Predizendo a Resistência do Concreto à Compressão: uma comparação entre três métodos de regressão

Matheus I. Nesteruk Moreira[1], Nícolas Arruda Maduro[2], Renan Mateus Bernardo do Nascimento[3]

Resumo

O concreto é o material mais importante da Engenharia Civil e é fundamental saber sua resistência à compressão, que é uma função altamente não-linear com muitas variáveis. Este trabalho usa uma database *de amostras de concreto para treinar uma Regressão Linear, um SVR e um* MLP Regressor*, e comparar o desempenho de predição dos três com a intenção de determinar qual deles seria mais adequado para aplicações no mundo real. Foi possível observar que o MLP e o SVR tiveram desempenho bom e muito semelhante, enquanto a Regressão Linear teve um desempenho menos satisfatório.*

Palavras-chave: Resistência à compressão do Concreto. Inteligência Computacional. Regressão Linear. Multilayer Perceptron. SVR.

1 – Introdução

A Engenharia Civil se figura como uma área de extrema importância para sociedade. Suas atividades são indispensáveis para ampliação da infraestrutura, melhoria na qualidade de serviços prestados à sociedade e para a resolução de problemas de caráter econômico e social. Logo é necessário que esteja sempre em constante desenvolvimento.

Dentre os materiais utilizados para construção de estruturas, o concreto é o mais utilizado e mais importante devido suas características de resistência, durabilidade, trabalhabilidade e inúmeras possibilidades arquitetônicas.

Sendo asism, produzir um material de qualidade e resistente à compressão se demonstra um desafio. Por tratar de uma mistura de vários ingredientes, este material é de alta complexidade e o desenvolvimento de um modelo para compreender seu comportamento é uma tarefa difícil.

O trabalho aqui presente tem como objetivo realizar a comparação entre três modelos de regressão (são eles: regressão linear; *Multilayer Perceptron*; *Support Vector Regression*) para predizer a resistência do concreto à compressão, assim, auxiliando no desenvolvimento de um material de qualidade.

A seção 2 trata do desenvolvimento do trabalho. Nela são abordadas fundamentação teórica, metodologia e análise e discussão dos resultados. Em seguida a seção 3 traz uma conclusão acerta dos resultados obtidos em relação as propostas feitas. Por último, encontra-se as referências bibliográficas utilizadas.

2 – Desenvolvimento

2.1 – Fundamentação teórica

Para a realização deste trabalho foram utilizados conceitos de Inteligência Computacional, são eles: problemas de regressão, normalização de dados, regrassão linear múltipla, multilayer perceptron regressor e support vector regression. Eles são discutidos abaixo:

[2]Autor correspondente: nicolasamaduro@gmail.com
[1]Centro Federal de Educação Tecnológica de Minas Gerais - CEFET-MG
[2]Centro Federal de Educação Tecnológica de Minas Gerais - CEFET-MG
[3]Centro Federal de Educação Tecnológica de Minas Gerais - CEFET-MG

2.1.1 Problemas de regressão

Problemas de regressão são técnicas estatísticas para estimar a relação entre variáveis que se relação com um resultado esperado (UYANIK; GÜLER, 2013). Quando não possível criar uma relação matemática entre variáveis e uma ou mais saídas, ou quando ela não é conhecida, podem se criar modelos estáticos que visam simular essa relação a fim de obter os resultados através de outros valores de entrado. Isso justifica o estudo artifícios para realização de regressões.

2.1.2 Normalização de dados

A normalização de dados consiste em tornar diferentes entradas com diferentes médias, escalas e desvios padrão e em valores em escala semelhante de maneira que pode-se comparálos. Isso é importante para o modelo de regressão não seja dominado por uma entrada que possui uma escala superior às demais. Diversos tipos de normalização podem ser aplicados, a utilizada neste trabalho pode ser formulada abaixo:

$$x_i = \frac{x_i}{\mu_{x_i}}$$

2.1.3 Regressão linear múltipla

A regressão usando uma única variável independente é chamada de regressão linear, enquanto a regressão usando mais de uma variável é chamada de regressão multilinear (TABACHNICK; FIDELL, 2007). No entanto, a relação entre cada entrada possui uma relação linear com a variável de resposta. Um modelo de regressão não linear pode ser formulado conforme abaixo:

