Machine Learning Nanodegree

Projeto Final

Elias Martins Guerra Prado
20 de dezembro de 2017

Definição

Visão Geral do Projeto

As técnicas de aprendizado de máquina vem sendo cada vez mais utilizadas para resolver problemas na área das geociências, alguns exemplos de estudos incluem aqueles aplicados a geração de alvos para a exploração mineral (Cracknell et al., 2014, Merdith et al., 2015), aqueles aplicados a estudos multidisciplinares, como estudos hidrogeológicos (Cracknell et al., 2015, 2016), aqueles aplicados a dados de testemunhos de sondagem (Reading e Gallagher, 2013, Hill et al., 2015), e aqueles aplicados a identificação automatizada de rochas a partir de dados geoquímicos (Petrelli e Perugini, 2016).

Neste trabalho é proposto a avaliação da utilização dos métodos de aprendizado de máquina para discriminar a litologia (tipo de rocha) a partir de dados de química de rocha. Os dados utilizados foram extraídos do banco de dados petrológico GEOROC (http://georoc.mpch-mainz.gwdg.de/georoc/). Pretende-se avaliar o desempenho na classificação litológica dos principais algoritmos de classificação supervisionada utilizados atualmente.

Enunciação do Problema

A discriminação litológica é de suma importância para os estudos geológicos de uma região. Esta discriminação normalmente é feita a partir da descrição dos minerais e estruturas presentes nas rochas ou a partir de diagramas geoquímicos específicos. Entretanto, estes métodos convencionais podem proporcionar classificações errôneas do litotipo, pois possuem algumas ambiguidades na classificação. Além disso, a descrição destas rochas é um processo demorado e sujeito a erros técnicos. Dessa forma a utilização dos métodos de aprendizado de máquina para classificar o litotipo a partir de dados geoquímicos, podem fornecer uma solução para a automatização deste processo, podendo também ser utilizado para investigar erros de classificação em bancos de dados preenchidos manualmente.

O algoritmo será desenvolvido para executar uma tarefa de classificação *multi-class*, recebendo como entrada dados estruturados, contendo as análises químicas de amostras de rocha, e retornando como saída o nome da rocha com a maior probabilidade (similaridade) dentre os vários nomes possíveis.

As tarefas envolvidas no projeto foram as seguintes:

1. Fazer o download dos dados de geoquímica do site GEOROC.
2. Processar os dados

3. Treinar os classificadores mais utilizados atualmente para resolução de problemas de classificação supervisionada *muti-class*, para determinar o tipo de rocha (nome da rocha).
4. Avaliar os classificadores treinados e determinar qual classificador obteve o melhor desempenho para resolução deste problema.

Os hiperparametros dos classificadores foram selecionados utilizando GridSearch com validação cruzada de 3 *folds*. Os hiperparametros que obtiveram a melhor acurácia foram selecionados para o modelo final.

Após a seleção dos hiperparametros, os classificadores foram treinados, e a acurácia de classificação nos dados de teste foi computada. Os três classificadores com as melhores acurácias foram utilizados para treinar um modelo de Stacking com validação cruzada (StackingCV), que foi utilizado como o modelo final para classificação das amostras.

Os valores do atributo classe, originalmente como *string*, foram transformados em valores numéricos inteiros para o treinamento.

Métricas de Avaliação

Neste trabalho será utilizado a mesma métrica de avaliação utilizada no modelo benchmark, a acurácia que consiste em dividir o número de amostras classificadas corretamente (positivos verdadeiros + negativos verdadeiros) pelo total de amostras na população de teste. Quando a pontuação é igual a 1, todas as amostras foram classificadas corretamente pelo modelo. Quando a pontuação é igual a zero, nem uma amostra foi classificada corretamente pelo modelo.

$$acurácia = \frac{positivos\ verdadeiros + negativos\ verdadeiros}{total\ de\ amostras}$$

Esta métrica foi utilizada para a avaliação do modelo pois tanto positivos falsos quanto negativos falsos prejudicam a aplicação real do modelo, pois levam a uma classificação errada da amostra.

Análise

Exploração de Dados

O banco de dados petrológicos GEOROC (http://georoc.mpch-mainz.gwdg.de/georoc/) contém mais de um milhão análises de geoquímica de rochas vulcânicas e de xenólitos do manto. Estes dados são extraídos de publicações cientificas, e contém análises dos elementos maiores e traços, análises isótopos radiogênicos e não radiogênicos, assim como dados analíticos das idades destas rochas. As amostras pertencem a 11 regiões com configurações geológicas distintas.

