Pontifícia Universidade Católica
do Rio de Janeiro

Alexandre Werneck Andreza

Aprendizado em dois estágios para métodos de comité de árvores de decisão

Dissertação de Mestrado

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre pelo Programa de Pós–graduação em Informática da PUC-Rio.

Orientador: Prof. Marcus Vinicius Soledade Poggi de Aragão

Rio de Janeiro
Maio de 2020

Pontifícia Universidade Católica
do Rio de Janeiro

Alexandre Werneck Andreza

Aprendizado em dois estágios para métodos de comité de árvores de decisão

Dissertação apresentada como requisito parcial para a obtenção do grau de Mestre pelo Programa de Pós–graduação em Informática da PUC-Rio. Aprovada pela Comissão Examinadora abaixo.

Prof. Marcus Vinicius Soledade Poggi de Aragão
Orientador
Departamento de Informática – PUC-Rio

Prof. Eduardo Sany Laber
Departamento de Informática – PUC-Rio

Prof. Thibaut Victor Gaston Vidal
Departamento de Informática – PUC-Rio

Rio de Janeiro, 18 de Maio de 2020

Alexandre Werneck Andreza

Graduado em Informática pela Pontifícia Universidade Católica, PUC-Rio (2016). Atualmente trabalhando no SGU, Sistema de Gerência Universitária da PUC-Rio.

Ficha Catalográfica

Andreza, Alexandre Werneck

Aprendizado em dois estágios para métodos de comité de árvores de decisão / Alexandre Werneck Andreza; orientador: Marcus Vinicius Soledade Poggi de Aragão. – Rio de janeiro: PUC-Rio, Departamento de Informática , 2020.

v., 115 f: il. color. ; 30 cm

Dissertação (mestrado) - Pontifícia Universidade Católica do Rio de Janeiro, Departamento de Informática .

Inclui bibliografia

1. Departamento de Informática – Teses. 2. Ciência de Dados – Teses. 3. Aprendizado de máquina;. 4. Métodos de floresta;. 5. Construção de características;. 6. Previsão otimizada. I. Poggi de Aragão, Marcus Vinicius Soledade. II. Pontifícia Universidade Católica do Rio de Janeiro. Departamento de Informática . III. Título.

CDD: 004

Agradecimentos

Agradecimentos especiais ao meu orientador Marcus Poggi por todo o conhecimento compartilhado, pelo incentivo e paciência ao longo dessa caminhada. Igualmente, a todos do Galgos pelas experiências acadêmicas divididas.

Obrigado à equipe do SGU pelo incentivo e compreensão nos momentos em que os estudos exigiam maior dedicação. Agradeço meus gestores, Gustavo Miranda e Floriano Mazini por acreditarem nesse objetivo e aceitarem as adaptações de horários necessárias nesse período.

Aos meus familiares e a minha noiva, que sempre e irrestritamente me apoiaram.

O presente trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001

Resumo

Andreza, Alexandre Werneck; Poggi de Aragão, Marcus Vinicius Soledade. **Aprendizado em dois estágios para métodos de comité de árvores de decisão**. Rio de Janeiro, 2020. 115p. Dissertação de Mestrado – Departamento de Informática , Pontifícia Universidade Católica do Rio de Janeiro.

Tree ensemble methods são reconhecidamente métodos de sucesso em problemas de aprendizado supervisionado, bem como são comumente descritos como métodos resistentes ao *overfitting*. A proposta deste trabalho é investigar essa característica a partir de modelos que extrapolem essa resistência. Ao prever uma instância de exemplo, os métodos de conjuntos são capazes de identificar a folha onde essa instância ocorre em cada uma das árvores. Nosso método então procura identificar uma nova função sobre todas as folhas deste conjunto, minimizando uma função de perda no conjunto de treino. Uma das maneiras de definir conceitualmente essa proposta é interpretar nosso modelo como um gerador automático de *features* ou um otimizador de predição.

Palavras-chave

Aprendizado de máquina; Métodos de floresta; Construção de características; Previsão otimizada

Abstract

Andreza, Alexandre Werneck; Poggi de Aragão, Marcus Vinicius Soledade (Advisor). **Two-stage learning for tree ensemble methods**. Rio de Janeiro, 2020. 115p. Dissertação de mestrado – Departamento de Informática , Pontifícia Universidade Católica do Rio de Janeiro.

In supervised learning, tree ensemble methods have been recognized for their high level performance in a wide range of applications. Moreover, several references report such methods to present a resistance of to overfitting. This work investigates this observed resistance by proposing a method that explores it. When predicting an instance, tree ensemble methods determines the leaf of each tree where the instance falls. The prediction is then obtained by a function of these leaves, minimizing a loss function or an error estimator for the training set, overfitting in the learning phase in some sense. This method can be interpreted either as an Automated Feature Engineering or a Predictor Optimization.

Keywords

Machine Learning; Ensemble methods; Feature construction; Optimizer prediction

Sumário

Lista de figuras

Lista de tabelas

Lista de Abreviaturas

2PL – Two Phase Learning
RF – Random forest
XG – XGBoost – Extreme gradient boosting
SVM – Singular vector machine
RBF – Radius basis function
SVM-R – Singular vector machine – rbf
SVM-L – Singular vector machine – linear
NN – Neural network
MSE – Mean squared error
MAE – Mean absolute error
ACC – Accuracy
ROC – Receiver operating characteristic
AUC – Area under the curve ROC
BL – Binary Leaves
EL – Estimation leaves
LV – Leaf value
PV – Path value

1
Introdução

A capacidade de inferir resultados, a partir do conhecimento existente, é uma das grandes motivações das pesquisas em aprendizado de máquina, tarefa que também pode ser entendida como *pattern recognition*, ou reconhecimento de padrões. De forma geral, os algoritmos, que resolvem essa classe de problemas, devem estimar uma função que represente este conhecimento a partir das características dos dados, ou comumente definidas como *features*.

Estes algoritmos são compostos por dois estágios fundamentais [15], a inferência do modelo, quando a partir do treinamento sobre um conjunto de dados uma função f fica definida e a fase subsequente de decisão, quando valores de estimativas são obtidos para dados desconhecidos inicialmente. Uma das abordagens utilizadas é a construção de árvores de decisão, modelo que permite reduzir a complexidade de uma decisão, em torno das *features* do problema, em um conjunto de estruturas condicionais sequenciais. O presente trabalho utiliza como referência o modelo de árvores conhecido como *CART*, definidas por Brieman [5], e portanto, todas essas condições são binárias, ou seja, a cada passo o algoritmo permite seguir apenas dois caminhos.

A combinação de métodos é uma outra abordagem, com resultados usualmente superiores às estimativas de modelos simples [2]. Esta metodologia é referenciada na literatura como *ensemble methods*. A motivação principal dessa abordagem se deve ao fato que a combinação dos modelos permite a correção de erros individuais [2].

Sendo assim, por óbvio, os *ensemble methods* podem considerar a combinação de modelos a partir de árvores de decisão, tornando-se *tree ensemble methods*, ou métodos de conjuntos de árvores. Diferentes formas de tratar essa união de modelos são definidas na literatura[15], algumas através da manipulação do conjunto de treinamento, seja pela remontagem de *features* ou pela randomização aleatória dos dados, e outras ainda adaptando os valores esperados de saída, promovendo treinamentos mais ajustados.

As florestas aleatórias, algoritmo conhecido como *Random Forest*, e as árvores impulsionadas pelo gradiente, denominada na literatura como *gradient tree boosting*, são abordagens tradicionais para a construção de modelos de conjuntos com árvores de decisão. Estes modelos, a partir de diferentes

heurísticas, promovem a combinação de múltiplas decisões a fim de melhor estimarem os dados, especialmente no conjunto de teste. Denomina-se conjunto de teste, o subconjunto formado a partir das informações originais, não utilizadas durante a construção dos modelos, destinado à avaliação de desempenho dos algoritmos.

Uma das grandes vantagens dos métodos de conjunto de árvores, citados acima, é a resistencia que apresentam à alta variância, em outros termos, eles são capazes de preservar uma margem de ajuste, ou tolerância ao erro de estimativa, mesmo que novas árvores sejam construídas. Essa capacidade, definida na literatura como resistência ao *overfitting* [3] [12], é uma característica fundamental ao bom desempenho dos métodos, prevenindo que os modelos piorem a precisão, na média, quando treinados em conjuntos maiores.

Nas *random forests*, conforme definido em Brieman, 2001 [3], esse suporte é garantido por um erro de generalização que converge a uma estimativa limitada quando o número de árvores cresce, *"it follows from the Strong Law of Large Numbers"*. A partir disso, entende-se que aumentar a quantidade de árvores indefinidamente em uma *random forest* não necessariamente resulta em melhoria do modelo.

Por outro lado, as *boosted trees* são limitadas por um parâmetro de regularização, aplicado igualmente a cada estimador. Esse controle pode ser interpretado como um valor de tolerância dos passos de iteração, garantindo um número mínimo de árvores e diminuindo o peso de cada correção sugerida pelas árvores de decisão.

Random Forest e *Gradient Tree Boosting* e algumas extensões estão entre as metodologias mais competitivas em desafios de aprendizado de máquina [7][16].

De forma geral, qualquer destes *tree ensemble methods*, é conceitualmente caracterizado como uma construção obtida a partir de um conjunto de *weak learners*, ou estimadores fracos, e uma estimativa final considerando uma função sobre estes resultados do conjunto. Portanto, as diferenças entre os algoritmos analisados neste trabalho, *random forest* e *gradient tree boosting*, podem ser reduzidas às fases de treinamento, ou a construção das árvores de decisão, e a função estimadora do valor esperado.

As árvores das *random forests* dispõem da possibilidade de serem construídas em paralelo, ou seja, os seus aspectos individuais não possuem correlação entre si. A premissa fundamental desse conjunto é que seja gerado, tendo como base subconjuntos randomizados e independentes. Estes subconjuntos são formados a partir dos dados de entrada do algoritmo. Ao final, espera-se uma decisão coletiva, onde cada árvore irá contribuir igualmente para a

estimativa do modelo.

A base de construção do conjunto de estimadores fracos no *gradient tree boosting* é a execução de passos incrementais, quando uma estimativa influencia diretamente na geração das próximas árvores. Uma *loss function*, a cada passo de iteração, irá redefinir os novos parâmetros em $i+1$, penalizando os exemplos classificados incorretamente pela árvore de decisão i. Esse controle é fundamental para o desempenho apurado desse modelo, que ao destacar um erro de estimativa, permite que os passos subsequentes façam as correções necessárias.

O método proposto neste trabalho procura redefinir essas duas etapas dos *ensemble methods*, propondo uma alternativa de *feature construction* utilizando as folhas das árvores de decisão e uma nova função estimadora, aplicada ao conjunto dos *weak learners*. O método resultante pode ser interpretado como gerador automático de *features*, *Automated Feature Engineering*, ou preditor otimizado, *Predictor Optimization*.

1.1 Motivação

As folhas das árvores de decisão representam o conhecimento obtido durante a fase de treinamento. Cada folha, aqui denominada por l, contém o peso das distribuições entre as classes, dado pelos exemplos que ocorrem em l, ou a estimativa média de valores reais, definido pelas instâncias em l, nos problemas de regressão. Entretanto, algumas dessas informações podem ser ignoradas pelos estimadores no conjunto de teste, levando em consideração que instâncias podem não ocorrer em determinados nós finais e que um único exemplo ocorre apenas em um nó folha de cada árvore de decisão. Nossa intuição é que a generalização final de um *ensemble method* possa ser potencializada, considerando todas as folhas de todas as árvores de decisão ao prever uma instância.

Colaborando com essa perspectiva, em exemplos de classificação, observamos empiricamente que algumas folhas possuem distribuições não homogêneas, enquanto outras representam apenas uma classe do problema. O contorno dessa inconsistência é obtido de forma gulosa, deixando as árvores completas, contendo apenas nós finais com classes exclusivas. Contudo, essa liberdade provoca um excesso de ajuste no modelo, causando o problema de *overfitting* local.

Com isso, uma das motivações desse trabalho é investigar pesos ótimos nas folhas das árvores de decisão. Considerando uma eventual sobrecarga de dados, o modelo proposto deseja avaliar os benefícios dessa abordagem, a

partir de métodos já resistentes ao *overfitting*, em comparação a utilização das *features* originais.

Este trabalho é motivado também por tentar redefinir a tarefa de predição, muitas vezes reduzidas a operações simplificadas nos métodos de conjunto, como a média e/ou a soma dos resultados. A título de comentário, no algoritmo de *Random Forest* o resultado é dado pelo voto da maioria [3] e no *Gradient Tree Boosting*, de forma geral, a estimativa considera a soma dos valores preditos nos passos anteriores [12].

Uma árvore de decisão, em especial no problema de classificação, é algumas vezes representada graficamente no espaço de duas dimensões como uma grade, onde cada subconjunto representa uma classe do problema. Para exemplificar essa estrutura, utilizando um conjunto de dados artificial, foi gerada a imagem abaixo, onde as classes $1, 2$ do problema são representadas pelas cores vermelho e azul, respectivamente. Os eixos x e y representam as duas *features* desse conjunto artificial.

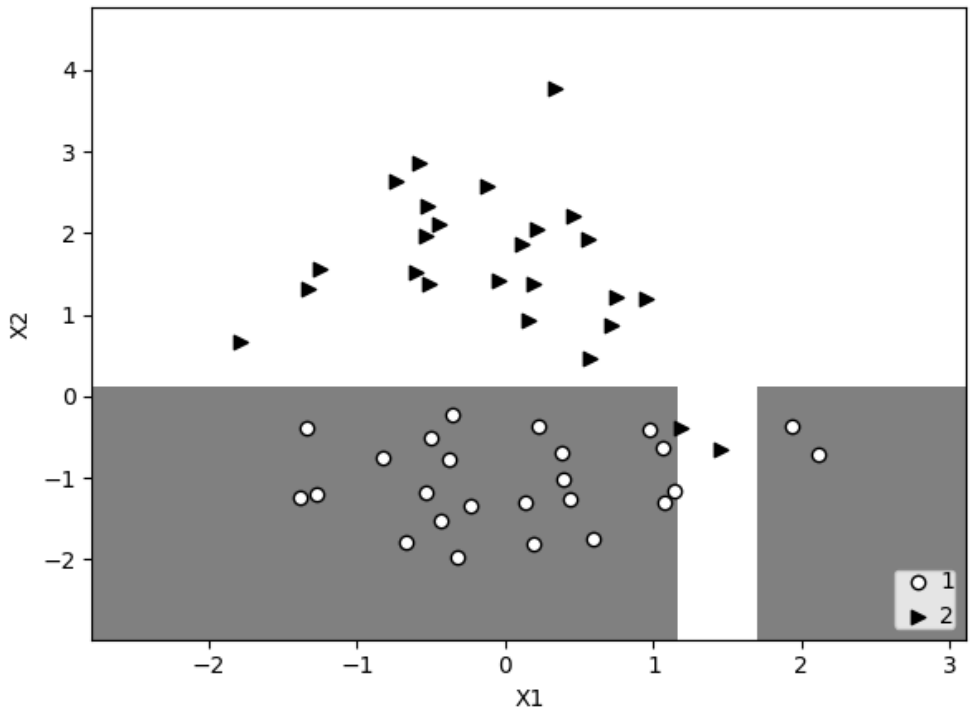

Figura 1.1: Representação em superfície da árvore de decisão.

Extrapolando essa análise para os *tree ensemble methods*, podemos visualizar como resultado a sobreposição de vários gráficos, semelhantes aos exibidos na imagem 1.1.

Dessa maneira, algumas bordas e subconjuntos serão sobrepostos. A falta de flexibilidade, nas operações dos preditores, impede qualquer correção na

classificação final. Busca-se uma função ótima que acrescentando algum grau de correção melhore as estimativas. Cabe ressaltar que, apesar dos modelos de *boost* já promoverem correções individuais, o interesse nessa abordagem fica restrito à função final de predição.

1.2 Contribuições desta dissertação

A estrutura obtida como resultado dos algoritmos de árvores de decisão, apresenta um elevado grau de simplicidade em compreensão, por isso são comumente utilizadas na representação do conhecimento em diversas áreas. Essa característica do modelo de árvores, somado a disponibilidade de serem combinados nos *ensemble methods*, tornaram essa metodologia bastante utilizada em pesquisas de aprendizado de máquina. Apesar deste amplo conhecimento e difusão, novos trabalhos, utilizando árvores de decisão, permanecem sendo elaborados, sugerindo novas oportunidades de pesquisa.

Das árvores de decisão, em especial das informações das folhas, este trabalho propõe um modelo gerador de características, na literatura de *machine learning* este gerador pode ser entendido como *feature construction*. Essas características, ou *features*, representam estes nós finais, as folhas, bem como seus respectivos pesos.

Experimentos demonstram que a utilização dessas *features* para um conjunto com poucas árvores, elevam a acurácia do modelo, tornando-o comparável com os mesmos algoritmos utilizando um número maior de árvores. Os resultados deste trabalho atestam também que a inclusão de uma função ótima como preditora, combinado às novas *features*, melhora os resultados gerais, na média dos *datasets* analisados.

1.3 Estrutura da dissertação

Este trabalho foi organizado da seguinte forma. No próximo capítulo, apresentamos uma revisão de aprendizado supervisionado, reduzida aos assuntos de interesse deste trabalho. As métricas usadas por experimentos e resultados finais são apresentadas neste capítulo. No capítulo 3, ainda apresentamos algumas revisões, desta vez sobre métodos de conjunto, ou *ensemble methods*, *random forest* e *gradient tree boosting*, além da apresentação do algoritmo *XGBoost*. Finalmente, as próximas duas seções explicarão nosso método. No capítulo 4, a metodologia é apresentada e, a seguir, no capítulo 5, um conjunto de algoritmos é definido. O capítulo 6 concentra-se em experimentos e

resultados obtidos pela metodologia proposta. O último capítulo é dedicado a conclusão a partir de discussões, ideias e trabalhos futuros.

2
Aprendizado supervisionado

O aprendizado supervisionado considera, como premissa básica, a existência de um conjunto de entradas previamente classificadas. Algoritmos especilistas devem, a partir dos pares de exemplos *input-output*, 'aprender' uma função que mapeia qualquer entrada em uma saída. Estamos interessados em uma função $f(x, y)$ que estime um valor de y, dado x.

Sendo assim, define-se o aprendizado supervisionado da seguinte forma:

> Dado (X, Y), um conjunto de exemplos de treinamento de tamanho N, formado pelos pares $\{(x_i, y_i), ..., (x_N, y_N)\}$, em que x_i é o i_{th} exemplo e y_i é uma resposta conhecida para x_i. Um algoritmo supervisionado procura uma função $f : X \rightarrow Y$, de modo que f tenha y esperado como o valor de saída. X é o conjunto de valores de entrada dos exemplos e Y é o conjunto de valores de saída dos exemplos.

A função f é denominada, no aprendizado supervisionado, como a hipótese, ou o preditor do problema. Esta função, ao final, será um estimador que mapeia novos dados de entrada x', desconhecidos inicialmente, em valores y'.

Soluções clássicas de *machine learning*, como as máquinas de vetores de suporte, ou svm, as regressões lineares e as árvores de decisão estão contidas neste escopo. Apesar de alguns métodos apresentarem desempenho melhor em uma grande gama de conjunto de dados, não existe uma solução única que funcione em todas as tarefas. Cada *dataset*, bem como a solução esperada, impõem abordagens diferentes, usando algum destes algoritmos ou ainda uma combinação deles.

Um dos fatores de sucesso, a ser observado, na construção de algoritmos para o aprendizado supervisionado é o controle frequente entre o baixo viés e baixa variância do modelo. Essa relação é classicamente descrita na literatura como o *trade-off: low bias x low variance*. De outra forma, a mitigação de alto viés e alta variância preservam esse controle.

O problema de alto viés, ou *high bias*, é conhecido como *underfitting*, quando um modelo estimador não representa com qualidade a correlação entre as *features* do problema e os valores de saída esperados. Modelos como

esses tendem a gerar funções pouco representativas dos dados. Por outro lado, um excesso de correlação tende a gerar modelos muito ajustados ao conjunto de treinamento, consequetemente, pouco flexíveis a novos exemplos. Esse problema é definido como alta variância, ou *high variance*, e é conhecido como *overfitting*. Dessa forma, tradicionalmente, os modelos clássicos de aprendizado supervisionado definem mecanismos de controle, especialmente para o problema de *overfitting*.

As tarefas de aprendizado supervisionado e os conjuntos de dados para experimentos são geralmente agrupados em duas classes de problema, classificação e regressão. A seguir, as especificidades de cada uma delas são apresentadas.

2.1 Classificação

Em problemas de classificação, Y é definido como um conjunto finito, restrito a um grupo de classes discretas, de tamanho K. Dessa forma, $Y = \{y_1, ..., y_k\}$. A função, ou hipótese, que estima valores de y é chamada de classificador. Cada classificador deve ser capaz de prever um, e apenas um, resultado, dentro do conjunto de valores em Y.

As seções seguintes apresentam medidas comumente utilizadas por algoritmos de classificação. São métricas que analisam o desempenho nos conjuntos de treinamento e teste.

2.1.1 Medidas para algoritmos de classificação

Accuracy é uma medida de precisão do modelo, representa o quão próximos estão os resultados obtidos e os valores esperados. É a métrica mais utilizada na validação de desempenho dos algoritmos de classificação. A acurácia é obtida pela divisão entre a soma das classificações corretas e o número de exemplos estimados.

Considerando $\hat{y}$ os valores estimados em f, n_{samples} o número de exemplos e y os valores esperados, temos o cálculo da *accuracy* dado por:

$$\texttt{accuracy}(y, \hat{y}) = \frac{1}{n_{\text{samples}}} \sum_{i=0}^{n_{\text{samples}}-1} 1(\hat{y}_i = y_i)$$

ROC Curve – Receiver Operating Characteristic Curve, ou curva ROC, é uma representação gráfica dos valores obtidos por uma hipótese, utilizada em *datasets* de classificação binária. Essa curva é a projeção da razão dos positivos verdadeiros (*True positive rate - TPR*), dada por *true positives / (true positives + false negatives)* e a razão dos falsos positivos (*False positive rate - FPR*),

dada por *false positives / (false positives + true negatives))*. A curva ROC permite medir o custo/benefício de uma função estimada, já que ela fornece a visualização de erros e acertos para todas as classes do problema.

AUC - Area under the curve ROC, ou a área abaixo da curva ROC, é derivada da curva ROC e tenta reduzir a projeção gráfica a uma estimativa de valor entre 0 e 1. Com isso, diferentes valores de AUC, obtidos a partir de diferentes hipóteses, podem ser comparados. Quanto mais próximos a 1, melhores são os modelos e a sua capacidade em separar corretamente as duas classes. Por outro lado, valores próximos ou abaixo de 0.5 representam a dificuldade da hipótese em classificar os exemplos nas classes corretas.

Considerando três estimadores quaisquer com valores de *AUC* em 0.5, 0.75 e 1.0 respectivamente, as curvas *ROC* desses modelos são dadas na imagem abaixo. Um estimador capaz de separar corretamente as classes do problema tem *AUC* igual a 1 e os pontos de estimativas que ocorrem no espaço *ROC* estão na coordenada (0,1). Um modelo com essa característica classifica os dados com 100% de especificidade, sem falsos positivos, e 100% de sensibilidade, sem falsos negativos.

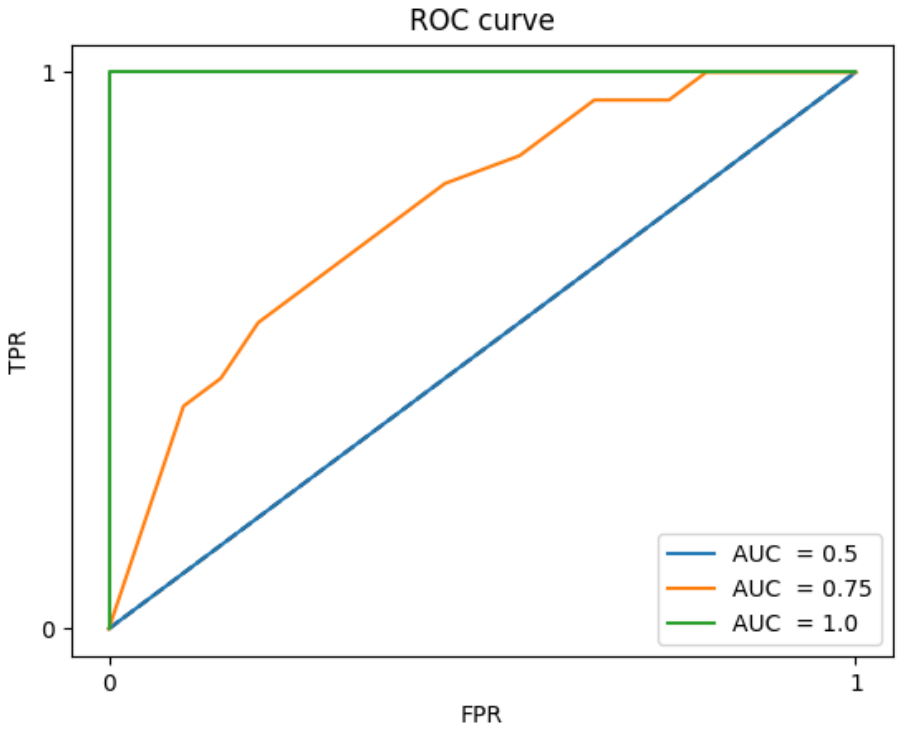

Figura 2.1: Curva *ROC* para valores de *AUC*: 0.5, 0.75 e 1.0

2.2 Regressão

O *output* de qualquer algoritmo de regressão é dado por um valor real, dessa forma, Y é um conjunto de valores não discretos. Novas métricas são apresentadas para estimarem o desempenho desses algoritmos, calculando as diferenças reais entre os valores obtidos e os valores esperados.

2.2.1 Medidas para algoritmos de regressão

As medidas comumente utilizadas para avaliar o desempenho desses algoritmos, e que fazem parte do escopo deste trabalho, são:

MAE - Mean absoluted error, ou erro absoluto médio, é dado pela divisão da soma das diferenças absolutas entre o valor esperado e o valor obtido e o número de exemplos.

$$\mathrm{MAE}(y, \hat{y}) = \frac{1}{n_{\mathrm{samples}}} \sum_{i=0}^{n_{\mathrm{samples}}-1} |y_i - \hat{y}_i| .$$

MSE - Mean squared error, ou erro médio quadrático, é dado pela divisão da soma das diferenças quadráticas entre o valor esperado e o valor obtido e o número de exemplos. MSE potencializa as falhas superiores à média dos erros.

$$\mathrm{MSE}(y, \hat{y}) = \frac{1}{n_{\mathrm{samples}}} \sum_{i=0}^{n_{\mathrm{samples}}-1} (y_i - \hat{y}_i)^2.$$

2.3 Árvore de Decisão (*Decision trees*)

Conforme já mencionado, as árvores de decisão são uma maneira de inferir funções para os problemas de aprendizado supervisionado. De forma geral, essa abordagem consiste na divisão de uma tarefa complexa, com grande número de *features*, em pequenas soluções simples, a partir de características individuais dos dados. Qualquer algoritmo irá produzir, recursivamente, subconjuntos que apresentam as predições semelhantes, por isso, quanto maior o número de nós, maior será a especificidade desses espaços.

Toda árvore de decisão é composta por nós internos, ou nós de decisão, e nós finais, ou nós folhas, conectados por regras condicionais. Por isso, uma outra maneira de entender os algoritmos geradores de *decision trees* é considerá-los como um conjunto formado a partir de sequências de estruturas condicionais $if - then - else$. Nós finais contém as estimativas de classes para os problemas de classificação e valores reais na regressão.

Uma *decision tree* pode ainda ser representada como um grafo direcionado, dado por $G = (V, E)$. V é o conjunto de nós do grafo, ou seja, os nós de

decisão e os nós folhas e E é o conjunto de pares (v_i, v_j) que representam um caminho na árvore de decisão.

Muitos algoritmos foram propostos para o problema de construção das árvores de decisão, tendo como foco principal a procura pelos melhores *split points*, ou pontos de ruptura, que representam os nós de decisão. Cada ponto de divisão é alcançado considerando a combinação, entre uma *feature* do conjunto de dados e um valor de *threshold*, que divide os dados de forma mais homogênea. Essa abordagem pode não alcançar o ótimo global, ou seja, é possível existir uma outra árvore de decisão que melhor separe os dados. Ótimos locais não garantem a divisão ótima de todo conjunto de dados[8].

O algoritmo utilizado neste trabalho, também suportado pelas bibliotecas de apoio, é conhecido como CART, Classification and Regression Trees, e foi demonstrado em artigo de mesmo nome[5], por Brieman, 1984. A documentação da biblioteca *scikit-learn*[6] define o processo de construção das árvores no algoritmo de *CART* como:

> "CART constructs binary trees using the feature and threshold that yield the largest information gain at each node."

Sendo assim, o ganho de informação, ou *information gain*, é calculado nos problemas de classificação e regressão a cada passo de iteração do algoritmo. A premissa dessa operação é que este valor precisa ser o maior entre os possíveis, *largest information gain*, ou seja, os subconjuntos gerados devem ser homogêneos ao máximo. Para exemplificar essa definição, considerando um típico problema de classificação binária, os subconjuntos serão homogêneos ao máximo quando, em cada um destes, todos os exemplos representam apenas uma classe do problema.

Cada nó, não folha, de uma árvore gerada pelo algoritmo de *CART* é composto por uma *feature*, contida nos exemplos do problema e um valor de *threshold* que permite a sua subdivisão binária. Cada um destes nós carrega uma função, *information gain*, estimada durante a fase de treinamento.

A função de ganho esperado pode ser definida como $Gain(n_x)$, onde n_x é qualquer nó, de uma árvore de decisão, candidato ao particionamento. Define-se então, n_l e n_r como subconjuntos de n_x, ou nós filhos. Um cálculo de impureza pode ser realizado em qualquer desses conjuntos e esse valor é dado por $Imp(n_i)$. A fórmula geral para obtenção do ganho de informação em n_x é dada por:

$$Gain(n) = Imp(n_x) - p_l * Imp(n_l) - p_r * Imp(n_r) \tag{2-1}$$

Da fórmula 2-1, p_l é a distribuição de exemplos no nó n_l em relação ao total de exemplos de n_x, igualmente, p_r é a distribuição de exemplos no nó n_r.

Ganhos ótimos, ou a maximização de $Gain$, devem tender à impureza do nó original, enquanto valores próximos de zero são compatíveis com subconjuntos de distribuições não homogêneas e a divisão implica em baixo ganho de informação.

O problema de encontrar o ganho máximo segue então de estimar as impurezas, dado por $Imp(n_i)$, em cada possível subconjunto a partir de n_x. As tarefas de classificação e regressão definem métodos diferentes para este cálculo, a seguir essas abordagens são definidas.

2.3.1 *Impurity* em algoritmos de classificação

Gini index, ou coeficiente de gini, é uma medida que representa a dispersão dos dados em função das classes do problema, por esse motivo é considerada capaz de inferir a impureza de subconjuntos. Voltando ao exemplo, considerando que Y é um conjunto de classes distintas de tamanho K, a impureza de qualquer nó $Imp(n_i)$ é igual a 1 menos a soma das probabilidades de cada classe na instância do nó n_i. Dessa formulação temos que os valores de impureza dos nós são sempre menores que 1. Valor de impureza igual a zero significa que todos os elementos de um subconjunto pertencem a apenas uma classe. Quanto mais esse valor tende a 1, menos homogêneo o subconjunto se torna.

$$Imp(n_x) = 1 - \sum_{k=1}^{K} p_k^2$$

Entropy é uma medida bastante semelhante à proposta do *gini index*, pois também mede a impureza de subconjuntos. A diferença básica entre os cálculos é na utilização, pela entropia, do valor do log das distribuições das classes nos subconjuntos.

$$Imp(n_x) = -\sum_{k=1}^{K} p_k * \log p_k$$

O algoritmo de $CART$ da biblioteca scikit-learn[6], por padrão, utiliza o *gini index* no cálculo de impureza dos nós.

2.3.2 *Impurity* em algoritmos de regressão

A função de ganho, definida em 2-1, é igualmente compatível com a procura pelos pontos de ruptura ótimos nos algoritmos de regressão. A diferença fica restrita ao cálculo de impureza dos subconjuntos.

As medidas de erro quadrático médio, **MSE**, 2.2.1, ou o erro absoluto médio, **MAE**, 2.2.1, são as soluções utilizadas, para esse cálculo, nas tarefas em que o conjunto Y é formado por valores contínuos.

3
Ensemble methods - Métodos de conjunto

Ao definir o aprendizado supervisionado, f, a função estimadora, fica caracterizada como a melhor hipótese. Os métodos de conjunto, que utilizam uma coleção de árvores de decisão, definem f a partir de um comitê formado por n hipóteses diferentes, ótimas localmente.

Sendo assim, dado H, um conjunto das n hipóteses, caracterizadas como *weak learners*, ou estimativas fracas, na forma de $h_i(X) = Y$, a função estimadora nos *ensemble methods* fica redefinida como $f(H(X), Y)$. Cada algoritmo possui uma operação própria aplicada ao comitê para obter o valor predito.

Ensemble algorithms, como *Random Forest* e *Gradient Tree Boosting*, são formados por esses dois processos, a construção das hipóteses utilizando árvores de decisão e a função de decisão, ou comitê, aplicada ao conjunto de estimadores. Este comitê, nos algoritmos citados, tem como diferença básica a atribuição de pesos a cada estimador. Apesar desse conceito não aparecer explicitamente na definição dos métodos, ele serve a este trabalho por encapsular $f(H(X), Y)$ em um termo comum aos algoritmos. Com isso, f é dado em função do somatório de todas hipóteses multiplicadas pelo seu peso w_i.

$$f(X, Y) = w_i * h_i(x_i) + ... + w_n * h_n(x_n)$$

Considere um típico problema de classificação binária, em que a decisão do comitê é dada pelo voto da maioria. Dessa forma, $Y = \{-1, 1\}$ e cada peso w_i é igual a $1/m$, onde m é o número de hipóteses ou o tamanho de H. A função f é então igual ao somatório das hipóteses dividido por m.

$$f(X, Y) = \frac{1}{m} \sum_{i=1}^{m} h_i(x_i)$$

Quando $f >= 0$ para qualquer exemplo i, o valor predito é igual a 1, a maioria das hipóteses é igual a 1. Ao contrário, o valor predito é igual a -1 quando $f < 0$.

Decisões coletivas permitem uma correção estatística no modelo. Voltando ao problema da classificação binária e supondo uma hipótese ótima, construída com apenas uma árvore de decisão, que será utilizada como classificador do problema, existe um conjunto de outras árvores ótimas que foram

desconsideradas. Assume-se um risco, inerente ao problema de *machine learning*, que esta árvore apresente erros no conjunto de testes. Se adicionarmos outras árvores este erro pode ser compensado pela decisão coletiva, por isso, os *tree ensemble methods* combinando as diferentes hipóteses tendem a minimizar os erros com um número maior de árvores. Estatisticamente, métodos de conjunto reduzem a variância [9], *low variance*, através da diminuição da média dos erros, encontrados nos m *weak learners*, em relação à média dos valores esperados.

Os *ensemble methods* são capazes também de reduzir o problema de *high bias*, ou alto viés. Modelos com elevado grau de liberdade encontram o problema do *underfitting*, ou seja, diminuem a correlação entre os dados e os valores esperados Y, as estimativas dessas árvores perdem precisão. O aumento na quantidade de *decision trees*, em alguns casos, implica na criação de um espaço maior de funções capazes de representar o problema. Essa sobrecarga de hipóteses gera um ajuste no modelo de conjunto, as distâncias médias dos valores estimados aproximam-se dos valores reais.

Computacionalmente, encontrar o ponto de ruptura ótimo global em uma árvore de decisão é considerado um problema difícil, apesar dos nós internos definerem mínimos locais ótimos para apenas uma característica dos dados. Sendo assim, a melhor hipótese, como definido no problema das *decision trees*, pode não ser alçancada. Esses *local mistakes*, ou erros locais, tendem a serem corrigidos na média através do aumento na quantidade de árvores de decisão.

Este trabalho é fortemente apoiado pelos métodos de conjunto *Random Forest* e *Gradiente Tree Boosting*. A seguir, alguns conceitos destes métodos são definidos, em especial os algoritmos utilizados para construção das árvores de decisão, os parâmetros de ajustes e como a predição final é obtida.

3.1 *Random Forest*

Conforme modelo proposto em Brieman, 2001 [3], as florestas aleatórias são capazes de gerar boas funções de estimativa, randomizando as características para a formação de subconjuntos de treinamento com baixa correlação e formando um comitê de diferentes árvores de decisão.

O artigo de mesmo nome [3], apresenta uma revisão bibliográfica de métodos para a construção de conjuntos de treinamento das árvores de decisão. Algumas abordagens constroem conjuntos aleatórios de treino a partir de X, como o método *bagging* de Brieman, 1996, outras utilizam um número K de nós candidatos aos melhores pontos de divisão, onde K é igual ao número de estimadores desejado, Dietterich, 1998, dentre outras. Por

fim, a abordagem proposta por Amit e Geman, 1997, utilizando o conjunto de dados *written character recognition*, os autores propõem a formação de conjuntos randomizados utilizando os melhores valores de divisão em *geometric features*, ou características geométricas. Esta última, é citada como grande influenciadora no trabalho do autor.

As florestas aleatórias, de forma geral, são construídas a partir da formação de vetores independentes a cada passo k, em uma floresta de K árvores de decisão, Brieman, 2001:

> "The common element in all of these procedures is that for the k^{th} tree, a random vector θ_k is generated, independent of the past random vectors $\theta_1..\theta_{k-1}$ but with the same distribution;..."

Uma hipótese para o problema de floresta aleatória fica então definida em função de θ_i, portanto, $h_i(x_i) = h_i(\theta_{k_i}(X))$. Nas *random forests* cada árvore contribui igualmente para a estimativa final. Nos problemas de classificação, o voto da maioria é a predição do algoritmo:

> "After a large number of trees is generated, they vote for the most popular class. We call these procedures random forests."Brieman, 2001

Em regressão, a predição é dada pela média das hipóteses h_i nas K árvores de decisão:

> "The random forest predictor is formed by taking the average over k of the trees $h(x, \theta_k)$."Brieman, 2001

A garantia da baixa correlação, entre os subconjuntos de treinamento, é fundamental para a geração de árvores independentes e, consequentemente, decisivo no bom desempenho do estimador. Eventuais erros individuais são compensados por outras árvores de decisão, diminuindo o impacto de votos incorretos no comitê.

Random forest utiliza a abordagem de *CART*, apresentada por Brieman[5], na formação das árvores de decisão, bem como, é o método padrão presente na biblioteca de apoio, scikit-learn[6], empregue neste trabalho.

3.2 *Gradient tree boosting*

O *gradient tree boosting* pode ser visto de forma geral como passos de *boosting*, ou impulsionamento, aplicado às *weak trees*, ou árvores de decisão de hipóteses fracas, para potencializar a capacidade de predição do algoritmo. O *gradient*, ou gradiente, serve para indicar a direção desse impulso a cada passo de iteração. Em outros termos, o passo gradiente permite as correções de erros de estimativa em cada árvore de decisão. Uma pré-definição ao *gradient tree boosting* é apresentada por Freund, 1999[10].

> "Kearns and Valiant [30, 31] were the first to pose the question of whether a 'weak' learning algorithm which performs just slightly better than random guessing in the PAC model can be 'boosted' into an arbitrarily accurate 'strong' learning algorithm."

Sendo assim, estamos de posse de duas premissas para qualquer modelo de *boosting*, os *weak leaners* e o aprendizado por reforço. Em geral, uma hipótese fraca é considerada como uma função capaz de inferir resultados um pouco melhores que a escolha aleatória. E depois, o impulsionamento que consiste na marcação de erros do modelo, ou seja, exemplos mal treinados ficam sobreajustados no passo $i + 1$.

O problema do aprendizado por reforço é uma tarefa de otimização. Friedman, 1999[12] o define a partir de F', uma função de X que minimiza os erros de estimativa de f, o valor do comitê dos métodos de conjunto, a partir de uma função de perda, *loss function*.

$$F'(X) = argmin_{f(X)} L(Y, f(X))$$

A função de perda, $L(Y, f(X))$, utilizada por cada algoritmo depende da tarefa em investigação, em geral, problemas de regressão utilizam o erro quadrático, dado por $(Y - f(x))^2$ ou o erro absoluto, dado por $|Y - f(x)|$ e os problemas de classificação, quando classes binárias, utilizam *negative binomial log-likelihood*, dado pelo $\log(1 + \epsilon^{-2y*f(x)})$.

Em seguida, f é redefinda em função de P, onde P é um conjunto de parâmetros, $f(x; P)$. Esse ajuste, proposto por Friedman, permite encapsular as hipóteses fracas em torno de parâmetros que reduzem as perdas locais.

$$f(x; P) = \sum_{m=0}^{M} \beta_m f'(x; a_m)$$

Considerando M, o número de árvores do modelo, um valor de regularização β controla o excesso de ajuste a cada passo do algoritmo. Os valores de a_m são dados nas árvores de decisão pelos pontos de ruptura e pela média dos nós finais de cada árvore.

Dado p_m como o conjunto de valores dos parâmetros no passo m, o problema do algoritmo de *boosting* segue por encontrar os valores locais que minimizam L na função f'. Os pontos de mínimo são dados pelo cálculo do gradiente. Por fim, p_m é obtido pela diferença entre os valores estimados nos $m-1$ passos e o *gradient descent* em m.

$$p_m = p_{m-1} - g_m$$

No *gradient tree boosting*, os passos incrementais permitem a correção do modelo, ao destacar os erros locais dos *weak learners*.

3.2.1 XGBoost - Extreme gradient boosting

XGBoost é uma extensão às árvores impulsionadas de Friedman[12].

O algoritmo *extreme gradient* apresenta alguns ajustes na função de regularização, *"We make minor improvements in the reguralized objective, which were found helpful in practice."*, que promovem um melhor controle do ajuste excessivo. Além disso, *"we provide insights on cache access patterns, data compression and sharding to build a scalable tree boosting system."* que garatem mais robustez do modelo em *datasets* maiores. Em geral, o *XGBoost* funciona muito bem para conjunto de dados com grande número de *features* e exemplos.

XGBoost é um algoritmo de elevada performance em muitas competições de *machine learning* [7], por isso é a escolha deste trabalho como investigação dos métodos de conjunto com árvores de decisão.

3.3 *Feature construction* nos métodos de conjunto

Composição de características a partir dos dados originais aparece na literatura de duas formas, a partir de métodos para redução de dimensionalidade ou utilizando heurísticas para geração de novas *features* por inferências [24].

A primeira abordagem, tradicionalmente utilizando a análise de componentes principais, ou PCA *Principal Component Analysis*, tem por motivação a eliminação de características não representativas do conjunto de dados. Além disso, a observação dos primeiros componentes permite a projeção de dados com alta dimensão em um plano representativo. A escolha por essa técnica permite ainda ganho no tempo de processamento de muitos algoritmos de aprendizado supervisionado.

A metodologia aqui proposta, por outro lado, está interessada na inferência de características por resultado do aprendizado obtido em árvores de

decisão dos métodos de conjunto. A técnica de expansão das features, pode induzir a geração de um novo espaço linearmente separável [24], facilitando encontrar uma função que melhor classifique os dados.

Os capítulos a seguir apresentam as diferentes abordagens utilizadas na construção de novas características a partir das folhas das árvores de decisão. A perspectiva é sempre por *feature induction*, aumentando o número de dimensões em relação aos dados originais.

4
Metodologia

Este trabalho promove a construção de um método de conjunto, a partir da redefinição de dois passos dos algoritmos de *tree ensemble methods*. O primeiro deles, a estrutura das árvores de decisão, especialmente através da aplicação de pesos aos nós folhas. E então, a projeção desses novos dados no problema de predição coletiva. Nossa proposta é encontrar uma nova função f, alternativa às operações finais de decisão, o voto da maioria nas *random forest* ou a soma dos estimadores individuais no *gradient tree boosting*, que melhorem a decisão global.

Considerando D, como a tupla (X, Y), que representa o conjunto de dados em um problema de aprendizado supervisionado, ficam definidos os subconjuntos de D, S e T, as fontes de dados para as fases de treinamento e teste dos algoritmos.

> Dado a tupla $D = (X, Y)$ com n exemplos, onde x_i é um vetor de dimensão d com as suas coordenadas assumindo valores categóricos, reais ou binários. Assume-se $S \subset D$, onde $S = \{x_i, y_i\}$, $i = 1, \ldots, n_s$, uma amostra de D, o *dataset* de treino. Deseja-se obter uma função preditora, $h : x \longrightarrow y$, do conhecimento de S, que minimize o valor esperado de uma função de perda $L(h(x), y)$, nos $out - of - sample$, ou seja, nos exemplos de teste $T \subset D$, onde $T = \{x_i, y_i\}$, $i = n_s + 1, \ldots, n = n_s + n_o$.

O método proposto é constituído de duas fases de aprendizado, por isso *two phase learning*, abreviado para **2PL**. Os tópicos a seguir cobrem essas duas etapas.

4.1
Nós folhas dos *tree ensemble methods* como *features*

Ao apresentar os conceitos iniciais dos algoritmos, que utilizam um conjunto de árvores de decisão, duas condições foram introduzidas em torno dos nós folhas. Alguns nós podem não ser atingidos por nenhum exemplo do conjunto de testes e para cada exemplo de estimativa apenas uma folha é alcançada em cada árvore. O processo de construção das novas *features*, neste

trabalho, utiliza a hipótese de que a função que mapeia os valores dos nós folhas na previsão estimada, pode não se caracterizar como um preditor ótimo.

Dado $\mathcal{A}$, um *tree ensemble method* qualquer, que mapeia o conjunto de treinamento S para um conjunto de folhas $\ell(S)$, temos m igual ao número de árvores que $\mathcal{A}$ gera a partir de S e l_t o número de folhas na árvore de decisão t. Dado ainda v_i, como um vetor de componentes, onde cada elemento na forma v_i^{tj} representa o valor da folha j na árvore t, temos v_i com dimensão igual ao número de folhas, $|\ell(S)| = \sum_{t=1}^{m} l_t$. Sendo assim, $\mathcal{A}$ pode ser compreendido como um modelo que faz o mapeamento de qualquer instância x_i em um vetor correspondente v_i.

$$\mathcal{A} = x_i \rightarrow v_i$$

Os elementos do vetor v_i, v_i^{tj}, são mapeados como uma função de *features* x_i. Este recurso é definido como g, uma função dada por $g(i, j, t, S)$, ou ainda, $v_i^{tj} = g(i, j, t, S)$.

$$v_i^{tj} = g(i, j, t, S)$$

Portanto, dado $\mathcal{X}$, $\mathcal{A}$ e S, o conjunto de dados $\mathcal{V}$, contendo tuplas $\{v_i, y_i\}$, $i = 1, \ldots, n = n_s + n_o$ é determinado diretamente e considerado neste modelo como a primeira fase do treinamento.

