UNIVERSIDADE FEDERAL DA BAHIA

INSTITUTO MULTIDISCIPLINAR EM SAÚDE

PROGRAMA DE PÓS-GRADUAÇÃO EM BIOCIÊNCIAS

DANIEL SILVA FERRAZ

DESENVOLVIMENTO DE PREDITOR DE EPÍTOPO PARA MHC-I LIGANTE DE HLA-A01*01 POR *DEEP LEARNING*

Vitória da Conquista, BA
8 de Dezembro de 2018

DANIEL SILVA FERRAZ

DESENVOLVIMENTO DE PREDITOR DE EPÍTOPO PARA MHC-I LIGANTE DE HLA-A01*01 POR *DEEP LEARNING*

Dissertação apresentada ao Programa de Pós-Graduação em Biociências, Universidade Federal da Bahia, como requisito para obtenção do título de Mestre em Biociências.

Orientador: Prof. Dr. Sávio Torres de Farias
Universidade Federal da Paraíba, João Pessoa, PB

Co-orientador: Prof. Dr. Leandro Martins Freitas
Universidade Federal da Bahia, Vitória da Conquista, BA

Vitória da Conquista, BA
8 de Dezembro de 2018

DANIEL SILVA FERRAZ

DESENVOLVIMENTO DE PREDITOR DE EPÍTOPO PARA MHC-I LIGANTE DE HLA-A01*01 POR *DEEP LEARNING*

Esta dissertação foi julgada adequada à obtenção do grau de Mestre em Biociências e aprovada em sua forma final pelo Programa de Pós-graduação em Biociências, Universidade Federal da Bahia.
Vitória da Conquista – BA, data.

Prof. Dr. Orientador Sávio Torres de Farias
Universidade Federal da Paraíba

Prof. Dr. Co - Orientador Leandro Martins de Freitas
Universidade Federal da Bahia

Prof. Dr.Lucas Miranda Marques
Universidade Federal da Bahia

Prof. Dr. Carlos Priminho Pirovani
Universidade Estadual de Santa Cruz

Dedico essa dissertação aos meus pais, razão de toda minha jornada de vida. Dedico à minha mãe, Lúcia, que se esforçou todo esse tempo para tornar possível meus sonhos. Você sempre será exemplo de garra, determinação e luta. Dedico ao meu pai que transmitiu com muito afeto seu companheirismo e apoio nos momentos em que precisei. Todos esses anos me dediquei a colocar em prática todos os conselhos dos meus pais, aprendi a valorizar, reconhecer, apreciar, dedicar, escutar, silenciar e estudar. Dedico, esse estudo à minha irmã que sempre me motivou e serviu de exemplo de superação e tranquilidade mediante períodos turbulentos. Dedico a todos que se esforção para produzir conhecimento científico apesar da falta de incentivos. Dedico ao Senhor que me guia e me proteje em todos os meus passos.

AGRADECIMENTOS

Aos professores Leandro Martins de Freitas e Sávio Torres de Farias, pelas orientações e ensinamentos em torno da Bioinformática. Também agradeço toda a compreensão durante esse tempo. A experiência, empenho e disposição de vocês, mesmo mediante diversas barreiras, foi essencial para que este trabalho se tornasse viável. À banca por ter aceitado o convite para a avaliação desse trabalho. Aos professores do programa Biociências por ter possibilitado a expansão do meu conhecimento científico. Aos meus queridos colegas de mestrado, por ter partilhado companheirismo e apoio durante esta jornada. A Cnpq, pelo apoio financeiro. Aos amigos Rodrigo, Lucas e Ana, por ter me demonstrado o real conceito de amizade duradoura, transpassando o tempo e a distância. Aos meus familiares que me deram apoio e sempre confiaram em meu potencial. Aos meus pais Omar Ferraz e Lúcia Maria, que são meu alicerce que me deram palavras sábias e acolheram em momento de receio, além de carinho e amor ao qual foi crucial para que motivações e razões florescessem ao longo dos últimos anos. À família Yelt, pelas palavras motivacionais e boas energias. À minha irmã Daiane e meu cunhado Danilo, pelo apoio, preocupação e por ter trago a minha adorada afilhada a este mundo. O sorriso de Laura ajuda a relembrar o que há de mais tênue e puro. Ao Senhor por jamais ter soltado a minha mão.

“Transmita o que aprendeu. Força, mestria. Mas fraqueza, insensatez, fracasso também. O maior professor, o fracasso é. Nós somos o que eles crescem além. Esse é o verdadeiro fardo de todos os mestres.”
Mestre Yoda

FERRAZ, Daniel Silva. Desenvolvimento de preditor de epítopo para MHC-I ligante de hla-a01*01 por *deep learning*. 68xx f. il. 2018. Dissertação (Mestrado) – Instituto Multidisciplinar de Saúde, Universidade Federal da Bahia, Vitória da Conquista, 2018.

RESUMO

Peptídeos derivados de antígenos endógenos podem ligar-se a moléculas de MHC-I. Aqueles que se ligam com alta afinidade podem invocar uma resposta imune $CD8^+$, resultando na destruição de células infectadas. Muito trabalho em imunoinformática envolveu a previsão algorítmica da afinidade de ligação peptídica a vários alelos de MHC-I. Uma série de ferramentas para predição de ligação de MHC-I foram desenvolvidas, muitas das quais estão disponíveis na web. No entanto, muitos métodos baseados em aprendizado de máquina são geralmente menos sensíveis no reconhecimento de interações agrupadas localmente, que podem estabilizar sinergicamente a ligação peptídica. Aqui nós propomos o preditor EPI, uma rede neural artificial baseada no alelo HLA-A01*01. Nossa ANN foi construída usando 3902 epítopos para o HLA-A01*01 e nove propriedades físico-químicas. Em seguida, as propriedades foram reduzidas para quatro PCA e usadas como características para cada aminoácido no treinamento e nos peptídeos de teste da rede neural. Nós treinamos/testamos o modelo usando 70/30% dos epítopos recuperados do banco de dados, respectivamente. Nosso modelo atingiu um bom nível de previsão com uma precisão de 77,09% e uma pontuação F1 de 0,77 no conjunto de testes. Além disso, vários testes de hiper-parâmetros foram realizados para identificar a rede neural multicamada mais adequada. Além do protocolo de teste de validação, procuramos comparar a taxa de precisão realizada pelo nosso preditor em comparação com várias outras abordagens preditivas disponíveis *online*. O *benchmark* demonstrou que o EPI exibia um excelente desempenho em contraste com alguns preditores confiáveis do MHC-I. O EPI atingiu 0,53 MCC (Matthew Correlation Coeficiency), indicando capacidade preditiva consistente. As estratégias para construir o EPI e desempenho em comparação com outros preditores são discutidas.

Palavras-chave: predição de epítopos, rede neural, aprendizado de máquinas.

FERRAZ, Daniel Silva. MHC-I Epitope predictor for HLA-A01*01 through deep learning approach. 68pp f. ill. 2018. Dissertation (Master degree) - Multidisciplinary Institute of Health, Federal University of Bahia, Vitória da Conquista, 2018.

ABSTRACT

Peptides derived from endogenous antigens can bind to MHC-I molecules. Those which bind with high affinity can invoke a CD8+ immune response, resulting in the destruction of infected cells. Much work in immunoinformatics has involved the algorithmic prediction of peptide binding affinity to various MHC-I alleles. A number of tools for MHC-I binding prediction have been developed, many of which are available on the web. However, many machine learning-based methods are generally less sensitive in recognizing locally-clustered interactions, which can synergistically stabilize peptide binding. Here we propose the EPI predictor, an artificial neural network-based for HLA-A*0101 allele. Our ANN was built using 3902 epitopes for HLA-A*0101 and nine physical-chemical properties. Then the physical-chemical properties were reduced to four PCA and used as features for each amino acid in the train and test peptides. We train/test the model using 70/30% of the epitopes retrieved from the database respectively. Our model reached a good prediction level with an accuracy of 77.09 and f1 score of 0.77 in the test set. Additionally, several hyperparameters test were done to identify the most suitable multilayer neural network. In addition to the validation testing protocol, we sought to compare the accuracy rate performed by our predictor in comparison to several other predictive approaches available online. The benchmark demonstrated EPI displaying an outstanding performance in contrast to some reliable MHC-I predictors. The EPI reached 0.53 MCC score, indicating consistent predictive capacity. The strategy to build the EPI and performance compared with other predictors are discussed.

Keywords: epitope prediction, artificial neural network, machine learning.

LISTA DE ILUSTRAÇÕES

Figura 1 Molécula de MHC classe I....16
Figura 2 Via de processamento e apresentação do epítopos de MHC I....18
Figura 3 Estrutura da molécula de MHC I....20
Figura 4 Esquema de um neuron (ou *node*) dentro de uma rede neural....24
Figura 5 Diferença entre rede de camadas simples e redes densas de múltiplas camadas....25
Figura 6 Esquema de análise combinatória para diAA....31
Figura 8 Scatterplot cenário 1....66
Figura 9 Scatterplot cenário 2....66
Figura 10 Scatterplot cenário 3....67
Figura 11 Scatterplot cenário 4....67
Figura 12 Scatterplot cenário 5....68
Figura 13 Scatterplot cenário 6....68
Figura 14 Scatterplot cenário 7....69
Figura 15 Scatterplot cenário 8....69

LISTA DE ILUSTRAÇÕES DO ARTIGO

Figure 1 diAA Scheme....43
Figure 2 Training process of the ANN....45
Figure 3 LogoPlot of all HLA-A01*01....48
Figure 4 The plot of model accuracy and loss on train with Shallow and deep learning....49
Figure 5 Accuracy measurements....51
Figure 6 Model accuracy and loss on on PC4 and ReLu with ⅔ negative....52
Figure 7 Accuracy performance in train and test set for the different ANN architecture. 53
Figure 8 3D ANN accuracy score....54
Figure 9 Predictors Accuracy comparison among 11 different predictors....56

LISTA DE TABELAS ARTIGO

Table 1 Positive epitopes sequence through filtering steps 47
Table 2 Four first PCA 49
Table 3 Activation functions performance (Tanh and ReLu) 51
Table 4 The best training and test scenario evaluated by statistical scores 52
Table 5 Benchmarking confusion matrix 55
Table 6 Benchmarking predictor method summary 56

LISTA DE ABREVIATURAS E SIGLAS

ANN	Artificial Neural Network
APC	Célula Apresentadora de Antígenos
$CD4^+$	Cytotoxic T cell 4+
$CD8^+$	Cytotoxic T cell 8+
CNN	Convolution neural network
ER	Endoplasmatic reticulum
ERP57	Thiol oxidoreductase
HLA	Human leucocyte allele
HMM	Hidden Markov Model
IEDB	Immune epitope database analysis resource
IMGT	international ImMunoGeneTics
kDa	Kilodaltons
MCC	Matthews Correlation Coefficient
MHC-I	Major histocompatibility complex II
MHC-II	Major histocompatibility complex I
ML	Machine learning
MM	Matrix motif
PCA	Principal Component Analysis
PLC	Peptide ligand carrier
PSSM	Position specific scoring matrices
QAM	Quantitative affinity matrix
QSAR	Quantitative spectrometric data-activity relationship
SMM	Stabilized matrix method
SVM	Support Vector Machine
TAP	Transporter associated with antigen processing

LISTA DE ABREVIATURAS E SIGLAS

IA	Inteligência artificial
RCT	Receptor de célula T
RCB	Receptor de célula B
ECTs	Epítopo de célula T
ECBs	Epítopo de célula B

SUMÁRIO

INTRODUÇÃO 14
REVISÃO DA LITERATURA 19
JUSTIFICATIVA 28
OBJETIVOS 29
GERAL 29
ESPECÍFICO 29
MATERIAL E MÉTODOS 30
AMOSTRAS DE ESTUDO 30
DESENVOLVIMENTO DE FEATURES 30
ARQUITETURA DE REDE NEURAL 31
OPTIMIZAÇÃO DE REDE NEURAL 32
COMPARAÇÕES E VALIDAÇÕES 33
REFERÊNCIAS 34
CAPITULO 1 40
CONSIDERAÇÕES FINAIS 65
ANEXO I 66

INTRODUÇÃO

1.1. Sistema Imunológico

O sistema imunológico é constituído da ação coordenada de células e moléculas especializadas em defesa. O sistema imunológico é capaz de gerar resposta de defesa de forma rápida, eficiente e específica contra um grande número de patógenos e substâncias estranhas. O sistema imune de vertebrados evoluiu para diversificação de resposta mediada por diferentes tipo celulares capazes de reconhecer e proteger contra potenciais invasores (Male, et al, 2013).

O sistema imune apresenta respostas diferenciadas contra antígenos que podem ser divididas em resposta não adaptativa (resposta inata) e resposta adaptativa. A resposta inata contra o invasor é mediada por ações coordenadas e sequenciais chamada de primeira etapa de defesa. A notável eficiência decorre das células da imunidade inata possuírem a capacidade de reconhecer padrões moleculares associados a patógenos presentes em inúmeros microrganismos (Murphy e Weaver, 2017). Este tipo de resposta é essencial para defender o organismo durante as primeiras horas ou dias durante a infecção antes da imunidade adaptativa agir.

A segunda etapa caracteriza-se pela resposta imune adaptativa, responsável em identificar e reagir frente a inúmeros compostos, criando uma resposta específica contra o antígeno. A principal característica deste é a resposta específica desempenhada pelas células imune em resposta ao agente infeccioso. O sistema adaptativo irá ativar células para diferenciação em células linfócitos T auxiliares, linfócitos B produtores de imunoglobulinas (anticorpos), e linfócitos T citotóxicos. A apresentação do antígeno irá direcionar o sistema imune adaptativo para combater o agente estranho ao hospedeiro, determinando quais tipos celulares devem ser produzidos. O agente infeccioso irá estimular o sistema imune adaptativo que irá armazenar as informações por células de memória, possibilitando uma resposta mais rápida e potente caso o indivíduo seja novamente exposto. O amplo espectro de substâncias microbianas e não microbianas que reagem com o sistema imune adaptativo é chamado de antígenos (Abbas, 2017).

Fragmentos de antígeno são potencialmente antigênicos e capaz de desencadear a ativação da resposta imune adaptativa. Tais fragmentos são chamados de epítopos, que são pequenas estruturas formada por diferentes compostos orgânico, como lipídeos, nucleotídeos, proteínas e carboidratos (Henot, 2015). O reconhecimento de epítopos pode acontecer em duas linhas linfocitárias, celular mediada por células T e humoral por intermédio dos linfócitos B

secretores de anticorpos. Nem todos fragmentos nucleicos ou proteicos são capazes de desencadear uma resposta imune, portanto a determinação de quais epítopos serão imunogênicos pode ser utilizado para diversos propósitos (Mamitsuka, 1998).

Os antígenos B e T, ECBs e ECTs respectivamente, são bem distintos e possuem características marcantes. O epítopos de linfócito B são hidrossolúveis sua grande maioria reconhecidos por receptor de célula B (RCB) presentes na membrana. fazendo o reconhecimento de superfície encontra-se aderido à imunoglobulina (Sanchez-Trincado, Gomez-Perosanz e Reche, 2017). A facilidade de reconhecimento é dada em função da fácil imersão dos peptídeos dentro do interior citoplasma. Uma vez ativada, as células B sofrem a diferenciação, expansão, e secretam imunoglobulinas solúveis (anticorpos), mediando assim a resposta humoral. ECBs podem ser lineares contendo uma sequência contínua de aminoácidos dentro de uma proteína ou tridimensionais em que as porções de epítopos residem em locais distantes na estrutura primaria da proteína, mas são estruturalmente próximas na conformação tridimensional da estrutura terceira da proteína (Blythe e Flower, 2009). As características dos epítopos de ligação aos RCB tornam as predições epitópicas ser menos acuradas comparado com ECTs. Algumas abordagens como cristalização e *docking* proteico são atualmente as principais formas de estudo sobre os epítopos descontínuos. Entretanto o uso de estratégias experimentais para descoberta do epítopos ligantes de RCB fazem as descobertas dessas regiões imunogênicas serem demoradas.

Os Linfócitos T possuem capacidade de reconhecimento de epítopos apresentados pelo complexo de histocompatibilidade principal (MHC)-I ou MHC-II. Os epítopos ligantes de MHC-I são de tamanho curto variando entre 8 e 12 nucleotídeos, sendo restrito somente ao reconhecimento de sequencias peptídicas (Patronov & Doytchinova, 2013, Abbas 2017). As células apresentadoras de antígenos, APC, apresentam os epítopos na superfície celular ligados à MHC, a fim de induzir a resposta imune. O MHC em humanos também é conhecido como antígenos leucitários humanos (HLA). Assim, é de crucial o entendimento da ligação entre peptídeo e MHC para que predições precisas sejam feitas a partir dos padrões físico, químico e estruturais entre estes.

1.2. MHC e processamento de antígenos

O MHC pode ser dividido em três grandes grupos, de classe I, II e III. O MHC-I e II possuem inúmeras moléculas distintas de MHC (Sharon *et al.*, 2016). Estes são tremendamente polimórfico, em função da sua síntese ser oriunda de um grande agrupamento gênico, o locus

MHC. Existe diversas moléculas de MHC para cada uma de suas classes com muitas variações contida dentro das subclasses. Esta característica confere uma inestimável vantagem evolutiva para o grande espectro de possibilidades de epítopos que podem ser apresentados para a célula T através do MHC. A maioria dos genes polimórficos codificam proteínas que diferem apenas em poucos aminoácidos, entretanto a variação alélica do MHC diferem uns dos outros por até vinte amino ácidos (Murphy e Weaver, 2017).

O MHC-I e II são glicoproteínas ligadas à uma membrana, altamente especializados em ligação de epítopo através de grande afinidade presente no sulco ou bolso do MHC capaz de manter ligações estáveis com o epítopo. Existe uma leve diferença na estrutura quaternária entre MHC-I, presente em todas células nucleada, e MHC II presente apenas em APCs. Para o MHC-I seu principal papel é a apresentação de peptídeos para o receptor de célula T (TCR) presente em linfócito T $CD8^+$. O MHC_II são transcritos e expressos em APC que apresenta seus antígenos para o linfócito T $CD4^+$ (Owen, 2013).

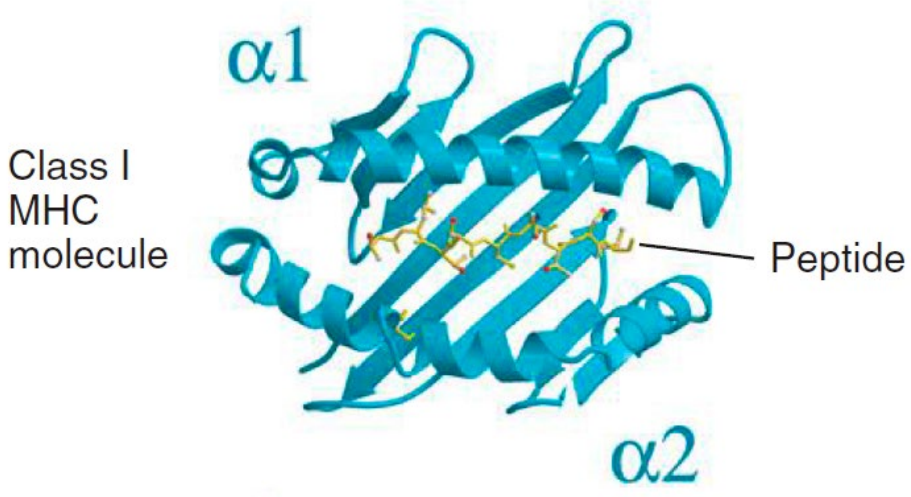

Figura 1. Molécula de MHC classe I. Está é uma representação tri dimensiona do alelo HLA-A2. a fenda da molécula de HLA I é fechado e como consequência as sequências aptas a se aderirem ao complexo deve ser obrigatoriamente menor. O sitie de ligação, ou região pocket se dá pela interação entre as laterais alpha Helix e a base beta pregueada.