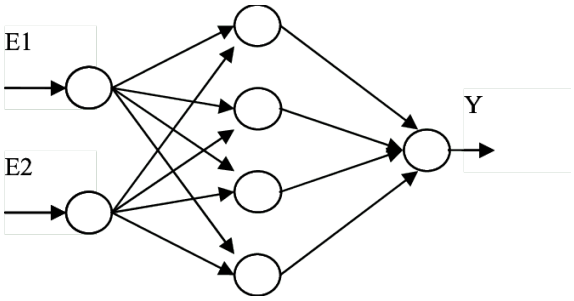
$$y = b_0 + b_1 x_1 + ...b_n * x_n$$

Em que, y é o valor de saída, x_i são as entradas e b_i são as constantes que ponderam as entradas. Vários métodos podem ser utilizados para determinar os valores b_i, neste trabalho, foi utilizado o Gradiente Descendente, que se baseia no caminhada oposta ao sentido do crescimento do erro do sistema. Atingindo ao final do processo de treinamento do modelo, b_i que produzam erros baixos para diferentes entradas.

2.1.4 *Multilayer Perceptron Regressor*

O *multilayer perceptron* consiste em um sistema com neurônios simples interconectados, os chamados nós, conforme a imagem abaixo 1. Estes são conectados por pesos, e ao final, tem seus valores somados e ativados ao passar por uma função (GARDNER; DORLING, 1998).

Figura 1 – *Multilayer Perceptron*



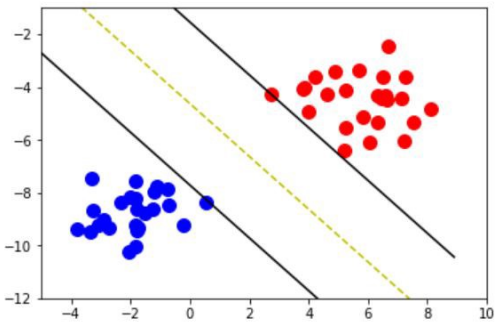
Uma das utilizações do *multilayer perceptron* é a regressão. A regressão utiliza do ajuste dos pesos entre os nós para encontrar a saída esperada. A função de saída utilizada em uma regressão não pode ser de valores discretos, porque isso tornaria a modelo incondizente com a realidade, que produz valores continuos. Outro fator que influencia no capacidade de predição do modelo é tamanho das camadas escondidas. Se for muito pequena, o algoritmo irá convergir para um mínimo local, no entanto, se for muito grande, ele sofrerá um *overfiting*, se tornando extremamente eficiente para a base de dados usada no treinamento, mas não apresentará um bom resultado para outras entradas. Para encontrar determinar os pesos dos vértices, o modelo precisa de um algoritmo para propagar

os valores dentro da rede. Novamente, existem vários opções a serem avaliadas: o Gradiente Descendente, assim como a regressão linear, BPFG que é baseado em modelos de otimização quasi-Newton, BFLM que também utiliza quasi-Newton.

2.1.5 *Support Vector Regression*

O *Support Vector Regression (SVR)* é baseado no *Support Vector Machine (SVM)* que é um modelo utilizado para a classificação de dois grupos. A máquina conceitualmente implementa a seguinte ideia: os vetores de entrada são mapeados não linearmente para um espaço de características de alta dimensão. Neste espaço, uma superfície de decisão linear é construída. Propriedades de borda da superfície de decisão garantem alta capacidade de generalização da máquina de aprendizagem (CORTES; VAPNIK, 1995). A imagem abaixo 2 mostra um exemplo da abordagem do SVM.