$\mathcal{V}$ corresponde aos dados em $\mathcal{X}$ como um novo conjunto de *features*. Procura-se então, um preditor $f : v \longrightarrow y$ que minimize o valor esperado de uma função de perda $L(f(v), y)$. A função f, treinada em $\mathcal{V}$, representa a segunda etapa do aprendizado.

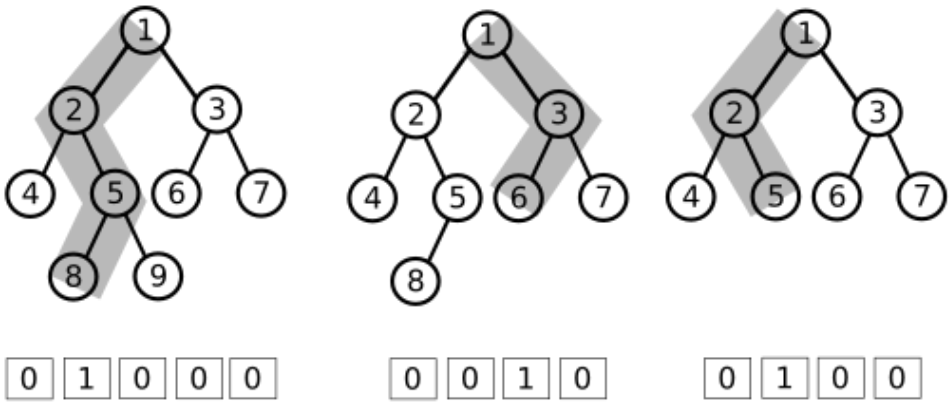

Figura 4.1: *Feature induction* por um conjunto de árvores de decisão. Os valores 0 e 1 são aplicados às folhas das árvores de decisão.

Portanto, pode-se compreender o método de conjunto de árvores, conforme definido no modelo $\mathcal{A}$, como um procedimento automatizado de geração

de *features*. As *features* do conjunto de dados original D, classificam uma instância i conforme uma configuração, dentre as configurações possíveis, $Pi_{t=1}^{m} l_t$, induzidas pelas m árvores de decisão fornecidas pelo método. Mesmo em conjuntos com m baixo, é garantido que para cada configuração exista pelo menos uma instância, sendo assim, um ajuste perfeito do conjunto de treinamento é possível, ou seja, o modelo pode alcançar o *overfitting*.

Esse processo de conversão dos dados, em $\mathcal{A}$, segue por reduzir a sensibilidade de um *ensemble method* às *features* cujo domínio de valores é muito largo, ou o intervalo entre os valores é muito grande, achatando condições a serem avaliadas a um número finito, relativamente pequeno, para determinar a configuração em que ocorre uma instância.

Essas principais características do conjunto de dados traduzido $\mathcal{V}$ são exploradas pelo preditor.

4.2 Otimização e sobreajuste do preditor

A partir da formação de um conjunto de árvores de decisão, uma função $\mathcal{F}$ que mapeia as instâncias v_i, para valores $\overline{y}_i = \mathcal{F}(v_i)$ de predição, pode ser estimada ou aproximada. Encontrar $\mathcal{F}$ incorre em um problema típico de aprendizado supervisionado, por isso, qualquer algoritmo de machine learning, para tarefas supervisionadas, pode ser considerado.

Uma importante característica a ser observada é que todas as abordagens neste trabalho avaliam a resistência à alta variância induzida nos preditores, ao minimizar a função de perda nas amostras S, dadas por $\mathcal{V}$. Essa resistência é compreendida na literatura como a capacidade dos algoritmos em manterem boas estimativas, evitando um excesso de correlação entre os dados de entrada e os valores de saída. Por outro lado, as funções de $\mathcal{F}$, na metodologia proposta, precisam promover este ajuste extremo em $\mathcal{V}$ para melhor estimarem o conjunto de teste. A avaliação utilizada para definir o excesso de ajuste é dado pela acurácia igual a 1 nos problemas de classificação e pelos erros médio e absoluto tendendo a 0 nas tarefas de regressão. De forma geral, $\mathcal{F}$, como um problema de otimização, deve ser o preditor ótimo no *dataset* de treino com as novas *features*.

O método $2PL$ fica assim definido. Resumidamente, consiste em aplicar um conjunto de dados S, obtidos em D original, a qualquer método de conjunto de árvores $\mathcal{A}$, a fim de obter um novo conjunto de dados transformados $\mathcal{V}$. Em seguida, qualquer outro método de aprendizado supervisionado $\mathcal{F}$ é aplicado em $\mathcal{V}$. O $2PL$ requer as escolhas de $\mathcal{A}$, $g(i, j, t, S)$ e $\mathcal{F}$. Diferentes propostas de combinações são realizadas.

5
Algoritmos

Esta seção apresenta os algoritmos utilizados no experimento, instâncias do modelo proposto $2PL$. Primeiro, descrevemos as funções $g(i, j, t, S)$, em tarefas de classificação e regressão, usadas para calcular os valores v_i^{tj} atribuídos às folhas, ou seja, as novas *features* do conjunto de dados transformado. O experimento, considerando um amplo espectro de *datasets*, permite a coleta de uma grande quantidade de informações. A seguir, listamos os *tree ensemble methods* utilizados para construir os conjuntos de árvores, bem como os métodos de *machine learning* para obter preditores em $\mathcal{V}$, o conjunto de dados transformado.

5.1
Valores das folha em algoritmos de classificação

Duas funções de valor, aplicado às folhas, foram experimentadas: *Binary leaves* (**BL**), ou folhas binárias e *Estimation leaves* (**EL**), ou estimativas nas folhas. Essas funções são aplicáveis às duas possíveis características de Y, conjunto de classes binárias ou multiclasse, se diferenciando apenas em EL na maneira como a previsão na segunda fase do método proposto é executada. O valor em EL requer que a predição seja feita em cada classe, considerando o modelo um-contra-todos. A classe de maior valor, *highest estimation score*, será a estimativa considerada. A maior pontuação é considerada aqui como a maior distância entre a instância e as funções de estimativas individuais no um-contra-todos. As duas funções são assim definidas:

Binary leaves - BL Dada uma folha j de uma árvore de decisão t, uma instância i terá *feature* v_i^{tj} igual a um quando x_i satisfaz a condição do caminho que alcança j na árvore t, e zero no caso contrário.

Estimation leaves - EL Seja p^{tj} o percentual de instâncias de S que ocorrem na folha j da árvore de decisão t que pertencem à classe 1, dado $Y = \{0, 1\}$. Então, a *feature* v_i^{tj} é igual a $-1 + 2.p^{tj}$ quando x_i satisfaz a condição do caminho que alcança j na árvore t, e zero no caso contrário. Essa abordagem

funciona para problemas de classificação binária, por isso, a abordagem um-contra-todos citada anteriormente.

5.2 Valores das folha em algoritmos de regressão

Nas tarefas de regressão foram testadas três funções. A primeira, igual definido em 5.1, **BL**. As outras duas são *Leaf values* (**LV**), ou valores das folhas, e *Path value* (bf PV), ou valor do caminho.

Leaf value - LV Essa função atribui à *feature* v_i^{tj}, quando x_i satisfaz a condição do caminho que alcança j na árvore t, o valor da média de valores y_i que ocorrem na folha j, e zero no caso contrário. Esse valor de média corresponde à predição da árvore de regressão.

Path value - PV Essa função atribui à feature v_i^{tj}, um valor ρ^{tj}, quando x_i satisfaz a condição do caminho que alcança j na árvore t, e zero no caso contrário. O valor de ρ^{tj} é dado pela média das médias dos valores y_i, das instâncias em S, de todos os nós da árvore de decisão que ocorrem no caminho que leva à folha j.

5.3 Métodos de conjunto de árvores

Como forma de avaliar o método $2PL$ e a sua capacidade de aprimorar métodos de alto desempenho, optamos por realizar experimentos usando os algoritmos de *Random Forest*, do pacote *scikit-learn* [6], e *Gradient Tree Boosting*, do pacote *XGBoost* [7]. Ao aplicar esses dois algoritmos ao conjunto de treinamento S, o método $2PL$ realiza a extração das árvores geradas e, transformando S, constroí o *dataset* $\mathcal{V}$ a partir de uma função sobre os valores. As funções acima definidas foram experimentadas em **RF** e **XG**, exceto a função **PV** que é testada apenas para RF em conjuntos de dados de regressão.

5.4 Métodos de previsão

Foram definidos três métodos de previsão para avaliar se o *overfitting* nos dados transformados $\mathcal{V}$ pode melhorar o desempenho dos métodos de conjunto de árvores, ao prever um conjunto de teste T, dado pelos exemplos fora do espaço de treinamento. Dois *kernels* do algoritmo de *support vector machine* foram testados, o modelo de função linear, aqui denominado (**SVM-L**) e o modelo não-linear RBF, aqui abreviado para (**SVM-R**), ambos do pacote

scikit-learn[6]. O outro teste utiliza o algoritmo de rede neural *perceptron*, denominado (**NN**), também do pacote scikit-learn.

Quando aplicados a conjuntos de dados de classificação, os preditores são usados de duas maneiras, mantendo os valores de y_i exatamente como no conjunto de dados original D e depois, adaptando y_i, no modelo um-contra-todos, para 1 se a classe associada for aquela em que a pontuação precisa ser estimada e -1 no caso contrário.

A utilização de diferentes preditores funciona também para estabelecer a correlação entre modelos mais ajustados, com maior precisão no conjunto de treino, e uma redução dos erros de estimativa nos $out-of-samples$, dado por T. Todos os algoritmos deste trabalho estão resumidos nas tabelas 5.1 e 5.2, para classificação e regressão, respectivamente. A coluna ID identifica o algoritmo, $\mathcal{A}$ especifica o método de conjunto de árvores usado, enquanto $g(i, j, t, S)$ indica as funções de valor aplicadas às folhas no conjunto de dados transformado. A quarta coluna existe apenas para os conjuntos de dados de classificação e indica se o preditor é usado diretamente (**D**) ou no modo multiclasse um-para-todos (**M**). Observe que os dois primeiros algoritmos para classificação e regressão são *Random Forest* (RF) e *XGBoost* (XG), eles servem como referência para comparação de desempenho das composições no $2PL$.

ID	$\mathcal{A}$	$g(i, j, t, S)$	$\mathcal{F}$	D/M
1	RF	-	-	-
2	XG	-	-	-
3	RF	BL	SVM-L	M
4	RF	BL	SVM-L	D
5	RF	EL	SVM-L	M
6	XG	BL	SVM-L	M
7	XG	BL	SVM-L	D
8	XG	BL	NN	D
9	XG	EL	SVM-R	M
10	RF	BL	SVM-R	M
11	RF	EL	SVM-R	M
12	XG	BL	NN	M
13	XG	BL	SVM-R	M
14	RF	BL	SVM-R	D
15	XG	BL	SVM-R	D
16	RF	EL	NN	M
17	RF	BL	NN	M
18	XG	EL	SVM-L	M
19	RF	BL	NN	D
20	XG	EL	NN	M

Tabela 5.1: Algoritmos de classificação

ID	$\mathcal{A}$	$g(i, j, t, S)$	$\mathcal{F}$
1	RF	-	-
2	XG	-	-
3	RF	PV	SVM-L
4	RF	LV	SVM-L
5	XG	BL	NN
6	RF	BL	NN
7	XG	BL	SVM-L
8	RF	PV	NN
9	RF	LV	NN
10	XG	LV	NN
11	XG	LV	SVM-L
12	RF	BL	SVM-L
13	XG	BL	SVM-R
14	XG	LV	SVM-R
15	RF	LV	SVM-R
16	RF	PV	SVM-R
17	RF	BL	SVM-R

Tabela 5.2: Algoritmos de regressão

6
Resultados

A performance de qualquer função estimadora depende de particularidades do problema para o qual ela é construída[12]. O tamanho do conjunto de dados $\mathcal{N}$, a real função dos dados $\mathcal{F}$ e o espaço dos valores esperados, dado pelo número de classes, nos problemas de classificação, com a sua distribuição no conjunto de dados e pelo intervalo dos valores reais nos problemas de regressão, impossibilitam a existência de um modelo único que funcione melhor em todos os problemas. Busca-se provar, a capacidade do modelo $2PL$ em generalização na média dos experimentos com dados artificais por funções e também em *datasets* de problemas reais. Considerando que o tamanho $\mathcal{N}$ das amostras é previamente conhecido e o espaço de valores é possível inferir pelo conjunto Y, o problema de generalização investigado fica em torno de $\mathcal{F}$.

As seções a seguir apresentam os resultados e análises dos algoritmos propostos, utilizando a metodologia $2PL$, em dados artificais, obtidos por funções geradoras de distribuições, disponíveis no pacote scikit-learn[6], e também em *datasets* clássicos para as tarefas de aprendizado supervisionado. Todos os conjuntos artificais gerados possuem 1000 exemplos, ou seja, $\mathcal{N}$ = 1000 nas tarefas de classificação e regressão.

6.1
Experimentos em *datasets* artificais

6.1.1
Conjunto de dados para classificação

As proposições a seguir concentram-se em torno de funções $\mathcal{F}$ capazes de gerar dados aleatórios, por funções conhecidas, lineares ou não. Todos os exemplos gerados para o problema de classificação possuem duas *features*, permitindo a sua representação no espaço de duas dimensões e são formados por classes binárias. A imagem a seguir apresenta uma matriz de nove projeções, as linhas representam cada experimento e as colunas são dadas pela distribuição dos dados, a acurácia no conjunto de treino e a acurácia no conjunto de teste. Em todos esses experimentos os conjuntos são divididos em proporções de 70% e 30%, $in-sample$ e $out-of-sample$ respectivamente. O primeiro conjunto

de dados representa uma distribuição linearmente separável, com outliers, seguido por duas abordagens não lineares que desafiam o modelo em amostragens cujos dados são gerados por agrupamento de *clusters*. Essa funções geradoras de amostras aleatórios seguem a abordagem da bilioteca scikit-learn[6].

Foram consideradas as configurações com 50, 100, 500 e 1000 árvores, seguindo os moldes dos resultados em *datasets* reais, que serão apresentados na próxima seção. Os algoritmos foram comparados aos modelos de *Random Forest* e *XGBoost* e apenas aquele com melhor desempenho é apresentado nos gráficos.

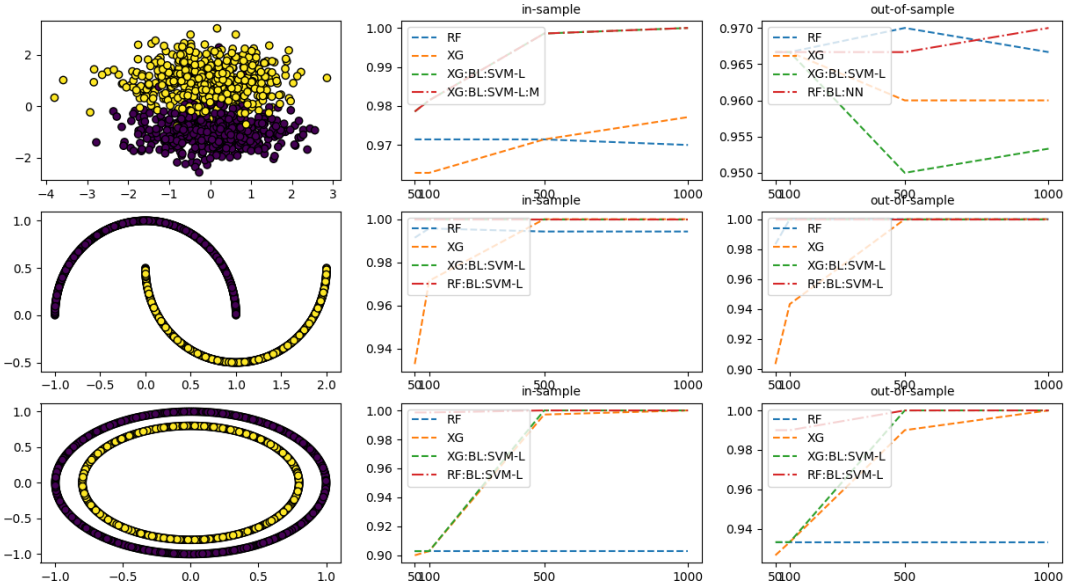

Figura 6.1: Experimentos a partir de dados artificais por funções lineares e não lineares

O algoritmo $XG : BL : SVM - L$ é aquele que apresentou os melhores resultados para conjuntos de mil árvores nos *datasets* utilizados como referência, por isso, o mesmo algoritmo segue como um dos modelos avaliados nessa seção. Detalhes desse desempenho são demonstrados na seção de resultados.

Os gráficos demonstram aspectos importantes do desempenho do algoritmo $XG : BL : SVM - L$, destacando-se a elevada acurácia do modelo no conjunto de treino, a consolidação dessa performance no teste e ainda o número inferior de árvores necessárias, em relação aos métodos de referência, para o modelo obter as melhores estimativas. Excluíndo-se os algoritmos RF, XG e $XG : BL : SVM - L$, a título de informação, o melhor desempenho dentre os outros é adicionado nos gráficos, representado pelas linhas na cor vermelha, colaborando na análise do $2PL$ e suas variações.

6.1.2
Conjunto de dados para regressão

A mesma lógica seguiu para as análises em funções geradoras de *datasets* de regressão. Dessa vez, a coluna que representa a distribuição dos dados foi omitida, dado que o conjunto Y está contido em uma faixa de valores reais. Sendo assim, uma matriz de 3 linhas e 2 colunas é construída, cada linha representa um experimento a partir de uma função geradora e as colunas representam os erros quadráticos médios em treino e teste, respectivamente. A primeira função propõe a criação de um *dataset* como uma combinação linear entre as *features*, duas nos experimentos, e o conjunto Y, essa estrutura é compatível com algoritmos de regressão linear. Em seguida, uma nova função linear é definida para os valores de Y e os dados de entrada X pertencem a uma distribuição normal X $N(0, 1)$. Demais features no conjunto de dados não devem ter influência nos valores de Y.

$$\mathcal{F} = x_1 + 2 * x_2 - 2 * x_3 - 1.5 * x_4$$

E por fim, uma função não linear, com ruídos, é definida em um conjunto de *features* uniformemente distribuídas no intervalo $[0, 1]$. Essa função é definida por Friedman,1991 [11] e aparece em Brieman, 1996 [4].

$$\mathcal{F} = 10{*}\sin(\pi{*}x_1{*}x_2){+}20{*}(x_3{-}0.5)^2{+}10{*}x_4{+}5{*}x_5{+}noise{*}N(0,1)$$

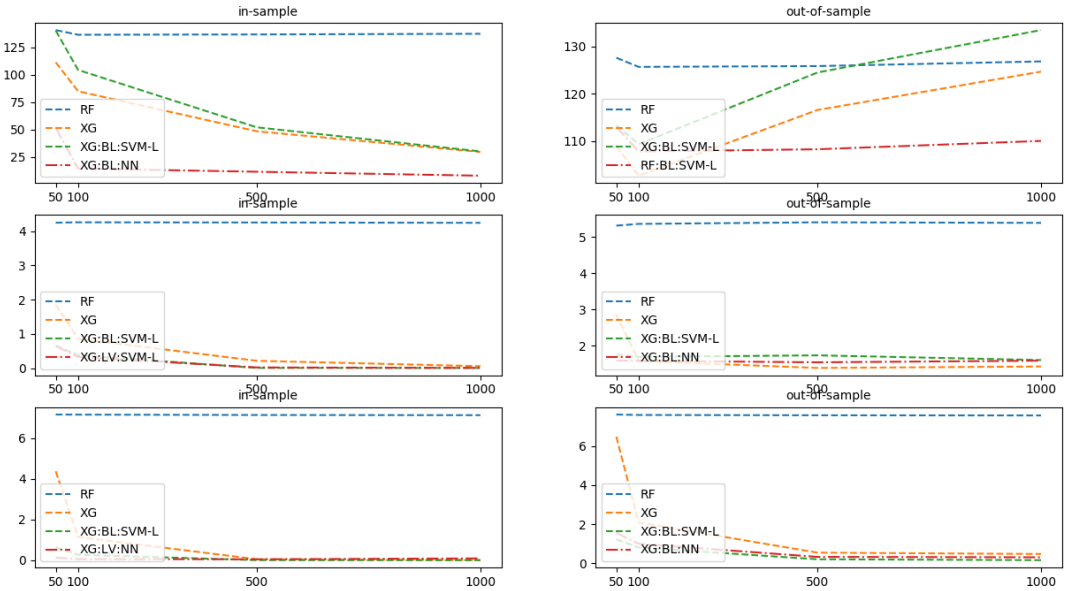

Figura 6.2: Experimentos a partir de dados artificais em regressão

Similar aos experimentos de classificação, os resultados demonstram uma consistência na diminuição do erro quadrático médio nos $in - sample$ e $out - of - sample$ *datasets*. Da mesma forma, fica destacado a capacidade

das melhores estimativas serem alcançadas com número de árvores inferior àquelas necessárias nos algoritmos de referência deste trabalho. O algoritmo de comparação foi o $XG : BL : SVM - L$, que demonstrou os melhores resultados nos *datasets* reais.

Mais uma vez, assim como na tarefa de classificação, um quarto algoritmo é adicionado aos gráficos, representado pela linhas na cor vermelha, indicando aquele com melhor desempenho, excluíndo-se os algoritmos RF, XG e $XG : BL : SVM - L$. Essa abordagem colabora na análise do $2PL$ e suas variações.

6.2 Resultados em problemas reais

Esta seção apresenta alguns dos resultados obtidos a partir de experimentos com o modelo proposto $2PL$, em *datasets* de problemas reais. Destaca-se o bom resultado, obtido pela média, de algumas configurações, ressaltando que ajustes de parâmetros nestes modelos podem influenciar alguns resultados individuais.

6.2.1 *Datasets*

Todos os experimentos a seguir, incluíndo os resultados do material complementar, foram baseados em 53 *datasets* de tarefas de classificação e 96 bases para tarefas de regressão. A tabela 6.1 apresenta detalhes dos conjuntos de dados de classificação. A primeira coluna é um identificador, seguida pelo nome do conjunto de dados, o número de *features* (NF) e o número de classes (NC). A próxima coluna, n_s, é o número de instâncias do conjunto de treino ($in - sample$) e a última coluna, com o rótulo n_o, é o número de instâncias de teste ($out - of - sample$). A tabela 6.2 apresenta os *datasets* de regressão e possui colunas análogas, exceto pelo número de classes, NC que não é aplicável.

Estes *datasets* foram obtidos a partir dos trabalhos de Bertsimas, 2017 [1] para classificação e um estudo de *benchmark* por Orzechowski, 2018 [18] dos algoritmos de regressão.

#	Dataset	NF	NC	n_s	n_o
1	Acute-inflammations-1	6	2	90	30
2	Acute-inflammations-2	6	2	90	30
3	Balance-scale	4	3	468	157
4	Blood-transfusion-service-center	4	2	561	187
5	Banknote-authentication	4	2	1029	343
6	Breast-cancer-wisconsin-diagnostic	30	2	426	143
7	Breast-cancer-wisconsin-prognostic	32	2	148	50
8	Breast-cancer	9	2	214	72
9	Car-evaluation	6	4	1296	432
10	Chess-king-rook-vs-king-pawn	36	2	2397	799
11	Climate-model-simulation-crashes	18	2	405	135
12	Congressional-voting-records	16	2	326	109
13	Connectionist-bench-sonar	60	2	156	52
14	Credit-approval	15	2	517	173
15	Cylinder-bands	39	2	405	135
16	Dermatology	34	6	274	92
17	Connectionist-bench	10	11	742	248
18	Contraceptive-method-choice	9	3	1104	369
19	Echocardiogram	7	3	99	33
20	Fertility	9	2	75	25
21	Haberman-survival	3	2	229	77
22	Hayes-roth	4	3	99	33
23	Heart-disease-Cleveland	13	5	227	76
24	Image-segmentation	19	7	157	53
25	Indian-liver-patient	10	2	437	146
26	Hepatitis	19	2	116	39
27	Ionosphere	34	2	263	88
28	Iris	4	3	112	38
29	Mammographic-mass	5	2	720	241
30	Monks-problems-1	6	2	93	31
31	Monks-problems-2	6	2	126	43
32	Monks-problems-3	6	2	91	31
33	Optical-recognition-handwritten-digits	64	10	2867	956
34	Ozone-level-detection-eight	72	2	1900	634
35	Ozone-level-detection-one	72	2	1902	634
36	Parkinsons	21	2	146	49
37	Planning-relax	12	2	136	46
38	Qsar-biodegradation	41	2	791	264
39	Seeds	7	3	157	53
40	Seismic-bumps	18	2	1938	646
41	Soybean-small	35	4	35	12
42	Spect-heart	22	2	60	20
43	Spectf-heart	44	2	60	20
44	Statlog-project-German-credit	20	2	750	250
45	Statlog-project-landsat-satellite	36	6	3326	1109
46	Teaching-assistant-evaluation	5	3	113	38
47	Thoracic-surgery	16	2	352	118
48	Thyroid-disease-ann-thyroid	21	3	2829	943
49	Thyroid-disease-new-thyroid	5	3	161	54
50	Spambase	57	2	3450	1151
51	Tic-tac-toe-endgame	9	2	718	240
52	Wall-following-robot-navigation-2	2	4	4092	1364
53	Wine	13	3	133	45

Tabela 6.1: *Datasets* de classificação

#	Dataset	NF	n_s	n_o
1	1027-ESL	4	366	122
2	1028-SWD	10	750	250
3	1029-LEV	4	750	250
4	1030-ERA	4	750	250
5	1089-USCrime	13	35	12
6	1096-FacultySalaries	4	37	13
7	192-vineyard	2	39	13
8	195-auto-price	15	119	40
9	207-autoPrice	15	119	40
10	210-cloud	5	81	27
11	228-elusage	2	41	14
12	229-pwLinear	10	150	50
13	230-machine-cpu	6	156	53
14	485-analcatdata-vehicle	4	36	12
15	505-tecator	124	180	60
16	519-vinnie	2	285	95
17	522-pm10	7	375	125
18	523-analcatdata-neavote	2	75	25
19	527-analcatdata-election2000	14	50	17
20	542-pollution	15	45	15
21	547-no2	7	375	125
22	556-analcatdata-apnea2	3	356	119
23	557-analcatdata-apnea1	3	356	119
24	560-bodyfat	14	189	63
25	561-cpu	7	156	53
26	579-fri-c0-250-5	5	187	63
27	581-fri-c3-500-25	25	375	125
28	582-fri-c1-500-25	25	375	125
29	583-fri-c1-1000-50	50	750	250
30	584-fri-c4-500-25	25	375	125
31	586-fri-c3-1000-25	25	750	250
32	589-fri-c2-1000-25	25	750	250
33	590-fri-c0-1000-50	50	750	250
34	591-fri-c1-100-10	10	75	25
35	592-fri-c4-1000-25	25	750	250
36	593-fri-c1-1000-10	10	750	250
37	594-fri-c2-100-5	5	75	25
38	595-fri-c0-1000-10	10	750	250
39	596-fri-c2-250-5	5	187	63
40	597-fri-c2-500-5	5	375	125
41	598-fri-c0-1000-25	25	750	250
42	599-fri-c2-1000-5	5	750	250
43	601-fri-c1-250-5	5	187	63
44	602-fri-c3-250-10	10	187	63
45	603-fri-c0-250-50	50	187	63
46	604-fri-c4-500-10	10	375	125
47	605-fri-c2-250-25	25	187	63
48	606-fri-c2-1000-10	10	750	250
49	607-fri-c4-1000-50	50	750	250
50	608-fri-c3-1000-10	10	750	250
51	609-fri-c0-1000-5	5	750	250
52	611-fri-c3-100-5	5	75	25
53	612-fri-c1-1000-5	5	750	250
54	613-fri-c3-250-5	5	187	63
55	615-fri-c4-250-10	10	187	63
56	616-fri-c4-500-50	50	375	125
57	617-fri-c3-500-5	5	375	125
58	618-fri-c3-1000-50	50	750	250
59	620-fri-c1-1000-25	25	750	250
60	621-fri-c0-100-10	10	75	25
61	622-fri-c2-1000-50	50	750	250
62	623-fri-c4-1000-10	10	750	250
63	624-fri-c0-100-5	5	75	25
64	626-fri-c2-500-50	50	375	125
65	627-fri-c2-500-10	10	375	125
66	628-fri-c3-1000-5	5	750	250
67	631-fri-c1-500-5	5	375	125
68	633-fri-c0-500-25	25	375	125
69	634-fri-c2-100-10	10	75	25
70	635-fri-c0-250-10	10	187	63
71	637-fri-c1-500-50	50	375	125
72	641-fri-c1-500-10	10	375	125
73	643-fri-c2-500-25	25	375	125
74	644-fri-c4-250-25	25	187	63
75	645-fri-c3-500-50	50	375	125
76	646-fri-c3-500-10	10	375	125
77	647-fri-c1-250-10	10	187	63
78	648-fri-c1-250-50	50	187	63
79	649-fri-c0-500-5	5	375	125
80	650-fri-c0-500-50	50	375	125
81	651-fri-c0-100-25	25	75	25
82	653-fri-c0-250-25	25	187	63
83	654-fri-c0-500-10	10	375	125
84	656-fri-c1-100-5	5	75	25
85	657-fri-c2-250-10	10	187	63
86	658-fri-c3-250-25	25	187	63
87	659-sleuth-ex1714	7	35	12
88	663-rabe-266	2	90	30
89	665-sleuth-case2002	6	110	37
90	666-rmftsa-ladata	10	381	127
91	678-visualizing-environmental	3	83	28
92	687-sleuth-ex1605	5	46	16
93	690-visualizing-galaxy	4	242	81
94	695-chatfield-4	12	176	59
95	706-sleuth-case1202	6	69	24
96	712-chscase-geyser1	2	166	56

Tabela 6.2: *Datasets* de regressão

6.2.2 Experimentos nos *datasets* clássicos

Os resultados obtidos seguiram uma linha de análise, assim como nos experimentos por funções de randomização dos dados, os números de estimadores, leia-se árvores, construídas na primeira fase de treinamento foram 50, 100, 500 e 1000. Os valores demonstrados nessa seção referem-se ao desempenho do $2PL$ em conjuntos de 1000 árvores, os demais estão contemplados no apêndice. A utilização desse número de árvores segue pela constatação dos melhores resultados, na média, com essa configuração em Orzechowski, 2018 [18].

A tabela 6.3 apresenta o número de vezes em que os algoritmos obtém o melhor desempenho no treino para os dados de classificação. Em sequência, a tabela 6.4 representa o número de vitórias dos algoritmos no conjunto de teste para classificação. Os casos de regressão seguem pelo mesmo caminho, em 6.5 os melhores algoritmos no treinamento e 6.6 os campeões nos dados de teste.

Em seguida às tabelas de desempenho dos algoritmos quatro novas tabelas são apresentadas. As duas primeiras demonstram o desempenho dos algoritmos nos conjuntos de dados de classificação, em treino e teste, e na sequência os mesmos resultados para os *datasets* de regressão.

#	No. of winners
XG:BL:SVM-L	53
XG:BL:SVM-L:M	52
RF:BL:SVM-L	28
RF:BL:SVM-L:M	28
RF:EL:SVM-L:M	27
XG:EL:SVM-L:M	27
XG:BL:NN	18
XG:EL:NN:M	16
XG:BL:NN:M	15
XG	14
RF:BL:NN	11
XG:EL:SVM:M	10
RF:BL:NN:M	10
RF:EL:NN:M	8
RF	7
SVM-L	6
XG:BL:SVM:M	5
RF:BL:SVM	5
RF:EL:SVM:M	4
XG:BL:SVM	4
RF:BL:SVM:M	4

#	No. of winners
XG:BL:SVM-L	53
XG	14
RF	7
SVM-L	6

Tabela 6.3: À esquerda o resultado considerando todos os algoritmos propostos. À direita o melhor algoritmo comparado com os *benchmarks*, nos *datasets* de classificação - in sample

#	No. of winners
XG:BL:SVM-L	19
XG:BL:SVM-L:M	16
XG:BL:SVM	13
XG	13
RF:EL:NN:M	12
RF:BL:NN	12
RF	12
XG:BL:SVM:M	11
XG:EL:NN:M	11
RF:BL:SVM-L	11
RF:BL:SVM-L:M	11
XG:BL:NN	10
XG:BL:NN:M	10
RF:EL:SVM-L:M	9
SVM-L	9
RF:BL:NN:M	9
XG:EL:SVM:M	8
RF:BL:SVM	8
XG:EL:SVM-L:M	8
RF:BL:SVM:M	7
RF:EL:SVM:M	6

#	No. of winners
XG:BL:SVM-L	29
XG	21
RF	15
SVM-L	12

Tabela 6.4: À esquerda o resultado considerando todos os algoritmos propostos. À direita o melhor algoritmo comparado com os *benchmarks*, nos *datasets* de classificação - out of sample

#	No. of winners
XG:BL:SVM-L	47
XG:LV:SVM-L	31
RF:LV:SVM-L	4
XG:BL:NN	4
XG	3
XG:LV:NN	3
RF:BL:NN	2
RF:PV:NN	1
RF:BL:SVM-L	1
RF:PV:SVM-L	0
XG:BL:SVM	0
RF:LV:NN	0
RF:LV:SVM	0
RF:BL:SVM	0
RF	0
RF:PV:SVM	0
SVM-L	0
XG:LV:SVM	0

#	No. of winners
XG:BL:SVM-L	76
XG	18
RF	2
SVM-L	0

Tabela 6.5: À esquerda o resultado considerando todos os algoritmos propostos. À direita o melhor algoritmo comparado com os *benchmarks*, nos *datasets* de regressão - in sample

#	No. of winners
XG:BL:SVM-L	45
XG:BL:NN	16
XG	8
RF	7
SVM-L	6
RF:PV:SVM-L	4
RF:BL:NN	3
XG:LV:NN	2
RF:BL:SVM-L	2
XG:LV:SVM-L	1
RF:PV:NN	1
RF:LV:SVM-L	1
XG:BL:SVM	0
RF:LV:NN	0
RF:LV:SVM	0
RF:BL:SVM	0
RF:PV:SVM	0
XG:LV:SVM	0

#	No. of winners
XG:BL:SVM-L	62
XG	18
RF	10
SVM-L	6

Tabela 6.6: À esquerda o resultado considerando todos os algoritmos propostos. À direita o melhor algoritmo comparado com os *benchmarks*, nos *datasets* de regressão - out of sample

#	SVM-L	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	XG	XG:BL:SVM-L	XG:BL:SVM	.:M	RF:EL:SVM:M	RF:EL:NN:M	XG:BL:SVM-L:M	XG:BL:SVM:M	XG:BL:NN:M	XG:EL:SVM-L:M	XG:EL:SVM:M	XG:EL:NN:M
1	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
2	0.989	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
3	0.876	0.838	0.989	0.831	0.902	0.940	**1.000**	0.908	**1.000**	0.976	0.835	0.885	0.968	0.831	0.865	**1.000**	0.908	0.998	**1.000**	0.912	0.996
4	0.781	0.786	0.911	0.781	0.800	0.836	**0.938**	0.781	0.809	0.911	0.781	0.802	0.895	0.781	0.802	**0.938**	0.781	0.811	0.895	0.832	0.843
5	0.828	0.913	0.998	0.952	0.942	0.997	**1.000**	0.995	0.996	0.998	0.952	0.955	0.997	0.949	0.953	**1.000**	0.995	0.994	0.999	0.998	0.998
6	**1.000**	0.972	**1.000**	0.962	0.953	0.995	**1.000**	0.984	0.991	**1.000**	0.962	0.948	**1.000**	0.962	0.948	**1.000**	0.984	0.993	**1.000**	0.995	0.995
7	0.845	0.865	**1.000**	0.770	0.980	0.993	**1.000**	0.770	**1.000**	**1.000**	0.770	0.980	**1.000**	0.770	0.986	**1.000**	0.770	**1.000**	**1.000**	**1.000**	**1.000**
8	0.780	0.841	0.949	0.762	0.869	0.897	**0.986**	0.762	0.879	0.949	0.762	0.860	0.939	0.762	0.813	**0.986**	0.762	0.883	0.949	0.766	0.883
9	0.705	0.829	0.922	0.840	0.835	0.969	**1.000**	0.931	0.981	0.909	0.840	0.840	0.897	0.840	0.840	**1.000**	0.926	0.959	0.996	0.965	0.964
10	0.938	0.941	0.940	0.941	0.940	0.976	**0.998**	0.970	0.966	0.940	0.941	0.941	0.941	0.941	0.941	**0.998**	0.970	0.972	0.987	0.980	0.977
11	0.975	0.963	**1.000**	0.911	0.978	0.998	**1.000**	0.911	0.998	**1.000**	0.911	0.983	**1.000**	0.911	0.983	**1.000**	0.911	0.998	**1.000**	0.998	0.998
12	0.945	0.975	**0.997**	0.975	0.975	0.979	**0.997**	0.975	0.975	**0.997**	0.975	0.975	**0.997**	0.975	0.975	**0.997**	0.975	0.975	0.988	0.979	0.979
13	**1.000**	0.962	**1.000**	0.904	**1.000**	**1.000**	**1.000**	0.782	**1.000**	**1.000**	0.904	**1.000**	**1.000**	0.904	**1.000**	**1.000**	0.782	**1.000**	**1.000**	**1.000**	**1.000**
14	0.859	0.913	0.975	0.880	0.901	0.956	**1.000**	0.892	0.928	0.975	0.880	0.897	0.971	0.872	0.878	**1.000**	0.892	0.957	0.985	0.956	0.954
15	0.785	0.835	**1.000**	0.874	0.914	0.985	**1.000**	0.694	0.993	**1.000**	0.874	0.884	0.998	0.852	0.919	**1.000**	0.694	0.988	0.995	0.990	0.990
16	0.985	0.964	0.993	0.974	0.974	**1.000**	**1.000**	0.923	**1.000**	0.993	0.974	0.974	0.993	0.971	0.974	**1.000**	0.989	0.993	**1.000**	0.996	**1.000**
17	0.811	0.547	0.903	0.586	0.658	**1.000**	**1.000**	0.472	0.980	0.898	0.682	0.547	0.885	0.660	0.592	**1.000**	0.993	0.892	**1.000**	0.993	0.961
18	0.554	0.574	0.667	0.578	0.579	0.650	**0.958**	0.584	0.606	0.667	0.598	0.574	0.637	0.598	0.588	0.949	0.604	0.597	0.890	0.636	0.605
19	0.545	0.758	**1.000**	0.424	0.990	0.990	**1.000**	0.424	**1.000**	**1.000**	0.869	0.980	**1.000**	0.869	0.939	**1.000**	0.970	**1.000**	**1.000**	0.939	**1.000**
20	0.867	0.907	**0.987**	0.867	**0.987**	0.907	**0.987**	0.867	**0.987**	**0.987**	0.867	**0.987**	**0.987**	0.867	0.973	**0.987**	0.867	0.907	0.973	0.867	0.907
21	0.782	0.812	0.961	0.782	0.873	0.869	**0.978**	0.782	0.891	0.961	0.782	0.869	0.956	0.782	0.913	**0.978**	0.782	0.873	0.930	0.782	0.869
22	0.626	0.818	**0.909**	0.808	0.899	**0.909**	**0.909**	0.384	**0.909**	**0.909**	0.818	**0.909**	**0.909**	0.818	0.899	**0.909**	0.879	**0.909**	**0.909**	**0.909**	**0.909**
23	0.670	0.696	**1.000**	0.546	0.960	0.982	**1.000**	0.546	0.996	**1.000**	0.700	0.731	**1.000**	0.714	0.824	**1.000**	0.581	0.952	**1.000**	0.714	0.947
24	0.987	0.860	**1.000**	0.834	0.975	**1.000**	**1.000**	0.554	**1.000**	**1.000**	0.898	0.917	**1.000**	0.898	0.898	**1.000**	0.987	**1.000**	**1.000**	0.975	**1.000**
25	0.723	0.785	0.995	0.723	0.769	0.908	**1.000**	0.723	0.952	0.995	0.723	0.762	0.993	0.723	0.778	**1.000**	0.723	0.934	0.979	0.934	0.929
26	0.871	0.940	**1.000**	0.810	0.940	0.974	**1.000**	0.810	0.966	**1.000**	0.810	0.940	**1.000**	0.810	0.940	**1.000**	0.810	0.966	**1.000**	0.810	0.974
27	0.894	0.939	0.996	0.913	0.939	0.996	**1.000**	0.935	**1.000**	0.996	0.913	0.924	0.996	0.916	0.924	**1.000**	0.935	**1.000**	**1.000**	0.996	0.996
28	0.991	**1.000**	**1.000**	0.982	**1.000**	0.991	**1.000**	0.964	0.991	**1.000**	0.982	0.982	**1.000**	0.973	0.982	**1.000**	0.964	0.991	**1.000**	0.991	0.991
29	0.828	0.850	0.894	0.846	0.846	0.871	**0.942**	0.853	0.846	0.894	0.846	0.846	0.890	0.832	0.828	**0.942**	0.853	0.847	0.892	0.872	0.872
30	0.742	0.935	**1.000**	0.839	**1.000**	0.978	**1.000**	0.688	**1.000**	**1.000**	0.839	**1.000**	**1.000**	0.882	**1.000**	**1.000**	0.688	**1.000**	0.989	0.957	0.978
31	0.651	0.722	**1.000**	0.651	0.960	0.897	**1.000**	0.651	0.944	**1.000**	0.651	0.952	**1.000**	0.651	0.889	**1.000**	0.651	0.968	0.968	0.825	0.913
32	0.835	0.934	**1.000**	0.934	0.967	0.945	**1.000**	0.868	0.934	**1.000**	0.934	0.956	**1.000**	0.934	0.967	**1.000**	0.868	0.934	0.978	0.945	0.945
33	0.940	0.584	0.911	0.719	0.810	0.993	**1.000**	0.955	0.966	0.903	0.783	0.692	0.900	0.774	0.663	**1.000**	0.967	0.949	1.000	0.970	0.971
34	0.947	0.942	0.995	0.934	0.960	0.986	**1.000**	0.934	0.971	0.995	0.934	0.951	0.993	0.934	0.934	**1.000**	0.934	0.967	0.999	0.991	0.987
35	0.976	0.974	0.997	0.972	0.972	0.994	**1.000**	0.972	0.989	0.997	0.972	0.972	0.997	0.972	0.972	**1.000**	0.972	0.975	**1.000**	0.997	0.992
36	0.959	0.966	**1.000**	0.884	**1.000**	**1.000**	**1.000**	0.747	**1.000**	**1.000**	0.884	**1.000**	**1.000**	0.911	**1.000**	**1.000**	0.747	**1.000**	**1.000**	**1.000**	**1.000**
37	0.757	0.824	**1.000**	0.757	0.993	0.978	**1.000**	0.757	0.993	**1.000**	0.757	0.993	**1.000**	0.757	0.993	**1.000**	0.757	0.993	**1.000**	0.978	**1.000**
38	0.877	0.861	0.986	0.876	0.872	0.956	**1.000**	0.895	0.905	0.986	0.876	0.869	0.976	0.875	0.858	**1.000**	0.895	0.906	0.989	0.952	0.956
39	0.981	0.981	**1.000**	0.981	0.981	**1.000**	**1.000**	0.968	0.987	**1.000**	0.981	0.981	**1.000**	0.981	0.981	**1.000**	0.962	**1.000**	**1.000**	0.975	**1.000**
40	0.929	0.929	0.972	0.929	0.929	0.941	**1.000**	0.929	0.929	0.972	0.929	0.929	0.970	0.929	0.929	**1.000**	0.929	0.929	0.998	0.953	0.955
41	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	0.600	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
42	0.800	0.883	**0.967**	0.817	0.950	0.867	**0.967**	0.717	0.950	**0.967**	0.817	0.950	**0.967**	0.817	0.883	**0.967**	0.717	0.900	0.900	0.850	0.867
43	**1.000**	**1.000**	**1.000**	0.917	**1.000**	**1.000**	**1.000**	0.533	**1.000**	**1.000**	0.917	**1.000**	**1.000**	0.950	**1.000**	**1.000**	0.533	**1.000**	**1.000**	**1.000**	**1.000**
44	0.769	0.791	0.965	0.776	0.783	0.892	**1.000**	0.699	0.916	0.965	0.776	0.804	0.940	0.763	0.791	**1.000**	0.699	0.947	0.975	0.909	0.907
45	0.824	0.789	0.883	0.811	0.833	0.961	**1.000**	0.893	0.939	0.881	0.820	0.796	0.880	0.820	0.793	**1.000**	0.904	0.894	1.000	0.903	0.916
46	0.522	0.708	**0.973**	0.372	0.965	0.920	**0.973**	0.372	0.912	**0.973**	0.788	0.841	**0.973**	0.805	0.823	**0.973**	0.885	0.947	**0.973**	0.841	0.947
47	0.861	0.866	0.991	0.861	0.875	0.898	**1.000**	0.861	0.909	0.991	0.861	0.878	0.983	0.861	0.861	**1.000**	0.861	0.912	0.997	0.898	0.932
48	0.946	0.996	0.999	0.998	0.994	0.999	**1.000**	0.998	0.994	0.999	0.998	0.994	0.999	0.998	0.995	**1.000**	0.998	0.991	**1.000**	1.000	0.998
49	0.925	0.950	**1.000**	0.826	0.950	**1.000**	**1.000**	0.795	**1.000**	**1.000**	0.932	0.963	**1.000**	0.932	0.944	**1.000**	0.801	0.994	**1.000**	0.857	**1.000**
50	0.946	0.912	0.944	0.912	0.894	0.962	**0.997**	0.958	0.948	0.944	0.912	0.908	0.941	0.914	0.898	**0.997**	0.958	0.952	0.988	0.962	0.962
51	0.657	0.825	0.907	0.834	0.834	0.960	**1.000**	0.816	0.947	0.907	0.834	0.834	0.907	0.834	0.852	**1.000**	0.816	0.957	0.997	0.979	0.978
52	1.000	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
53	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	0.992	0.992	**1.000**	0.992	0.992	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**

