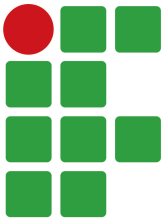

INSTITUTO FEDERAL
Goiás
Câmpus Anápolis

Pedro Henrique Silva Rodrigues

Análise Comparativa dos Algoritmos de Random Forest e Random Regression

Anápolis-GO

Novembro - 2018

Pedro Henrique Silva Rodrigues

Análise Comparativa dos Algoritmos de Random Forest e Random Regression

Relatório Científico apresentado ao curso de Bacharelado em Ciências da Computação, como exigência da Disciplina de Inteligência Artificial

Instituto Federal de Goiás - Campus Anápolis

Anápolis-GO
Novembro - 2018

Lista de ilustrações

Sumário

Resumo

Um dos mais importantes métodos computacionais modernos são os algoritmos de Inteligência Artificial. Cada um dos algoritmos existentes possui peculiaridades e são bons para problemas específicos. No geral, ao realizar comparações entre esses métodos deve-se levar em conta todo os parâmetros do procedimento envolvido, pois somente assim é possível se chegar a uma avaliação condizente. Com este relatório pretende-se entender e pesquisar sobre algoritmos de Random Forest e Random Regression. Em seguida avaliar os detalhes da implementação desses, e verificar em quais situações são passíveis de aplicação.

Introdução

Inteligência Artificial é um campo de estudo que tem por objetivo representar entidades do mundo real através de modelos computacionais. Sua aplicabilidade é ampla, uma vez que possui aplicabilidade nos mais diversos ramos da Engenharia, Biologia, Estudos Sociais, e principalmente Ciência da Computação, abrangendo subcampos de aprendizagem, simulação, percepção entre outros, até problemas específicos, como a solução de um jogo, desenvolvimento de equações matemáticas e diagnósticos médicos (RUSSELL; NORVIG, 2016).

Um dos ramos de Inteligência Artificial é Machine Learning (em português Aprendizado de Máquina), que consiste em programar um computador para que este aprenda a partir de um conjunto de entradas, de um modo geral consiste em transformar experiência em conhecimento dentro de uma máquina, ou seja trata-se de analisar dados. Dentro desse ramo muitos algoritmos são utilizados tais como: Árvores de Decisão, Regressão Linear e Logística, Clustering e combinações entre eles de forma a produzir um modelo fiel aos dados. (SHALEV-SHWARTZ; BEN-DAVID, 2014)

Há uma grande necessidade em Machine Learning em se determinar um algoritmo que se adeque ao problema. Até então o melhor jeito de se avaliar se um algoritmo é viável para um dado problema é realizando testes e comparando os resultados com outros métodos.

Dois métodos muito utilizados em Machine Learning são Random Forest Regression e Regressão Linear, esses métodos são considerados bons porém em circunstâncias bem definidas. Random Regression é um método proposto que tem como ideia associar conceitos de ambos os métodos. O objetivo desse trabalho é então analisar as qualidades e os defeitos desse método, comparando-o com Random Forest.

1 Referencial Teórico

1.1 Inteligência Artificial

É o ramo da Ciência que tenta ensinar a um computador conhecimento inerente ao aprendizado de um dado conjunto de informações, logo é o hardware e software com o objetivo de produzir informações talqualmente o ser humano.

Historicamente o conceito de inteligência está relacionado a saber fazer uma escolha (do latim INTER: “entre” e LEGERE: “escolher”) (RUSSELL; NORVIG, 2016). Essas escolhas são feitas com base em experiências vividas, entretanto para uma máquina essa experiência provém dos padrões existentes num grupo de dados, correlação das variáveis com o problema, entre outros.

1.1.1 Machine Learning

Machine Learning é a programação de um computador para que este aprenda com base em conjuntos de dados, esse processo ocorre, segundo Mitchell et al. (1997), da seguinte forma: Um programa de computador aprende com a experiência E em relação a alguma tarefa T e alguma medida de desempenho P, se seu desempenho em T, medido por P, melhora com a experiência em E.

1.2 Regressão Linear

Regressão linear é um algoritmo de Machine Learning que gera como modelo uma reta que mais se aproxima dos dados. O objetivo desse método é minimizar o RMSE (Root Mean Square Error) dos valores preditos para os valores reais. Esse método calcula coeficientes para cada feature de uma instância dos dados. O resultado predito é o produto escalar desse vetor de coeficientes com a tupla acrescido do intercept, também calculado no método, e é considerado no método como estimativa base.

1.3 Random Forest

Nesse algoritmo são geradas um conjunto (ensemble) de árvores sejam elas de decisão ou regressão conforme necessidade do problema. A predição se baseia nas avaliações individuais de cada árvore, com base na média, maior ocorrência e pesos das árvores (Figura 1). Os hiper-parâmetros desse método são: profundidade das árvores, tamanho da floresta, quantidade de features por árvores (BREIMAN, 2001).

