

Plataforma interativa para análise de padrões de atividade cerebral

Interactive platform for analysis of brain activity patterns

Plataforma interactiva para análisis de los patrones de actividad cerebral

Laura Angélica Tomaz da Silva¹, Davy Charles Vanderweyen², Beatriz Terezinha Borsoi³, Jane Elizabeth Joseph⁴

RESUMO

Descritores: Interface Usuário-Computador; Sistemas de Informação; Aprendizado de Máquina

Objetivo: Implementar interface gráfica e método Kruskal-Wallis, substituindo a ANOVA, para a ferramenta baseada em *Multi-Voxel Pattern Analysis*, desenvolvida em MATLAB, utilizada pelo departamento de neurociência da Universidade Médica da Carolina do Sul, Campus Charleston, para identificar padrões de atividade cerebral em dados obtidos por fMRI. **Métodos:** Desenvolvimento de interface gráfica para a Princeton MPVA e implementação de um método de análise estatística utilizando Kruskal-Wallis para a seleção de características mais adequadas à aprendizagem de máquina. **Resultados:** A quantidade de testes realizada foi pequena, mas os resultados foram promissores. Os dados obtidos do algoritmo implementado sustentam que o procedimento original utilizando *runs* e ANOVA tem menor acurácia. Mudando para Kruskal-Wallis ou para *blocks* a acurácia obtém melhora não estaticamente significativa. Porém, quando usado *blocks* e Kruskal-Wallis o aumento na acurácia é estatisticamente significativo. **Conclusão:** O desempenho do método implementado foi melhor que o existente, especialmente quando utilizando *blocks* e Kruskal-Wallis.

ABSTRACT

Keywords: User-Computer Interface; Information Systems; Machine Learning

Objective: To implement graphical user interface and Kruskal-Wallis, replacing the ANOVA, to the MPVA toolbox using MATLAB. The toolbox was being used by the Medical University of South Carolina to identify brain activity patterns in fMRI data. **Methods:** Development of a graphic user interface for the Princeton MPVA and implementation of a machine learning statistical analysis method to the feature selection. **Results:** Although the number of tests performed was small, the results obtained with the new method were promising. The data obtained from the tests of the implemented algorithm hold that the original procedure using runs and ANOVA has the least accuracy. Changing to Kruskal-Wallis or to blocks the accuracy had a slight improvement, not being statically significant. However, when used blocks and Kruskal-Wallis the increase in accuracy is statistically significant. **Conclusion:** The performance of the implemented method was better than the existing one, especially when using blocks and Kruskal-Wallis.

RESUMEN

Descriptores: Interfaz Usuario-Computador; Sistemas de Información; Aprendizaje Automático

Objetivo: Implementar la interfaz gráfica del usuario y el método Kruskal-Wallis, sustituyendo el ANOVA, técnica basada en MPVA, implementada en MATLAB, utilizada por el departamento de neurociencia de la Universidad de Carolina del Sur, para identificar patrones de actividad cerebral en datos obtenidos por fMRI. **Métodos:** Crear la interfaz gráfica para el MPVA e implementar un método estadístico utilizando Kruskal-Wallis para seleccionar características adecuadas para aprendizaje de máquina. **Resultados:** Para un número pequeño de pruebas realizadas, con el nuevo método, los resultados fueron prometedores. Los datos obtenidos con el algoritmo implementado sustentan que la precisión mejora respecto del procedimiento original que utiliza runs y ANOVA. Cuando se utiliza Kruskal-Wallis o blocks la precisión tiene una ligera mejoría estadísticamente significativa. No obstante, cuando usado blocks y Kruskal-Wallis la precisión tiene una mejoría estadísticamente significativa. **Conclusión:** El rendimiento del método implementado fue mayor que el original, en particular com blocks y Kruskal-Wallis.

¹ Mestranda em Ciência da Computação na Pontifícia Universidade Católica do Rio Grande do Sul – PUCRS, Porto Alegre (RS), Brasil.

² Graduando de Medicina na Universidade de Sherbrooke, (Québec) Canada.

³ Doutorado em Sistemas Digitais pela Escola Politécnica da Universidade de São Paulo, São Paulo (SP), Brasil.

⁴ Professora e Pesquisadora do Instituto de Neurociências, Universidade Médica da Carolina do Sul, Departamento Neurociências, Charleston (SC), EUA.

INTRODUÇÃO

Uma questão essencial em neurociência cognitiva é a que trata da representação da informação⁽¹⁾: qual informação é representada em diferentes estruturas cerebrais; como a informação é representada; e como a informação é transformada nos diferentes estágios de processamento. O desenvolvimento de novas tecnologias e o aumento crescente da capacidade de computação têm tornado possível obter informações sobre os padrões de atividade cerebral e analisá-los em tempo hábil. Os adventos recentes de técnicas de imagens funcionais permitem visualizar quais partes do cérebro estão ativas durante determinadas circunstâncias. A Ressonância Magnética Funcional, do inglês *functional Magnetic Resonance Imaging* (fMRI), mede a taxa de hemoglobina oxigenada para desoxigenada no sangue em relação a uma linha base de controle, sendo utilizada como indicador de atividade cerebral⁽²⁾.

A *Functional Magnetic Resonance Imaging* obtém dados da atividade cerebral, por meio de ressonância magnética, capturados a cada 2 ou 3 segundos. O tempo entre as aquisições de dados é chamado de Tempo de Repetição (TR). Cada mapeamento consiste em uma matriz de três dimensões chamada de volume e representa uma imagem instantânea do estado de atividade cerebral em um momento específico de tempo. Cada plano da matriz de volumes é chamado de pedaço (*slice*) e cada elemento representa um *pixel* na imagem daquele pedaço.