Tabela 6.7: Resultados em *datasets* de classificação - in sample

#	SVM-L	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	XG	XG:BL:SVM-L	XG:BL:SVM	:M	RF:EL:SVM:M	RF:EL:NN:M	XG:BL:SVM-L:M	XG:BL:SVM:M	XG:BL:NN:M	XG:EL:SVM-L:M	XG:EL:SVM:M	XG:EL:NN:M
1	**1.000**	**1.000**	**1.000**	0.967	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	0.967	**1.000**	**1.000**	0.967	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
2	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
3	0.885	0.834	0.898	0.841	0.879	0.930	0.924	0.930	**0.955**	0.898	0.841	0.898	0.879	0.841	0.885	0.911	0.924	0.930	0.924	0.943	0.904
4	0.706	0.706	0.701	0.706	0.743	0.738	0.695	0.706	0.743	0.701	0.706	0.733	0.690	0.706	0.738	0.695	0.706	**0.749**	0.706	0.733	0.727
5	0.825	0.904	0.988	0.953	0.962	0.985	**0.994**	0.985	0.988	0.988	0.953	0.959	0.988	0.950	0.936	**0.994**	0.985	0.988	0.985	0.985	0.985
6	0.958	0.958	0.958	0.958	0.958	0.965	**0.979**	0.965	0.944	0.958	0.958	0.944	0.958	0.951	0.944	**0.979**	0.965	0.944	0.965	0.965	0.972
7	0.700	**0.760**	0.640	0.740	0.620	0.680	0.680	0.740	0.720	0.660	0.740	0.640	0.680	0.740	0.660	0.680	0.740	0.700	0.700	0.660	0.680
8	**0.792**	0.722	0.708	0.764	0.736	0.708	0.694	0.764	0.722	0.708	0.764	0.722	0.750	0.764	0.736	0.694	0.764	0.736	0.694	0.764	0.750
9	0.685	0.794	0.903	0.808	0.815	0.944	**0.991**	0.926	0.968	0.887	0.808	0.808	0.870	0.808	0.808	0.986	0.898	0.942	0.984	0.951	0.942
10	0.939	0.940	0.944	0.940	0.949	0.972	**0.991**	0.970	0.962	0.944	0.940	0.940	0.940	0.940	0.940	**0.991**	0.970	0.966	0.985	0.975	0.972
11	**0.970**	0.941	0.956	0.926	0.941	0.948	0.948	0.926	0.956	0.956	0.926	0.941	0.956	0.926	0.933	0.948	0.926	0.956	0.963	0.948	0.948
12	0.945	0.963	0.963	0.963	0.963	**0.972**	0.917	0.963	0.963	0.963	0.963	0.963	0.963	0.963	0.963	0.917	0.963	**0.972**	0.954	**0.972**	**0.972**
13	0.865	0.788	0.885	0.788	0.846	0.923	**0.942**	0.635	0.904	0.885	0.788	0.846	0.885	0.788	0.865	**0.942**	0.635	0.904	0.923	0.923	0.923
14	0.867	0.850	0.827	0.867	0.844	0.832	0.809	**0.873**	0.821	0.827	0.867	0.844	0.832	0.867	0.855	0.809	**0.873**	0.815	0.809	0.832	0.832
15	0.778	0.763	0.785	0.763	0.785	0.815	0.822	0.689	**0.844**	0.785	0.763	0.756	0.785	0.770	0.756	0.822	0.689	0.822	0.830	0.807	0.807
16	**0.978**	0.913	0.935	0.913	0.913	**0.978**	0.957	0.891	0.946	0.924	0.924	0.924	0.924	0.924	0.913	**0.978**	0.935	0.946	**0.978**	0.946	0.957
17	0.706	0.536	0.694	0.560	0.593	0.875	**0.964**	0.359	0.883	0.669	0.581	0.528	0.665	0.573	0.548	0.940	0.907	0.790	0.923	0.891	0.839
18	0.526	0.509	0.539	0.526	0.547	0.556	0.458	0.512	0.566	0.537	0.537	0.534	0.542	0.539	0.550	0.477	0.520	0.547	0.482	**0.572**	0.553
19	0.333	0.333	0.394	**0.485**	0.364	0.303	0.333	**0.485**	0.455	0.455	0.394	0.303	0.455	0.394	0.364	0.394	0.364	0.364	0.394	0.424	0.333
20	0.920	**0.960**	**0.960**	0.920	0.920	**0.960**	0.880	0.920	**0.960**	**0.960**	0.920	0.920	**0.960**	0.920	0.920	0.880	0.920	**0.960**	0.840	0.920	0.920
21	0.597	0.636	0.597	0.597	0.623	**0.649**	**0.649**	0.597	0.623	0.597	0.597	0.623	0.623	0.597	0.623	**0.649**	0.597	0.636	0.584	0.597	**0.649**
22	0.576	0.727	0.818	0.667	0.818	0.758	0.788	0.394	0.818	0.818	0.727	0.788	0.818	0.727	**0.848**	0.818	0.818	0.758	0.758	0.758	0.758
23	0.526	**0.618**	0.553	0.526	0.526	0.539	0.526	0.526	0.526	0.526	0.579	0.605	0.526	0.605	0.579	0.526	0.526	0.539	0.500	0.487	0.579
24	0.830	0.736	**0.943**	0.660	0.849	0.868	0.906	0.453	0.830	0.887	0.736	0.792	0.906	0.736	0.774	0.887	0.849	0.849	0.887	0.868	0.906
25	0.685	0.685	0.671	0.685	0.692	0.692	0.692	0.685	0.678	0.671	0.685	0.678	0.664	0.685	0.692	0.692	0.685	0.658	**0.699**	0.678	0.678
26	0.846	0.769	0.718	0.744	0.795	**0.872**	0.821	0.744	0.846	0.718	0.744	0.795	0.718	0.744	0.821	0.821	0.744	0.821	0.769	0.744	**0.872**
27	0.784	0.920	0.932	0.920	0.943	**0.977**	**0.977**	0.920	**0.977**	0.932	0.920	0.920	0.932	0.920	0.920	**0.977**	0.920	**0.977**	0.943	**0.977**	**0.977**
28	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	0.921	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	0.921	**0.974**	**0.974**	0.921	0.921	**0.974**
29	0.793	**0.830**	0.784	0.813	0.826	0.826	0.759	**0.830**	0.817	0.784	0.813	0.826	0.788	0.788	0.793	0.759	**0.830**	0.809	**0.830**	0.826	0.826
30	0.677	0.903	**1.000**	0.806	**1.000**	0.774	**1.000**	0.645	0.871	**1.000**	0.806	**1.000**	**1.000**	0.806	**1.000**	**1.000**	0.645	0.968	0.806	0.742	0.774
31	0.535	0.605	**0.814**	0.535	0.721	0.628	0.791	0.535	0.721	**0.814**	0.535	0.674	0.791	0.535	0.744	0.791	0.535	0.698	0.721	0.558	0.674
32	0.903	0.935	0.935	0.935	**0.968**	0.935	0.935	0.806	0.935	0.935	0.935	0.935	0.935	0.935	**0.968**	0.935	0.806	0.935	0.903	0.935	0.935
33	0.806	0.564	0.861	0.681	0.788	0.957	**0.974**	0.945	0.954	0.867	0.752	0.664	0.859	0.735	0.618	0.972	0.951	0.929	0.963	0.960	0.953
34	0.943	0.943	0.926	**0.945**	**0.945**	0.940	0.935	**0.945**	0.938	0.926	**0.945**	0.943	0.927	**0.945**	**0.945**	0.935	**0.945**	0.942	0.937	0.940	0.938
35	**0.970**	**0.970**	0.957	**0.970**	**0.970**	0.967	0.964	**0.970**	0.959	0.957	**0.970**	**0.970**	0.956	**0.970**	**0.970**	0.964	**0.970**	**0.970**	0.957	0.961	0.968
36	0.857	0.918	0.939	0.939	0.939	0.939	**0.959**	0.776	0.918	0.939	0.939	0.939	0.939	0.939	0.939	**0.959**	0.776	0.918	0.918	0.939	0.918
37	0.587	0.565	0.630	0.587	**0.696**	0.630	0.565	0.587	0.587	0.630	0.587	0.630	0.630	0.587	0.630	0.565	0.587	0.587	0.522	0.609	0.609
38	0.845	0.841	0.867	0.864	0.852	0.890	0.875	**0.898**	0.845	0.867	0.864	0.830	0.871	0.856	0.822	0.875	**0.898**	0.845	0.871	0.883	0.890
39	0.943	0.943	**0.962**	0.943	**0.962**	**0.962**	**0.962**	**0.962**	**0.962**	**0.962**	0.925	**0.962**	**0.962**	0.925	**0.962**	**0.962**	**0.962**	0.943	0.943	**0.962**	**0.962**
40	0.950	0.950	0.924	0.950	0.950	**0.952**	0.920	0.950	0.950	0.924	0.950	0.950	0.927	0.950	0.950	0.920	0.950	0.950	0.915	0.943	0.940
41	**1.000**	0.917	0.917	0.917	0.917	0.917	0.917	0.500	0.917	0.917	0.917	0.917	0.917	0.917	0.917	0.917	0.917	0.917	0.917	0.917	0.917
42	**0.850**	0.600	0.450	0.500	0.450	0.750	0.600	0.500	0.650	0.450	0.500	0.500	0.450	0.500	0.750	0.600	0.500	0.550	0.800	0.550	0.750
43	0.850	**0.950**	0.900	0.600	**0.950**	0.850	0.850	0.400	0.900	0.900	0.600	**0.950**	0.900	0.700	**0.950**	0.850	0.400	0.900	0.900	0.850	0.850
44	0.736	0.728	0.728	0.728	0.740	0.768	0.720	0.704	0.760	0.728	0.728	0.752	0.740	0.720	0.740	0.720	0.704	0.752	0.696	0.772	**0.776**
45	0.794	0.807	0.852	0.826	0.838	0.904	**0.913**	0.878	0.904	0.857	0.829	0.813	0.857	0.834	0.813	0.912	0.881	0.889	0.909	0.883	0.896
46	0.289	0.368	0.553	0.184	0.553	0.474	0.500	0.184	0.447	**0.579**	0.421	0.447	0.526	0.421	0.421	0.553	0.447	0.447	0.447	0.421	0.474
47	0.822	0.822	0.831	0.822	0.805	0.822	0.831	0.822	0.814	0.831	0.822	0.805	0.822	0.822	0.822	0.831	0.822	0.831	**0.847**	0.822	0.839
48	0.943	0.995	**0.997**	**0.997**	0.992	0.995	**0.997**	0.995	0.990	**0.997**	**0.997**	0.992	**0.997**	**0.997**	0.993	**0.997**	0.995	0.989	0.996	0.995	0.994
49	0.926	**0.981**	0.944	0.870	0.963	**0.981**	**0.981**	0.796	**0.981**	0.944	0.963	0.944	0.944	0.963	0.963	**0.981**	0.852	**0.981**	**0.981**	0.870	**0.981**
50	0.920	0.899	0.913	0.901	0.879	0.934	0.935	**0.936**	0.924	0.913	0.901	0.894	0.916	0.899	0.887	0.935	**0.936**	0.932	0.934	**0.936**	0.934
51	0.642	0.796	0.896	0.812	0.812	0.908	**1.000**	0.762	0.875	0.896	0.812	0.812	0.896	0.812	0.833	**1.000**	0.762	0.892	0.967	0.925	0.925
52	0.999	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
53	0.844	**0.978**	**0.978**	**0.978**	**0.978**	0.956	0.956	**0.978**	0.956	**0.978**	**0.978**	**0.978**	**0.978**	0.956	**0.978**	0.956	0.956	0.956	0.956	0.956	0.956

Tabela 6.8: Resultados em *datasets* de classificação - out of sample

#	SVM-L	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	RF:LV:SVM-L	RF:LV:SVM	RF:LV:NN	RF:PV:SVM-L	RF:PV:SVM	RF:PV:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	XG:LV:SVM-L	XG:LV:SVM	XG:LV:NN
1	0.290	0.436	0.209	0.466	0.311	0.199	0.459	0.205	0.198	0.463	0.189	0.186	0.149	0.524	0.266	**0.145**	0.391	0.163
2	0.395	0.385	0.341	0.378	0.372	0.344	0.378	0.361	0.343	0.379	0.359	0.314	0.336	0.349	**0.311**	0.336	0.335	0.329
3	0.402	0.484	0.397	0.438	0.452	0.401	0.439	0.403	0.401	0.435	0.424	**0.357**	0.392	0.403	0.364	0.392	0.377	0.367
4	3.911	2.602	2.403	2.554	2.331	2.399	2.494	2.261	2.398	2.538	2.246	2.209	2.381	2.640	2.225	2.388	2.320	**2.208**
5	262.495	72.838	**0.010**	1.75e+03	4.706	0.010	1.75e+03	38.648	0.010	1.75e+03	40.878	3.024	0.010	1.76e+03	0.028	0.010	1.76e+03	674.204
6	4.691	1.446	0.009	17.436	0.081	**0.009**	17.389	0.862	0.009	17.420	1.956	0.431	0.009	17.218	0.030	0.009	17.177	6.270
7	10.216	3.810	1.612	23.544	**0.833**	1.295	23.134	1.657	1.313	23.393	1.749	1.694	0.964	24.397	0.852	1.079	23.913	1.423
8	1.34e+07	2.66e+06	2.40e+07	3.92e+07	3.99e+06	1.11e+05	3.92e+07	7.92e+05	1.09e+05	3.92e+07	2.13e+05	3.57e+05	2.60e+07	3.92e+07	3.44e+07	**3.50e+04**	3.92e+07	1.10e+07
9	1.34e+07	2.66e+06	2.40e+07	3.92e+07	3.44e+07	1.11e+05	3.92e+07	6.06e+05	1.09e+05	3.92e+07	1.39e+06	3.59e+05	2.60e+07	3.92e+07	3.44e+07	**3.50e+04**	3.92e+07	3.57e+07
10	0.357	0.076	0.009	0.731	0.034	0.009	0.545	0.047	0.008	0.623	0.037	0.009	0.007	0.733	0.012	**0.006**	0.438	0.032
11	128.920	52.153	22.544	526.355	16.874	21.574	525.737	28.335	21.067	526.114	24.561	30.800	24.076	530.152	**16.814**	26.294	525.256	40.858
12	6.739	3.471	1.733	12.532	1.979	1.631	13.030	1.118	1.641	12.532	1.159	0.622	**0.010**	16.931	0.017	0.011	16.637	0.062
13	3.08e+04	6.06e+03	1.26e+04	3.39e+04	330.349	120.130	3.39e+04	1.01e+04	123.767	3.39e+04	6.00e+03	661.809	1.27e+04	3.41e+04	229.942	**88.935**	3.39e+04	1.42e+04
14	8.73e+04	8.63e+03	1.24e+04	8.76e+04	2.92e+03	239.584	8.75e+04	3.33e+03	247.307	8.75e+04	3.76e+03	1.52e+03	1.95e+04	8.77e+04	53.625	**0.010**	8.75e+04	2.91e+03
15	7.840	5.311	0.705	181.837	0.371	0.458	157.847	0.760	0.451	162.006	0.837	0.098	0.009	206.756	0.025	**0.009**	165.875	0.085
16	8.734	2.393	2.175	4.254	2.394	2.166	3.818	2.122	2.166	3.999	2.109	2.060	1.942	5.313	1.998	1.938	3.880	**1.833**
17	0.623	0.497	0.015	0.467	0.361	0.009	0.461	0.190	0.009	0.466	0.199	0.150	0.008	0.456	0.107	**0.008**	0.210	0.057
18	1.106	0.907	0.778	3.460	0.764	0.778	2.687	0.763	0.778	2.739	0.762	0.760	0.778	3.104	0.803	0.778	2.357	**0.760**
19	8.52e+09	1.07e+09	8.45e+09	8.50e+09	2.39e+09	**0.010**	8.50e+09	470.199	0.010	8.50e+09	390.041	6.25e+07	8.48e+09	8.50e+09	7.07e+09	0.010	8.50e+09	1.13e+09
20	2.02e+03	951.846	240.710	3.87e+03	945.409	**0.010**	3.87e+03	761.708	0.010	3.87e+03	843.597	123.973	285.465	3.88e+03	3.87e+03	0.010	3.88e+03	1.45e+03
21	0.315	0.260	0.038	0.242	0.198	0.021	0.242	0.157	0.021	0.243	0.123	0.071	0.008	0.230	0.057	**0.008**	0.125	0.087
22	8.81e+06	8.27e+05	7.90e+06	8.82e+06	4.76e+05	2.24e+05	8.82e+06	8.15e+06	2.24e+05	8.82e+06	5.46e+05	8.95e+05	8.32e+06	8.82e+06	1.80e+06	**4.72e+03**	8.82e+06	8.14e+06
23	9.41e+06	7.08e+05	8.40e+06	9.41e+06	3.48e+05	2.26e+05	9.41e+06	1.12e+06	2.25e+05	9.41e+06	8.34e+06	8.05e+05	8.90e+06	9.41e+06	9.51e+05	**4.17e+03**	9.41e+06	8.68e+06
24	3.079	2.527	0.591	46.076	0.814	0.061	40.770	0.135	0.061	42.701	0.771	0.130	**0.007**	49.283	0.026	0.007	38.791	0.060
25	2.88e+04	4.19e+03	1.10e+04	3.13e+04	129.055	71.156	3.15e+04	6.24e+03	72.284	3.14e+04	1.03e+03	485.072	1.20e+04	3.17e+04	39.320	**0.010**	3.14e+04	1.51e+04
26	0.665	0.405	0.009	0.383	0.194	0.009	0.387	0.134	0.016	0.379	0.123	0.031	**0.008**	0.451	0.035	0.009	0.455	0.047
27	0.747	0.209	0.026	0.172	0.120	0.031	0.193	0.109	0.042	0.187	0.144	0.020	**0.006**	0.220	0.040	0.008	0.425	0.030
28	0.641	0.279	0.035	0.208	0.237	0.043	0.209	0.205	0.063	0.208	0.310	0.018	**0.006**	0.222	0.027	0.007	0.441	0.029
29	0.584	0.268	0.064	0.170	0.207	0.072	0.180	0.183	0.092	0.186	0.260	0.027	**0.006**	0.073	0.045	0.009	0.286	0.038
30	0.621	0.239	0.034	0.195	0.168	0.040	0.236	0.189	0.044	0.208	0.147	0.021	**0.007**	0.200	0.066	0.008	0.344	0.035
31	0.775	0.262	0.042	0.188	0.169	0.048	0.211	0.200	0.058	0.209	0.179	0.029	**0.006**	0.105	0.056	0.008	0.319	0.045
32	0.670	0.258	0.081	0.193	0.206	0.086	0.196	0.261	0.097	0.199	0.270	0.035	**0.006**	0.065	0.053	0.009	0.197	0.048
33	0.664	0.450	0.076	0.286	0.350	0.088	0.287	0.438	0.106	0.297	0.462	0.046	**0.007**	0.141	0.046	0.010	0.370	0.050
34	0.482	0.138	0.008	0.268	0.076	0.008	0.263	0.052	0.008	0.256	0.135	**0.005**	0.008	0.624	0.011	0.006	0.536	0.021
35	0.771	0.249	0.057	0.163	0.173	0.062	0.174	0.166	0.073	0.171	0.178	0.035	**0.006**	0.099	0.100	0.010	0.342	0.045
36	0.663	0.269	0.077	0.173	0.233	0.084	0.178	0.197	0.093	0.179	0.260	0.029	**0.007**	0.057	0.055	0.008	0.218	0.045
37	0.599	0.137	0.008	0.352	0.092	0.009	0.346	0.054	0.010	0.319	0.151	0.008	0.007	0.605	0.026	**0.006**	0.615	0.024
38	0.703	0.453	0.071	0.272	0.255	0.085	0.277	0.340	0.106	0.278	0.288	0.051	**0.008**	0.133	0.062	0.011	0.349	0.059
39	0.727	0.199	0.013	0.148	0.178	0.015	0.151	0.095	0.028	0.159	0.193	0.019	**0.007**	0.251	0.043	0.007	0.282	0.036
40	0.767	0.202	0.037	0.135	0.173	0.040	0.135	0.129	0.049	0.141	0.221	0.025	**0.006**	0.105	0.049	0.008	0.224	0.041
41	0.681	0.465	0.070	0.286	0.317	0.081	0.306	0.245	0.102	0.306	0.442	0.047	**0.007**	0.148	0.055	0.011	0.388	0.051
42	0.651	0.240	0.057	0.172	0.241	0.061	0.176	0.195	0.068	0.184	0.262	0.033	**0.006**	0.055	0.072	0.009	0.128	0.046
43	0.544	0.223	0.015	0.195	0.124	0.017	0.220	0.115	0.023	0.195	0.101	0.013	0.007	0.295	0.023	**0.007**	0.422	0.023
44	0.659	0.209	0.015	0.221	0.143	0.019	0.264	0.164	0.023	0.251	0.229	0.016	**0.006**	0.340	0.054	0.007	0.377	0.033
45	0.647	0.391	0.009	0.470	0.275	0.009	0.511	0.122	0.009	0.488	0.101	0.012	0.007	0.649	0.018	**0.006**	0.577	0.020
46	0.828	0.249	0.031	0.214	0.146	0.035	0.255	0.140	0.046	0.247	0.172	0.025	**0.006**	0.207	0.032	0.008	0.437	0.037
47	0.490	0.211	0.009	0.202	0.122	0.010	0.227	0.092	0.012	0.214	0.210	0.008	**0.006**	0.404	0.014	0.006	0.406	0.018
48	0.627	0.266	0.101	0.214	0.269	0.105	0.222	0.272	0.112	0.228	0.277	0.035	**0.007**	0.068	0.103	0.009	0.198	0.054
49	0.726	0.258	0.044	0.194	0.162	0.048	0.244	0.161	0.055	0.230	0.286	0.030	**0.006**	0.115	0.085	0.008	0.324	0.047
50	0.817	0.247	0.043	0.165	0.175	0.048	0.199	0.157	0.058	0.191	0.183	0.034	**0.006**	0.100	0.066	0.010	0.321	0.045

#	SVM-L	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	RF:LV:SVM-L	RF:LV:SVM	RF:LV:NN	RF:PV:SVM-L	RF:PV:SVM	RF:PV:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	XG:LV:SVM-L	XG:LV:SVM	XG:LV:NN
51	0.666	0.427	0.067	0.268	0.288	0.080	0.285	0.394	0.105	0.294	0.245	0.049	**0.008**	0.101	0.051	0.013	0.326	0.060
52	0.809	0.150	0.009	0.501	0.066	0.012	0.561	0.041	0.013	0.548	0.094	0.008	0.007	0.662	0.028	**0.006**	0.614	0.024
53	0.658	0.279	0.102	0.194	0.231	0.107	0.205	0.222	0.116	0.202	0.300	0.029	**0.007**	0.056	0.054	0.008	0.206	0.047
54	0.761	0.200	0.017	0.252	0.143	0.021	0.289	0.102	0.026	0.271	0.203	0.022	**0.007**	0.324	0.044	0.008	0.439	0.043
55	0.704	0.255	0.013	0.309	0.083	0.013	0.339	0.117	0.018	0.336	0.208	0.017	**0.007**	0.457	0.031	0.007	0.526	0.031
56	0.711	0.228	0.029	0.205	0.162	0.035	0.260	0.170	0.044	0.242	0.254	0.019	**0.006**	0.245	0.026	0.007	0.405	0.033
57	0.678	0.244	0.029	0.209	0.165	0.033	0.257	0.156	0.040	0.245	0.293	0.025	**0.006**	0.174	0.038	0.010	0.381	0.044
58	0.751	0.232	0.048	0.154	0.189	0.052	0.165	0.212	0.063	0.163	0.205	0.027	**0.006**	0.135	0.095	0.008	0.409	0.039
59	0.637	0.271	0.095	0.190	0.220	0.104	0.198	0.267	0.110	0.199	0.297	0.030	**0.006**	0.067	0.038	0.008	0.279	0.044
60	0.649	0.253	0.009	0.442	0.036	0.009	0.454	0.044	0.009	0.449	0.036	0.007	0.008	0.589	0.007	**0.005**	0.494	0.013
61	0.596	0.253	0.083	0.177	0.213	0.087	0.179	0.249	0.093	0.186	0.260	0.032	**0.006**	0.071	0.054	0.009	0.219	0.044
62	0.736	0.271	0.040	0.183	0.181	0.046	0.202	0.193	0.060	0.202	0.315	0.035	**0.007**	0.100	0.044	0.010	0.296	0.054
63	0.715	0.260	0.009	0.520	0.097	0.009	0.525	0.165	0.009	0.510	0.062	0.010	0.008	0.695	0.029	**0.006**	0.558	0.024
64	0.579	0.222	0.038	0.170	0.183	0.043	0.168	0.175	0.053	0.173	0.230	0.017	**0.006**	0.178	0.043	0.007	0.310	0.030
65	0.619	0.218	0.036	0.161	0.165	0.040	0.170	0.218	0.048	0.172	0.231	0.025	**0.006**	0.151	0.041	0.008	0.274	0.037
66	0.787	0.280	0.052	0.169	0.133	0.054	0.181	0.281	0.062	0.177	0.310	0.034	**0.007**	0.075	0.089	0.011	0.264	0.054
67	0.675	0.283	0.039	0.192	0.167	0.044	0.202	0.157	0.059	0.202	0.304	0.024	**0.007**	0.154	0.036	0.008	0.346	0.040
68	0.673	0.405	0.009	0.276	0.238	0.012	0.283	0.271	0.027	0.284	0.236	0.029	**0.007**	0.288	0.033	0.009	0.422	0.034
69	0.473	0.116	0.009	0.395	0.058	0.009	0.430	0.127	0.009	0.372	0.127	0.004	0.008	0.687	**0.003**	0.005	0.595	0.013
70	0.746	0.359	0.009	0.350	0.111	0.009	0.348	0.159	0.011	0.343	0.193	0.022	**0.007**	0.504	0.029	0.008	0.508	0.033
71	0.613	0.262	0.041	0.198	0.165	0.044	0.211	0.119	0.057	0.220	0.317	0.016	**0.007**	0.236	0.035	0.007	0.441	0.032
72	0.682	0.224	0.043	0.158	0.160	0.049	0.171	0.169	0.055	0.159	0.224	0.018	**0.006**	0.115	0.033	0.007	0.281	0.036
73	0.678	0.248	0.036	0.192	0.196	0.039	0.195	0.151	0.044	0.195	0.251	0.020	**0.007**	0.278	0.048	0.008	0.423	0.031
74	0.656	0.214	0.009	0.284	0.161	0.011	0.324	0.146	0.018	0.322	0.147	0.013	0.006	0.477	0.036	**0.006**	0.508	0.034
75	0.727	0.232	0.025	0.192	0.149	0.034	0.224	0.259	0.049	0.230	0.241	0.022	**0.006**	0.323	0.034	0.008	0.501	0.042
76	0.717	0.216	0.028	0.188	0.152	0.032	0.227	0.194	0.036	0.208	0.220	0.022	**0.006**	0.165	0.056	0.007	0.387	0.039
77	0.596	0.207	0.010	0.173	0.164	0.012	0.184	0.204	0.025	0.187	0.222	0.010	0.006	0.378	0.013	**0.006**	0.399	0.021
78	0.514	0.201	0.008	0.145	0.100	0.008	0.166	0.201	0.010	0.162	0.202	0.005	0.007	0.363	0.026	**0.005**	0.378	0.020
79	0.629	0.377	0.018	0.306	0.202	0.023	0.325	0.350	0.036	0.314	0.372	0.029	**0.007**	0.253	0.041	0.009	0.430	0.048
80	0.555	0.376	0.009	0.274	0.167	0.009	0.293	0.333	0.019	0.285	0.214	0.025	**0.007**	0.308	0.028	0.008	0.428	0.028
81	0.543	0.268	0.009	0.558	0.074	0.008	0.586	0.013	0.009	0.556	0.068	**0.005**	0.009	0.736	0.014	0.006	0.604	0.007
82	0.593	0.342	0.009	0.362	0.131	0.009	0.376	0.172	0.009	0.370	0.305	0.015	0.007	0.570	0.013	**0.006**	0.551	0.022
83	0.640	0.406	0.017	0.301	0.198	0.028	0.307	0.369	0.057	0.318	0.198	0.039	**0.007**	0.279	0.048	0.009	0.397	0.048
84	0.428	0.177	0.008	0.418	0.082	0.009	0.464	0.161	0.009	0.441	0.154	0.009	0.007	0.502	0.021	**0.006**	0.551	0.019
85	0.684	0.183	0.019	0.170	0.120	0.020	0.178	0.120	0.026	0.167	0.129	0.013	0.007	0.325	0.064	**0.007**	0.332	0.035
86	0.606	0.227	0.019	0.309	0.149	0.021	0.385	0.090	0.024	0.355	0.083	0.012	0.006	0.432	0.028	**0.006**	0.418	0.038
87	6.59e+06	1.08e+06	6.90e+06	8.41e+06	8.40e+06	0.010	8.41e+06	496.137	0.010	8.41e+06	**0.008**	2.40e+05	7.54e+06	8.41e+06	2.55e+06	0.010	8.41e+06	1.50e+05
88	1.35e+03	250.772	168.051	2.55e+03	16.167	21.915	2.51e+03	137.136	21.969	2.50e+03	112.406	3.821	0.013	2.58e+03	0.130	**0.010**	2.52e+03	85.251
89	63.306	35.884	10.795	77.516	**0.590**	0.752	78.754	8.168	0.717	78.821	10.198	16.272	3.426	81.809	0.646	0.792	75.315	4.406
90	4.595	2.270	0.730	5.671	0.692	0.345	5.609	0.319	0.351	5.686	0.632	0.847	0.009	6.446	0.114	**0.009**	5.082	0.066
91	9.577	5.728	0.020	10.840	0.083	0.010	10.675	0.127	0.010	10.801	0.248	1.546	0.010	11.315	0.044	**0.010**	10.217	0.022
92	225.537	35.398	0.277	199.232	1.139	**0.010**	198.687	20.921	0.010	199.083	21.180	3.678	0.010	206.530	0.250	0.010	205.759	34.392
93	2.74e+03	881.742	588.550	8.43e+03	796.388	265.675	8.43e+03	2.31e+03	298.385	8.43e+03	1.97e+03	198.785	103.218	8.69e+03	8.96e+03	**0.045**	8.69e+03	1.96e+03
94	2.47e+03	751.751	195.847	2.37e+03	1.612	1.514	2.34e+03	664.985	2.351	2.36e+03	498.668	80.496	10.969	2.41e+03	0.048	**0.010**	2.26e+03	573.020
95	3.74e+03	1.42e+03	520.422	9.98e+03	10.436	1.508	9.94e+03	930.595	1.508	9.95e+03	1.79e+03	171.595	326.841	1.00e+04	0.380	**0.010**	9.96e+03	2.59e+03
96	37.647	28.326	17.112	121.184	8.123	11.489	122.301	21.974	11.458	121.886	29.135	19.718	14.311	138.301	**7.467**	10.365	135.234	47.905

Tabela 6.9: Resultados em *datasets* de regressão - in sample

#	SVM-L	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	RF:LV:SVM-L	RF:LV:SVM	RF:LV:NN	RF:PV:SVM-L	RF:PV:SVM	RF:PV:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	XG:LV:SVM-L	XG:LV:SVM	XG:LV:NN
1	**0.254**	0.492	0.293	0.557	0.284	0.296	0.543	0.263	0.294	0.551	0.258	0.257	0.359	0.679	0.257	0.458	0.508	0.286
2	0.418	0.422	0.435	0.417	0.406	0.424	0.417	0.412	0.424	0.418	0.413	0.396	0.449	0.393	0.407	0.454	0.390	**0.387**
3	0.396	0.462	0.378	0.423	0.433	**0.376**	0.423	0.404	0.377	0.420	0.406	0.385	0.407	0.406	0.389	0.418	0.395	0.399
4	4.029	2.974	2.775	2.708	2.649	2.777	2.642	2.594	2.775	2.689	**2.580**	2.646	2.688	2.805	2.681	2.766	2.630	2.646
5	577.455	519.209	540.690	958.667	610.384	540.841	956.571	730.768	538.216	957.645	581.735	433.578	**341.726**	969.797	468.572	438.587	967.060	669.282
6	7.429	**2.037**	2.599	25.161	2.456	2.641	25.114	3.452	2.621	25.152	2.891	2.762	3.468	24.541	2.914	3.759	24.516	9.207
7	**3.536**	4.225	8.070	7.542	9.945	9.994	7.467	6.279	10.063	7.540	8.129	4.621	7.269	8.221	6.556	7.125	8.109	6.715
8	1.24e+07	**4.16e+06**	2.05e+07	3.42e+07	5.78e+06	6.10e+06	3.42e+07	7.71e+06	6.81e+06	3.42e+07	2.02e+07	4.57e+06	2.35e+07	3.42e+07	3.45e+07	4.91e+06	3.42e+07	9.00e+06
9	1.24e+07	**4.16e+06**	2.05e+07	3.42e+07	3.45e+07	6.12e+06	3.42e+07	4.23e+06	6.82e+06	3.42e+07	4.95e+06	4.53e+06	2.35e+07	3.42e+07	3.45e+07	4.98e+06	3.42e+07	3.58e+07
10	0.437	0.390	0.353	0.654	**0.330**	0.379	0.513	0.360	0.362	0.578	0.370	0.349	0.395	0.676	0.375	0.427	0.449	0.376
11	185.189	114.451	**93.165**	637.857	129.964	155.543	639.735	97.775	154.569	639.154	122.236	119.081	137.166	646.167	105.679	158.179	641.808	119.798
12	8.620	6.598	2.715	15.388	3.002	2.620	16.756	3.724	**2.596**	16.703	3.710	2.789	3.584	20.946	3.638	6.473	22.716	3.396
13	1.26e+04	6.46e+03	4.53e+03	1.48e+04	4.28e+03	7.20e+03	1.48e+04	9.34e+03	6.99e+03	1.48e+04	7.19e+03	**3.13e+03**	3.92e+03	1.50e+04	3.30e+03	4.29e+03	1.48e+04	1.49e+04
14	1.01e+05	3.68e+04	3.15e+04	1.01e+05	4.54e+04	3.35e+04	1.01e+05	3.73e+04	3.37e+04	1.01e+05	4.88e+04	2.45e+04	2.64e+04	1.02e+05	**1.78e+04**	2.45e+04	1.01e+05	6.61e+04
15	20.716	7.554	3.396	191.485	3.437	5.819	167.430	5.538	5.996	172.046	4.214	1.500	**0.778**	214.914	0.850	2.541	172.152	2.000
16	7.935	**2.204**	2.753	4.863	2.254	3.020	4.280	2.764	3.021	4.473	2.704	2.666	3.425	6.329	3.111	3.460	4.500	3.419
17	0.645	0.551	0.709	0.556	0.491	0.750	0.553	0.489	0.740	0.555	0.499	**0.427**	0.480	0.564	0.466	0.523	0.430	0.464
18	**0.538**	0.585	0.586	2.945	0.584	0.586	2.261	0.580	0.586	2.284	0.588	0.592	0.622	2.861	0.562	0.589	2.371	0.576
19	1.25e+09	1.72e+09	1.24e+09	1.24e+09	3.77e+09	1.66e+09	1.24e+09	8.31e+09	1.55e+09	1.24e+09	6.75e+09	1.49e+09	1.24e+09	1.24e+09	**1.00e+09**	1.99e+09	1.24e+09	8.72e+09
20	1.78e+03	1.90e+03	1.69e+03	3.53e+03	3.75e+03	1.79e+03	3.53e+03	3.42e+03	1.79e+03	3.53e+03	3.27e+03	**1.61e+03**	1.66e+03	3.53e+03	3.66e+03	1.69e+03	3.53e+03	3.82e+03
21	0.271	0.246	0.310	0.233	0.222	0.413	0.235	0.212	0.416	0.235	0.224	**0.179**	0.185	0.227	0.201	0.202	0.183	0.185
22	1.51e+07	2.09e+06	1.34e+07	1.51e+07	**1.22e+06**	2.48e+06	1.51e+07	1.37e+07	2.44e+06	1.51e+07	1.77e+06	2.09e+06	1.43e+07	1.51e+07	2.73e+06	1.79e+07	1.51e+07	1.37e+07
23	1.51e+07	1.83e+06	1.33e+07	1.51e+07	**1.71e+06**	1.74e+06	1.51e+07	2.71e+06	1.77e+06	1.51e+07	1.29e+07	2.42e+06	1.43e+07	1.51e+07	3.52e+06	4.97e+07	1.51e+07	1.37e+07
24	**1.193**	3.335	1.922	42.094	2.409	3.803	37.413	2.294	3.707	38.917	1.700	5.278	7.193	45.333	6.661	4.199	35.821	4.338
25	1.07e+04	6.07e+03	3.48e+03	1.25e+04	2.62e+03	2.78e+03	1.26e+04	8.06e+03	**2.35e+03**	1.26e+04	3.04e+03	4.98e+03	3.25e+03	1.27e+04	2.44e+03	3.05e+03	1.26e+04	1.52e+04
26	0.875	0.626	0.354	0.659	0.521	0.381	0.640	0.446	0.421	0.642	0.402	0.267	**0.213**	0.696	0.240	0.340	0.578	0.274
27	0.679	0.341	0.143	0.278	0.225	0.135	0.286	0.210	0.150	0.281	0.259	0.152	**0.134**	0.296	0.148	0.154	0.467	0.152
28	0.577	0.242	0.165	0.190	0.219	0.168	0.191	0.192	0.148	0.185	0.267	0.081	0.074	0.231	**0.068**	0.120	0.416	0.085
29	0.617	0.330	0.174	0.230	0.271	0.175	0.242	0.246	0.177	0.250	0.310	0.106	**0.073**	0.141	0.103	0.132	0.316	0.099
30	0.967	0.308	0.135	0.285	0.208	0.144	0.347	0.222	0.140	0.302	0.182	0.099	**0.089**	0.382	0.127	0.118	0.662	0.111
31	0.679	0.298	0.129	0.231	0.196	0.129	0.255	0.230	0.130	0.255	0.215	0.080	**0.059**	0.136	0.086	0.119	0.343	0.090
32	0.638	0.272	0.125	0.222	0.242	0.126	0.227	0.284	0.138	0.228	0.292	0.081	**0.055**	0.115	0.090	0.106	0.219	0.086
33	0.653	0.382	0.206	0.261	0.310	0.194	0.264	0.371	0.172	0.272	0.388	0.128	**0.098**	0.189	0.105	0.178	0.356	0.130
34	0.971	0.331	0.221	0.402	0.324	0.210	0.379	0.257	**0.187**	0.391	0.357	0.218	0.195	0.686	0.210	0.220	0.545	0.227
35	0.897	0.346	0.168	0.263	0.251	0.162	0.285	0.222	0.146	0.276	0.243	0.102	**0.078**	0.218	0.185	0.143	0.549	0.122
36	0.573	0.280	0.208	0.195	0.241	0.199	0.204	0.219	0.187	0.201	0.275	0.065	**0.044**	0.094	0.069	0.092	0.260	0.080
37	0.582	0.268	0.208	0.624	0.195	0.217	0.571	0.189	0.233	0.556	0.258	**0.141**	0.171	0.977	0.186	0.194	0.954	0.153
38	0.665	0.521	0.203	0.342	0.304	0.198	0.337	0.402	0.192	0.341	0.336	0.106	**0.080**	0.174	0.094	0.141	0.414	0.103
39	0.899	0.390	0.222	0.334	0.381	0.230	0.324	0.229	0.234	0.338	0.404	0.144	**0.116**	0.414	0.138	0.149	0.403	0.155
40	0.715	0.221	0.165	0.198	0.209	0.154	0.190	0.184	0.145	0.206	0.253	0.095	**0.068**	0.151	0.109	0.094	0.220	0.104
41	0.709	0.432	0.220	0.303	0.336	0.204	0.317	0.293	0.175	0.314	0.418	0.110	**0.074**	0.177	0.089	0.142	0.342	0.101
42	0.605	0.257	0.122	0.184	0.255	0.120	0.190	0.185	0.120	0.196	0.284	0.067	**0.046**	0.087	0.093	0.076	0.154	0.079
43	0.636	0.315	0.140	0.301	0.179	0.156	0.329	0.207	0.161	0.301	0.196	0.115	0.108	0.397	**0.105**	0.122	0.377	0.151
44	1.171	0.509	0.224	0.672	0.340	0.226	0.775	0.378	0.211	0.750	0.530	0.212	**0.200**	0.841	0.249	0.207	0.991	0.274
45	0.992	0.403	0.342	0.452	0.471	0.354	0.453	0.416	0.341	0.440	0.362	0.301	**0.270**	0.637	0.275	0.300	0.437	0.294
46	0.742	0.283	0.150	0.216	0.166	0.148	0.242	0.163	0.137	0.240	0.201	0.088	0.078	0.229	**0.077**	0.110	0.395	0.090
47	0.936	0.405	0.274	0.416	0.299	0.281	0.430	0.259	0.282	0.422	0.431	0.205	0.189	0.631	**0.170**	0.204	0.555	0.202
48	0.615	0.280	0.177	0.232	0.291	0.176	0.244	0.289	0.174	0.253	0.302	0.074	**0.051**	0.121	0.129	0.088	0.220	0.094
49	0.976	0.295	0.124	0.235	0.163	0.126	0.314	0.171	0.115	0.288	0.330	0.086	**0.071**	0.180	0.118	0.133	0.477	0.095
50	0.790	0.245	0.152	0.185	0.191	0.142	0.222	0.162	0.142	0.216	0.191	0.065	**0.051**	0.129	0.086	0.106	0.353	0.074

#	SVM-L	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	RF:LV:SVM-L	RF:LV:SVM	RF:LV:NN	RF:PV:SVM-L	RF:PV:SVM	RF:PV:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	XG:LV:SVM-L	XG:LV:SVM	XG:LV:NN
51	0.742	0.555	0.224	0.362	0.370	0.202	0.378	0.494	0.191	0.387	0.317	0.106	**0.066**	0.173	0.086	0.146	0.450	0.118
52	0.957	0.265	**0.102**	0.581	0.132	0.113	0.733	0.121	0.105	0.683	0.177	0.117	0.106	0.586	0.108	0.114	0.592	0.123
53	0.614	0.310	0.180	0.249	0.276	0.173	0.253	0.268	0.181	0.250	0.327	0.065	**0.047**	0.094	0.094	0.075	0.223	0.086
54	0.623	0.283	0.155	0.252	0.244	0.156	0.248	0.211	0.154	0.246	0.261	0.092	**0.068**	0.321	0.086	0.098	0.416	0.118
55	0.835	0.329	0.252	0.515	0.202	0.268	0.585	0.190	0.260	0.585	0.337	**0.151**	0.159	0.614	0.168	0.196	0.703	0.196
56	1.305	0.283	0.169	0.333	0.207	0.168	0.464	0.247	0.153	0.417	0.308	0.136	0.141	0.492	**0.131**	0.161	0.709	0.140
57	0.849	0.317	0.203	0.285	0.252	0.209	0.335	0.267	0.218	0.330	0.378	0.114	**0.082**	0.233	0.103	0.127	0.374	0.123
58	0.866	0.354	0.159	0.260	0.276	0.166	0.290	0.316	0.160	0.289	0.316	0.131	**0.102**	0.221	0.145	0.138	0.485	0.127
59	0.828	0.372	0.269	0.260	0.292	0.266	0.269	0.383	0.244	0.277	0.425	0.094	**0.065**	0.134	0.078	0.126	0.414	0.098
60	1.262	0.888	0.825	1.155	0.827	0.830	1.126	0.879	0.820	1.133	0.828	0.705	0.713	1.473	**0.673**	0.775	1.265	0.743
61	0.609	0.261	0.174	0.202	0.232	0.178	0.203	0.255	0.168	0.209	0.261	0.103	**0.077**	0.137	0.097	0.127	0.214	0.105
62	0.784	0.336	0.157	0.234	0.219	0.152	0.252	0.242	0.142	0.254	0.378	0.085	**0.057**	0.151	0.075	0.116	0.354	0.093
63	0.896	0.713	0.461	0.723	0.618	0.438	0.714	0.721	**0.428**	0.707	0.491	0.495	0.476	0.785	0.500	0.471	0.610	0.469
64	0.620	0.317	0.187	0.288	0.289	0.186	0.275	0.273	0.186	0.282	0.304	0.148	**0.134**	0.309	0.144	0.150	0.350	0.143
65	0.465	0.289	0.201	0.219	0.254	0.190	0.221	0.294	0.178	0.228	0.299	0.130	**0.100**	0.254	0.137	0.192	0.403	0.152
66	0.787	0.391	0.136	0.262	0.201	0.140	0.270	0.373	0.133	0.267	0.389	0.085	**0.056**	0.109	0.112	0.097	0.298	0.098
67	0.502	0.276	0.185	0.195	0.197	0.171	0.199	0.185	0.161	0.197	0.278	0.087	**0.064**	0.158	0.098	0.111	0.320	0.101
68	0.766	0.520	0.296	0.401	0.353	0.297	0.403	0.413	0.284	0.410	0.359	0.187	**0.140**	0.434	0.144	0.202	0.517	0.185
69	0.882	0.449	0.396	0.620	0.469	0.397	0.584	0.369	0.409	0.578	0.397	0.368	0.348	0.704	**0.306**	0.386	0.612	0.382
70	0.839	0.484	0.341	0.459	0.354	0.325	0.462	0.411	0.321	0.457	0.407	0.276	0.261	0.597	**0.255**	0.284	0.604	0.287
71	0.769	0.349	0.227	0.267	0.204	0.247	0.277	0.212	0.211	0.282	0.393	0.157	**0.130**	0.275	0.143	0.169	0.304	0.150
72	0.749	0.306	0.191	0.253	0.248	0.180	0.260	0.260	0.180	0.252	0.310	0.099	**0.091**	0.226	0.099	0.110	0.334	0.126
73	0.794	0.291	0.142	0.231	0.248	0.143	0.241	0.235	0.135	0.247	0.303	0.115	**0.101**	0.331	0.111	0.129	0.332	0.118
74	0.691	0.293	0.217	0.345	0.263	0.196	0.392	0.271	0.193	0.399	0.288	**0.169**	0.173	0.524	0.199	0.191	0.491	0.181
75	0.707	0.332	0.165	0.248	0.176	0.158	0.283	0.345	0.141	0.275	0.314	0.154	0.141	0.361	**0.137**	0.174	0.411	0.140
76	1.090	0.330	0.163	0.417	0.246	0.172	0.504	0.286	0.178	0.466	0.352	0.134	**0.114**	0.433	0.174	0.152	0.721	0.171
77	0.749	0.291	0.146	0.267	0.253	0.153	0.272	0.277	0.162	0.276	0.281	0.135	**0.123**	0.537	0.124	0.139	0.561	0.144
78	0.920	0.270	0.171	0.238	0.168	0.164	0.261	0.324	0.167	0.252	0.335	0.137	**0.132**	0.419	0.146	0.144	0.327	0.149
79	0.714	0.510	0.319	0.457	0.429	0.301	0.465	0.498	0.302	0.459	0.511	0.187	**0.142**	0.389	0.160	0.229	0.485	0.209
80	0.897	0.603	0.351	0.521	0.400	0.358	0.530	0.571	0.336	0.523	0.478	0.326	0.282	0.583	**0.257**	0.347	0.588	0.309
81	1.223	0.847	0.693	0.951	0.776	0.734	0.950	0.772	0.737	0.938	0.807	0.705	0.681	1.016	**0.664**	0.739	0.811	0.714
82	0.773	0.654	0.385	0.616	0.440	0.431	0.601	0.458	0.462	0.593	0.640	0.393	0.350	0.725	**0.345**	0.410	0.689	0.384
83	0.779	0.568	0.394	0.503	0.369	0.365	0.508	0.495	0.325	0.517	0.344	0.221	**0.162**	0.534	0.169	0.232	0.633	0.201
84	0.708	0.547	0.319	1.055	0.367	0.345	1.115	0.473	0.358	1.069	0.455	0.301	0.335	1.193	0.303	0.320	1.237	**0.294**
85	0.419	0.157	0.121	0.165	0.109	0.121	0.179	0.106	0.114	0.162	0.121	0.081	**0.080**	0.321	0.112	0.090	0.365	0.100
86	1.034	0.526	0.289	0.596	0.331	0.302	0.706	0.317	0.284	0.674	0.310	0.254	0.229	0.701	**0.224**	0.258	0.746	0.235
87	3.23e+06	1.23e+06	3.58e+06	4.53e+06	4.65e+06	9.94e+05	4.53e+06	3.65e+06	9.37e+05	4.53e+06	2.04e+06	1.22e+06	3.79e+06	4.53e+06	4.48e+06	**8.85e+05**	4.53e+06	3.10e+06
88	1.67e+03	456.766	420.107	2.88e+03	104.613	55.896	2.83e+03	433.128	55.006	2.83e+03	332.064	18.750	**17.538**	2.91e+03	49.354	20.746	2.85e+03	302.370
89	126.142	**86.836**	121.213	110.921	131.965	154.473	110.639	146.036	153.577	111.010	161.749	110.952	138.855	112.095	138.107	145.082	111.541	136.268
90	4.272	3.525	4.222	6.091	4.787	8.608	6.169	5.014	8.126	6.226	4.488	**3.068**	3.326	7.037	3.598	3.346	5.864	4.026
91	**11.278**	12.577	16.882	13.322	22.425	18.246	13.182	19.763	17.307	13.272	21.691	12.385	14.064	13.409	14.237	14.114	13.092	13.471
92	499.816	**128.791**	170.089	290.755	137.939	168.905	290.241	253.865	170.280	290.557	143.173	230.946	242.309	295.791	221.714	289.051	295.200	169.512
93	2.36e+03	963.114	785.657	8.47e+03	959.883	519.656	8.46e+03	3.37e+03	537.494	8.46e+03	1.81e+03	458.836	**357.307**	8.66e+03	8.65e+03	360.225	8.66e+03	1.65e+03
94	1.29e+03	938.185	729.747	1.10e+03	1.10e+03	1.09e+03	1.09e+03	1.08e+03	1.04e+03	1.10e+03	1.23e+03	695.886	660.224	1.10e+03	**603.626**	634.195	1.07e+03	741.955
95	2.30e+03	**1.98e+03**	2.20e+03	4.45e+03	2.41e+03	2.67e+03	4.45e+03	4.18e+03	2.63e+03	4.45e+03	3.59e+03	2.87e+03	3.08e+03	4.51e+03	3.38e+03	3.43e+03	4.47e+03	4.62e+03
96	**41.522**	43.470	58.861	159.877	71.514	73.508	161.585	44.786	74.122	160.973	46.000	48.238	57.520	180.666	64.003	66.509	177.244	76.934

Tabela 6.10: Resultados em *datasets* de regressão - out of sample

7
Conclusões e Trabalhos Futuros

Neste trabalho investigamos a abordagem $2PL$ para a construção de um novo aprendizado sob árvores de decisão treinadas no conjunto de dados original. Este trabalho propôs a criação de um conjunto de características, *feature construction*, a partir das folhas destas árvores e a aplicação de uma função estimadora nestes novos dados.