Para este projeto apenas uma parte do banco de dados foi utilizada. Foram selecionadas as análises de rochas localizadas em cratons arqueanos, sendo eles: Escudo Aldan, Craton Amazônico, Escudo Báltico, Craton Bastar, Província Churchill, Craton Dharwar, Craton Gawler, Craton Kaapvaal, Craton do Atlântico Norte, Craton

do Norte da China, Craton Rae, Craton São Francisco, Craton Sarmatian, Craton Siberiano, Craton Singhbhum, Provincia Slave, Província Superior, Craton da Tanzânia, Escudo Ucraniano, Craton do Oeste Africano, Craton do Oeste Australiano e Craton do Zimbabwe. Os dados selecionados totalizam 22.232 análises.

Os dados incluem os seguintes campos:

- "**CITATIONS; TECTONIC SETTINGS**": citações e ambiente tectônico. (*string*)
- "**LOCATION; LOCATION COMMENT; LATITUDE MIN; LATITUDE MAX; LONGITUDE MIN; LONGITUDE MAX; LAND OR SEA; ELEVATION MIN; ELEVATION MAX"**: localização geográfica, latitude, longitude e altitude. (*float/string*)
- "**SAMPLE NAME**": nome da amostra. (*string*)
- ❖ "**ROCK NAME**": classificação do nome de rocha, *atributo classe.* (*string*)
- "**MIN. AGE; MAX. AGE; GEOL.; AGE; ERUPTION DAY; ERUPTION MONTH; ERUPTION YEAR**": idade da amostra. (*string*)
- "**ROCK TEXTURE; ROCK TYPE**": textura e tipo de rocha. (*string*)
- "**DRILL DEPTH MIN; DRILL DEPTH MATH**": profundidade da perfuração de onde a amostra foi coletada. (*float*)
- "**ALTERATION**": grau de alteração da rocha, oxidação/intemperismo. (*string*)
- "**MATERIAL**": material analisado. No caso todas as análises são de rocha total, WR. (*string*)
- ❖ "**SIO2(WT%); TIO2(WT%); B2O3(WT%); AL2O3(WT%); CR2O3(WT%); FE2O3(WT%); FEO(WT%); FEOT(WT%); CAO(WT%); MGO(WT%); MNO(WT%); NIO(WT%); K2O(WT%); NA2O(WT%); P2O5(WT%); H2O(WT%); H2OP(WT%); H2OM(WT%); H2OT(WT%); CO2(WT%); CO1(WT%); F(WT%); CL(WT%); CL2(WT%); OH(WT%); CH4(WT%); SO2(WT%); SO3(WT%); SO4(WT%); S(WT%); LOI(WT%); VOLATILES(WT%); O(WT%); OTHERS(WT%); HE(CCM/G); HE(CCMSTP/G); HE3(CCMSTP/G); HE3(AT/G); HE4(CCM/G); HE4(CCMSTP/G); HE4(AT/G); HE4(MOLE/G); HE4(NCC/G); HE(NCC/G); LI(PPM); BE(PPM); B(PPM); C(PPM); CO2(PPM); F(PPM); NA(PPM); MG(PPM); AL(PPM); P(PPM); S(PPM); CL(PPM); K(PPM); CA(PPM); SC(PPM); TI(PPM); V(PPM); CR(PPM); MN(PPM); FE(PPM); CO(PPM); NI(PPM); CU(PPM); ZN(PPM); GA(PPM); GE(PPM); AS(PPM); SE(PPM); BR(PPM); RB(PPM); SR(PPM); Y(PPM); ZR(PPM); NB(PPM); MO(PPM); RU(PPM); RH(PPM); PD(PPM); AG(PPM); CD(PPM); IN(PPM); SN(PPM); SB(PPM); TE(PPM); I(PPM); CS(PPM); BA(PPM); LA(PPM); CE(PPM); PR(PPM); ND(PPM); SM(PPM); EU(PPM); GD(PPM); TB(PPM); DY(PPM); HO(PPM); ER(PPM); TM(PPM); YB(PPM); LU(PPM); HF(PPM); TA(PPM); W(PPM); RE(PPM); OS(PPM); IR(PPM); PT(PPM); AU(PPM); HG(PPM); TL(PPM); PB(PPM); BI(PPM); TH(PPM); U(PPM)":** análises químicas dos elementos maiores e traços, *colunas utilizadas como features para a classificação*. (*float*)
- **"ND143_ND144; ND143_ND144_INI; EPSILON_ND; SR87_SR86; SR87_SR86_INI; PB206_PB204; PB206_PB204_INI; PB207_PB204; PB207_PB204_INI; PB208_PB204; PB208_PB204_INI; OS184_OS188; OS186_OS188; OS187_OS186; OS187_OS188; RE187_OS186;**

RE187_OS188; HF176_HF177; HE3_HE4; HE3_HE4(R/R(A)); HE4_HE3; HE4_HE3(R/R(A)); K40_AR40; AR40_K40": análises de isótopos radiogênicos e não radiogênicos. (*float*)