Quando esses pedaços são agrupados, os *pixels* em duas dimensões eles se tornam tridimensionais já que o espaço entre esses pedaços precisa ser preenchido. Esses *pixels* cúbicos são chamados *voxels*. A próxima sessão de digitalização traz informações sobre o estado do cérebro de um TR posterior e assim sucessivamente. Assim, uma sessão de ressonância (aquisição de dados) de 5 minutos com um TR de 2 segundos é como um filme das mudanças de ativação do cérebro de 5 minutos com taxa de amostragem de 0,5 imagens por segundo.

A grande quantidade (geralmente vários *gigabytes*) e o alto nível de ruído inerente aos dados obtidos com fMRI tornam a mineração desses dados, para obter informações sobre processos cognitivos, um grande desafio⁽¹⁾. Vinte minutos de fMRI com um indivíduo produzem uma série de imagens do cérebro cada uma contendo aproximadamente 15000 *voxels*, gerando dezenas de milhões de dados para observação⁽²⁾.

Estudos têm demonstrado que fMRI contém informações que permitem o discernimento entre estados mentais e condições neurológicas. Descobrir e classificar essas diferenças pode permitir “ler mentes” e construir aplicativos computacionais que possibilitem diagnosticá-las automaticamente de forma a auxiliar médicos na tomada de decisão. MVPA é um desses métodos e se refere à ideia de aplicar métodos multivariados a dados de fMRI, permitindo analisar diversos *voxels* simultaneamente.

A *functional Magnetic Resonance Imaging* melhora significativamente a habilidade humana de observar correlações de atividades cerebrais com alta resolução

espacial (alguns milímetros) em todo o cérebro⁽²⁾. Trata-se de uma ferramenta eficiente para estudo não invasivo da atividade cerebral em resposta a diferentes estímulos⁽³⁾.

Considerando o volume de dados gerados por uma fMRI, seria extremamente árduo, por meio de análise manual, encontrar padrões de atividade cerebral, mas técnicas computacionais podem ser utilizadas para identificar e examinar esses dados. O método de *Machine Learning*, aprendizagem de máquina, pode ser utilizada para prever quais padrões de atividade cerebral estão associados a determinados pensamentos. Encontrar a relação entre os padrões em pequena escala de atividade cerebral e os estados mentais como emoções e percepções, é uma das questões de pesquisa em neurociência. A técnica *Multi Voxel Pattern Analysis* (MVPA) tem sido bem-sucedida para decifrar informações a partir de padrões de ativação cerebral.

Trata-se de uma técnica de *Machine Learning* que analisa e classifica padrões de ativação cerebral. *Machine Learning* está relacionada com a construção de algoritmos computacionais para melhorar o desempenho por meio da experiência ou de exemplos (conjuntos de treinamento)⁽⁴⁾. MVPA envolve buscar padrões espaciais altamente reproduzíveis para atividade que se diferencia por meio de condições experimentais⁽⁵⁾. Em MVPA um classificador supervisionado tenta capturar os relacionamentos entre padrões espaciais de atividade fMRI e condições experimentais⁽⁶⁾.

Na Universidade Médica da Carolina do Sul, Campus Charleston, os profissionais da divisão de neuroimagem do departamento em neurociência utilizam uma ferramenta desenvolvida pela Universidade de Princeton em MATLAB, baseada em MPVA e *Analysis of Variance* (ANOVA). O código da Princeton MVPA está disponível gratuitamente, mas requer conhecimento em computação para usá-lo e customizá-lo e a interação com a ferramenta é realizada por meio de linhas de comando. É necessário digitar as instruções que devem ser executadas. A usabilidade de sistemas é um fator de alta relevância para o seu adequado uso, especialmente por profissionais que não são da área de computação. Estudiosos realizaram avaliação de usabilidade em sistemas médicos e confirmam a importância do atendimento de padrões de usabilidade⁽⁷⁾.

Como resultado deste trabalho uma interface gráfica de interação foi desenvolvida para a ferramenta Princeton MVPA e um novo método para a seleção de características mais adequadas à aprendizagem de máquina, o Kruskal-Wallis, em substituição ao ANOVA, foi implementado. Como resultado, além de maior acurácia de dados e redução de falsos positivos, o aplicativo desenvolvido tem o propósito de tornar MVPA mais acessível para pessoas que não possuem experiência com MATLAB ou conhecimento em *Machine Learning*. A ferramenta permite que o usuário possa selecionar interativamente a condição regressora e escolher entre os algoritmos de *Machine Learning* implementados.

A ferramenta desenvolvida pode ser utilizada para, por exemplo, diagnóstico de pacientes com doença de Alzheimer. Para isso seria utilizado um conjunto de dados

do estado de repouso de pacientes com Alzheimer e pacientes saudáveis (o grupo de controle)⁽⁸⁾. Autores descrevem uma aplicação que, utilizando técnicas de mineração de dados, que auxilia no diagnóstico de pacientes com suspeita clínica de Alzheimer⁽⁹⁾. Eles utilizam métodos de classificação *naïve bayes*, redes *bayesianas* e árvores de decisão para minerar os dados. Neste trabalho é utilizada a técnica *Machine Learning* associada ao método estatístico Kruskal-Wallis para a Princeton MPVA.

