João Victor Barbosa Alves

Seleção Incremental de Variáveis para Aprendizado de Máquina utilizando Preditor Linear e Validação Cruzada

Belo Horizonte, Minas Gerais - Brasil

2018

João Victor Barbosa Alves

Seleção Incremental de Variáveis para Aprendizado de Máquina utilizando Preditor Linear e Validação Cruzada

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus lacinia, erat vitae pulvinar malesuada, dolor enim laoreet lectus, a tincidunt lorem lorem in lorem. Suspendisse sagittis sem ex.

Universidade Federal de Minas Gerais - UFMG

Escola de Engenharia

Programa de Graduação em Engenharia Elétrica

Orientador: Antônio de Pádua Braga

Belo Horizonte, Minas Gerais - Brasil

2018

João Victor Barbosa Alves

Seleção Incremental de Variáveis para Aprendizado de Máquina utilizando Preditor Linear e Validação Cruzada

Lorem ipsum dolor sit amet, consectetur adipiscing elit. Phasellus lacinia, erat vitae pulvinar malesuada, dolor enim laoreet lectus, a tincidunt lorem lorem in lorem. Suspendisse sagittis sem ex.

Trabalho aprovado. 12 de novembro de 2018:

Antônio de Pádua Braga
Orientador

Professor
Convidado 1

Belo Horizonte, Minas Gerais - Brasil
2018

Agradecimentos

Os agradecimentos principais são direcionados à Gerald Weber, Miguel Frasson, Leslie H. Watter, Bruno Parente Lima, Flávio de Vasconcellos Corrêa, Otavio Real Salvador, Renato Machnievscz[1] e todos aqueles que contribuíram para que a produção de trabalhos acadêmicos conforme as normas ABNT com LaTeX fosse possível.

Agradecimentos especiais são direcionados ao Centro de Pesquisa em Arquitetura da Informação[2] da Universidade de Brasília (CPAI), ao grupo de usuários *latex-br*[3] e aos novos voluntários do grupo *abnTeX2*[4] que contribuíram e que ainda contribuirão para a evolução do abnTeX2.

[1] Os nomes dos integrantes do primeiro projeto abnTeX foram extraídos de <http://codigolivre.org.br/projects/abntex/>

[2] <http://www.cpai.unb.br/>

[3] <http://groups.google.com/group/latex-br>

[4] <http://groups.google.com/group/abntex2> e <http://www.abntex.net.br/>

São as perguntas que não sabemos responder que mais nos ensinam.
Elas nos ensinam a pensar.
Se você dá uma resposta a um homem, tudo o que ele ganha é um fato qualquer.
Mas, se você lhe der uma pergunta, ele procurará suas próprias respostas.
(. . .)
Assim, quando ele encontrar as respostas, elas lhe serão preciosas.
Quanto mais difícil a pergunta, com mais empenho procuramos a resposta.
Quanto mais a procuramos, mais aprendemos.
Rothfuss, Patrick - O Temor do Sábio - a Crônica do Matador do Rei - Segundo Dia.

Resumo

Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Ut purus elit, vestibulum ut, placerat ac, adipiscing vitae, felis. Curabitur dictum gravida mauris. Nam arcu libero, nonummy eget, consectetuer id, vulputate a, magna. Donec vehicula augue eu neque. Pellentesque habitant morbi tristique senectus et netus et malesuada fames ac turpis egestas. Mauris ut leo. Cras viverra metus rhoncus sem. Nulla et lectus vestibulum urna fringilla ultrices. Phasellus eu tellus sit amet tortor gravida placerat. Integer sapien est, iaculis in, pretium quis, viverra ac, nunc. Praesent eget sem vel leo ultrices bibendum. Aenean faucibus. Morbi dolor nulla, malesuada eu, pulvinar at, mollis ac, nulla. Curabitur auctor semper nulla. Donec varius orci eget risus. Duis nibh mi, congue eu, accumsan eleifend, sagittis quis, diam. Duis eget orci sit amet orci dignissim rutrum.

Palavras-chave: latex. abntex. editoração de texto.

Lista de ilustrações

Figura 1 – Fluxo geral de desenvolvimento de sistemas de aprendizado de máquina. 16
Figura 2 – Relação entre underfitting-overfitting, generalização-complexidade e viés-variância. 17
Figura 3 – Crescimento do espaço de variáveis em virtude do aumento das dimensões. 18
Figura 4 – Performance em relação ao número de dimenções e amostras. 18
Figura 5 – Validação cruzada: *5-fold*. 23
Figura 6 – Estruturas de teste 33
Figura 7 – Seleção de Variáveis *Stepwise: Communities and Crime*. 35
Figura 8 – Seleção de Variáveis *Stepwise: Forest Fires*. 36
Figura 9 – Seleção de Variáveis *Stepwise: USA Housing*. 37
Figura 10 – Seleção de Variáveis *Stepwise: Wisconsin Breast Cancer Dataset*. 37
Figura 11 – Efeito do parâmetro de regularização (λ): *Wisconsin Breast Cancer Dataset*. 38

Lista de tabelas

Tabela 1 – Conjuntos de dados utilizados para validação. 31

Sumário

1 INTRODUÇÃO . . . 15
1.1 Aprendizado de Máquina . . . 15
1.2 Generalização e Complexidade . . . 16
1.3 O problema da dimensionalidade . . . 18
1.4 Motivação . . . 19
1.5 Objetivos . . . 19

2 FUNDAMENTAÇÃO TEÓRICA . . . 21
2.1 Regressão Linear . . . 21
2.1.1 Treinamento de modelos lineares . . . 21
2.2 Validação Cruzada . . . 22
2.2.1 *Leave-one-out Cross-validation* e Regressão Linear . . . 23
2.3 Aprendizado Incremental . . . 25

3 DESENVOLVIMENTO E METODOLOGIA . . . 29
3.1 Algoritmo . . . 29
3.1.1 Seleção de Variáveis *Stepwise* . . . 29
3.1.2 Método Incremental . . . 30
3.2 Testes . . . 31
3.2.1 Conjuntos de dados . . . 31
3.2.1.1 Preprocessamento . . . 31
3.2.2 Metodologia de Teste . . . 32
3.3 Estudo de Caso: Vallourec - Caracterização de tubos de aço . . . 32
3.3.1 Descrição do Problema . . . 32
3.3.2 Descrição do Conjunto de dados . . . 33
3.3.3 Metodologia . . . 33

4 RESULTADOS E DISCUSSÃO . . . 35
4.1 Teste do Algoritmo . . . 35
4.1.1 Resultados . . . 35
4.1.1.1 *Communities and Crime* . . . 35
4.1.1.2 *Forest Fires* . . . 36
4.1.1.3 *USA Housing* . . . 36
4.1.1.4 *Wisconsin Breast Cancer* . . . 36
4.1.2 Discussão . . . 37
4.2 Estudo de Caso: Vallourec - Caracterização de tubos de aço . . . 38

5 CONCLUSÃO 39

REFERÊNCIAS 41

1 Introdução

O aumento da capacidade de armazenamento e processamento de dados tem possibilitado o desenvolvimento de sistemas inteligentes através do aprendizado de máquina. Tal desenvolvimento, por sua vez, permitiu avanços em diversas áreas da computação, microeletrônica e sensoriamento.

Hoje, aplicações tais como pesquisas web, sistemas *anti-spam*, reconhecimento de voz, recomendações de produto e diversas outras são frutos desse progresso. Porém a disponibilidade de quantidades massivas de dados, apresenta não só um horizonte vasto de possíveis aplicações, mas também desafios para seleção e processamento desses dados.

1.1 Aprendizado de Máquina

Aprendizado de máquina é o campo da computação responsável por desenvolver e empregar sistemas ou modelos que, através da sua exposição à experiências, são capazes de melhorar sua performance na realização de determinada tarefa (MITCHELL, 1997).