Como destacado acima, apenas as colunas contendo as análises químicas dos elementos foram utilizadas como atributos para o treinamento do modelo. Além disso, antes do treinamento os dados foram filtrados, de maneira a remover amostras (linhas) que poderiam influenciar negativamente o modelo, como por exemplo amostras de rocha alteradas, amostras de rochas de tipos diferentes (foram selecionadas apenas rochas vulcânicas), amostras com analises de poucos elementos (poucas *features*) e amostras com poucas ocorrências do atributo classe (menos de 10 amostras para o mesmo nome de rocha). As colunas contendo as análises químicas também foram filtradas. As colunas de elementos com poucos resultados analíticos (elevado número de *missings*) foram removidas para o treinamento. Esta etapa de pré-processamento dos dados será abordada em maior detalhe a frente.

Os valores do atributo classe (ROCK NAME), apresentam um sufixo numérico entre colchetes que indica o número de referência do trabalho no qual foi feita a classificação do nome da rocha. Este sufixo provoca uma duplicação nos valores do atributo classe, pois rochas com o mesmo nome, apresentam valores diferentes no campo ROCK NAME, como por exemplo: BASALT [12345] e BASALT [678910]. Estes sufixos foram removidos na etapa de pré-processamento dos dados.

Visualização Exploratória dos Dados

O gráfico abaixo mostra a distribuição de ocorrências de valores únicos do atributo classe 'ROCK NAME' (Fig. 1). Como o gráfico mostra, as classes se encontram muito desbalanceadas. As classes KOMATIITE (3065 ocorrências) e BASALT (2563 ocorrências) ocorrem com muito mais frequência que as demais classes, e juntas representam mais de 25% dos dados. Também é possível observar no gráfico que a terceira classe com maior contagem de ocorrências é classe NOT GIVEN (2439 ocorrências), que representam amostras não classificadas, portanto, foram removidas.

Após o pré-processamento dos dados, foi criado um classificador Random Forest com os seguintes hiperparametros:

n_estimators	150
max_depth	8
min_samples_leaf	4
max_features	0.2
criterion	gini
min_samples_split	2

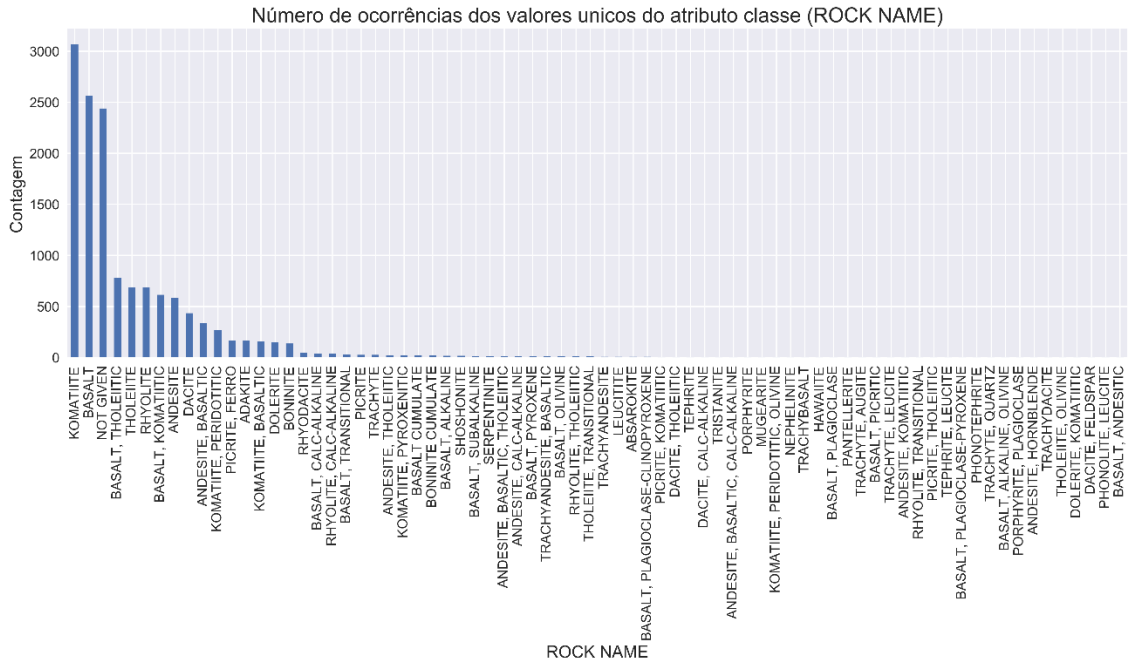

Fig. 1 *Gráfico de barras com a contagem de valores únicos do atributo classe 'ROCK NAME'.*

O classificador foi utilizado para calcular a importância de cada atributo na classificação dos dados, baseado no coeficiente de gini. A importância dos 25 atributos melhor pontuados é mostrada na figura 2. Os atributos com maior importância foram MgO(WT%), Cu(PPM), Zn(PPM), SiO_2(WT%), Co(PPM). Ou seja, estes atributos são aqueles que fornecem o maior ganho de informação para o classificador.