MÉTODOS

Os principais passos para a realização do trabalho foram:

a) Implementar de uma interface gráfica para a ferramenta Princeton MVPA.

b) Criar um repositório para armazenar os filtros predefinidos para que o usuário possa selecionar os filtros desejados para análise e criar um filtro espacial que permite que os *voxels* da área desejada façam parte da análise.

c) Permitir que o usuário defina especificadores de condições (condição regressora) interativamente enquanto visualiza os dados em vez de criar uma matriz de regressão. Condição regressora é um arquivo que especifica quais condições experimentais estão ativas em determinado momento.

d) Implementar o seletor de características: o método utilizado na ferramenta Princeton MVPA é a ANOVA, que tem sido criticado por predispor os resultados de aprendizagem de máquina favorecendo maior acurácia em uma forma chamada de “dupla imersão”. Portanto, outro método foi implementado na ferramenta, o Kruskal-Wallis, que é o equivalente da ANOVA como teste não-paramétrico, mas não supõe que todos os resultados estejam dispostos em uma distribuição Gaussiana.

O teste Kruskal-Wallis usa classificação de dados por

postos (*rank*s) das observações em cada grupo. O método de classificação atribui a cada valor observado, um posto, sempre atribuindo ao menor valor o menor posto e ao maior valor o maior posto, retirando, então, o índice numérico dessa ordenação⁽¹⁰⁾.

O teste paramétrico equivalente a Kruskal-Wallis é a ANOVA, em que a análise de variância dos testes depende da hipótese de que todas as populações em questão são independentes e normalmente distribuídas, enquanto que Kruskal-Wallis não apresenta nenhuma restrição quanto a essa comparação. Outra suposição feita com o uso da ANOVA é que a variância entre os grupos de dados é igual. Kruskal-Wallis não supõe que os dados estão distribuídos normalmente, mas que diferentes grupos têm a mesma distribuição e grupos com diferentes desvios padrões possuem diferentes distribuições.

O teste Kruskal-Wallis assume que todas as amostras de dados são provenientes de populações com a mesma distribuição contínua, separadas de possíveis locais diferentes devido aos efeitos de grupos, e que todas as observações são independentes entre si⁽¹⁰⁾. Em contraste, ANOVA substitui a primeira suposição com uma suposição mais forte de que a população está normalmente distribuída⁽¹¹⁾.

Quando ANOVA é utilizada supõe-se que os dados são paramétricos, o que significa que, os valores de ativação dos *voxels* estão dentro de uma distribuição Gaussiana ou normal. No entanto, nem sempre é o caso. Utilizar ANOVA nesses dados não-paramétricos pode levar a falsos resultados. Justificando-se, assim, a implementação do teste com Kruskal-Wallis.

O processo de uso de MPVA ocorreu em duas etapas: treinamento e testes. Na etapa de treinamento, MVPA primeiramente “aprende” os diferentes padrões de ativações cerebrais (entrada) associadas às condições experimentais (alvo), então esse conhecimento é usado para prever a categoria de um novo conjunto de dados

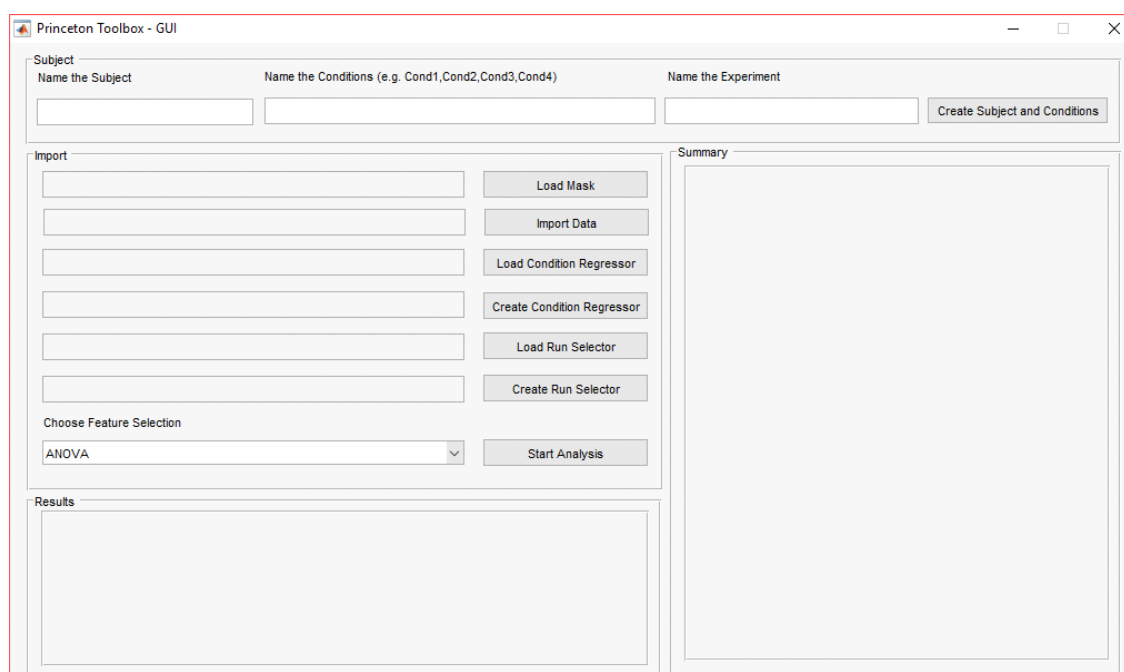


Figura 1 - Tela principal do aplicativo desenvolvido

na etapa de testes. O resultado é uma pontuação com o nível de acurácia calculado para estimar o desempenho do classificador. Posteriormente, esses dados são separados em um conjunto de treinamento e um conjunto de testes. Esses dados são inseridos no software MVPA com rótulos correspondentes às tarefas realizadas.

A Figura 1 apresenta a tela principal da ferramenta desenvolvida em MATLAB. A interface é em inglês para manter compatibilidade com a Princeton MPVA e por ser essa uma linguagem padrão desse tipo de aplicativo.