A molécula de MHC-I é formada pela montagem de dois polipeptídios: cadeia α (45 quilodaltons, kDa) e a β2 microglobulina, figura 1. A cadeia alfa possui três domínios externos: α1, α2 e α3, além de um domínio transmembrana que ancora o complexo na superfície celular. A β2 microglobulina, juntamente com a α3, tem a primordial função de estabilidade estrutural, formando a estrutura de imunoglobulina pelo qual a célula T $CD8^+$ interage. Os domínios α1 e

α2 juntos formam um bolso localizado na superfície do MHC sendo largo o suficiente para se ligar a peptídeos de 8 até 12 amino ácidos. Todas as porções que compõem o MHC são essenciais para o correto enovelamento e expressão de MHC-peptídeo na superfície celular (Patronov e Doytchinova, 2013).

O MHC-I se expressa por várias células, estando ou não infectadas por algum patógeno. Quando exposta a um invasor que invade a membrana e se utiliza da maquinaria replicativa do hospedeiro para se multiplicar (Moretta *et al.*, 1996). Os antígenos presentes no citoplasma são ubiquitinados e então degradado pelo complexo do proteasoma, um complexo que essencialmente irá degradar proteínas completas em pequenos fragmentos de aproximadamente 15 amino ácidos de comprimento. Na membrada do retículo endoplasmático encontra-se a proteína transmembrana de transporte associada com processamento de antígenos, ou TAP1 e TAP2 (Li *et al.*, 2000). Durante o transporte dos produtos proteolíticos do proteasoma para o interior do retículo endoplasmático, a TAP1 e 2 são fosforiladas e ativadas. Alguns aminoácidos são clivados decaindo para 8-12 aminoácidos, tamanho ideal para o reconhecimento do MHC-I.

Ribossomos do retículo endoplasmático produzem endogenamente MHC-I incompleto que se adere em sua membrana pela porção transmembrana. A proteína calnexina, uma chaperona, irá auxiliar o dobramento e a qualidade da estrutura terciária do MHC-I incompleto. Uma vez que a β2 microglobulina se adere ao MHC incompleto, este se tornará completo e irá se aderir à outras proteínas como a oxiredutase ERP57 e calreticulina, que essencialmente cobre a região de ligamento do MHC prevenindo assim a incorreta ligação de peptídeos endógenos do hospedeiro. Adicionalmente e calreticulina agirá como chaperona, conduzindo o MHC até a tapasina, proteína auxiliar que estabiliza a TAP, aumentado assim o influxo de peptídeos e possibilita a desassociarão da ERP57 e calreticulina, formando o complexo de peptídeo carreado MHC-I (PLC) (Rizvi e Raghavan, 2010; Weijer, van de, Luteijn e Wiertz, 2015). Uma vez que o sulco de ligação peptídica se torna exposto, somente pouquíssimas sequências com afinidade irão se ligar.

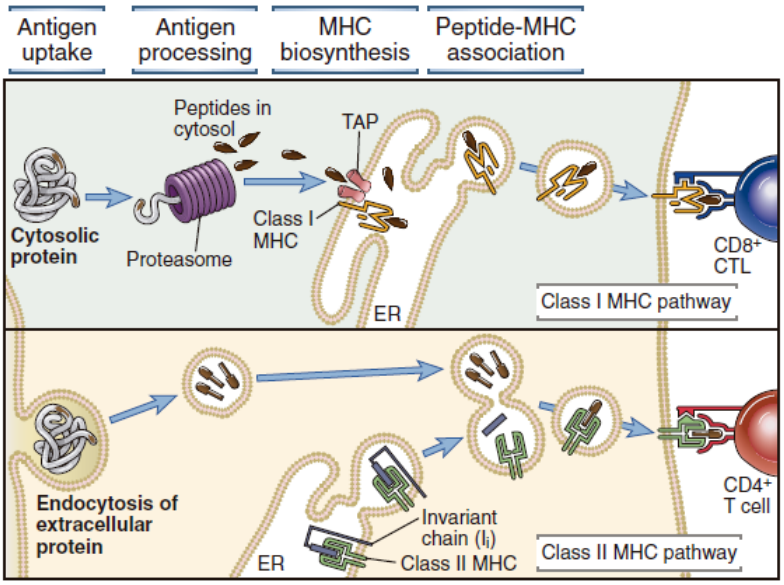

Figura 2. Via de processamento e apresentação do epítopos de MHC I. Pelo topo: os antígenos proteíco no citosol são processadas por proteassomas, esses fragmentos são transportados para o retículo endoplasmático via TAP, onde ira se ligar ao MHC. ER, retículo endoplasmático e TAP, transportador associado com processamento de antígeno. (Abbas, Lichtman e Pillai, 2014)

Apesar da notável variação alélica, as fendas de ligação peptídica no MHC possuem certas características que possibilitam detectar padrões e consequentemente tentar predize-los. As características são o comprimento dos epítopos e é o fato de algumas posições de certos alelos possuírem alta afinidade por determinados tipos de amino ácidos, nomeadas como regiões âncora. A apertada fenda do MHC-I possui 6 regiões bolso nomeadas de A até F (Groot, de *et al.*, 2017). As regiões B e F estão normalmente vinculada às regiões conservadas de ancoragem, por exemplo, os aminoácidos da 2ª e 9ª posição se ligam as regiões B e F respectivamente para o supertipo humano HLA02 (Remesh *et al.*, 2017).

As interações dos aminoácidos presentes no epítopo e no bolso do MHC são l baseados nas interações físico-químicas entre as 2 α hélices localizadas nas bordas da fenda e as 8 β pregueadas em seu assoalho. Pontes de hidrogênio são formadas entre as cadeias laterais dos epítopo e a molécula de MHC. Adicionalmente, foi descoberto que na ausência de peptídeo de forte ligação, a molécula de MHC-I sofrem modificações conformacional, tornando a região do bolso F mais flexível para o emparelhamento de peptídeos com baixa afinidade (Wieczorek *et al.*, 2017).

Para que a ativação das células citotóxicos aconteça, o MHC-I ligado ao peptídeo saem do retículo endoplasmático através de vesículas, migrando em direção ao complexo de Golgi,

sendo assim devidamente endereçada à superfície da camada externa da membrana celular. A célula infectada passará a expressar o MHC-I com epítopo ligado, ao qual fará interação com o linfócito T citotóxico. O linfócito $CD8^+$ possui diversos receptores de superfície, dentre eles o receptor CD8 ao qual tem a atribuição de manter a proximidade entre as células infectadas e citotóxicas durante a ativação antígeno específica. O receptor de celular T (TCR), que deverá verificar a ligação específica com o epítopo que está sendo apresentado. Caso ambas verificações sejam procedentes, a $CD8^+$ irá lançar proteínas (perforinas e granzimas) que terão o encargo de matar a célula infectada (Abbas, Lichtman e Pillai, 2014). A ativação da T $CD8^+$ envolve outras co-estimulações, mas é notório o fato da apresentação dos epítopos por MHC-I ser a etapa mais crucial.

Durante o curso evolutivo, muitos vírus e patógenos adquiriram inibidores da apresentação de antígeno pelo MHC-I, capaz de estressarem a maquinaria do MHC-I. Os mecanismos de evasão além de fornecerem elucidação sobre a vias de resistência viral, possibilitaram maior entendimento sobre à apresentação de antígenos do MHC, bem como revelar processos celulares mais gerais, incluindo vias de degradação de proteínas por ubiquitinação e pelo proteasoma (Weijer, van de, Luteijn e Wiertz, 2015). As principais rotas de evasão virão são: bloqueio de tradução de genes responsáveis na inserção do MHC na membrana do reticulo endoplasmático; repetições de glicina e alanina que causam distúrbio com o proteasoma, inviabilizando a degradação; interferência da associação de peptídeo com a proteína de transporte TAP; inibição das funções das tapisina, responsável pela estabilização do MHC I bem como na seleção de epítopos de alta afinidade.

REVISÃO DA LITERATURA

2.1 HLA e variação alélica

O antígeno leucocitário humano, HLA, participa da apresentação de antígenos e possui uma grande diversidade genética, portanto, uma grande importância imunológica. A importância do HLA é evidenciada em situações como o principal fator de compatibilidade entre doadores de órgão; a possibilidade de compatibilidade residindo nas variantes alélicas do HLA do doador e receptor. Também participam do desenvolvimento das doenças autoimune incluindo a diabete tipo I, narcolepsia e doença celíaca (Markov e Pybus, 2015). Cada tipo alélico possui a aptidão de detectar um único conjunto de epítopos, e para cada *locus* os

indivíduos podem apresentar até dois alelos diferente (indivíduo heterozigoto). Alguns fatores são cruciais para a variabilidade deste gene, como a evidência de estudos que indicam influência do MHC em casais de alelos discordantes e sua taxa de fecundidade aumentada (Khankhanian *et al.*, 2010).

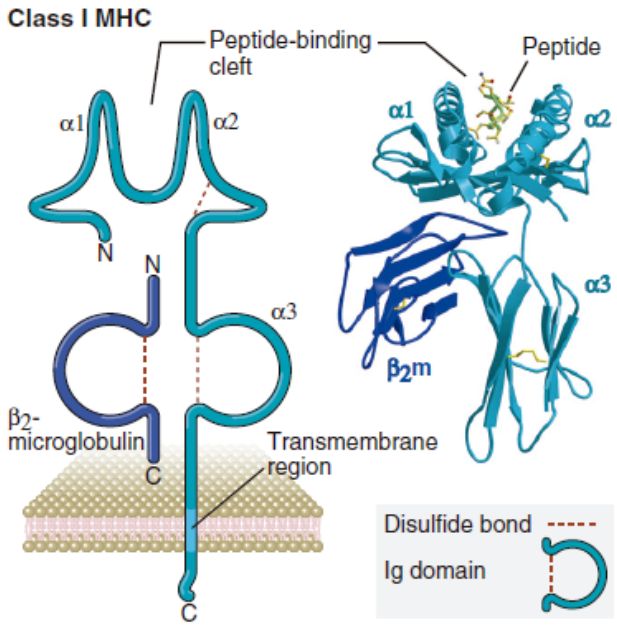

Figura 3. Estrutura da molécula de MHC I. A figura à esquerda ilustra diferentes regiões da molécula. O MHC é composto de cadeias alfa polimórficas aderido à microglobulina beta. O diagrama de fitas da direita representa a ligação entre o fragmente peptídico e a região *pocket* do alelo HLA-B27.

A parte da ação do sistema imune depende do reconhecimento de peptídeos ligantes ao HLA para disparar resposta específica contra o patógeno. Esse reconhecimento de epítopos e desenvolvimento de vacinas é dependente da variação alélica do HLA. Atualmente, mais de 8.000 variações genicas para HLA de classe I foram identificados, ressaltando o desafiador processo de capturar predições para epítopos que contemple toda a vasta variedade (Khankhanian *et al.*, 2014).

Mudanças em um único aminoácido dentro da fenda do HLA pode alterar a especificidade de ligação dado uma molécula de MHC. Mutações de substituições simples ou SNP eventualmente ocorrem dentro da sequência gênica de HLA, a mais polimórfica do genoma humano (Santos Francisco, dos *et al.*, 2015). O HLA classe I possui três loci: HLA-A, HLA-B e HLA-C, os supertipos que são agrupados por semelhanças em seus domínios de

ligação. As principais mutações no DNA são do tipo substituição não sinonímias, favorecendo a hipótese que leva a crer que seleção balanceadora tem mantido as variações nestas regiões genômicas (Meyer *et al.*, 2018). Os altos níveis de variação observado nos sítios de ligação ao peptídeo apontam para o modelo coevolutivo entre o patógeno e o hospedeiro, indicando que a diversidade do HLA deve ser alta para o reconhecimento da grande quantidade de patógenos presentes no meio ambiente (dos Santos Francisco et al., 2015).

2.2 Imunoinformática

A necessidade de manusear rapidamente a grande quantidade crescente de informações imunológicas exigiu que novas tecnologias fossem desenvolvidas para suprir essas demandas (Korber, LaBute e Yusim, 2006). Não somente para gerenciar a grande quantidade de dados, mas também análises e predições *in silico,* possuindo um grande papel na definição de novas hipóteses relacionadas à resposta imune conhecido com imunoinformática. A imunoinformática trata de apresentar aplicações de métodos computacionais para resolver problemas imunológicos (Patronov e Doytchinova, 2013). Historicamente, ferramentas para predições das ligações entre HLA e peptídeos foram as primeiras ferramentas desenvolvidas especificamente para essa aplicação (Backert e Kohlbacher, 2015).

Ensaios de ligação de alto rendimento com antígenos de HLA possibilitaram grande avanço nessa área. Atualmente, o sequenciamento de última geração produz uma grande quantidade de informação e isso tem facilitado o treinamento de métodos baseados em inteligência artificial. Os preditores de epítopos começaram a ser treinados com dados advindos destas técnicas de biologia molecular, refletindo no aumento da acurácia dessas ferramentas preditivas (Tomar e De, 2014). Normalmente, a investigação da afinidade de ligação de peptídeo antigênico à molécula de MHC é o principal objetivo de um preditor de epítopos. A determinações de epítopos por meio de experimentação em bancada são difíceis, dispendiosas e demoradas (Tomar e De, 2014). Portanto, diversas metodologias *in silico* são criadas constantemente para melhorar a identificação de epítopos. A confiabilidade das predições só é viabilizada pelos bancos de dados que aglutinam diversos experimentos laboratoriais, que servem de referência ao aprendizado de máquina.

2.3 Banco de dados

Inúmeras técnicas imunológicas para entender os sistemas imunes e suas respostas as vários agentes infecciosos como: cromatografia de afinidade, citometria de fluxo, radioimunoensaio, ELISA e ensaio de competição inibitiva, foram utilizadas para alimentar bancos de dados curados como IEDB, SYFPEITHI e IMGT (Ehrenmann e Lefranc, 2011; Schuler, Nastke e Stevanović, 2007; Vita *et al.*, 2015). Os bancos de dados imunômicos fornecem informações sobre a imunogenicidade, habilidade de um antígeno induzir uma resposta imune e antigenicidade, a propriedade de um antígeno se ligar a um anticorpo (Khalili *et al.*, 2014). O IEDB (immune epitope databases and analysis resources) em parceria com o Instituto Nacional de Alergia e Doenças infecciosas reúne mais de 120.000 epítopos curados, baseados em artigos disponíveis no PubMed, (Vita *et al.*, 2015).

Os métodos preditivos utilizados para identificar epítopos são amplamente diversos, mas resumidamente caracterizados em dois grupos: por métodos diretos e indiretos. Métodos diretos são baseados na análise de estrutura e sequência dos epítopos, que se baseiam em características como anfipaticidade e *motifs* de ligação ao MHC (Sanchez-Trincado, Gomez-Perosanz e Reche, 2017). Essas abordagens apresentaram baixa confiabilidade e alta taxa de falso positivo. Enquanto os métodos indiretos utilizam-se de ferramentas baseadas em aprendizado estatístico, aprendizado de máquina (*machine learning*), contendo acurácia maior (Thus *et al.*, 2016). Adicionalmente à predição do sítio de clivagem proteassomal, transporte pela TAP, ligação MHC-pepíteo, e reconhecimento do complexo pelo receptor de células T (TCR) também são componente importante para serem preditos. Indubitavelmente, a ligação do peptídeo à fenda do MHC é a etapa mais relevante para determinação epitópicas.

A fenda de ligação do MHC-I, em seu sulco, é estruturalmente formado por duas α hélices e a proximidades destas possibilita apenas pequenos oligopeptídeos de se ligarem por meio das extremidades C e N terminal através das pontes de hidrogênio (Sanchez-Trincado, Gomez-Perosanz e Reche, 2017). Algumas regiões da fenda ligam-se com mais intensidade, em função das interações físico-químicas com os fragmentos peptídicos. Pesquisadores concluíram que em função disto diferentes tamanhos de epítopos têm a capacidade de ligar à mesma molécula de MHC, sendo, portanto, um complicador para os preditores, aos quais necessitam estipular um tamanho específico para predictabilidade (Yang e Yu, 2009). Como a grande maioria dos epítopos de MHC-I possuem nove aminoácidos de comprimento, é convencionalmente utilizado este tamanho na maioria dos preditores.

Diversas são as abordagens *in silico* para melhorar a identificação de epítopos, dentre

elas os métodos baseados em matrizes, análises relação estrutura-atividade quantitativa (QSAR), identificação de sequências *motifs*, modelagem de homologia, docagem molecular e desenho de diversificados algoritmos de aprendizado de máquina (Calis *et al.*, 2013). Tais métodos são empregados predominantemente por uma abordagem baseado somente nas sequências de aminoácidos dos epítopos. Contudo a predição de epítopos, especialmente para MHC-II e células B utilizam-se das características estruturais do peptídeo, podendo inclusive utilizar metodologia hibridas entre estrutural e sequencial ou ainda *consensus*, utilizando a combinação dos pontos fortes dos diferentes métodos (Blythe e Flower, 2009).

2.4 Aprendizado de máquinas

Aprendizado de máquina (ML) é definido como um processo automático de aprendizagem de um algoritmo, que extrai padrões oriundo de um conjunto de dados (Kelleher, Namee, Mac e D'Arcy, 2015). E um ramo da inteligência artificial (IA) baseado na ideia de que o sistema possa aprender com os dados de referência, identificar padrões e fazer decisões com o mínimo de intervenção humana. Dentre as ferramentas que constituem a IA destacam-se algoritmos genéticos, processamento de linguagem natural, métodos probabilísticos e redes neurais (Lundegaard, Lund e Nielsen, 2011). Aprendizado de máquina pode ser categorizado de variadas formas, como por exemplo quanto à supervisão humana, sendo assim dividida em supervisionada e não-supervisionada ou *reinforcement learning*.

Os modelos supervisionados são utilizados para descobrir padrões em conjunto de dados conhecidos fazendo a associação de dado de entrada com o dado (resultado) de saída do algoritmo. Um conjunto de dados, descrito como dados de treinamento, alimenta o algoritmo e serve como guia para referenciar quais serão os padrões a serem detectados. Esses padrões devem ser reconhecidos em outro conjunto de dados não usados durante o treinamento, chamados de dados de teste. Os propósitos das máquinas supervisionadas normalmente buscam obter classificações ou regressões de dados de entrada. Para responder tais questionamentos os algoritmos supervisionados se validam por meio de estatísticas como, regressões lineares, decisões de arvores, *k-Nearest Neighbor*, dentre outros (Koprinkova-Hristova, Mladenov e Kasabov, 2015).

Um dos métodos de aprendizado de máquina que pode ser supervisionado são as redes neurais artificiais (ANN). As ANN são definidas por um mapeamento não linear de um vetor de espaço de entrada para um vetor de espaço de saída, opcionalmente utilizando camadas de neurônios, ao qual os valores de coordenada de entrada são somados ao valor do seu peso a fim

de produzir um valor de saída simples, quer é ativado ou não de acordo com determinado nível de disparo (Kelleher, Namee, Mac e D'Arcy, 2015). A analogia provem do cérebro que possui neurônios responsáveis em emitir impulsos nervosos em resposta a um estimulo externo. O complexo sistema interconectado neuronais é referência para modelagem matemática de um sistema computacional que simule tal ambiente. Cada estrutura que executa funções matemática dentre de uma rede artificial é chamada de neurônio e os valores de entrada ou saída em cada neurônio são vistos como dendritos. Os pesos estipulados para engatilhar a transmissão de um valor computado para outro neurônio é similar à uma sinapse (Kelleher, Namee, Mac e D'Arcy, 2015).