A figura 3 ilustra a matriz de correlação de Pearson dos atributos utilizados para o treinamento, após o pré-processamento dos dados. Os elementos terrar raras (La, Ce, Pr, Nd, Sm, Eu, Gd, Tb, Dy, Ho, Er, Tm, Yb, Lu) apresentam elevada correlação positiva, alguns com correlação acima de 0.99. Dentre estes elementos o La, Eu e o Lu foram os que apresentaram a menor correlação com os demais elementos terras raras. Outros elementos que apresentaram elevada correlação foram: Al_2O_3 com MgO e Ni (-83% e -80% respectivamente), MgO com Ni (90%), K_2O e Rb (81%).

Algoritmos e Técnicas

Os seguintes modelos de aprendizagem supervisionada foram utilizados: Support Vector Classifier (SVC), K-Nearest Neighbors Classifier (KNN), Random Forest Classifier (RF) e Xgboost.

O classificador SVC também conhecido como Support Vector Machines (SVM), classifica os dados calculando os hiperplanos que melhor separam as classes. O SVC calcula a posição e a forma desta superfície de maneira a maximizar a distância euclidiana entre a superfície e os pontos mais próximos. Essa distância entre o hiperplano e o primeiro ponto de cada classe costuma ser chamada de margem. Estes classificadores não funcionam muito bem em conjuntos de dados com muitas dimensões, ou com muita quantidade de ruídos, entretanto em conjuntos de dados menores, costumam apresentar boa performance.

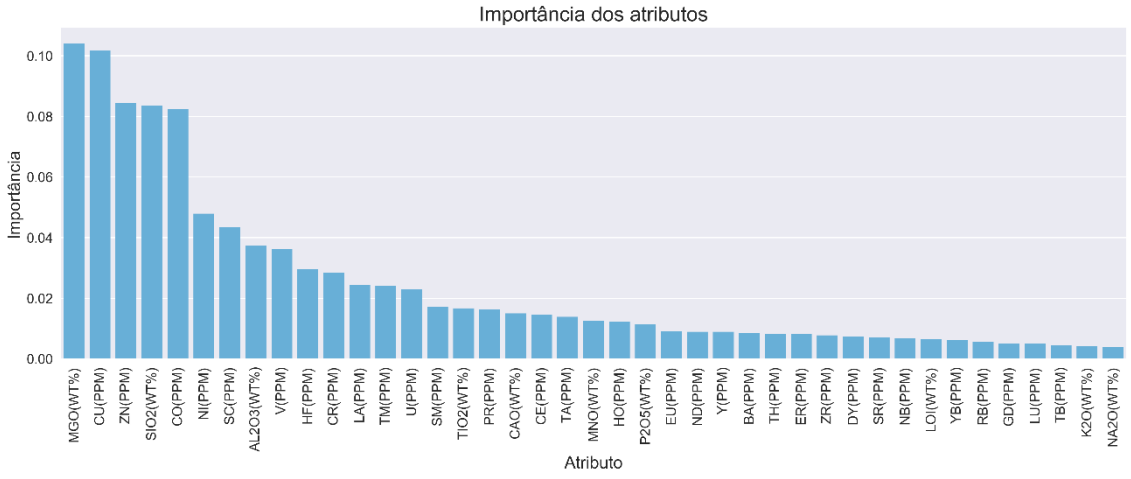

Fig. 2 *Importância dos atributos calculada a partir de um modelo Random Forest.*