O *layout* do aplicativo é simples e dividido em quatro partes para auxiliar o usuário na sua utilização. Na parte superior está a criação do indivíduo, ou seja, a parte inicial da utilização da ferramenta. Logo abaixo está a localização para a entrada de dados e a escolha do método de *feature selection* para iniciar a análise dos dados. Abaixo da área de entrada de dados está a apresentação do resultado da análise realizada, com a respectiva acurácia. Por último, no lado direito da tela há um quadro para apresentar o resumo das entradas de dados ou objetos criados para auxiliar na análise.

Na Figura 2 é apresentada a entrada de dados na qual o usuário buscará os diretórios que possuem armazenadas os dados do indivíduo. Ele carregará informações de máscara, os dados obtidos da fMRI, o Condition Regressor e o Run Selector. É possível, ainda, criar o Condition Regressor e o Run Selector. Para isso devem ser especificados, respectivamente, o número de condições ou número de *runs* e o número de *timepoints*. Após isso, será criada outra tela para que o usuário informe quando determinada condição ou *run* estava ativa e salvar essa matriz ou vetor criados. É importante ressaltar que o número de *timepoints* informado deve possuir a mesma dimensão de colunas da matriz de dados provindos da fMRI. Após carregar todos os dados na tela, o usuário deverá escolher entre os métodos de *feature selection* disponíveis e iniciar a análise dos dados.

Ao clicar no botão de “*Start Analysis*”, o sistema iniciará a análise de dados para então retornar a acurácia dos resultados. É recuperado qual foi o método selecionado pelo usuário e é iniciada a análise fazendo o

zscore do *pattern*, que são os dados vindos da fMRI, de acordo com o número de *runs*. Após isso, são criados os índices para o *n-minus-one* que define a quantidade de iterações de acordo com o número de *runs* para realizar o *cross-validation*. No caso estudado, *runs* são substituídos por *blocks* visando implementar melhor resultado no desempenho do algoritmo.

Antes de realizar a classificação é realizada análise estatística. No caso de uso de *Machine Learning* essa análise é realizada por *feature selection* que é responsável por verificar quais *voxels* são mais relevantes. O método estatístico determina qual a probabilidade da atividade de um determinado *voxel* variar significativamente entre as condições em todo o experimento⁽¹²⁾.

O passo seguinte é verificar qual o método de *feature selection* foi selecionado e chamar a função responsável por tal, a qual possui os parâmetros de informações de indivíduo, *pattern* após o *zscore*, *condition regressor* e os índices de iterações criados.

Com esses dados é possível começar a classificação de dados de acordo com a *n-minus-one cross-validation*. É necessário adicionar alguns parâmetros básicos para o classificador de *backpropagation*, sendo este um método comum de treinamento de redes neurais na qual o resultado inicial do sistema é comparado com o resultado desejado e o sistema é ajustado até que a diferença entre os dois seja minimizada. Os dados são separados em teste e treinamento. É chamada a função de *cross-validation* que classificará os dados após o *zscore* ter sido realizado de acordo com as condições e que chamará o *backpropagation* diversas vezes, uma por iteração de *n-minus-one*, iterando pelo número de índices criados e a *mask* (máscara) definida que auxiliará na classificação informando quais características serão alimentadas ao classificador.

Após a classificação ter sido finalizada, o resultado total de acurácia do desempenho do classificador é armazenado em uma variável e apresentado em “*Results*”. Tendo isso sido feito, é iniciada a remoção de todos os objetos auxiliares criados para essa análise, de maneira a não interferir em análises subsequentes.

The screenshot shows a window titled "Import" with the following elements:

- Five text input fields containing file paths:
 - C:\Program Files\MATLAB\R2014b\princeton\nimh2.1b\mask\
 - C:\Program Files\MATLAB\R2014b\princeton\nimh2.1b\910\
 - C:\Program Files\MATLAB\R2014b\princeton\nimh2.1b\910\
 - (Empty)
 - C:\Program Files\MATLAB\R2014b\princeton\nimh2.1b\910\
- Five buttons on the right side:
 - Load Mask
 - Import Data
 - Load Condition Regressor
 - Create Condition Regressor
 - Load Run Selector (highlighted with a blue dashed border)
 - Create Run Selector
- A dropdown menu labeled "Choose Feature Selection" with "ANOVA" selected.
- A "Start Analysis" button at the bottom right, circled in red.

Figura 2 - Importação/entrada de dados

Na Figura 3, é possível verificar a acurácia do resultado do desempenho do algoritmo de acordo com os dados na tela e o método de *feature selection* escolhidos.

Por último, a Figura 4 apresenta de maneira resumida todas as informações de entradas na tela. Desde os dados escolhidos pelo usuário, assim como os objetos criados para auxiliar na análise.

RESULTADOS

Alguns experimentos realizados com a ajuda de fMRI não possuem um número grande de *runs*. Para esses experimentos, a MVPA, ferramenta de Princeton, se torna inadequada. Essa ferramenta foi projetada para fazer uma *cross correlation* em um conjunto de dados com no mínimo 10 *runs*.

Quanto menos *runs* um conjunto de dados possui menos rodadas de *cross correlation* serão realizadas, o que tornará os resultados menos confiáveis. Uma forma de aumentar o número de vezes que a *cross correlation* ocorrerá é mudando a maneira como é feita a separação dos dados pelo delimitador de *regressors*, portanto, de *runs* para *blocks*. Fazendo isso será efetivamente multiplicado o número de *cross correlations* do número de *runs* para o número de *runs* vezes o número de *blocks* em cada *run*. Por exemplo, considerando um experimento com 4 *runs* e cada *run* contém 20 *blocks*, usando esse método será possível ter

80 rodadas em vez de apenas 4, já que $4 \times 20 = 80$.

