

# Variação de Parâmetros

## Esquema de Experimentação

Carlos Alberto da Silva Carvalho de Freitas, Heitor de  
Castro Felix e Vinícius Bezerra Araújo da Silva.  
{casc2, hcf2, vbas}@cin.ufpe.br

## Introdução

Esse relatório tem como objetivo apresentar o nosso plano de variação inicial dos hiperparâmetros dos classificadores utilizados no projeto da disciplina. Esse esquema representa uma experimentação inicial, podendo, portanto, sofrer alterações no decorrer do desenvolvimento do projeto.

## Metodologia

Para variação dos hiperparâmetros do MLP será utilizado a biblioteca Talos, cuja função principal é otimizar as configurações de parâmetros de modelos do Keras. Este último será utilizado em conjunto com o pacote Theano (*backend*) para a implementação de fato do classificador. Com o Talos é possível variar todo e qualquer parâmetro aceito pelo Keras na construção do modelo da rede MLP. Dentre eles estão a taxa de aprendizagem, quantidade de neurônios de uma camada, funções de ativação e a quantidade de camadas.

A variação dos hiperparâmetros da SVM será feita a partir da biblioteca Sklearn, que fornece uma classe denominada *GridSearchCV*. Essa classe permite, bem como o método explicado anteriormente, a variação dos parâmetros do classificador e a aplicação do K-fold Cross Validation para otimização.

Para testar variações de parâmetros posteriores, onde alguns parâmetros do classificador já foram definidos, será criado subgrupos de testes. Estes subgrupos terão valores diferentes para os principais parâmetros do classificador. Por exemplo: para testar a rede com outra função de ativação vamos treinar e testar em redes com os outros parâmetros já escolhidos como uma MLP com 32 neurônios na primeira camada, 7 camadas intermediárias e taxa de aprendizado 0.01. Em seguida testar a mesma função de ativação mas com os parâmetros de um grupo dados por 128 neurônios na primeira camada, 3 camadas intermediárias e taxa de aprendizagem 1,5. Iremos criar 8 grupos de parâmetros diferentes.

Os demais classificadores, incluindo o formado pela sua união (*ensemble*), serão treinados com os parâmetros padrão do método da biblioteca utilizada, podendo, mais tarde, serem submetidos a estratégias de variação de parâmetros como as descritas aqui.

## Experimentos

### 1. MLP

A variação dos parâmetros na rede MLP se dará da seguinte forma:

- quantidade de neurônios na primeira camada: 2, 32, 128 e 256;
- quantidade de camadas escondidas: 1, 3, 7 e 13;
- taxa de aprendizagem: 0.01, 0.1, 0.5, 1.0, 1.5 e 3.0;
- otimizador: SGD, RMSprop, Adam;
- funções de ativação das camadas: sigmóide, tangente hiperbólica, relu e softmax;
- tamanho dos pacotes:
- nos demais parâmetros serão utilizados valores fixos (100.000 épocas, etc).

Visto que a quantidade de permutações produzidas por esses valores é muito alta e levaria muito tempo para se treinar, algumas técnicas serão utilizadas para reduzir o escopo de experimentação. Estas são: a execução de apenas uma parcela dessas permutações escolhida de forma aleatória e a parada precoce do treinamento, que visa abortar o processo de treinamento, com determinada configuração de parâmetros, após um número fixo de épocas com resultados piores no conjunto de validação.

## 2. SVM

Para os hiperparâmetros do classificador SVM, serão usados os seguintes valores:

- parâmetro de penalidade (C): de 1 a 10;
- tipo de núcleo do algoritmo: linear e rbf;
- coeficiente gamma: auto, scale, 0.1, 0.5 e 2.0;
- a quantidade de *folds* no *cross validation* será de cinco.

## 3. Outros classificadores

Para os demais classificadores (*Random Forest*, *Gradient Boosting* e *MLP Ensemble*) os hiperparâmetros serão mantidos com os valores padrões estabelecido pelas bibliotecas Sklearn e Keras, e podem ser encontrados nos links de referência.

## Referências

- [1] Keras Documentation. <https://keras.io/>
- [2] Talos - User Manual. [https://autonomio.github.io/docs\\_talos/](https://autonomio.github.io/docs_talos/)
- [3] Theano 1.0.0 Documentation. <http://deeplearning.net/software/theano/>
- [4] Grid Search Cross Validation. Sklearn.  
[https://scikit-learn.org/stable/modules/generated/sklearn.model\\_selection.GridSearchCV.html](https://scikit-learn.org/stable/modules/generated/sklearn.model_selection.GridSearchCV.html)
- [5] Random Forest Classifier.  
<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html>
- [6] Gradient Boosting Classifier.  
<https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html>