# O Uso de Diferentes Métodos de Machine Learning na Predição da Solubilidade de Compostos via Características Estruturais

Matheus Gomes Cordeiro[1] and Abel Pinheiro de Figueiredo[2]

Universidade Federal do Ceará (UFC) - Departamento de Engenharia de Teleinformática (DETI)
60430-160, Fortaleza - Brasil

**Abstract**. O propósito desta pesquisa é analisar diferentes compostos químicos e correlacionar os seus dados com a sua respectiva solubilidade. Nós utilizamos técnicas estatísticas para realizar o pré-processamento dos dados e , posteriormente, realizamos técnicas de âmbito *regressivo* para se obter um modelo. Desta forma, procurou-se observar o relacionamento de uma estrutura química com o seu grau de solubilidade.Nessa busca o modelo que mais traduz essa relação é o linear.

## 1 Introdução

A solubilidade química, por definição, é o quanto que uma substância consegue se dissolver em um liquido. Tal conhecimento é de extrema importância para um grande número de aplicações que vão desde o processamento de minérios até a utilização adequada de medicamentos. Atrelado a solubilidade de um composto estão a possível existência de determinadas subestruturas químicas e descritores do composto (e.g o peso molecular) os quais serão analizados posteriormente.

Este trabalho se dispôs a realizar o pré-processamento adequado dos dados primeiramente investigados por **Tetko et al** [2] e **Huuskonen** [3] e a construção de modelos preditivos, promovendo assim um melhor entendimento da estruturação do conjunto de dados estudado e como as características estruturais dos compostos químicos, assim como outros descritores, se relacionam com a solubilidade do composto. Assim, se faz necessário o estudo de diferentes modelos preditivos, averiguando o que melhor traduz essa relação.

## 2 Metodologia

Para esse estudo foi utilizado um dataset contendo **1267 amostras** de um composto, e para cada amostra existem **228 variáveis de predição**, dentre as quais 208 são identificadores binários que apontam a ausência ou a presença de uma subestrutura. Além destas temos 20 preditores restantes onde 16 repassam quantidades de ligações e de átomos de bromo e as outras 4 são os pesos moleculares e a área de superfície.[1]

O trabalho foi dividido em dois setores: o *Pré-Processamento dos Dados* e o *Treinamento dos Modelos*. Os dois são esmiuçados a seguir:

### 2.1 Pré-processamento dos Dados

A princípio foi feito uma análise das distribuições das variáveis, das dispersões cruzadas entre as variáveis , da Matriz de Correlação entre as variáveis (não binárias) (Figura 1) e uma análise das componentes principais. As conclusões foram as seguintes:

- Existe uma predominância de "ausência" das 208 subsestruturas descritas pelo dataset.
- Muito dos descritores não binários apresentam comportamento normal como o Número de Átomos (**NumAtom**).
- A Matriz de Correlação (Figura 1) mostrou colinearidades altas entre algumas variáveis ( 34 Variáveis), como entre o Número de Átomos (**NumAtoms**) e o Número de Ligações (**NumBonds**) . Essas variáveis muito correlatas - também analaisadas pela Matriz de Correlação - foram retiradas pois elas aumentam o viés do modelo, aumentando assim o seu erro [1].

Por último, foi feito o escalonamento e a centralização dos dados e , além disso, foi escolhida a transformação *Yeo-Johnson* para reduzir a *skewness* da distribuição do dataset, pois estava muito alta. A transformação *Yeo-Johnson* é a adequada pois ela é uma *Power Transformation* que consegue lidar com números negativos [7]. Ela é mostrada logo abaixo:

$$\psi(\lambda, y) = \begin{cases} ((y-1)^{\lambda} - 1)/\lambda & \lambda \neq 0, \quad y \geq 0 \\ log(y+1) & \lambda = 0, \quad y \leq 0 \\ -((-y-1)^{2-\lambda} - 1)/(2-\lambda) & \lambda \neq 2, \quad y \leq 0 \\ -log(-y+1) & \lambda = 2, \quad y \leq 0 \end{cases} \tag{1}$$

### 2.2 Treinamento dos Modelos

Feito as transformações de pré-processamento os dados foram passados para os modelos, onde cada um é detalhado abaixo:

#### *2.2.1 Regressão Linear*

$$RSS(\beta) = \sum_{i=1}^{N} \left( y_i - \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 \tag{2}$$

1. Acima é visto a função de erro de mínimos quadrados. Ela é utilizada para otimizar os modelos de Regressão Linear simples. Quanto melhor o modelo mais próximo de 0 essa função vai estar. O objetivo sempre é encontrar parâmetros beta que façam o **RSS** se aproximar de 0, dado x amostras de uma variável do dataset.