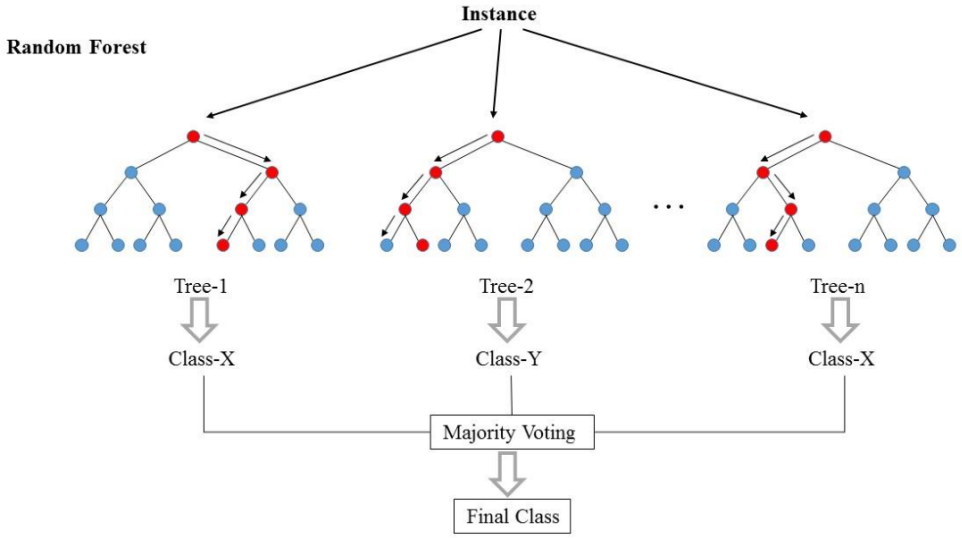

Figura 1 – Esquematização Random Forest

Os modelos de árvore de decisão ou regressão consistem em fazer subdivisões nos conjuntos de dados com base em características semelhantes. Os grupos gerados por essas subdivisões possuem informações em comum e possivelmente pertencem a uma mesma classe (classificação), ou possuem um variação linear semelhante (regressão). Essas subdivisões acontecem nos nós da árvores com base em alguma métrica que tenha por objetivo maximizar a divisão desses grupos da forma mais simples.

Quanto maior a quantidade de nós e profundidade da árvore mais o árvore gerada é específica para o conjunto de dados de treino, dessa forma não foi capaz de generalizar as classes e caiu em overfitting. Uma forma de evitar esse problema é limitar o tamanho e profundidades de uma árvore.

1.4 Random Regression

Random Regression semelhante ao Random Forest Regression calcula uma Regressão para conjuntos de instâncias aleatórios, entretanto diferentemente desse não calcula para grupos de dados em regiões semelhantes, e sim calcula uma reta com todos os valores sorteados.

O valor predito então é a média de todas as Regressões calculadas. Essa média de todas as Regressões acaba se tornando uma Regressão única global, em sua grande maioria diferente de uma única Regressão visto que 66,67% é sorteado e portanto há repetições para cada Regressão, e dessa forma algumas instâncias serão calculadas mais vezes e influenciarão no resultado final.

Esse método embora gere uma Regressão Linear ao final, dá maior peso para conjuntos semelhantes de instâncias, uma vez que são maior parte do conjunto de dados,

entretanto é ruim para instâncias distantes desses valores.

1.5 Cross-Validation

Cross-Validation é uma técnica de Machine Learning que consiste em subdividir os dados em partes com o objetivo de testar o método sem utilizar as instâncias de treino. Com base no conjunto de treino algum algoritmo gera um modelo, para ser avaliado, testado e ter seus parâmetros ajustados com os dados de teste e validação.

Uma forma de aplicar Cross-Validation é definir uma porcentagem dos dados para treino e o restante para testes.Uma outra abordagem consiste em separar os dados em Folds, que contenham o mesmo número de instâncias sorteadas aleatoriamente dos dados. Então aplica-se o teste de predições em um Fold por vez, e para esse caso o conjunto de treino fica sendo os demais Folds. Os resultados obtidos consideram uma média dos resultados de cada Fold.

2 Metodologia Experimental

Com o intuito de realizar a comparação dos métodos de Random Forest e Random Regression realizou-se a implementação dos mesmos conforme explicitado nos próximos parágrafos.

Para avaliar a predição dos métodos foi utilizados os dados de diabetes disponibilizados pela biblioteca do scikit-learn. De forma que ambos os métodos utilizassem as mesmas instâncias dos dados, foi aplicado um shuffle (embaralhar) assim que a base de dados foi lida. Esse embaralhamento tem por objetivo evitar que o treino dos algoritmos seja realizado em cima de conjuntos de tuplas semelhantes evitando assim overfitting, portanto tornando o treino menos enviesado.

Em seguida aplicando a técnica de Cross-Validation dividimos o conjunto de dados em 5 Folds. Foi então aplicado o teste de predições em um Fold por vez, enquanto o conjunto de treino ficava sendo os demais 4 Folds. Os resultados obtidos e que serão apresentados na seção de Resultados foram obtidos como sendo a média das avaliações individuais desses Folds.