Além disso, alguns experimentos possuem tipos de estímulos diferentes dentro da mesma *run*. Portanto, para que isso seja possível os dados foram separados de acordo com o *block* ao qual pertence, em vez de ter uma condição associada com cada *run*. Tendo vários *blocks* rotulados com a mesma indicação é possível também ignorar o erro de “rótulo/etiqueta único (a)”, que ocorre quando existe um número igual ou maior de *runs* que o número de condições. Nesses casos, o *software* reterá a única amostra de uma condição em particular, será incapaz de treinar-se na condição e tentará classificá-la como outra.

A estratégia estabelecida para demonstrar as mudanças na ferramenta, foi realizar testes em conjuntos de dados provindos de experimentos com poucas condições⁽¹³⁾. Os indivíduos do experimento tinham entre 18 e 32 anos (média de 23 anos), todos destros, não fumantes e sem relatos de problemas psiquiátricos, neurológicos ou problemas médicos significativos; sem deficiência de aprendizagem e visão normal ou corrigida para normal. Em cada experimento quatro condições foram observadas, sendo elas: esquemáticos de rostos de pessoas, imagens reais de rostos de pessoas (fotos de anuários), peixes e guitarras. Cada *run* era composta de 16 blocos de tarefas intercalados com 17 blocos de *rest* (repouso). Em cada bloco de tarefa, com duração de 21 segundos, os indivíduos viam dois estímulos da categoria de interesse.

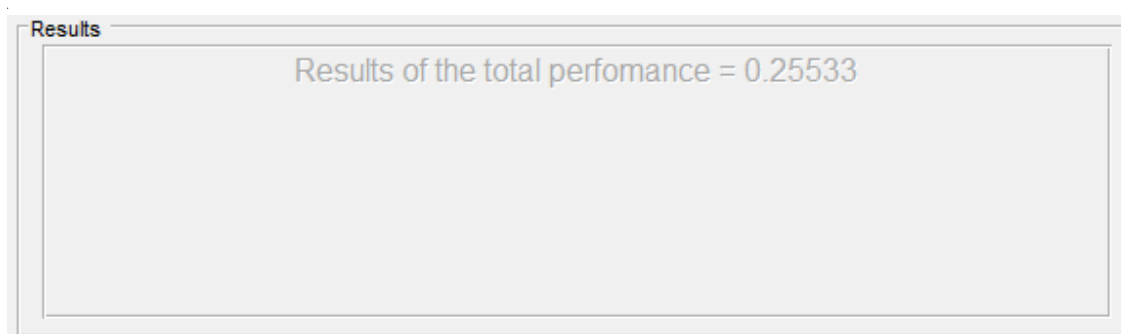


Figura 3 - Resultado do desempenho da análise de dados

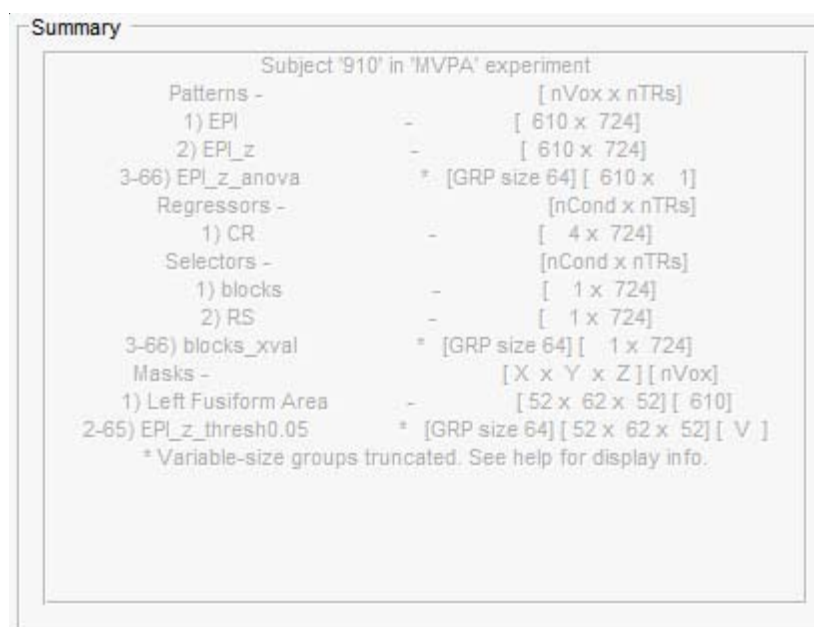


Figura 4 - Resumo dos dados

Cinco dos pares de um bloco retratavam dois objetos diferentes (ou faces) e dois dos outros pares eram compostos de estímulos idênticos. Os indivíduos deveriam pressionar um botão com o dedo indicador direito para indicar que dois estímulos eram idênticos ou usar o dedo médio para indicar que os dois estímulos eram diferentes. Existiam 4 níveis diferentes de similaridade manipulados entre os blocos de um *run*, sendo cada nível de similaridade repetido 4 vezes, portanto, produzindo 16 blocos de tarefas por *run*. Já os blocos de *rest* duravam 12 segundos e os indivíduos olhavam para uma cruz localmente centralizada e deveriam não pensar (pensar em nada). Cada *run* durou 9,05min com 181 volumes coletados por *run* (TR=3seg; TE=30ms; 3,5mm espessura do corte; 3.5mm² de resolução do plano; ângulo de inclinação=81 graus; Matriz 64x64; Trio 3T scanner

Siemen). Portanto, o número de *runs* desse experimento era de apenas quatro tornando-o inadequado para a análise com a ferramenta Princeton MVPA.