O processo de desenvolvimento de modelos com aprendizado de máquina pode ser dividido, em linhas gerais, em duas grandes etapas. A primeira delas refere-se à análise e tratamento dos dados. Nessa etapa procura-se identificar correlações entre as informações disponíveis e as variáveis de interesse. Além disso, são explorados possíveis filtros, transformações e outros algoritmos que possam facilitar o aprendizado do modelo. Também é necessário realizar nessa etapa a separação dos dos dados que serão utilizados para treinamento, validação e teste no decorrer do processo.

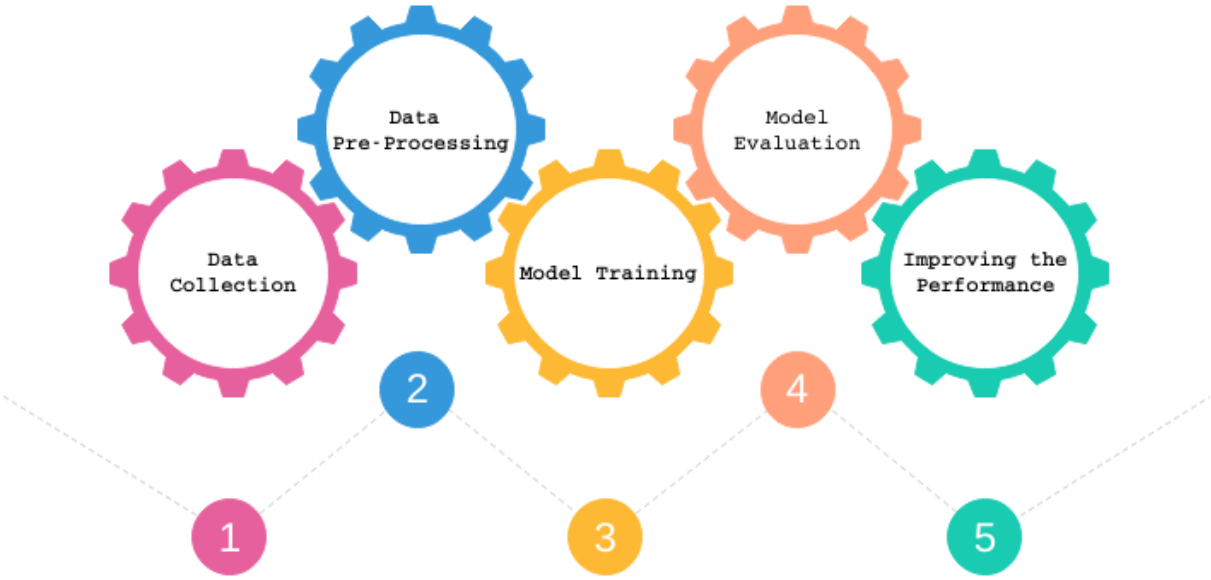
Na etapa seguinte objetiva-se obter um modelo capaz de reproduzir o comportamento do sistema que gerou o conjunto de dados. Para tal, um ou mais modelos e algoritmos são selecionados e treinados. Duas características são de fundamental importância e devem ser controladas: a capacidade de representação do sistema (complexidade) e a capacidade de extrapolação das saídas para novas entradas (generalização).

Os algoritmos utilizados nesta segunda etapa podem ser classificados em uma de três categorias, de acordo com as características dos dados disponíveis. (RUSSELL; NORVIG, 2010)

Na categoria de aprendizado não supervisionado, desenvolve-se sistemas capazes identificar padrões implícitos em um conjunto de dados não rotulados.

No aprendizado por reforço, os sistemas se adaptam de acordo com dados de resposta oriundos do ambiente no qual estão envolvidos.

Figura 1: Fluxo geral de desenvolvimento de sistemas de aprendizado de máquina.



Fonte: *Machine Learning: A Gentle Introduction.* (ABHISHEK, 2018)

Por fim, no aprendizado supervisionado, estão modelos que, a partir de um conjunto de entradas e saídas conhecidas, são capazes de extrapolar a dinâmica do sistema em questão.

1.2 Generalização e Complexidade

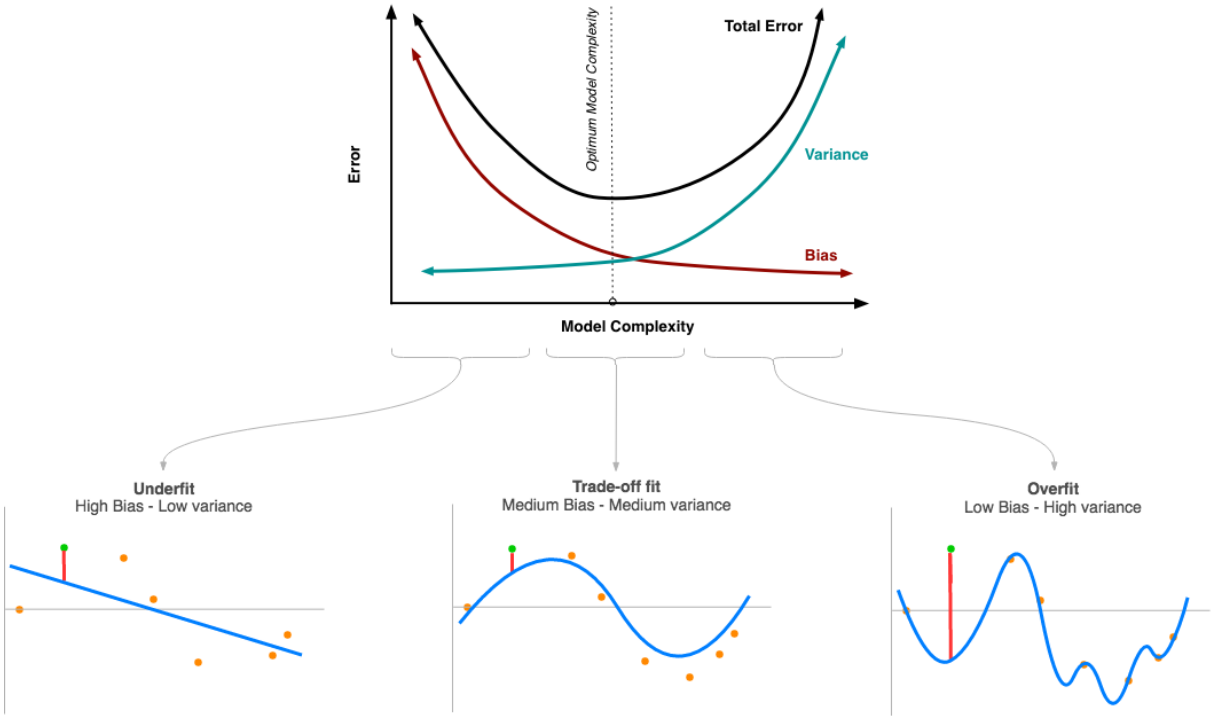
O processo de aprendizado supervisionado é a inferência de um mapeamento de dados de entrada a variáveis de saída a partir de um conjunto de observações. Este pode, portanto, ser interpretado como um ajuste, linear ou não, de uma curva (HAYKIN, 2009). Consequentemente, os modelos obtidos através desse processo também estão sujeitos a *overfitting* e *underfitting*.

Overfitting, ou sobre-ajuste, é caracterizado pela alta performance no conjunto de dados de treinamento e baixa performance em dados nunca observados. Isto é, o modelo assimila desvios causados por erros de medição ou fatores aleatórios presentes no treinamento. Dessa maneira, o erro em relação aos dados de treinamento é reduzido, porém essa melhora não corresponde a uma melhora na representação da realidade.

De maneira similar, o *underfitting*,ou sub-ajuste, é caracterizado pela baixa performance em ambos os conjuntos de dados, indicando que o modelo utilizado não é capaz de

representar de maneira satisfatória a complexidade do sistema real.

Figura 2: Relação entre underfitting-overfitting, generalização-complexidade e viés-variância.



Fonte: *Bias and Variance* (CRUELLS, 2017)

Generalização é o termo usado para descrever a capacidade de um sistema de reagir de maneira satisfatória a novos dados. Isto é, após realizado o treinamento, a capacidade de um sistema de realizar predições precisas (sem *overfitting* ou *underfitting*) para dados nunca observados.