Figura 2 – *Exemplo do SVM*



No entanto, a ideia básica no SVR é mapear os dados de entrada x em um espaço de características dimensionais maiores F através de um mapeamento não linear Φ e, em seguida, um problema de regressão linear é obtido e resolvido neste espaço de recurso. Portanto, a aproximação de regressão resolve o problema de estimar uma função com base em um dado conjunto de dados $G = (x_i, y_i)_{i=1}$ (x_i é vetor de entrada, y_i é o valor desejado) (WANG; XU, 2004).

2.2 – Metodologia

A metodologia do trabalho aqui presente consistiu na seleção das ferramentas para realizar os processos, escolha da *database*, pré-processmanto da mesma e realização dos experimentos.

2.2.1 Ferramentas utilizadas

A linguagem de desenvolvimento escolhida foi o *Python 3.7* devido a versatilidade e facilidade de uso além do fato dela proporcionar uma série de bibliotecas de qualidade para se trabalhar com aprendizado de máquina.

Dentre as diversas bibliotecas oferecidas para aprendizado de máquina, *scikit-learn* foi o escolhido. Ela é de código aberto e possui vários algoritmos de classificação, regressão e agrupamento.

Como ambiente de desenvolvimento, o *Jupyter Notebook* oferece documentos com textos explicativos sobre análises e resultados junto a códigos em *Python*. Dessa forma, é possível ter pequenos trechos de código com explicações dos processos realizados de forma clara e objetiva auxiliando na documentação do experimento.

2.2.2 *Database* utilizada

Como *database* para a realização dos experimentos, utilizou-se *Concrete Compressive Strength Data Set* que se encontra em (CONCRETE. . . ,). Ela possuí 1030 instâncias e 9 atributos cuja relação está mostrada na tabela 1.

2.2.3 Pré-processamento

Como pré-processamento e preparação para realização dos experimentos, após realizar a lei-

Tabela 1 – *Relação dos atributos da database*

Nome	Tipo	Medida	Descrição
Cement	Quantitativo	Kg/m^3	Entrada
Blast Furnace Slag	Quantitativo	Kg/m^3	Entrada
Fly Ash	Quantitativo	Kg/m^3	Entrada
Water	Quantitativo	Kg/m^3	Entrada
Superplasticizer	Quantitativo	Kg/m^3	Entrada
Coarse Aggregate	Quantitativo	Kg/m^3	Entrada
Fine Aggregate	Quantitativo	Kg/m^3	Entrada
Age	Quantitativo	Dias (1-365)	Entrada
Concrete Compressive Strength	Quantitativo	MPa	Saída

tura da *database*, os dados foram normalizados divindido-se o valor corrente x_{ij} pela média dos valores do atributo$_i$, onde i é o atributo e j o seu valor.

2.2.4 Experimentos

Para realizar os experimentos, dividiu-se o conjunto de dados em dois grupos de forma aleatória onde 70% foi para o conjunto de treino e os 30% restante para o conjunto de teste.

Tendo o mesmo conjunto de treinamento e teste, realizou-se os três métodos de regressão: regressão linear, *Multilayer Perceptron Regressor* e *Support Vector Regression*.

No *Multilayer Perceptron Regressor* utilizou-se como parâmetro o *solver 'lbfgs'* que é um otimizador da família *quasi-Newton* para realizar a otimização dos pesos. Para fator de apredizagem *alpha*, o valor utilizado foi $0,01$. Estruturou-se a rede com 8 camadas escondidas e função de ativação sigmoide. E 10.000 foi o número máximo de iterações escolhido.

Os damais parâmetros para ajustar as funções de regressão linear e *Support Vector Regression* do *scikit-learn* foram mantidos seus valores *default*.

2.3 – Análise e discussão dos resultados

A seguir, encontra-se o gráfico de saída para cada um dos três métodos. O gráfico mostra o valor real das amostras no eixo x versus o valor predito pelo modelo no eixo y. A reta traçada tem função y = x, e representa todos os pontos em que a predição coincidiria com o valor real. Ou seja, em um modelo perfeito de predição todos os pontos estariam posicionados em cima da reta. E em oposição, quanto mais longe da reta o ponto estiver, maior o erro de predição.