A partir dos experimentos, demonstrados no capítulo 6, os algoritmos definidos para o $2PL$ com melhores resultados no conjunto de treino apresentam também os melhores resultados no conjunto de teste, $out-of-sample$ *dataset*, avaliados pela acurácia média dos classificadores e pela média de erros entre as tarefas de regressão. Essa consistência de bons resultados foi observada nas diferentes configurações e experimentos realizados, independente do número de árvores do modelo.

Estas observações sugerem que a aplicação desta metodologia de transformação dos dados no conjunto de árvores de decisão, realizada na primeira fase, e a aplicação de uma função estimadora, em especial o SVM linear, permite uma generalização melhor nas estimativas do modelo.

Esperava-se um desempenho melhor do algoritmo de SVM não linear, considerando uma flexibilidade maior frente o modelo linear, entretanto isso não foi observado nos experimentos. O mesmo raciocínio seguiu para as redes neurais, experimentos demonstraram que o modelo linear apresentou as melhores estimativas. Considerando que alguns exemplos exigiam um processamento muito elevado para encontrar o estimador, algumas funções ótimas não foram alcançadas nos modelos SVM-R e NN, deixando em aberto a possibilidade de ajustes nos modelos a fim de obter preditores melhores, e por consequência, melhorar as respostas do $2PL$.

Algumas questões ficam sugeridas para trabalhos futuros. Uma delas, em relação ao preditor do $2PL$, é avaliar a geração de novos preditores a partir de múltiplos conjuntos de árvores de decisão, ou seja, combinar em passos de iteração vários *tree ensemble models*. Em outras palavras, aplicar alguma metodologia de *boosting*, por exemplo, no modelo definido, pode melhorar o desempenho do $2PL$? Outra questão em aberto é saber a possibilidade de construir árvores de decisão utilizando outras abordagens. As árvores devem

procurar mais a generalização ou a minimização das funções de perda? Como o algoritmo será seguido por algum outro preditor, esse pode ser o caso.

Referências bibliográficas

[1] BERTSIMAS, D.; DUNN, J.. **Optimal classification trees**. Machine Learning, 106(7):1039–1082, 2017.

[2] BISHOP, C. M.. **Pattern Recognition and Machine Learning**. Information science and statistics. Springer, 1st ed. 2006. corr. 2nd printing edition, 2006.

[3] BREIMAN, L.. **Random Forests**. Machine Learning, 45(1):5–32, 2001.

[4] BREIMAN, L.. **Bagging predictors**. Machine learning, 24(2):123–140, 1996.

[5] BREIMAN, L.. **Classification and regression trees**. Routledge, 2017.

[6] BUITINCK, L.; LOUPPE, G.; BLONDEL, M. ; PEDREGOSA, F.. **API design for machine learning software: experiences from the scikitlearn project**. In: ECML PKDD Workshop: Languages for Data Mining and Machine Learning, p. 108–122, 2013.

[7] CHEN, T.; GUESTRIN, C.. **XGBoost: A Scalable Tree Boosting System**. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '16, p. 785–794, New York, NY, USA, 2016. Association for Computing Machinery.

[8] CHIO, C.; FREEMAN, D.. **Machine Learning and Security - Protecting Systems with Data and Algorithms**. O'Reilly Media, 02 2018.

[9] DIETTERICH, T. G.; OTHERS. **Ensemble learning**. The handBook of brain theory and neural networks, 2:110–125, 2002.

[10] FREUND, Y.; SCHAPIRE, R. ; ABE, N.. **A short introduction to boosting**. Journal-Japanese Society For Artificial Intelligence, 14(771-780):1612, 1999.

[11] FRIEDMAN, J. H.. **Multivariate adaptive regression splines**. The annals of statistics, p. 1–67, 1991.

[12] FRIEDMAN, J. H.. **Greedy Function Approximation: A Gradient Boosting Machine**. In: IMS, 1999.

[13] JEROME FRIEDMAN, T. H. R. T.. **The Elements of Statistical Learning**. 2001.

[14] KANTCHELIAN, A.; TYGAR, J. D. ; JOSEPH, A. D.. **Evasion and Hardening of Tree Ensemble Classifiers**. CoRR, abs/1509.07892, 2015.

[15] KOZAK, J.. **Decision Tree and Ensemble Learning Based on Ant Colony Optimization**. Springer, 2019.

[16] MISIĆ, V. V.. **Optimization of Tree Ensembles**. 2017.

[17] NAN, F.; WANG, J. ; SALIGRAMA, V.. **Feature-Budgeted Random Forest**. In: Bach, F.; Blei, D., editors, Proceedings of the 32nd International Conference on Machine Learning, volumen 37 de **Proceedings of Machine Learning Research**, p. 1983–1991, Lille, France, 07-09 Jul 2015. PMLR.

[18] PATRYK ORZECHOWSKI, W. L. C.; MOORE, J. H.. **Where are we now? A large benchmark study of recent symbolic regression methods.** In: Genetic and Evolutionary Computation Conference (GECCO '18), Kyoto, Japan, 2018. ACM.

[19] PRETORIUS, A.; BIERMAN, S. ; STEEL, S.. **A bias-variance analysis of ensemble learning for classification**. 12 2016.

[20] SAFAVIAN, S. R.; LANDGREBE, D.. **A survey of decision tree classifier methodology**. IEEE Transactions on Systems, Man, and Cybernetics, 21(3):660–674, 1991.

[21] STUART RUSSELL, P. N.. **Artificial Intelligence: A Modern Approach**. Prentice Hall Series in Artificial Intelligence. Prentice Hall, 3rd edition, 2010.

[22] SUN, X.; REN, X.; MA, S. ; WANG, H.. **MeProp: Sparsified Back Propagation for Accelerated Deep Learning with Reduced Overfitting**. CoRR, abs/1706.06197, 2017.

[23] USTINOVSKIY, Y.; FEDOROVA, V.; GUSEV, G. ; SERDYUKOV, P.. **Meta–Gradient Boosted Decision Tree Model for Weight and Target Learning**. In: Balcan, M. F.; Weinberger, K. Q., editors, Proceedings of The 33rd International Conference on Machine Learning, volumen 48 de **Proceedings of Machine Learning Research**, p. 2692–2701, New York, New York, USA, 20-22 Jun 2016. PMLR.

[24] VENS, C.; COSTA, F.. **Random forest based feature induction**. In: 2011 IEEE 11TH INTERNATIONAL CONFERENCE ON DATA MINING, p. 744–753, 2011.

A
Material complementar com configurações de 50, 100 e 500 árvores de decisão

Esta seção define análises complementares, bem como os experimentos realizados nos métodos de conjunto usando as configurações de 50, 100 e 500 árvores. As tabelas mostram os resultados médios obtidos, nos conjuntos de teste, em cada uma dessas parametrizações, ordenadas pelos melhores algoritmos. Resultados individuais de classificação e regressão são apresentados após as médias. A primeira coluna dessas tabelas referem-se aos datasets identificados no capítulo de resultados, em 6.1 e 6.2.

É importante ressaltar que os valores médios representam a acurácia média entre os conjuntos de dados de classificação e a média do erro quadrático entre os exemplos de regressão. Uma observação, nos exemplos com 500 árvores os algoritmos considerados em análise foram reduzidos a um escopo menor, e ainda, alguns exemplos de regressão foram desconsiderados.

#	Average
XG:BL:SVM-L:M	84.765
XG:BL:SVM-L	84.656
RF:EL:SVM-L:M	84.650
RF:BL:SVM-L:M	84.640
RF:BL:SVM-L	84.590
RF	84.089
XG:BL:NN	83.606
RF:BL:NN	83.507
XG:EL:SVM-L:M	83.232
XG:BL:NN:M	83.023
XG	82.903
RF:EL:SVM:M	82.087
RF:BL:SVM:M	82.057
XG:EL:SVM:M	81.848
RF:BL:NN:M	81.349
XG:BL:SVM:M	80.995
RF:EL:NN:M	80.587
XG:EL:NN:M	80.258
RF:BL:SVM	79.865
XG:BL:SVM	78.311

Tabela A.1: Média na classificação - out of sample - 50 árvores

#	No. of winners
XG:BL:SVM-L	23
XG:BL:SVM-L:M	21
RF:BL:NN	18
XG:EL:NN:M	17
RF:EL:SVM-L:M	17
RF:BL:SVM-L	17
RF:BL:SVM-L:M	17
RF:BL:NN:M	17
RF:EL:NN:M	16
RF:EL:SVM:M	15
RF:BL:SVM	15
XG:BL:NN	15
RF	14
RF:BL:SVM:M	14
XG:BL:SVM:M	12
XG	12
XG:BL:NN:M	12
XG:BL:SVM	11
XG:EL:SVM-L:M	11
XG:EL:SVM:M	9

Tabela A.2: Campeões na classificação - out of sample - 50 árvores

#	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	RF:BL:SVM:M	RF:BL:SVM-L:M	RF:BL:NN:M	RF:EL:SVM:M	RF:EL:SVM-L:M	RF:EL:NN:M	XG:BL:SVM:M	XG:BL:SVM-L:M	XG:BL:NN:M	XG:EL:SVM:M	XG:EL:SVM-L:M	XG:EL:NN:M
1	**1.000**	**1.000**	0.967	**1.000**	0.833	**1.000**	0.833	**1.000**	0.967	**1.000**	**1.000**	0.967	**1.000**	**1.000**	0.833	**1.000**	**1.000**	0.833	0.833	**1.000**
2	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
3	0.834	0.803	0.841	0.841	0.917	0.885	0.904	0.911	0.854	0.803	0.866	0.854	0.803	0.904	0.904	0.860	**0.936**	0.911	0.911	0.885
4	0.711	0.695	0.706	0.706	0.711	**0.738**	0.706	0.706	0.706	0.695	0.706	0.706	0.701	0.706	0.706	**0.738**	0.706	0.706	0.706	0.706
5	**0.988**	**0.988**	0.985	0.980	0.980	**0.988**	0.980	0.959	0.985	**0.988**	0.980	0.985	**0.988**	0.980	0.980	**0.988**	0.968	0.980	0.974	0.977
6	0.972	0.965	0.965	0.951	0.951	**0.979**	0.951	0.944	0.965	0.965	0.951	0.965	0.965	0.951	0.951	**0.979**	0.937	0.951	0.951	0.951
7	**0.760**	**0.760**	0.740	0.720	0.740	0.640	0.740	0.660	0.740	**0.760**	0.680	0.740	**0.760**	0.680	0.740	0.640	0.640	0.740	0.720	0.740
8	0.750	0.722	**0.764**	**0.764**	0.736	0.722	**0.764**	0.722	**0.764**	0.722	0.736	**0.764**	0.722	**0.764**	**0.764**	0.722	0.722	**0.764**	**0.764**	**0.764**
9	0.981	0.986	0.914	0.972	0.914	**0.998**	0.907	0.940	0.954	0.986	0.898	0.954	0.986	0.933	0.931	0.988	0.914	0.928	0.984	0.826
10	**0.996**	**0.996**	0.994	0.987	0.950	0.986	0.965	0.964	0.994	**0.996**	0.987	0.994	**0.996**	0.987	0.965	0.986	0.962	0.966	0.964	0.962
11	0.948	0.941	0.926	0.941	0.941	**0.956**	0.926	**0.956**	0.926	0.941	0.948	0.926	0.941	0.941	0.926	**0.956**	0.948	0.926	0.948	0.926
12	0.963	0.963	**0.972**	0.954	0.963	**0.972**	0.963	0.954	**0.972**	0.963	**0.972**	**0.972**	0.963	0.963	0.963	**0.972**	0.954	0.963	0.954	0.963
13	0.769	0.827	0.769	0.827	0.846	**0.904**	0.769	0.885	0.769	0.827	0.827	0.769	0.827	0.808	0.769	**0.904**	0.827	0.885	0.885	0.865
14	0.827	0.821	**0.861**	0.855	**0.861**	0.827	0.855	0.809	**0.861**	0.821	0.827	**0.861**	0.821	0.832	0.855	0.827	0.809	**0.861**	0.838	0.844
15	0.778	0.800	0.719	0.770	0.778	**0.837**	0.733	**0.837**	0.719	0.800	0.763	0.726	0.800	0.763	0.733	**0.837**	0.800	0.785	0.785	0.578
16	0.935	0.935	0.924	0.924	0.967	0.967	0.880	0.946	0.924	0.935	0.913	0.924	0.935	0.913	0.946	**0.978**	0.967	0.946	**0.978**	0.880
17	0.903	0.911	0.548	0.798	0.758	**0.931**	0.504	0.790	0.847	0.911	0.371	0.847	0.911	0.431	0.831	0.927	0.738	0.839	0.927	0.734
18	0.515	0.499	0.477	0.439	0.526	0.509	0.477	0.547	0.526	0.501	0.439	0.528	0.501	0.439	0.545	0.509	0.496	**0.566**	0.528	0.439
19	0.394	0.333	**0.485**	0.364	0.333	0.273	**0.485**	0.394	0.394	0.333	0.364	0.394	0.333	0.364	0.394	0.303	0.364	0.455	0.364	**0.485**
20	0.960	0.960	0.920	0.920	0.920	**1.000**	0.920	0.920	0.920	0.960	0.920	0.920	0.960	0.920	0.920	**1.000**	0.920	0.920	0.920	0.920
21	0.636	0.610	0.597	0.597	0.636	**0.649**	0.597	0.610	0.597	0.610	0.597	0.597	0.610	0.597	0.597	**0.649**	0.636	0.597	0.597	0.597
22	**0.848**	0.818	0.606	0.818	**0.848**	0.818	0.455	0.788	0.818	0.818	**0.848**	0.788	0.818	0.818	**0.848**	0.818	0.818	0.758	0.818	0.667
23	0.526	0.513	0.526	0.526	0.566	0.539	0.526	0.553	0.526	0.526	0.526	0.526	0.526	0.526	0.539	0.526	0.539	0.553	0.526	**0.579**
24	0.868	**0.925**	0.717	0.830	0.755	0.830	0.566	0.849	0.811	0.906	0.868	0.811	0.906	0.830	0.774	0.868	0.755	0.736	0.868	0.736
25	0.678	**0.685**	**0.685**	**0.685**	0.658	0.644	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	0.644	0.623	0.644	0.658	**0.685**
26	0.795	0.769	0.744	**0.821**	0.769	0.744	0.744	0.744	0.744	0.769	**0.821**	0.744	0.769	**0.821**	0.744	0.744	0.718	0.744	0.795	0.744
27	0.932	0.943	0.909	0.920	0.920	**0.955**	0.920	0.920	0.909	0.943	0.920	0.909	0.943	0.920	0.920	**0.955**	0.920	0.920	0.920	0.920
28	**0.974**	**0.974**	**0.974**	**0.974**	0.921	0.921	0.921	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	0.921	0.921	0.947	0.921	0.921	**0.974**
29	0.797	0.788	0.809	0.817	0.813	0.805	0.817	**0.822**	0.809	0.788	0.809	0.809	0.788	0.780	0.817	0.805	**0.822**	0.813	0.817	**0.822**
30	0.839	**1.000**	0.806	**1.000**	0.774	**1.000**	0.742	0.935	0.806	**1.000**	0.935	0.806	**1.000**	**1.000**	0.742	**1.000**	0.935	0.742	0.742	0.677
31	0.814	**0.884**	0.535	0.837	0.558	0.674	0.535	0.651	0.535	**0.884**	0.837	0.535	**0.884**	0.535	0.535	0.674	0.744	0.535	0.535	0.535
32	0.935	0.968	0.903	0.968	0.935	0.935	0.935	0.935	0.903	0.968	**1.000**	0.903	0.968	0.968	0.935	0.935	0.935	0.935	0.935	0.935
33	0.938	0.953	0.832	0.893	0.922	**0.964**	0.924	0.919	0.923	0.952	0.700	0.923	0.952	0.748	0.934	0.962	0.907	0.947	**0.964**	0.907
34	0.940	0.931	**0.945**	**0.945**	0.943	0.938	**0.945**	**0.945**	**0.945**	0.931	**0.945**	**0.945**	0.931	**0.945**	**0.945**	0.938	**0.945**	0.942	0.940	**0.945**
35	0.968	0.965	**0.970**	**0.970**	**0.970**	0.962	**0.970**	**0.970**	**0.970**	0.965	**0.970**	**0.970**	0.965	**0.970**	**0.970**	0.962	**0.970**	**0.970**	0.967	**0.970**
36	0.918	**0.959**	0.918	0.918	**0.959**	0.918	0.898	0.939	0.918	**0.959**	0.918	0.918	**0.959**	0.918	0.898	0.918	0.918	0.939	0.918	0.878
37	0.678	**0.685**	**0.685**	**0.685**	0.658	0.644	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	**0.685**	0.644	0.637	0.644	0.658	**0.685**
38	0.609	**0.652**	0.587	0.630	0.587	0.543	0.587	0.565	0.587	**0.652**	0.609	0.587	**0.652**	0.587	0.587	0.543	0.543	0.587	0.587	0.587
39	**0.886**	0.879	0.883	0.875	0.871	0.860	**0.886**	0.841	0.883	0.879	0.879	0.883	0.879	**0.886**	**0.886**	0.860	0.848	0.871	0.875	0.720
40	0.943	0.943	0.887	**0.962**	0.943	**0.962**	0.943	0.943	0.943	0.943	**0.962**	**0.962**	0.943	**0.962**	0.943	**0.962**	0.943	0.943	**0.962**	**0.962**
41	0.943	0.920	**0.950**	**0.950**	**0.950**	0.927	**0.950**	**0.950**	**0.950**	0.920	**0.950**	**0.950**	0.920	**0.950**	**0.950**	0.927	**0.950**	**0.950**	**0.950**	**0.950**
42	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	0.750	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**
43	0.550	0.550	0.550	0.600	**0.850**	0.800	0.500	**0.850**	0.550	0.550	0.650	0.550	0.550	0.600	0.500	0.800	**0.850**	0.500	0.600	0.750
44	0.750	0.750	0.600	**0.850**	**0.850**	0.750	0.450	0.800	0.600	0.750	**0.850**	0.600	0.750	0.800	0.450	0.750	**0.850**	0.650	0.800	**0.850**
45	0.732	0.724	0.704	0.704	**0.748**	0.736	0.708	0.724	0.704	0.724	0.704	0.704	0.724	0.704	0.708	0.736	0.728	0.720	**0.748**	0.704
46	0.893	0.902	0.830	0.874	0.876	**0.906**	0.869	0.867	0.890	**0.906**	0.537	0.889	**0.906**	0.365	0.872	**0.906**	0.861	0.874	0.905	0.846
47	0.500	0.526	0.184	0.526	0.447	0.579	0.184	0.447	0.526	0.553	0.447	0.526	0.553	0.184	0.447	**0.632**	0.474	0.526	0.500	0.184
48	**0.822**	**0.822**	**0.822**	**0.822**	**0.822**	0.805	**0.822**	**0.822**	**0.822**	**0.822**	**0.822**	**0.822**	**0.822**	**0.822**	**0.822**	0.805	**0.822**	**0.822**	**0.822**	**0.822**
49	**0.997**	0.996	0.994	**0.997**	0.996	**0.997**	0.995	0.984	0.994	0.996	0.966	0.994	0.996	0.968	0.995	**0.997**	0.992	0.996	**0.997**	0.958
50	0.926	0.926	0.870	0.926	0.963	**1.000**	0.870	0.944	0.889	0.926	0.926	0.907	0.926	0.926	0.870	0.981	0.981	0.852	0.963	0.944
51	**0.937**	0.936	0.921	0.922	0.919	0.929	0.919	0.911	0.921	0.936	0.924	0.921	0.936	0.923	0.919	0.929	0.911	0.917	0.915	0.913
52	0.954	0.983	0.812	0.642	0.883	**0.992**	0.812	0.925	0.812	0.983	0.642	0.812	0.983	0.967	0.812	**0.992**	0.892	0.942	0.933	0.642
53	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
54	**0.978**	**0.978**	**0.978**	**0.978**	0.911	0.956	0.911	0.889	**0.978**	**0.978**	0.956	**0.978**	**0.978**	0.956	0.911	0.956	0.889	0.867	0.956	0.889

Tabela A.3: Resultados em *datasets* de classificação - out of sample - 50 árvores

#	Average
RF:PV:SVM-L	9366.585
RF	9405.814
RF:LV:SVM-L	9747.692
RF:BL:NN	10070.177
XG:BL:NN	10950.428
XG:LV:NN	11293.207
RF:PV:NN	11333.238
RF:LV:NN	12022.431
XG:BL:SVM-L	12321.998
XG:LV:SVM-L	21223.086
RF:BL:SVM-L	21298.246
XG:BL:SVM	22031.821
XG:LV:SVM	22223.497
RF:LV:SVM	22243.441
RF:PV:SVM	22251.338
RF:BL:SVM	22252.489
XG	41666.241

Tabela A.4: Média na regressão - out of sample - 50 árvores

#	No. of winners
RF	28
RF:PV:SVM-L	22
XG:BL:NN	10
XG:BL:SVM-L	9
XG:LV:NN	7
RF:PV:NN	5
RF:BL:NN	5
RF:LV:NN	3
RF:BL:SVM-L	3
RF:LV:SVM-L	2
XG:LV:SVM-L	1
RF:PV:SVM	1
XG	0
XG:BL:SVM	0
RF:LV:SVM	0
RF:BL:SVM	0
XG:LV:SVM	0

Tabela A.5: Campeões na regressão - out of sample - 50 árvores

#	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	RF:LV:SVM-L	RF:LV:SVM	RF:LV:NN	RF:PV:SVM-L	RF:PV:SVM	RF:PV:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	XG:LV:SVM-L	XG:LV:SVM	XG:LV:NN
1	**0.265**	0.341	2.127	2.318	0.350	0.760	0.342	0.328	1.134	0.295	6.371	0.304	0.719	0.376	0.299	0.569	0.342
2	0.427	0.490	0.616	0.674	0.470	0.460	0.646	0.474	0.500	0.653	1.141	0.444	0.438	**0.416**	0.450	0.422	0.418
3	0.410	**0.403**	0.806	0.920	0.426	0.478	0.919	0.418	0.551	0.918	1.278	0.404	0.490	0.413	0.414	0.472	0.529
4	2.671	2.706	3.629	2.655	2.771	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	2.694	2.969	**2.615**	2.769	2.615	2.683
5	39.303	29.607	87.849	46.198	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	35.876	82.332	45.061	57.376	82.099	46.337
6	**33.677**	49.802	129.568	42.372	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	46.001	124.872	37.700	37.322	119.020	35.918
7	11.165	16.589	19.724	15.690	15.160	19.247	15.966	15.876	19.599	15.837	25.478	**9.355**	19.147	16.853	23.197	17.852	15.292
8	1134.106	1211.786	1457.159	1320.176	1271.163	1455.079	1337.880	1292.879	1456.363	1511.416	4106.115	**1008.477**	1440.404	1467.091	1661.626	1459.140	1438.767
9	1134.615	1215.430	1457.163	1322.800	1271.048	1455.068	1344.840	1295.122	1456.364	1514.772	4106.115	**1008.477**	1440.404	1433.952	1661.626	1459.140	1470.151
10	**91.861**	344.854	629.644	93.686	119.565	622.239	98.744	102.824	627.932	93.525	1086.988	126.570	594.757	197.894	349.660	602.525	139.117
11	188.897	181.952	275.034	234.194	214.115	274.783	188.378	223.815	275.179	202.100	437.570	**120.589**	251.652	205.932	183.660	252.780	144.225
12	939.532	2678.960	3330.957	972.317	938.208	3332.585	1662.481	**922.855**	3333.382	1176.175	6849.109	1551.867	3268.656	1323.410	1710.428	3329.195	2349.738
13	**809.022**	905.763	1019.133	1010.064	815.675	1018.616	1214.062	859.120	1019.251	1343.836	2462.279	840.330	1014.709	1248.636	1363.475	1017.363	1204.579
14	210.021	148.862	138.488	245.198	210.673	138.345	214.507	219.018	**138.310**	241.868	362.152	178.766	138.786	415.018	348.365	138.619	316.574
15	1088.878	1819.397	2292.282	1196.849	1170.354	2283.789	1432.708	1148.163	2291.253	1262.770	4483.964	1134.099	2258.874	**849.050**	1189.758	2265.174	918.868
16	**13.550**	14.299	23.525	14.623	15.523	18.605	14.917	14.462	19.679	14.277	47.029	14.082	18.313	13.920	13.984	16.038	13.737
17	519.226	685.379	787.767	502.379	516.298	748.464	812.995	507.368	769.154	660.889	1012.596	490.551	783.079	**463.072**	503.675	748.751	664.255
18	3.286	3.456	3.490	3.309	3.360	3.471	3.266	3.415	3.477	**3.248**	4.389	3.663	3.667	3.339	3.688	3.673	3.315
19	190.070	318.350	485.301	291.565	244.583	482.104	233.899	237.011	484.918	232.901	952.864	221.378	475.633	212.641	430.344	477.327	**177.155**
20	**277.899**	290.172	294.644	303.977	339.466	294.719	535.194	327.805	294.727	364.552	549.944	331.860	294.614	494.006	445.414	294.325	501.396
21	**4411.062**	10109.608	11031.900	4774.816	4972.351	11012.092	6633.744	4727.651	11027.701	6704.388	23848.568	6090.975	10939.055	7994.009	15912.491	10990.240	6723.768
22	**2746.028**	4221.644	5992.534	3135.342	3054.177	5438.770	5292.336	3013.691	5835.805	3716.191	4241.233	3705.400	5963.517	2828.715	3450.207	5462.049	3163.566
23	**2430.474**	3158.008	4103.790	2970.686	2906.626	3854.245	4374.756	2928.124	4049.186	3273.460	3032.853	2960.019	4090.360	2869.621	3509.179	3820.233	3248.381
24	775.383	1916.917	2522.242	762.051	697.534	2485.259	1316.969	710.408	2518.466	1049.807	3260.350	935.424	2420.197	516.490	**412.413**	2454.066	870.664
25	591.778	689.483	898.589	623.200	543.588	897.852	429.629	508.657	898.764	**373.289**	2254.681	541.634	878.663	430.399	611.170	888.960	449.685
26	**4541.614**	5385.036	5615.599	5258.197	5238.984	5616.351	10091.658	5075.026	5616.364	5828.447	12280.704	5080.148	5597.533	5852.808	6089.676	5616.002	6476.691
27	**8529.780**	18140.408	19415.859	9275.225	9511.304	19418.395	13140.574	8715.087	19418.513	12315.697	34598.641	10557.763	19217.737	9770.486	11823.907	19416.999	9179.090
28	8088.439	19354.782	20705.572	7991.539	8153.952	20708.864	14194.072	**7724.451**	20708.910	8979.363	36557.031	10652.043	20476.262	9545.484	12020.780	20705.275	8814.996
29	35527.409	87565.288	90424.146	38907.869	38029.831	90428.989	38753.925	37173.980	90429.081	38113.126	155167.660	46892.275	89636.466	**35359.670**	69824.847	90271.563	44327.863
30	9569.780	18694.643	19948.929	10174.235	9668.490	19951.902	15164.634	**9097.280**	19951.978	13515.820	38327.386	10728.031	19728.332	10611.075	14542.093	19949.486	9170.504
31	30803.008	84640.099	87791.384	32936.048	31279.866	87796.134	35079.502	**30247.544**	87796.970	35110.031	136417.321	43422.788	86728.374	34199.498	83682.812	87731.439	35199.682
32	24090.431	77011.403	80117.602	**23749.483**	26385.234	80122.852	28973.574	24447.774	80122.928	28691.816	142752.429	35823.094	78987.444	35064.360	96026.402	80120.668	33692.137
33	**55499.262**	81896.902	83242.020	59383.532	57219.803	83244.330	59709.016	56677.384	83244.330	61361.063	154128.602	65339.423	82993.581	66570.011	85902.259	83243.148	71114.333
34	**582.672**	628.675	718.095	677.177	591.789	718.408	1321.700	593.571	718.432	823.233	1260.115	601.990	711.979	897.983	870.919	717.841	613.567
35	**26564.867**	77347.086	80152.305	32238.424	28314.132	80156.865	36001.646	27304.097	80156.952	31304.993	156033.756	38698.481	79437.308	32987.576	116978.558	80153.324	32194.204
36	27457.010	83707.517	86883.754	27076.800	28020.208	86888.513	34734.081	**26796.392**	86889.221	30114.196	139620.244	42192.460	85969.681	35040.311	95697.970	86707.741	27916.997
37	461.164	636.116	851.299	477.656	452.576	851.334	629.850	444.740	852.317	476.781	1731.967	501.555	836.385	567.783	671.188	846.704	**314.970**
38	39229.416	78200.934	80252.146	40630.398	38768.145	80255.656	40867.464	**37908.101**	80255.658	48355.561	161471.171	52816.765	79946.094	48451.680	62915.256	80254.819	45622.426
39	2757.583	4733.188	5318.572	2865.860	2615.697	5320.352	2389.847	2481.793	5320.910	**2268.007**	11560.457	3204.798	5230.505	3143.449	3467.535	5311.031	3315.338
40	**6626.457**	19082.151	20730.981	8460.382	8303.458	20732.555	10297.134	7259.556	20734.784	9141.036	39912.553	9138.794	20537.037	9216.418	12004.617	20711.597	7986.970
41	52908.464	82395.216	83989.265	50842.658	52083.724	83992.026	60972.776	49537.533	83992.026	54937.222	162789.407	61045.816	83671.681	**49246.660**	66169.417	83992.026	56120.330
42	**19990.332**	85504.946	89218.054	22232.166	23620.507	89221.019	27154.389	21090.420	89224.580	23435.404	146626.600	38440.669	87851.569	20598.753	49357.257	89042.621	24827.569
43	3073.484	4437.282	5001.876	3454.516	2820.546	5001.877	3474.079	**2787.621**	5003.311	3494.806	8652.711	3658.061	4966.308	3472.456	3334.727	4992.628	3269.766
44	2443.727	4526.958	5068.356	2472.508	2451.715	5069.934	4671.932	2308.777	5070.124	2513.979	10840.395	2996.402	5019.021	2421.734	3117.267	5067.718	**2303.047**
45	4028.925	5149.477	5322.620	4443.030	4265.957	5323.142	11014.160	4132.131	5323.143	7897.504	12445.525	**3945.853**	5306.536	4931.048	5494.791	5323.114	5260.549
46	8433.543	19295.173	20724.386	9897.634	8565.410	20726.963	9232.435	8217.167	20727.877	9417.368	40404.816	9532.684	20531.990	8859.650	12217.518	20721.205	**8079.529**
47	2891.912	4887.020	5354.960	2762.369	**2624.688**	5356.449	5766.116	2665.356	5356.532	3923.234	10761.095	3502.535	5294.984	3264.103	3677.180	5354.512	3430.109
48	28348.218	81701.957	84826.124	29584.556	29080.280	84831.431	30926.337	**27226.176**	84831.513	35176.458	154545.073	39582.532	83807.823	31032.787	79406.126	84814.589	32709.989
49	**25209.163**	82431.376	85397.764	28187.439	25490.562	85402.763	28170.109	25343.801	85402.779	28857.189	162952.382	38818.046	84691.770	31119.265	62148.247	85357.794	32539.195
50	28714.609	80227.381	83199.135	32861.160	30535.463	83203.885	36518.550	**28631.996**	83204.167	36204.848	157222.423	40409.656	82266.545	37244.497	108607.279	83110.850	33971.692
51	40078.556	76606.864	78590.856	44462.915	43072.594	78594.369	50196.354	**40070.904**	78594.394	49648.545	167727.896	54931.514	78132.206	42945.158	88624.603	78590.592	55414.457
52	744.271	853.709	1041.434	612.262	554.995	1041.730	853.944	601.537	1042.141	604.085	2215.328	855.633	1037.946	**397.489**	528.179	1041.517	819.339
53	**26409.910**	81549.500	84737.586	30140.182	28034.832	84739.900	32192.029	26948.538	84743.235	29864.193	139641.951	41297.550	83856.837	34065.411	63849.671	84468.254	33668.854
54	**2435.985**	4315.293	4869.331	2590.041	2575.094	4870.890	3638.124	2446.373	4871.144	3267.515	9839.630	2639.126	4792.437	2975.366	3279.831	4857.363	2563.536
55	3056.453	5406.265	5822.228	**2398.214**	3083.785	5823.438	5186.385	2755.622	5823.521	3701.237	9647.644	4166.122	5795.896	2625.571	3398.109	5823.087	3521.767
56	11934.917	20756.114	22044.140	13726.727	12818.933	22047.593	12992.848	12122.598	22047.624	13745.617	47694.506	14637.806	21831.719	**10523.427**	13258.820	22047.030	12420.320
57	**8340.698**	18102.978	19533.001	9452.213	8908.074	19534.285	11235.731	8951.221	19536.437	11018.025	42730.962	9725.153	19239.909	10660.000	18795.021	19507.624	9852.341
58	34541.494	83538.704	86306.279	35253.620	34833.698	86310.943	41253.510	**33984.674**	86310.970	40959.075	149351.015	47571.803	85305.972	39898.964	115868.099	86293.830	46031.199
59	30041.220	77920.358	80557.048	31826.611	31763.094	80561.429	40816.701	**29704.927**	80561.560	38238.583	159309.521	40557.499	79744.605	36280.379	80540.235	80548.991	37721.247
60	872.388	972.231	1063.802	885.441	755.189	1064.291	**727.138**	747.927	1064.375	861.356	2614.906	824.249	1059.618	1045.257	1105.934	1063.318	840.856
61	**39361.140**	87442.305	90318.573	44911.072	40294.210	90323.476	42990.003	40233.390	90323.506	42493.464	177556.610	45443.905	89474.715	47869.300	116253.058	90322.656	45190.809
62	25503.265	77049.700	80000.760	26700.134	27996.333	80005.508	28977.317	**25267.083**	80005.849	27601.641	150984.445	38942.937	78998.334	31610.390	86876.852	79933.789	26109.229
63	821.464	835.789	898.573	**688.214**	715.597	898.712	845.824	701.739	898.898	894.966	1891.567	803.854	895.721	890.127	973.627	898.250	813.860
64	**7961.592**	16525.353	17631.845	8379.871	8519.416	17634.189	11602.740	8030.762	17634.503	14784.006	40807.634	10077.361	17385.867	10509.667	15119.240	17633.226	10657.084
65	**7385.743**	21591.177	23164.378	7511.896	8468.504	23168.000	12569.000	8076.205	23168.156	10522.563	39939.465	11481.976	22879.403	8160.605	10805.121	23159.644	8652.239
66	26002.542	74222.798	77399.747	31439.873	25418.559	77405.078	28094.924	**25393.130**	77405.709	31496.805	164754.638	34435.157	76252.159	34305.960	91127.655	77216.254	35586.518
67	7485.051	17195.934	18543.383	7654.694	6834.689	18546.455	12077.397	**6780.142**	18546.793	9187.514	37469.474	10029.170	18165.327	10827.569	20137.584	18531.360	9973.720
68	16008.695	22199.298	22810.152	16618.655	16532.234	22811.839	25465.478	16176.705	22811.840	21152.261	41105.291	18195.724	22720.346	**15646.316**	16919.155	22811.817	19955.601
69	715.030	723.788	905.825	639.464	626.255	906.604	729.541	656.927	906.645	652.357	2109.270	696.427	897.086	747.092	902.081	905.566	**610.008**
70	**3684.749**	4886.426	5166.725	4555.296	4139.230	5167.687	10455.904	4098.523	5167.697	6331.656	10808.567	4006.236	5142.281	4926.817	4932.860	5167.387	5475.389
71	14260.884	21776.237	22712.941	14185.214	14937.709	22715.167	19972.765	**14088.956**	22715.181	15954.582	40187.526	16462.109	22537.099	16140.243	17533.986	22696.464	18570.137
72	11478.655	21277.626	22672.790	12560.840	10974.846	22674.792	12489.389	10998.652	22675.833	**10432.063**	42174.560	12895.610	22377.053	11620.593	15208.709	22642.794	10449.779
73	8273.293	18688.259	19950.485	**7152.196**	8026.339	19953.318	12686.434	7643.520	19953.471	8846.511	38588.905	10797.177	19734.611	10009.300	13276.726	19952.051	10565.966
74	2395.996	5286.982	5914.072	**1990.597**	2610.897	5916.081	6288.640	2368.209	5916.148	3490.531	9349.336	3319.825	5840.268	2487.437	2760.807	5915.454	2692.559
75	**11659.901**	17927.412	18901.825	12629.322	12536.004	18904.025	13946.801	12069.783	18904.149	15793.803	42234.803	11805.619	18731.454	13523.920	19447.157	18903.079	13625.001
76	8466.603	20979.571	22432.682	7491.001	7628.330	22435.887	11824.034	**7401.016**	22436.163	9918.780	41082.519	12358.484	22199.194	8790.149	11882.919	22434.570	9603.990
77	2589.096	4854.337	5446.265	3076.344	2774.251	5446.896	3457.938	2833.721	5448.029	3608.457	11425.185	2908.818	5387.248	3314.453	3184.015	5444.123	**2443.501**
78	3347.334	5437.409	5964.421	3136.741	3214.178	5965.871	5437.647	3075.034	5965.947	3713.556	10893.154	4000.617	5916.959	4033.737	3661.105	5965.514	**2811.690**
79	**13846.491**	21100.007	22024.501	15018.579	14666.306	22026.505	17770.441	13891.777	22026.519	18158.448	38412.011	17230.601	21919.368	17826.394	17001.796	22024.890	15506.174
80	17482.549	20627.433	21097.046	17982.658	17157.937	21098.191	25837.075	**17010.310**	21098.192	23439.132	44981.586	17027.188	21020.437	18143.863	19774.942	21098.176	20360.296
81	662.292	699.633	770.171	653.914	689.248	770.337	1369.436	**652.726**	770.350	892.353	1483.354	658.248	770.277	989.338	1039.749	769.158	1026.166
82	4933.677	4488.599	4593.687	5440.262	4732.737	4594.062	8880.486	4829.537	4594.063	7587.993	10936.921	**4432.712**	4585.299	4990.636	5458.122	4593.998	6124.677
83	13903.520	21718.148	22664.285	13460.022	13210.700	22666.818	14143.395	**12919.485**	22666.862	17705.928	49506.182	15843.008	22493.928	13135.395	16195.715	22665.887	15254.906
84	431.944	681.392	863.511	473.116	**398.379**	864.265	558.848	426.984	864.446	505.585	1743.923	532.586	853.034	664.458	718.462	862.340	539.144
85	**1403.777**	4732.449	5367.818	1584.357	1579.655	5369.243	4111.969	1418.808	5369.796	3391.454	11666.704	2007.458	5243.483	1657.951	2478.713	5358.913	2040.099
86	3619.393	5686.831	6153.689	3399.930	3524.633	6155.194	5090.773	**3366.586**	6155.248	4801.177	12228.484	4540.845	6098.731	3710.231	4674.434	6150.823	4445.870
87	88.504	100.193	181.785	93.270	59.008	181.605	**11.767**	79.632	182.102	47.336	480.228	133.735	179.191	85.428	132.571	180.322	103.943
88	**368.194**	461.660	694.058	432.053	425.129	688.636	441.594	395.498	693.997	480.797	1577.514	435.419	676.397	555.324	641.448	672.672	538.941
89	20.137	20.672	27.094	25.091	25.401	23.011	30.170	23.533	26.147	26.893	42.443	**17.842**	26.491	28.465	20.292	25.438	28.045
90	11841.366	**10779.118**	10922.357	13644.229	12276.202	10921.461	21547.633	12122.851	10922.140	19183.213	24318.720	11147.248	10907.625	14749.126	23856.364	10911.962	13271.133
91	92.097	98.784	127.717	100.308	104.752	126.334	103.845	97.921	126.839	**89.857**	159.046	117.869	124.509	127.339	101.327	120.066	122.695
92	183.694	192.604	199.396	236.172	217.264	199.278	370.536	212.864	199.416	332.348	428.810	197.002	200.267	**171.858**	201.239	200.307	202.518
93	180.229	2450.302	3277.247	168.482	180.730	3230.555	363.389	**152.061**	3259.545	264.931	6016.649	269.059	3132.329	190.328	216.248	3054.781	201.668
94	3486.609	3763.092	3970.672	4015.048	3701.983	3971.404	7167.117	3677.634	3971.428	5513.044	7938.371	**2938.247**	3950.233	3930.493	3932.687	3969.641	4467.914
95	494.786	**443.277**	463.137	557.074	613.336	463.245	822.061	589.181	463.250	625.780	990.051	529.929	464.010	866.013	741.680	463.248	875.741
96	47.133	69.740	180.847	53.807	63.575	160.822	56.835	61.412	166.028	61.098	337.767	**43.482**	127.665	47.587	47.708	120.121	45.929

Tabela A.6: Resultados em *datasets* de regressão - out of sample - 50 árvores

#	Average
RF:BL:SVM-L:M	84.611
RF:EL:SVM-L:M	84.597
XG:BL:SVM-L:M	84.588
RF:BL:SVM-L	84.472
XG:BL:SVM-L	84.372
RF	84.099
RF:EL:NN:M	83.880
RF:BL:NN:M	83.739
XG:EL:SVM-L:M	83.727
XG:BL:NN	83.618
XG:BL:NN:M	83.412
RF:BL:NN	83.360
XG	83.330
XG:EL:NN:M	81.303
XG:BL:SVM:M	80.603
RF:BL:SVM:M	80.554
RF:EL:SVM:M	80.496
XG:EL:SVM:M	79.885
XG:BL:SVM	76.576
RF:BL:SVM	75.787

Tabela A.7: Média na classificação - out of sample - 100 árvores

#	No. of winners
XG:BL:SVM-L	19
XG:BL:SVM-L:M	17
RF:BL:SVM-L	15
RF:EL:NN:M	14
RF	14
RF:BL:NN:M	14
RF:EL:SVM-L:M	13
RF:BL:SVM-L:M	13
RF:EL:SVM:M	11
XG	11
RF:BL:NN	11
RF:BL:SVM:M	11
XG:BL:SVM:M	10
RF:BL:SVM	10
XG:BL:SVM	9
XG:EL:NN:M	9
XG:EL:SVM-L:M	9
XG:BL:NN	8
XG:EL:SVM:M	7
XG:BL:NN:M	6