Os resultados obtidos dos testes realizados com a ferramenta desenvolvida podem ser observados nas figuras a seguir, sendo que cada uma delas corresponde a diferentes combinações de classificação. Na Figura 5 está uma combinação de classificação ANOVA com *blocks*, com resultado de desempenho de 0,27086.

Na Figura 6 é apresentado o resultado total do desempenho de 0,37180 que foi obtido por meio do uso de Kruskal-Wallis e *blocks*.

Nas Figuras 7 e 8 são apresentados os resultados obtidos com o uso de *runs* como estabelecido originalmente na ferramenta e com os métodos estatísticos ANOVA e Kruskal-Wallis. É possível observar que o desempenho total

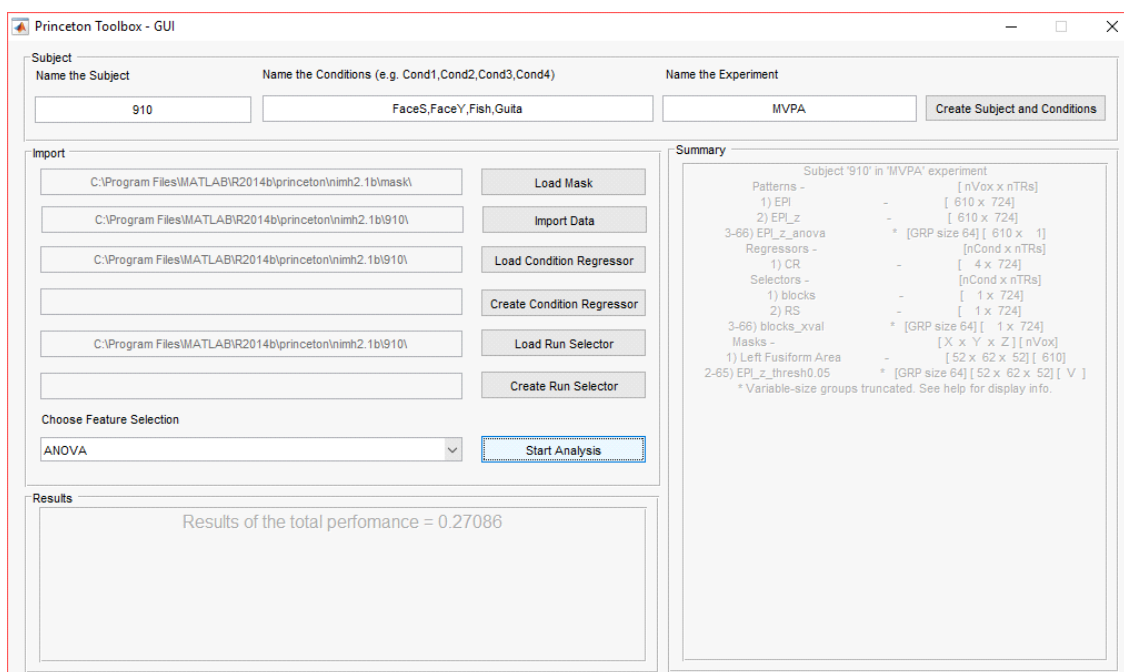


Figura 5 - ANOVA + *blocks*

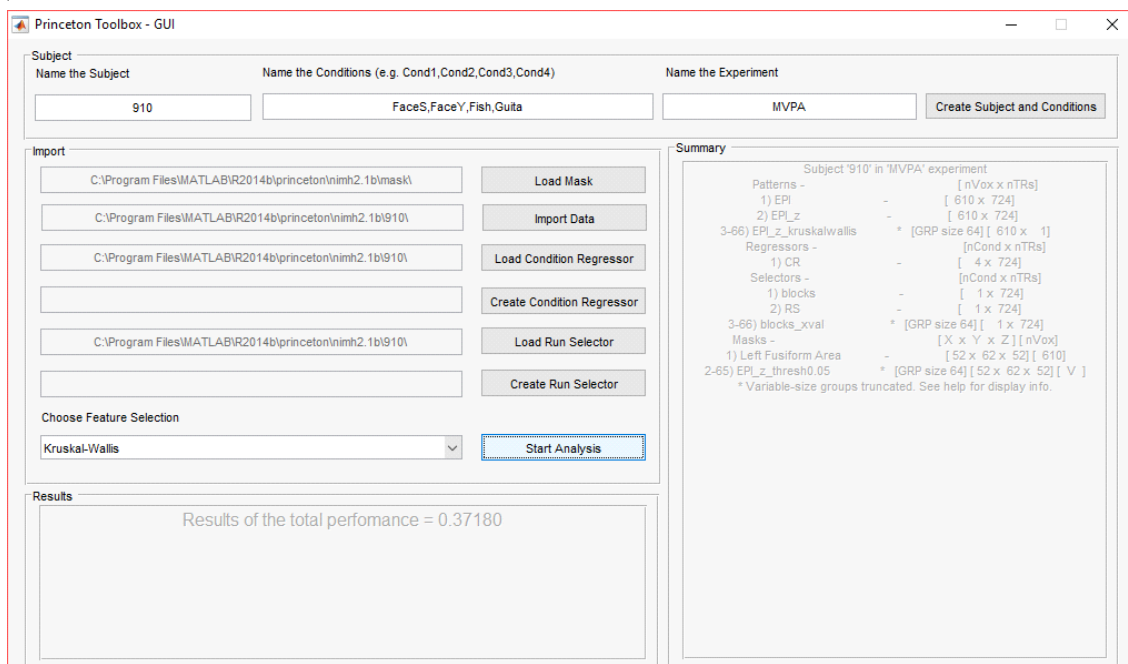


Figura 6 - Kruskal-Wallis + *blocks*

obteve resultados menores que aqueles apresentados com o uso dos métodos estatísticos e *blocks*.