De forma a realizar uma comparação mais representativa dos métodos analisou-se os parâmetros de forma equivalente: O número de features utilizadas por ambos foi igualado, o "estado Aleatório"foi definido com a mesma seed para ambos, o valor utilizado foi 9001 sorteado ao acaso. E ambos os métodos foram analisados sempre com os mesmos números de estimadores, árvores de Regressão para o Random Forest, e Regressão Linear em si para o Random Regression.

Após a validação e teste das funções implementadas, para verificar a corretude das mesmas, evitando assim privilegiar um método, foram realizados os testes e os resultados obtidos serão mostrados na seção de resultados.

3 Resultados

3.1 Avaliação do número de features

Foi testado 30%, 50%, 70% e 100% do conjunto de features do problema, mantendo constante o número de estimadores (Regressões ou Árvores de Regressão) em 100 (Figura 2), 200 (Figura 3), 300 (Figura 4), 500 (Figura 5) e 1000 (Figura 6). Com base nesses gráficos é possível ver que o aumento das features tem um impacto positivo no RMSE quando se trata de Random Forest pois, conforme os gráficos, o RMSE decresce ou se mantém constante com esse aumento. Embora nesse caso também é possível notar convergência do RMSE com porcentagens de 70% a 100% de features. Isso ocorre pois a correlação de algumas features com os valores dos labels é pequena, portanto essa features influenciam pouco a variação dos labels e considerá-las não altera significativamente os dados.

O impacto do uso dessas features fica evidente com o crescimento do RMSE para o algoritmo de Random Regression. Ao tentar predizer uma instância com base em um conjunto não representativo de features o RMSE erro aumenta signficativamente. Isso se mostra claramente com o crescimento aproximadamente linear do número de features utilizadas e o valor de RMSE obtido.