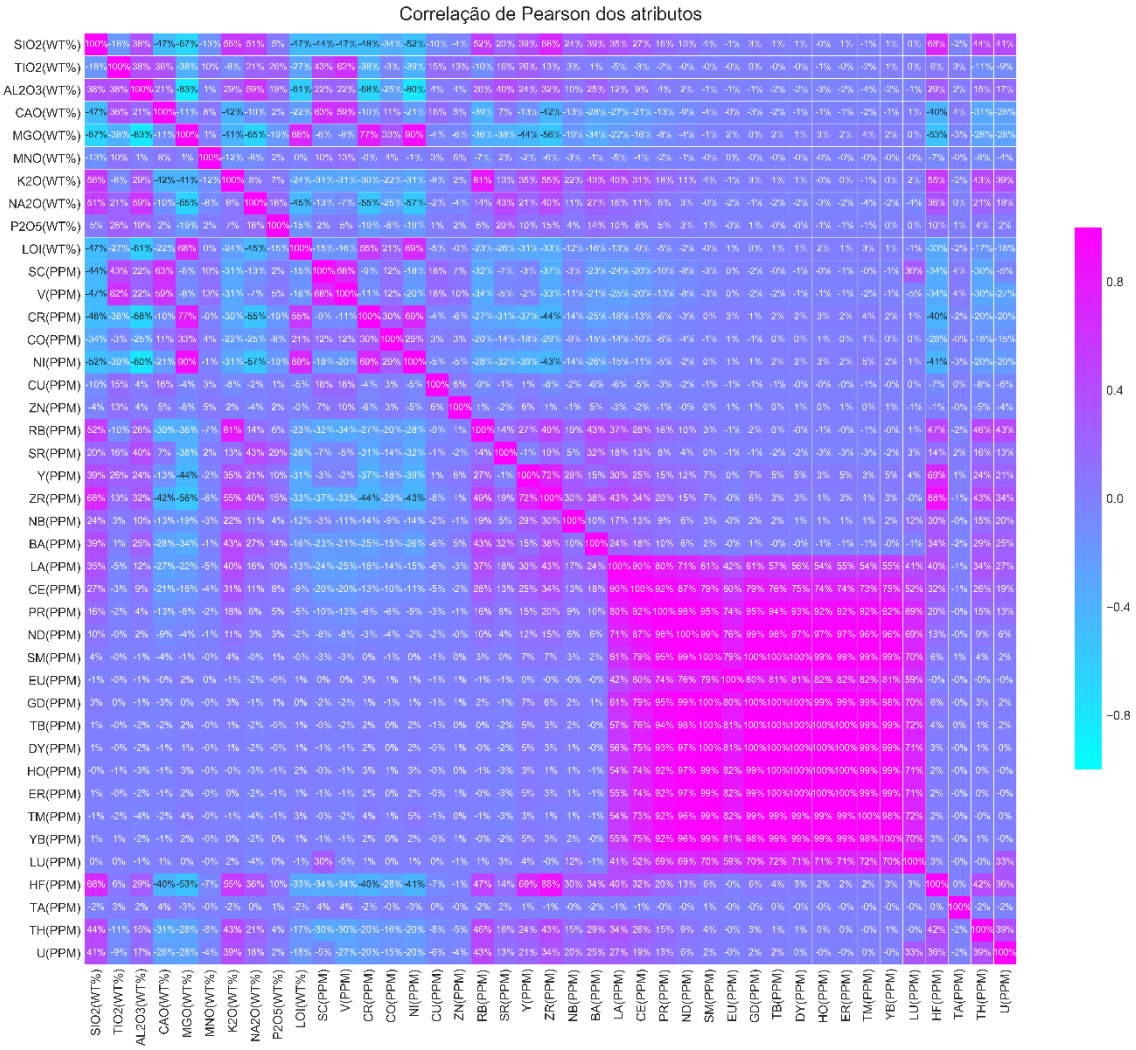

Fig. 3 *Matriz de correlação de Pearson para os atributos do modelo. Os valores de correlação se encontram em porcentagem. A direita, escala de cor das células da matriz.*

O classificador KNN é o que apresenta o menor tempo de treinamento. Este classificador é treinado apenas armazenando as instâncias dos dados de treinamento. A classificação é feita pelo algoritmo calculando os k vizinhos mais próximo do ponto a ser classificado. A proximidade é calculada a partir de uma métrica de distância que pode ser definida. A distância euclidiana é a mais utilizada. Apesar deste classificador ser rápido no treinamento, ele costuma ser lendo na predição, pois é preciso calcular a distância entre o ponto a ser classificado e todos os pontos de treinamento.

O classificador RF é um método de classificação do tipo *ensemble*, que funciona construindo múltiplas árvores de decisão (*decision trees*) durante o treinamento, e retorna durante a previsão a classe classificada com maior frequência pelas árvores construídas. O modelo RF corrige a tendência de sobreajuste das árvores de decisão. As árvores de decisão são treinadas calculando os atributos que dividem os dados de entradas em subconjuntos com o maior número de indivíduos da mesma classe (geralmente é utilizada a entropia para calcular esta separação), até que os subconjuntos gerados apresentem apenas indivíduos da mesma classe, ou quando a divisão dos subconjuntos deixa de aumentar o desempenho do classificador. Estes classificadores costumam ser relativamente rápidos no treinamento e na classificação.