O número de conjuntos de dados dos experimentos foram seis e o resultado do desempenho obtido com cada

um deles é apresentado na Tabela 1. Os valores dessa Tabela estão entre 0 e 1, expressando porcentagem da acurácia dos dados.

Os dados da Tabela 1 indicam que o procedimento

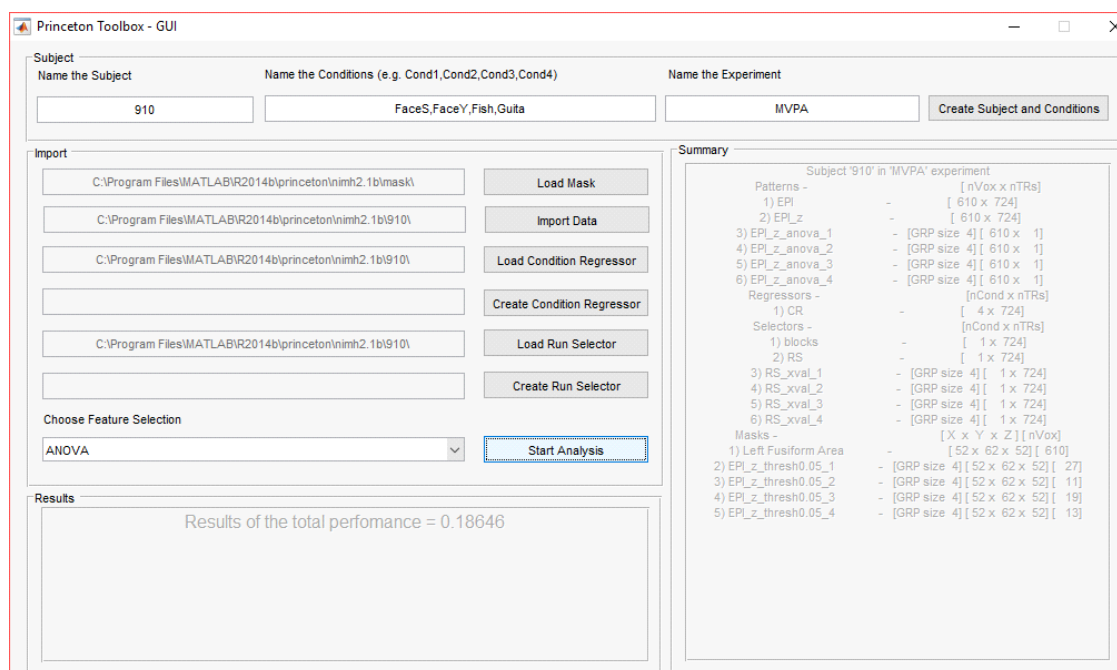


Figura 7 - ANOVA + runs

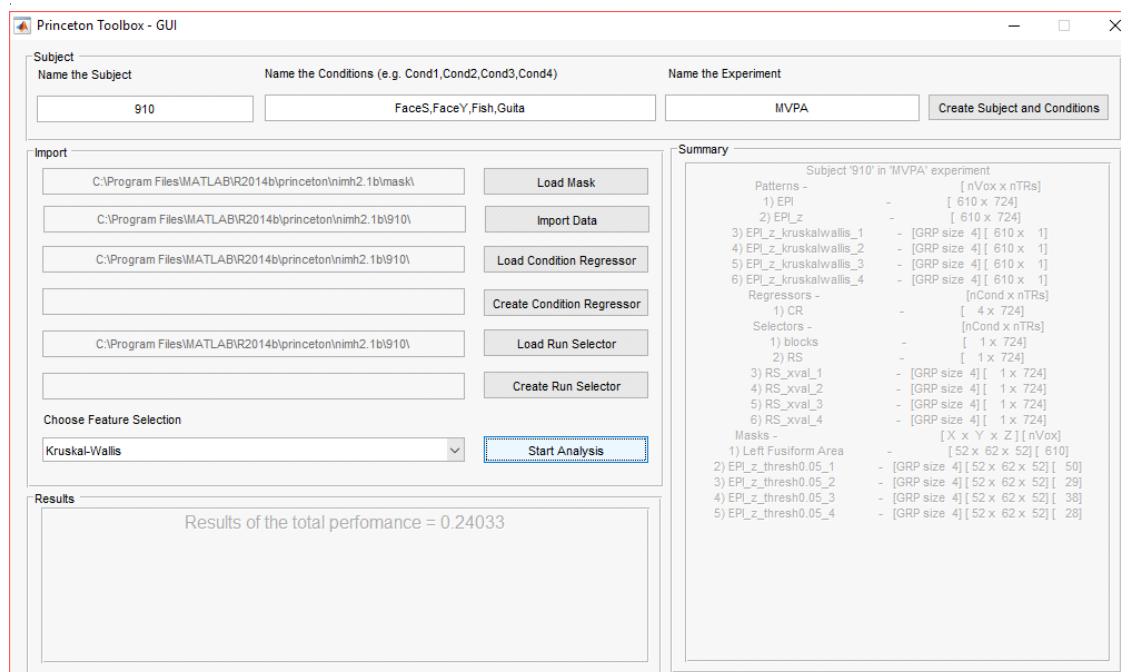


Figura 8 - Kruskal-Wallis + runs

	Métodos			
	ANOVA+runs	Kruskal-Wallis+runs	ANOVA+blocks	Kruskal-Wallis+blocks
Indivíduo1-910	0,18646	0,24033	0,27086	0,37180
Indivíduo2-1010	0,12431	0,13536	0,22828	0,28042
Indivíduo3-1025	0,11188	0,26934	0,13108	0,23317
Indivíduo4-1049	0,04558	0,07044	0,15829	0,21100
Indivíduo5-1062	0,18232	0,09254	0,07090	0,13485
Indivíduo6-1087	0,15608	0,06630	0,19105	0,20839
Médias	13,4438	14,5718	17,5076	23,9993

