

Instituto Superior Técnico

3º Ano – 2º Semestre

Ano Letivo 2016/2017

Mestrado Integrado em Engenharia Biomédica



Relatório

Projeto de Algoritmos e Modelação Computacional

Agrupamento (*Clustering*) para Modelos Farmacocinéticos

Optimização com *Simulated Annealing*

Professor Paulo Mateus

Trabalho realizado pelo **Grupo 22:**

Constança Sousa, 81073

Cátia Fortunato, 81497

Miguel Bhagubai, 81789

Manual de Utilizador

A interface gráfica foi construída através do *WindowBuilder*. Depois de se seleccionar o executável *Login*, surge a caixa de diálogo apresentada abaixo:

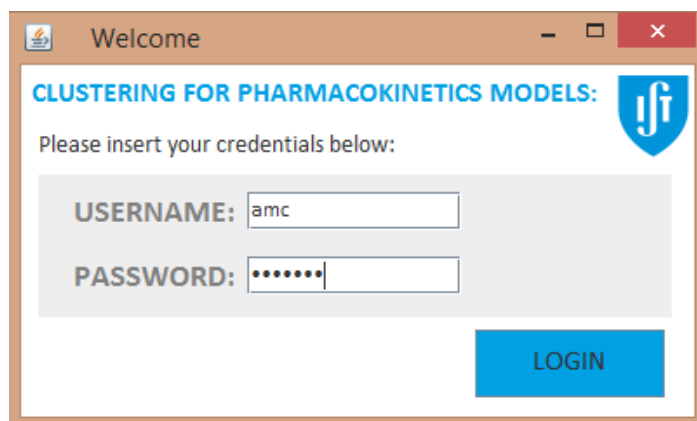
A screenshot of a Windows-style dialog box titled "Welcome". The main heading is "CLUSTERING FOR PHARMACOKINETICS MODELS:" followed by a logo. Below the heading, it says "Please insert your credentials below:". There are two input fields: "USERNAME:" with the text "amc" and "PASSWORD:" with masked characters ".....". A blue "LOGIN" button is at the bottom right.

Figura 1 - Caixa de diálogo inicial.

- *USERNAME*, *PASSWORD*: Inserir as credenciais de acesso ao programa. O utilizador deve ter em atenção que as caixas são sensíveis a maiúsculas, espaços, acentuação e caracteres especiais. Caso não se insira as credenciais certas, é mostrado um *pop-up* com "Invalid Username or Password, try again".

Premindo o botão LOGIN, e inserindo as credenciais corretas, surge outra janela, apresentada na Figura 2, onde o utilizador tem as seguintes opções.

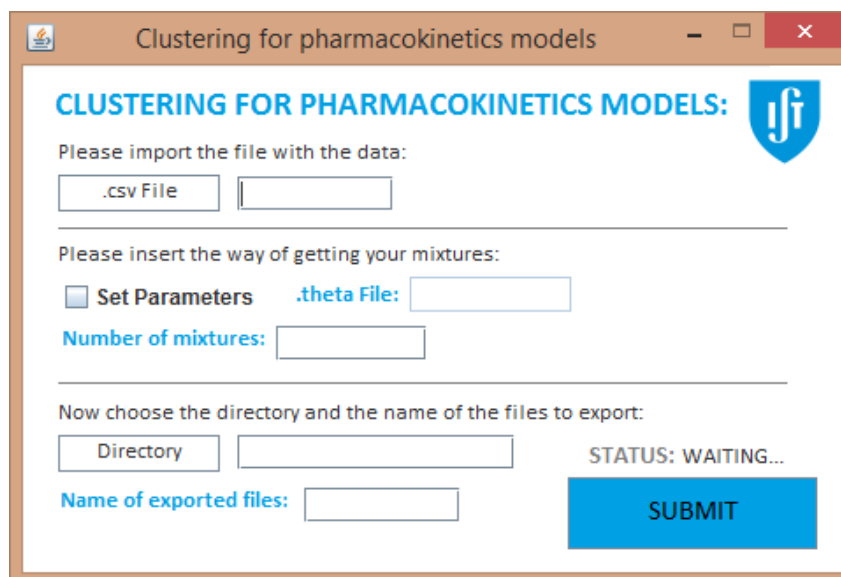
A screenshot of a Windows-style dialog box titled "Clustering for pharmacokinetics models". The main heading is "CLUSTERING FOR PHARMACOKINETICS MODELS:" followed by a logo. Below the heading, it says "Please import the file with the data:". There is a ".csv File" input field. Below that, it says "Please insert the way of getting your mixtures:". There is a checkbox "Set Parameters" and a ".theta File:" input field. Below that, it says "Number of mixtures:" with an input field. Below that, it says "Now choose the directory and the name of the files to export:". There is a "Directory" input field and a "Name of exported files:" input field. A blue "SUBMIT" button is at the bottom right. The status "STATUS: WAITING..." is displayed above the button.