Tabela A.8: Campeões na classificação - out of sample - 100 árvores

#	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	RF:BL:SVM:M	RF:BL:SVM-L:M	RF:BL:NN:M	RF:EL:SVM:M	RF:EL:SVM-L:M	RF:EL:NN:M	XG:BL:SVM:M	XG:BL:SVM-L:M	XG:BL:NN:M	XG:EL:SVM:M	XG:EL:SVM-L:M	XG:EL:NN:M
1	**1.000**	**1.000**	0.833	**1.000**	0.833	**1.000**	0.833	**1.000**	0.833	**1.000**	**1.000**	0.833	**1.000**	**1.000**	0.833	**1.000**	**1.000**	0.833	**1.000**	**1.000**
2	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
3	0.822	0.790	0.739	0.841	**0.911**	0.885	0.904	0.904	0.771	0.796	0.892	0.771	0.796	0.898	0.904	0.866	0.898	0.898	0.885	0.904
4	0.695	0.701	0.706	0.706	0.722	0.701	0.706	**0.759**	0.706	0.701	0.706	0.706	0.706	0.706	0.706	0.701	0.706	0.706	0.727	0.706
5	0.988	0.988	0.983	0.980	0.980	**0.994**	0.980	0.980	0.983	0.988	0.974	0.983	0.988	0.980	0.980	**0.994**	0.962	0.980	0.980	0.980
6	**0.972**	0.965	0.958	0.958	0.951	0.965	0.951	0.951	0.958	0.965	0.951	0.958	0.965	0.958	0.951	0.965	0.951	0.958	0.951	0.951
7	0.700	0.680	0.740	0.680	**0.760**	0.640	0.740	0.620	0.740	0.680	0.660	0.740	0.680	0.680	0.740	0.640	0.660	0.740	0.640	0.680
8	0.750	0.708	**0.764**	0.708	0.750	0.708	**0.764**	0.708	**0.764**	0.722	**0.764**	**0.764**	0.708	0.708	**0.764**	0.708	0.736	**0.764**	**0.764**	**0.764**
9	0.986	0.988	0.896	0.972	0.914	**0.991**	0.845	0.935	0.938	0.988	0.919	0.938	0.988	0.928	0.873	0.988	0.940	0.873	0.988	0.900
10	**0.996**	**0.996**	0.990	0.987	0.950	0.991	0.962	0.970	0.990	**0.996**	0.987	0.989	**0.996**	0.987	0.962	0.991	0.950	0.961	0.971	0.962
11	0.948	0.941	0.926	0.948	0.941	0.948	0.926	0.948	0.926	0.941	0.948	0.926	0.941	**0.956**	0.926	0.948	0.948	0.926	**0.956**	0.926
12	0.963	0.963	**0.972**	0.954	0.963	0.954	0.963	0.954	**0.972**	0.963	0.954	**0.972**	0.963	0.954	0.963	0.954	0.954	0.954	0.954	0.954
13	0.769	0.827	0.788	0.827	0.865	**0.904**	0.788	0.885	0.788	0.827	0.827	0.788	0.827	0.827	0.788	**0.904**	0.827	0.827	0.865	0.846
14	0.832	0.832	**0.867**	0.838	0.855	0.832	**0.867**	**0.867**	**0.867**	0.832	0.832	**0.867**	0.832	0.827	**0.867**	0.832	0.832	0.850	0.838	**0.867**
15	0.800	**0.830**	0.741	0.778	0.778	0.822	0.741	0.778	0.741	**0.830**	0.778	0.741	**0.830**	0.778	0.741	0.815	0.815	0.719	0.793	0.785
16	0.935	0.935	0.870	0.924	0.967	0.967	0.859	0.967	0.924	0.935	0.935	0.924	0.935	0.924	0.924	**0.978**	0.957	0.935	**0.978**	0.880
17	0.911	0.903	0.069	0.806	0.806	**0.940**	0.242	0.823	0.827	0.919	0.657	0.827	0.919	0.774	0.831	0.935	0.794	0.839	0.931	0.794
18	0.526	0.509	0.439	0.439	0.534	0.450	0.472	0.537	**0.553**	0.515	0.439	**0.553**	0.520	0.439	0.537	0.477	0.550	0.523	0.507	0.482
19	0.333	0.333	**0.485**	0.364	0.273	0.303	**0.485**	0.364	0.333	0.333	0.364	0.333	0.333	0.333	0.424	0.364	0.333	0.424	0.364	0.303
20	**0.960**	**0.960**	0.920	0.920	0.920	0.880	0.920	**0.960**	0.920	**0.960**	0.920	0.920	**0.960**	0.920	0.920	0.880	**0.960**	0.920	0.920	0.920
21	0.623	0.584	0.597	**0.662**	0.636	0.610	0.597	0.649	0.597	0.584	0.584	0.597	0.584	**0.662**	0.597	0.610	0.649	0.597	0.649	0.597
22	0.788	0.818	0.394	0.818	0.818	0.818	0.394	0.818	0.788	0.818	0.818	0.788	0.818	0.788	**0.848**	0.818	0.788	**0.848**	0.788	0.788
23	0.539	0.513	0.526	0.500	0.566	0.526	0.526	0.500	0.526	0.526	**0.579**	0.526	0.526	0.513	0.526	0.526	0.539	0.539	0.487	0.539
24	0.887	**0.906**	0.604	0.868	0.792	0.887	0.453	0.774	0.830	0.887	0.868	0.830	0.887	0.868	0.774	0.887	0.755	0.736	0.887	0.736
25	0.671	0.678	0.685	0.685	0.664	**0.719**	0.685	0.623	0.685	0.678	0.685	0.685	0.678	0.705	0.685	**0.719**	0.630	0.685	0.671	0.685
26	0.744	0.769	0.744	**0.846**	0.795	0.769	0.744	0.795	0.744	0.769	**0.846**	0.744	0.769	**0.846**	0.744	0.769	0.769	0.744	0.821	0.744
27	0.909	0.943	0.932	0.920	0.932	**0.977**	0.920	0.932	0.932	0.943	0.920	0.920	0.943	0.920	0.920	**0.977**	0.920	0.932	0.932	0.920
28	**0.974**	**0.974**	**0.974**	**0.974**	0.921	0.921	0.921	0.921	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	**0.974**	0.921	0.921	0.921	0.921	0.921	0.921
29	0.788	0.793	0.714	0.817	0.822	0.805	**0.830**	0.826	0.714	0.793	0.809	0.718	0.788	0.805	**0.830**	0.805	0.826	0.817	0.826	0.826
30	0.903	**1.000**	0.806	0.935	0.806	**1.000**	0.710	0.968	0.806	**1.000**	0.935	0.806	**1.000**	0.935	0.710	**1.000**	0.968	0.710	0.710	0.710
31	**0.860**	**0.860**	0.535	0.535	0.558	0.767	0.535	0.767	0.535	**0.860**	**0.860**	0.535	**0.860**	0.837	0.535	0.767	0.767	0.535	0.605	0.535
32	0.968	0.968	0.871	0.968	0.935	0.935	0.935	0.935	0.871	0.968	**1.000**	0.871	0.968	0.968	0.935	0.935	0.935	0.935	0.935	0.935
33	0.946	0.957	0.792	0.895	0.932	**0.968**	0.900	0.915	0.903	0.959	0.866	0.903	0.959	0.897	0.909	0.967	0.915	0.931	0.964	0.927
34	0.942	0.938	**0.945**	0.938	0.943	0.940	**0.945**	0.940	**0.945**	0.938	0.943	**0.945**	0.938	0.943	**0.945**	0.940	0.937	**0.945**	0.942	**0.945**
35	**0.970**	0.961	**0.970**	**0.970**	**0.970**	0.962	**0.970**	**0.970**	**0.970**	0.961	**0.970**	**0.970**	0.961	**0.970**	**0.970**	0.962	**0.970**	**0.970**	0.959	**0.970**
36	0.939	**0.959**	0.816	0.918	0.939	0.918	0.857	0.918	0.816	**0.959**	0.918	0.816	**0.959**	0.918	0.857	0.918	0.918	0.776	0.898	0.898
37	0.671	0.678	0.685	0.699	0.664	**0.719**	0.685	0.623	0.685	0.678	0.671	0.685	0.678	0.699	0.685	**0.719**	0.644	0.685	0.671	0.685
38	**0.609**	0.587	0.587	0.587	0.587	0.543	0.587	0.543	0.587	0.587	0.587	0.587	0.587	**0.609**	0.587	0.543	0.522	0.587	0.522	0.543
39	0.886	0.867	0.848	0.860	0.871	0.867	**0.890**	0.845	0.848	0.867	0.879	0.848	0.867	0.860	**0.890**	0.867	0.848	0.864	0.867	0.879
40	**0.962**	**0.962**	**0.962**	**0.962**	0.943	**0.962**	0.943	0.943	**0.962**	0.943	**0.962**	**0.962**	0.943	**0.962**	0.943	**0.962**	0.943	0.943	**0.962**	0.943
41	0.943	0.924	0.950	0.950	**0.952**	0.927	0.950	0.950	0.950	0.924	0.950	0.950	0.924	0.950	0.950	0.927	0.950	0.950	0.935	0.950
42	**0.917**	**0.917**	0.500	**0.917**	**0.917**	**0.917**	0.500	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**	**0.917**
43	0.550	0.550	0.400	0.600	**0.850**	0.650	0.500	0.800	0.400	0.550	0.750	0.400	0.550	0.550	0.500	0.650	0.800	0.400	0.600	0.800
44	0.800	0.850	0.600	**0.900**	**0.900**	0.700	0.450	0.800	0.600	0.850	**0.900**	0.600	0.850	**0.900**	0.450	0.700	0.850	0.400	**0.900**	**0.900**
45	0.732	0.716	0.704	0.704	**0.740**	0.704	0.704	0.708	0.704	0.716	0.720	0.704	0.716	0.724	0.704	0.704	0.736	0.704	0.732	0.704
46	0.822	0.822	0.822	0.822	0.822	0.805	0.822	0.822	0.822	0.822	0.822	0.822	0.822	**0.831**	0.822	0.805	0.822	0.822	0.822	0.822
47	0.944	0.944	0.704	0.926	**1.000**	**1.000**	0.741	0.963	0.852	0.944	0.944	0.833	0.944	0.926	0.815	**1.000**	0.963	0.741	0.981	0.963
48	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
49	**0.978**	**0.978**	0.956	0.956	0.911	**0.978**	0.933	0.911	0.956	**0.978**	0.956	0.956	**0.978**	**0.978**	0.911	**0.978**	0.911	0.911	0.956	0.911
50	0.936	**0.941**	0.908	0.928	0.921	0.931	0.921	0.911	0.908	**0.941**	0.927	0.908	**0.941**	0.929	0.921	0.931	0.912	0.919	0.917	0.918
51	0.963	0.983	0.713	0.971	0.892	**0.996**	0.808	0.950	0.713	0.983	0.946	0.708	0.983	0.967	0.808	**0.996**	0.950	0.738	0.967	0.642
52	0.892	0.901	0.819	0.881	0.880	**0.911**	0.866	0.867	0.878	0.904	0.785	0.878	0.904	0.872	0.864	0.910	0.874	0.864	0.908	0.866
53	0.474	0.526	0.184	0.474	0.447	0.553	0.184	0.447	0.553	0.579	0.421	0.553	0.579	0.526	0.421	**0.605**	0.421	0.447	0.553	0.184
54	**0.997**	0.996	0.994	**0.997**	0.996	**0.997**	0.996	0.990	0.996	0.996	**0.997**	0.996	0.996	0.968	0.996	**0.997**	0.996	0.996	0.996	0.994

Tabela A.9: Resultados em *datasets* de classificação - out of sample - 100 árvores

#	Average
RF:PV:SVM-L	9120.614
RF	9287.651
RF:LV:SVM-L	9428.282
XG:BL:NN	9524.839
RF:BL:NN	10008.330
XG:BL:SVM-L	10756.448
RF:PV:NN	11686.526
RF:LV:NN	11880.406
XG:LV:NN	12157.668
XG:LV:SVM-L	13249.619
RF:BL:SVM-L	20429.289
XG:BL:SVM	22052.620
XG:LV:SVM	22241.634
RF:LV:SVM	22244.135
RF:PV:SVM	22251.307
RF:BL:SVM	22252.490
XG	22741.264

Tabela A.10: Média na regressão - out of sample - 100 árvores

#	No. of winners
RF	22
RF:PV:SVM-L	19
XG:BL:NN	18
RF:BL:SVM-L	9
XG:LV:NN	8
XG:BL:SVM-L	7
RF:BL:NN	6
XG:LV:SVM-L	2
RF:LV:NN	1
RF:PV:NN	1
RF:LV:SVM-L	1
RF:PV:SVM	1
XG:LV:SVM	1
XG	0
XG:BL:SVM	0
RF:LV:SVM	0
RF:BL:SVM	0

Tabela A.11: Campeões na regressão - out of sample - 100 árvores

#	RF	RF:BL:SVM-L	RF:BL:SVM	RF:BL:NN	RF:LV:SVM-L	RF:LV:SVM	RF:LV:NN	RF:PV:SVM-L	RF:PV:SVM	RF:PV:NN	XG	XG:BL:SVM-L	XG:BL:SVM	XG:BL:NN	XG:LV:SVM-L	XG:LV:SVM	XG:LV:NN
1	**0.263**	0.337	2.125	0.289	0.338	0.762	0.334	0.327	1.136	0.290	2.685	0.293	0.782	0.277	0.334	0.511	0.277
2	0.426	0.488	0.615	0.668	0.471	0.458	0.658	0.473	0.499	0.664	0.682	0.445	0.437	**0.405**	0.451	0.414	0.408
3	0.413	**0.403**	0.806	0.925	0.426	0.479	0.501	0.418	0.553	0.447	0.752	0.410	0.498	0.438	0.417	0.444	0.414
4	2.665	2.706	3.629	2.660	2.771	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	2.692	3.037	2.663	2.777	**2.599**	2.657
5	37.551	**27.970**	87.861	48.608	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	33.899	83.703	31.120	47.075	84.448	31.666
6	**34.894**	43.990	129.548	41.050	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	[illegible]	38.778	125.457	56.985	40.730	122.609	37.780
7	**11.093**	17.337	19.712	16.822	16.718	19.248	16.553	16.989	19.600	17.589	13.292	14.276	19.216	17.722	19.324	19.074	17.846
8	1090.315	**1029.556**	1457.219	1306.806	1284.485	1454.784	1285.947	1311.554	1456.257	1300.962	2390.534	1112.469	1442.568	1389.422	1471.854	1458.007	1413.234
9	1090.485	**1032.370**	1457.221	1307.063	1284.710	1454.773	1429.692	1312.731	1456.254	1471.015	2390.912	1112.161	1442.568	1390.548	1470.497	1458.007	1670.783
10	**92.149**	191.789	629.615	95.032	124.074	622.227	127.578	107.692	627.795	164.204	496.316	99.528	599.106	111.003	187.262	615.958	102.777
11	185.774	153.397	274.952	240.165	215.632	274.741	256.934	224.979	275.147	209.829	265.879	**118.225**	256.415	249.139	209.941	264.972	166.552
12	**908.405**	2146.117	3330.942	979.086	943.925	3333.009	1603.033	928.986	3333.430	1379.557	3910.082	1347.574	3281.238	1072.278	1526.828	3331.832	1415.183
13	**781.302**	830.321	1019.101	1016.823	874.695	1018.645	1329.279	887.372	1019.257	1357.686	1589.127	858.070	1015.426	1223.346	1061.310	1018.589	1356.674
14	210.592	163.871	138.496	245.037	208.127	138.371	201.438	212.873	**138.334**	221.405	258.941	159.297	138.820	393.138	410.095	138.560	342.752
15	1071.827	1472.264	2292.306	1172.961	1114.182	2285.299	1435.651	1091.688	2291.254	1302.347	2574.265	1028.705	2259.630	820.901	1192.871	2276.612	**777.859**
16	**13.494**	14.176	23.529	14.716	15.390	18.562	14.543	14.508	19.691	14.547	26.650	14.017	18.417	13.987	14.093	16.181	14.199
17	513.783	624.951	787.779	522.019	528.389	748.856	934.730	520.620	769.341	576.300	696.626	**452.844**	782.875	505.448	454.813	761.189	497.249
18	**3.299**	3.545	3.544	3.332	3.508	3.510	3.321	3.529	3.524	3.328	3.849	3.643	3.637	3.333	3.685	3.630	3.329
19	184.981	214.995	485.285	276.354	232.876	482.145	220.397	236.342	484.952	222.491	572.438	224.628	475.849	221.781	331.278	481.427	**147.496**
20	**281.295**	290.795	294.652	319.491	339.027	294.727	457.553	336.912	294.729	392.080	401.489	361.092	294.813	533.453	453.036	294.608	538.494
21	**4365.260**	9299.411	11031.837	4785.473	4706.695	11012.324	4623.988	4641.626	11027.663	6328.957	13139.765	5247.132	10958.876	5883.332	5857.633	11010.918	5197.504
22	**2668.133**	3620.203	5992.948	3151.963	2938.786	5468.076	4182.207	2876.970	5836.074	3410.714	3383.846	3334.374	5970.164	3276.142	3608.825	5475.435	3871.443
23	**2404.469**	2803.130	4103.788	3041.217	2936.098	3884.533	4040.816	2897.929	4045.992	3458.362	2591.799	2634.687	4092.315	3022.296	3615.719	3828.554	2725.427
24	773.192	1503.429	2522.163	805.136	692.238	2484.231	1133.932	720.655	2518.058	1098.135	1477.213	738.821	2437.216	536.834	**409.944**	2482.605	528.160
25	517.359	563.679	898.515	495.009	491.797	897.878	567.879	457.179	898.778	**442.624**	1427.145	497.587	882.090	480.055	617.633	896.393	544.052
26	**4532.324**	5201.048	5615.597	5093.202	5054.127	5616.358	7840.604	4923.926	5616.364	6413.364	8159.098	5038.920	5598.979	5110.505	5347.924	5616.318	6730.307
27	**8353.633**	16935.652	19415.905	9447.998	9095.797	19418.408	11083.042	8507.424	19418.513	11321.263	18624.148	9148.635	19233.200	9520.571	8838.947	19418.502	9632.940
28	8037.235	18147.963	20705.586	**7490.140**	7862.749	20708.878	8294.942	7581.883	20708.910	10406.826	19542.502	9213.144	20498.595	8481.631	9618.961	20708.802	9770.336
29	34165.153	84771.070	90424.115	35905.569	35738.537	90429.048	40675.749	34450.813	90429.081	39908.070	85845.681	41475.882	89699.547	**33014.515**	44643.593	90429.081	47636.222
30	9270.914	17575.833	19948.971	9852.848	9266.516	19951.917	9856.430	8914.313	19951.980	11492.614	21149.832	9600.740	19751.338	**8595.859**	9088.245	19951.976	10125.622
31	31055.636	81678.239	87791.444	32091.879	29910.698	87796.279	40549.351	**29496.887**	87796.969	42526.815	70150.743	36780.384	86775.281	31494.064	51168.573	87796.668	43639.648
32	24439.846	74075.974	80117.625	24499.656	25094.847	80122.881	31706.108	**23906.550**	80122.928	29594.870	73007.094	30586.911	79201.449	30725.369	34093.550	80122.923	32098.087
33	56412.046	80716.216	83242.122	59059.825	56761.451	83244.330	73515.610	**56348.008**	83244.330	69350.959	93406.277	62974.875	83027.967	56965.036	61678.010	83244.330	64549.094
34	**543.188**	585.286	718.003	659.152	551.943	718.407	1281.510	561.494	718.427	799.126	787.656	570.192	713.778	690.828	647.725	718.361	766.961
35	**26878.049**	74640.366	80152.391	33411.571	27920.972	80156.921	30478.475	27128.138	80156.952	35569.080	81039.290	32271.770	79483.554	29357.264	45057.124	80156.904	45959.565
36	26593.267	80667.146	86883.711	**25531.781**	27177.451	86888.724	36683.101	25953.975	86889.222	30600.082	71317.384	35393.388	86028.117	25850.074	48694.863	86887.843	34547.083
37	454.598	514.772	851.319	408.272	372.474	851.336	477.514	387.390	852.320	589.755	994.810	446.658	839.104	**363.963**	401.161	851.236	756.908
38	38424.582	76225.591	80252.131	38790.717	37489.484	80255.657	43783.609	**36427.507**	80255.658	45921.228	91316.173	45856.709	79964.434	37423.531	46301.807	80255.658	48143.508
39	2656.327	4237.321	5318.660	2671.787	2496.426	5320.511	3736.204	2409.810	5320.924	3442.354	6772.570	2800.384	5244.882	2680.610	2718.854	5319.633	**2041.861**
40	**6488.612**	17596.681	20730.948	8250.733	7263.239	20733.107	9740.839	6728.226	20734.791	8503.962	20026.829	7592.486	20541.030	8537.474	9165.199	20731.332	10678.024
41	51465.369	80903.458	83989.310	49606.120	50701.015	83992.026	58030.261	48555.128	83992.026	61022.383	96170.511	55617.562	83707.657	**45383.693**	54689.198	83992.026	70489.646
42	19486.690	81865.646	89218.010	21758.370	21537.513	89221.900	26262.331	19631.053	89224.582	27449.950	72831.039	31663.276	87998.163	**18962.016**	38087.736	89199.008	40349.158
43	3057.819	3963.204	5001.807	3386.240	2742.488	5002.113	3301.375	**2718.819**	5003.325	3555.466	5348.280	3234.277	4969.353	3360.137	3674.254	5001.795	3727.576
44	2412.897	4104.522	5068.414	2403.370	2288.768	5069.928	3783.136	2267.533	5070.124	2892.992	6187.982	2369.460	5024.563	2302.684	3255.649	5069.773	**1967.129**
45	3974.540	4995.126	5322.602	4202.566	4204.897	5323.142	5635.016	4092.195	5323.143	6407.556	8001.484	**3928.042**	5308.642	4558.153	5115.327	5323.141	5238.192
46	8310.453	17948.086	20724.368	10086.206	8297.016	20727.201	9677.926	8196.070	20727.887	9470.949	21788.947	7699.716	20528.169	**7251.037**	8850.200	20727.512	10869.193
47	2782.793	4473.802	5354.928	2619.417	2572.771	5356.463	6780.446	2636.071	5356.536	5184.807	6347.438	3226.930	5304.201	2886.925	3184.081	5356.417	**2357.094**
48	27172.855	78800.669	84826.230	31815.446	27038.924	84831.456	40546.650	25992.239	84831.513	27380.764	79474.896	32991.573	83935.109	**24611.480**	35467.544	84830.989	31908.817
49	**25400.897**	79509.907	85397.814	29463.291	26620.186	85402.765	36818.415	25798.810	85402.779	30049.500	84381.520	31066.552	84686.283	27282.327	35100.529	85402.661	44920.838
50	28101.406	77197.481	83199.014	31508.683	28742.354	83203.947	39304.174	27411.967	83204.167	36325.818	80849.577	33002.138	82328.568	**27117.405**	69119.891	83201.814	32976.241
51	39392.835	74662.843	78590.818	46870.076	41253.917	78594.372	50593.718	38705.749	78594.394	57254.031	94081.643	48326.055	78169.928	**36432.513**	48536.899	78594.339	58828.077
52	691.892	726.878	1041.397	617.986	572.357	1041.688	873.078	611.098	1042.131	636.529	1430.373	691.022	1036.461	**491.473**	522.304	1041.548	529.068
53	26613.174	78436.117	84737.578	31425.500	27568.222	84739.556	31264.889	**26539.361**	84743.231	34396.616	72193.295	35720.152	83880.625	26925.620	44540.000	84718.315	39124.725
54	2431.217	3854.519	4869.366	2604.365	2532.116	4870.908	3366.851	2396.938	4871.149	3170.509	5305.835	2298.217	4801.088	2232.033	**1945.036**	4869.044	2153.503
55	3081.454	5056.617	5822.269	**2679.115**	3047.580	5823.494	2769.862	2878.348	5823.522	3356.213	5876.209	3741.867	5797.453	2827.005	3347.361	5823.481	3726.768
56	11267.168	19688.504	22044.279	12923.363	11917.786	22047.600	12769.064	11539.166	22047.625	15213.136	27746.537	13117.956	21850.968	**9751.451**	12379.782	22047.614	16108.140
57	**8395.034**	16828.341	19533.020	9798.498	9096.500	19534.521	10455.107	9051.140	19536.437	9726.998	23809.160	9030.759	19276.399	12330.782	14431.607	19531.972	11007.128
58	34382.525	80844.378	86306.244	34886.233	34717.676	86310.949	39927.099	**33431.696**	86310.970	39022.995	79402.754	40121.219	85460.024	38106.878	47091.700	86310.968	45871.479
59	30344.980	75280.391	80556.991	34991.674	31904.706	80561.482	36916.271	30139.329	80561.561	41030.591	84508.794	34897.989	79819.819	**28786.161**	47406.192	80561.513	37495.171
60	895.771	931.947	1063.866	923.898	812.233	1064.331	927.293	**799.628**	1064.380	1019.797	1816.484	886.323	1060.282	911.805	1168.796	1064.161	989.632
61	38776.271	84490.987	90318.540	43746.271	40252.156	90323.489	44120.612	39504.645	90323.506	44822.076	98963.370	39597.245	89558.692	40427.628	52543.018	90323.506	**36852.841**
62	25339.999	74233.032	80000.784	26247.528	25626.608	80005.669	35452.670	**24415.275**	80005.849	39533.121	77153.471	31994.327	79066.671	28626.641	56776.364	80001.714	37982.692
63	846.266	823.074	898.659	760.373	759.256	898.775	867.216	**741.389**	898.901	774.130	1251.892	775.035	895.722	745.426	800.976	898.786	953.798
64	7967.262	15496.561	17631.796	7642.699	7932.310	17634.344	13485.525	**7561.213**	17634.511	9812.051	22816.330	8719.436	17434.156	7642.873	10250.976	17634.516	9291.753
65	7159.218	20144.242	23164.403	6829.276	7941.748	23168.026	8930.831	7484.446	23168.156	8924.958	21081.809	9380.177	22925.769	**6209.727**	8341.442	23167.746	9452.026
66	25247.196	71035.940	77399.702	31590.254	24841.485	77405.188	34207.789	**24608.331**	77405.708	33700.539	86193.219	30274.333	76266.693	31306.264	74727.084	77374.947	33402.963
67	7451.076	15925.159	18543.356	7629.509	6854.926	18546.494	10270.987	**6672.107**	18546.794	13030.274	19717.447	8921.541	18218.564	7911.706	11926.973	18545.115	9607.628
68	15431.149	21558.495	22810.123	15583.826	15553.972	22811.839	20612.525	**15081.704**	22811.840	17770.447	25730.948	16844.606	22729.640	15144.864	16574.222	22811.839	16009.581
69	714.301	**645.684**	905.810	700.204	708.681	906.586	958.732	694.679	906.633	816.648	1386.041	729.988	897.268	778.044	906.464	906.588	906.319
70	**3666.587**	4646.279	5166.721	4228.268	3773.103	5167.689	6474.670	3761.821	5167.697	4648.360	6783.517	3835.751	5143.204	4632.977	4805.847	5167.666	4780.328
71	14348.467	20924.895	22712.932	14475.781	13803.079	22715.171	14557.806	**13601.893**	22715.182	15249.652	24333.556	15198.388	22559.230	15774.443	15720.357	22714.551	17195.998
72	11461.190	20023.773	22672.817	12795.607	11111.285	22675.027	12749.803	11156.334	22675.837	12939.175	23498.689	11396.003	22423.818	**10824.107**	14067.130	22675.688	11349.510
73	8270.418	17510.467	19950.475	**6985.051**	8213.203	19953.374	10974.296	7693.922	19953.474	10106.323	21423.627	9431.575	19771.129	9845.550	10589.138	19953.443	9566.146
74	2361.548	4760.308	5914.062	**2098.991**	2547.322	5916.099	4267.070	2334.638	5916.149	3441.698	5335.400	2893.785	5856.423	2138.288	2749.626	5916.143	3253.815
75	11358.682	16933.829	18901.688	12378.397	11963.167	18904.003	12983.933	11530.707	18904.147	13418.056	24262.484	10585.968	18750.922	**10548.136**	14292.806	18904.075	13103.517
76	7932.827	19618.344	22432.663	7604.646	7598.489	22435.877	9817.878	7206.447	22436.163	9009.190	22605.132	9246.777	22219.285	**7132.289**	9523.943	22435.975	10101.549
77	2567.306	4319.308	5446.224	2868.751	2598.766	5447.096	3921.426	**2524.026**	5448.054	2979.803	6826.978	2861.094	5396.815	2795.131	2987.712	5447.788	3374.934
78	3390.191	5027.911	5964.496	3265.811	3222.305	5965.918	6469.570	3102.425	5965.952	2984.491	6379.442	3155.312	5924.024	2850.951	3061.307	5965.932	**2452.763**
79	13724.975	20284.455	22024.449	14964.828	13855.838	22026.487	17229.068	**13437.865**	22026.515	17014.194	22848.356	16279.380	21929.361	15545.182	15425.809	22026.395	18271.642
80	17098.609	20183.309	21097.025	17387.715	16847.903	21098.192	22415.495	16570.308	21098.192	22065.275	28089.810	**15900.526**	21028.871	16394.803	19236.840	21098.192	21721.685
81	**624.890**	669.774	770.174	647.863	673.913	770.348	1445.722	646.382	770.353	969.302	1023.066	688.835	770.456	786.105	894.220	770.090	956.968
82	4687.273	**4400.364**	4593.666	5066.261	4486.512	4594.062	8357.198	4612.205	4594.063	8600.552	7225.092	4528.818	4585.858	4926.538	5168.313	4594.061	6893.659
83	14208.282	20899.867	22664.270	13644.693	12960.361	22666.847	16240.511	**12916.093**	22666.863	16022.964	29884.188	14243.606	22513.409	13261.193	15323.439	22666.847	15154.884
84	403.815	567.497	863.515	484.277	**393.584**	864.335	820.481	404.243	864.463	540.723	1092.453	464.181	854.370	591.100	648.914	864.163	615.650
85	1367.429	3903.278	5367.733	1457.697	1466.173	5369.403	3419.805	1336.166	5369.825	3177.714	5937.911	1725.257	5275.921	**1156.495**	1557.982	5368.571	1905.655
86	3597.284	5268.283	6153.624	**3133.893**	3145.868	6155.200	3716.425	3200.470	6155.245	4023.096	7652.896	4083.480	6105.355	3724.594	4095.718	6154.982	3559.820
87	96.367	85.712	181.836	93.960	76.051	181.757	**36.119**	84.166	182.136	60.620	301.704	118.775	179.084	87.027	123.736	181.567	82.649
88	392.384	**323.953**	694.063	446.950	409.525	688.926	436.455	397.510	694.104	402.884	903.366	379.638	678.798	449.301	597.180	679.782	402.629
89	20.558	23.167	27.107	28.187	27.141	23.422	37.467	24.913	26.219	30.539	31.234	**19.042**	26.566	28.278	29.258	24.794	26.065
90	11520.658	**10620.869**	10922.302	13334.702	11527.814	10921.513	12764.969	11678.670	10922.114	14567.161	16189.345	11547.938	10910.140	14355.266	15037.958	10919.027	13322.041
91	**95.561**	105.175	127.722	102.864	103.568	126.168	105.407	101.752	126.925	109.763	127.909	113.332	124.956	130.289	100.164	123.401	123.265
92	191.113	196.099	199.411	252.975	226.603	199.271	335.091	219.762	199.422	350.847	313.668	184.010	199.946	191.097	237.220	199.729	**162.545**
93	184.880	1788.591	3277.225	179.494	171.862	3231.431	301.056	**155.885**	3259.320	224.061	2430.488	161.774	3144.334	220.880	197.514	3174.379	193.441
94	3336.131	3573.277	3970.641	3839.050	3520.955	3971.425	6808.954	3458.396	3971.430	4590.224	5024.054	**2936.933**	3954.595	3735.755	3415.889	3971.367	4010.824
95	518.239	**450.280**	463.171	625.218	588.039	463.249	1205.872	588.744	463.249	673.220	671.801	542.662	463.597	844.415	788.214	463.265	921.686
96	45.535	56.707	180.850	53.527	62.000	160.652	58.860	60.241	165.892	51.519	155.083	**41.657**	137.677	62.224	52.587	149.031	54.941

Tabela A.12: Resultados em *datasets* de regressão - out of sample - 100 árvores

#	Average
XG:BL:SVM-L:M	83.728
RF:BL:SVM-L	83.707
RF:BL:SVM-L:M	83.647
XG:BL:SVM-L	83.544
RF	83.129
XG	82.973

Tabela A.13: Média na classificação - out of sample - 500 árvores

#	No. of winners
XG:BL:SVM-L	15
XG	14
XG:BL:SVM-L:M	14
RF:BL:SVM-L	13
RF:BL:SVM-L:M	12
RF	11

Tabela A.14: Campeões na classificação - out of sample - 500 árvores

#	RF	RF:BL:SVM-L	XG	XG:BL:SVM-L	RF:BL:SVM-L:M	XG:BL:SVM-L:M
1	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
2	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**	**1.000**
3	0.822	0.783	**0.917**	0.885	0.796	0.866
4	0.690	0.695	**0.727**	0.679	0.695	0.679
5	0.988	0.988	0.988	**0.994**	0.988	**0.994**
6	**0.979**	0.951	0.972	**0.979**	0.951	**0.979**
7	0.720	0.640	**0.740**	0.680	0.640	0.680
8	**0.750**	0.722	0.722	0.736	0.722	0.736
9	0.981	0.988	0.972	**0.991**	0.988	0.988
10	**0.996**	**0.996**	0.982	0.994	**0.996**	0.994
11	0.948	0.948	0.948	**0.956**	0.948	**0.956**
12	**0.963**	**0.963**	**0.963**	0.945	**0.963**	0.945
13	0.885	0.885	0.904	**0.962**	0.885	**0.962**
14	**0.838**	0.832	**0.838**	0.803	0.832	0.803
15	0.800	**0.822**	0.815	**0.822**	**0.822**	**0.822**
16	0.935	0.935	**0.978**	0.957	0.935	**0.978**
17	0.911	0.919	0.863	**0.960**	0.923	0.956
18	0.509	0.512	**0.553**	0.455	0.512	0.493
19	0.273	**0.364**	0.303	0.333	0.333	**0.364**
20	**0.960**	**0.960**	0.920	0.880	**0.960**	0.880
21	0.610	0.597	**0.649**	0.610	0.597	0.610
22	**0.818**	**0.818**	0.758	**0.818**	**0.818**	**0.818**
23	0.526	0.539	0.539	0.513	**0.553**	0.526
24	0.887	**0.906**	0.849	**0.906**	0.887	0.887
25	0.658	0.685	0.678	**0.692**	0.685	**0.692**
26	0.744	0.769	**0.821**	**0.821**	0.769	**0.821**
27	0.932	0.943	**0.966**	0.943	0.943	0.943
28	**0.974**	**0.974**	**0.974**	0.921	**0.974**	0.921
29	0.801	0.788	**0.817**	0.755	0.788	0.755
30	0.935	**1.000**	0.806	**1.000**	**1.000**	**1.000**
31	0.767	**0.860**	0.651	0.744	**0.860**	0.744
32	**1.000**	**1.000**	0.935	**1.000**	**1.000**	**1.000**

Tabela A.15: Resultados em *datasets* de classificação - out of sample - 500 árvores

#	Average
RF:PV:SVM-L	8241.258
XG	8274.155
XG:BL:SVM-L	8386.829
RF	8511.247
RF:BL:SVM-L	14921.838

Tabela A.16: Média na regressão - out of sample - 500 árvores

#	No. of winners
XG	16
RF:PV:SVM-L	13
RF	8
XG:BL:SVM-L	8
RF:BL:SVM-L	3

Tabela A.17: Campeões na regressão - out of sample - 500 árvores

#					XG:BL:SVM-L
1	0.260	0.329	0.317	**0.249**	0.340
2	0.424	0.488	0.473	**0.415**	0.445
3	0.413	**0.403**	0.418	0.406	0.405
4	2.657	2.706	2.702	**2.647**	2.690
5	41.144	39.072	45.540	40.993	**30.978**
6	**33.866**	41.059	41.317	38.220	43.460
7	10.655	17.810	17.303	**10.021**	15.003
8	**1061.714**	1228.726	1262.278	1219.519	1544.453
9	**1059.652**	1228.697	1262.323	1220.635	1552.971
10	**90.069**	101.306	108.241	94.444	95.894
11	185.316	228.053	222.654	**146.536**	176.536
12	907.108	919.072	**858.342**	1093.842	995.374
13	**790.627**	885.612	866.075	818.931	929.211
14	**208.501**	223.961	225.177	248.007	289.136
15	984.363	**939.832**	976.933	1098.591	968.685
16	13.797	14.096	14.419	**13.767**	13.923
17	499.222	485.089	487.913	**434.956**	438.114
18	3.271	3.529	3.512	**3.268**	3.649
19	**186.709**	219.230	238.151	230.751	234.147
20	**299.248**	302.154	317.726	364.054	392.794
21	4387.032	5617.175	4577.657	4410.146	**4167.657**
22	2694.193	2870.325	2928.530	**2490.458**	2651.706
23	2452.790	2491.106	2963.621	**2245.754**	2487.053
24	781.397	744.652	729.663	**572.284**	669.546
25	517.117	442.606	**417.912**	529.077	470.969
26	4526.620	4740.065	4763.860	4800.550	**4456.372**
27	7941.046	11072.195	**7611.808**	7643.059	7915.326
28	8229.611	11921.002	**8044.483**	8070.641	8305.666
29	34888.723	66181.934	**34398.102**	36895.322	36642.884
30	8899.974	11788.427	8591.879	**8381.092**	8498.011
31	30165.594	61971.281	**27842.607**	28672.013	30584.821
32	24134.727	54648.323	23116.539	23288.317	**22919.300**
33	55985.830	72438.023	**54943.640**	56141.001	57009.422
34	531.014	566.550	**530.536**	576.152	576.819
35	26493.693	56497.706	26629.105	**25941.514**	26791.883
36	26929.022	60312.660	26090.015	**24348.347**	26235.562
37	428.602	404.068	**391.752**	417.584	412.834
38	38084.491	63029.379	**36482.902**	38231.910	38308.450
39	2739.969	2541.549	**2401.208**	2672.098	2541.869
40	6483.348	9448.796	6621.425	6555.247	**6275.083**
41	49502.266	70288.508	46256.286	47272.006	**45882.216**
42	19035.163	57409.978	18304.769	**17995.441**	18572.274
43	2971.319	**2647.927**	2675.338	2806.820	2726.772
44	2326.385	2336.446	**2183.285**	2473.734	2412.742
45	3907.868	4090.860	3892.934	3866.654	**3718.463**
46	8151.133	10564.422	7817.916	7562.741	**7132.177**
47	2563.053	2894.293	**2434.736**	2661.181	2606.767
48	26408.837	59406.758	24986.076	**22558.028**	23866.918

Tabela A.18: Resultados em *datasets* de regressão - out of sample - 500 árvores

B
Aspectos estruturais das árvores de decisão

As próximas figuras representam a estrutura das árvores em cada exemplo. Para muitas dessas, o algoritmo de *random forest* gera as árvores com o mesmo número de folhas em cada instância. Por outro lado, o *XGBoost* demonstra uma distribuição mais homogênea entre os números de folhas de cada árvore de decisão.