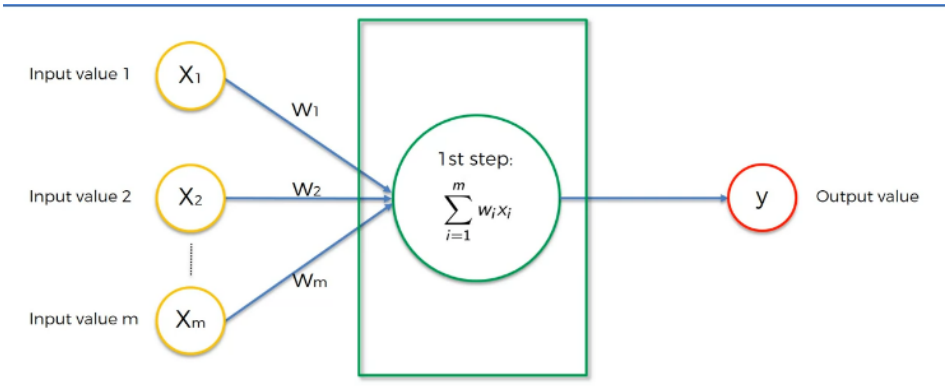

Figura 4. Figura esquemática de um neuron (ou *node*) dentro de uma rede neural. O neurônio possui sinais de entrada, input, em amarelo, e sinas de saída, output em vermelho. A transmissão dos dados de entradas através de impulsos ou sinapses. W é definido com o peso estabelecido para cada impulso ser ativado. (SuperdataScience ©, 2017). O sinal no neurônio de todos os inputs é somado de acordo o respectivo peso.

Basicamente uma rede neural possui três tipos diferentes camadas: camada de entrada, camada oculta e camada de saída (figura 4). A camada de entrada recebe os dados do ambiente externo, ao qual possuem valores que representam as características do objeto (*features*). Esses valores de entrada são passados durante o treinamento ou durante o uso da ANN. A camada oculta é a coleção de neurônios que possuem suas respectivas funções de ativações, responsável em extrair e processar as características contidas nos valores iniciais (Lundegaard, Nielsen e Lund, 2006). A quantidade de camadas é uma variável de intenso debate dentro da comunidade científica. As redes multicamadas, ou *perceptrons,* pode implicar em aumento na capacidade

de processamento e generalização da rede (Du e Swamy, 2014). A saída de um valor dentro das camadas ocultas serve de entrada para a camada subsequente, não havendo retroalimentação pois a propagação é unidirecional até a camada de saída. Está última tem o encargo de transmitir a informação de acordo com o que foi desenhado pela rede.

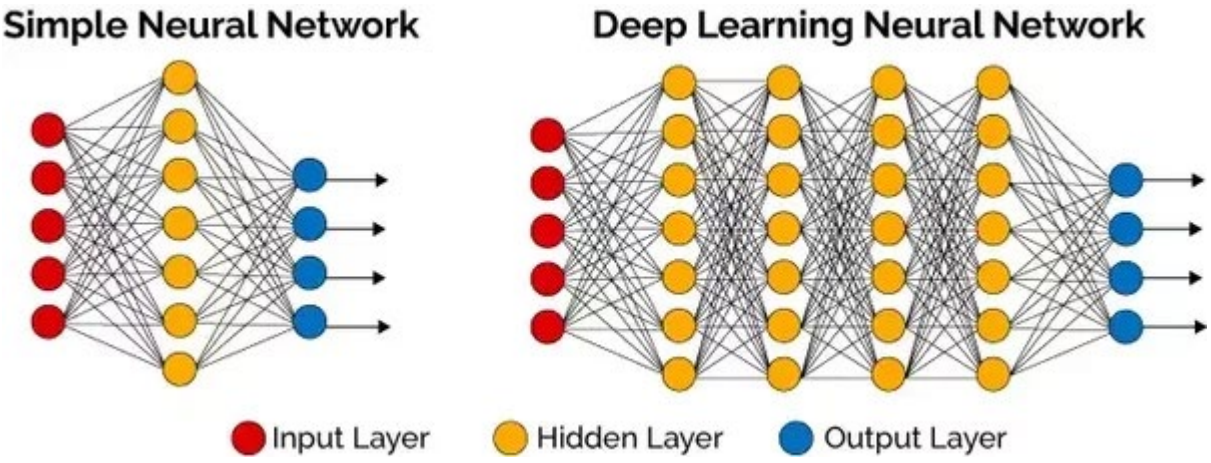

Figura 5. Esquema de diferença entre rede de camadas simples e redes densas de múltiplas camadas. A esquerda a rede apresenta somente uma camada escondida, em laranja. Entretendo, a direita observa-se uma rede multicamadas, então chamado de *deep learning*.

Os preditores baseados em sequências de aminoácidos, dependem de banco de dados e usam de posições âncora de ligação com referência para as predições, estes são chamados de *sequence motif*, SM (Sanchez-Trincado, Gomez-Perosanz e Reche, 2017). Estes primeiros preditores foram desacreditados pela possibilidade de regiões não ancora ter grande papel na ligação ao MHC. Subsequentemente, matrizes *motif* (MM) ampliaram a relação de região âncora com as outras posições de um peptídeo. Os preditores SYFPEITHI e Rankpep são exemplos de preditores que usam MM (Reche *et al.*, 2004; Schuler, Nastke e Stevanović, 2007). Outro tipo de preditor mais sofisticado leva em consideração características de afinidade de ligação entre os aminoácidos e a região *pocket* do MHC, este chamado de *quantitative affinity matrix* (QAM) (Soria-Guerra *et al.*, 2015). Preditores como BIMAS, TEPITOPE e Epijen são baseados no modelo QAM (Doytchinova, Guan e Flower, 2006; Parker, Bednarek e Coligan, 1994; Zhang *et al.*, 2012).

Evidências acerca da influência dos aminoácidos vizinhos sobre as regiões ancoras levaram a formulação do *quantitative structure-activity relationship* (QSAR) em que o valor de afinidade de ligação é computado com base na interação entre todas as outras posições de um epítopo (Karosiene, Nielsen e Lund, 2013). MHCPred é o preditor mais famoso que utiliza

QSAR (Guan *et al.*, 2006). Com o acumulo informações biológicas nas duas últimas décadas, os preditores fundamentado em ML popularizam em função da sua robustez. O preditores baseados em ANN, Máquina de vetores de suporte (SVM), árvores de decisões (DT) e modelo oculto de Markov (HMM) são exemplo de redes neurais utilizadas para fins preditivos. Como um dos principais limitantes da classificação por ML é a quantidade de sequência de entrada a serem usadas no treinamento da rede, os preditores renomados como NETMHC, IEDB-MHCI, nHLApred e SVMHC utilizaram diferentes abordagens para contornar este fator limitante (Bhasin e Raghava, 2007; Dönnes e Elofsson, 2002; Kim *et al.*, 2012; Lundegaard *et al.*, 2008)

2.5 Métodos de validação

Apesar dos testes de eficiência comprovarem desempenho superior dos métodos de ML, um grande impeditivo de melhoras de predição está no número de entradas de epítopos validados e curados em relação ao grande universo de possibilidade para nove posições de aminoácidos. As principais abordagens para qualificar e validar um preditor ocorre através da *Confusion Matrix* (Koprinkova-Hristova, Mladenov e Kasabov, 2015). A matriz fornece informações sobre os acertos e erros, do tipo I e II, cometido pelo preditor. A acurácia é a taxa total de predições corretas sobre o número total de predições (Koprinkova-Hristova, Mladenov e Kasabov, 2015; Tomar e De, 2014). Precisão, em contrapartida, é o valor de predições positivas corretas sobre o número total positivos preditos (Tomar e De, 2014). Além disso, Recall, outro termo, conhecido também como sensibilidade é a proporção de verdadeiros positivos sobre o número real de positivos. Por fim, F1, que se caracteriza pelo balanço entre precisão e sensibilidade, em que sua performance é melhor vista em situações de desbalanço entre valores de entrada positivo e negativo (Lee *et al.*, 2018).

Apesar de acurácia e F1 serem amplamente utilizados na estatística, seus resultados podem ser enganosos pois estes não levam em consideração todos os quatro cenários existente na *Confusion Matrix*. Em contrapartida, *Matthews Correlation Coeficiente*, MCC, é o único método de validação que leva em consideração corretamente taxa de todos os valores da *Confusion Matrix* (Matthews, 1975). Sua escala varia entre -1 e 1, em que 1 configura-se como o mais confiável possível (Boughorbel, Jarray e El-Anbari, 2017). O desempenho de um preditor para MCC só será satisfatório, mediante bons resultados entre os elementos positivos e negativos.

2.6 Aplicações

É vasta as implicações dos preditores de epítopos para a saúde pública e a pesquisa básica. Dentre as inúmeras aplicações de predições para pesquisa, o uso da vacinologia reversa para prospectar peptídeo candidatos para desenvolvimento de vacinas de subunidade, estudos a respeito das doenças autoimune, no tratamento de doenças alérgicas e estruturas proteicas são certamente as áreas de interesse que mais despontam em progressos e em ferramentas para pesquisadores. Vacinas baseadas em epítopos são normalmente dispendiosas em tempo e investimento de bancada, mas seus resultados são promissores em conferir proteção para modelos animais em triagem clínica. Todavia, apesar das enormes implicações na saúde pública, segurança e meio científico, predições de epítopos podem ser utilizados para meios não honráveis, como desenvolvimento de arma bioquímica, que possibilitaria a evolução patógenos mutantes.

Outra evidente aplicação é a imunoterapia personalizada ao que vem sendo investigada em diversos laboratórios ao redor do mundo (Backert e Kohlbacher, 2015; Korber, LaBute e Yusim, 2006). O objetivo principal é o uso de antígenos específicos de tumores de células somáticas, para que então seja possível delinear uma resposta imune específica contra as células tumorais. Com o uso de predição de epítopos por células T, foi possível obter enormes avanços em desenvolvimento de vacinas baseada em subunidades bem como os avançados tratamentos de imunoterapia.

JUSTIFICATIVA

Recentes avanços nas tecnologias de caracterização identificação ode epítopos têm possibilitado esforços cada vez mais sistemáticos para predição de potenciais sequencias capaz de desencadear uma resposta imune controlada. A identificação de novos epítopos pode fornecer informações importantes sobre novas abordagens de gerar imunogenização por meio de vacinas de subunidades. Nesse raciocínio, diversos preditores surgiram ao longo das últimas duas décadas, tendo diversas estratégias preditivas. Atualmente, mediante diversas possibilidades, o processo de escolha do mais adequado preditor requer considerável expertise. Além disso, poucos são aptos para identificar e predizer possíveis sequências epitópicas com acurácia e confiança. Infelizmente, ainda não disponibilizamos de um grande acervo de sequências reconhecidas pelo MHC. Dessa forma as várias técnicas que variam abordagem para contornar os dados modelos tem sido proposta. As predições por rede neural possibilitam a inversão do estudo clássico do desenvolvimento de vacinas, propiciando assim a vacinologia reversa., que busca primariamente identificar possíveis fragmentos proteicos capaz de gerar resposta imune de memória. Sem sombra de dúvidas essa metodologia é menos dispendiosa em termos de tempo e dinheiro.

Foi notado que a literatura carece de comparativos em larga escala averiguando o desempenho de novos métodos. Os estudos de benchmarking, que ainda são escassos, normalmente se restringem aos alelos com mais depósitos nos bancos de dados. Somente nos últimos quatro anos pode-se observar o crescimento o substancial das sequencias depositadas parar os alelos HLA-A01*01. Com isso fez-se a necessidade de melhorar a performance preditiva para tal alelo. Desenvolver um sistema de rede neural baseado em múltiplas camadas, analisando diversas variáveis de hiper parâmetros para melhor sintonizar os valores de variáveis adequando ao tipo de dado proposto, desponta como uma abordagem capaz de retornar boa capacidade de distinção entre ligantes ao MHC e não ligantes. Dentre estes hiper parâmetros, o volume e tipo de dados de sequências negativas alimentado no preditor é uma variável que influencia significativamente o desempenho do preditor. Buscamos diversificar o tipo de sequencia negativa geradas por meio de distintas metodologias e em diferente proporção. Tal abordagem não é levada em consideração por inúmeros preditores disponíveis.

OBJETIVOS

Objetivo Geral

Desenvolver um preditor de epítopos baseado em *deep learning* supervisionado para identificação de epítopos ligantes para o alelo HLA-A01*01.

Objetivos Específicos

- Aplicar pipeline de filtragem das sequências depositadas no banco de dados.
- Implementar métodos para seleção de *features* que represente informação relevantes dos epítopos.
- Desenvolver e treinar a rede neural com *deep learning* em Python, atendendo a combinação de diversas variáveis e cenários distintos.
- Medir sistemas de acurácias, eficiência e eficácia para com os múltiplos cenários gerados e para com a análise comparativa realizada com outros preditores.
- Validar e comparar o desempenho do preditor com outros métodos preditivos.

MATERIAL E MÉTODOS

Amostras de estudo

O dado utilizado foi coletado pelo banco de dados do IEDB em abril de 2018 (Vita et al., 2015). Esses dados podem ser podem ser encontrados no site http://tools.iedb.org/main/datasets/. O objetivo foi obter o alelo ode HLA que tivesse uma quantidade significativa de epítopos positivamente validados. Portanto, 8.647 sequências lineares, proveniente de qualquer patógeno de origem, para HLA-A01*01 foi coletada. Para os pós processamento, foi utilizando o indicador de ligante e não ligante em uma das colunas do conjunto de dados. Somente peptídeos que satisfizessem os seguintes critérios: força de ligação abaixo de 500nm r tamanho de peptídeo contendo 9 aminoácidos. Peptídeos que contivesse aminoácidos não específicos, como por exemplo ‘X’ ou ‘B’ foram excluídos dos dados de treinamento. Ao final 3902 epítopos foram randomicamente divididos em dados de treinamento e dados de teste, 70% e 30% respectivamente.

Os valores de entrada negativos foram gerados por meio de três métodos diferentes. Os ctipos são: (1) aleatoriamente gerados utilizando a mesma frequencia de aminoácidos (1/20), (2) peptídeos negativos gerados pelo embaralhamento de posição de aminoácidos em epítopos positivos para cada um dos 3092 epítopos e (3) randomicamente gerados mas mantendo a proporção de AA encontrados na natureza, disponibilizada pelo SwissProt. Além disso, nos objetivamso determinar a quantidadede sequências negativas que poderiam potencialmente influenciar o comprotamento da rede neural. Por fim, nos decidimos investigar a predição com proporção balanceada de positivos e negativos (1:1) e desbalanceada (2:1) somente para oscenário que contemple a proporção de AA pelo Swiss-Prot.

Desenvolvimento de *features* com informações relevantes

A rede neural utiliza dados que permita a melhor interpretação dos dados de entrada, os *features*. A vantagem de empregar esta transformação dimensional é que as correlações entre a possíveis *features* podem ser avaliadas independentemente. O primeiro método é a combinação de dois aminoácidos por posição (diAA) todos combinados com todos, criando consequentemente 36 *features* por epítopo. Para cada um dos diAA, uma numeração randômica foi designada entre os valores de 0.05 e 0.95 foram gerados ser ter nenhuma correlação bioquímica.

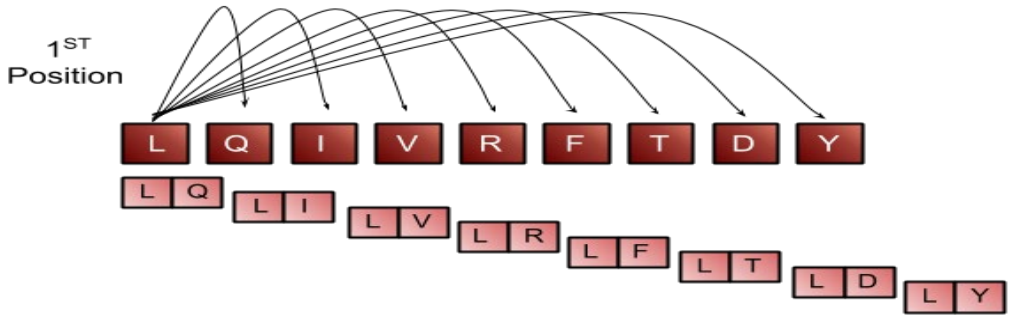

Figura 6. Esquema de análise combinatória para a primeira posição de um epítopo, gerando diAA *feature*

O segundo método é pela utilização das propriedades físico químicas descrito por Higgs (Higgs, 2009) em que é descrito nove propriedades físicas dos aminoácidos, interações de Van der Walls, volume, polaridade, ponto isoletrônico, hidrofobicidade, hidrofilicidade, área de contato, requerimento de água é perda de área durante o enovelamento proteico. As características descritas por Higgs foram transformadas por *principal componente analysis* (PCA) para atingir 90% da representatividade de variação dos dados originais, então alcançamos 4 dimensões e reduzimos os ruídos proveniente de dados desnecessários.

Arquitetura de rede neural

Toda investigação do ANN foi feita na linguagem de programação utilizando *scpits* autorais. Adicionalmente, as bibliotecas Numpy e Pandas permitiram a importação e processamento dos dados. No utilizamos a biblioteca scikit-learn (Pedregosa et al., 2011) para converter, normalizar os dados e realizar o fatiamento do dado em treinamento e teste, 70 e 30% respectivamente. Nós construímos ANN com a assistência da biblioteca Keras, que adicionou neurônios, camadas e funções de ativação. Também o Keras utiliza internamente o TensorFlow como processamento de fundo na compilação e implementação do *scprit.* Duas diferentes funções de ativação faram tuilizadas: Tangennte hiperbólica (Tanh) e unidade linear retificada (ReLu). O modelo de função de perda utilizado foi o *categorical cross-entropy*. A taxa de aprendizado foi estabelecida para evoluir em 0.05% por cada *epoch.* A arquitetura de ANN que nós propomos nesse trabalho consistem em combinar múltiplos hiper parâmetros de rede neural para determinar a melhor configuração. Finalmente, o ultimo parâmetro levado em consideração é a técnica de optimização, gradiente descente (GD), que é utilizado para encontrar os valores dos coeficientes de uma função. Em muitos casos, o GD pode ser linear ou

de regressão logística, neste experimente o segundo modelo adequa-se ao tipo de dados que temos. A avaliação de quão perto o modelo de aprendizado de máquina se adequa à função desejada pode ser calculado por diversas formas, normalmente um modelo específico para cada um dos distintos tipos de ANN. Nosso modelo utilizou batch GD, a forma mais usual de GD descrito em aprendizado de máquinas.