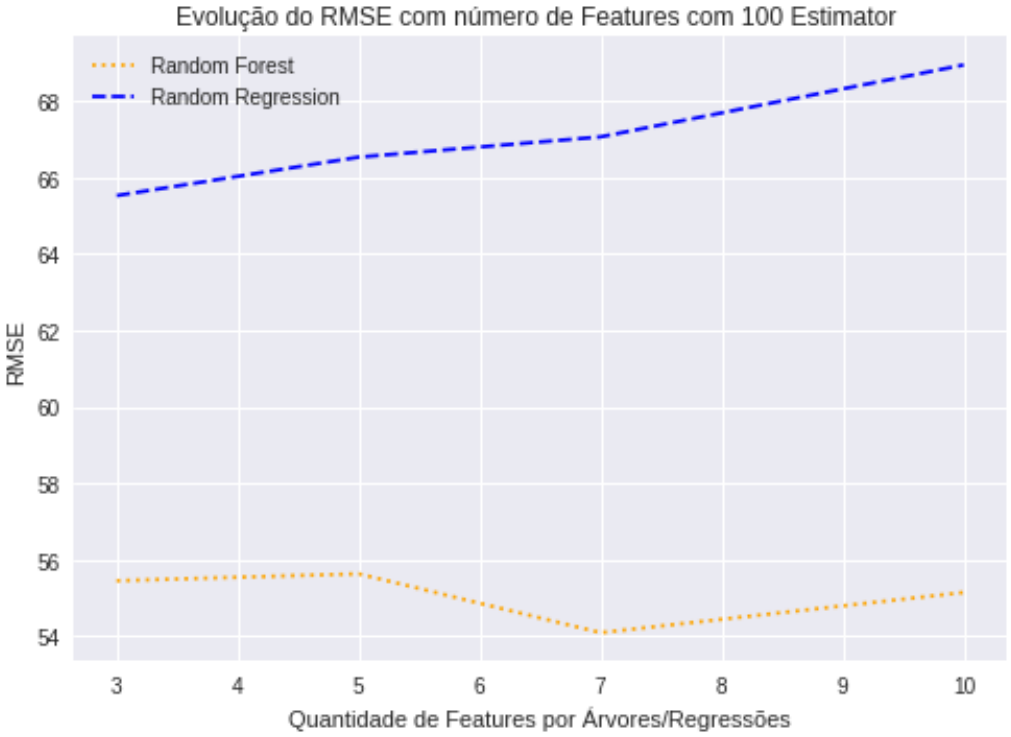

Figura 2 – Evolução do RMSE para os algoritmos de RF e RR com 100 estimadores

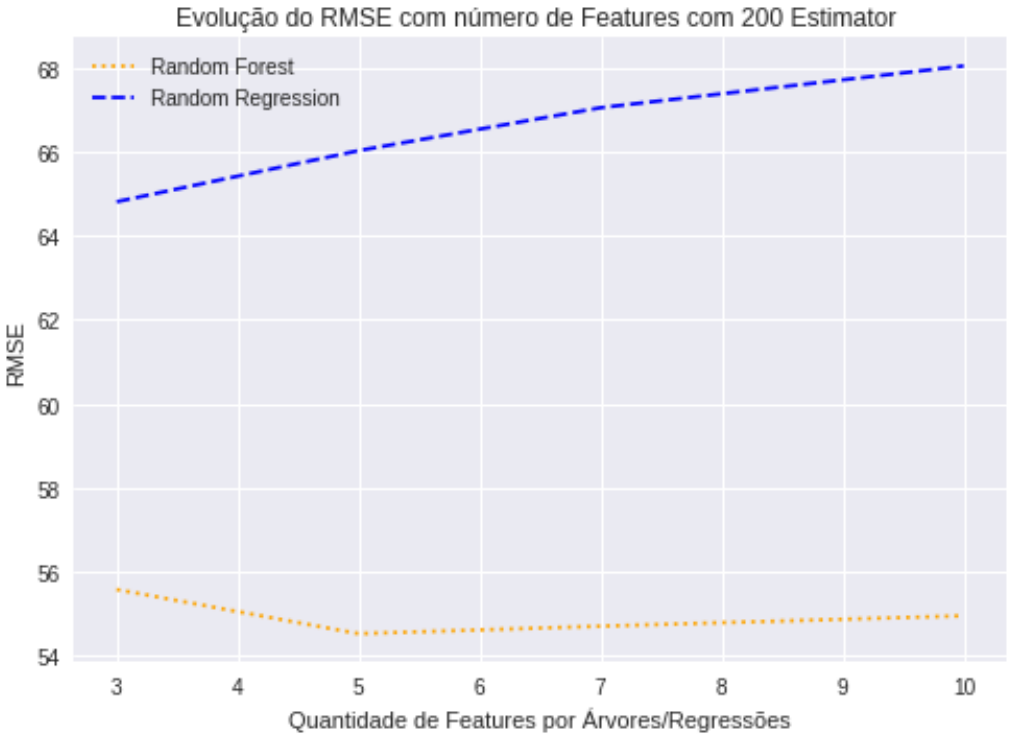

Figura 3 – Evolução do RMSE para os algoritmos de RF e RR com 200 estimadores

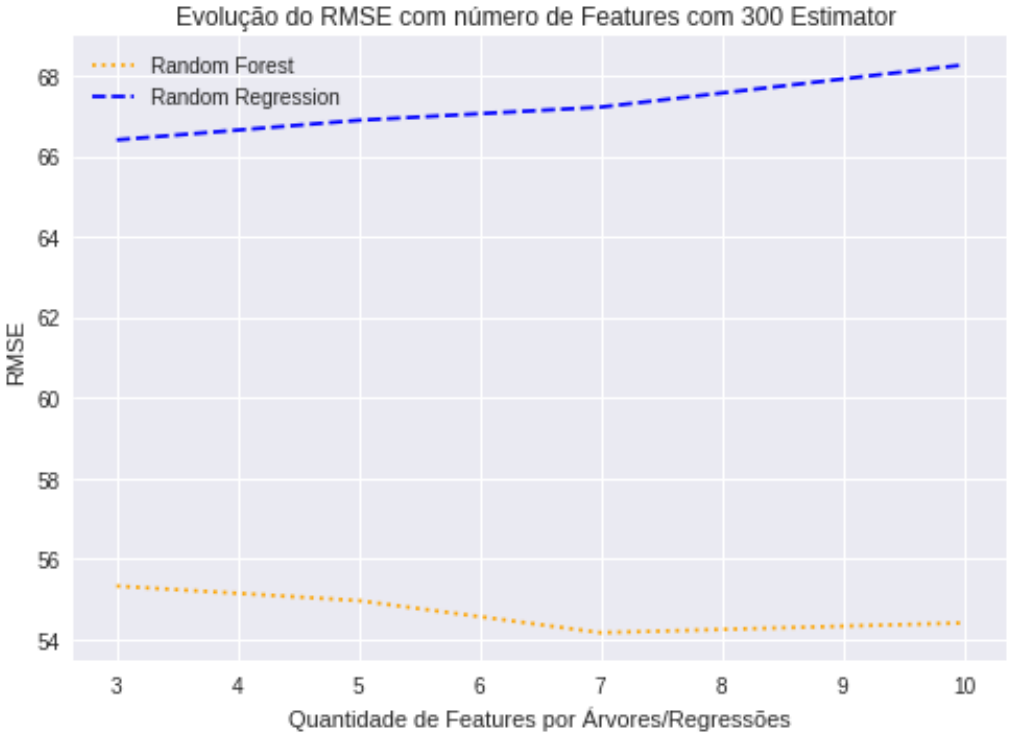

Figura 4 – Evolução do RMSE para os algoritmos de RF e RR com 300 estimadores

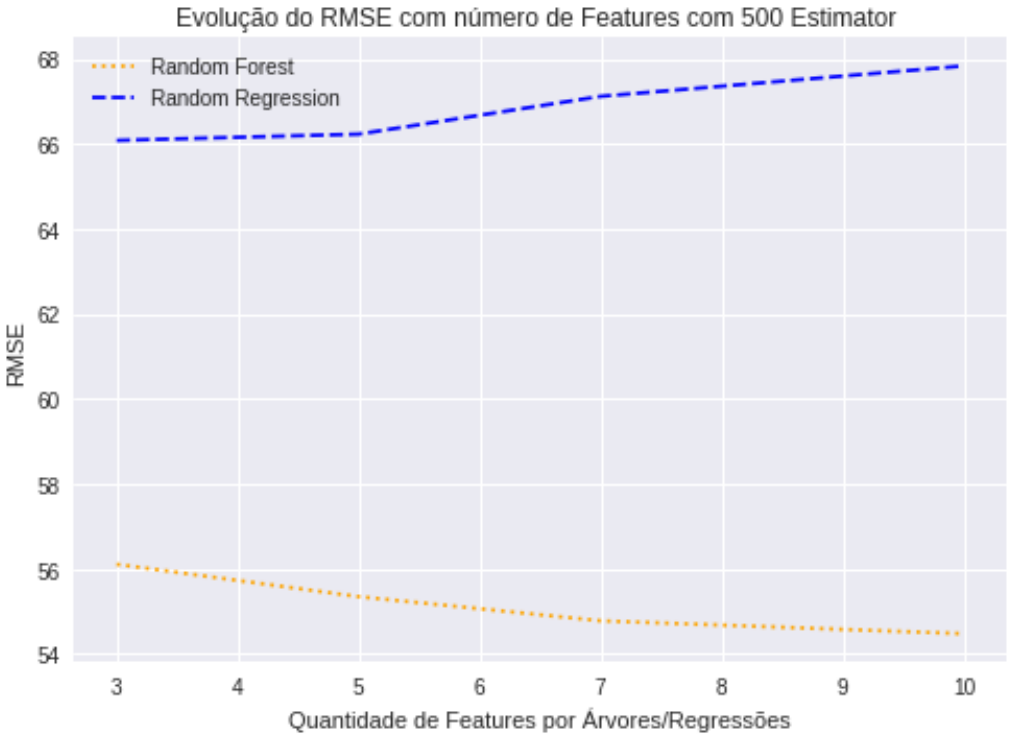

Figura 5 – Evolução do RMSE para os algoritmos de RF e RR com 500 estimadores

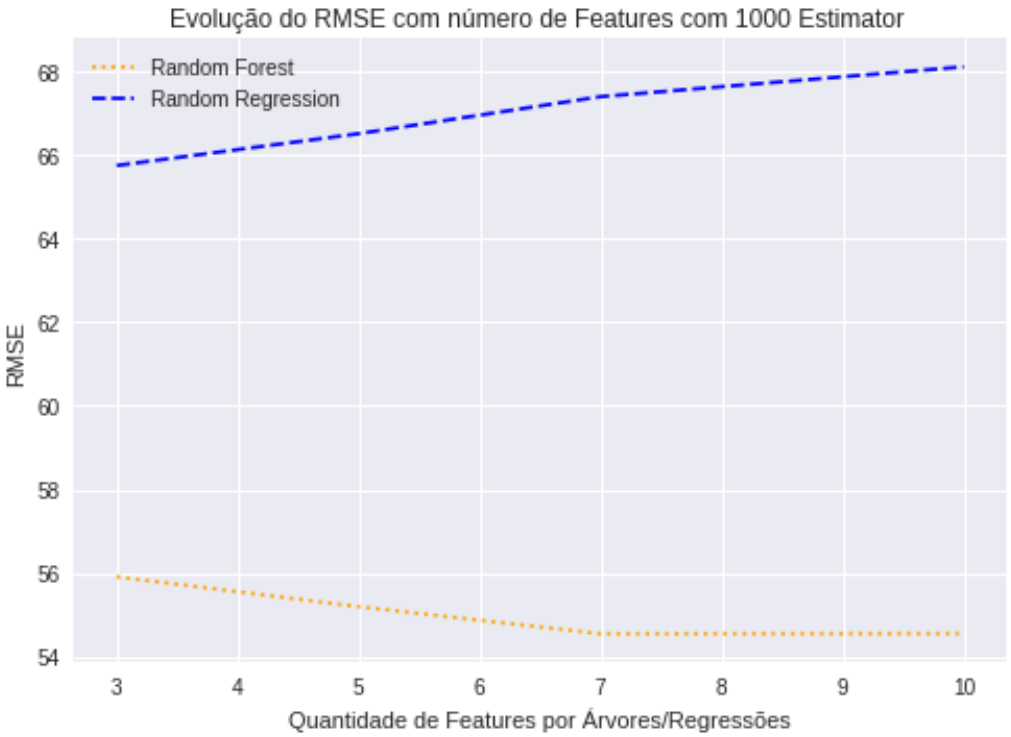

Figura 6 – Evolução do RMSE para os algoritmos de RF e RR com 1000 estimadores

3.2 Avaliação do número de Estimadores

Foi testado os métodos de Random Regression e Random Forest avaliando o impacto da quantidade de estimadores (Regressões ou Árvores de Regressão respectivamente),

mantendo constante a porcentagem de features em 30% (Figura 7), 50% (Figura 8), 70% (Figura 9) e 100% (Figura 10). Com base nesses gráficos é possível ver que o aumento do número de estimadores é pouco relevante quando acima de 400 pois, conforme mostram os gráficos, não alterou significativamente o valor de RMSE nos testes. Entretanto fica evidente o impacto quando são poucos os estimadores, devido as grandes alterações que acontecem até 200 estimadores, em seguida já se aproximam da convergência.