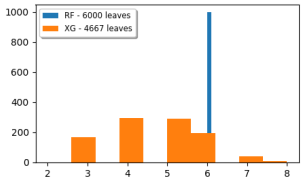

Figura B.1: Connectionist-bench-sonar

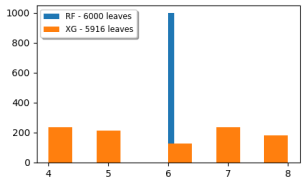

Figura B.2: Chess-king-rook-vs-king-pawn

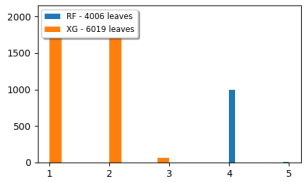

Figura B.3: Soybean-small

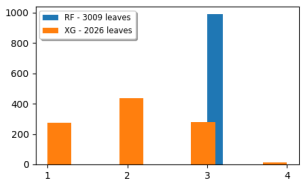

Figura B.4: Acute-inflammations-2

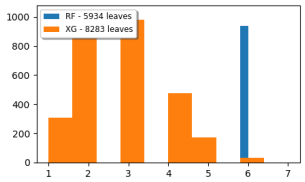

Figura B.5: Wine

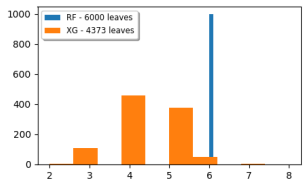

Figura B.6: Monks-problems-1

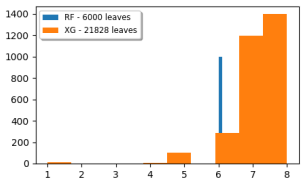

Figura B.7: Balance-scale

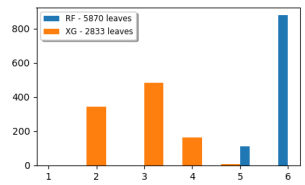

Figura B.8: Spectf-heart

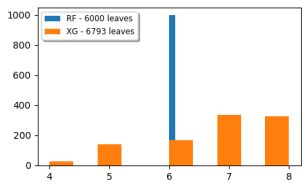

Figura B.9: Spambase

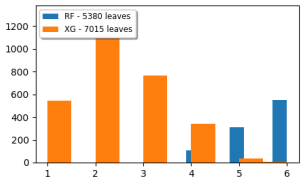

Figura B.10: Iris

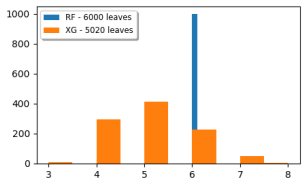

Figura B.11: Breast-cancer-wisconsin-prognostic

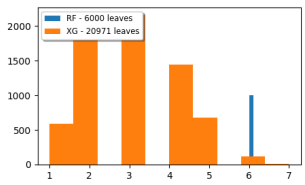

Figura B.12: Image-segmentation

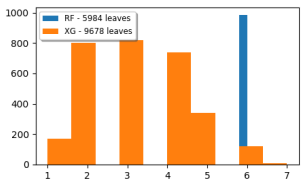

Figura B.13: Seeds

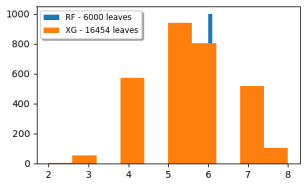

Figura B.14: Echocardio-gram

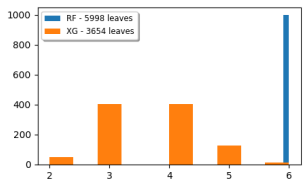

Figura B.15: Hepatitis

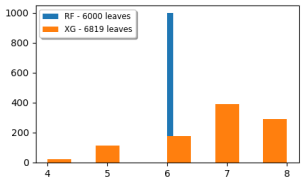

Figura B.16: Blood-transfusion-service-center

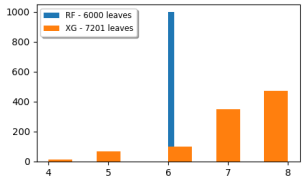

Figura B.17: Seismic-bumps

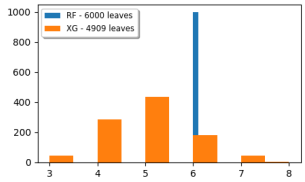

Figura B.18: Ionosphere

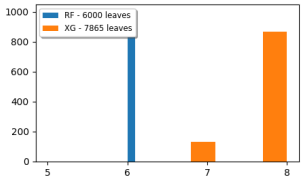

Figura B.19: Tic-tac-toe-endgame

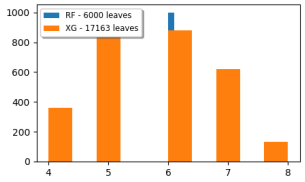

Figura B.20: Teaching-assistant-evaluation

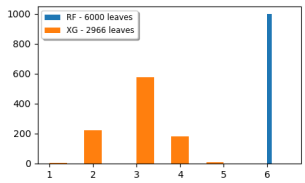

Figura B.21: Spect-heart

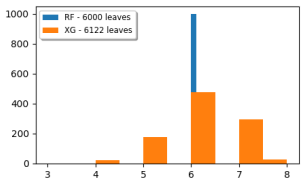

Figura B.22: Monks-problems-2

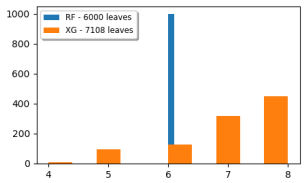

Figura B.23: Ozone-level-detection-eight

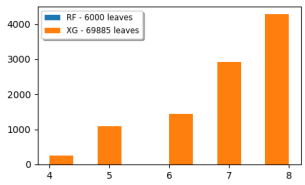

Figura B.24: Optical-recognition-handwritten-digits

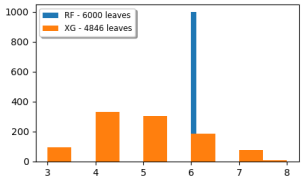

Figura B.25: Climate-model-simulation-crashes

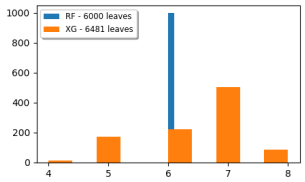

Figura B.26: Banknote-authentication

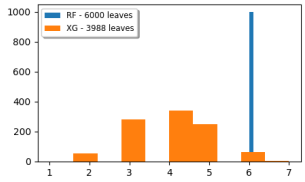

Figura B.27: Congressional-voting-records

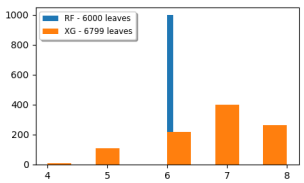

Figura B.28: Credit-approval

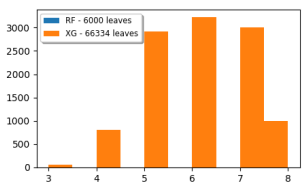

Figura B.29: Connectionist-bench

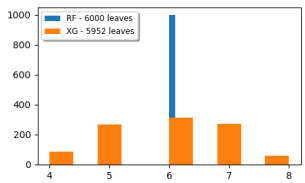

Figura B.30: Haberman-survival

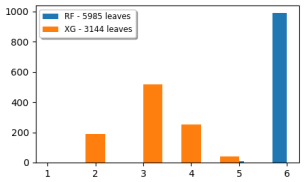

Figura B.31: Monks-problems-3

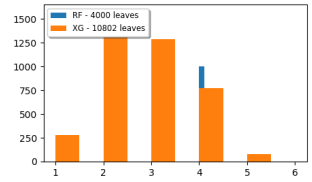

Figura B.32: Wall-following-robot-navigation-2

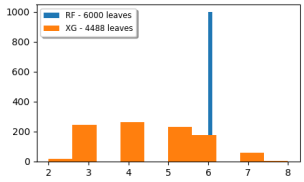

Figura B.33: Breast-cancer-wisconsin-diagnostic

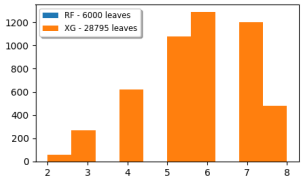

Figura B.34: Heart-disease-Cleveland

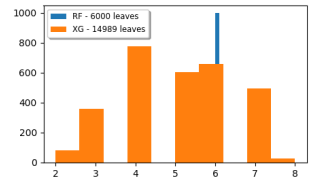

Figura B.35: Hayes-roth

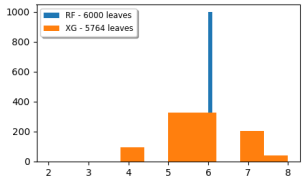

Figura B.36: Thoracic-surgery

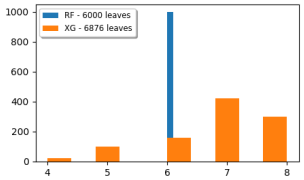

Figura B.37: Mammographic-mass

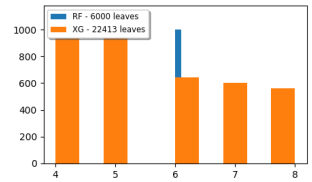

Figura B.38: Car-evaluation

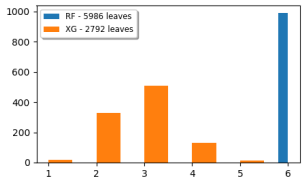

Figura B.39: Fertility

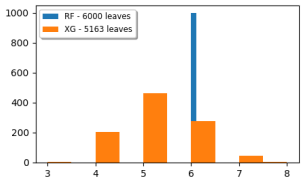

Figura B.40: Planning-relax

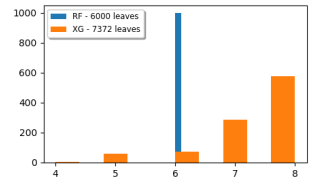

Figura B.41: Statlog-project-German-credit

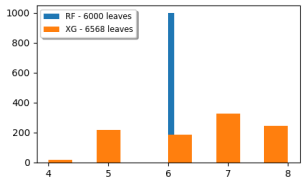

Figura B.42: Cylinder-bands

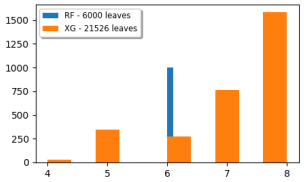

Figura B.43: Contraceptive-method-choice

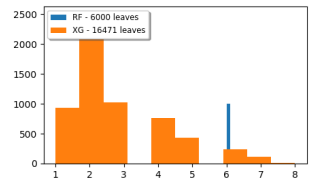

Figura B.44: Dermato-logy

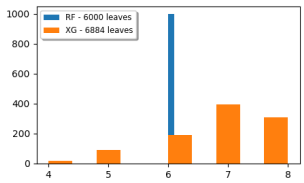

Figura B.45: Indian-liver-patient

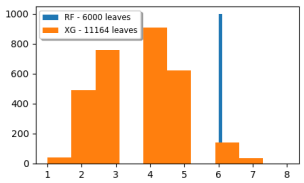

Figura B.46: Thyroid-disease-new-thyroid

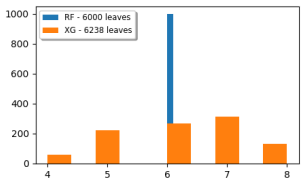

Figura B.47: Ozone-level-detection-one

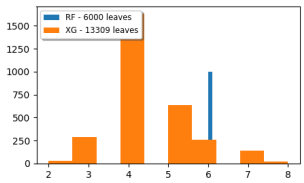

Figura B.48: Thyroid-disease-ann-thyroid

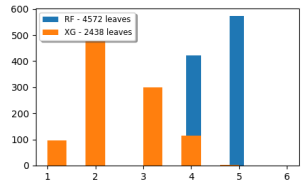

Figura B.49: Acute-inflammations-1

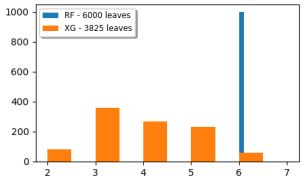

Figura B.50: Parkinsons

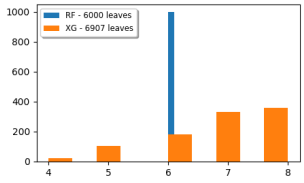

Figura B.51: Qsar-biodegradation

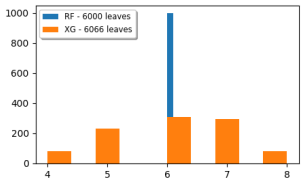

Figura B.52: Breast-cancer

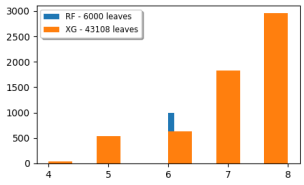

Figura B.53: Statlog-project-landsat-satellite

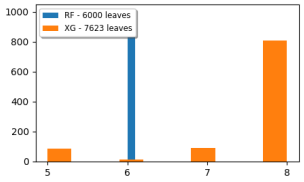

Figura B.54: 593-fri-c1-1000-10

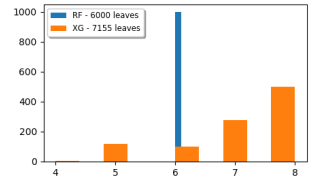

Figura B.55: 695-chatfield-4

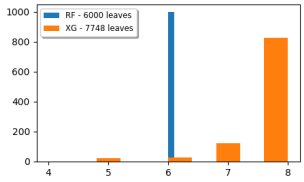

Figura B.56: 637-fri-c1-500-50

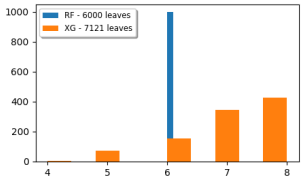

Figura B.57: 621-fri-c0-100-10

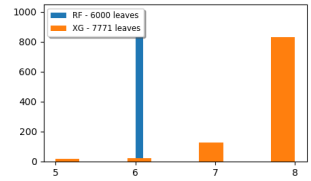

Figura B.58: 649-fri-c0-500-5

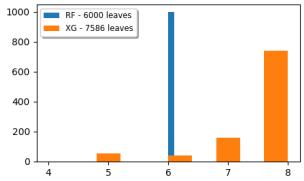

Figura B.59: 628-fri-c3-1000-5

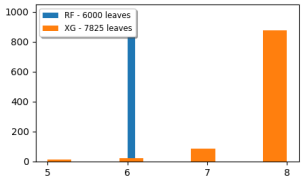

Figura B.60: 590-fri-c0-1000-50

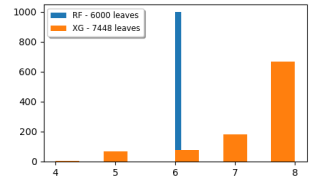

Figura B.61: 584-fri-c4-500-25

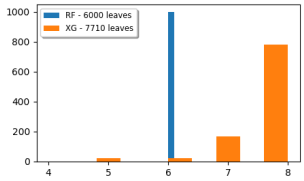

Figura B.62: 604-fri-c4-500-10

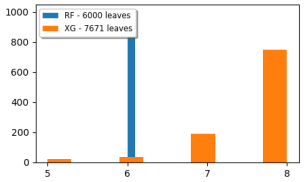

Figura B.63: 647-fri-c1-250-10

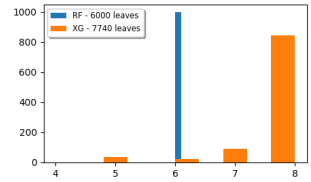

Figura B.64: 589-fri-c2-1000-25

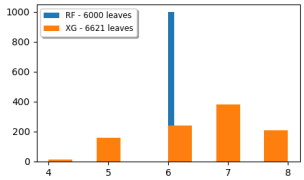

Figura B.65: 561-cpu

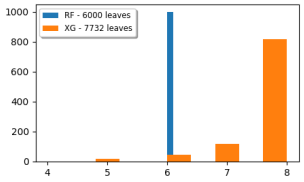

Figura B.66: 650-fri-c0-500-50

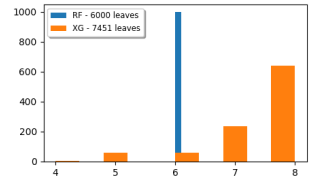

Figura B.67: 579-fri-c0-250-5

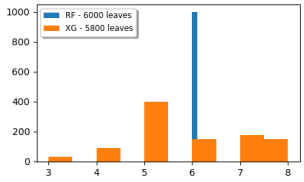

Figura B.68: 195-auto-price

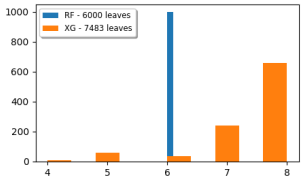

Figura B.69: 601-fri-c1-250-5

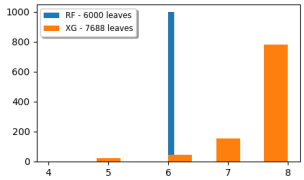

Figura B.70: 641-fri-c1-500-10

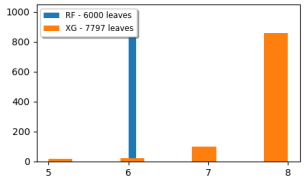

Figura B.71: 618-fri-c3-1000-50

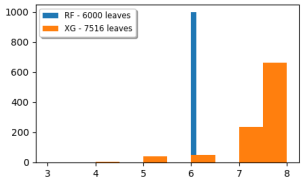

Figura B.72: 522-pm10

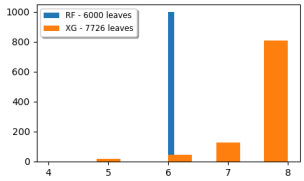

Figura B.73: 612-fri-c1-1000-5

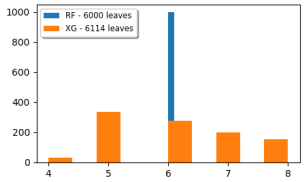

Figura B.74: 210-cloud

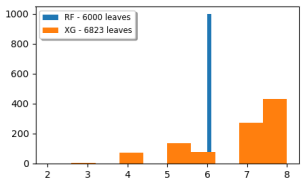

Figura B.75: 665-sleuth-case2002

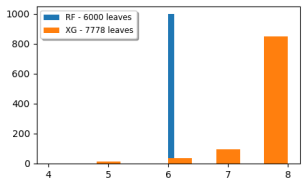

Figura B.76: 622-fri-c2-1000-50

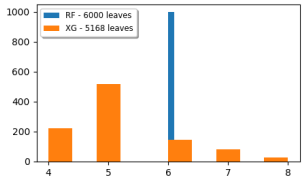

Figura B.77: 560-bodyfat

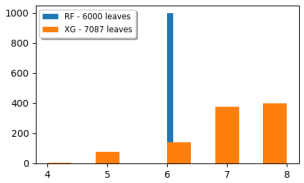

Figura B.78: 594-fri-c2-100-5

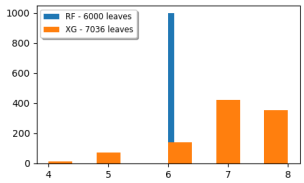

Figura B.79: 663-rabe-266

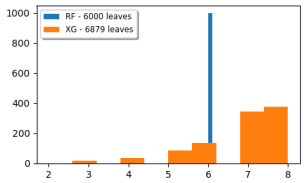

Figura B.80: 485-analcatdata-vehicle

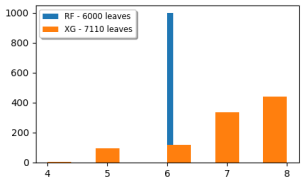

Figura B.81: 613-fri-c3-250-5

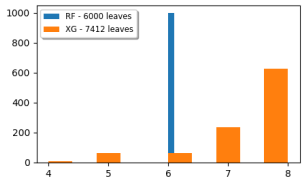

Figura B.82: 648-fri-c1-250-50

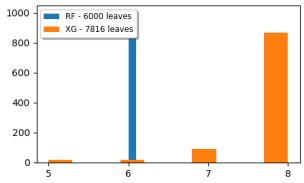

Figura B.83: 595-fri-c0-1000-10

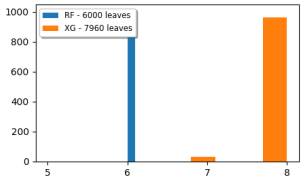

Figura B.84: 557-analcatdata-apnea1

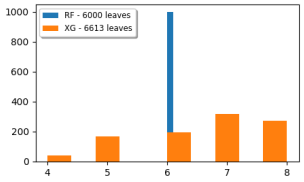

Figura B.85: 634-fri-c2-100-10

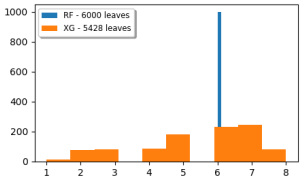

Figura B.86: 192-vineyard

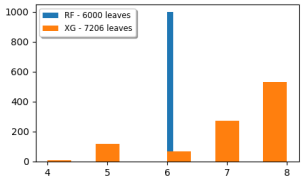

Figura B.87: 617-fri-c3-500-5

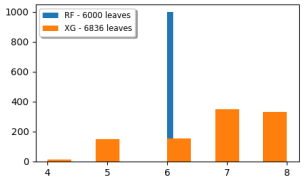

Figura B.88: 591-fri-c1-100-10

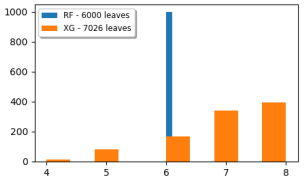

Figura B.89: 651-fri-c0-100-25

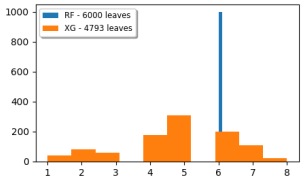

Figura B.90: 1089-USCrime

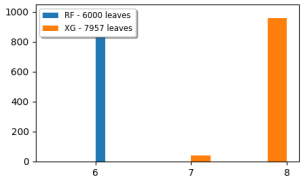

Figura B.91: 229-pwLinear

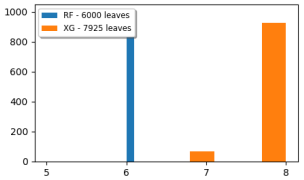

Figura B.92: 1029-LEV

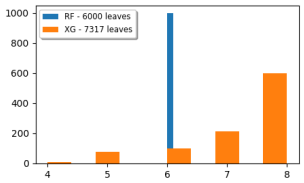

Figura B.93: 645-fri-c3-500-50

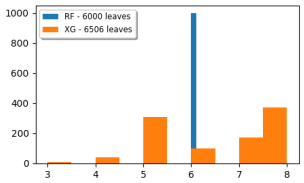

Figura B.94: 666-rmftsa-ladata

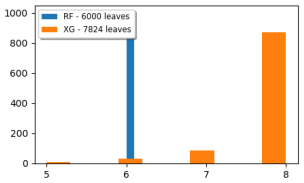

Figura B.95: 620-fri-c1-1000-25

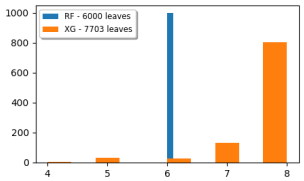

Figura B.96: 633-fri-c0-500-25

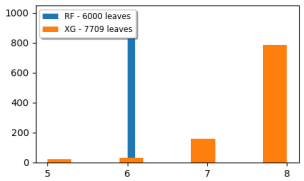

Figura B.97: 643-fri-c2-500-25

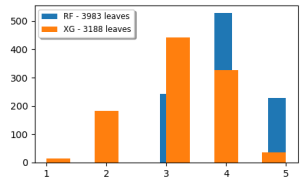

Figura B.98: 523-analcatdata-neavote

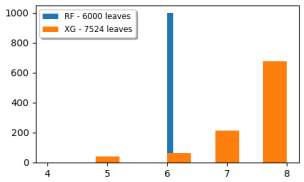

Figura B.99: 603-fri-c0-250-50

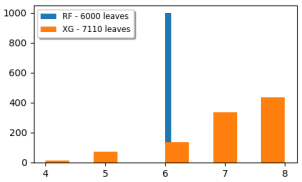

Figura B.100: 656-fri-c1-100-5

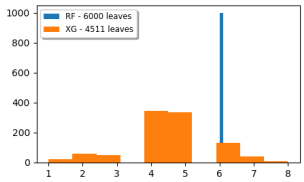

Figura B.101: 659-sleuth-ex1714

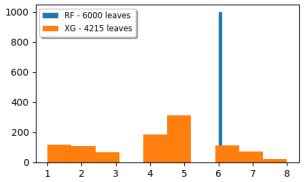

Figura B.102: 1096-FacultySalaries

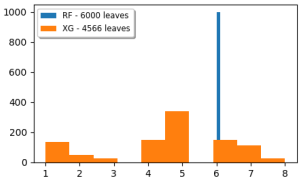

Figura B.103: 542-pollution

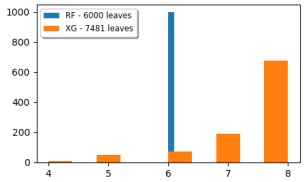

Figura B.104: 616-fri-c4-500-50

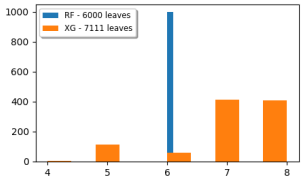

Figura B.105: 230-machine-cpu

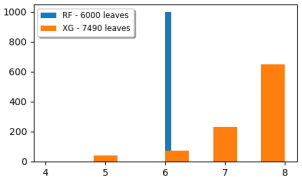

Figura B.106: 635-fri-c0-250-10

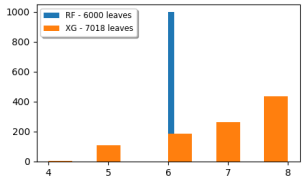

Figura B.107: 658-fri-c3-250-25

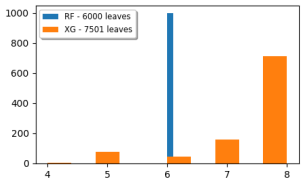

Figura B.108: 608-fri-c3-1000-10

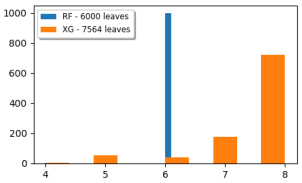

Figura B.109: 623-fri-c4-1000-10

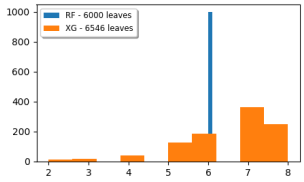

Figura B.110: 678-visualizing-environmental

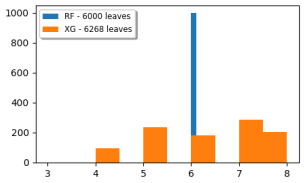

Figura B.111: 1027-ESL

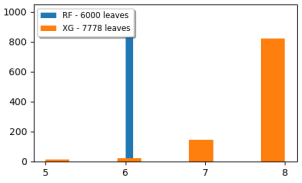

Figura B.112: 586-fri-c3-1000-25

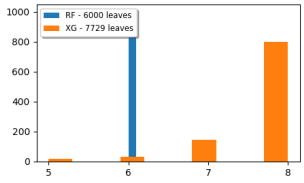

Figura B.113: 599-fri-c2-1000-5

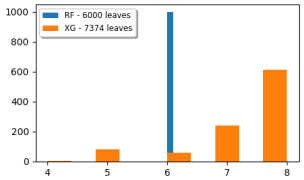

Figura B.114: 657-fri-c2-250-10

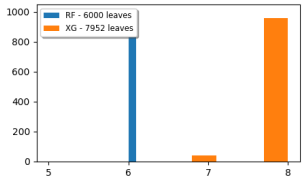

Figura B.115: 556-analcatdata-apnea2

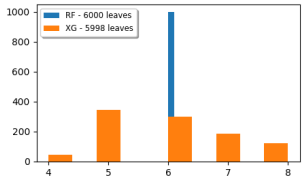

Figura B.116: 505-tecator

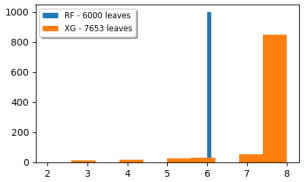

Figura B.117: 1028-SWD

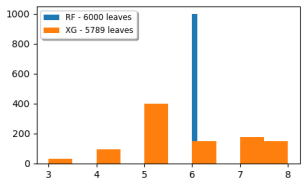

Figura B.118: 207-autoPrice

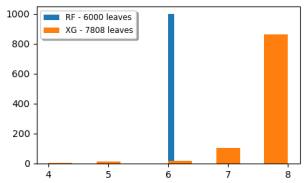

Figura B.119: 609-fri-c0-1000-5

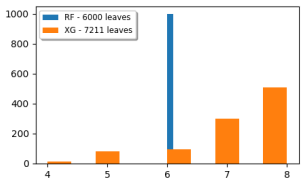

Figura B.120: 602-fri-c3-250-10

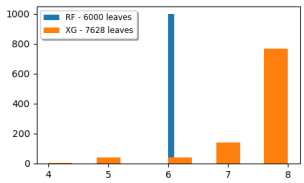

Figura B.121: 607-fri-c4-1000-50

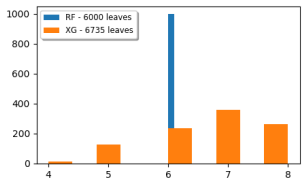

Figura B.122: 611-fri-c3-100-5

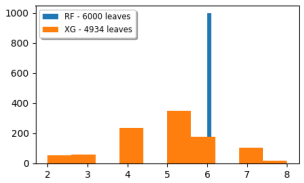

Figura B.123: 228-elusage

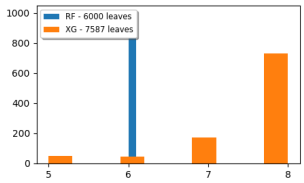

Figura B.124: 626-fri-c2-500-50

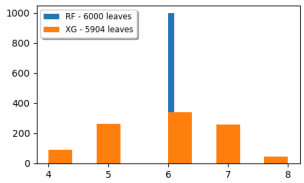

Figura B.125: 527-analcatdata-election2000

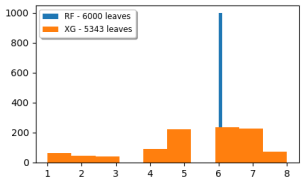

Figura B.126: 687-sleuth-ex1605

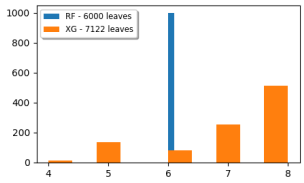

Figura B.127: 596-fri-c2-250-5

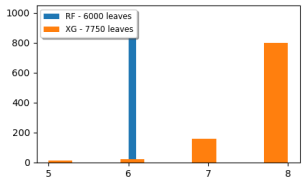

Figura B.128: 597-fri-c2-500-5

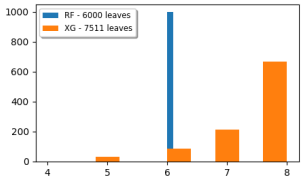

Figura B.129: 653-fri-c0-250-25

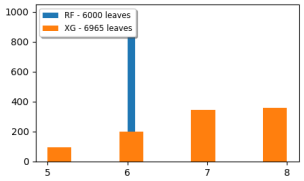

Figura B.130: 1030-ERA

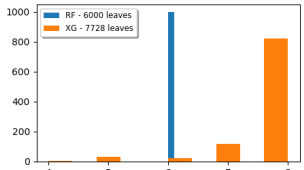

Figura B.131: 606-fri-c2-1000-10

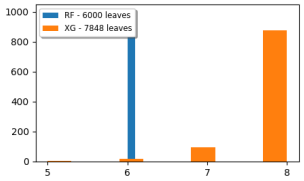

Figura B.132: 598-fri-c0-1000-25

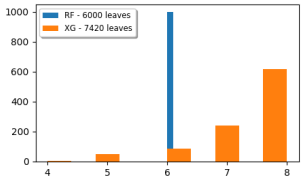

Figura B.133: 646-fri-c3-500-10

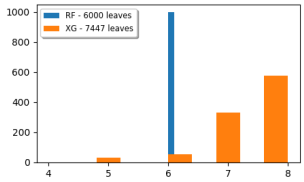

Figura B.134: 519-vinnie

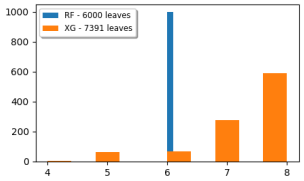

Figura B.135: 615-fri-c4-250-10

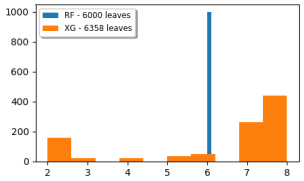

Figura B.136: 712-chscase-geyser1

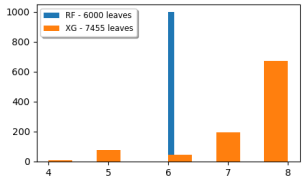

Figura B.137: 605-fri-c2-250-25

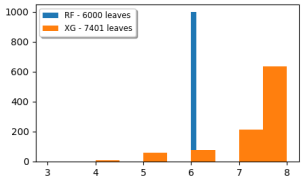

Figura B.138: 547-no2

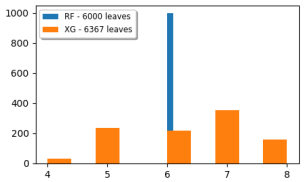

Figura B.139: 706-sleuth-case1202

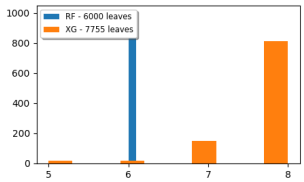

Figura B.140: 582-fri-c1-500-25

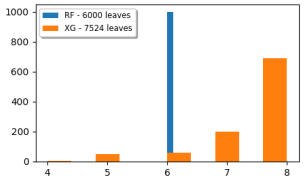

Figura B.141: 581-fri-c3-500-25

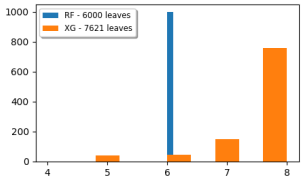

Figura B.142: 627-fri-c2-500-10

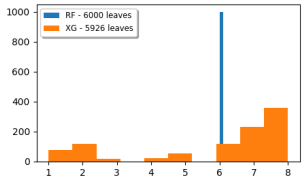

Figura B.143: 690-visualizing-galaxy

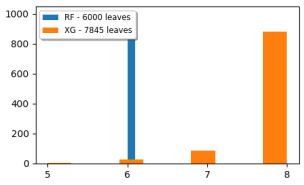

Figura B.144: 583-fri-c1-1000-50

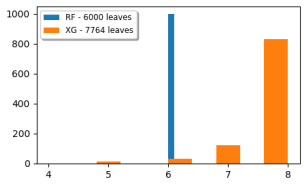

Figura B.145: 592-fri-c4-1000-25

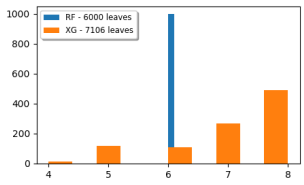

Figura B.146: 644-fri-c4-250-25

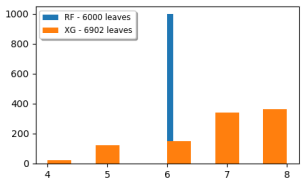

Figura B.147: 624-fri-c0-100-5

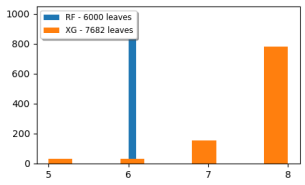

Figura B.148: 654-fri-c0-500-10

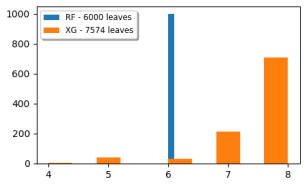

Figura B.149: 631-fri-c1-500-5

C
Visualizações gráficas de y e f(x) nos modelos

Finalmente, apresentam-se algumas visualizações gráficas, da posição dos valores de y e os valores obtidos nos algoritmos, projetados em um espaço de duas dimensões. Cabe ressaltar, essa interpretação só é possível para os valores reais, em tarefas de regressão. Observamos, empiricamente, que os gráficos demonstram uma proximidade entre os valores estimados pelo modelo $2PL$ e os valores reais de y, especialmente na fase de treinamento, sugerindo adequação ao *overfitting*.