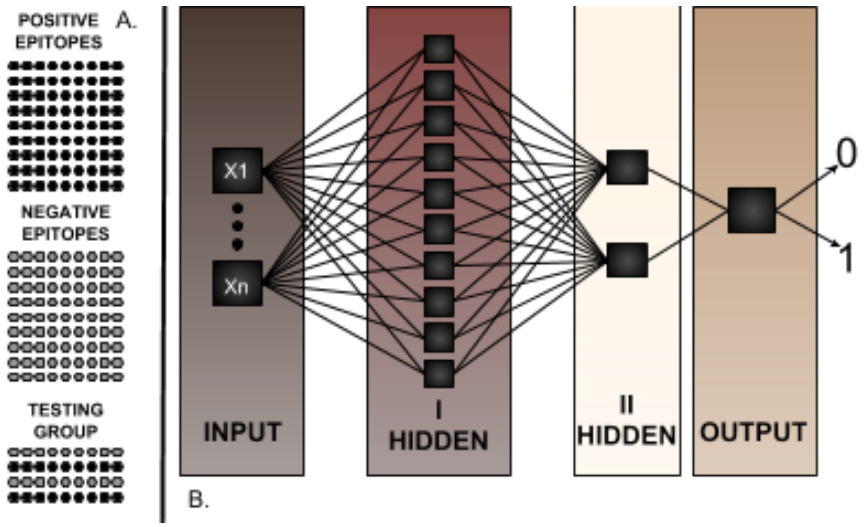

Figura 7. Representação esquemática das vias estruturais do ANN. (a) representação dos grupos de treinamento (positivos e negativos) e do grupo de teste. (b) a rede neural é formada por 3 camadas: camada de entrada, *input*, igual ao tamanho das *features* por posição de AA em um epítopo; duas camadas ocultas de nós variados e a saída binária, output, em que 0 representa não ligante e 1 representa ligante.

Optimização de Rede Neural

Os modelos de ANN testados com somente uma camada oculta nos aqui referimos como *ANN shallow model*. Posterior mente foi criado outro modelo contendo uma segunda camada oculta à rede e para esse nos iremos referir como ANN *deep model*. Então, após a camada de entrada, temos uma primeira camada densa, que é chamada de camada oculta ou *hidden layer,* pois esta não está diretamente conectada com as outras camadas. *Deep learning* é uma abordagem muito eficiente para funções de aproximação em espaços de várias dimensões (Du and Swamy, 2014). Para implementar o *perceptron* multicamadas, projetamos um modelo de duas camadas com cenários de 2 a 17 neurônios possíveis na primeira camada oculta e de 2 a 30 na segunda camada oculta.

Comparações e validações

Uma considerável variedade de preditores foram desenvolvidas na última década. No entanto, as imensas opções de ferramentas de predição fazem com que o processo de escolha não seja uma tarefa fácil. A decisão de qual preditor utilizado é baseado no desempenho da predição do algoritmo nos dados de teste. A fim de comparar o desempenho com os outros métodos de previsão, aplicamos uma comparação de desempenho para 11 preditores diferentes de MHC-I. BIMAS, EpiJen, IEDB (consenso, ANN, SMM), MHCPred, NetMHC, nHLAPred, Rankpep, ConvMHC e SYFPEITHI são preditores gratuitos disponíveis *online,* ao quais foram usados para obter a pontuação de predição (Bhasin e Raghava, 2007; Doytchinova et al. , 2006; Lundegaard e outros, 2008; Parker e outros, 1994; Reche e outros, 2004; Schuler e outros 2007, Vita e outros, 2015, Han & Kim, 2017).

O valor de corte foi importante para determinar se o epítopo candidato pode ser considerado como uma sequência epitópica variou de acordo com o tipo de saída dado de cada preditor. Para garantir que todos os dados possíveis possam ser incluídos na analise comparativa, ou *benchmark*, foram considerados três tipos diferentes de medição: classificação binária (positiva ou negativa para ligação baseada em algum limiar), IC50 (concentração inibitória para superar 50% de um ligante de referência de alta afinidade) pode aproximar KD), t1 / 2 (meia-vida de ligação). Para avaliar o desempenho do preditor e compreender melhor a matriz de confusão, *confusion matrix*, geralmente é usado um esquema de tabela que mostra as instâncias em uma classe prevista ("real" e "previsto"). Mas há modelos alternativos por meio de métricas de exatidão, precisão, recordação e pontuação F1 que fornecem uma breve explicação das métricas de confusão. As fórmulas são demonstradas abaixo:

$$\text{precision} = TP/\ TP+FP \qquad (1)$$

$$\text{recall} = TP/\ TP+FN \qquad (2)$$

$$F1 = 2TP/\ 2TP+FP+\ FN \qquad (3)$$

onde TP representa verdadeiro positivo, FP falso positivo, TN verdadeiro negativo e FN falso negativo.

Recentemente, o coeficiente de correlação de Matthews (MCC), uma medida de classificação binária, tornou-se uma maneira confiável de levar em consideração toda a matriz de confusão, mesmo que seja desequilibrada. Em outras palavras, o MCC é um coeficiente de correlação entre as classificações binárias observadas e previstas. Neste experimento, aplicamos esses métodos estatísticos para descobrir o quanto nosso modelo foi bom.

REFERÊNCIAS

ABBAS, Abul K. e LICHTMAN, Andrew H. H. e PILLAI, Shiv. **Cellular and Molecular Immunology: with STUDENT CONSULT Online Access**. [S.l: s.n.], 2014. Disponível em: <https://ebookcentral.proquest.com/lib/manchester/reader.action?docID=1772152&query=%0Ahttps://books.google.com/books?hl=en&lr=&id=RWYWBAAAQBAJ&pgis=1>.

BACKERT, Linus e KOHLBACHER, Oliver. **Immunoinformatics and epitope prediction in the age of genomic medicine**. Genome Medicine, v. 7, n. 1, p. 119, 2015. Disponível em: <http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=4654883&tool=pmcentrez&rendertype=abstract>.

BOUGHORBEL, Sabri e JARRAY, Fethi e EL-ANBARI, Mohammed. **Optimal classifier for imbalanced data using Matthews Correlation Coefficient metric**. PLOS ONE, v. 12, n. 6, p. e0177678, 2 Jun 2017. Disponível em: <https://dx.plos.org/10.1371/journal.pone.0177678>. Acesso em: 28 nov 2018.

CALIS, Jorg J A e colab. **Role of peptide processing predictions in T cell epitope identification: contribution of different prediction programs**. Immunogenetics, v. 67, n. 2, p. 85–93, 2014.

DE GROOT, Natasja G. e colab. **A Specialist Macaque MHC Class I Molecule with HLA-B*27–like Peptide-Binding Characteristics**. The Journal of Immunology, v. 199, n. 10, p. 3679–3690, 15 Nov 2017. Disponível em: <http://www.jimmunol.org/lookup/doi/10.4049/jimmunol.1700502>.

DONNES, P. e KOHLBACHER, O. **SVMHC: a server for prediction of MHC-binding peptides**. Nucleic Acids Research, v. 34, n. Web Server, p. W194–W197, 1 Jul 2006. Disponível em: <https://academic.oup.com/nar/article-lookup/doi/10.1093/nar/gkl284>. Acesso em: 26 nov 2018.

DOS SANTOS FRANCISCO, Rodrigo e colab. **HLA supertype variation across populations: new insights into the role of natural selection in the evolution of HLA-A and HLA-B polymorphisms**. Immunogenetics, v. 67, n. 11–12, p. 651–663, 12 Nov 2015. Disponível em: <http://link.springer.com/10.1007/s00251-015-0875-9>.

DOYTCHINOVA, Irini A. e FLOWER, Darren R. **Class I T-cell epitope prediction: Improvements using a combination of proteasome cleavage, TAP affinity, and MHC binding**. Molecular Immunology, v. 43, n. 13, p. 2037–2044, 2006.

GIGUÈRE, Sébastien e colab. **MHC-NP: predicting peptides naturally processed by the MHC.** Journal of immunological methods, v. 400–401, p. 30–6, 31 Dez 2013. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/24144535>. Acesso em: 8 dez 2018.

GONZÁLEZ-GALARZA, Faviel F. e colab. **Allele frequency net 2015 update: new features for HLA epitopes, KIR and disease and HLA adverse drug reaction associations**. Nucleic Acids Research, v. 43, n. D1, p. D784–D788, 28 Jan 2015. Disponível em: <http://academic.oup.com/nar/article/43/D1/D784/2438866/Allele-frequency-net-2015-update-new-features-for>. Acesso em: 9 dez 2018.

HONEYMAN, M C e colab. **Neural network-based prediction of candidate T-cell epitopes**. Nat Biotechnol, v. 16, n. 10, p. 966–969, 1998. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/9788355%5Cnhttp://www.nature.com/nbt/journal/v16/n10/pdf/nbt1098-966.pdf>.

HOOF, Ilka e colab. **NetMHCpan, a method for MHC class i binding prediction beyond humans**. Immunogenetics, v. 61, n. 1, p. 1–13, 2009.

HSING, Lianne C. e RUDENSKY, Alexander Y. **The lysosomal cysteine proteases in MHC class II antigen presentation**. Immunological Reviews, v. 207, p. 229–241, 2005.

HU, Jianjun e LIU, Zhonghao. **DeepMHC: Deep Convolutional Neural Networks for High-performance peptide-MHC Binding Affinity Prediction**. bioRxiv, p. 239236, 24 Dez 2017. Disponível em: <https://www.biorxiv.org/content/early/2017/12/24/239236>. Acesso em: 15 nov 2018.

KAROSIENE, Edita e NIELSEN, Morten e LUND, Ole. **Epitope prediction methods**. . [S.l.]: Technical University of Denmark (DTU). Disponível em: <http://orbit.dtu.dk/en/publications/epitope-prediction-methods(9dd493f7-e7ca-40b4-9416-bfc7355be3f1).html>. Acesso em: 26 nov 2018. , 2013

KELLEHER, John D. e MAC NAMEE, Brian e D'ARCY, Aoife. **Fundamentals of machine learning for predictive data analytics : algorithms, worked examples, and case studies**. [S.l: s.n.], [S.d.]. Disponível em: <https://books.google.com.br/booksFundamentals of Machine Learning for Predictive Data Analytics&f=false>. Acesso em: 4 dez 2018.

KHANKHANIAN, P-A e colab. **HLA Diversity in the 1000 Genomes Dataset**. PLoS ONE, v. 9, n. 7, p. 97282, 2014. Disponível em: <www.plosone.org>. Acesso em: 19 nov 2018.

KHANKHANIAN, Pouya e colab. **Genetic variation in the odorant receptors family 13 and the mhc loci influence mate selection in a multiple sclerosis dataset.** BMC genomics, v. 11, p. 626, 10 Nov 2010. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/21067613>. Acesso em: 20 nov 2018.

KIM, Yohan e colab. **Immune epitope database analysis resource**. Nucleic Acids Research, v. 40, n. W1, p. 1–6, 2012.

KOPRINKOVA-HRISTOVA, Petia e MLADENOV, Valeri e KASABOV, Nikola K. (Org.). **Artificial Neural Networks**. Cham: Springer International Publishing, 2015. v. 4. Disponível em: <http://link.springer.com/10.1007/978-3-319-09903-3>. Acesso em: 26 nov 2018. (Springer Series in Bio-/Neuroinformatics).

KORBER, Bette e LABUTE, Montiago e YUSIM, Karina. **Immunoinformatics comes of age**. PLoS Computational Biology, v. 2, n. 6, p. 0484–0492, 2006.

LEE, Christine K. e colab. **Development and Validation of a Deep Neural Network Model for Prediction of Postoperative In-hospital Mortality**. Anesthesiology, v. 129, n. 4, p. 649–662, 1 Out 2018. Disponível em: <http://insights.ovid.com/crossref?an=00000542-201810000-00016>. Acesso em: 29 nov 2018.

LI, Yuanhao e colab. **Regulation of transporter associated with antigen processing by phosphorylation**. Journal of Biological Chemistry, v. 275, n. 31, p. 24130–24135, 2000.

LUNDEGAARD, Claus e colab. **State of the art and challenges in sequence based T-cell**

epitope prediction. Immunome Research, v. 6, n. Suppl 2, p. S3, 3 Nov 2010. Disponível em: <http://immunome-research.biomedcentral.com/articles/10.1186/1745-7580-6-S2-S3>. Acesso em: 30 nov 2018.

LUNDEGAARD, Claus e colab. **NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8-11.** Nucleic acids research, v. 36, n. Web Server issue, p. 509–512, 2008.

LUNDEGAARD, Claus e colab. **Major histocompatibility complex class i binding predictions as a tool in epitope discovery**. Immunology, v. 130, n. 3, p. 309–318, 2010.

MAMITSUKA, H. **Predicting peptides that bind to MHC molecules using supervised learning of hidden Markov models.** Proteins, v. 33, n. 4, p. 460–74, 1998. Disponível em: <http://www.ncbi.nlm.nih.gov/pubmed/9849933>.

MARKOV, Peter V. e PYBUS, Oliver G. **Evolution and diversity of the human leukocyte antigen(HLA)**. Evolution, Medicine and Public Health, v. 2015, n. 1, p. 1, 1 Jan 2015. Disponível em: <https://academic.oup.com/emph/article-lookup/doi/10.1093/emph/eou033>. Acesso em: 19 nov 2018.

MEYER, Diogo e colab. **A genomic perspective on HLA evolution**. Immunogenetics, v. 70, n. 1, p. 5–27, 7 Jan 2018. Disponível em: <http://link.springer.com/10.1007/s00251-017-1017-3>. Acesso em: 22 nov 2018.

PATRONOV, A. e DOYTCHINOVA, I. **T-cell epitope vaccine design by immunoinformatics**. Open Biology, v. 3, n. 1, p. 120139–120139, 2013. Disponível em: <http://rsob.royalsocietypublishing.org/cgi/doi/10.1098/rsob.120139>.

PETERS, Bjoern e colab. **A Community Resource Benchmarking Predictions of Peptide Binding to MHC-I Molecules**. PLoS Computational Biology, v. 2, n. 6, p. e65, 2006. Disponível em: <https://dx.plos.org/10.1371/journal.pcbi.0020065>. Acesso em: 10 dez 2018.

RECHE, Pedro A. e GLUTTING, John Paul e REINHERZ, Ellis L. **Prediction of MHC class I binding peptides using profile motifs**. Human Immunology, v. 63, n. 9, p. 701–709, 2002.

REMESH, Soumya G e colab. **Peptide-induced opening of the F pocket of MHC class I**. p. 1–21, 2017.

SALES, Ana Paula e TOMARAS, Georgia D e KEPLER, Thomas B. **Improving peptide-MHC class I binding prediction for unbalanced datasets**. BMC Bioinformatics, v. 9, n. 1, p. 385, 19 Set 2008. Disponível em: <http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-9-385>. Acesso em: 6 dez 2018.

SANCHEZ-TRINCADO, Jose L. e GOMEZ-PEROSANZ, Marta e RECHE, Pedro A. **Fundamentals and Methods for T- and B-Cell Epitope Prediction**. Journal of Immunology Research, v. 2017, 2017.

SARAAV, Iti e colab. **Predicting promiscuous antigenic T cell epitopes of Mycobacterium tuberculosis mymA operon proteins binding to MHC Class I and Class II molecules**. Infection, Genetics and Evolution, v. 44, p. 182–189, 2016. Disponível em: <http://dx.doi.org/10.1016/j.meegid.2016.07.004>.

SHARON, Eilon e colab. **Genetic variation in MHC proteins is associated with T cell receptor expression biases**. Nature Genetics, v. 48, n. 9, p. 995–1002, 2016.

TOMAR, Namrata e DE, Rajat K. Immunoinformatics: A Brief Review. 1st. ed. [S.l: s.n.], 2014. p. 23–55. Disponível em: <http://link.springer.com/10.1007/978-1-4939-1115-8_3>. Acesso em: 25 nov 2018.

VITA, Randi e colab. **The immune epitope database (IEDB) 3.0**. Nucleic Acids Research, v. 43, n. D1, p. D405–D412, 28 Jan 2015. Disponível em: <http://academic.oup.com/nar/article/43/D1/D405/2439504/The-immune-epitope-database-IEDB-30>. Acesso em: 25 nov 2018.

YANG, Xingdong e YU, Xinglong. **An introduction to epitope prediction methods and software**. Reviews in medical virology, v. 19, n. 2, p. 77–96, 2009.

ZHANG, Lianming e colab. **TEPITOPEpan: Extending TEPITOPE for Peptide Binding Prediction Covering over 700 HLA-DR Molecules**. PLoS ONE, v. 7, n. 2, p. e30483, 23 Fev 2012. Disponível em: <https://dx.plos.org/10.1371/journal.pone.0030483>. Acesso em: 26 nov 2018.

ZHANG, Xue Wu. **A combination of epitope prediction and molecular docking allows for good identification of MHC class i restricted T-cell epitopes**. Computational Biology and Chemistry, v. 45, p. 30–35, 2013. Disponível em: <http://dx.doi.org/10.1016/j.compbiolchem.2013.03.003>.

Biblioteca Universitária Campus Anísio Teixeira – UFBA

Ferraz, Daniel Silva
Desenvolvimento de preditor de epítopo para MHC-I ligante de HLA-A01*01 por *deep learning* / Daniel Silva Ferraz.- 2018.
68 f.: il.

Orientador: Prof. Dr. Sávio Torres de Farias.
Co-orientador: Prof. Dr. Leandro Martins Freitas.
Dissertação (Mestrado) – Universidade Federal da Bahia, Instituto Multidisciplinar em Saúde, Programa de Pós-Graduação em Biociências. 2018.

1. Predição de epítopos. 2. Rede neural. 3. Aprendizado de máquinasI. Universidade Federal da Bahia. II. Instituto Multidisciplinar em Saúde. III. Farias, Sávio Torres de. IV. Freitas, Leandro Martins. V.Título

CDU: 004:573.6

CAPÍTULO 1

Trabalho para ser enviado à revista **Journal of Theoretical Biology / ISSN: 0022-5193**

Title: MHC-I Epitope predictor for HLA-A01*01 through deep learning approach

Full names and full postal addresses of all Authors,
Authors: Daniel Silva Ferraz[1]; Daniele Santoni[3]; Leandro Martins de Freitas[1]*; Sávio Torres de Farias[2];

Filiation:

1 Núcleo de Biointegração, Instituto Multidisciplinar em Saúde, Universidade Federal da Bahia, Rua Rio de Contas, 58, Vitória da Conquista, BA, 45029-094, Brazil

2 Departamento de Biologia Molecular, DBIM - CCEN - Departamento de Biologia Molecular Universidade Federal da Paraíba, João Pessoa, PB, 58059-900, Brazil

3 Istituto di Analisi dei Sistemi ed Informatica "Antonio Ruberti", Via dei Taurini, 19, 00185 Roma, Italy *Correspondence: leandromartins@ufba.br

Abstract

Peptides derived from endogenous antigens can bind to MHC-I molecules. Those which bind with high affinity can invoke a $CD8^+$ immune response, resulting in the destruction of infected cells. Much work in immunoinformatics has involved the algorithmic prediction of peptide binding affinity to various MHC-I alleles. A number of tools for MHC-I binding prediction have been developed, many of which are available on the web. However, many machine learning-based methods are generally less sensitive in recognizing locally-clustered interactions, which can synergistically stabilize peptide binding. Here we propose the EPI predictor, an artificial neural network-based for HLA-A*0101 allele. Our ANN was built using 3902 epitopes for HLA-A*0101 and nine physical-chemical properties. Then the physical-chemical properties were reduced to four PCA and used as features for each amino acid in the train and test peptides. We train/test the model using 70/30% of the epitopes retrieved from the database respectively. Our model reached a good prediction level with an accuracy of 77.09 and f1 score of 0.77 in the test set. Additionally, several hyperparameters test were done to identify the most suitable multilayer neural network. In addition to the validation testing protocol, we sought to compare the accuracy rate performed by our predictor in

comparison to several other predictive approaches available online. The benchmark demonstrated EPI displaying an outstanding performance in contrast to some reliables MHC-I predictors. The EPI reached 0.53 MCC score, indicating consistent predictive capacity. The strategy to build the EPI and performance compared with other predictors are discussed.