Esses resultados mostram que para se ter boas predições utilizando esses métodos é necessário verificar o parâmetro quantidade de estimadores. Embora o número de acertos cresça com o aumento de estimadores o número de erros também aumenta, mantendo o RMSE aproximadamente constante. Já com pouco número de estimadores esses métodos não chegaram a boas generalizações. Nesse caso encontrar a quantidade de regressões ideal pode significar melhoras de performance.

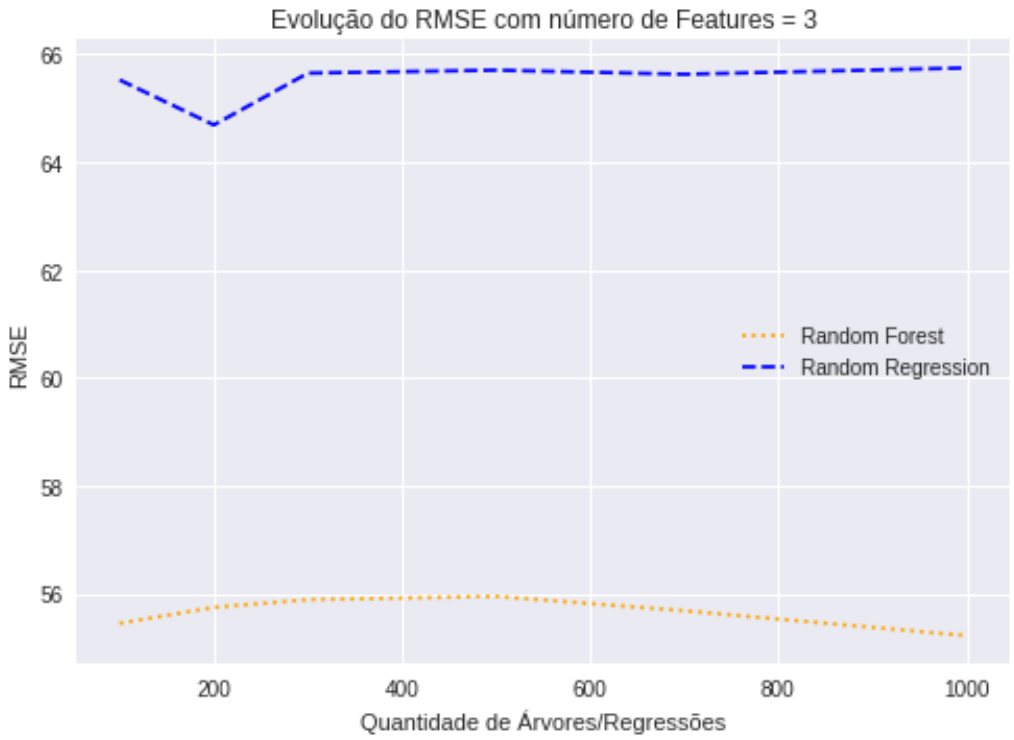

Figura 7 – Evolução do RMSE para os algoritmos de RF e RR com 30% features por estimador

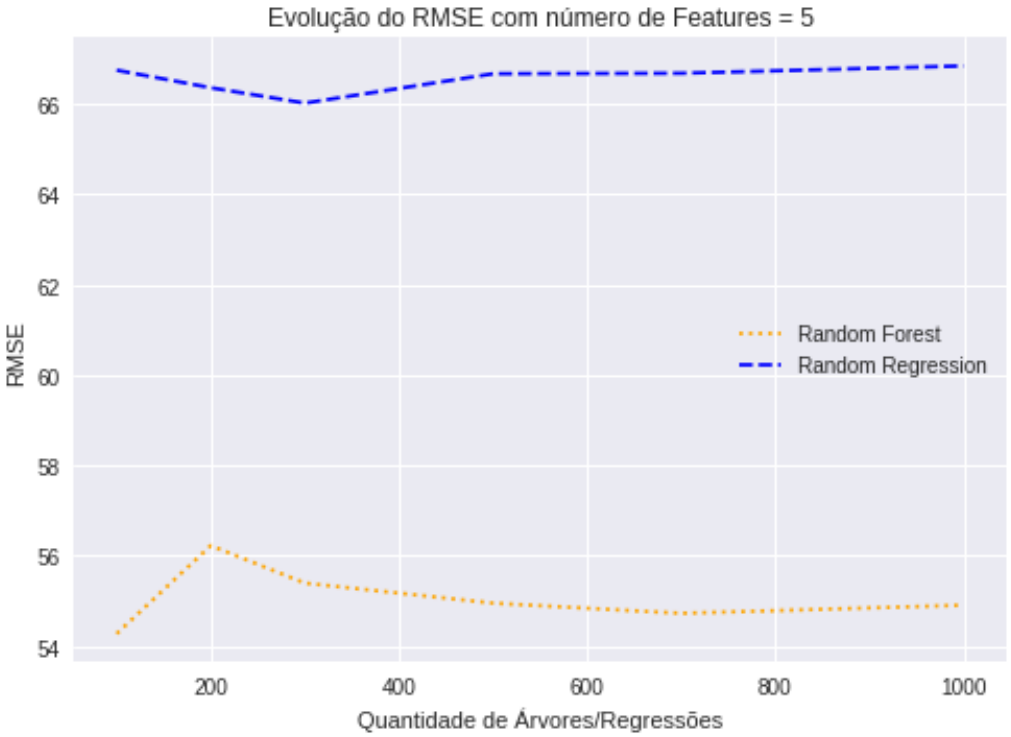

Figura 8 – Evolução do RMSE para os algoritmos de RF e RR com 50% features por estimador