Analogamente, pode-se avaliar o modelo em termos de complexidade, onde modelos mais complexos são capazes de ajustar os dados em uma família maior de curvas, porém estão mais sujeitos a *overfitting* e exigem, em geral uma complexidade amostral (*sample complexity*) maior.

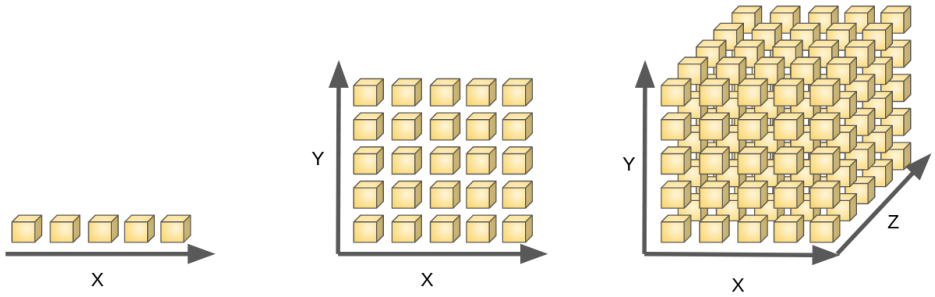
Pode-se ainda avaliar o sistema em termos de viés e variância (*bias-variance tradeoff*), onde o viés representa a parcela do erro atribuida à diferença entre o modelo e a função real do sistema, e a variância a parcela devido à sensibilidade do modelo à pequenas variações nos dados.

1.3 O problema da dimensionalidade

Sistemas que se utilizam de aprendizado de máquina apresentam respostas baseados nos dados de entrada que lhe são apresentados. Dessa maneira, a determinação de que variáveis são relevantes para o sistema é uma etapa fundamental de seu processo de desenvolvimento.

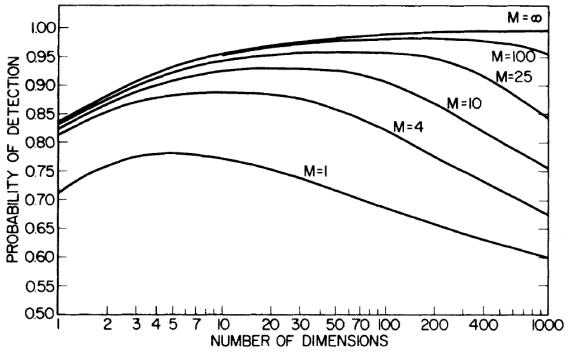
A inclusão de variáveis irrelevantes resulta no aumento da complexidade do modelo, o tornando mais suscetível ao *overfitting* e menos interpretável (JAMES et al., 2017, p. 204). Além disso, um número elevado de variáveis de entrada resulta em um efeito conhecido como a maldição da dimensionalidade (*curse of dimensionality*), onde o aumento do número de dimensões (quantidade de variáveis), implica no aumento exponencial do número de amostras necessárias para representar o espaço de variáveis (BISHOP, 2006, p. 34).

Figura 3: Crescimento do espaço de variáveis em virtude do aumento das dimensões.



Fonte: The Curse of Dimensionality (GLEESON, 2017)

Figura 4: Performance em relação ao número de dimenções e amostras.



Fonte: A Problem of Dimensionality: A Simple Example (TRUNK, 1979)

É desejável então que se utilize o menor número possível de variáveis de entrada que permitam ao modelo produzir a resposta correta. Uma maneira de realizar essa redução de dimensionalidade é selecionar um subconjunto das variáveis disponíveis, descartando aquelas que apresentarem pouca ou nenhuma relevância para o problema (JAMES et al., 2017, p. 204).

1.4 Motivação

A quantidade e qualidade das variáveis incluidas em um modelo é responsável por parcela significativa de sua complexidade e potencial de generalização. Para o desenvolvimento de um modelo satisfatório, é necessário, portanto, a aplicação de procedimentos adequados para a seleção e processamento dos dados iniciais.

Esse trabalho apresenta um método de seleção automática de variáveis de entrada para modelos treinados através de aprendizado supervisionado.

1.5 Objetivos

O algoritmo desenvolvido visa a obtenção de um conjunto de variáveis que possa ser usado para o treinamento de modelos com alta capacidade de generalização.

É um método de seleção de variáveis *stepwise* e, portanto, realiza uma busca gananciosa (*greedy search*) em um conjunto de variáveis candidatas, selecionando, a cada iteração, a variável que apresentar a maior melhora na performance, segundo a métrica adotada.

Nesse trabalho, faz-se o uso de um modelo de baixa complexidade (regressão linear regularizada), aliado ao erro de validação cruzada *leave-one-out*, para se determinar a variação da performance ao se incluir uma variável.

2 Fundamentação Teórica

Nesta seção são abordados conceitos fundamentais para a compreensão deste trabalho, assim como tendências atuais da literatura em relação ao tema estudado.

2.1 Regressão Linear

Os modelos de regressão linear constituem uma classe de modelos que utilizam funções lineares com parâmetros ajustáveis. O exemplo mais simples dessa classe é a função que realiza a combinação linear das entradas com os parâmetros ajustados para gerar uma predição (eq. 2.1).

$$\hat{y} = w_0 + w_1 x_1 + ... + w_p x_p = \sum_{i=0}^{p} w_i X = XW \tag{2.1}$$

Onde p corresponde ao número de dimensões da variável de entrada e $x_0 = 1$.

Modelos mais complexos e de maior aplicabilidade podem ser obtidos ao se considerar um conjunto fixo de transformações não-lineares ($\phi_n(X)$) ao invés das variáveis originais, ou em conjunto com elas. Esses modelos são lineares em relação as suas variáveis independentes, porém são não-lineares em relação as variáveis de entrada (BISHOP, 2006).

2.1.1 Treinamento de modelos lineares

O treinamento de modelos lineares consiste em encontrar o vetor de parâmetros W que maximize a similaridade entre o modelo e sistema modelado. Esse treinamento é normalmente realizado minimizando-se o somatório dos erros quadráticos (eq. 2.2) em função do vetor W.

$$\begin{aligned} E_{sq}(W) &= \frac{1}{2}\sum_{n=1}^{N}(y_n - \hat{y}_n)^2 \\ &= \frac{1}{2}(Y - \hat{Y})^T(Y - \hat{Y}) \\ &= \frac{1}{2}\left(Y^T Y - 2\hat{Y}^T Y + \hat{Y}^T\hat{Y}\right) \\ &= \frac{1}{2}\left(Y^T Y - 2W^T X^T Y + \sum_{n=1}^{N}(X_n W)^2\right) \end{aligned} \tag{2.2}$$

$$\begin{aligned}\frac{\partial E_{sq}(W)}{\partial W} &= \frac{1}{2}\left(0 - 2X^TY + 2\sum_{n=1}^{N}\left(X_n^TX_n\right)W\right) \\ &= -X^TY + X^TXW\end{aligned} \tag{2.3}$$

Igualando-se a eq. 2.3 a zero e isolando o vetor W, obtém-se a eq. 2.4, conhecida como a equação normal para o problema dos mínimos quadrados.

$$\begin{aligned}X^TXW &= X^TY \\ W &= (X^TX)^{-1}X^TY \\ W &= X^+Y\end{aligned} \tag{2.4}$$

O termo (X^TX) pode se aproximar de uma matriz singular se muitas das variáveis envolvidas forem linearmente dependentes, resultando assim em dificuldades para o cálculo numérico dos valores e possivelmente em um vetor de parâmetros de alta magnitude. Um termo de regularização pode ser adicionado na eq. 2.2 para minimizar esse problema, garantindo que a nova matriz não é singular.

A adição do termo de regularização tem ainda o efeito de limitar a complexidade efetiva do modelo, reduzindo o *overfitting* e possibilitando a utilização de conjuntos de dados menores (BISHOP, 2006).