Keywords: epitope prediction, artificial neural network, machine learning

1. Introduction

The recognition of infected cells by cytotoxic $CD8^+$ T cell is mediated by the major histocompatibility complex I (MHC-I) proteins that are located on the cell surface of each nucleated cell. In humans, the set of genes in charge of encoding the MHC proteins are the human leukocyte antigens (HLA). HLA is the most polymorphic gene because this complex immune system needs to fine-tune detect a wide range of foreign peptides and present them correctly to the T cell receptor (dos Santos Francisco et al., 2015). MHC-I molecules bind peptides processed in the cytosol, which means that the antigen processing and presentation are generated endogenously through proteolysis (Hu and Liu, 2017; Patronov and Doytchinova, 2013). This enables the host immune machinery to target the cells infected by parasites resident in the cytoplasm, ending up in cytotoxic $CD8^+$ T cell activation. As consequence, several cytotoxins are then released, triggering a cascade of events leading to cell apoptosis (perforin/granzyme cell death pathway) (Paust et al., 2010).

As the HLA I is highly specific, they are capable of bind small fraction of peptides (epitopes) available through the antigen presenting pathway. Among some crucial peptide processing steps, such as degradation of proteins marked by ubiquitination, enzyme digestion and peptide translocation to the lumen of reticulum endoplasmatic, the specificity to the MHC binding site is the most important (Tomar and De, 2014). On account of this, accurately predict epitopes that bind to well-known HLA alleles may have significant medical value in terms of immunotherapy and to design more effective protein subunit-based vaccines (Vang and Xie, 2017).

Given the main role of MHC-I to the immune response, several *in silico* methods have been developed to either predict the binding affinity or if the peptide can bind or not the MHC pocket region. The availability of large-scale epitopes sequences stored on online databases with the curated data set, e.g. IEDB, has allowed machine learning approach to reach high accuracy level (Vita et al., 2015). Among many possibilities, Artificial Neural Network (ANN), with many hidden layers, can handle mutual information from non-anchor epitope position through high order correlation. In particular, the good performance of ANNs in many epitope predictors was verified (Lata et al., 2007; Lundegaard et al., 2008; Yang and Yu, 2009).

Hu and Liu reported (Hu and Liu, 2017) that there are still several obstacles an efficient epitope predictor must face. The huge HLA allelic variation, unbalanced epitopes database, peptide size preference, common affinity threshold, and nonlinear sequences are the challenges that need to be overcome (Lundegaard et al., 2010). The correct neural network architecture can perfectly deal with some of these difficulties. For instance, learning methods, which are conventionally divided into supervised, unsupervised and reinforcement learning, can effectively serve to different purposes. Supervised approaches, widely used in classification, adjust the network parameters by direct comparison between the actual and expected output (Kelleher et al., 2015). Many other variables play an important role in network performance. Learning rate, activation function, epoch number, batch size, hidden layer number, number of neurons per layer, and methods to update the weights of each neuron, are usually a set of scenarios that can be changed to build an ANN that better classify the data that is being processed.

Whereas adding new experimentally positive data depends on the increase of research on this field, which might take time and a massive financial investment, ANN designing may allow reaching near-optimal architecture without getting enormous training data (Smithson et al., 2016). Concerning epitope prediction, feature selection has been used to determine whether an amino acid (AA) or motif might set a role in the prediction. Feature selection is to select the best subset of the features out of the original set since irrelevant features create noise in the data and worsen the performance (Du and Swamy, 2014). Our predictor aimed to set the features from the amino acid distance matrix based on nine physicochemical characteristics (Higgs, 2009). Moreover, this project goal is to develop a supervised MHC-I epitope predictor for HLA-A*0101 allele using ANN with deep learning, so-called Epitope Predictor Initiative (EPI). Regarding the hyper-parameter several benchmarking were done to identify the most adequate. Here we seek to provide a complimentary benchmarking across open sources online predictors in order to cross-compare different predictive techniques for MHC-I epitopes.

We used 3902 positive epitopes for HLA-A*0101 collected from IEDB and we used 70% of these epitopes in the ANN training. Different ANN architectures were tested by varying the number of neurons in the two hidden layers. The best architecture was selected based on the classification performance of the test dataset (30%). The EPI presented a good classification performance and the final outcome was 77.23% of F1-score in the test dataset. The EPI showed a good performance in the classification of 9mers epitopes and the prediction of HLA-A*0101 peptide binding is discussed.

2. Material and methods

2.1 **Dataset**

The dataset used was collected from the IEDB database (Vita et al., 2015), in April 2018. These data can be found at http://tools.iedb.org/main/datasets/. The aim was to obtain an HLA allele that contained a significant amount of experimentally curated as positive epitopes. Therefore, 8.647 linear sequences, from any kind of disease, for HLA-A*0101 were retrieved. For a data post-processing, target indicating binding or non-binding was readily given as one of the columns in the processed data-set. Only under 500 nM and 9 AA epitope length where chosen to be part of our filtered training dataset. Peptides containing non-specific amino acids, e.g. 'X' or 'B', were excluded from the training dataset. The final dataset (3902 epitopes) was randomly split into 70% training set and 30% validation set.

The negative input where then generated through three different methods. The negative sets are: (1) random negative epitope using all amino acids with the same frequency (1/20), (2) negative peptides generated from the 3902 positive epitopes performing a random position shuffling in each peptide, and (3) randomly generated negative sequences maintaining the AA ratio, as available on SwissProt. Besides that, we intended to determine whether the negative sequences quantity would positively influence the neural network performance. Thus, we decide to investigate balanced (1:1) and unbalanced (2:1) negative and positive proportion only for the random epitopes generated with the swiss-prot amino acid frequency.

2.2 **Amino acids physicochemical principal component analysis and di-aminoacids features**

Feature extraction is usually conducted by using orthogonal transformation, in which the features are used for pattern understanding and analysis. The advantage of employing an orthogonal transformation is that the correlations among the candidate features are decomposed so that the significance of the individual features can be evaluated independently. The study was conducted to check two possible features source. The first method is the combination of 2 amino acids per position (diAA), all against all, thus creating 36 features for each epitope. For each of the diAA, a random label number between 0.05 and 0.95 was generated (tag) without any biochemical relation.

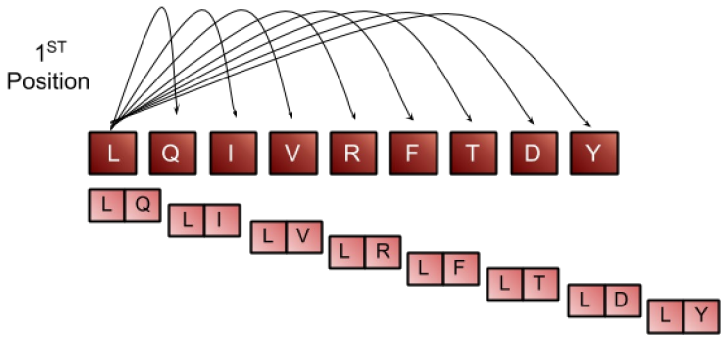

Figure 1. diAA Scheme depicting the combinatorial analysis of an epitope at the first position.

The second method is the amino acid physical properties described by Higgs (Higgs, 2009) where he described nine features by each amino acid; van der Waals radii, bulkiness, polarity, isoelectronic point, hydrophobicity, hydrophilic, the surface area accessible to water, polar requirement and lost area during the protein folding (Higgs, 2009). The Higgs features were transformed by principal component analysis (PCA) to reach 90% of the variation of the original features, thus reaching four dimensions and reducing the noise in the features, PC4.

2.3 **Artificial neural network script**

All the ANN investigation was implemented in Python 3.6 program language using in-house scripts. Additionally, NumPy and Pandas libraries allowed the data to be imported, sliced and processed. We used the scikit-learn (Pedregosa et al., 2011) library to converts, normalize the data, and perform the train/test split in 70/30% respectively. We built the ANN with the assistance of Keras library, that built the ANN adding neurons, layers, and activation functions. Keras is internally using TensorFlow as a backend system for computation. Two different activation function was used: Hyperbolic tangent (Tanh) and Rectified Linear Unit (ReLu). The loss function model was the categorical cross-entropy. The learning rate was established to evolve at 0.05 percent each epoch. The ANN architecture we propose in this paper consists of multiple combined hyper-parameters variation to find the most adequate configuration to predict accurately epitopes[1] . Finally, the last parameter was the optimization technique, gradient descent (GD), that is used to find the values of coefficients of a function that minimizes a cost function. In many cases, the GD is Linear or Logistic regression and the evaluation of how close a fit a machine learning model estimates the target function can be calculated a number of different ways, often specific to the machine learning algorithm. Our predictor used Batch GD, which is the most common form of gradient descent described in machine learning.

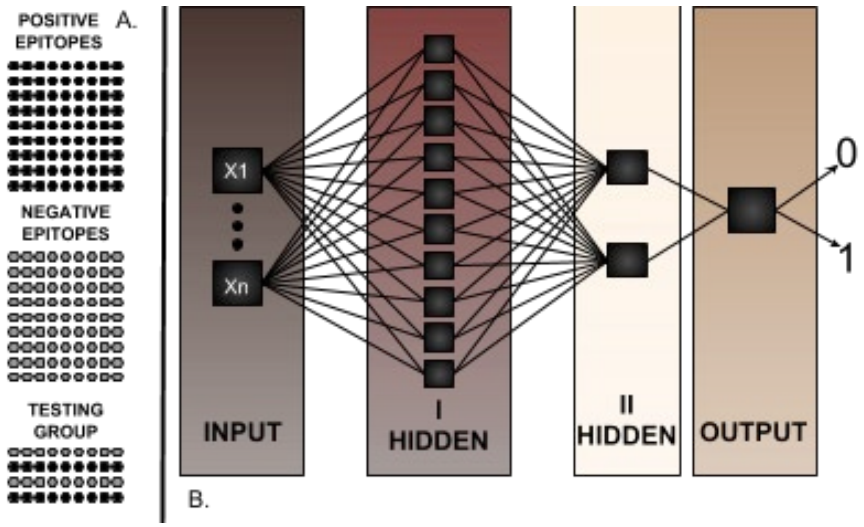

Figure 2. Schematic representation of the overall training process of the ANN. (a) a representation of the training group (positive and negative) and the testing group. (b) the neural network is composed the input data of n size equal to the features, two hidden layers of variable nodes and depth and the binary output layer, in which 0 stand for nonbinding and 1 for binding.

2.4 **Artificial network tuning**

The models tested were ANN with one layer of each type (input, hidden, and output) and we refer to this model as ANN shallow model. Later we created other models with the addition of a second hidden layer to the ANN network, and we refer to these collections of models as ANN deep models. So, after the input layer, we have a first dense layer, which is called the hidden layer because it is not directly connected to either the input or output data. Deep learning is very efficient for function approximation in high-dimensional spaces (Du and Swamy, 2014). In order to implement the multilayer perceptron, we designed a model of two layers with scenarios from 2 to 17 possibles neurons on the first hidden layer and from 2 to 30 on the second hidden layer.

2.5 **Benchmarking and Validation**

A considerable variety of predictors has been developed over the last decade. However, the tremendous options of predictions tools make the choosing process not an easy task. The decision of which predictor to used is based on the performance of the algorithm prediction on test data. In order to compare the performance with the others predicting methods, we applied a performance comparison to 11 different MHC-I predictors. BIMAS, EpiJen, IEDB (consensus, ANN, SMM), MHCPred, NetMHC, nHLAPred,

Rankpep, ConvMHC and SYFPEITHI are the free source, online predictors that were used to get the prediction score (Bhasin and Raghava, 2007; Doytchinova et al., 2006; Lundegaard et al., 2008; Parker et al., 1994; Reche et al., 2004; Schuler et al., 2007; Vita et al., 2015, Han & Kim, 2017).[1]

The cut-off value was important to determine if the candidate epitope might be considered as an epitopic sequence varied according to the given output type of each predictor. To ensure that all possible data can be included in the benchmark, three different measurement types were considered: binary classification (positive or negative for binding based on some threshold), IC50 (inhibitory concentration to outcompete 50% of a high-affinity reference ligand, can approximate KD), t1/2 (half-life of binding). To evaluate the predictor's performance, and better understand the confusion matrix, a table layout that shows the instances in a predicted class ("actual" and "predicted"), is generally used. But, there are alternative models via accuracy, precision, recall, and F1 score metrics that provides a brief explanation of the confusion metrics. The formulas are demonstrated below:

$$precision = TP/\ TP + FP \quad (1)$$

$$recall = TP/\ TP + FN \quad (2)$$

$$F1 = 2TP/\ 2TP + FP + FN \quad (3)$$

where TP represent true positive, FP false positive, TN true negative, and FN false negative.

Recently, Matthews correlation coefficient (MCC), a binary classification measurement, has become a reliable way to take into account the whole confusion matrix, even if it is unbalanced. In other words, MCC is a correlation coefficient between the observed and predicted binary classifications. In this experiment, we applied these statistical methods to figure out how good our model has performed.

3. Results

3.1 **Dataset filtering**

Developing a machine learning algorithm ultimately means building a model that outputs correct information (classification) given a provided input data. The data have to be the scope best representation and to do so, we proposed to select peptide structures that increase training value for the neural network, the majority amount of sequence downloaded by IEDB had 9 amino acids length. T cell epitopes presented by MHC-I molecules are typically peptides between 8 and 11 amino acids in length. However, as described by the literature, 9 is the optimal sequence size to fit correctly on the MHC pocket size (Giguère, 2013).

[1] The batch (number of samples to train the ANN) and epoch (number of steps to trains the ANN) were 128 and 100 respectively.

After applying the filters for epitope selection, we checked that 3,902 epitopes fulfilled all the requirements (table 1).

Table 1. Positive epitopes sequence through filtering steps

Data Filtering Steps	**Total Amount**
Raw IEDB[2] HLA-A01*01	8.647
Non-specific AA exclusion	8.212
Selecting 9 size-length epitope	4.046
Lower than IC50 500nM	3.902

Anchor regions are few peptides residues that bind to specific pockets on the MHC-I, resulting in higher specificity of interactions between the epitope and the protein. Each HLA allele varies from their amino acid and position preferences, and it can be related to the genetic variability. This diversity arises because of mutations but also more frequently due to mechanisms such as gene conversion (Dendrou et al., 2018). We investigated the epitope sequence to check the amino acid preferably used in some positions indicating the anchor position of the HLA-A*0101. The epitope logoplot was done with WebLogo, a web-based application designed to make the generation of sequence logos (Crooks et al., 2004). We noticed a week anchor for HLA-A01*01, binding preferentially to tyrosine, leucine, and phenylalanine (fig. 3). The other eight positions are not anchored because they don't have a set of amino acids more frequent, although there is some variation in the use of the amino acid in each position. We compared the epitope logo with the non-epitope logos from the negative data generated with an amino acid frequency of 1/20, and also using the shuffle sequence. Instead of positions with different amino acid sequence as it was noticed in the epitope logo, it is possible to check there is no preference for any amino acid in the non-epitope data, showing that these data have different information stored and are suitable for ANN training.

3.2 **ANN architecture design**

The selection of the features to train the model influence the prediction outcome and must be avoided the features introducing noise in the network. In order to improve the model's performance, the feature selection must be taken considering if the features are informative, creating a good model, or noise

[2] The data are a collection of experimentally validated and published sequences

in the data, what would mislead to the wrong conclusion. The poor generalization capability of the machine learning was solved by removing irrelevant noise information in the features from the set using PCA. The PCA was then applied in the amino acid physicochemical properties to reduce the dimensions of nine dimensions dataset by projecting it onto a (k)-dimensional subspace. We, therefore, decided to select 90% of the variability which represents the four principal components (PC) as the ideal k dimensions of the AA features (table 2). These four new PC were used in the next analysis as amino acid features in the input layer.

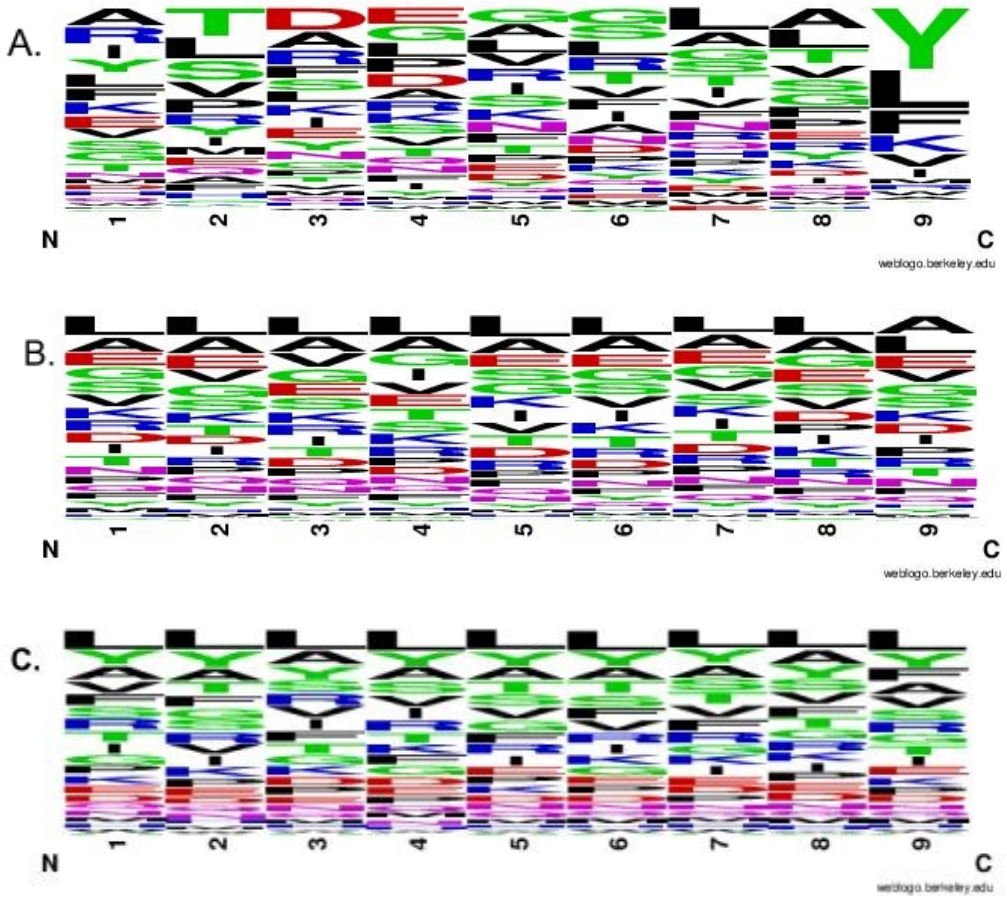

Figure 3. LogoPlot of all HLA-A01*01 retrieved from IEDB. The panels represent (a) amino acids frequency per positive epitope position, (b) the amino acids frequency per 1/20 random negative epitope position, (c) shows the amino acids frequency per shuffled random negative epitope position.