2. Para escolher o modelo sem levar em consideraç a forma que os dados são repassados para o treinamento, é feita uma **Validação Cruzada** conhecida como *K-Fold*, onde o dataset é dividido em "k" subconjuntos mutualmente excludentes e desses conjuntos é utilizado para obter métricas do erro e os "k-1" restantes é utilizado para o treino do modelo, esse processo é repetido "k" vezes [1].

3. Foram testadas validação *5-Fold* e *10-Fold*, as quais não apresentaram muita diferença. O Gráfico dos valores observados em função dos valores perditos é mostrado na (Figura 3):

#### *2.2.2 Modelo Penalizado*

$$RSS_{L_2} = \sum_{i=1}^{N} \left( y_i - \beta_0 + \sum_{j=1}^{p} x_{ij}\beta_j \right)^2 + \lambda \sum_{k=1}^{p} \beta_j^2 \tag{3}$$

1. Assim como os modelos de Regressão Linear Simples o objetivo desta é minimizar a função de erro vista logo acima. Porém os modelos penalizados (e.g o *Ridge Regression*) possuem a capacidade de controlar a flexibilidade do modelo pemitindo que não ocorra uma aumento desmedido da variâcia durante o treinamento, resultando em um *overfiting*[1].

2. O parâmetro lâmbda permite um maior controle do tamanho dos coeficientes. Assim, durante o aprendizado é possível escolher quantos serão testados. Nesse estudo foram testados 5 parâmetros, onde o mais relevante teve valor 0.01. Esse aprendizado também foi feito em uma validação cruzada do tipo: *5-Fold* e *10-Fold*, obtendo resultados bem semelhantes nas duas validações. O Gráfico dos valores observados em função dos valores perditos é mostrado na (Figura 4):

## 3 Resultados

Diante do exposto e da análise da Tabela 1, é possível concluir que o modelo que mais se adequa é o *linear*, pois o mesmo apresenta o menor *RMSE* de treino e o maior *R2* de treino, apesar de ficar muito próximo do *L2-penalized* - inclusive perde em treino para ele, ou seja, o modelo linear é o que melhor traduz a relação entre características estruturais e a solubilidade. Todos os códigos foram feitos com o uso da linguagem de programação **R**[7] no ambiente de desenvolvimento **RStudio**[8], e os gráficos foram feitos com o uso do pacote **ggplot2**[9]. Esses códigos estão no repositório **matheus123deimos**[6].

## References


[1] **Kuhn, M., Johnson, K. (2013). *Applied Predictive Modeling* New York: Springer.**

[2] **Tetko, I., Tanchuk, V., Kasheva, T., and Villa, A. (2001). Estimation of aqueous solubility of chemical compounds using E-state indices. Journal of Chemical Information and Computer Sciences, 41(6), 1488-1493.**

[3] **Huuskonen, J. (2000). Estimation of aqueous solubility for a diverse set of organic compounds based on molecular topology. Journal of Chemical Information and Computer Sciences, 40(3), 773-777.**

[4] **Wikibooks. *Guia de Latex* Retrieved from https://pt.wikibooks.org/wiki/Late**

[5] **Weisberg, S. (2001) *Yeo-Johnson Power Transformations***

[6] **Cordeiro M.G. Machine-Learning. Repositório GitHub: https://github.com/matheus123deimos/Machine-Learning.**

[7] **R Development Core Team, R: a language and environment for statistical computing, R Foundation for Statistical Computing, Vienna, Austria, ISBN: 3-900051-00-3, 2008, http://www.R-project.orghttp://www.R-project.org.**

[8] **RStudio Team (2016). RStudio: Integrated Development for R. RStudio, Inc., Boston, MA, URL http://www.rstudio.com/http://www.rstudio.com/.**

[9] **Wickham H (2016). ggplot2: Elegant Graphics for Data Analysis. Springer-Verlag New York. ISBN 978-3-319-24277-4, http://ggplot2.orghttp://ggplot2.org.**

# 4 Anexo

| Model | RMSE(Train) | RMSE(Test) | R2(Train) | R2(Test) |
|---|---|---|---|---|
| Linear Regression | 0.50280 | 0.80462 | 0.93958 | 0.85146 |
| $L_2 - penalized$ | 0.51138 | 0.76924 | 0.93750 | 0.86371 |
| PCR | 0.54990 | 0.75509 | 0.92773 | 0.86838 |
| PLS | 1.2798 | 1.2638 | 0.60852 | 0.6378 |