Comumente utiliza-se norma-L2 do vetor de parâmetros como termo de regularização, dando origem a *ridge regression*. A dedução da equação normal com termo de regularização é análoga à apresentada e resulta na eq. 2.5.

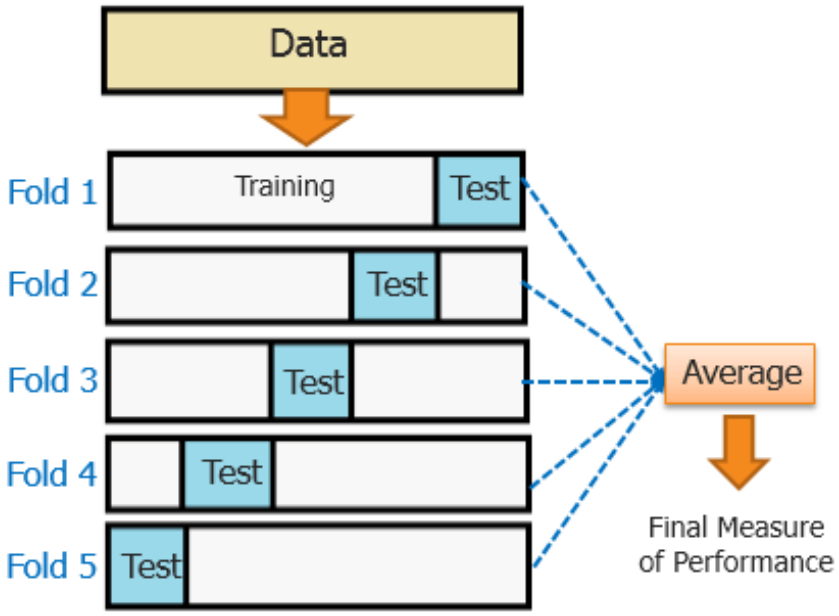
$$W = (\lambda I + X^TX)^{-1}X^TY = A^{-1}X_{[i]}^TY \tag{2.5}$$

2.2 Validação Cruzada

A validação cruzada (*cross-validation*) é um conjunto de metodologias de treinamento e validação que, de maneira simples e efetiva, permitem a estimativa do erro de generalização do modelo.

No método *k-fold* de validação cruzada, os dados de treinamento são divididos em k conjuntos aleatórios de tamanhos aproximadamente iguais. Realizada essa divisão, a cada iteração do treinamento, os parâmetros do modelo são determinados utilizando k-1 conjuntos e sua performance é avaliada no conjunto restante. O processo se repete até que todos os k conjuntos tenham sido utilizados para avaliação e a média das performances observadas fornecem uma estimativa da performance de generalização do modelo (CAWLEY; TALBOT, 2010).

Figura 5: Validação cruzada: *5-fold*.



Fonte: Different types of Validations in Machine Learning (Cross Validation) (SRINIDHI, 2018)

Um caso especial do método descrito é o *leave-one-out cross-validation* (LOOCV), onde k é igual ao total de amostras disponíveis (N), ou seja, o treinamento é realizado N vezes com N-1 amostras, e validado na amostra excluída. Tal abordagem apesar de apresentar, em geral, um custo computacional elevado, resulta em um modelo com menor variabilidade e viés para modelos de regressão linear (BURMAN, 1989), produzindo assim resultados com melhor performance e menos *overfitting*.

2.2.1 *Leave-one-out Cross-validation* e Regressão Linear

Especificamente para o caso do preditor linear, é possível determinar o resultado do LOOCV sem a necessidade de se calcular N preditores, tornando essa uma estratégia interessante para determinar a capacidade de generalização de um modelo linear. A seguir uma dedução adaptada de (SEBER; LEE, 2012, p. 268) é apresentada.

O erro de LOOCV corresponde a média dos erros quadráticos de cada estimador obtido excluindo-se uma amostra.

$$LOOCV = \frac{1}{N}\sum_{i=1}^{N} e_{[i]}^2 \tag{2.6}$$

$$e_{[i]} = y_i - \hat{y}_{[i]} = y_i - x_i W_{[i]} \tag{2.7}$$

Onde o i indexa a amostra excluída no treinamento, $\hat{y}_{[i]}$ corresponde à saída do preditor calculado sem a amostra (x_i, y_i) e $W_{[i]}$ os parâmetros desse preditor.

Pode-se definir o vetor de parâmetros $W_{[i]}$ conforme a eq. 2.8.

$$W_{[i]} = A_{[i]}^{-1} X_{[i]}^T Y_{[i]} \tag{2.8}$$

É possivel perceber as seguintes igualdades:

$$\begin{aligned} A_{[i]} &= \lambda I + X_{[i]}^T X_{[i]} \\ &= \lambda I + X^T X - x_i^T x_i \\ &= A - x_i^T x_i \end{aligned} \tag{2.9}$$

$$X_{[i]}^T Y_{[i]} = X^T Y - x_i^T y_i \tag{2.10}$$

Aplicando a formula de Sherman–Morrison à eq. 2.9, temos:

$$\begin{aligned} h_i &= x_i A^{-1} x_i^T \\ A_{[i]}^{-1} &= A^{-1} + \frac{A^{-1} x_i x_i^T A^{-1}}{1 - h_i} \end{aligned} \tag{2.11}$$

Substituindo as eq. 2.10 e 2.11 na eq. 2.8.

$$\begin{aligned} W_{[i]} &= \left(A^{-1} + \frac{A^{-1} x_i^T x_i A^{-1}}{1 - h_i} \right) \left(X^T Y - x_i^T y_i \right) \\ &= W - A^{-1} x_i^T y_i + \frac{A^{-1} x_i^T x_i W}{1 - h_i} - \frac{A^{-1} x_i^T x_i A^{-1} x_i^T y_i}{1 - h_i} \\ &= W - \frac{A^{-1} x_i^T}{1 - h_i} \left(y_i (1 - h_i) - x_i W + h_i y_i \right) \\ &= W - \frac{A^{-1} x_i^T}{1 - h_i} \left(y_i - \hat{y}_i \right) \end{aligned} \tag{2.12}$$

Substituindo a eq. 2.12 na eq. 2.7.

$$\begin{aligned} e_{[i]} &= y_i - x_i W_{[i]} \\ &= y_i - x_i \left(W - \frac{A^{-1} x_i^T}{1 - h_i} \left(y_i - \hat{y}_i \right) \right) \\ &= y_i - \hat{y}_i + \frac{h_i}{1 - h_i} \left(y_i - \hat{y}_i \right) \\ &= \frac{y_i - \hat{y}_i}{1 - h_i} \end{aligned} \tag{2.13}$$

Seja P a matriz de projeção (BASILEVSKY, 2005, p. 303) do preditor (também conhecida como matriz chapéu (SEBER; LEE, 2012, p. 266)) e a matriz de aniquilação M (HAYASHI, 2001, p. 18) definidas conforme a eq. 2.14

$$\begin{aligned} \hat{Y} = PY &\implies P = XA^{-1}X^T \\ MY = Y - \hat{Y} &\implies M = I - P = I - XA^{-1}X^T \end{aligned} \tag{2.14}$$

Para cada índice i, o numerador da eq. 2.13 corresponde a linha de mesmo índice da matriz MY. Além disso o denominador da equação corresponde ao elemento da diagonal da matriz M de mesmo índice. A partir dessas observações, pode-se agrupar todos os valores de $e_{[i]}$ na forma da matriz E_{LOOCV}, conforme a equação 2.15.

$$E_{LOOCV} = diag(M)^{-1}MY \tag{2.15}$$

Como a matriz M é simétrica, a eq. 2.6 pode ser reescrita conforme eq. 2.16.

$$\begin{aligned} LOOCV &= \frac{1}{N}\sum_{i=1}^{N} e_{[i]}^2 \\ &= \frac{1}{N}E_{LOOCV}^2 \\ &= \frac{1}{N}E_{LOOCV}^T E_{LOOCV} \\ &= \frac{1}{N}Y^T M\ diag(R)^{-2}MY \end{aligned} \tag{2.16}$$

2.3 Aprendizado Incremental

A atualização dos parâmetros do modelo de regressão linear para uma nova amostra pode ser realizada de maneira incremental, sem a necessidade de se calcular um novo modelo (ROSASCO, 2015). Nesse trabalho porém, é de interesse calcular a atualização incremental ao se adicionar uma nova variável. Em especial, é de interesse a determinação de uma forma fechada para a matriz de aniquilação, usada no calculo do erro de validação cruzada.