O Xgboost é um classificador criado a partir do modelo RF, utilizando o método de *boosting* para melhorar a performance dos classificadores RF. O *boosting* costuma melhorar a tendência (*bias*) e a variância dos modelos de aprendizado supervisionado. A método consiste em criar classificadores fracos (classificadores com baixo desempenho na classificação) e agregar o resultado destes classificadores para gerar um classificador mais robusto. Ao agregar os resultados dos classificadores fracos, pesos proporcionais as acurácias destes classificadores são atribuídos a cada um deles. Após o classificador fraco ser adicionado ao modelo final os pontos de treinamento são recalculados, atribuindo novos pesos aos mesmos, de maneira que o peso dos pontos classificados corretamente diminuí e o peso dos pontos classificados erroneamente aumenta. Dessa maneira, os novos classificadores fracos gerados irão focar nos pontos que os classificadores fracos anteriores erraram. O Xgboost é um dos classificadores mais utilizados nos dias de hoje, pois costuma apresentar desempenho superior aos demais classificadores. Entretanto o treinamento e a classificação neste modelo geralmente é lenta.

Modelo de Benchmark

O modelo deste trabalho é baseado no artigo de Petrelli, 2016. Neste artigo o autor utiliza os dados do banco GEOROC e PetDB para treinar um Support Vector Classifier (SVC) na classificação automatizada do ambiente geotectônico da rocha. Para avaliar seu modelo o autor utilizou a acurácia. A acurácia obtida no artigo para os dados de teste foi de 98%.

Métodos

Pré-processamento dos Dados

As seguintes tarefas foram realizadas durante o pré-processamento dos dados:

1. Manter apenas amostras de rochas vulcânicas nos dados (ROCK TYPE = VOL).
2. Remover amostras alteradas (ALTERATION = nan).
3. Remover sufixo numérico presente nos valores do atributo classe.
4. Remover espaços em branco no começo e final dos valores do atributo classe.
5. Remover amostras com contagem do atributo classe inferior a 10.
6. Remover amostras com valores vazios do atributo classe (ROCK NAME = NOT GIVEN e nan)
7. Manter apenas colunas com análises químicas e o atributo classe.
8. Verificar e corrigir valores vazios.
9. Verificar correlação entre as varáveis.
10. Transformar distribuição dos dados utilizando box-cox (Box e Cox, 1964), para se obter distribuições próximas à distribuição normal nas análises químicas.
11. Calcular a importância das variáveis utilizando Random Forest.
12. Transformar valores da variável alvo de string para inteiro.
13. Dividir os dados em Teste e Treino.

As rochas plutônicas apresentam química semelhante aos seus pares vulcânicos, pois possuem mesma composição, modificando apenas o processo de formação. Plutônicas são formadas pela cristalização magmática em profundidade (interior da Terra) e vulcânicas são formadas pela cristalização magmática subaérea ou subaquosa. Portanto, uma determinada rocha plutônica como o granito, apresenta a mesma composição química do riolito, seu correspondente vulcânico. Logo, as rochas plutônicas foram removidas. As rochas Metamórficas, também podem possuir química semelhante as demais rochas, pois são produto da deformação de uma determinada rocha. Ex. Se uma rocha sedimentar é deformada ela se torna uma rocha metamórfica porem sua química continua semelhante, ou muito parecida com a química de seu parente sedimentar. Portanto estas rochas também podem confundir o modelo e foram removidas. As rochas sedimentares são formadas pela erosão das rochas, portanto, podem carregar características químicas de todos os tipos de rocha, e por esse motivo também foram removidas. Os demais tipos de rocha não foram considerados pela sua ocorrência restrita na natureza e também pelo pequeno número de amostras destes tipos no banco de dados.

As amostras alteradas foram removidas do banco de dados pois podem apresentar variações químicas provocadas pelo intemperismo, que modificam a composição química original da rocha, podendo levar a erros no modelo.

Os valores do atributo classe foram processados de maneira a remover caracteres que estavam provocando duplicação dos valores, como o sufixo numérico e espaços antes e após os valores.

As amostras que apresentavam menos de 10 ocorrências do atributo classe foram removidas, pois apresentavam poucas amostras para o treinamento. Amostras sem o valor do atributo classe também foram removidas.

Para o treinamento foram consideradas apenas as colunas contendo as análises químicas as demais colunas foram removidas.