Table 1: Modelos e Seus Pesos

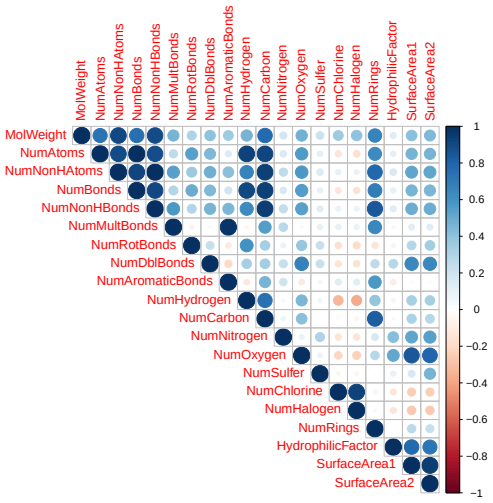


Fig. 1: Matriz de Correlaçã

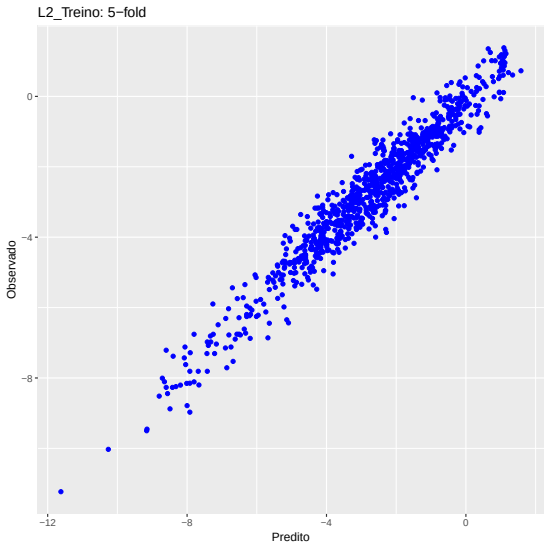


Fig. 2: Treino(5-fold): L2 - penalized

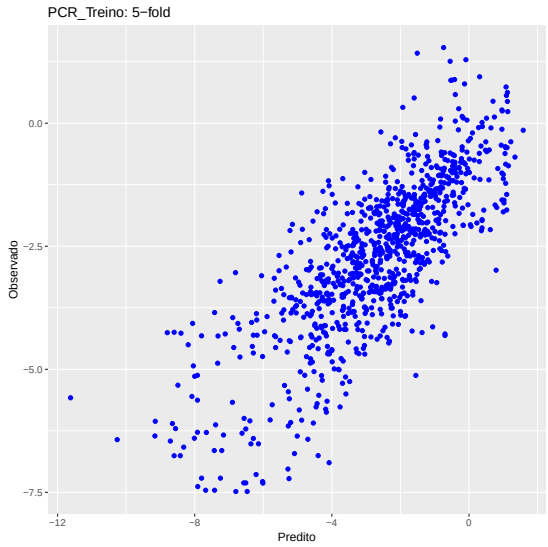


Fig. 3: Treino(5-fold): PCR

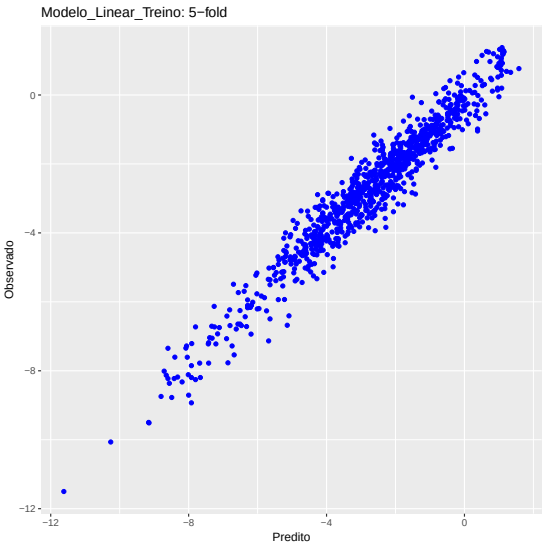


Fig. 4: Treino(5-fold): Regresão Linear

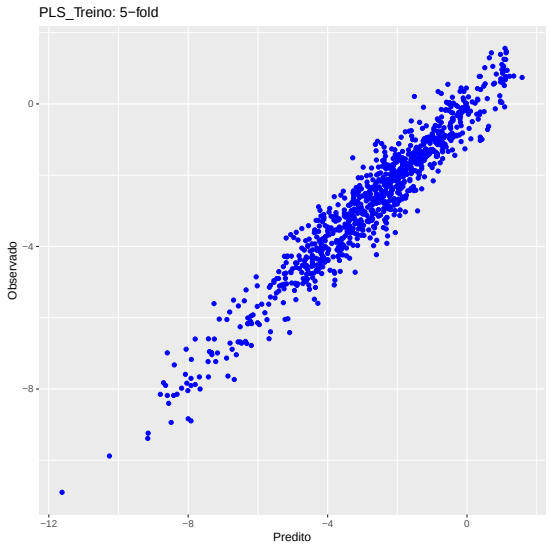


Fig. 5: Treino(5-fold): PLS