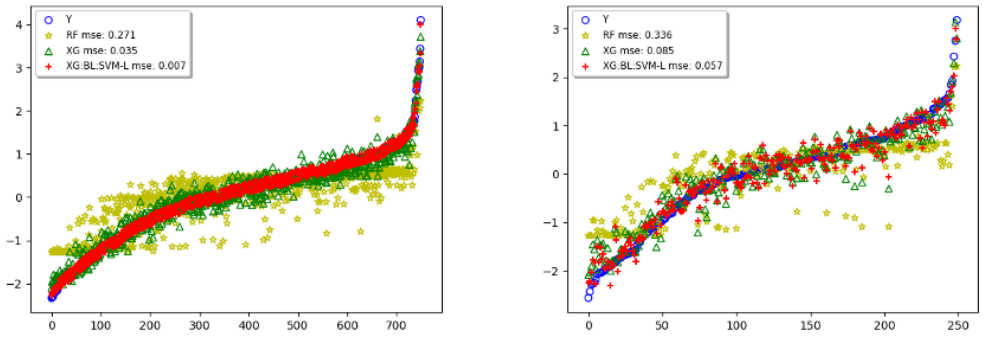

Figura C.1: In Sample x Out of sample - 623-fri-c4-1000-10

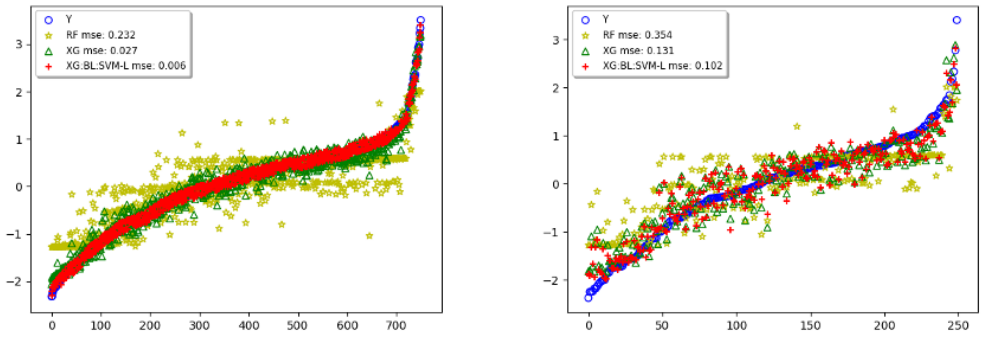

Figura C.2: In Sample x Out of sample - 618-fri-c3-1000-50

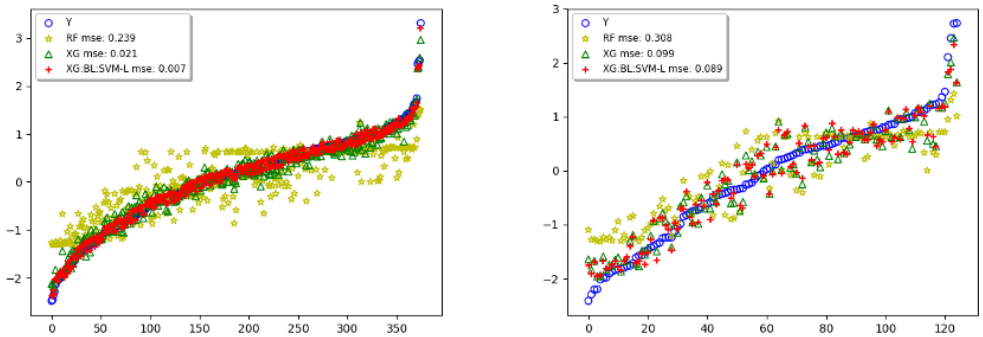

Figura C.3: In Sample x Out of sample - 584-fri-c4-500-25

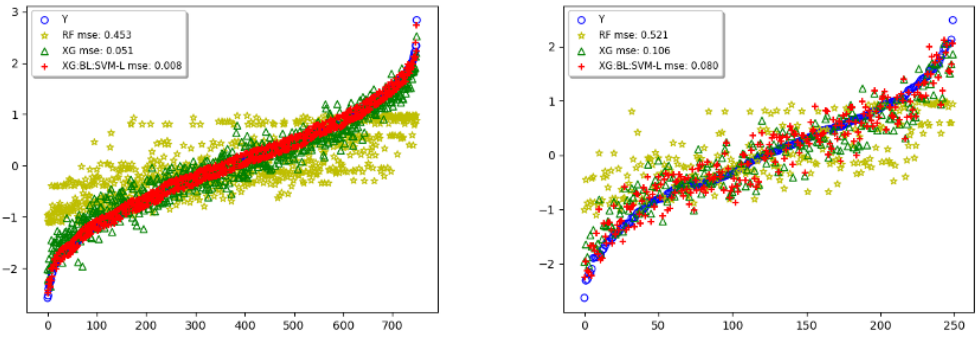

Figura C.4: In Sample x Out of sample - 595-fri-c0-1000-10

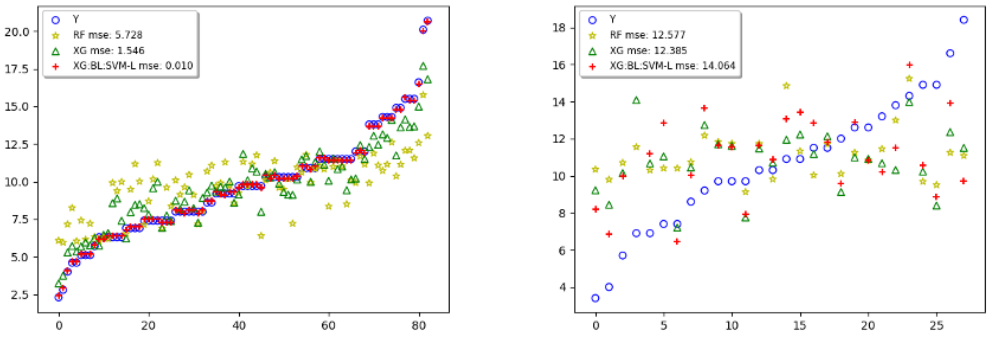

Figura C.5: In Sample x Out of sample - 678-visualizing-environmental

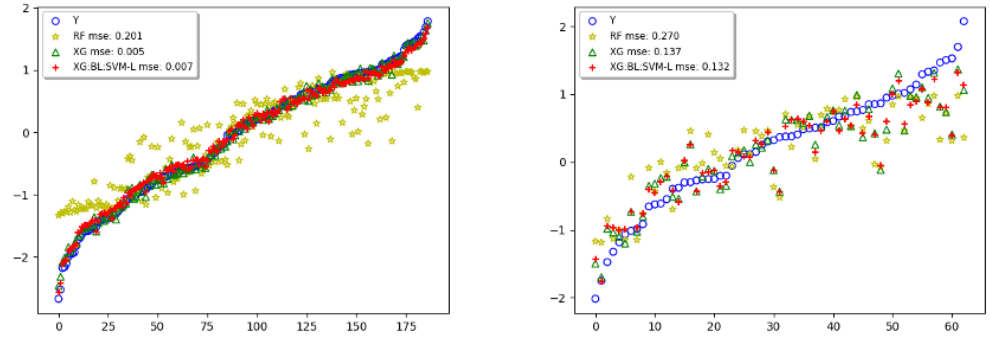

Figura C.6: In Sample x Out of sample - 648-fri-c1-250-50

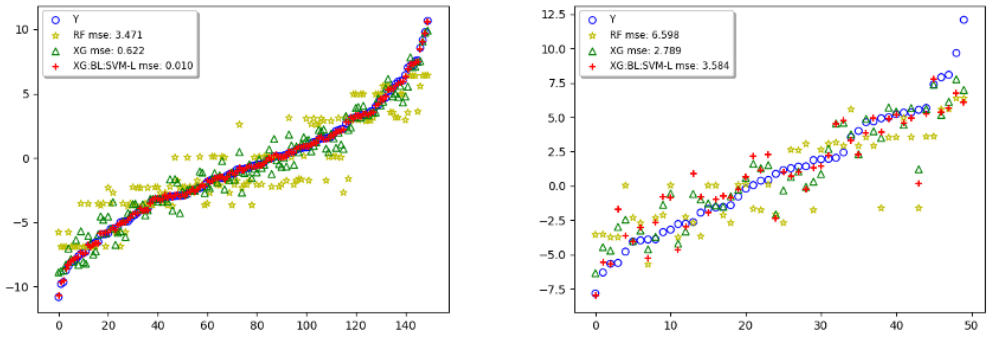

Figura C.7: In Sample x Out of sample - 229-pwLinear

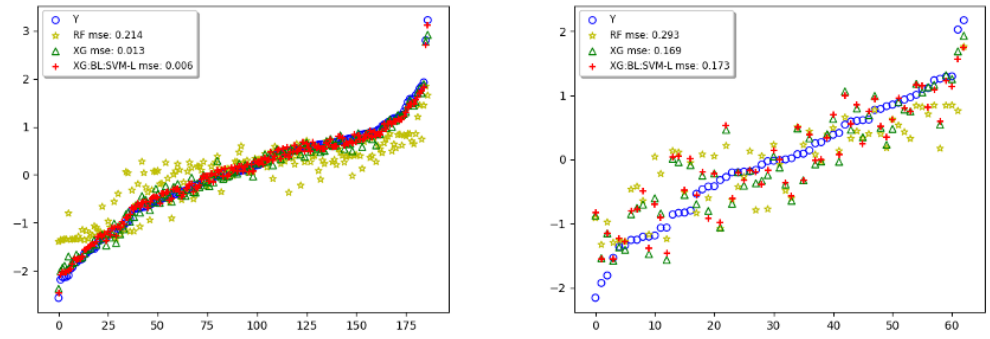

Figura C.8: In Sample x Out of sample - 644-fri-c4-250-25

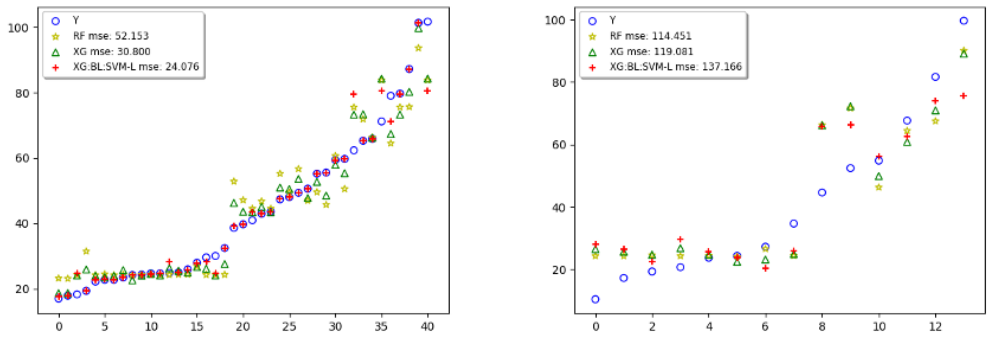

Figura C.9: In Sample x Out of sample - 228-elusage

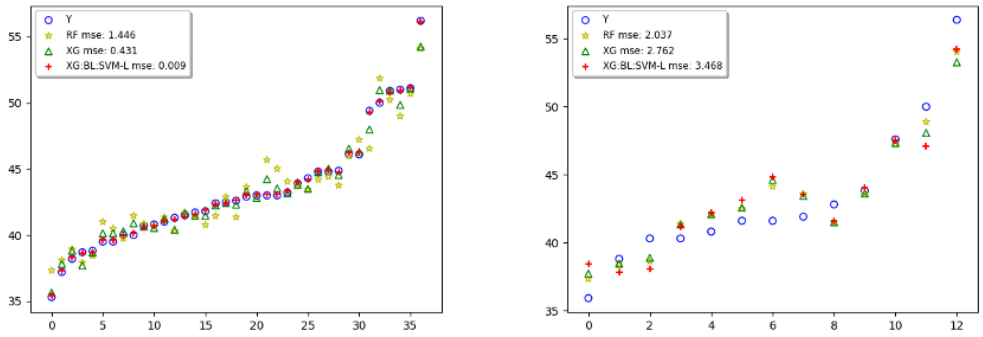

Figura C.10: In Sample x Out of sample - 1096-FacultySalaries

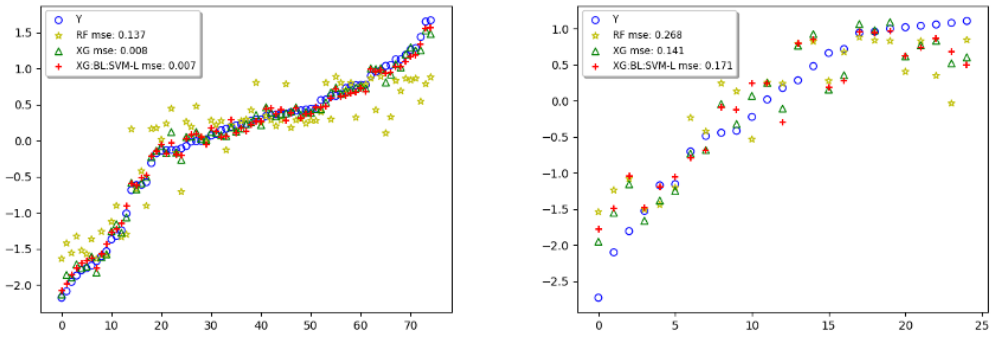

Figura C.11: In Sample x Out of sample - 594-fri-c2-100-5

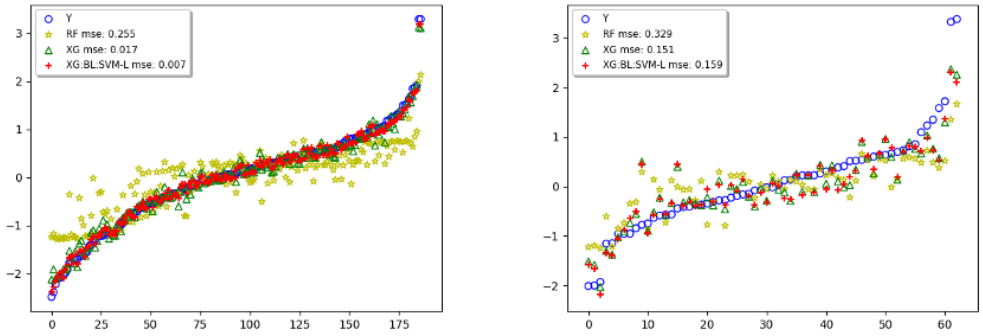

Figura C.12: In Sample x Out of sample - 615-fri-c4-250-10

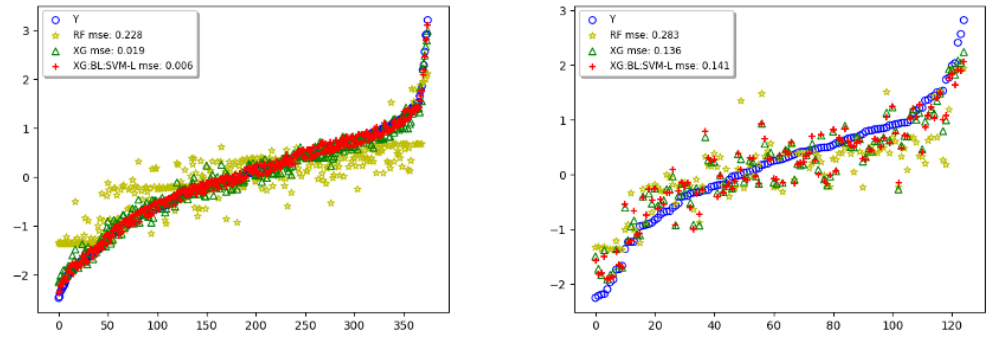

Figura C.13: In Sample x Out of sample - 616-fri-c4-500-50

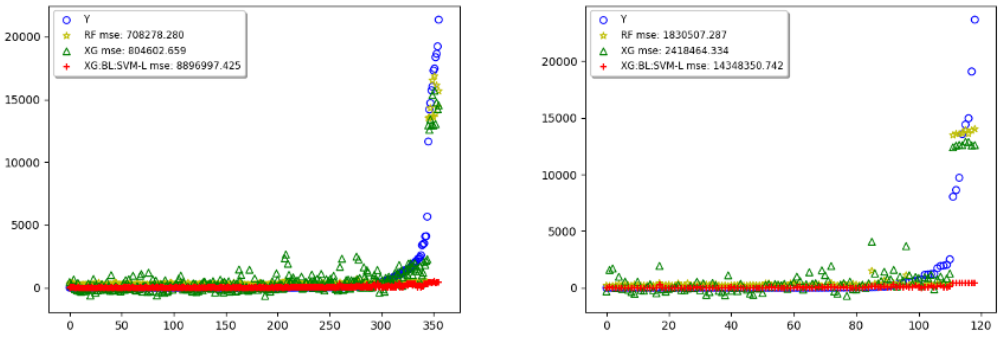

Figura C.14: In Sample x Out of sample - 557-analcatdata-apnea1

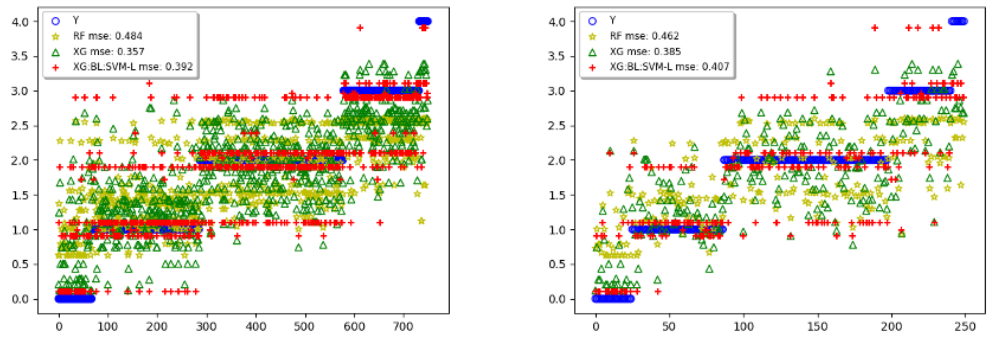

Figura C.15: In Sample x Out of sample - 1029-LEV

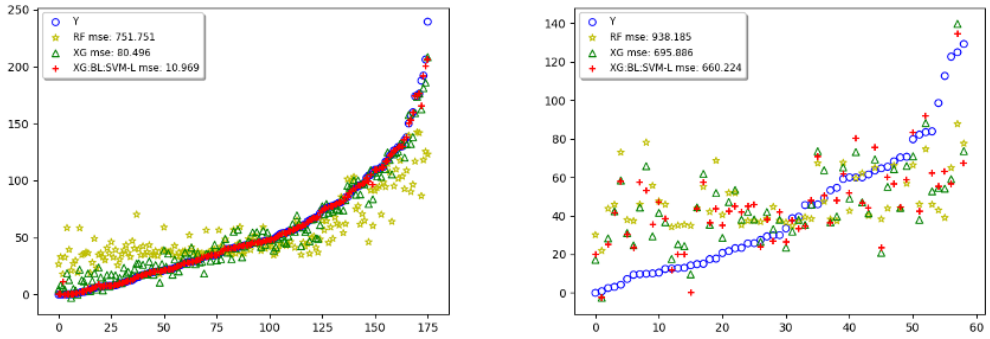

Figura C.16: In Sample x Out of sample - 695-chatfield-4

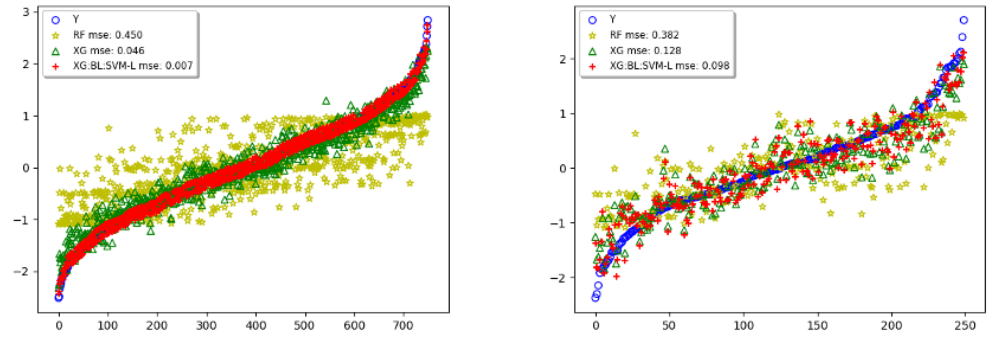

Figura C.17: In Sample x Out of sample - 590-fri-c0-1000-50

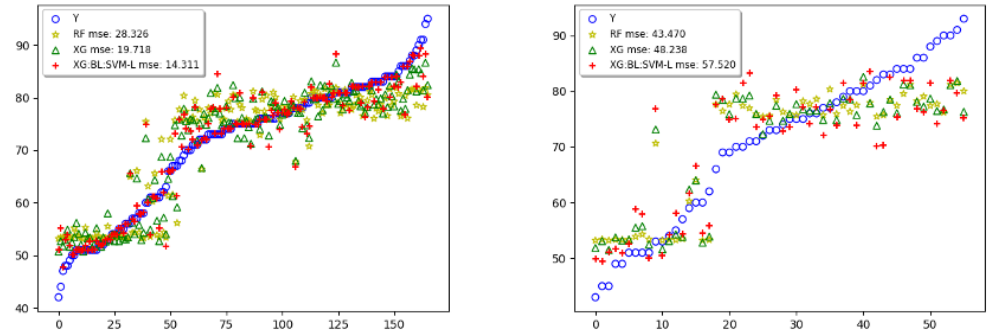

Figura C.18: In Sample x Out of sample - 712-chscase-geyser1

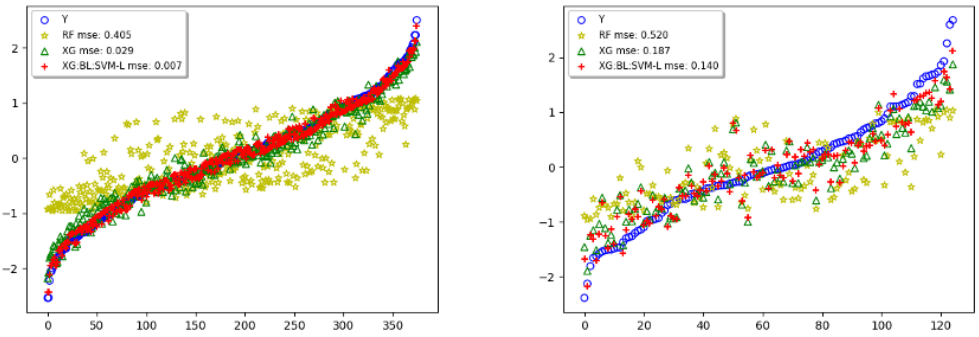

Figura C.19: In Sample x Out of sample - 633-fri-c0-500-25

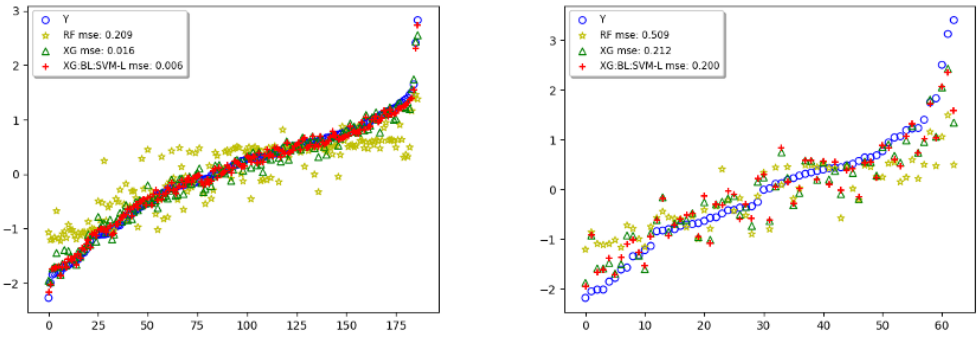

Figura C.20: In Sample x Out of sample - 602-fri-c3-250-10

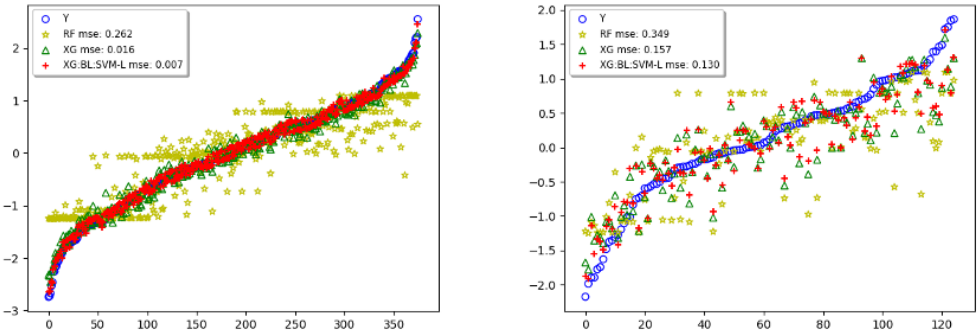

Figura C.21: In Sample x Out of sample - 637-fri-c1-500-50

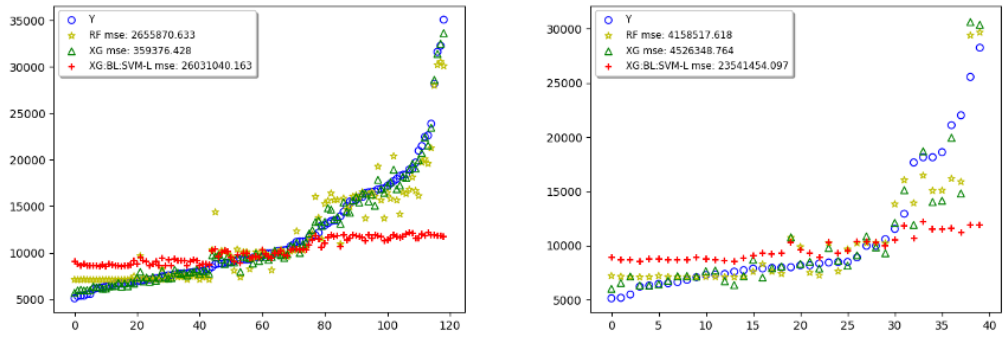

Figura C.22: In Sample x Out of sample - 207-autoPrice

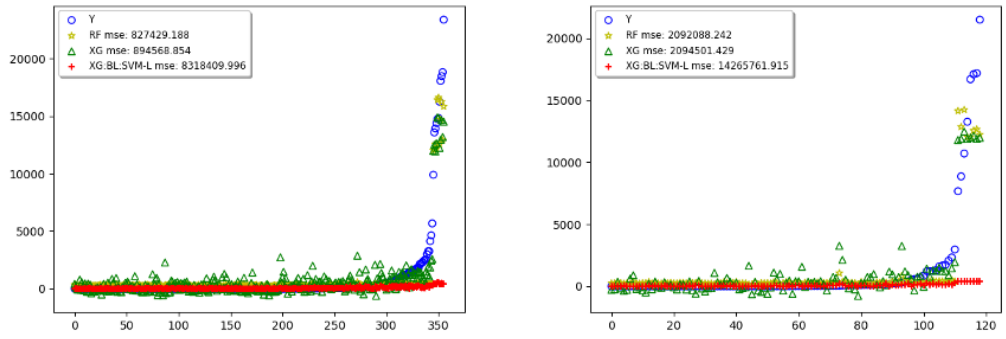

Figura C.23: In Sample x Out of sample - 556-analcatdata-apnea2

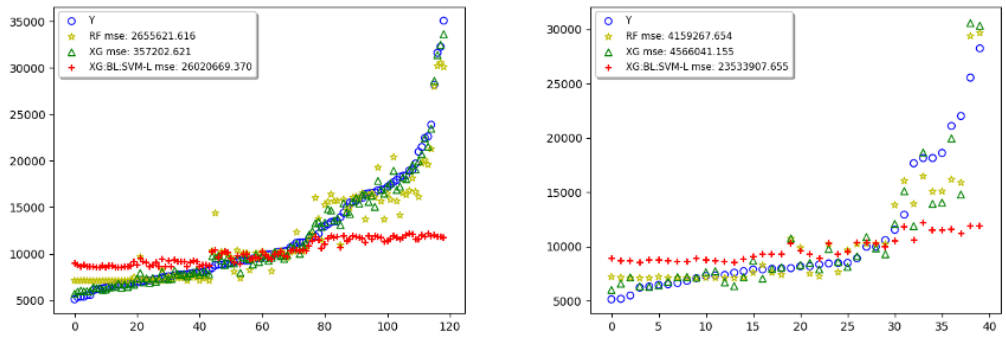

Figura C.24: In Sample x Out of sample - 195-auto-price

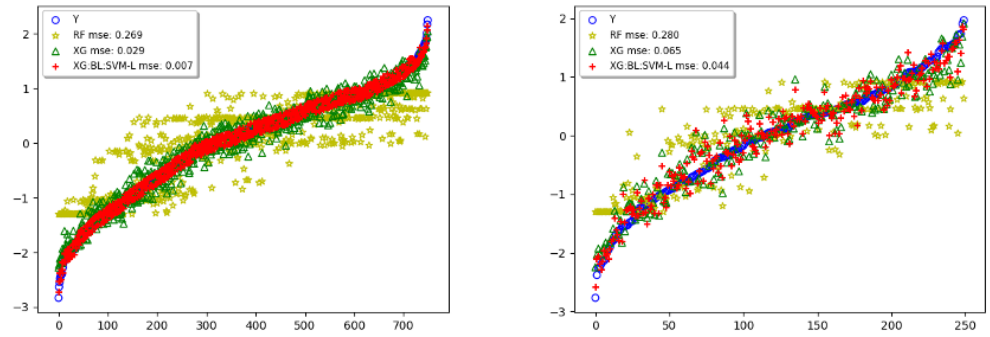

Figura C.25: In Sample x Out of sample - 593-fri-c1-1000-10

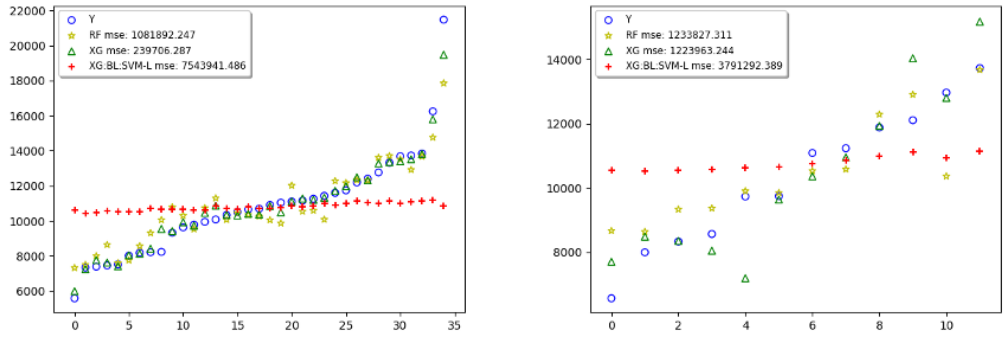

Figura C.26: In Sample x Out of sample - 659-sleuth-ex1714

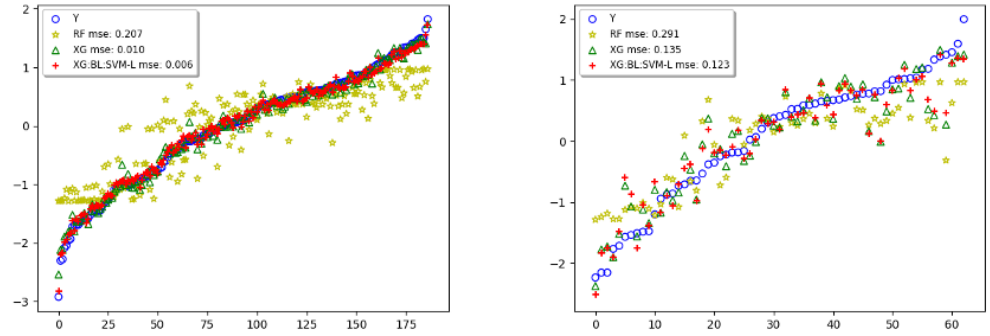

Figura C.27: In Sample x Out of sample - 647-fri-c1-250-10

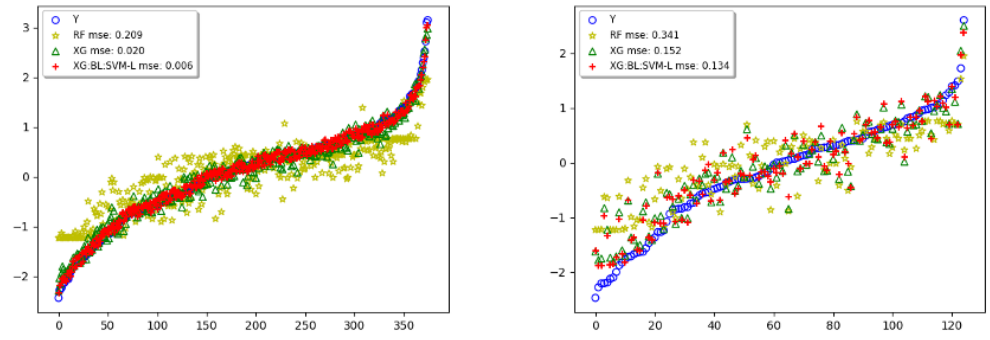

Figura C.28: In Sample x Out of sample - 581-fri-c3-500-25

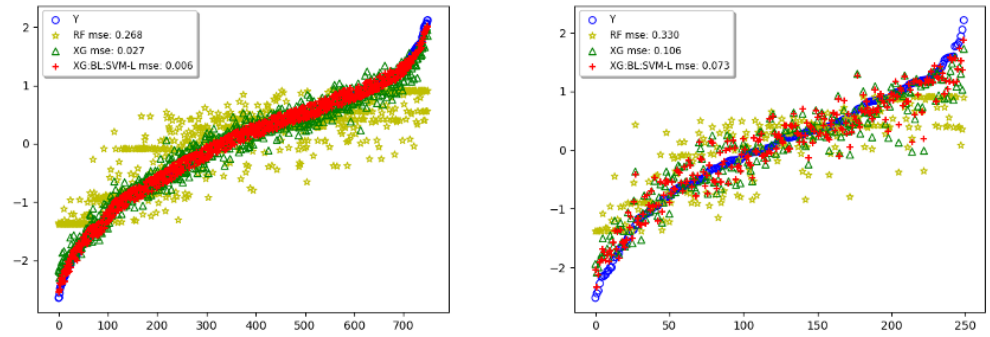

Figura C.29: In Sample x Out of sample - 583-fri-c1-1000-50

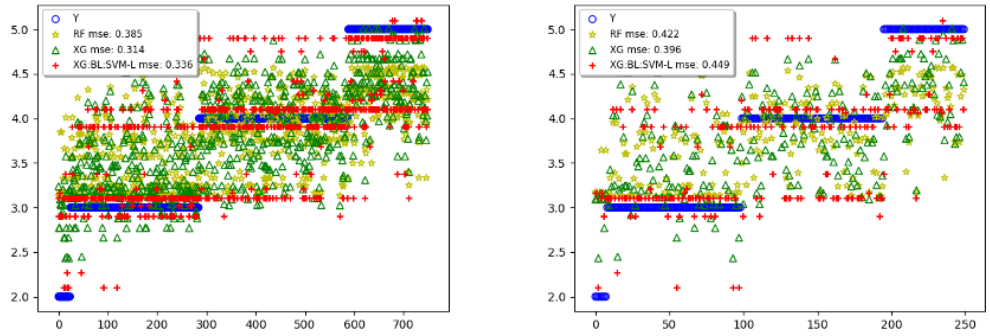

Figura C.30: In Sample x Out of sample - 1028-SWD

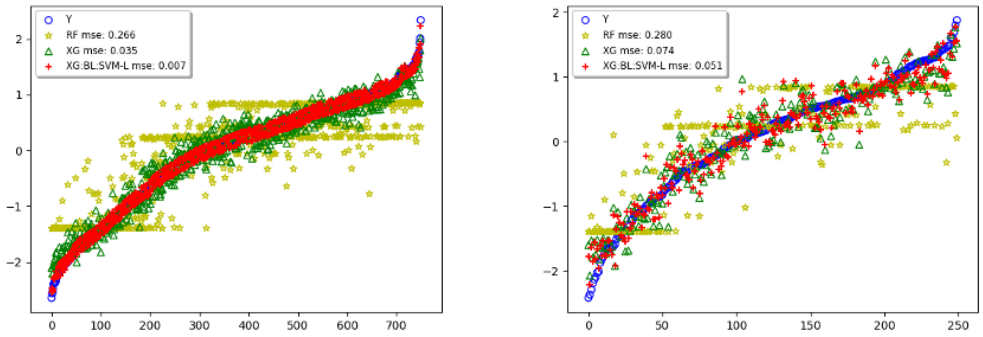

Figura C.31: In Sample x Out of sample - 606-fri-c2-1000-10

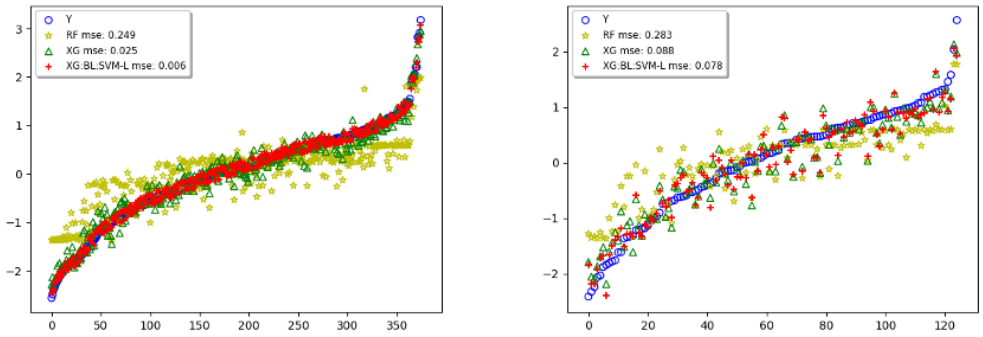

Figura C.32: In Sample x Out of sample - 604-fri-c4-500-10

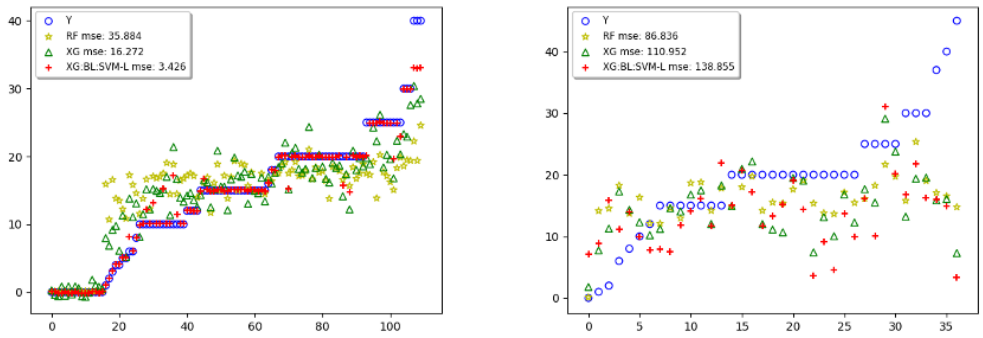

Figura C.33: In Sample x Out of sample - 665-sleuth-case2002

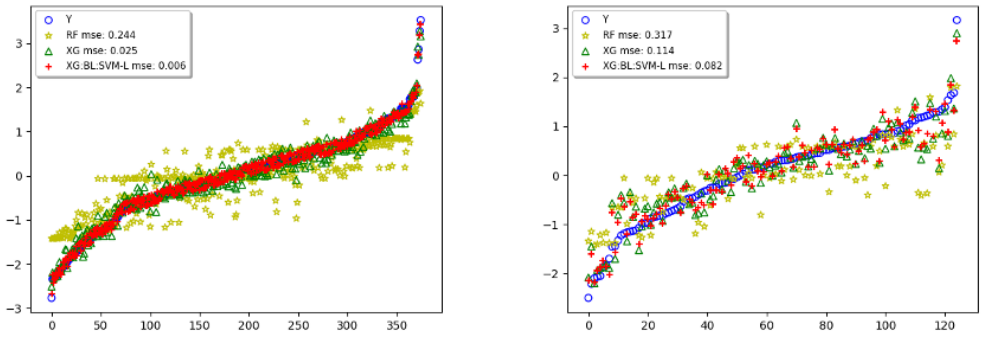

Figura C.34: In Sample x Out of sample - 617-fri-c3-500-5

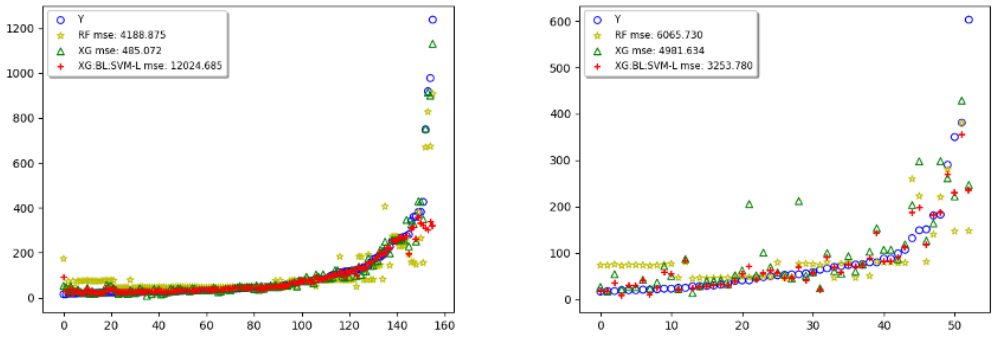

Figura C.35: In Sample x Out of sample - 561-cpu

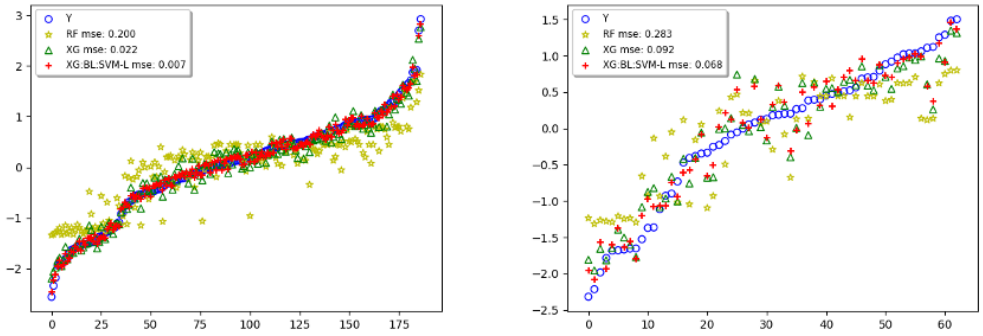

Figura C.36: In Sample x Out of sample - 613-fri-c3-250-5

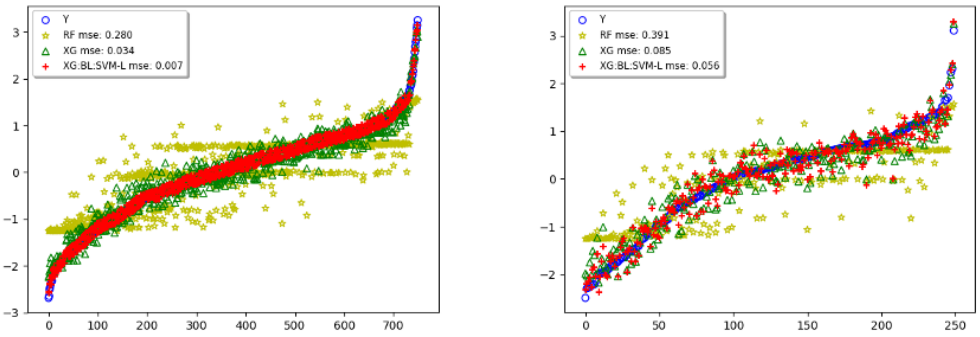

Figura C.37: In Sample x Out of sample - 628-fri-c3-1000-5

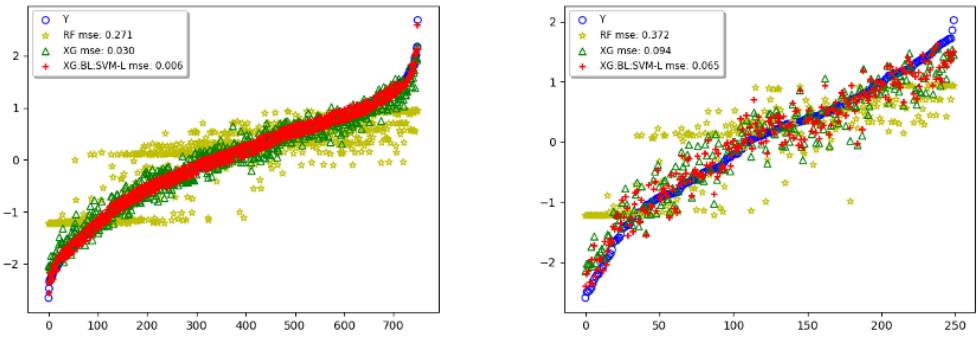

Figura C.38: In Sample x Out of sample - 620-fri-c1-1000-25

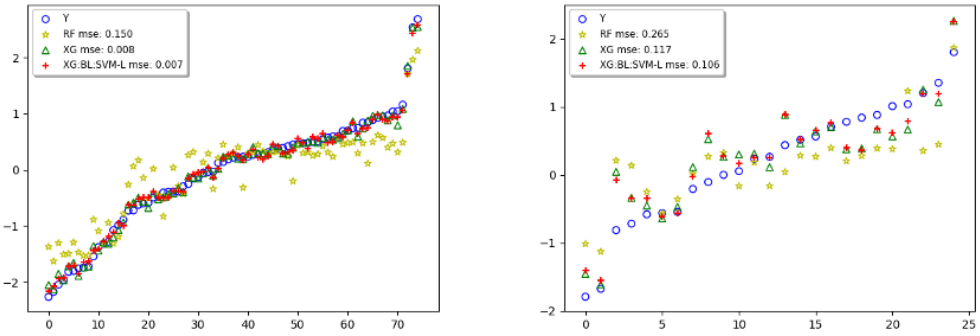

Figura C.39: In Sample x Out of sample - 611-fri-c3-100-5

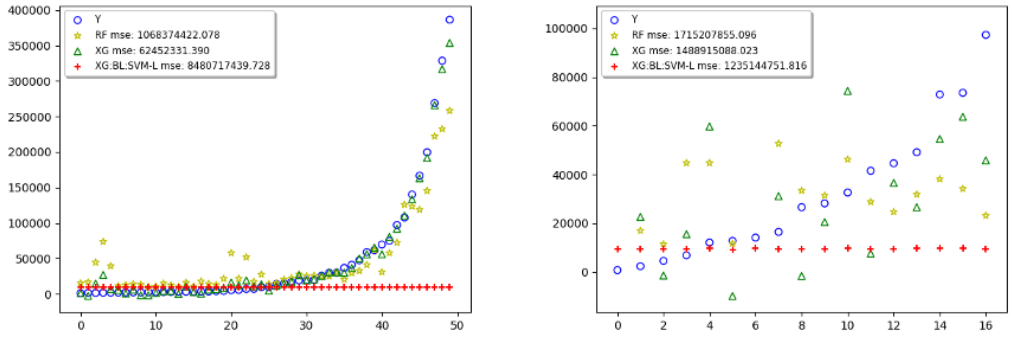

Figura C.40: In Sample x Out of sample - 527-analcatdata-election2000

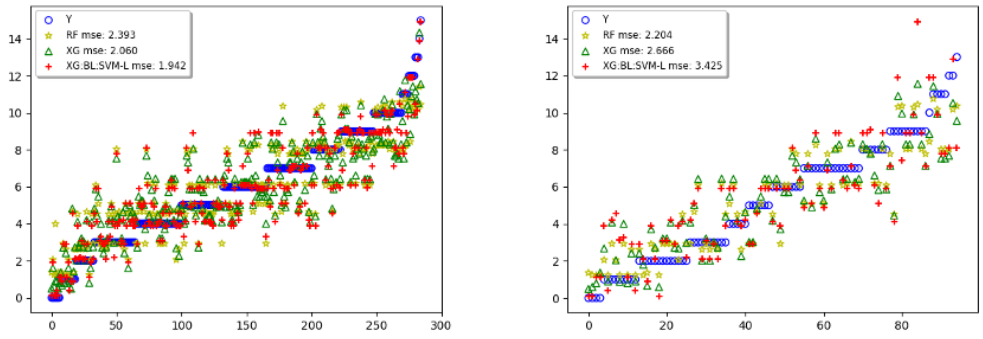

Figura C.41: In Sample x Out of sample - 519-vinnie

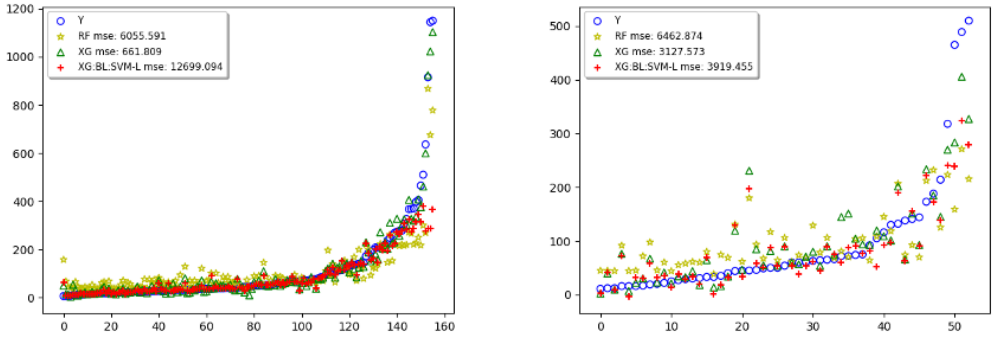

Figura C.42: In Sample x Out of sample - 230-machine-cpu

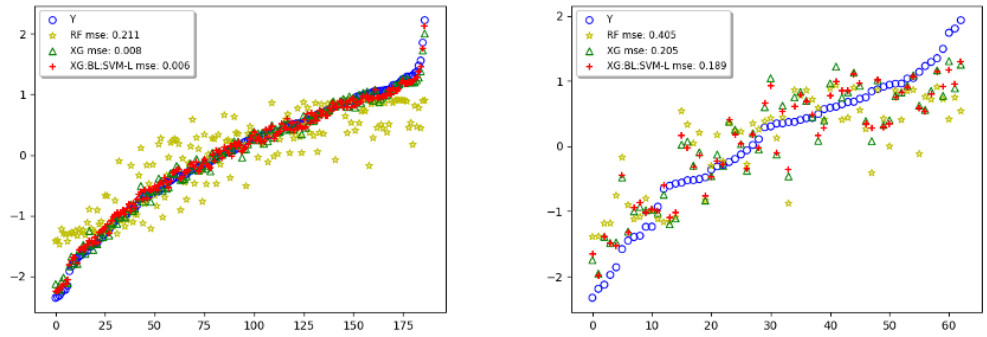

Figura C.43: In Sample x Out of sample - 605-fri-c2-250-25

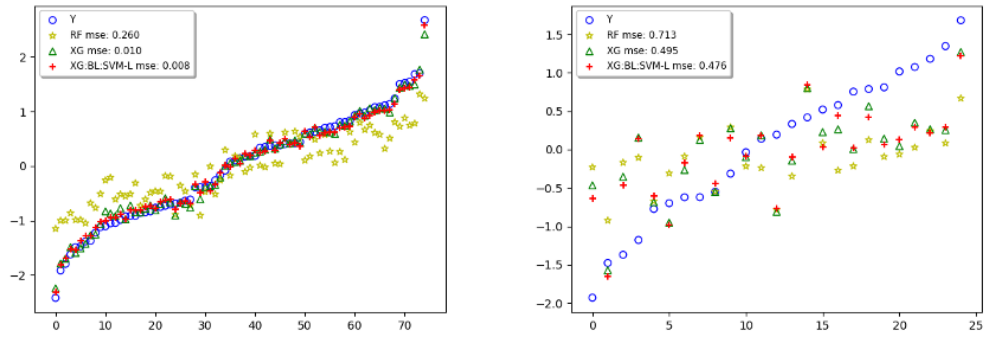

Figura C.44: In Sample x Out of sample - 624-fri-c0-100-5

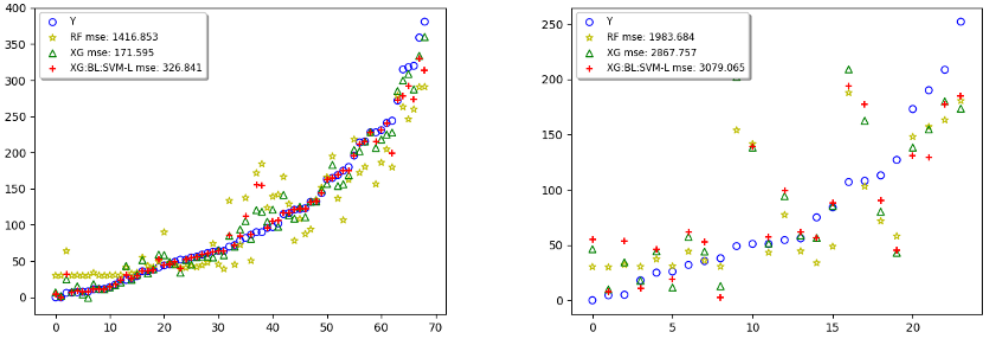

Figura C.45: In Sample x Out of sample - 706-sleuth-case1202

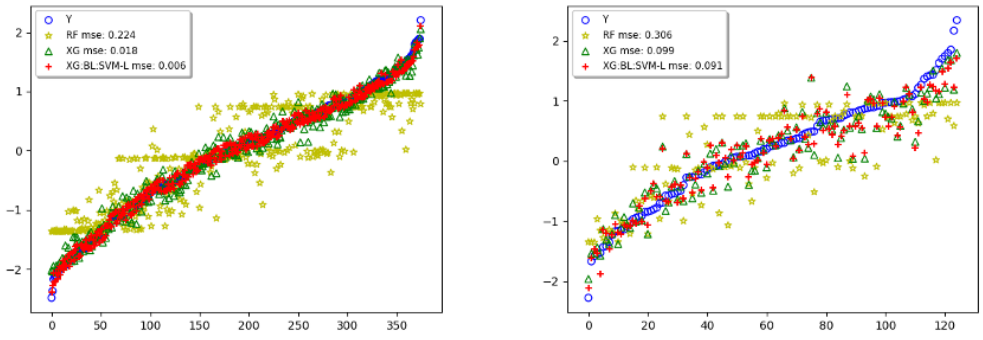

Figura C.46: In Sample x Out of sample - 641-fri-c1-500-10

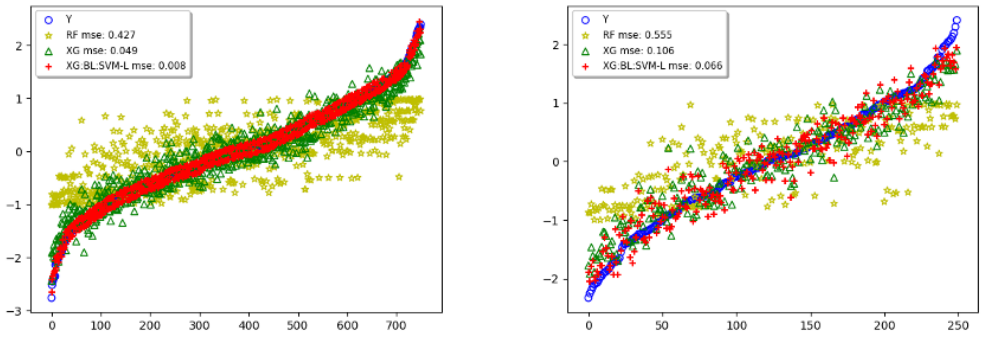

Figura C.47: In Sample x Out of sample - 609-fri-c0-1000-5

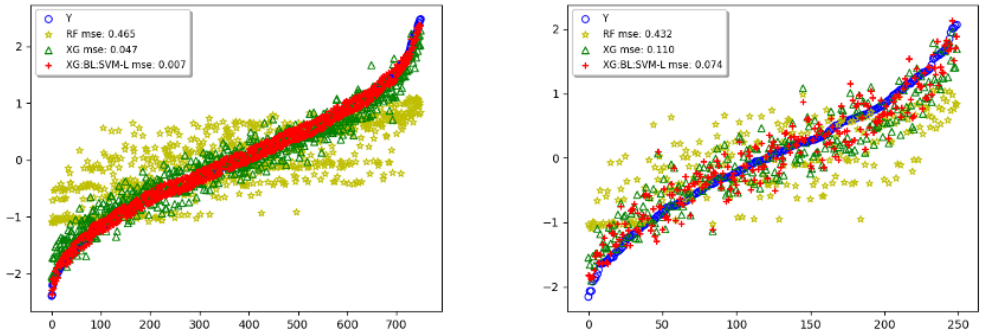

Figura C.48: In Sample x Out of sample - 598-fri-c0-1000-25