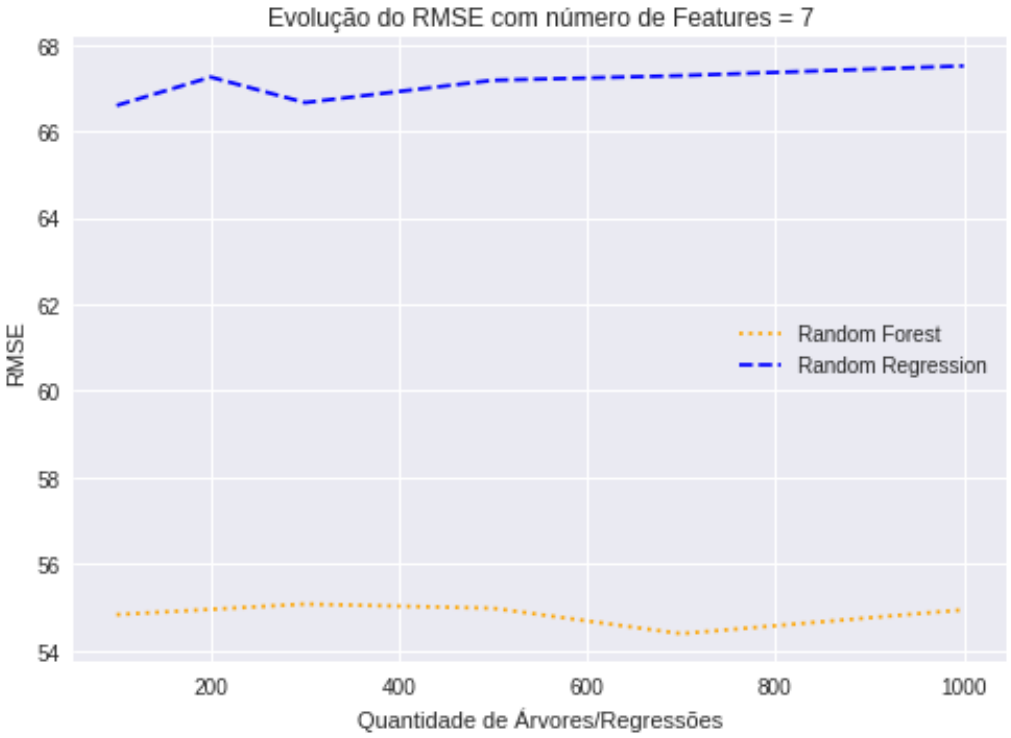

Figura 9 – Evolução do RMSE para os algoritmos de RF e RR com 70% features por estimador

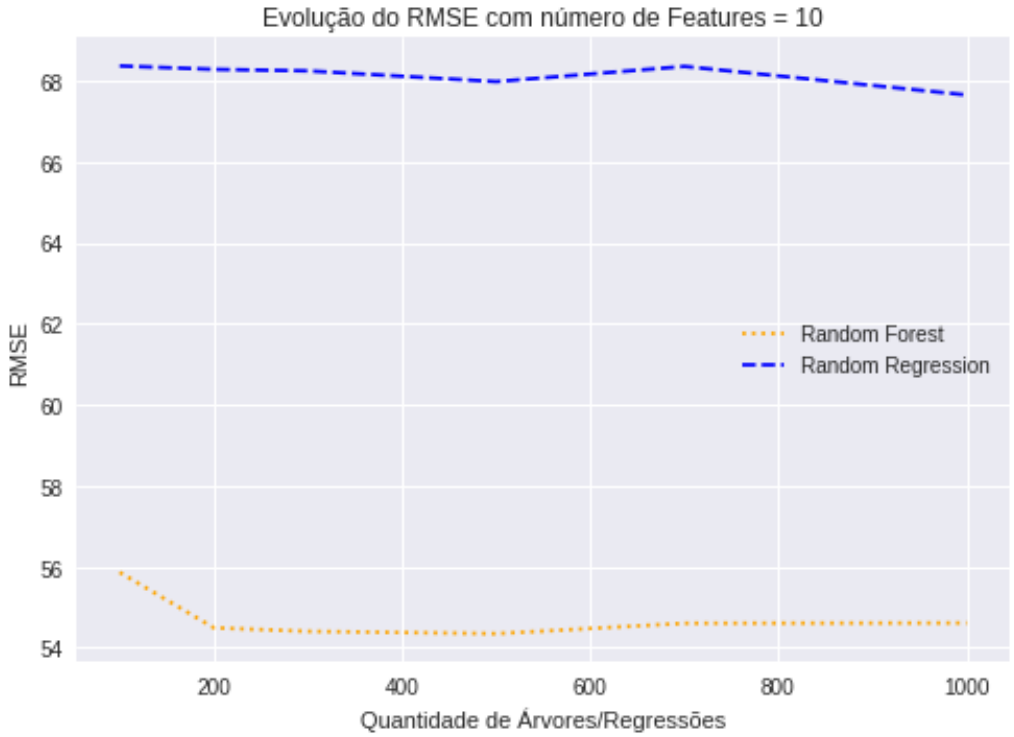

Figura 10 – Evolução do RMSE para os algoritmos de RF e RR com 100% features por estimador