Figura 2 - Caixa de diálogo onde o utilizador insere os parâmetros necessários para o funcionamento do algoritmo.

- *.csv File*: Importar o ficheiro no formato .csv que contém os dados da Amostra que se pretende utilizar (ex: EM20.csv). Este ficheiro deve estar organizado por índice, tempo e valor.
- *Set Parameters*: Importar um ficheiro .theta com a aproximação inicial de θ . Ao seleccionar esta *checkbox*, o utilizador não consegue inserir o número de misturas, dado que, para esta opção, não é necessário para correr o algoritmo.

- *Number of mixtures*: Inserir o número natural ($M \in \mathbb{N}$) de misturas pretendido.
- *Directory*: Escolher a diretoria para onde o utilizador pretende exportar os ficheiros que irão conter os resultados do algoritmo.
- *Name of exported files*: Inserir o nome dos ficheiros (um *.txt* e outro *.theta*) que irão conter os resultados do algoritmo. O ficheiro *.txt* destina-se à leitura rápida dos resultados pela parte do utilizador enquanto o ficheiro *.theta* permite que o programa volte a correr os dados caso o utilizador assim o pretenda.
- *SUBMIT*: Pressionar o botão após preencher todas as caixas de texto necessárias para inicializar o algoritmo.
- *STATUS*: Informa o utilizador sobre o estado do programa. Quando este está a ser executado, o Status é alterado para “RUNNING...” e quando o programa termina assume o estado “DONE!”.

No caso de o utilizador cometer algum erro no preenchimento da caixa de diálogo, irá surgir uma janela *pop-up* que dará informação sobre o erro. Após a execução do programa, é gerada uma janela *pop-up* a informar o utilizador que os resultados já foram obtidos e que se encontram sob a forma de ficheiros *.txt* e *.theta* na diretoria selecionada.

Opções de Implementação e Otimização

O projeto é constituído por duas classes principais: a classe “Amostra” e a classe “Mistura”. A classe “Amostra” representa as medições de todos os indivíduos que se querem testar. Esta é definida como uma lista ligada de indivíduos, onde cada indivíduo é uma lista de vetores. Estes vetores são compostos por três parâmetros: o índice do indivíduo, o tempo e o valor da medição. Foi criada uma classe “VetorMed” para representar estes vetores. A amostra foi construída de modo a que as listas de indivíduos estejam ordenadas por índice (para que a posição da lista do indivíduo na amostra corresponda aos valores do indivíduo *i*). Esta implementação foi feita admitindo que a base de dados (ou no nosso caso o ficheiro *.csv* com todas as medições) tinha todos os indivíduos com índices consecutivos e com o primeiro com índice 0.

A classe “Mistura” representa o conjunto de misturas de gaussianas (Theta) que se quer estimar. Esta é composta por um inteiro *M* (corresponde ao número de misturas de gaussianas – número de *thetaj*’s) e uma lista ligada de misturas de gaussianas (constituída por *M* *thetaj*’s). Foi implementada uma classe “*thetaj*” onde é definida a mistura de gaussianas. Esta é construída com base num peso *w*, variância *sigma* e parâmetros da curva de valores médios (*a1*, *a2*, *b1* e *b2*) de cada gaussiana da mistura. Nesta classe foram definidas funções para manipular o conjunto de misturas e também funções que permitem o cálculo de vários parâmetros das misturas com o objetivo de utilizar no algoritmo de Expectation Maximization. Este algoritmo também é definido na classe “Mistura”.

Ao longo do projeto foram encontrados problemas com a comparação de números muito reduzidos. Isto impedia o melhoramento das gaussianas devido a valores de probabilidades muito pequenas (que em java eram aproximados para 0.0) e, portanto, incomparáveis. Isto foi resolvido com a implementação de duas classes adicionais: “*exp*” e “*exp10*”. Estas definem exponenciais de base *e* e de base 10 respetivamente. Nestas classes foram definidas todas as operações entre números exponenciais que foram precisos para o projeto. Com elas, a precisão dos números muito pequenos foi mantida e possível de comparar, sendo que não é calculado o valor da exponencial mas é guardado o expoente. Caso o multiplicador da base seja pequeno, foram implementadas guardas para que este não seja aproximado a 0.0 aumentando o expoente.