Table 2. The first four orthogonal dimensions of AA physicochemical properties

AA	PCA1	PCA2	PCA3	PCA4	AA	PCA1	PCA2	PCA3	PCA4
F	-2.438016	1.213172	-0.509091	-0.117927	H	1.397834	0.950292	0.414252	-1.164595
L	-2.470406	0.717807	-0.139582	-0.275643	Q	1.179179	0.068456	-0.219370	1.221079
I	-2.708896	0.743711	-0.153541	-0.499185	N	1.573644	-0.950327	-0.234802	0.917882
M	-1.876821	0.545397	-0.118477	-0.093121	K	3.240673	1.981050	1.274011	-0.007918
V	-2.416792	0.070693	-0.107720	-0.530033	D	3.270095	-1.222021	-2.104670	-0.577588
S	0.283396	-2.201712	0.577130	0.407465	E	2.997796	-0.214947	-2.084278	-0.372672
P	-0.035038	-0.709882	0.467241	0.897819	C	-1.995068	-1.316844	0.035654	-0.796540
T	-0.434167	-0.806749	0.411764	0.517319	W	-1.657604	2.663225	-0.658211	0.660523
A	-0.693904	-2.236960	0.683072	-0.242361	R	3.198809	2.908758	1.665135	-0.551287
Y	-0.793069	1.485350	-0.360689	0.844420	G	0.378354	-3.688470	1.162174	-0.237636

The table lists the 4-dimensions space distributed representation of amino acid with Physicochemical interpretation. Each number stands for the coefficients of linear combination.

Once the model was implemented and properly trained, some metrics were applied to observe the progress in the learning after each epoch. The Keras provides the capability to register callbacks when training an ANN model, recording metrics for each epoch during the training. This includes the loss and the accuracy, for epitope classification, as well as the validation loss and accuracy. We observed that in our shallow ANN model during the training the value of accuracy in the batch and data validation quickly increased, while the loss decreased in the very first steps for both batch and validation (fig. 4). The deep ANN model has the same behavior of the shallow model, quickly increase accuracy and decrease loss. However, the deep ANN model reached higher values of accuracy and lower levels of loss.

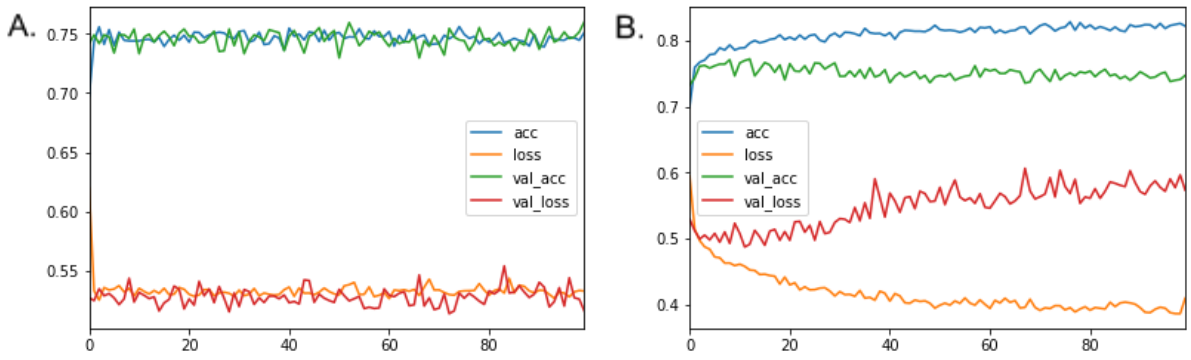

Figure 4. The plot of model accuracy and loss on train and validation datasets. The panels (a) shows the ANN shallow model, and (b) the ANN deep model. The X-axis gives the number of epochs and Y-axis the accuracy and

loss values˙ The lines represent the accuracy in the train (blue) and validation (green) data, and also the loss in the train (orange) and validation (red) data.

The metrics resulted from the training present an indication and useful information about the ANN model. One of this information is the slope or how fast it covers the model over epochs. Another handy fact that can be extracted from these analyses is whether the model may have already converged in a plateau or whether the model may be inflecting the training data, that is over-leading for validation. Figure 4 was done with the best feature set as described in table 4. It can be seen that in the shallow ANN model the accuracy raises to 0.75 on the first few epochs, but it cannot evolve over the time, reaching the plateau. Meanwhile, the validation accuracy not yet over-learned the training dataset, showing comparable skill on both datasets (fig. 4a). The ANN shallow model showed the worst performance compared with the ANN deep model (fig. 4a). However, the accuracy in the train and validation set does not share the same accuracy after some epochs. We can see that the model has not a comparable performance on both train and validation data (fig. 4b). Whenever they start to depart consistently, it might be a sign to stop training at an earlier epoch. If the model is overfitting the graph would have shown great performance on the training data and poor performance on the test data.

In an effort to reduce overfitting, improve accuracy and reduce training time, we performed a feature selection test. For each negative input model, two feature strategies were utilized, PC4 and diAA. Independent of the non-epitope data, it was found no statistical difference between the PC4 and diAA accuracy distribution, yet the PC4 shown higher mean accuracy score than diAA. The results showed that depending on the type and quantity of negative data there might be different results (Fig. 5). When the ANN architecture was based on the set of negative peptides with a higher number of sequences compared with true epitopes, it presented better accuracy score on the testing data. Regarding shuffling positive epitopes positions and randomly generating sequences, they have proven low ability to correctly predict positive epitopes, indicating that the ANN has difficulties to separate the true epitopes from shuffled peptides.

The activation function was evaluated through two experimental functions, Tahn and ReLu. The activation function simply takes into account the weighted sum of the input and then decide if the neuron should trigger or not a signal transmission. As we can see on the formula below, when Z is above a certain threshold it activates a neuron to make a connection (fire). Tahn and ReLu are nonlinear in nature which gives a smooth gradient that may help a good classifier. Tahn, as a scaled sigmoid function, has a strong gradient that gives a steep derivative (Naire et al, 2010). It tends to bring the activations to either side of the curve, making clear distinctions on prediction (Krizhevsky et al., 2012). ReLu is less computationally expensive than Tanh and sigmoid because it involves simpler mathematical operations.

The range of ReLu is from 0 to infinite, so it gives 0 to negative values of x.

Figure 5. Accuracy measurements in the different types of features and negative peptides. The diAA represents the measurements done using features created by the combination of every two pairs of amino acid, while the PC4 represents the four first principal components in the PCA. The bars measure the accuracy of each test using different features and type of negative peptides. The negative peptides are: ⅔ swiss-prot representing the peptides produced with the swiss-prot amino acid frequency and number of peptides is twice the number of epitopes (light blue); random peptides produced by equal amino acid frequency (blue); shuffled of peptides produced on the epitope sequence (light green), and swiss-prot representing random peptides produced with the swiss-prot amino acid frequency (green).

Table 3. Activation functions performance (Tanh and ReLu) and feature's highest test score. The experiment was split into 16 unique scenarios and then assessed with accuracy. For each combination of activation function and feature type, four different type of negative input were computed.

	Tanh		**ReLu**	
Negative Input	**PC4**	**DiAA**	**PC4**	**DiAA**
2:1 Swiss Prot freq.	75.75	72.25	81.12	78.02
Even Swiss Prot freq.	75.36	75.13	77.23	74.12
Shuffled	69.77	68.66	72.5	69.8
Random	73.14	71.20	77.09	72.67

In order to understand which activation function fits better on our given data, we have implemented both, Tahn and ReLu, in EPI predictor. The highest predicting accuracy was the PC4 feature with the ReLu activation function scenario (table 4). Nonetheless, it was noticed a small overfitting, which can be diagnosed in Fig. 6. The train loss slopes down and the validation loss slopes down hit an inflection point and start to slowly slope up again (Norinder and Boyer, 2017).

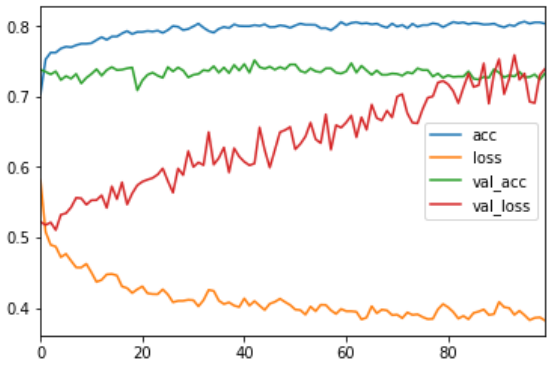

Figure 6. Model accuracy and loss on train and validation datasets on PC4 and ReLu with ⅔ negative ratio scenario. The lines represent the accuracy in the train (blue) and validation (green) data, and also the loss in the train (orange) and validation (red) data.

Therefore, we decide to use the PC4 as features and an equal positive and negative (swiss-prot frequency) input number as our ideal hyper-parameters, as shown in Fig. 4 (b). The performance of the model is good on both the train and validation sets. Also, the accuracy on the training set is considerably high, 77.28%. A summary of the EPI predictor error rate on the testing data is demonstrated in table 4. It also shows three statistical evaluation methods: precision (out of all the classes, how much we predicted correctly), recall (out of all the positive classes, how much we predicted correctly) and F1 score (it uses harmonic mean replacing arithmetic mean by punishing the extreme values more) (Han and Kim, 2017). Overall, the performance analysis revealed higher training results over testing. It was also observed that for negative peptide, was lower than the positive prediction (epitope).

Table 4. The best training and test scenario evaluated by statistical scores. They conclusive demonstrates whether a model is being trained correctly and how it may perform generally. The measurements are done separately with negative and positive data.

	Precision	Recall	F1-Score	Support
	Training			
Non-epitope	0.84	0.76	0.79	2758
Epitope	0.77	0.85	0.81	2704
Average	0.81	0.80	0.80	5462
	Test			
Non-epitope	0.78	0.74	0.76	1144
Epitope	0.76	0.81	0.78	1198
Average	0.77	0.77	0.77	2342

3.3 **Deep learning performance**

To decide what would be the ideal neuron quantity for the hidden layers, a python script was created that runs multiple models with different neurons quantity for both hidden layers. We decide to select the highest score test in a reasonable high training score (Fig. 5). According to this, there is a statistical correlation between the training and test accuracy score, that is higher is the training accuracy lower is the test efficiency.

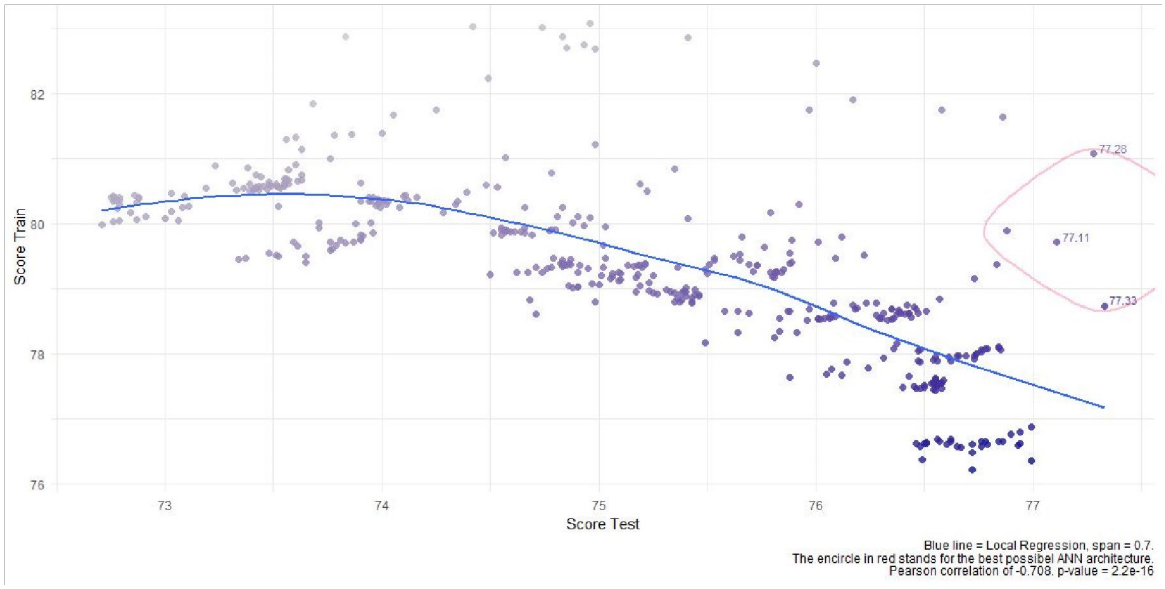

Figure 7. Accuracy performance in train and test set for the different ANN architecture. The x-axis represents the accuracy in the test set, while the y-axis represents the accuracy in the train set. The blue line represents the non-linear fitting of the scatterplot. The architecture with the best accuracy in the train set is the one with the worst test result (overfitting). The architectures with the best accuracy for train and test are highlighted by the red circle.

There are some empirically-derived rules-of-thumb, about the most commonly relied optimal size of the hidden layer, that is approximately between the size of the input and size of the output layers, but this consensus has not yet been proven. The universal formula for neurons quantity in any kind of ANN given data remains in the dark. There are some possible explanations that elucidate the number of hidden layers and neurons per each hidden layer in a classification problem. It is based on the data, draw an expected decision boundary to separate the classes. In this context, the ANN requires multiple hidden layers if and only if the data must be separated non-linearly (Koprinkova-Hristova et al., 2015). The unnecessary layer can lead to undesired network complexity. So, after performing a bulk of neuron and layers combination, the best combination was ten neurons on the first hidden layer and two on the second one (Fig 7). It represents the best accuracy for train and test which are highlighted by the red circle.

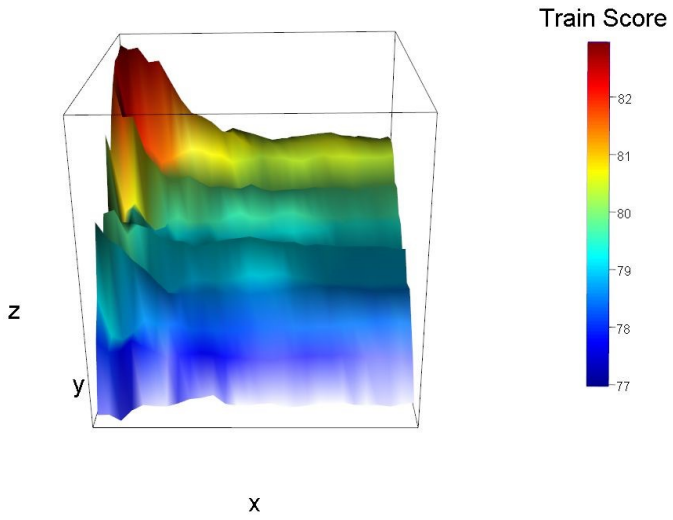

Figure 8. 3D ANN accuracy score. The X-axis is the number of neurons on the first hidden layer. Y-axis is the number of neurons no the second hidden layer. Z-axis is the accuracy value. An R package called plot 3D were used to generate this three-dimensional surface plot.

It is also possible to point out, the proportionality which accuracy increases as a function of the layers and neurons number (Fig 8). By the time the amount of neurons in the first hidden layer is increased is noticed the accuracy improvement. The same observation is pointed out on the Y-axis, that the second hidden layer neurons are incremented.

3.4 **Benchmarking and Validation**

A large set of publicly available MHC-I predictors has been benchmarked by Bhattacharya et al., (2017), Trolle et al. (2015) and H. Zhang, Lundegaard, & Nielsen (2009). The performance comparison of other predictors with EPI was done using the test data. We choose the predictors based on four criterions: (1) viability in predicting epitopes of the HLA-A 01*01 allele, (2) prediction availability via a website or standalone script, (3) ability to return measurable values within the predictive metrics, and (4) unique predictive method that differed from the others. The prediction of each algorithm showed that on average the sum of negative predicted value is lower than the overall positive. In general, the predictors tend to have the type two error lower than type I. Regarding the non-epitope, all the predictors have similar results, a high number of true negative. Although the EPI has the highest value of false positive. The prediction results in the true positive showed that EPI returned the highest number of true positive, while the other predictors returned a high amount of false negative.

Table 5. Benchmarking confusion matrix. The test data is composed by 2342 randomly chosen epitope (positive) and non-epitope (negative) data. The test data run against 11 different methods. The type of output is divided into a true positive, false negative, false positive, and true negative.

TYPE	EPI	NET MHC	BIMAS	SYFPEITHI	IEBD concesus	IEDB - ANN	IEDB - SMM	nHLAPred	Rankpep	EpiJen	MHCPred	ConvMHC
TRUE POSITIVES	983	315	131	394	150	144	129	50	212	238	279	175
FALSE NEGATIVE	215	857	1041	778	1022	1028	1043	1122	960	934	893	997
FALSE POSITIVE	341	29	16	86	7	8	5	5	15	57	89	10
TRUE NEGATIVES	803	1140	1153	1083	1162	1161	1164	1164	1154	1112	1080	1159

The confusion matrix interpretation can be better seen through statistical data processing representing the predictions in scores. Several evaluation metrics can be encountered, such as specificity, sensitivity, accuracy, precision and so forth. Thereby, recall, or the ability of a model to find all the relevant cases within a dataset by focusing on identifying the positive cases, is one one the metrics to be considered. The recall is also known as sensitivity. Our predictor, EPI displayed a fairly high sensitivity to detect positive values (Fig. 9). On the other side, the lowest in term of recall was nHLAPred, a predictor solely based on ANN which was training with only 88 positive epitopes sample (Bhasin & Raghava, 2007).

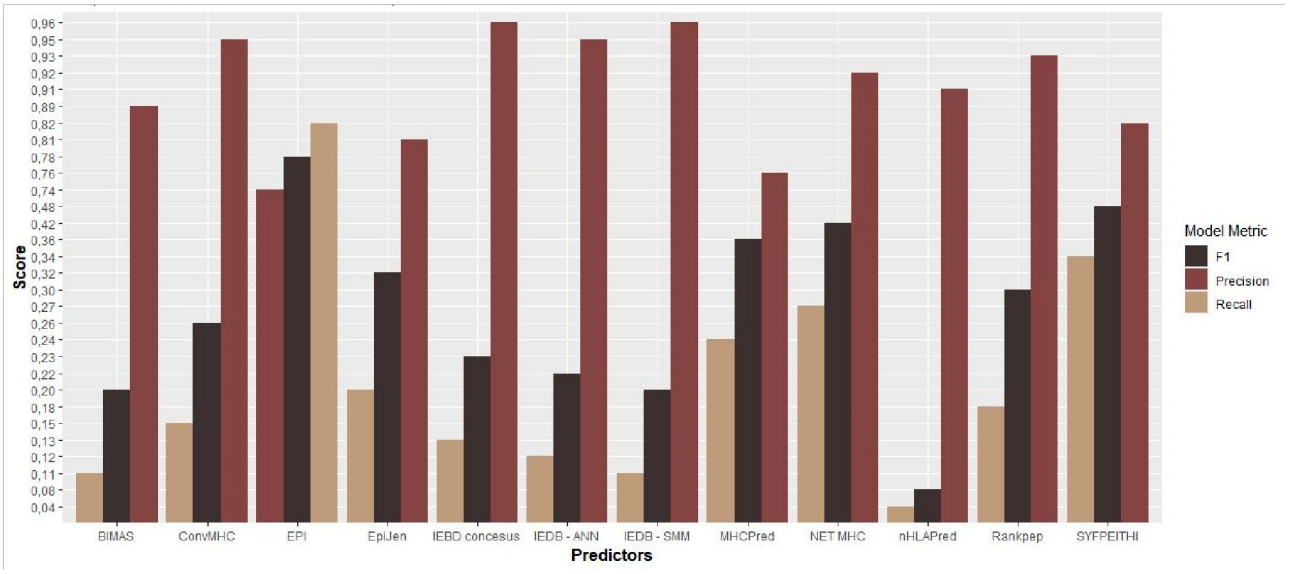

Figure 9. Predictors Accuracy comparison among 11 different predictors. It was analyzed how the predictor performance varies among different predictive models using the test data. Three statistical metrics were used: recall, also known as sensitivity, the amount of positive captured by the model; precision, strand for accuracy or how many of the positive predicted are actual tue; and F1 score, the weighted average of precision and recall.