A matrizes de interesse para o problema de dimensão $p+1$ tem a forma descrita na eq. 2.17

$$\begin{aligned} X_{p+1} &= [\mathbf{X_p}\ x_{p+1}] \\ Y_{p+1} &= [\mathbf{Y_p}\ y_{p+1}] \\ A_{p+1} &= (X_{p+1}^T X_{p+1} + \lambda I) = \begin{bmatrix} \mathbf{A_p} & \mathbf{X_p}^T x_{p+1} \\ x_{p+1}^T \mathbf{X_p} & x_{p+1}^T x_{p+1} + \lambda \end{bmatrix} \\ M_{p+1} &= I - X_{p+1}A_{p+1}^{-1}X_{p+1}^T \end{aligned} \tag{2.17}$$

É necessário então determinar a matriz A_{p+1}^{-1} em função das matrizes já conhecidas. Para tal, aplica-se a identidade da inversa de uma matriz em blocos, conforme a eq. 2.18, na matriz A_{p+1}.

$$\begin{aligned} \Delta &= (\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}) \\ \begin{bmatrix} \mathbf{A} & \mathbf{B} \\ \mathbf{C} & \mathbf{D} \end{bmatrix}^{-1} &= \begin{bmatrix} \mathbf{A}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \mathbf{A}^{-1}\mathbf{B}\Delta^{-1}\mathbf{C}\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{B}\Delta^{-1} \\ -\Delta^{-1}\mathbf{C}\mathbf{A}^{-1} & \Delta^{-1} \end{bmatrix} \end{aligned} \tag{2.18}$$

Através da eq. 2.17, obtém-se os termos A, B, C, D e Δ na eq. 2.19.

$$\begin{aligned} A &= A_p \\ B &= X_p^T x_{p+1} \\ C &= x_{p+1}^T X_p = B^T \\ D &= \lambda + x_{p+1}^T x_{p+1} \\ \Delta &= \lambda + x_{p+1}^T x_{p+1} - x_{p+1}^T X_p A_p^{-1} X_p^T x_{p+1} \end{aligned} \tag{2.19}$$

O termo Δ pode ser simplificado para a forma da eq. 2.20.

$$\begin{aligned} \Delta &= \lambda + x_{p+1}^T x_{p+1} - x_{p+1}^T X_p A_p^{-1} X_p^T x_{p+1} \\ &= \lambda + x_{p+1}^T (I x_{p+1} - X_p A_p^{-1} X_p^T x_{p+1}) \\ &= \lambda + x_{p+1}^T (I - X_p A_p^{-1} X_p^T) x_{p+1} \\ &= \lambda + x_{p+1}^T M_p x_{p+1} \end{aligned} \tag{2.20}$$

Substituindo, obtém-se a eq. 2.21

$$\begin{aligned} A_{p+1}^{-1} &= \begin{bmatrix} \mathbf{A}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{\Delta} \begin{bmatrix} \mathbf{A}^{-1}\mathbf{B}\mathbf{C}\mathbf{A}^{-1} & -\mathbf{A}^{-1}\mathbf{B} \\ -\mathbf{C}\mathbf{A}^{-1} & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A_p}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{\Delta} \begin{bmatrix} A_p^{-1}(X_p^T x_{p+1})(X_p^T x_{p+1})^T A_p^{-1} & -A_p^{-1}(X_p^T x_{p+1}) \\ -(X_p^T x_{p+1})^T A_p^{-1} & 1 \end{bmatrix} \\ &= \begin{bmatrix} \mathbf{A_p}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{\Delta} \begin{bmatrix} A_p^{-1}(X_p^T x_{p+1}) \\ -1 \end{bmatrix} \begin{bmatrix} x_{p+1}^T X_p A_p^{-1} & -1 \end{bmatrix} \end{aligned} \tag{2.21}$$

Por fim, substituindo a eq. 2.21 no termo M_{p+1} da eq. 2.17, obtém-se a expressão para a nova matriz de aniquilação.

$$
\begin{aligned}
M_{p+1} &= I - X_{p+1} A_{p+1}^{-1} X_{p+1}^T \\
&= I - X_{p+1} \left(\begin{bmatrix} \mathbf{A_p}^{-1} & 0 \\ 0 & 0 \end{bmatrix} + \frac{1}{\Delta} \begin{bmatrix} A_p^{-1}(X_p^T x_{p+1}) \\ -1 \end{bmatrix} \begin{bmatrix} x_{p+1}^T X_p A_p^{-1} & -1 \end{bmatrix} \right) X_{p+1}^T \\
&= \left(I - \begin{bmatrix} \mathbf{X_p} & x_{p+1} \end{bmatrix} \begin{bmatrix} \mathbf{A_p}^{-1} & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \mathbf{X_p}^T \\ x_{p+1}^T \end{bmatrix} \right) - \\
&\qquad \frac{1}{\Delta} \left(\begin{bmatrix} \mathbf{X_p} & x_{p+1} \end{bmatrix} \begin{bmatrix} A_p^{-1}(X_p^T x_{p+1}) \\ -1 \end{bmatrix} \begin{bmatrix} x_{p+1}^T X_p A_p^{-1} & -1 \end{bmatrix} \begin{bmatrix} \mathbf{X_p}^T \\ x_{p+1}^T \end{bmatrix} \right) \qquad (2.22) \\
&= M_p - \frac{1}{\Delta} \left(\mathbf{X_p} A_p^{-1} X_p^T x_{p+1} - x_{p+1} \right) \left(x_{p+1}^T X_p A_p^{-1} \mathbf{X_p}^T - x_{p+1}^T \right) \\
&= M_p - \frac{1}{\Delta} \left(\mathbf{X_p} A_p^{-1} X_p^T - I \right) x_{p+1} x_{p+1}^T \left(X_p A_p^{-1} \mathbf{X_p}^T - I \right) \\
&= M_p - \frac{M_p x_{p+1} x_{p+1}^T M_p}{\Delta} \\
M_{p+1} &= M_p - \frac{M_p x_{p+1} x_{p+1}^T M_p}{\lambda + x_{p+1}^T M_p x_{p+1}}
\end{aligned}
$$

3 Desenvolvimento e Metodologia de Testes

3.1 Algoritmo

3.1.1 Seleção de Variáveis *Stepwise*

O algoritmo proposto nesse trabalho é baseado no método de seleção de variáveis *stepwise* (*Stepwise Selection*), comumente utilizado em conjunto com modelos de regressão linear. É um método de *greedy search*, onde, a cada iteração, a variável que apresentar o melhor ganho de performance é adicionada ao conjunto de entradas.

O modelo é construido incrementalmente até que não haja mais melhora de performance ao acrescentar alguma das variáveis restantes ou não haja mais variáveis para serem consideradas. O método é descrito em pseudocódigo no algoritmo 1.

Algoritmo 1: *Forward Stepwise Selection* (FSS)

Entrada: *variáveis*: lista contendo as variáveis disponíveis;
Entrada: *saídas*: lista contendo as saídas do modelo;
Saída: *selecionadas*: lista de variáveis relevantes.