A classe “exp” é definida por um a e um b , onde o primeiro é o fator multiplicativo da base e o segundo é o expoente. O a é definido como um “exp10”, ou seja, um número exponencial de base 10 e o b é definido como double. A classe “exp10” é implementada da mesma maneira apenas com o fator multiplicativo definido como double. Desta maneira foi possível aumentar a eficiência do algoritmo e abranger mais casos de execução do mesmo.

Alterações Efetuadas

Da primeira para a segunda entrega, definiu-se a classe “VetorMed” e a classe “Amostra” foi redefinida para ser uma lista de lista de vetores. Portanto, foi necessário adaptar as funções presentes nesta classe de modo a torná-las concordantes com a mudança na implementação (nomeadamente na função *add* e *índice*). Esta alteração permitiu diminuir o tempo de execução do algoritmo e obter resultados para as amostras maiores.

Adicionou-se a função *read* à classe “Amostra”, que permite ler os ficheiros .csv disponibilizados pelo docente, com os elementos que definem os indivíduos.

Na classe Mistura o grupo definiu erradamente a função *prob*. Assim, implementou-se, adicionalmente, a função *probj* (que corresponde à função de densidade de probabilidade da distribuição normal) e a função *prob* passou a corresponder à probabilidade da amostra ser vista pela j -ésima mistura (θ_{tj}) da Mistura total.

Foram adicionadas à classe Mistura as funções que permitem otimizar os parâmetros que definem os θ_{tj} 's: w , σ , a_1 , a_2 , b_1 e b_2 , de acordo com o enunciado disponibilizado pelo docente. Todas as funções encontram-se comentadas no código.

Adicionalmente criaram-se as classes “exp” e “exp10” já explicadas. Para a aplicação gráfica foram implementadas as classes “Login” e “Interface” utilizando o *WindowBuilder*, um Java GUI Designer.

Experimentação

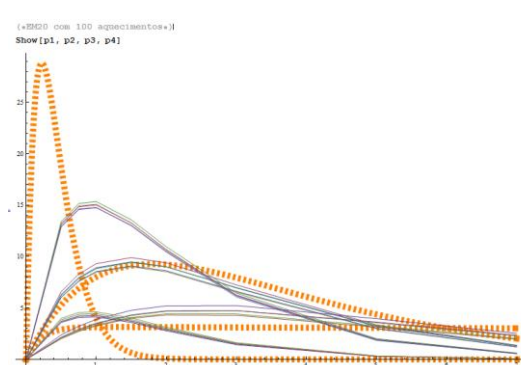


Figura 3 – EM com 20 indivíduos e 100 aquecimentos.

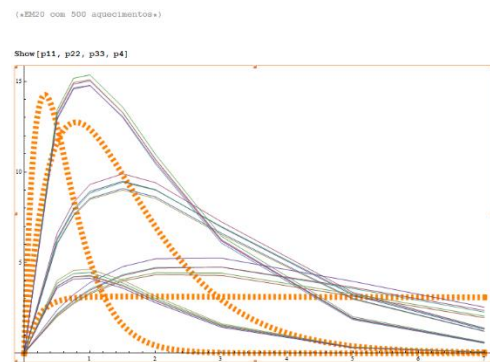


Figura 4 – EM com 20 indivíduos e 500 aquecimentos.

Nestas duas figuras estão apresentadas as curvas dos indivíduos correspondente à amostra de 20 indivíduos e o resultado obtido (a tracejado) no caso em que fazemos 100 aquecimentos (Figura 3) e 500 aquecimentos (Figura 4). É possível desde já constatar que um maior número de aquecimentos permite obter curvas mais próximas das curvas da amostra.

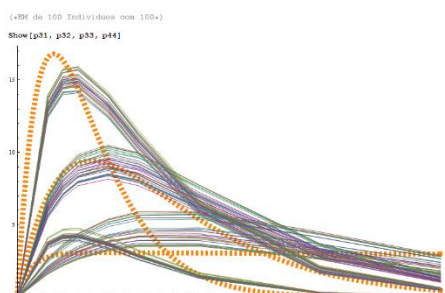


Figura 5 – EM com 100 indivíduos e 100 aquecimentos.