Figura 3 – *Regressão Linear*

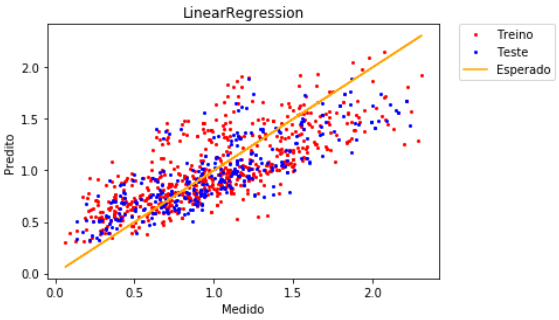


Figura 4 – *Multilayer Perceptron Regressor*

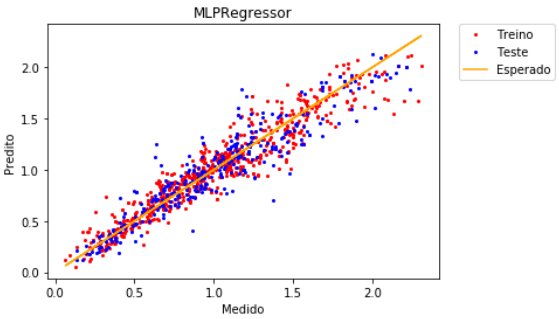


Figura 5 – *Support Vector Regression*

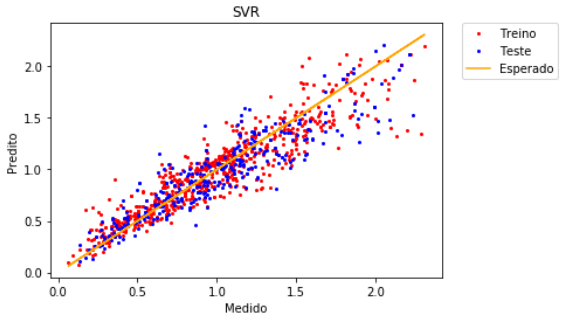


Figura 6 – *Erro quadrático*

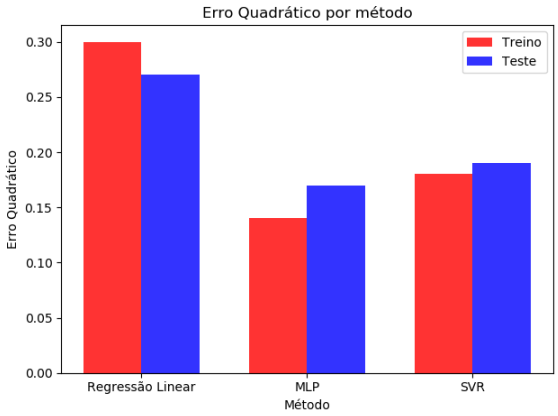
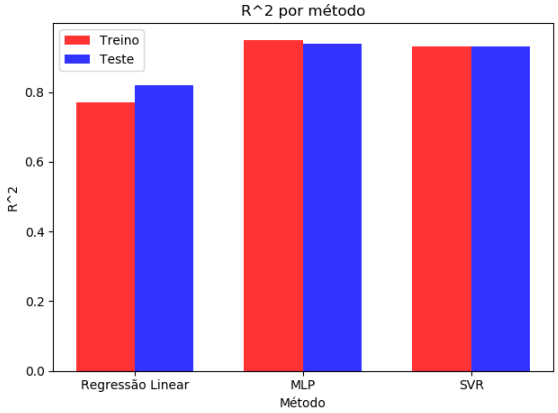


Figura 7 – R^2

Na Figura **Figura 3** vê-se o resultado do método de Regressão Linear e é possível perceber que os pontos, apesar de apresentarem uma tendência em direção semelhante à da reta, estão muito espalhados. Isso significa que o modelo está fazendo a predição com certo grau de acurácia, mas possui um erro bastante considerável. Já na **Figura 4** observa-se uma nuvem de pontos bem mais compacta e mais próxima da reta, principalmente nos pontos de valor mais baixo. Talvez pelo fato de existirem visivelmente mais pontos na metade esquerda do gráfico (aproximadamente de Medido = 0 a Medido = 1,25) do que no restante, fazendo com que o treinamento desta faixa de valores tenha sido mais adequado. Na **Figura 5**, por fim, a nuvem de pontos também se mostrou mais compacta que a **Figura 3**, mas bastante semelhante à da **Figura 4**. Sem grande distinção, ao menos visual, entre as duas. De modo a comparar os métodos de forma quantitativa, a seguir estão apresentados os Erros Quadráticos e o valor R^2 de cada um:



A **Figura 6** mostra o erro medido em cada método e vê-se uma diferença drástica entre a Regressão Linear e os outros métodos, tanto nos pontos de teste quanto nos de de treino. O que está de acordo com o que se observa nos gráficos de pontos e revela a Regressão como detentora do pior desempenho. Entre os restantes houve uma diferença menor, porém significativa, em que o SVR mostrou maior acurácia que o *MLP Regressor* tanto em treino quanto em teste.

A seguir, na **Figura 7**, tem-se o valor R^2, chamado de Coeficiente de Determinação, que é uma medida descritiva da qualidade do ajuste obtido e cujo valor vai de zero a um (PORTALACTION,). É possível observar valores

bastante semelhante entre os três métodos, já que o MLP e o SVR apresentam valores quase iguais entre si e a Regressão, um valor ligeiramente menor. Este resultado leva a crer que os ajustes que foram feitos nos três casos tiveram boa qualidade, visto que o menor R^2 obtido foi de aproximadamente 0,8 e os maiores foram de aproximadamente 0,9. O que condiz com o gráfico da **Figura 6**, que apresentou erros relativamente pequenos, corroborando assim a existência de boa acurácia na predição.

3 – Conclusão

Tendo em vista o objetivo de realizar uma comparação entre os três métodos de regressão, apresentou-se uma análise quantitativa em termos do erro quadrático e R^2.

O que se observa é que o método *Multilayer perceptron* apresentou os melhores resultados em comparação aos demais. Por apresentar menor erro quadrático, quer dizer que ele possui menor erro entre o esperado e o predito; e o maior R^2 indica que esse modelo consegue explicar de forma satisfatória os valores observados.

Logo, em um cenário real, utilizando-se o modelo MLP é possível predizer com boa acertividade a resistência do concreto à compressão dada as variáveis pertinentes da sua formação visando um material de melhor qualidade.

Abstract

Concrete is the most important material in Civil Engineering and it is very important to know its resistence to compression, which is a very non-linear function with many variables. This paper uses a dataset of concrete samples to train a Linear Regression, a SVR and a MLP Regressor, and compare the results to determine which one of them would be most adequate for real world applications. It was possible to observe that MLP and SVR achieved good performance, and very similar, and that the Linear Regression's performance was much less satisfactory.

Referências

CONCRETE Compressive Strength Data Set. Disponível em: <https://archive.ics.uci.edu/ml/datasets/Concrete+Compressive+Strength>. Acesso em: 21 de junho de 2018.

CORTES, C.; VAPNIK, V. Support-vector networks. *Machine learning*, Springer, v. 20, n. 3, p. 273–297, 1995.

GARDNER, M. W.; DORLING, S. Artificial neural networks (the multilayer perceptron)—a review of applications in the atmospheric sciences. *Atmospheric environment*, Elsevier, v. 32, n. 14-15, p. 2627–2636, 1998.

PORTALACTION. Disponível em: <http://www.portalaction.com.br/analise-de-regressao/16-coeficiente-de-determinacao>. Acesso em: 21 de junho de 2018.

TABACHNICK, B. G.; FIDELL, L. S. *Using multivariate statistics*. [S.l.]: Allyn & Bacon/Pearson Education, 2007.

UYANIK, G. K.; GÜLER, N. A study on multiple linear regression analysis. *Procedia-Social and Behavioral Sciences*, Elsevier, v. 106, p. 234–240, 2013.

WANG, W.; XU, Z. A heuristic training for support vector regression. *Neurocomputing*, Elsevier, v. 61, p. 259–275, 2004.