4 Discussões

Uma hipótese para que o Random Regression tenha sido pior no geral para os testes nesse conjunto de dados é que esses dados podem não ter um comportamento linear. Nesse caso as subdivisões do Random Forest em grupos que possuem características semelhantes foi melhor. Ou seja em sub-regiões os dados se comportam linearmente, porém espalhados pelo conjunto de dados, como o Random Regression avalia, não se adequam a esse comportamento.

Na Figura 11 pode-se ver uma comparação entre uma Regressão Linear global, semelhante ao Random Regression pois esse é a media de Regressões de valores sorteados aleatoriamente em todo o cojunto de treino, e várias Regressões Locais representando o Random Forest, que define comportamentos lineares localmente e por isso se assemelha mais aos dados.

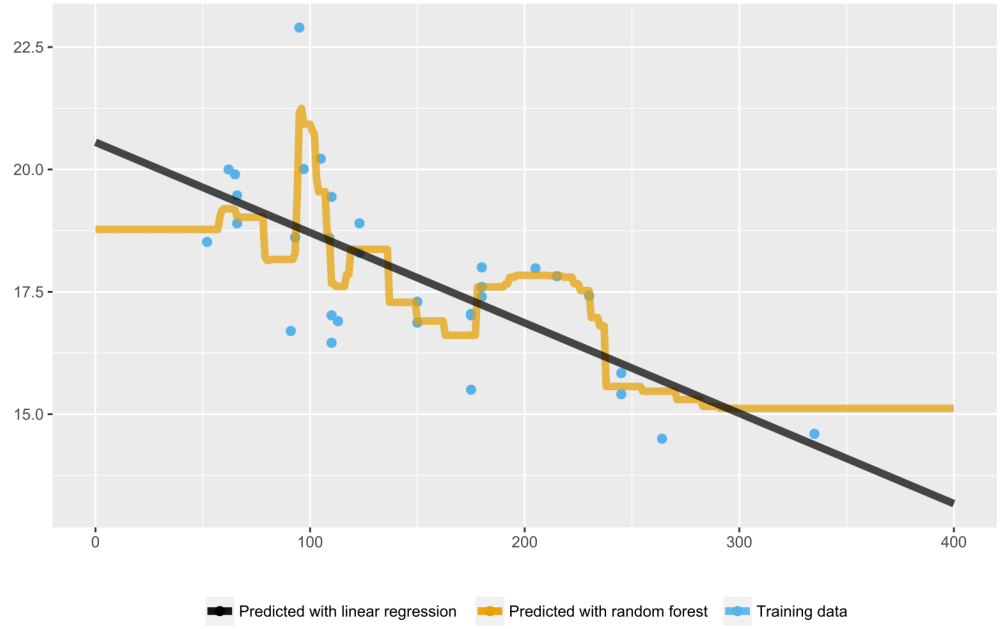

Figura 11 – Comparação Random Forest e Regressão Linear

5 Conclusão

Conforme visto nos resultados o algoritmo de Random Forest foi melhor para esse conjunto de dados devido ao caráter linear em locais específicos do conjunto de treino. Embora tenha sido melhor que o Random Regression, o conjunto de dados foi melhor para o modelo de Random Forest, nesse caso não é possível generalizar e dizer que Random Regression sempre terá resultados piores que o outro, caso o conjunto de dados fosse mais próximo de um linear globalmente o Random Regression teria sido mais efetivo, nesse caso em futuros testes pode-se comparar esses métodos utilizando bases de dados com modelos já bem determinados com o intuito de verificar essa informação.

Foi possível com a realização desses testes comparar a diferença entre modelos não lineares (Random Forest) e lineares (Random Forest), esse é um problema mais comum uma vez que os problemas do mundo real possuem muitos parâmetros e sofrem influência de formas bem características. Entretanto modelos lineares são simples e podem servir como base de comparação para os demais métodos.

Referências

BREIMAN, L. Random forests. *Machine learning*, Springer, v. 45, n. 1, p. 5–32, 2001. Citado na página 6.

MITCHELL, T. M. et al. Machine learning. 1997. *Burr Ridge, IL: McGraw Hill*, v. 45, n. 37, p. 870–877, 1997. Citado na página 6.

RUSSELL, S. J.; NORVIG, P. *Artificial intelligence: a modern approach*. [S.l.]: Malaysia; Pearson Education Limited,, 2016. Citado 2 vezes nas páginas 5 e 6.

SHALEV-SHWARTZ, S.; BEN-DAVID, S. *Understanding machine learning: From theory to algorithms*. [S.l.]: Cambridge university press, 2014. Citado na página 5.