$selecionadas \leftarrow \{\ \}$
$melhorErro \leftarrow \infty$
repita
  $melhorVariável \leftarrow NULL$
  para cada *elemento var em variáveis* **faça**
    $entradas \leftarrow (var \cup selecionadas)$
    $model \leftarrow ajuste(entradas, saídas)$
    $erro \leftarrow avalia(model, entradas, saídas)$
    se $erro < melhorErro$ **então**
      $melhorErro \leftarrow erro$
      $melhorVariável \leftarrow var$
  $selecionadas \leftarrow (melhorVariável \cup selecionadas)$
  $variáveis \leftarrow (variáveis \setminus \{melhorVariável\})$
até $variáveis == \{\ \}$ *ou (critério de parada)*;

Esse algoritmo apresenta uma redução expressiva na quantidade de modelos ajustados em relação ao método da seleção do melhor subconjunto, que é um algoritmo de busca exaustiva, onde todas as combinações possíveis das variáveis de entrada são testadas. Na primeira iteração, o modelo nulo deve ser ajustado, em seguida, todas as p varíaveis

devem ser avaliadas e, a cada iteração posterior, uma delas é retirada. Portanto o total de modelos ajustados pode ser descrito pela eq. 3.1.

$$Ajustes = 1 + \sum_{k=0}^{p-1}(p-k) = 1 + \frac{p(p+1)}{2} = \frac{1}{2}(p^2 + p + 2) \tag{3.1}$$

Porém, ao contrário do método da seleção do melhor subconjunto, não há garantia que o subconjunto determinado é a combinação ótima das variáveis (JAMES et al., 2017, p. 208).

Um fator determinante para a eficácia do algoritmo é a escolha da métrica pela qual os modelos serão avaliados. A utilização de uma métrica que considere simplesmente os dados ajustados não é adequada, uma vez que, nessa situação, o incremento de uma variável no modelo sempre acarretará em uma melhora no erro de treinamento, porém não necessariamente na capacidade de generalização do modelo.

Algumas técnicas tentam determinar a capacidade de generalização através de informações obtidas com os dados de treinamento, tais como o critério de Informação de Akaike (AIC) ou critério de informação Bayesiano (BIC), porém elas se baseiam no comportamento assintótico, isto é, quando a quantidade de amostras é bastante elevada.

Uma alternativa a essas técnicas é a utilização de validação cruzada, onde o modelo selecionado é aquele que apresenta a melhor performance no conjunto de testes. Dessa maneira, obtém-se diretamente uma estimativa do erro de generalização, além de se assumir menos condições em relação ao modelo e aos dados utilizados. O grande desafio dessa técnica é o custo computacional da validação cruzada que, quando associado ao custo do método de seleção de variáveis *stepwise* pode tornar proibitiva sua implementação.

Em especial, a utilização desse método associada ao *leave-one-out cross-validation* resultaria no ajuste de $^1/_2(p^2 + p + 2)(N - 1)$ modelos, tornando o algoritmo computacionalmente inviável.

3.1.2 Método Incremental

Para minimizar o problema do custo computacional, o método implementado utiliza o resultado apresentado na eq. 2.16 para calcular o erro de validação cruzada sem a necessidade de cálcular $N - 1$ modelos de regressão linear. Além disso, o cálculo da matriz de aniquilação é realizado de maneira incremental, conforme deduzido na eq. 2.22.

O algoritmo 1 é então modificado para realizar o cálculo incremental, utilizando LOOCV como métrica para seleção dos das variáveis e atualização incremental do modelo.

Algoritmo 2: *Forward Stepwise Incremental Selection*

Entrada: *variáveis*: lista contendo as variáveis disponíveis;
Entrada: *saídas*: lista contendo as saídas do modelo;
Saída: *selecionadas*: lista de variáveis relevantes.

$selecionadas \leftarrow \{\ \}$
$melhorErro \leftarrow \infty$
$M \leftarrow I_N$
repita
  $melhorVariável \leftarrow NULL$
  para cada *elemento var em variáveis* **faça**
    $M' \leftarrow ajusteIncremental(M, var)$
    $erro \leftarrow erroLOOCV(M, saídas)$
    se $erro < melhorErro$ **então**
      $melhorErro \leftarrow erro$
      $melhorVariável \leftarrow var$
      $melhorM \leftarrow M'$
  $M \leftarrow melhorM$
  $selecionadas \leftarrow (melhorVariável \cup selecionadas)$
  $variáveis \leftarrow (variáveis \setminus \{melhor_{variável}\})$
até $variáveis == \{\ \}$ *ou (critério de parada)*;

3.2 Testes

3.2.1 Conjuntos de dados

Para teste e validação do algoritmo desenvolvido, utilizou-se três bancos de dados conhecidos na literatura, especificados na tabela 1.

Tabela 1: Conjuntos de dados utilizados para validação.

Nome	Variáveis	Amostras
Communities and Crime	101	2215
Forest Fires	12	517
USA Housing Dataset	80	1460
Wiscoin Breast Cancer	32	194

3.2.1.1 Preprocessamento

De maneira geral, em um conjunto de amostras, as variáveis apresentam diferentes unidades, magnitudes e escalas, isso dificulta a comparação de valores e pode prejudicar o treinamento de modelos. Por isso é comum aplicar uma transformação nos dados, de

maneira a torna-los comparáveis entre si (*feature scaling*), além de facilitar a detecção de *outliers*.

Embora o algoritmo utilizado se baseie em regressões lineares que, em sua forma mais simples, são robustas em relação a magnitude das variáveis, a adição do termo de regressão penaliza o crescimento do vetor de parâmetros, tornando o modelo sensivel à diferença de magnitudes entre variáveis.

Nesse trabalho, todas as variáveis foram normalizadas, de maneira a se obter entradas e saídas que apresentam média nula e desvio padrão unitário. Tal transformação é conhecida como *z-score* (eq. 3.2).

$$X_p = \frac{X_p - \bar{X_p}}{SD(X_p)} \tag{3.2}$$

Além disso, uma variável de entrada com valor unitário foi adicionada posteriormente a cada conjunto de dados, permitindo a consideração do termo livre (w_0) nos cálculos de maneira transparente.

3.2.2 Metodologia de Teste

Com o intuito de identificar um subconjunto ótimo de variáveis para cada conjunto de dados, o algoritmo desenvolvido foi empregado e o erro de validação cruzada, conforme a eq. 2.16, foi registrado ao final de cada iteração. Dessa maneira é possível analisar o efeito da inclusão de uma nova variável na capacidade de generalização do modelo e avaliar até onde é benéfico o aumento de uma nova dimensão.

3.3 Estudo de Caso: Vallourec - Caracterização de tubos de aço

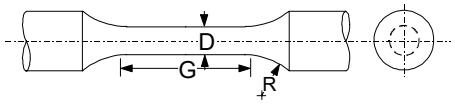
3.3.1 Descrição do Problema

Para que um tubo possa ser utilizado na extração de petróleo e gás, ele deve ser submetido a diversos testes e receber uma certificação que comprove que o mesmo está adequado para utilização.

O teste em questão, verifica a resistência à corrosão sob tensão. Durante a sua realização, um corpo de prova extraído do tubo é imerso em uma solução ácida e deve resistir sem fraturas durante 720 horas. Todo o lote de tubos deve aguardar o resultado do teste para ser despachado. Caso 2 corpos de prova falhem, o material deve ser retratado e o teste refeito.

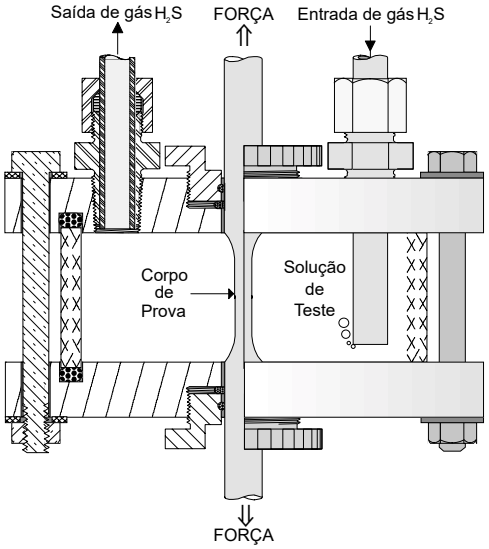
O grande tempo despendido durante o teste e a necessidade de aguardar o resultado para prosseguir para as etapas seguintes de produção resulta em aumento de estoques e

Figura 6: Estruturas de teste



Dimension	Standard specimen	Subsize specimen
D	6.35 ±0.13 mm (0.250 ±0.005 in.)	3.81 ±0.05 mm (0.150 ±0.002 in.)
G	25.4 mm (1.00 in.)	25.4 mm (1.00 in.)
R (min.)	15 mm (0.60 in.)	15 mm (0.60 in.)