Após esta primeira etapa do pré-processamento, os dados apresentavam 11.143 linhas, com 119 atributos selecionados para o treinamento. Então, foi feita a verificação e correção dos valores vazios nos dados. As colunas de análise química que não apresentavam nem uma análise foram removidas, sendo elas: B_2O_3(WT%), FeOT(WT%), H_2OT(WT%), CO_1(WT%), Cl(WT%), Cl_2(WT%), OH(WT%), CH_4(WT%), SO_4(WT%), VOLATILES(WT%), O(WT%), OTHERS(WT%), He(CCM/G), He(CCMSTP/G), He_3(CCMSTP/G), He_3(AT/G), He (CCM/G), He_4(CCMSTP/G), He_4(AT/G), He_4(MOLE/G), He_4(NCC/G), He(NCC/G), Na(PPM), Mg(PPM), Al(PPM), Ca(PPM), Fe(PPM), I(PPM). As colunas que apresentavam menos de 1000 análises (>91% de vazios) foram removidas, sendo elas: 'NiO(WT%)', 'H_2O(WT%)', 'H_2OP(WT%)', 'H_2OM(WT%)', 'F(WT%)', 'SO_2(WT%)', 'SO_3(WT%)', 'Li(PPM)', 'Be(PPM)', 'B(PPM)', 'C(PPM)', 'CO_2(PPM)', 'F(PPM)', 'P(PPM)', 'Cl(PPM)', 'K(PPM)', 'Ti(PPM)', 'Mn(PPM)', 'Ge(PPM)', 'Se(PPM)', 'BR(PPM)', 'Mo(PPM)', 'Ru(PPM)', 'Rh(PPM)', 'Ag(PPM)', 'Cd(PPM)', 'In(PPM)', 'Sb(PPM)', 'Te(PPM)', 'Re(PPM)', 'Os(PPM)', 'Au(PPM)', 'Hg(PPM)', 'Tl(PPM)', 'Bi(PPM)'

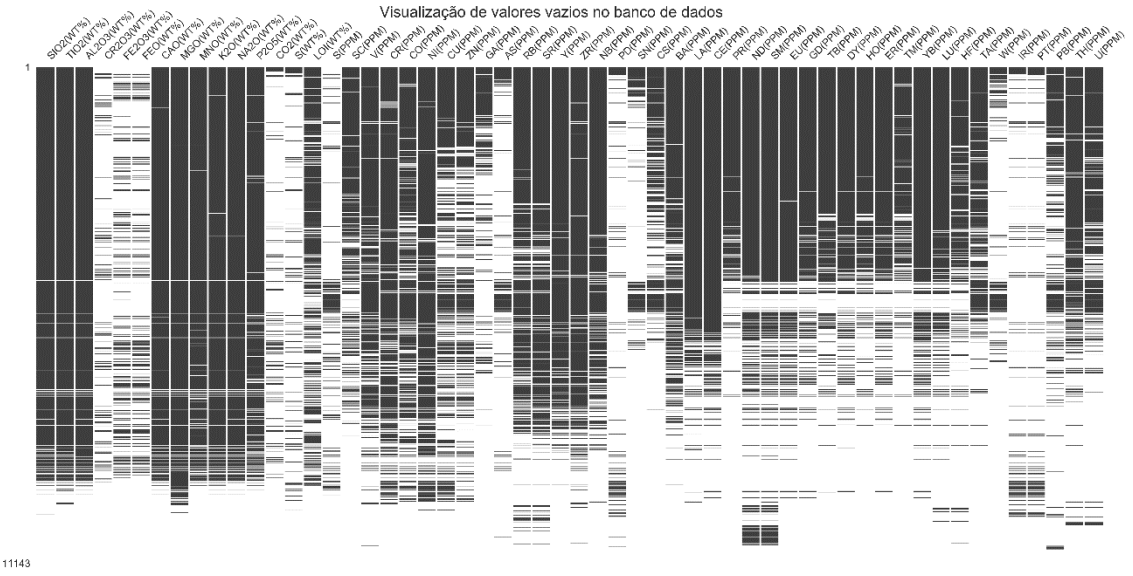

Fig. 4 *Gráfico ilustrando valores vazios no banco de dados. Linhas coloridas de preto nas colunas indicam existência de dado, e linha em branco ausência. Os dados plotados foram processados apenas até a etapa 7 (ver início desta seção).*

A figura 4 ilustra a distribuição de vazios nos dados após os procedimentos descritos acima. A partir desta figura foram selecionas as colunas remanescentes com elevada quantidade de vazios, para serem removidas. As colunas selecionadas foram: Cr_2O_3(WT%), Fe_2O_3(WT%), FeO(WT%), CO_2(WT%), S(WT%), S(PPM), Ga(PPM), As(PPM), Pd(PPM), Sn(PPM), Cs(PPM), W(PPM), Ir(PPM), Pt(PPM), Pb(PPM).