Tabela 1 - Desempenho dos métodos implementados.

original utilizando *rums* e ANOVA apresenta a menor acurácia. Mudando para Kruskal-Wallis ou para *blocks* a acurácia teve leve melhora, mas não sendo estaticamente significativa. Porém, quando usado *blocks* e Kruskal-Wallis o aumento na acurácia é estatisticamente significativo. Ainda que testado em poucos indivíduos, há bons fundamentos para justificar que a abordagem utilizada pode produzir uma acurácia melhor e que deveria ser mais amplamente testada e avaliada.

CONCLUSÃO

Há uma grande quantidade de informações nos padrões de ativação cerebral e é por isso que técnicas como MVPA, que permitem estudar a dinâmica de ativação do cérebro, têm se tornado valiosas e têm o potencial de auxiliar no entendimento em um nível mais profundo de como o cérebro processa a informação.

Considerando as possibilidades de análise de dados obtidos de atividade cerebral e a diversidade de profissionais envolvidos nessas análises e que nem sempre possuem conhecimento aprofundado para lidar com ferramentas computacionais complexas, o desenvolvimento de uma interface gráfica de fácil uso se apresentou necessária para

incentivar o uso da ferramenta MVPA da Universidade de Princeton. Além disso, tornar essa ferramenta gratuita permitirá que um maior número de pessoas possa utilizá-la para realizar análises sem necessidade de implementar códigos, agilizando essa análise sem necessidade de conhecimento em ciência da computação ou *Machine Learning*. A ferramenta desenvolvida permitirá experimentação com MVPA por meio de uma interface de interação simples, sem requerer conhecimento em programação para que possa ser utilizada.

Por fim, a realização desse trabalho permitiu a comparação dos resultados entre dois métodos estatísticos para as análises dos padrões de atividade cerebral. Ainda que os testes realizados tenham sido feitos em um pequeno grupo de dados, foi possível verificar um aumento significativo na acurácia dos resultados quando combinado o uso de Kruskal-Wallis com *blocks* em relação ao uso de ANOVA com *rums*, como planejado originalmente pela ferramenta MVPA. Contudo, ressalta-se que mais testes devem ser realizados para verificar a real significância da acurácia de resultados. Com a nova abordagem proposta neste trabalho é possível que experimentos que utilizam grupos menores de indivíduos possam obter maior acurácia de resultados.

REFERÊNCIAS

1. Norman KA, Polyn SM, Detre GJ, Haxby JV. Beyond mind-reading: multi-voxel pattern analysis of fMRI data. *Trends Cognit Scienc*. 2006;10(9):424-30.
2. Mitchell T, Hutchinson R, Niculescu RS, Pereira F, Wang X, Just M, Newman S. Learning to decode cognitive states from brain images. *Machine Learning*. 2004; 57(1-2):145-75.
3. Song X, Iordanescu G, Wyrwicz AM. One-class machine learning for brain activation detection. *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*; 2007 Jun 17-22; Minneapolis, MN, USA.
4. Vidhate D, Kulkarni P. Cooperative machine learning with information fusion for dynamic decision making in diagnostic applications. *Proceedings of the International Conference on Advances in Mobile Network, Communication and Its Applications*. 2012 Ago 1-2; Bangalore, India.
5. Mahmoudi A, Takerkart S, Regragui F, Boussaoud D, Brovelli A. Multivoxel pattern analysis for fMRI data: a review. *Comput Mathematic Methods Med*. 2014; 2012:1-14.
6. Davatzikos C, Ruparel K, Fan, Y, Shen Dg, Acharyya M, loughhead JW, Gur RC, Langleben DD. Classifying spatial patterns of brain activity with machine learning methods: application to lie detection. *Neuroimage*. 2005;28(3):663-8.
7. Yamamoto TTI, Bandiera-Paiva P, Ito M. Avaliação da usabilidade de interface gráfica de dois sistemas de gestão hospitalar. *J. Health Inform*. 2015;7(2):37-41.
8. Wang L, Zangd Y, Hed Y, Liang M, Zhang X, Tian L, Wu T, Jiang T, Li K. Changes in hippocampal connectivity in the early stages of Alzheimer's disease: evidence from resting state fMRI. *Neuroimage*. 2006;31(2):496-504.
9. Moreira LB, Namen AA. Sistema preditivo para a doença de Alzheimer na triagem clínica. *J. Health Inform*. 2016;8(3):87-94.
10. Mcdonald JH. Kruskal-Wallis test. *Handbook of biological statistics*. 3a ed. 2014: 157-64. [cited 2017 Jan 28]. Available from: <http://www.biostathandbook.com/kruskalwallis.html>
11. Kruskal-Wallis. *MathWorks*. [cited 2016 nov 28]. Available from: <http://www.mathworks.com/help/stats/kruskalwallis.html>PrequestedDomain=fr.mathworks.com&requestedDomain=www.mathworks.com
12. Benharrosh M, Bouvier S, Carrol MK, Chigirey D, Detre G, Engell A, et al. Princeton multi-voxel pattern analysis (MVPA) toolbox. [cited 2016 nov 06]. Available from: <https://github.com/PrincetonUniversity/princeton-mvpa-toolbox/wiki/TutorialIntro>
13. Joseph JE. Neuroimaging of shape similarity in object recognition. Grant #: 1R01MH063817. University of Kentucky. 2003 - 2007.