(a) Corpo de prova utilizado para o teste



(b) Esquemático da câmara para a realização do teste

Fonte: NACE Standard TM0177 (NACE International, 2005, p. 7,8) (Adaptada)

altos tempos para o atendimento dos pedidos dos clientes (lead time).

Diversas características que podem influenciar no resultado do teste foram levantadas e objetiva-se a determinação daquelas com maior significância para a determinação de um modelo capaz de realizar predições do resultado do teste.

3.3.2 Descrição do Conjunto de dados

Utilizando-se o conhecimento a priori do processo, uma filtragem inicial dos dados foi realizada, removendo caracteristicas irrelevantes para o processo de modelagem, tais como identificações, lotes e datas. Variáveis que não apresentam nenhuma variação no conjunto de dados também foram removidas. Além disso, por motivos de confidencialidade, o nome das variáveis foram anonimizados. Por fim, os dados passaram pelo processo de normalização descrito anteriormente.

Após o processamento inicial, o conjunto de dados possui 1064 amostras, com 30 possíveis variáveis de entrada e 3 varíaveis de saída.

3.3.3 Metodologia

4 Resultados e Discussão

4.1 Teste do Algoritmo

O algoritmo foi aplicado a cada um dos bancos de dados indicados na sessão anterior, de maneira a determinar os subconjunto ótimos de variáveis para uso em um modelo de regressão. O resultado obtido determina, a cada iteração, a variável ainda não utilizada que, ao ser acrescentada no modelo, traz a maior redução do erro de validação cruzada (ou o menor aumento).

A cada iteração "i", um subconjunto candidato de i variáveis é determinado. O melhor subconjunto é aquele que minimiza o erro LOOCV. É importante reiterar que o algoritmo utiliza uma estratégia de *greedy search* para buscar os subconjuntos e, portanto, a optimalidade global desses subconjuntos não é garantida.

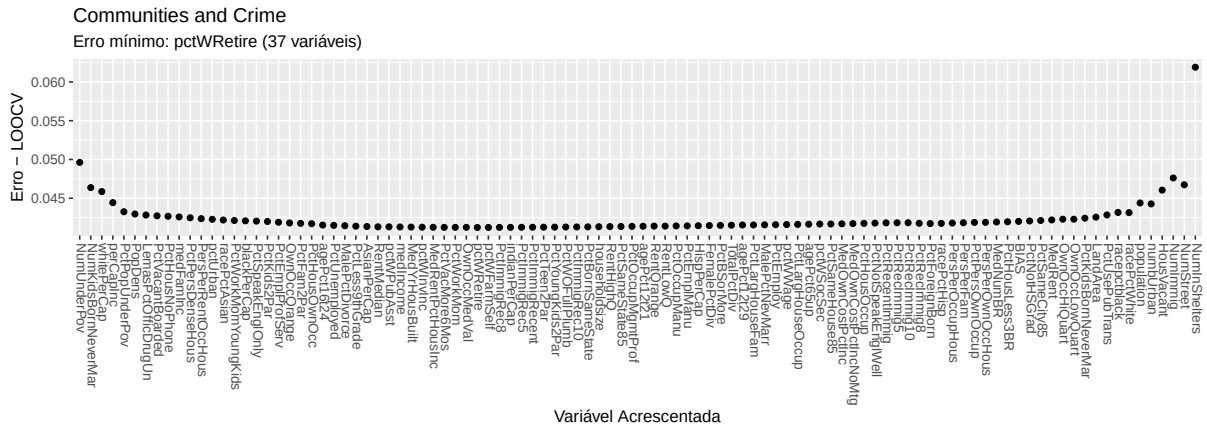
4.1.1 Resultados

4.1.1.1 *Communities and Crime*

No banco de dados *Forest Fires*, o erro de validação cruzada LOOCV mínimo foi observado em um conjunto com apenas 37 das 102 variáveis. A adição de variáveis posteriormente teve pouca influência no erro de validação cruzada, com uma tendência de overfitting na inclusão das últimas variáveis.

O resultado para cada iteração pode ser verificado na figura 7.

Figura 7: Seleção de Variáveis *Stepwise: Communities and Crime*.



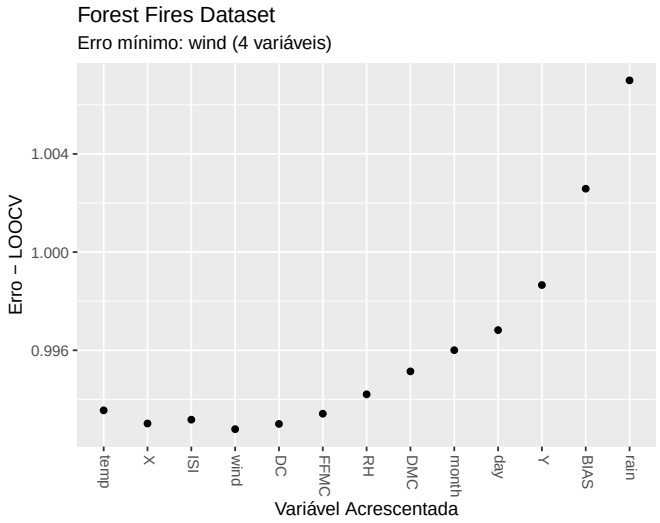
Fonte: Própria.

4.1.1.2 *Forest Fires*

No banco de dados *Forest Fires*, o erro de validação cruzada LOOCV mínimo foi observado em um conjunto com apenas 4 das 12 variáveis. A adição de variáveis posteriormente resultou em overfitting significativo.

O resultado para cada iteração pode ser verificado na figura 8.

Figura 8: Seleção de Variáveis *Stepwise: Forest Fires*.



Fonte: Própria.

4.1.1.3 *USA Housing*

No banco de dados *USA Housing*, o erro de validação cruzada LOOCV mínimo foi observado em um conjunto com apenas 35 das 80 variáveis. A adição de variáveis posteriormente teve pouca influência no erro de validação cruzada, porém aumentando-o lentamente.

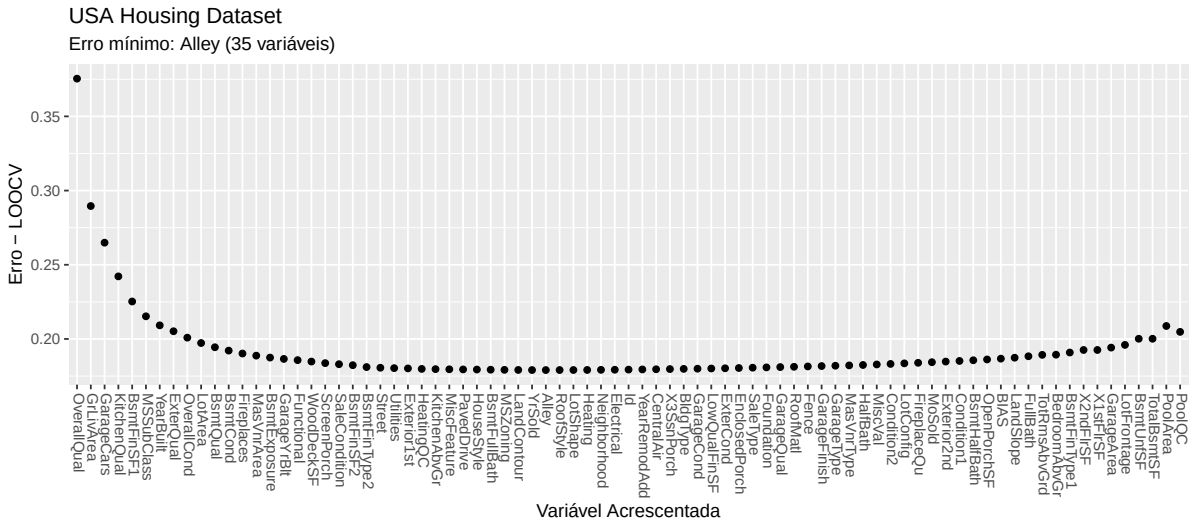
O resultado para cada iteração pode ser verificado na figura 9.