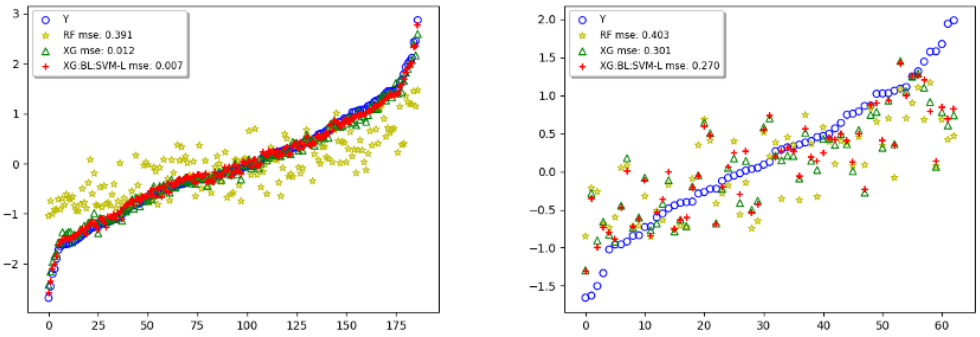

Figura C.49: In Sample x Out of sample - 603-fri-c0-250-50

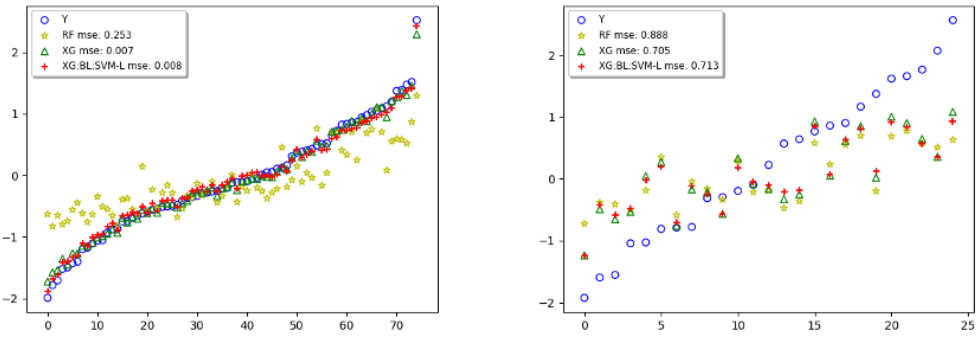

Figura C.50: In Sample x Out of sample - 621-fri-c0-100-10

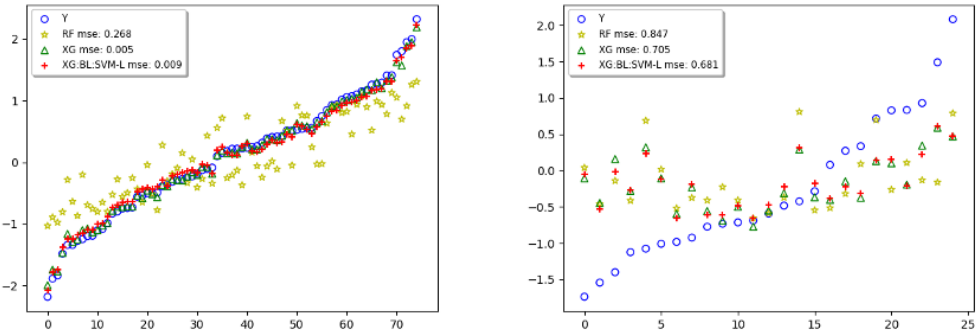

Figura C.51: In Sample x Out of sample - 651-fri-c0-100-25

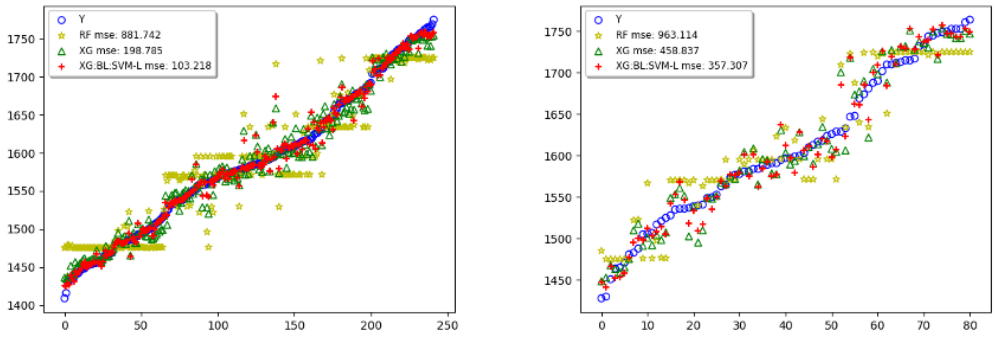

Figura C.52: In Sample x Out of sample - 690-visualizing-galaxy

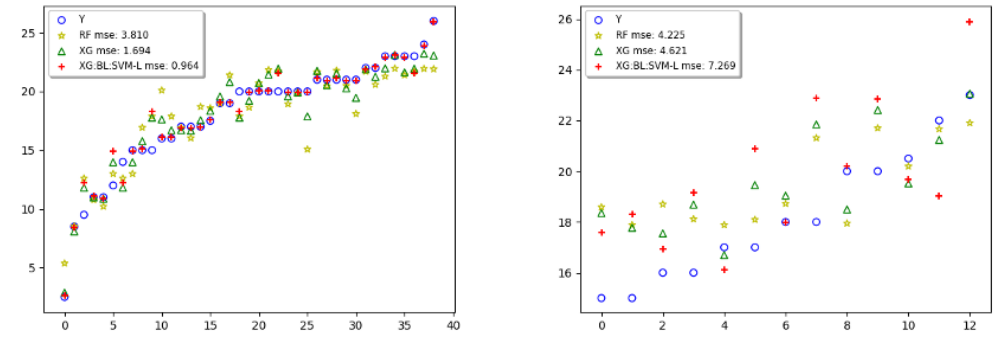

Figura C.53: In Sample x Out of sample - 192-vineyard

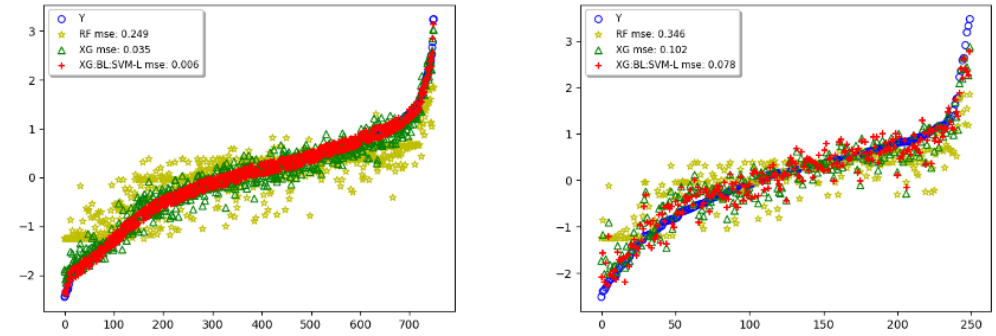

Figura C.54: In Sample x Out of sample - 592-fri-c4-1000-25

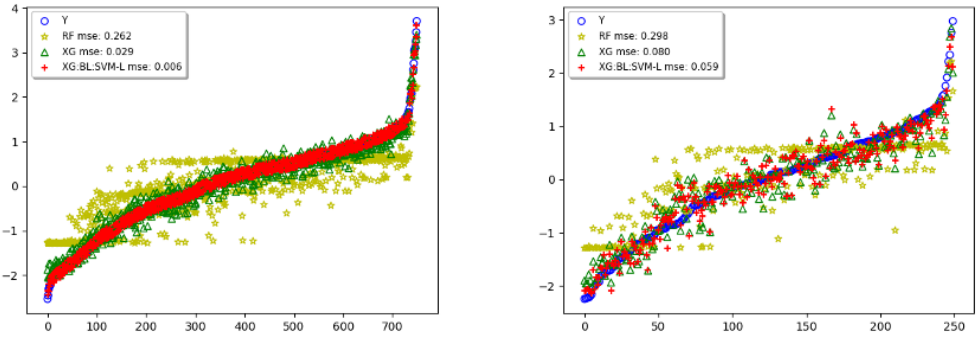

Figura C.55: In Sample x Out of sample - 586-fri-c3-1000-25

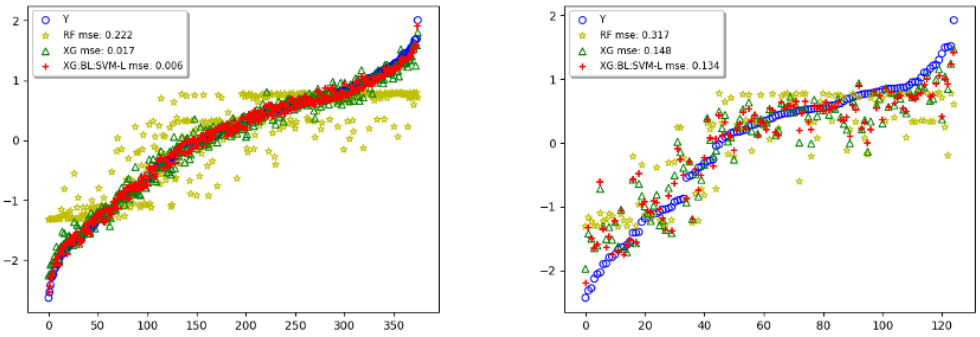

Figura C.56: In Sample x Out of sample - 626-fri-c2-500-50

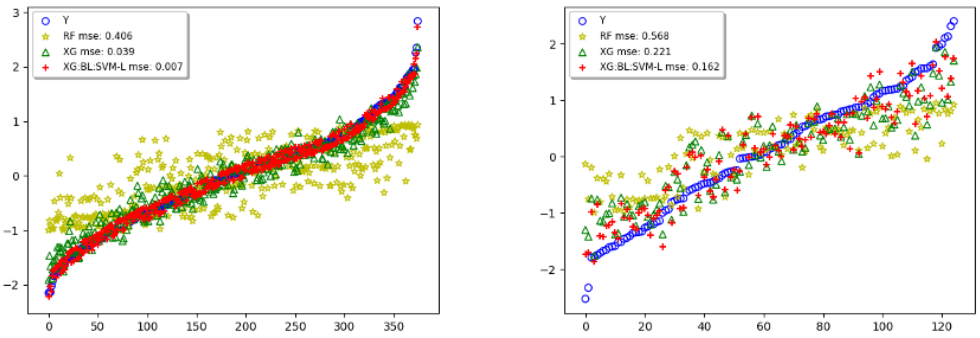

Figura C.57: In Sample x Out of sample - 654-fri-c0-500-10

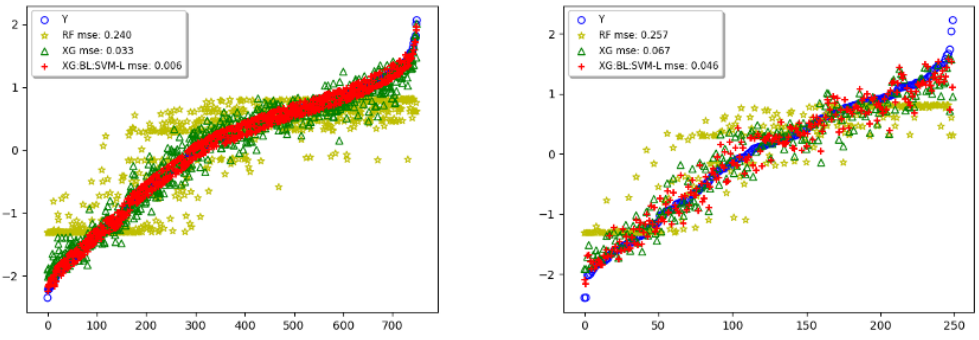

Figura C.58: In Sample x Out of sample - 599-fri-c2-1000-5

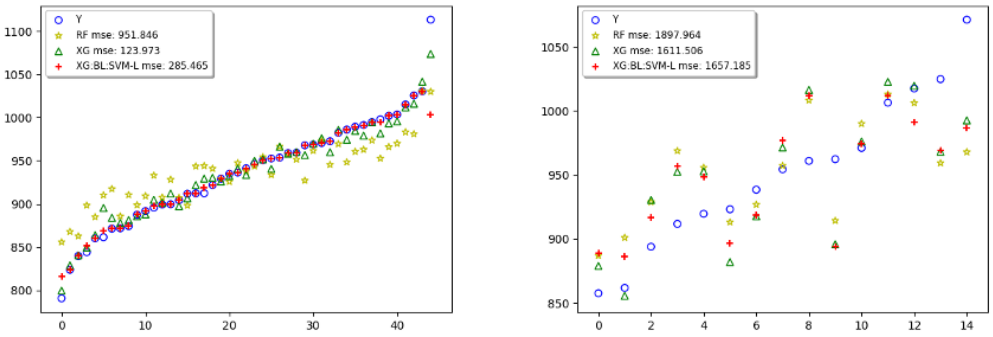

Figura C.59: In Sample x Out of sample - 542-pollution

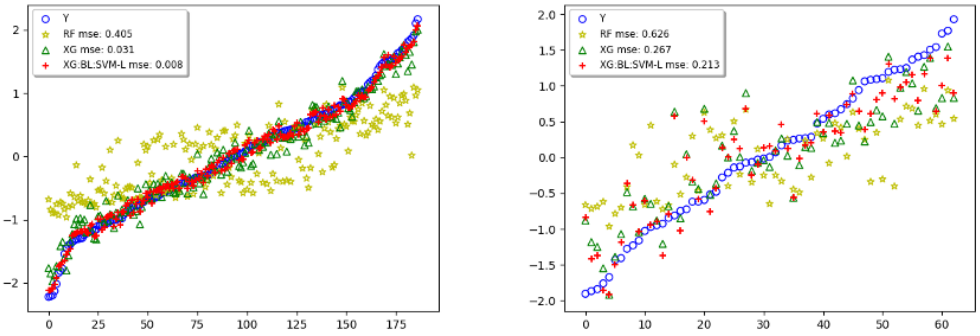

Figura C.60: In Sample x Out of sample - 579-fri-c0-250-5

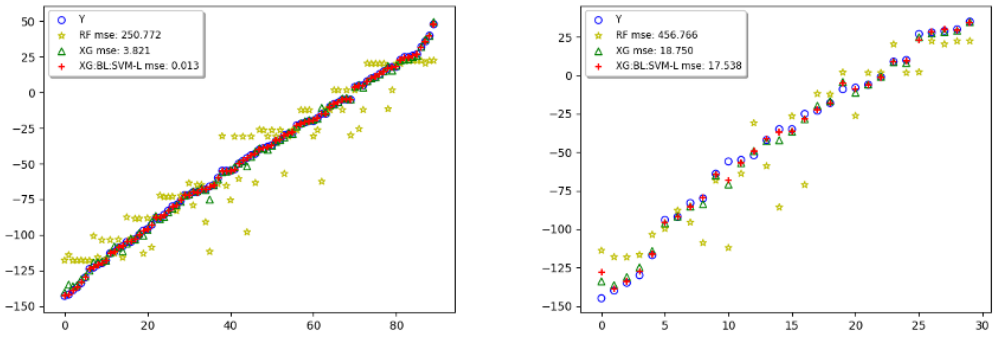

Figura C.61: In Sample x Out of sample - 663-rabe-266

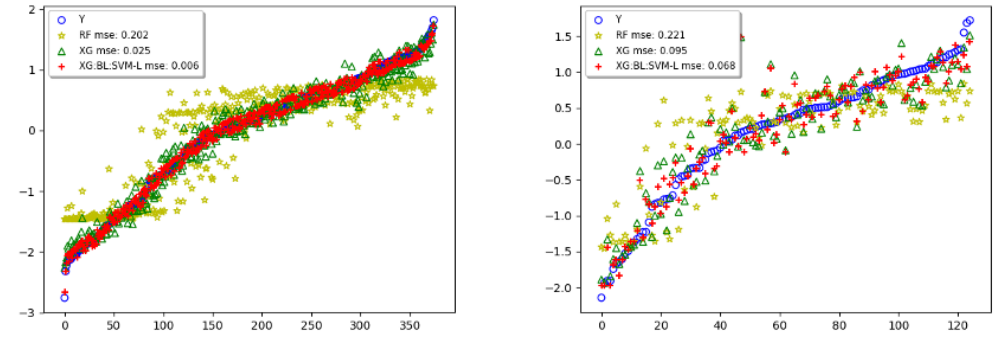

Figura C.62: In Sample x Out of sample - 597-fri-c2-500-5

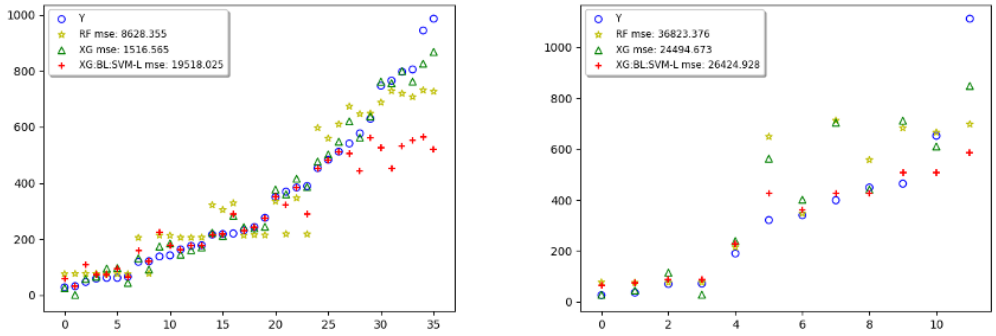

Figura C.63: In Sample x Out of sample - 485-analcatdata-vehicle

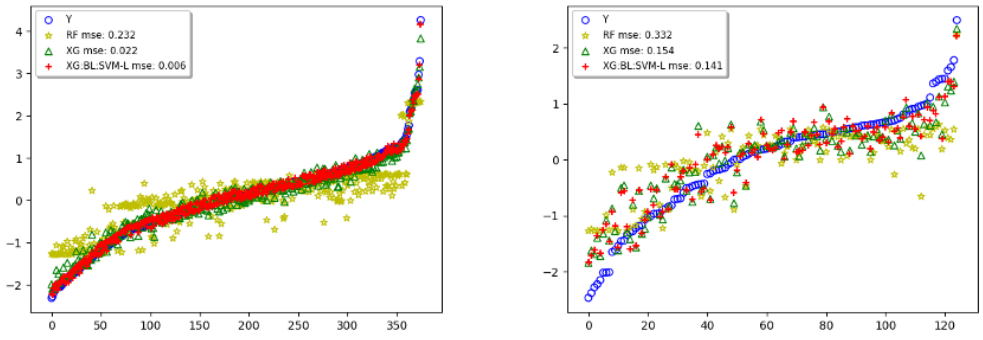

Figura C.64: In Sample x Out of sample - 645-fri-c3-500-50

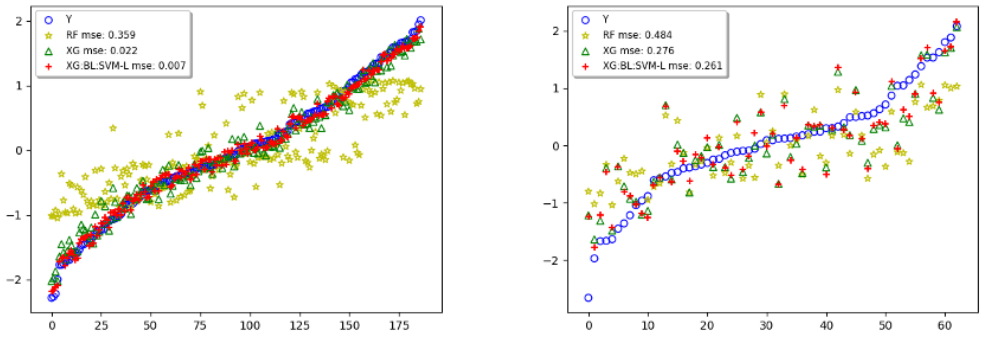

Figura C.65: In Sample x Out of sample - 635-fri-c0-250-10

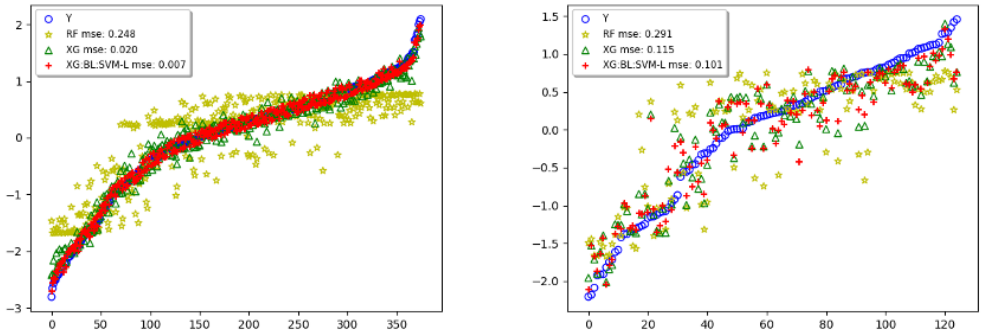

Figura C.66: In Sample x Out of sample - 643-fri-c2-500-25

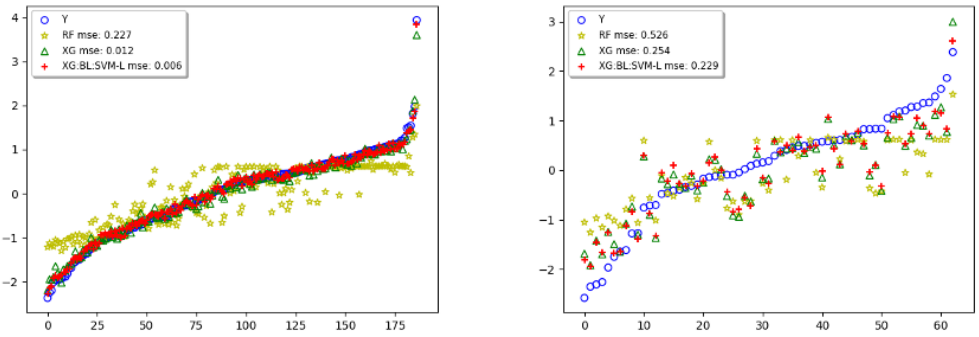

Figura C.67: In Sample x Out of sample - 658-fri-c3-250-25

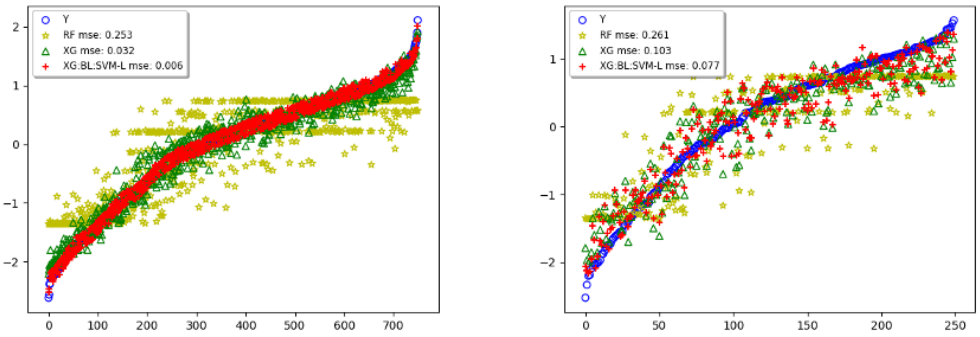

Figura C.68: In Sample x Out of sample - 622-fri-c2-1000-50

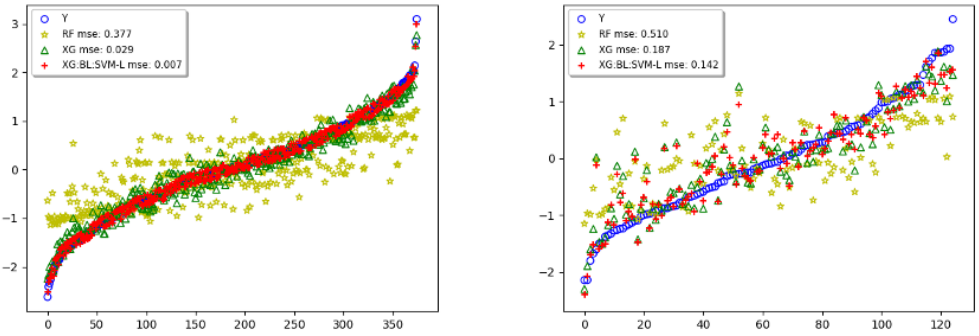

Figura C.69: In Sample x Out of sample - 649-fri-c0-500-5

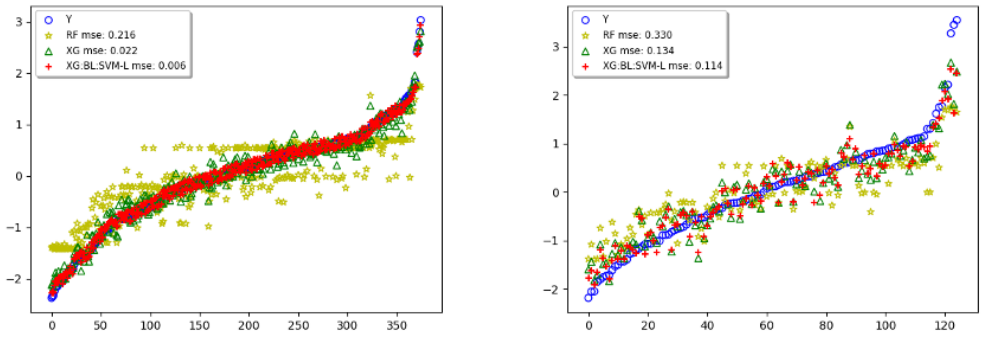

Figura C.70: In Sample x Out of sample - 646-fri-c3-500-10

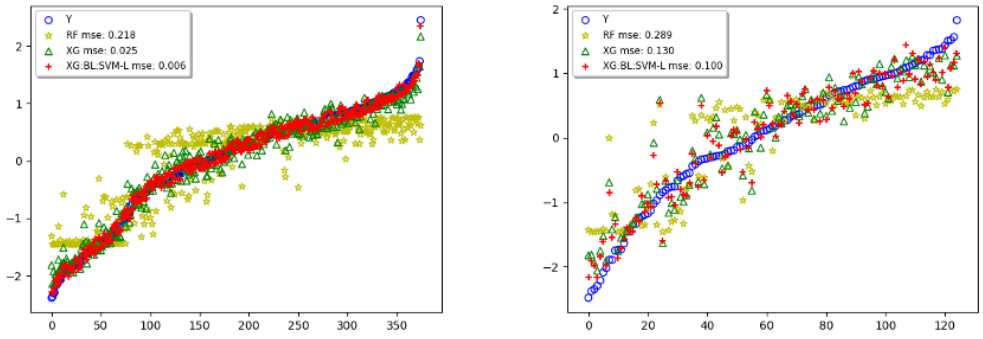

Figura C.71: In Sample x Out of sample - 627-fri-c2-500-10

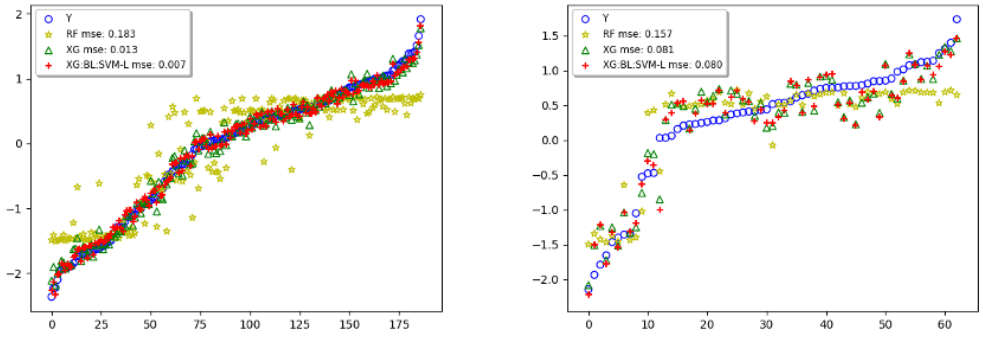

Figura C.72: In Sample x Out of sample - 657-fri-c2-250-10

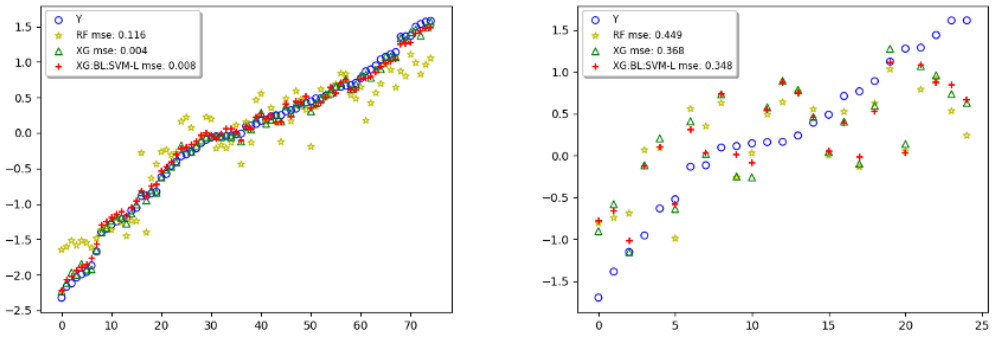

Figura C.73: In Sample x Out of sample - 634-fri-c2-100-10

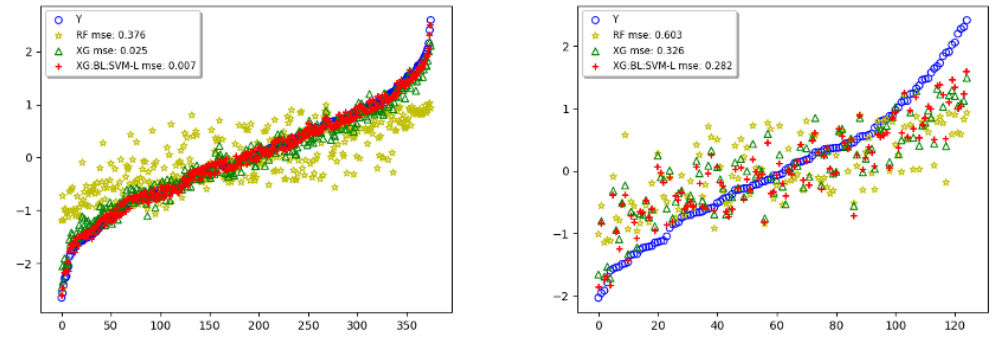

Figura C.74: In Sample x Out of sample - 650-fri-c0-500-50

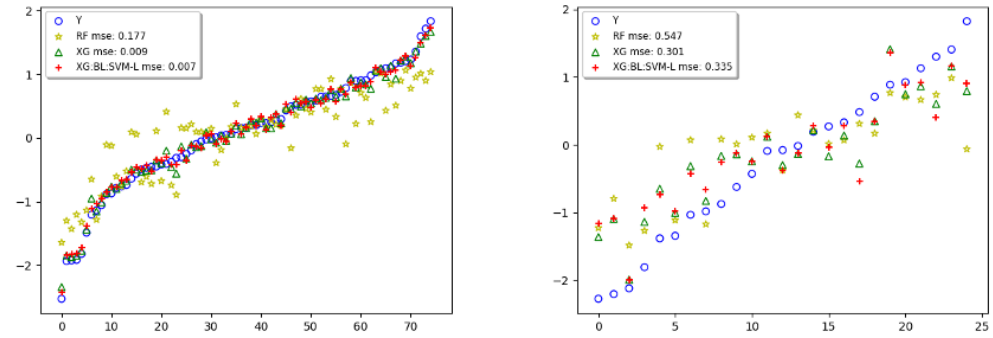

Figura C.75: In Sample x Out of sample - 656-fri-c1-100-5

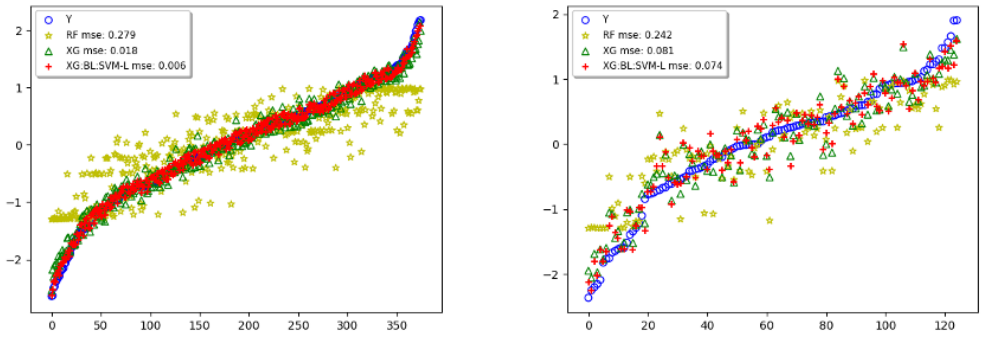

Figura C.76: In Sample x Out of sample - 582-fri-c1-500-25

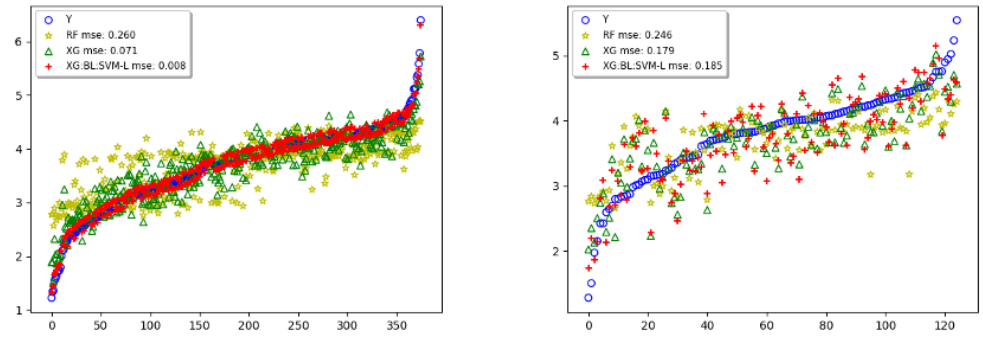

Figura C.77: In Sample x Out of sample - 547-no2

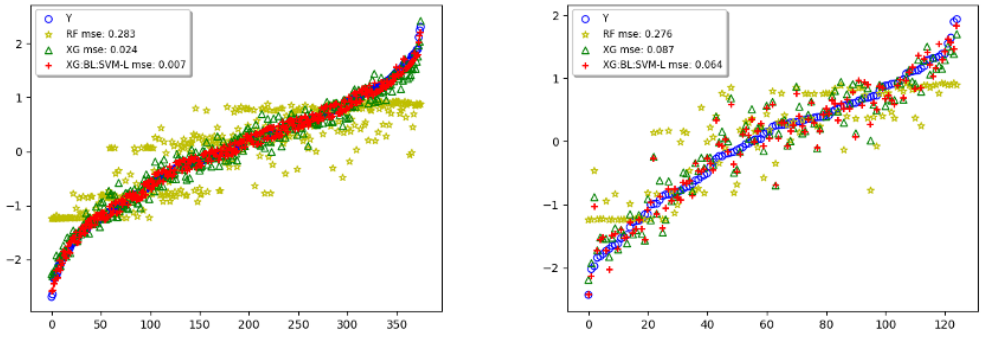

Figura C.78: In Sample x Out of sample - 631-fri-c1-500-5

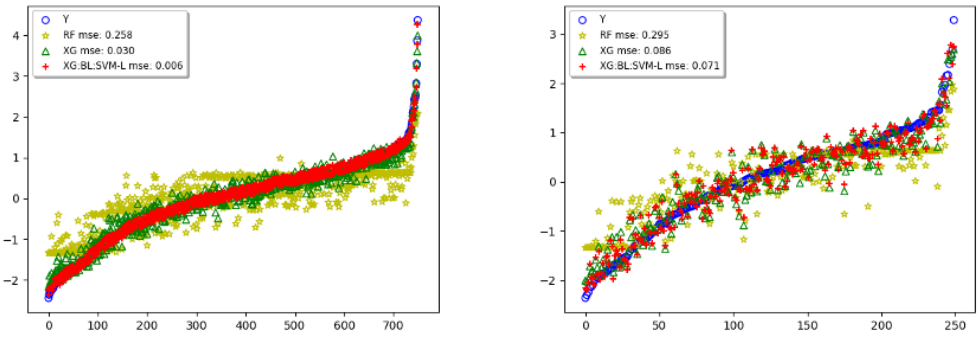

Figura C.79: In Sample x Out of sample - 607-fri-c4-1000-50

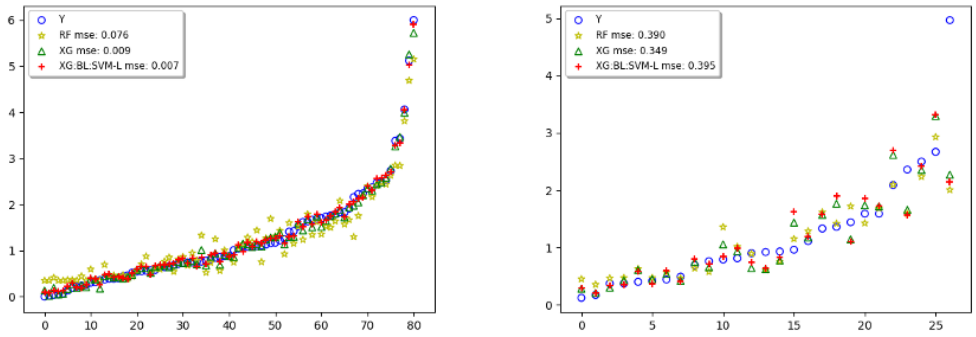

Figura C.80: In Sample x Out of sample - 210-cloud

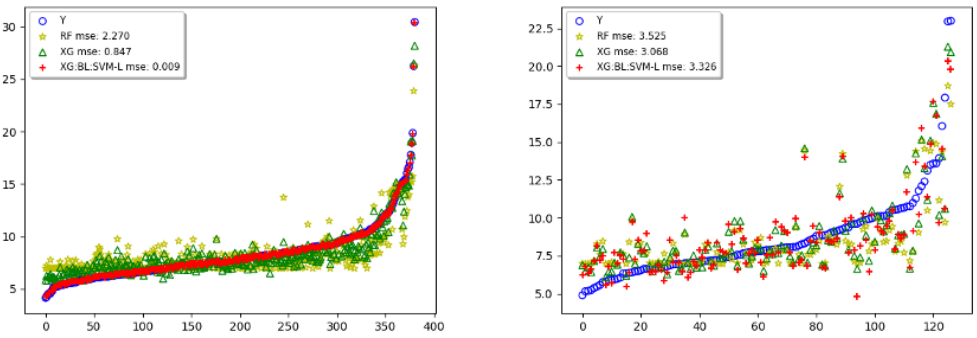

Figura C.81: In Sample x Out of sample - 666-rmftsa-ladata

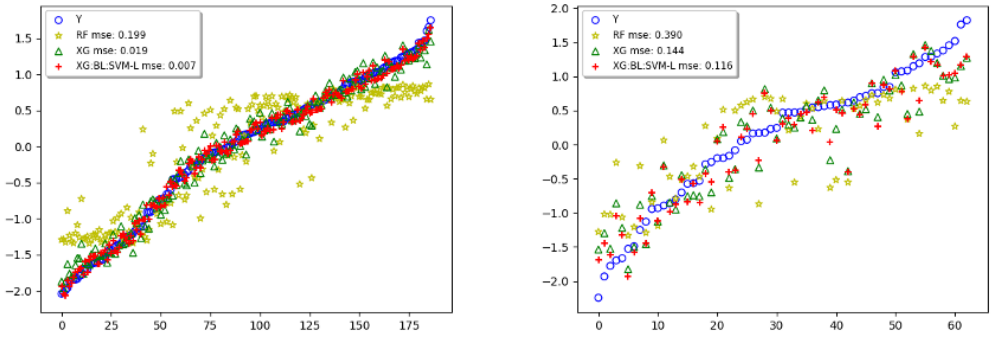

Figura C.82: In Sample x Out of sample - 596-fri-c2-250-5

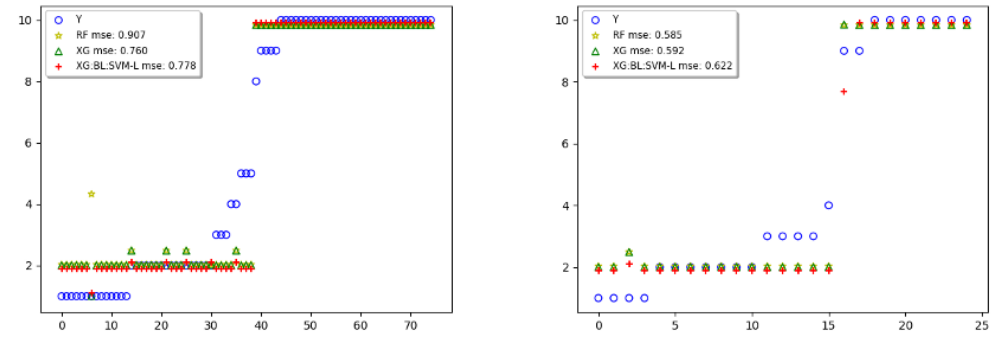

Figura C.83: In Sample x Out of sample - 523-analcatdata-neavote

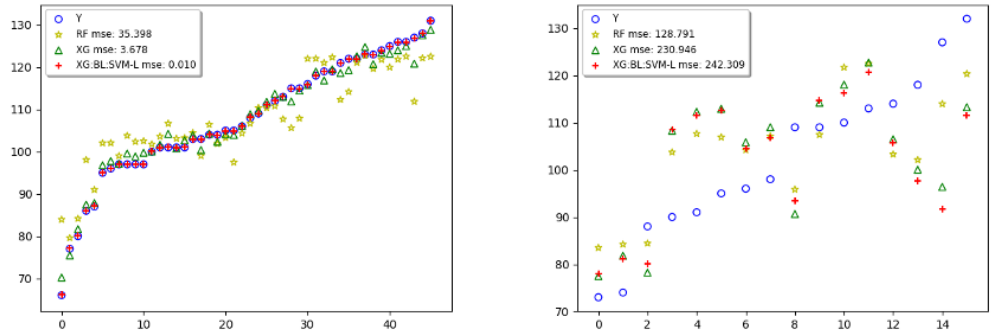

Figura C.84: In Sample x Out of sample - 687-sleuth-ex1605

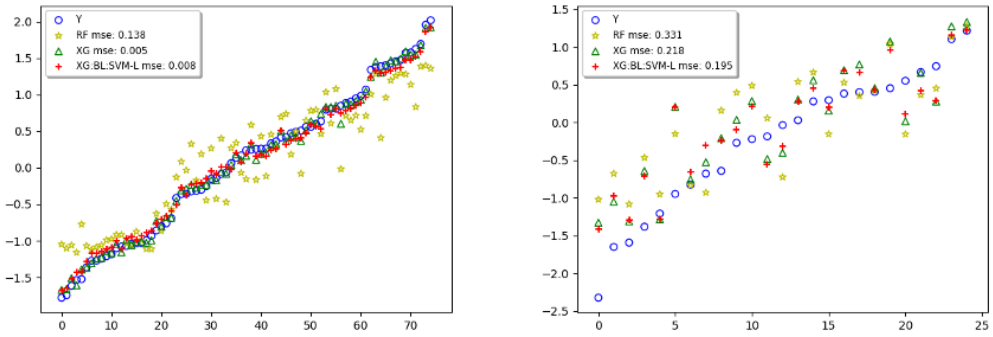

Figura C.85: In Sample x Out of sample - 591-fri-c1-100-10

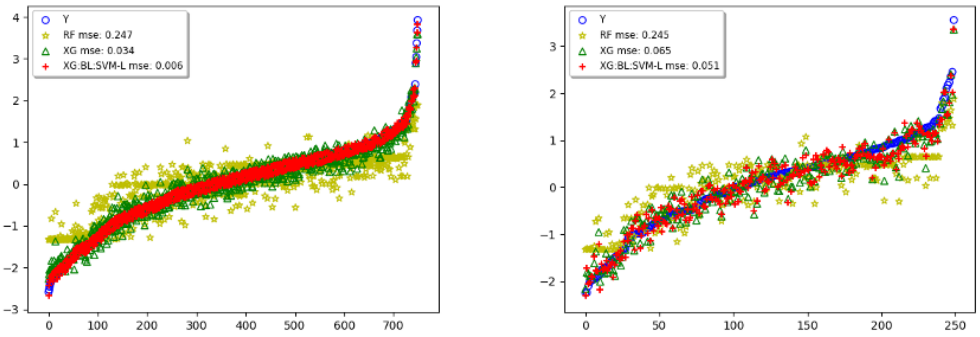

Figura C.86: In Sample x Out of sample - 608-fri-c3-1000-10

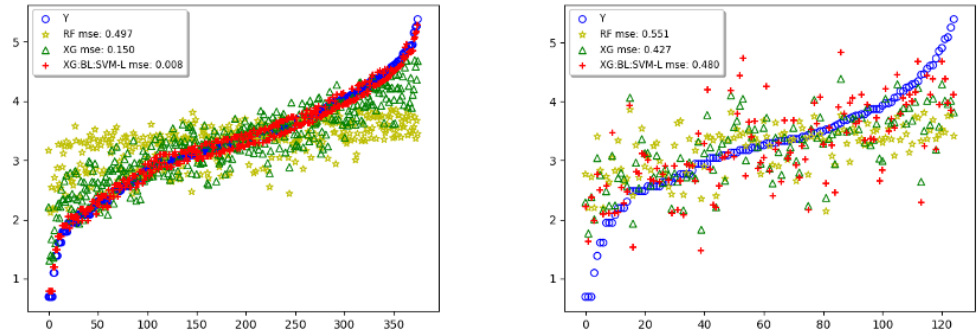

Figura C.87: In Sample x Out of sample - 522-pm10

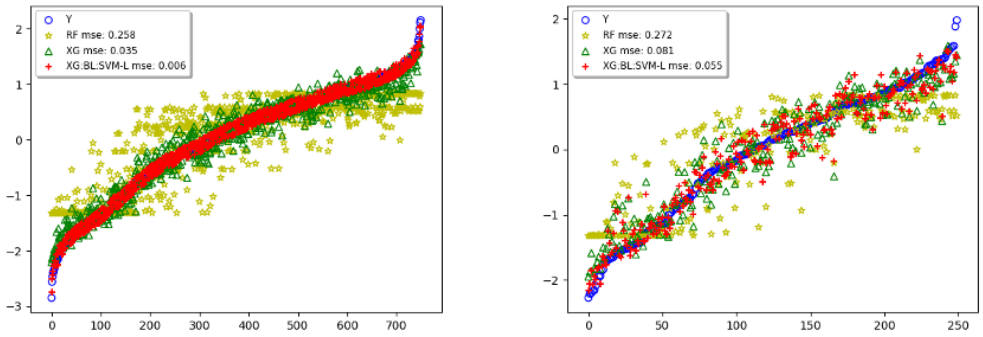

Figura C.88: In Sample x Out of sample - 589-fri-c2-1000-25

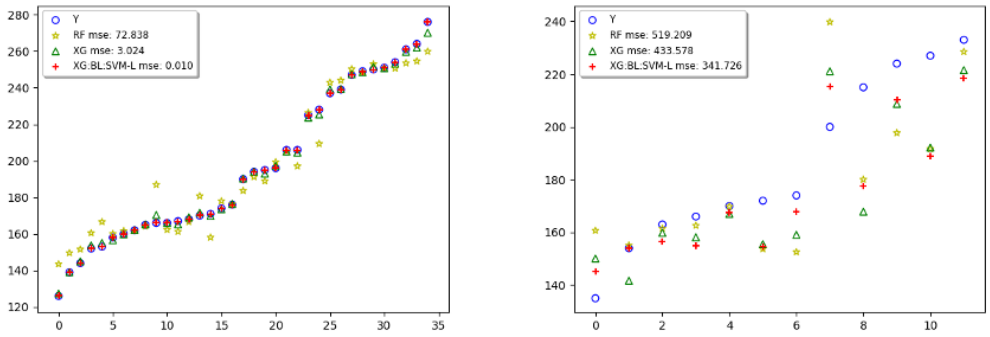

Figura C.89: In Sample x Out of sample - 1089-USCrime

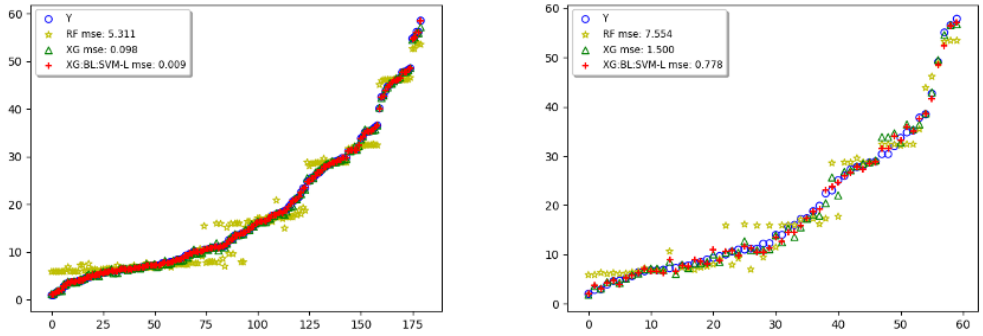

Figura C.90: In Sample x Out of sample - 505-tecator

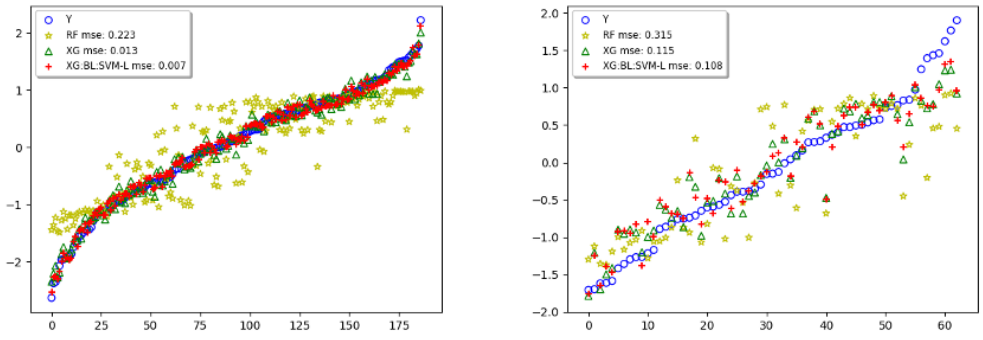

Figura C.91: In Sample x Out of sample - 601-fri-c1-250-5

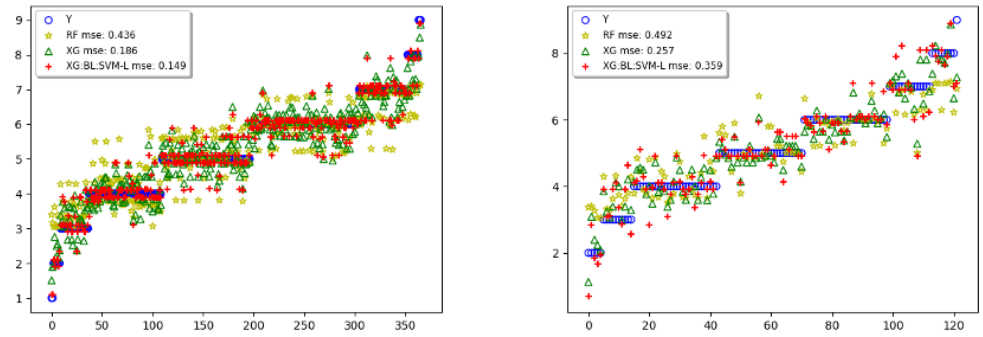

Figura C.92: In Sample x Out of sample - 1027-ESL

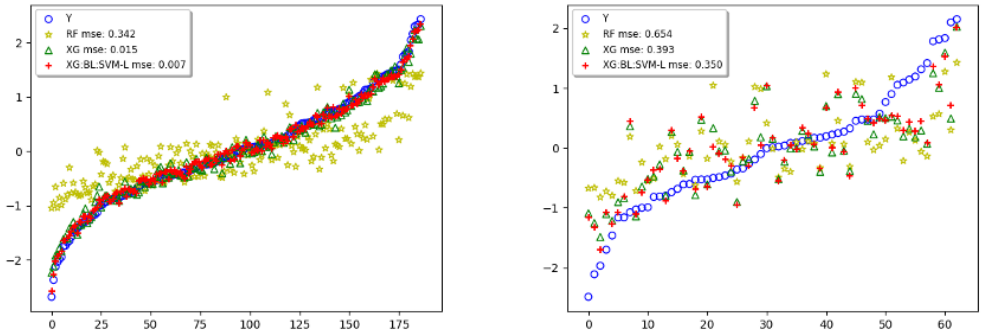

Figura C.93: In Sample x Out of sample - 653-fri-c0-250-25

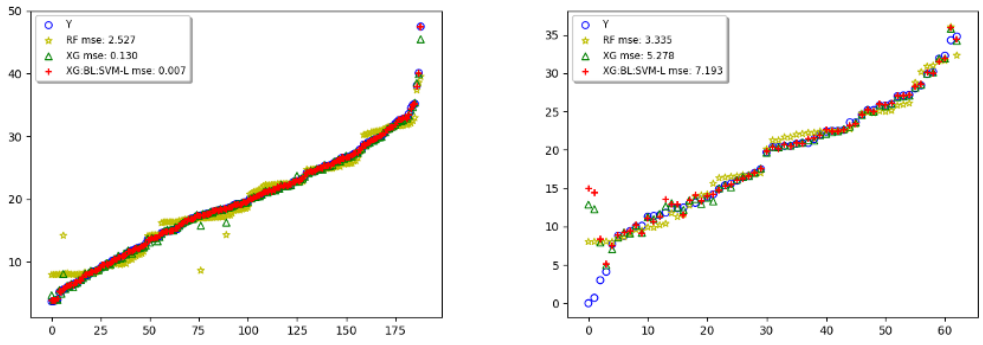

Figura C.94: In Sample x Out of sample - 560-bodyfat

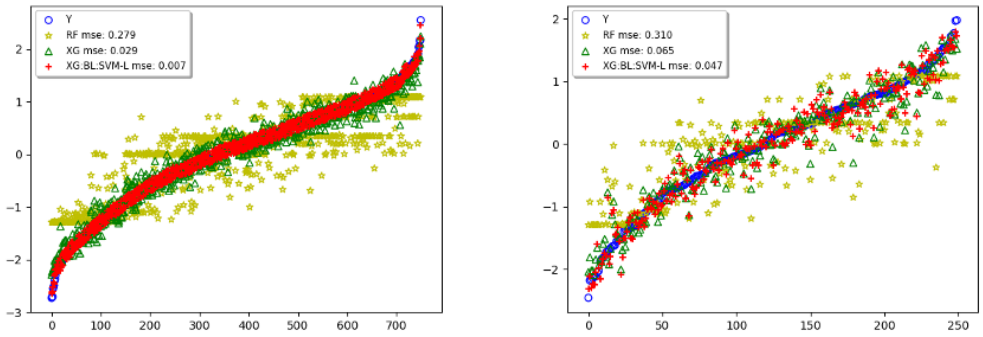

Figura C.95: In Sample x Out of sample - 612-fri-c1-1000-5

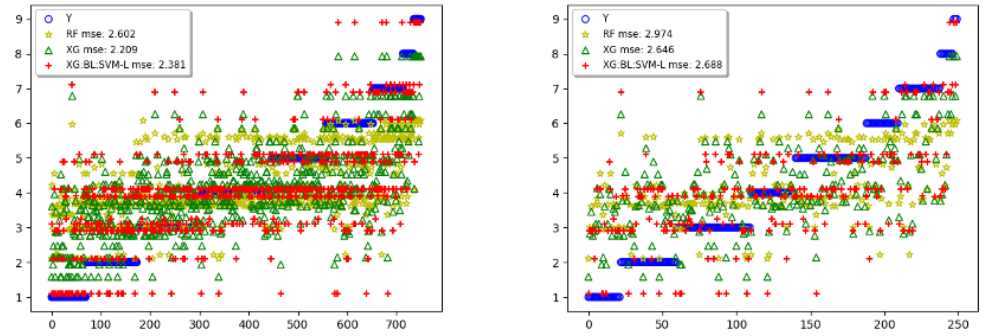

Figura C.96: In Sample x Out of sample - 1030-ERA