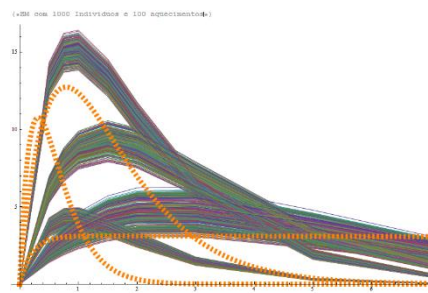


Figura 6 – EM com 1000 indivíduos e 100 aquecimentos.

Nas figuras 5 e 6 apresentamos os resultados para a execução do EM com 100 aquecimentos e com as amostras de 100 e 1000 indivíduos respetivamente. No primeiro caso os pesos das curvas encontram-se bem distribuídos e estas estão razoavelmente próximas das curvas obtidas da amostra. Relativamente à segunda figura uma das curvas não está de acordo com a amostra no entanto o peso é desprezável (10^{-48}).

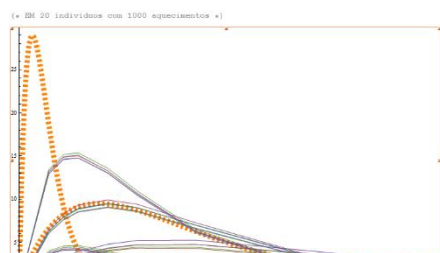


Figura 7 – EM com 20 indivíduos e 1000 aquecimentos.

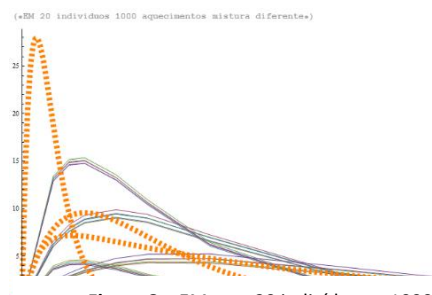


Figura 8 – EM com 20 indivíduos e 1000 aquecimentos e mistura diferente.

Nas figuras 7 e 8 foi possível comparar os resultados obtidos tendo misturas iniciais diferentes. Ambas apresentam uma curva que não se adequa aos indivíduos, no entanto, em ambos os casos estas curvas têm um peso pouco significativo (ordem 10^{-88}). Isto permite perceber que a mistura definida inicialmente também tem influência sobre o resultado obtido.

Apresenta-se agora o resultado obtido colocando na mistura inicial 4 curvas.

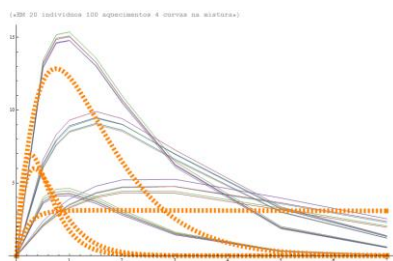


Figura 9 – EM com 20 indivíduos e 100 aquecimentos e mistura inicial com 4 curvas.

Na figura 9 está representado o resultado quando a mistura inicial tem 4 curvas (a tracejado). As duas curvas que não estão enquadradas com a amostra apresentam peso muito

pequeno (na ordem de 10^{-29}) pelo que podem ser ignoradas uma vez que não descrevem uma percentagem significativamente relevante.

Foi ainda possível observar através dos testes que os parâmetros a_1 e a_2 tomam valores muito elevados quando os parâmetros b_1 e b_2 se encontram próximos um do outro. O contrário é verificado quando os parâmetros b_1 e b_2 estão mais afastados. Isto acontece para que as curvas das misturas sejam adequadas às curvas obtidas da amostra.

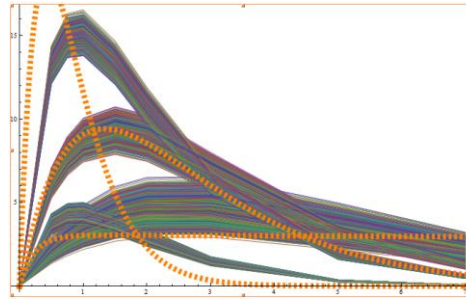


Figura 10 – EM com 3000 indivíduos e 25 aquecimentos.

Por fim apresenta-se o resultado obtido para a amostra de 3000 indivíduos onde é possível observar 3 curvas bem distribuídas. Neste caso não havia nenhuma curva com o peso significativamente menor em relação às restantes.