Recall, however, may mislead to false conclusions because of its characteristic of overestimating the positive predictions. The recall and precision are inversely proportional when recall increases the precision decrease and vice versa. So precision is another metric approach to identify only the relevant data

points. It allows the possibility of verifying the false positive. All IEBD predictors methods display a remarkable high precision, while our predictor and MHCpred were not so precise by detecting epitopes that were wrongly classified as positive.

Table 6. Benchmarking predictor method summary

Predictor	Predictive Method
EPI	Deep learning ANN
NET MHC	ANN based method
BIMAS	Published coefficient tables
SYFPEITHI	Published motifs
IEBD concesus	ANN + SMM
IEDB - ANN	ANN based method
IEDB - SMM	SMM method
nHLAPred	ANN + QM
Rankpep	PSSM
EpiJen	Multi-step algorithm
MHCPred	Additive method
ConvMHC	Convolutional neural network

Other metrics are needed because of its antagonistic concept when recall increases as a consequence precision decrease and vice-versa. This condition is notoriously noticed in fig 9. In some situations, we might know that we want to maximize either recall or precision at the expense of the other metric. In some situations, maximize either recall or precision is what a good metrics measurement requires. This blending might be achieved with the F1 score, a harmonic mean of precision and recall (Han and Kim, 2017). The F1 score gives equal weight to both measures, and because of that, it punishes extreme values. Our predictor reached 0.74% of F1 score, which standout comparing to the others, unveiling a good predicting capability. But considering the great presence of false negatives in benchmarking and knowing that F1 is not recommended if you care about the TN, as these are completely ignored.

Matthews Correlation Coefficient (MCC) and Cohens Kappa are alternatives to handle highly imbalanced datasets and achieve better scores if FPs and FNs are minimal. MCC is essentially correlation coefficient value between -1 and +1 and it can be more appropriate when negatives actually mean something. A reasonable interpretation of Fig 10 is that a good choice to determine whether or not a peptide fragment can be recognized by MHC pocket site is the EPI predictor. It is accurate, specific and effective in both: predicting correctly positive and negative without considerable errors.

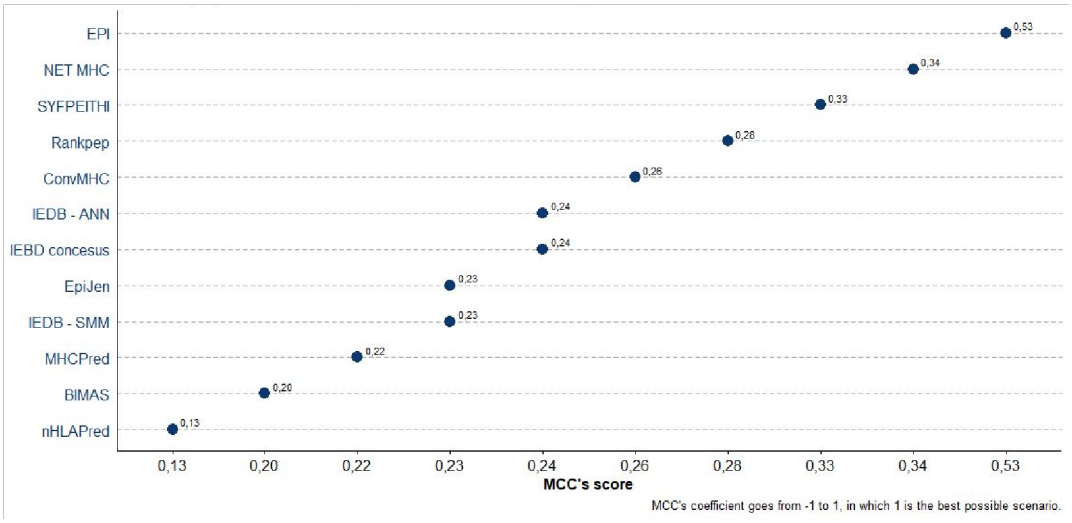

Figure 10. MHC I predictors Matthews correlation coefficient. MCC's coefficient goes from -1 to 1, in which 1 is the best possible scenario.

4. Discussion

The epitope binding prediction is critical and several immunoinformatics strategies using amino acid substitution matrices have been designed to develop predictive algorithms. As a result of the exponential amount of epitopes experimentally validated and deposited online, the predictive power based on machine learning has been increasing. For instance, in 2008 Lin et al built an algorithm to predict MHC-I binding peptide using only 176 HLA-A01*01 epitopes to training their model. Conversely, our study was able to collect 3902 positive epitopes from the immunological database. But, because of the large amino acid combining universe in nine positions, the current data still representing an insignificant portion. Neural networks are directly affected by non-representatives training set. To compensate for a shortage of data, some datasets are enriched for non-ligands, and thus highly unbalanced datasets might end up in poor performance of prediction (Sales et al., 2008). There is still no consensus on the ideal class distribution for training sets, extremely unbalanced datasets can be detrimental to the performance of prediction algorithms.

Even though, some predictor can achieve outstanding performances, all of them suffer from the fact that the available data for training is heavily biased towards one class of peptides (either binders or non-binders) (Sales et al., 2008). The main explanation for this imbalance towards non-binders is the fact that binders are extremely rare events. Therefore, when facing such disproportions designing an intelligent system that is able to overcome such bias is essential. Caution must be taken while dealing with the assumption of randomized data as negative. It can lead to a removal of essential samples or introduction of meaningless new objects. Therefore, adequating the correct amount of negative input helps to maintain structures of groups and/or generate new data according to underlying distributions (Krawczyk, 2016).

According to Norinder and Boyer, 2017, when undersampling data was used to build a model with SVM, loss of information occurs resulting in degraded performance. This is manifested by the decrease in specificity for the former models. In agreement to this affirmation, our result depicts how an unbalanced data cause raises in validation loss (fig. 5).

As an alternative strategy to mitigate common mistakes or over-optimistic results, it is to inspect different negative data source. Beside randomizing oligopeptide sequence based or not into biological amino acid frequency, randomly shuffle the dataset instance is a strategy to determine if your machine learning project will succeed (Chicco, 2017). There was no guarantee that the shuffling of the amino acids positions would provide truly non-binding sequences. Another important point is to give the importance and the uniqueness of each dataset domain, machine learning projects can succeed once the dataset details are clearly understood.

IEDB curation is of highly detailed and relies heavily on a meticulous process, checking journal articles experiments results, based on the combined use of PubMed queries, and manual inspection (Vita et al., 2015). However, an MHC binder *in vitro* does not necessarily mean it can be naturally processed and presented *in vivo*. The binding of a peptide and a MHC molecule is insufficient to ensure that the peptide will be processed to the surface of the cell. Indeed, only a small group of peptides that bind to an MHC molecule can be naturally processed (Giguère et al., 2013). Moreover, the problem has inherent noise in the data, due to the fact that a peptide can be a known binder, but not yet identified by elution. These aspects lead to a conclusion that even reliable database may contain inner error.

Nowadays, several allele-specific predictors have been developed focusing on the well-studied alleles, for example, the HLA-A and B supertype. Allele-specific preferences exist and the epitopes bind a specific class I molecule shares the same middle anchor residues but may vary in length and sequence of other residues. The HLA-A01*01 display a notorious anchor position at the ninth position (fig. 2a). This allele is frequent on Europeans and South Americans countries, but it is not available to several epitope predictors (González-Galarza et al., 2015). Tyrosine and phenylalanine are aromatic amino acids and due to their conformational configuration, binds firmly to the C-terminal region of the MHC-I groove. Any other amino acid is a low affinity at that position. Such preferences are learned during training step and storage in the neuron connections of the ANN, therefore there is no need to set any specific input weight.

Epitope size is a challenging aspect to consider. The length of binding peptides varied from 8 to 11 amino acids and most learning algorithms need to compute vector operations which are only defined for vectors of the same length. Therefore, any real-valued vector representation of a peptide that depends on its length would ultimately fail. When facing such problem, some authors have preferred to group peptides by length and to define independent learning problems for each length (Giguère, 2013). Furthermore, the current project concentrates on the prediction of 9mer peptides, as this length is the most common.

Often different feature selection approaches are used prior to ANN to feed the input variables for its models. The function of 'selected' versus 'arbitrary' features on the outcome of ANN models is investigated with a variety of objectively selected and arbitrarily chosen variables from chemical properties. The collection features, namely descriptors, for an input can be very large in term of size, but the number of critical features to predict a phenomenon is limited (Goodarzi et al., 2009). Many machine learning-based predictors only take into account the features associated with amino acids and the MHC-I molecule independently. As consequence, they could be less sensitive in recognizing the locally-clustered interactions, which could produce peptide-MHC-I binding (Han and Kim, 2017). In this perspective, often BLOSUM62 or PAM160, substitution matrices, have been preferably used over the past few years. Comparably, they do not display any significant prediction enhancement (Yao et al., 2012). Similarly, PC4 and diAA do not have significant performance difference. However, due to the fact that principal component amino acid analysis was carried out on the physical properties of amino acids measured, the overall accuracy value was higher on PC4. A possible explanation that principal component is numerically appropriately weighted reliable neural networks source (Bremel and Homan, 2010).

Concerning the activation model, Krizhevsky affirms the major advantage of ReLu is indeed non-saturation of its gradient, which greatly accelerates the convergence of stochastic gradient descent compared to the sigmoid/tanh functions (Krizhevsky et al., 2012). This provides support for our results of better performance in methods of activation via rectified linear units. Regarding the architecture configuration, the systematic experimentation is the best approach to draw a conclusion about size, depth, and width in a neural network (Krizhevsky et al., 2012). Based on the data and the expected decision boundary to separate the classes, a set of lines were expressed. The number of selected lines represents the number of hidden neurons in the first hidden layer systematic experimentation. It was numerical represented half of the total input, expressed as a combination of the features per epitope position.

Given a large number of available methods, it is difficult for the end user to judge which method is optimal for a given task. Such diversity leads to the uncertainty on determining which predictor is truly the most accurate. We acknowledge that a single summary ranking, based solely on one statistical method, will not in all situations accurately reflect performances. While setting up the benchmarking framework, we chose to include a broad set of methods. We chose three representative methods: matrix-based methods (stabilization matrix alignment method (SMM) and Quantitative matrix (QM)), neural network-based methods (ANN, CNN), and motif-based methods. The motif-based methods rely on the patterns of anchor residues, the motifs. SYFPEITHI and Rankpep, for instance, is a motif-based predictor that takes into account the position of amino acids in the peptide, such as the anchor position, unusual anchor position, and auxiliary anchor position (Schuler et al., 2007). Even though SYFPEITHI used one of the first predicting methods, its results were above the MCC score mean.

QM provides a mean for quantitative representation of qualitative data/information. For each amino acid of the peptide is assigned a coefficient value depending on the type of amino acid and its position in 9-mer (Tomar and De, 2014). ANNpred, the predictor hosted on nHLApred server, is based in QM+ANN. On our benchmark nHLApred has shown the lowest MCC score, indicating weak prediction accuracy. Besides this, the matrix-based methods (BIMAS, IEDB SMM) have also displayed an average performance. This approach does not learn by the time new data are being stored on the immunological dataset. Thus, QM, SMM, and PSSM algorithms are limited to the small set to which they were created. On the other hand, machine learning is adaptive methods that learned given by an input model.

It is possible that the ANN may recognize correctly the patterns in the test dataset, but may commit errors as well. The error committed by the ANN can be corrected, wherein the network connections would be readjusted. Most of the ANN used in this benchmark had good predictive power (table 6). Similar results were found by Trolle's benchmarking, of the six HLA-A*02:01 9mer datasets,
ANN was the top performing method for three datasets yet in the last place for the other three (Trolle, 2015). Likewise, Peters showed that ANN method outperformed the two matrix-based predictions average relative binding (ARB) and SMM, independent of the size of the training dataset (Peters et al., 2006). A common problem to consider is the negative binding data being considered not interesting enough for publication. By randomly generated peptides, most predictors immunologists may fail in accuracy. Previous studies have shown that the use of true nonbinding peptides allows for the generation of better predictions (Hertz and Yanover, 2006).

The present predictor and benchmarking are solely concerned with the prediction of peptide binding to MHC-I molecules. Binding of a peptide is a prerequisite for recognition during an immune response. We have proven good sensitivity and accuracy to predict MHC-I epitopes of alleles HLA-A01*01, thus being a good choice for determining which peptides may be potentially epitopic. We anticipate that our ANN would become more reliable in peptide binding predictions through further training and evaluations on more experimental data. However, it is not easy to build a reliable prediction model using ANN because deep learning tasks require large amounts of training data to extract high-level and generalized representations from the data.

Funding

This work was supported by CNPq fellowship from UFBA/Biociências postgraduate program. The authors declare that the funding agency had no influence in data analyses or journal selection.

Acknowledgments

We thank all authors and lab members for their contributions to the scientific work enabling the EPI predictor to become available.

References

Bhasin, M., Raghava, G.P.S., (2007). A hybrid approach for predicting promiscuous MHC class I restricted T cell epitopes. *J. Biosci*. 32, 31–42. https://doi.org/10.1007/s12038-007-0004-5

Bhattacharya, R., Sivakumar, A., Tokheim, C., Guthrie, V. B., Anagnostou, V., Velculescu, V. E., & Karchin, R. (2017). Evaluation of machine learning methods to predict peptide binding to MHC Class I proteins. *BioRxiv*. Retrieved from http://biorxiv.org/content/early/2017/07/27/154757.abstract

Bremel, R.D., Homan, E.J., (2010). An integrated approach to epitope analysis I: Dimensional reduction, visualization and prediction of MHC binding using amino acid principal components and regression approaches. *Immunome Res*. 6, 7. https://doi.org/10.1186/1745-7580-6-7

Crooks, G.E., Hon, G., Chandonia, J.-M., Brenner, S.E., (2004). WebLogo: a sequence logo generator. *Genome Res*. 14, 1188–90. https://doi.org/10.1101/gr.849004

Dos Santos Francisco, R., Buhler, S.,Nunes, J.M., Bitarello, B.D., França, G.S., Meyer, D., Sanchez-Mazas, A., (2015). HLA supertype variation across populations: new insights into the role of natural selection in the evolution of HLA-A and HLA-B polymorphisms. *Immunogenetics* 67, 651–663. https://doi.org/10.1007/s00251-015-0875-9

Doytchinova, I.A., Guan, P., Flower, D.R., (2006). EpiJen: a server for multistep T cell epitope prediction. *BMC Bioinformatics* 7, 131. https://doi.org/10.1186/1471-2105-7-131

Du, K.-L., Swamy, M.N.S., (2014). Multilayer Perceptrons: Architecture and Error Backpropagation, in: Neural Networks and Statistical Learning. *Springer London*, London, pp. 83–126. https://doi.org/10.1007/978-1-4471-5571-3_4

Giguère, S., Drouin, A., Lacoste, A., Marchand, M., Corbeil, J., Laviolette, F., (2013). MHC-NP: predicting peptides naturally processed by the MHC. *J. Immunol*. Methods 400–401, 30–6. https://doi.org/10.1016/j.jim.2013.10.003

González-Galarza, F.F., Takeshita, L.Y.C., Santos, E.J.M., Kempson, F., Maia, M.H.T., Silva, A.L.S. da, Silva, A.L.T. e, Ghattaoraya, G.S., Alfirevic, A., Jones, A.R., Middleton, D., (2015). Allele frequency net 2015 update: new features for HLA epitopes, KIR and disease and HLA adverse drug reaction associations. *Nucleic Acids Res.* 43, D784–D788. https://doi.org/10.1093/nar/gku1166

Goodarzi, M., Deshpande, S., Murugesan, V., Katti, S., Prabhakar, Y., (2009). Is Feature Selection Essential for ANN Modeling? QSAR Comb. *Sci*. 28, 1487–1499. https://doi.org/10.1002/qsar.200960074

Han, Y., Kim, D., (2017). Deep convolutional neural networks for pan-specific peptide-MHC class I binding prediction. *BMC Bioinformatics* 18, 585. https://doi.org/10.1186/s12859-017-1997-x

Hertz, T., Yanover, C., (2006). PepDist: A New Framework for Protein-Peptide Binding Prediction based on Learning Peptide Distance Functions. *BMC Bioinformatics* 7, S3. https://doi.org/10.1186/1471-2105-7-S1-S3

Higgs, P.G., (2009). A four-column theory for the origin of the genetic code: tracing the evolutionary pathways that gave rise to an optimized code. *Biol. Direct* 4, 16. https://doi.org/10.1186/1745-6150-4-16

Hu, J., Liu, Z., (2017). DeepMHC: Deep Convolutional Neural Networks for High-performance peptide-MHC Binding Affinity Prediction. *bioRxiv* 239236. https://doi.org/10.1101/239236

Kelleher, J.D., Mac Namee, B., D'Arcy, A., (2015). Fundamentals of machine learning for predictive data analytics : algorithms, worked examples, and case studies. *The MIT Press* ©.

Koprinkova-Hristova, P., Mladenov, V., Kasabov, N.K. (Eds.), (2015). Artificial Neural Networks, *Springer Series in Bio-/Neuroinformatics*. Springer International Publishing, Cham. https://doi.org/10.1007/978-3-319-09903-3

Krawczyk, B., (2016). Learning from imbalanced data: open challenges and future directions. Prog. *Artif. Intell*. 5, 221–232. https://doi.org/10.1007/s13748-016-0094-0

Krizhevsky, A., Sutskever, I., Hinton, G.E., (2012). ImageNet classification with deep convolutional neural networks. *Commun*. ACM 60, 84–90. https://doi.org/10.1145/3065386

Lata, S., Bhasin, M., Raghava, G.P.S., (2007). Application of Machine Learning Techniques in Predicting MHC Binders. *Humana Press*, pp. 201–215. https://doi.org/10.1007/978-1-60327-118-9_14

Lundegaard, C., Hoof, I., Lund, O., Nielsen, M., (2010). State of the art and challenges in sequence based T-cell epitope prediction. *Immunome Res*. 6, S3. https://doi.org/10.1186/1745-7580-6-S2-S3

Lundegaard, C., Lamberth, K., Harndahl, M., Buus, S., Lund, O., Nielsen, M., (2008). NetMHC-3.0: accurate web accessible predictions of human, mouse and monkey MHC class I affinities for peptides of length 8-11. *Nucleic Acids Res*. 36, 509–512. https://doi.org/10.1093/nar/gkn202

Nair, Vinod, and Geoffrey E. Hinton. "Rectified linear units improve restricted Boltzmann machines." (2010). *Proceedings of the 27th international conference on machine learning* (ICML-10).

Norinder, U., Boyer, S., (2017). Binary classification of imbalanced datasets using conformal prediction. *J. Mol. Graph. Model*. 72, 256–265. https://doi.org/10.1016/J.JMGM.2017.01.008

Parker, K.C., Bednarek, M.A., Coligan, J.E., (1994). Scheme for ranking potential HLA-A2 binding peptides based on independent binding of individual peptide side-chains. *J. Immunol.* 152, 163–75.

Patronov, A., Doytchinova, I., (2013). T-cell epitope vaccine design by immunoinformatics. *Open Biol.* 3, 120139–120139. https://doi.org/10.1098/rsob.120139

Paust, S., Senman, B., Von Andrian, U.H., (2010). Adaptive immune responses mediated by natural killer cells. *Immunol*. Rev. 235, 286–296. https://doi.org/10.1111/j.0105-2896.2010.00906.x

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, É., (2011). Scikit-learn: Machine Learning in Python. *J. Mach. Learn. Res.* 12, 2825–2830.

