitbucket* (<https://bitbucket.org>). Todos os alunos terão que criar uma conta no *Bitbucket* com o endereço de email do ISEP (i.e. XXXXXXX@isep.ipp.pt) e cada grupo terá que criar um repositório. A designação do repositório deve seguir o formato: "LAPR1_TurmaDAB-Grupo01". O repositório deve ser partilhado com todos os docentes que lecionam a turma onde o grupo está inserido.
- O grupo deve guardar todo o material desenvolvido para a realização do projeto na pasta Ficheiros do seu canal privado no Teams. Não é necessário incluir nesta pasta o código que está disponível no repositório do *BitBucket*.

5 Submissão do Trabalho

Datas e entregas de trabalho a efetuar através do Moodle:

- Dia 23 de Janeiro de 2022, até às 23h00m
 - Submeter o projeto desenvolvido, versão final, incluindo toda a estrutura de diretorias e ficheiros do projeto (incluindo o executável), num único ficheiro comprimido (ZIP).
 - Relatório em formato pdf não ultrapassando as 25 páginas. A escrita do relatório deve seguir as instruções formais e o modelo disponibilizado nas aulas TP (módulo de competências).

Nota: Os ficheiros deverão identificar, obrigatoriamente, a designação do grupo e a turma a que os alunos pertencem (Exemplo: "LAPR1_TurmaDAB-Grupo01-projeto.ZIP" e "LAPR1_TurmaDAB-Grupo01-relatorio.PDF").

Referências

- Delhij, A., & Solingen, R. (2013). *The eduscrum guide: The rules of the game*. (Disponível em http://eduscrum.nl/file/CKFiles/The_eduScrum_Guide_EN_December_2013_1.0.pdf)
- Horstmann, C. (2015). *Big java: Early objects, 6th edition*. Wiley. Retrieved from <https://books.google.pt/books?id=ib12CwAAQBAJ>

Larson, R. (2012). *Elementary linear algebra*. Cengage Learning.
Press, W. H., Teukolsky, S. A., Vetterling, W. T., & Flannery, B. P. (1992). *Numerical recipes in c* (Second ed.). Cambridge, USA: Cambridge University Press.