4.1.1.4 *Wisconsin Breast Cancer*

No banco de dados *Wisconsin Breast Cancer*, o erro de validação cruzada LOOCV mínimo foi observado em um conjunto com apenas 8 das 32 variáveis, e foi identificada também uma tendência significativa ao overfitting quando utilizado um subconjunto candidato de mais de 16 variáveis.

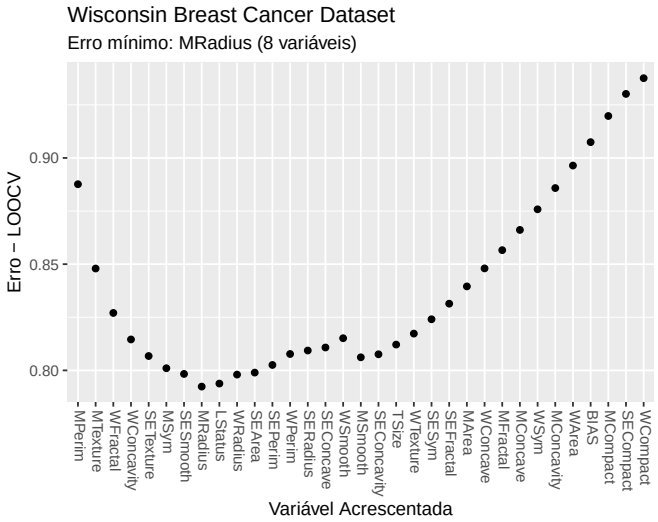
O resultado para cada iteração pode ser verificado na figura 10.

Figura 9: Seleção de Variáveis *Stepwise: USA Housing*.



Fonte: Própria.

Figura 10: Seleção de Variáveis *Stepwise: Wisconsin Breast Cancer Dataset*.



Fonte: Própria.

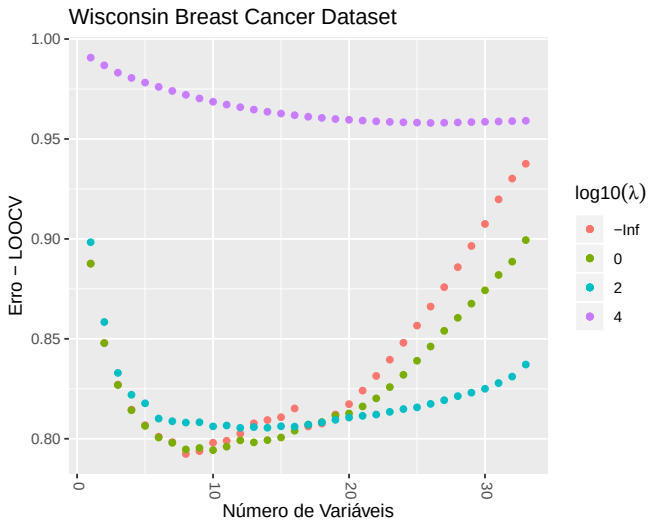
4.1.2 Discussão

Observou-se o decréscimo do erro com o aumento do número de variáveis selecionadas até um ponto de mínimo, a partir do qual a tendência se inverte e o erro passa a aumentar. Tal dinâmica é esperada, uma vez que o aumento do número de parâmetros, na regressão linear, resulta no aumento da complexidade do modelo (HASTIE; TIBSHIRANI; FRIEDMAN, 2017, p. 224).

De fato, ao se reduzir a complexidade do modelo através do aumento do parâme-

tro de regularização (λ), observa-se a diminuição da tendência ao overfitting, conforme demonstra a figura 11.

Figura 11: Efeito do parâmetro de regularização (λ): *Wisconsin Breast Cancer Dataset*.



Fonte: Própria.

É interessante notar que valores de λ reduzidos tiveram pouca ou nenhuma influência nos primeiros subconjuntos ótimos selecionados pelo algoritmo, alterando apenas a seleção na região com tendência ao overfitting. Dessarte, a utilização de um parâmetro de regularização não nulo de pequena magnitude tem pouca influência no conjunto selecionado, enquanto melhora a estabilidade numérica do método, evitando operações com matrizes singulares.

4.2 Estudo de Caso: Vallourec - Caracterização de tubos de aço

5 Conclusão

Referências

ABHISHEK, N. *Machine Learning: A Gentle Introduction. – Towards Data Science*. Towards Data Science, 2018. Disponível em: <https://towardsdatascience.com/machine-learning-a-gentle-introduction-17e96d8143fc>. Acesso em: 12/11/2018. Citado na página 16.

BASILEVSKY, A. *Applied matrix algebra in the statistical sciences*. [S.l.]: Dover Publications, 2005. Citado na página 24.

BISHOP, C. M. *Pattern recognition and machine learning*. [S.l.]: Springer, 2006. Citado 3 vezes nas páginas 18, 21 e 22.

BURMAN, P. A comparative study of ordinary cross-validation, v-fold cross-validation and the repeated learning-testing methods. *Biometrika*, v. 76, n. 3, 1989. Citado na página 23.

CAWLEY, G. C.; TALBOT, N. L. On over-fitting in model selection and subsequent selection bias in performance evaluation. *J. Mach. Learn. Res.*, JMLR.org, v. 11, p. 2079–2107, ago. 2010. ISSN 1532-4435. Disponível em: <http://dl.acm.org/citation.cfm?id=1756006.1859921>. Citado na página 22.

CRUELLS, E. B. i. *Bias and Variance*. 2017. Disponível em: <http://www.ebc.cat/2017/02/12/bias-and-variance/>. Acesso em: 12/11/2018. Citado na página 17.

GLEESON, P. *The Curse of Dimensionality*. Medium, 2017. Disponível em: <https://medium.freecodecamp.org/the-curse-of-dimensionality-how-we-can-save-big-data-from-itself-d9fa0f872335>. Acesso em: 31/10/2018. Citado na página 18.

HASTIE, T.; TIBSHIRANI, R.; FRIEDMAN, J. H. *The elements of statistical learning: data mining, inference, and prediction*. 2. ed. [S.l.]: Springer, 2017. Citado na página 37.

HAYASHI, F. *Econometrics*. Princeton: Princeton University Press, 2001. ISBN 9781400823833. Citado na página 24.

HAYKIN, S. S. *Neural networks and learning machines*. 3. ed. [S.l.]: Prentice Hall, 2009. Citado na página 16.

JAMES, G. et al. *An introduction to statistical learning with applications in R*. [S.l.]: Springer, 2017. Citado 3 vezes nas páginas 18, 19 e 30.

MITCHELL, T. M. *Machine learning*. [S.l.]: McGraw-Hill, 1997. Citado na página 15.

NACE International. *NACE Standard TM0177-2005 Item No. 21212*. 2005. Citado na página 33.

ROSASCO, P. *Machine Learning: a Regularization Approach: Chapter 7 - Online Learning*. [S.l.]: MIT-9.520 Lectures Notes, Manuscript, 2015. Citado na página 25.

RUSSELL, S. J.; NORVIG, P. *Artificial intelligence - A Modern Approach*. 3. ed. [S.l.]: Prentice Hall, 2010. Citado na página 15.

SEBER, G. A. F.; LEE, A. J. *Linear regression analysis*. 2. ed. [S.l.]: Wiley, 2012. Citado 2 vezes nas páginas 23 e 24.

SRINIDHI, S. *Different types of Validations in Machine Learning (Cross Validation)*. 2018. Disponível em: <https://blog.contactsunny.com/data-science/different-types-of-validations-in-machine-learning-cross-validation>. Acesso em: 31/10/2018. Citado na página 23.

TRUNK, G. V. A problem of dimensionality: A simple example. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-1, n. 3, p. 306–307, July 1979. ISSN 0162-8828. Citado na página 18.