Reche, P., Glutting, J.-P., Zhang, H., Reinherz, E.,(2004). Enhancement to the RANKPEP resource for the prediction of peptide binding to MHC molecules using profiles. *Immunogenetics* 56, 405–419. https://doi.org/10.1007/s00251-004-0709-7

Sales, A.P., Tomaras, G.D., Kepler, T.B., (2008). Improving peptide-MHC class I binding prediction for unbalanced datasets. BMC Bioinformatics 9, 385. https://doi.org/10.1186/1471-2105-9-385

Schuler, M.M., Nastke, M.-D., Stevanović, S., (2007). Syfpeithi. *Immunoinformatics*, pp. 75–93. https://doi.org/10.1007/978-1-60327-118-9_5

Smithson, S.C., Yang, G., Gross, W.J., Meyer, B.H., (2016). Neural Networks Designing Neural Networks: Multi-Objective Hyper-Parameter Optimization.

Tomar, N., De, R.K., (2014). Immunoinformatics: A Brief Review. pp. 23–55. https://doi.org/10.1007/978-1-4939-1115-8_3

Trolle, T., Metushi, I. G., Greenbaum, J. A., Kim, Y., Sidney, J., Lund, O., … Nielsen, M. (2015). Automated benchmarking of peptide-MHC class I binding predictions. *Bioinformatics*, 31(13), 2174–2181. https://doi.org/10.1093/bioinformatics/btv123

Vang, Y.S., Xie, X., (2017). HLA class I binding prediction via convolutional neural networks. *Bioinformatics* 33, 2658–2665. https://doi.org/10.1093/bioinformatics/btx264

Vita, R., Overton, J.A., Greenbaum, J.A., Ponomarenko, J., Clark, J.D., Cantrell, J.R., Wheeler, D.K., Gabbard, J.L., Hix, D., Sette, A., Peters, B., (2015). The immune epitope database (IEDB) 3.0. *Nucleic Acids Res.* 43, D405–D412. https://doi.org/10.1093/nar/gku938

Yang, X., Yu, X., (2009). An introduction to epitope prediction methods and software. *Rev. Med. Virol.* 19,77–96. https://doi.org/10.1002/rmv.602

Yao, B., Zhang, L., Liang, S., Zhang, C., (2012). SVMTriP: A Method to Predict Antigenic Epitopes Using Support Vector Machine to Integrate Tri-Peptide Similarity and Propensity. *PLoS One* 7, e45152. https://doi.org/10.1371/journal.pone.0045152

Zhang, H., Lundegaard, C., Nielsen, M., (2009). Pan-specific MHC class I predictors: A benchmark of HLA class I pan-specific prediction methods. *Bioinformatics* 25, 83–89. https://doi.org/10.1093/bioinformatics/btn579

CONSIDERAÇÕES FINAIS

Este estudo abrangente uma vasta investigação das peculiaridades de um bom preditor de epítopos. Vários estudos propuseram diferentes abordagem para identificar possíveis sequências ligantes de MHC classe I. Nesse estudo o a metodologia se embasa na utilização de rede neural ANN, analisando a combinação de diversos hiperparâmetros para identificar as configurações de maior poder preditivo para o alelo HLA-A01*01 que possuam nove aminoácidos de comprimento. Embora muitos estudos que tenham desenvolvidos preditores para a peptídeos ligante ao MHC I tenham sido publicados, comparamos aqui alguns dos que tivessem propriedade metodológicas distintas da nossa, sendo possível averiguar a performance. No entanto, apesar do nosso preditor apresentar notória capacidade preditiva acima dos outros métodos., reconhecemos aqui as limitações em termos de precisão e ausência de estratificação de valores para os epítopos ao invés somente de valores binários. Um dos principais fatores que determinarão o sucesso futuro dos preditores gira em torno do aumento substancial de sequencias depositadas e curadas nos bancos de dados imunológicos .

Prof. Dr. Lucas Miranda Marques
Coordenador do Programa de Pós-graduação em
Biociências IMS/UFBA

Vitória da Conquista, 8 de Dezembro de 2018.

ANEXO I

Figura 8. Scatterplot cenário 1: *feature* PC4 com negativos gerados randomicamente. Correlação entre a performance de treinamento e teste

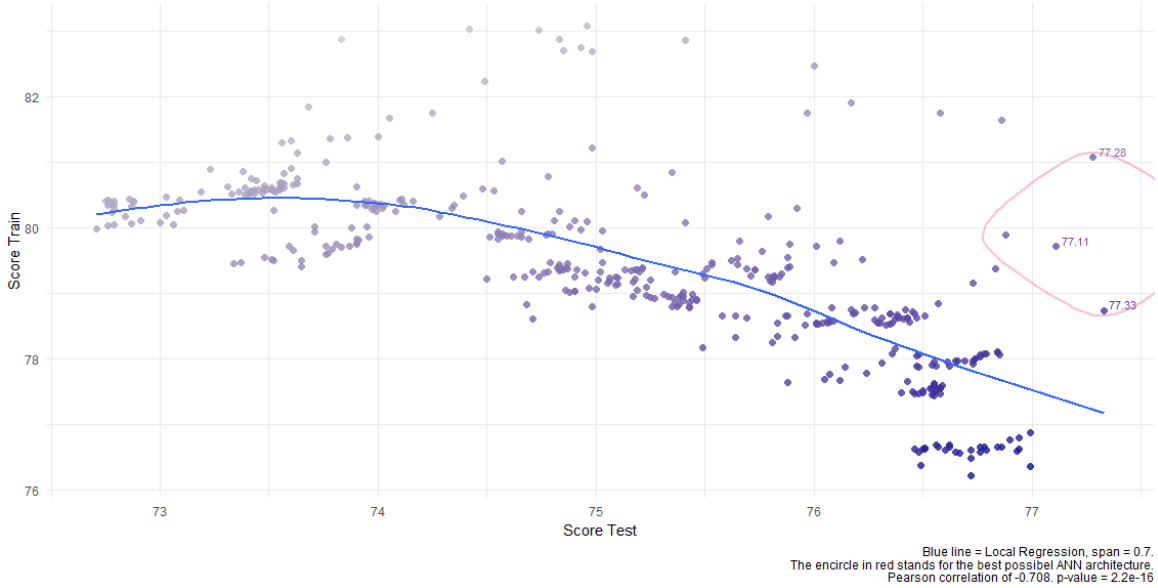

Figura 9. Scatterplot cenário 2: *feature* PC4 com negativos randômicos mantendo a proporção do Swiss-Prot em balanço. Correlação entre a performance de treinamento e teste

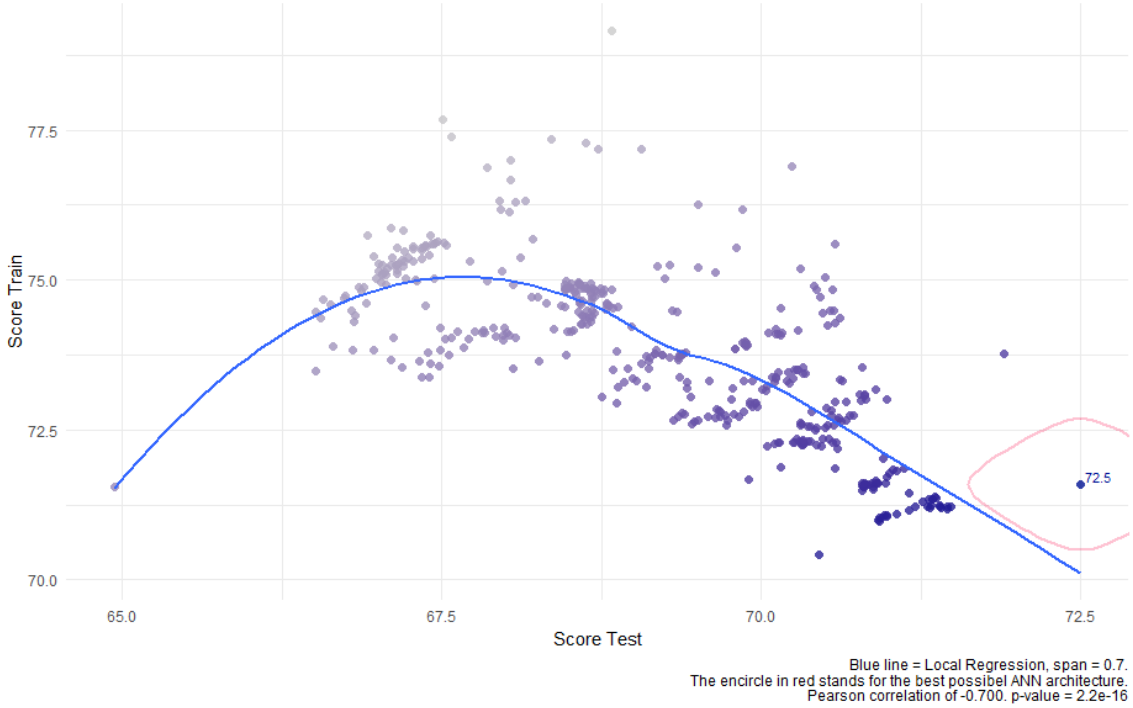

Figura 10. Scatterplot cenário 3: *feature* PC4 com negativos gerados randomicamente. Correlação entre a performance de treinamento e teste

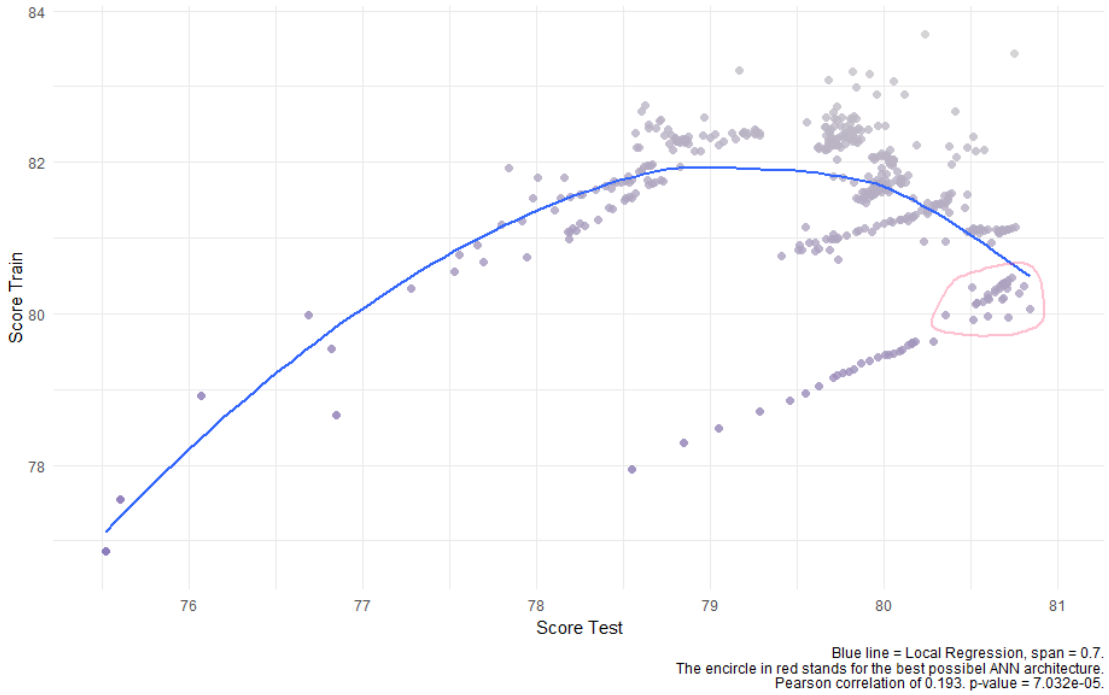

Figura 11. Scatterplot cenário 4: *feature* PC4 com negativos randômicos mantendo a proporção do Swiss-Prot em desbalanço 2/3 para negativos. Correlação entre a performance de treinamento e teste

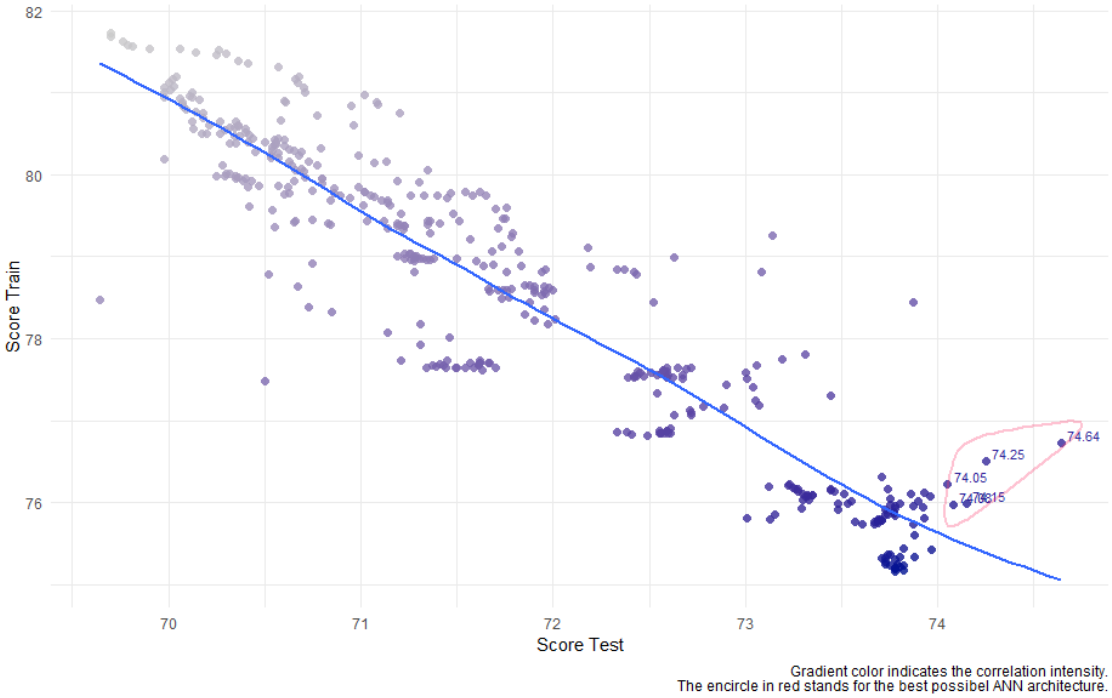

Figura 12. Scatterplot cenário 5: *feature* PC4 com negativos gerados randomicamente e função de ativação Tanh. Correlação entre a performance de treinamento e teste

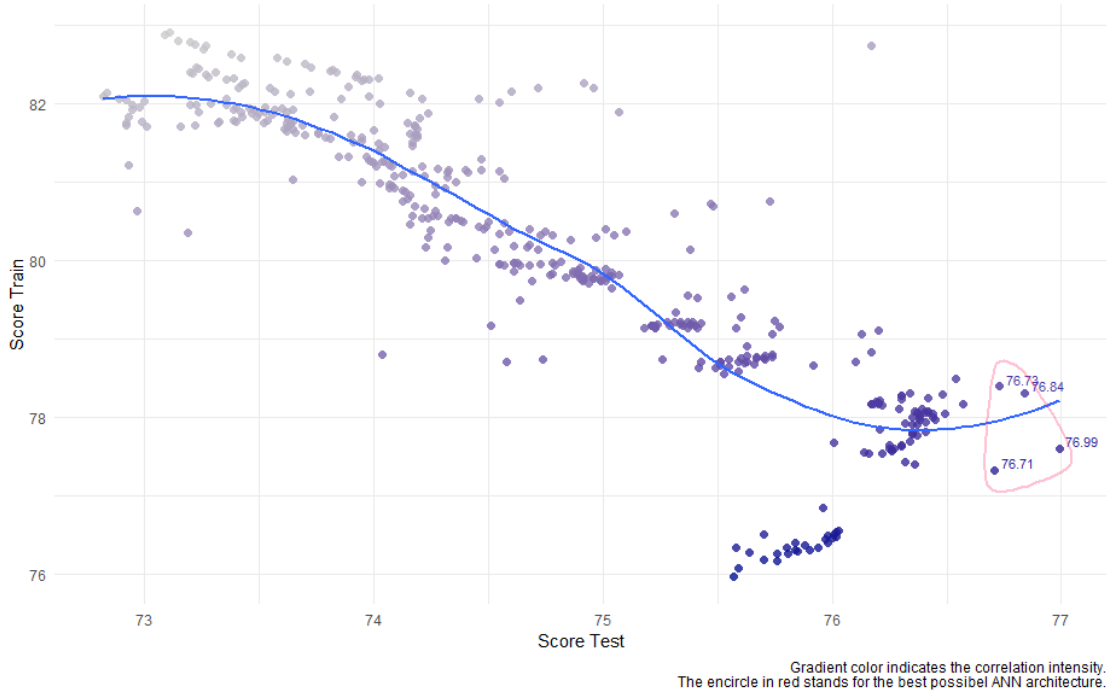

Figura 13. Scatterplot cenário 6: *feature* PC4 com negativos randômicos mantendo a proporção do Swiss-Prot em desbalanço 2/3 para negativos e função de ativação Tanh. Correlação entre a performance de treinamento e teste

Figura 14. Scatterplot cenário 7: *feature* PC4 com negativos randômicos mantendo a proporção do Swiss-Prot em balanço e função de ativação Tanh. Correlação entre a performance de treinamento e teste

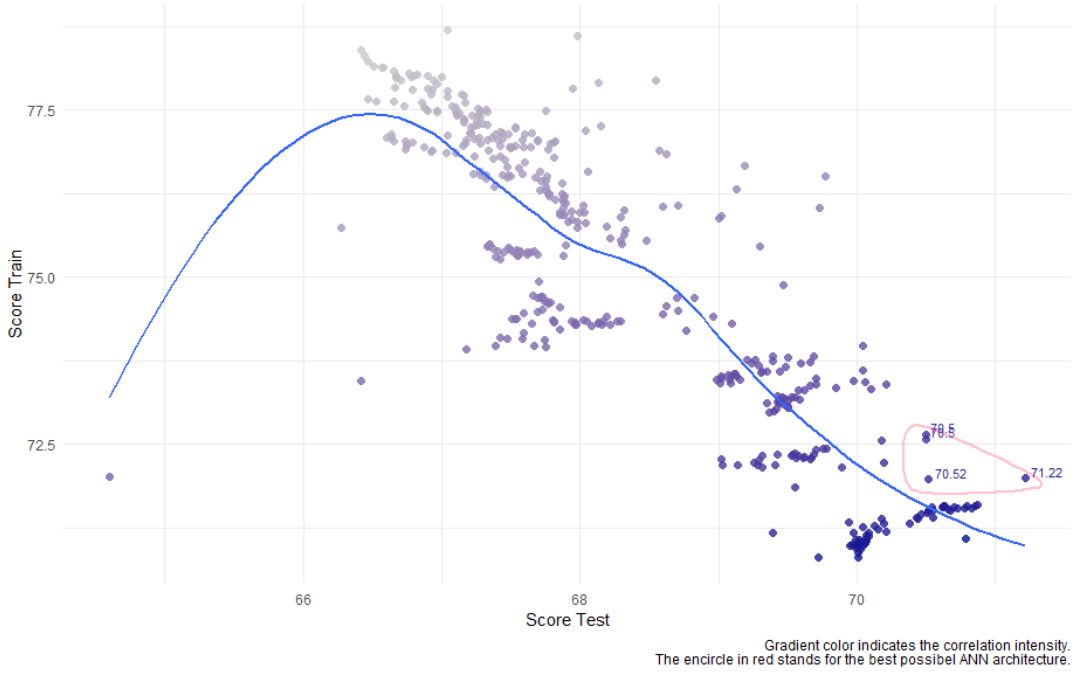

Figura 15. Scatterplot cenário 8: *feature* PC4 com negativos embaralhados e função de ativação Tanh. Correlação entre a performance de treinamento e teste