Os valores vazios restantes foram processados da seguinte forma: as amostras sem análise de SiO_2 (principal elemento constituinte das rochas) foram removidas; Amostras com mais de 30% das colunas restantes (mais 12 das 41 colunas restantes em branco) sem análise química foram removidas. Após este processamento restaram 5191 amostras e 41 colunas de análises químicas. Estas colunas restantes foram os atributos selecionados para o treinamento dos dados sendo eles: SiO_2(WT%), TiO_2(WT%), Al_2O_3(WT%), CaO(WT%), MgO(WT%), MnO(WT%), K_2O(WT%), Na_2O(WT%), P_2O_5(WT%), LOI(WT%), Sc(PPM), V(PPM), Cr(PPM), Co(PPM), Ni(PPM), Cu(PPM), Zn(PPM), Rb(PPM), Sr(PPM), Y(PPM), Zr(PPM), Nb(PPM), Ba(PPM), La(PPM), Ce(PPM), Pr(PPM), Nd(PPM), Sm(PPM), Eu(PPM), Gd(PPM), Tb(PPM), Dy(PPM), Ho(PPM), Er(PPM), Tm(PPM), Yb(PPM), Lu(PPM), Hf(PPM), Ta(PPM), Th(PPM), U(PPM). Os valores vazios restantes foram preenchidos com a média aritmética das análises do elemento para as amostras da mesma classe. Ex. os valores vazios nas análises de Al_2O_3 das amostras de granito foram preenchidos com a média das análises de Al_2O_3 dos demais granitos. Após este procedimento, as amostras que ainda apresentavam algum valor vazio (amostras que não apresentavam nem uma análise no elemento para sua classe) foram removidas (77 amostras removidas), assim como as amostras de classes com menos de 10 ocorrências foram verificadas e removidas novamente. Dessa forma após o processamento dos vazios, os dados ficaram com 5.114 linhas e 41 atributos.

A figura 5 mostra a distribuição final dos valores únicos do atributo classe, onde é possível verificar o grande desbalanceamento entre as classes ainda existente.

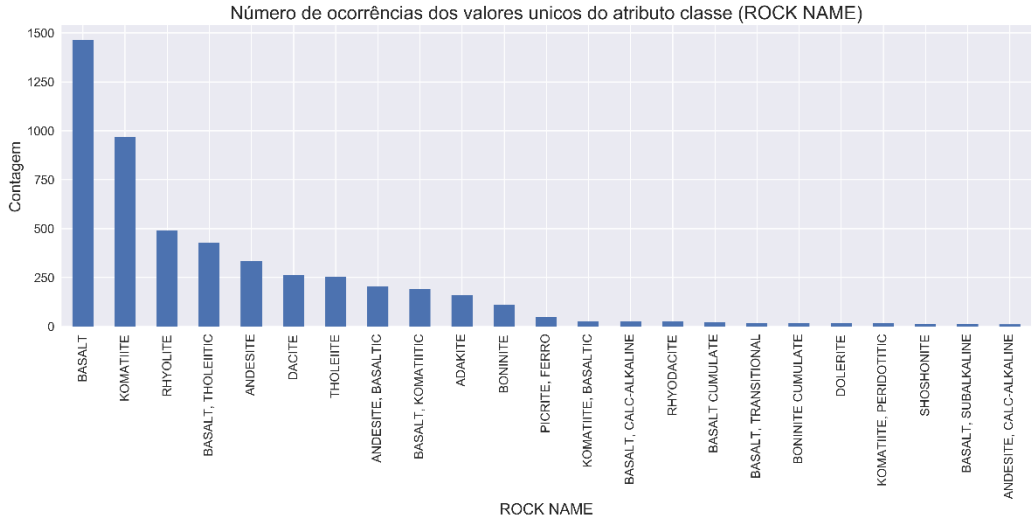

Fig. 5 *Gráfico de barras com a contagem de valores únicos do atributo classe ROCK NAME após o pré-processamento dos dados.*

A distribuição de frequência dos valores dos atributos utilizados para o treinamento possuí grande assimetria para a esquerda, como é possível visualizar na figura 6 abaixo. Esta grande diferença entre o valor da média e da moda pode prejudicar o treinamento do modelo. Os dados foram transformados utilizando o método box-cox (Box e Cox, 1964), para que a distribuição de frequência dos valores dos atributos ficassem mais próximas de uma distribuição normal, com a média próxima da moda. Os histogramas dos atributos após a transformação box-cox são mostrados na figura 7.

Encerrando o pré-processamento, os dados foram separados em teste e treino utilizando o *StritifiedShuffleSplit* do *scikitLearn*. Dessa forma, a divisão dos dados foi feita de forma estratificada, ou seja, de maneira que as fatias resultantes da divisão possuíssem o mesmo número de classes, com uma quantidade semelhante de amostras por classe. Foram separados 20% dos dados para teste e 80% para treino. Antes da divisão dos dados, os valores do atributo classe, originalmente em *string*, foram transformados para valores numéricos inteiros, sendo atribuído o valor 0 para a primeira classe e N-1 para a ultima classe onde N é o total de classes únicas nos dados.

Implementação e Refinamento

Os dados processados foram treinados em 4 classificadores diferentes, com intuito de analisar o desempenho de cada classificador na resolução do problema. Os classificadores utilizados foram: Support Vector Classifier (SVC), K-Nearest Neighbors Classifier (KNN), Random Forest Classifier (RF) e Xgboost. Esta etapa de implementação foi dividida nas seguintes etapas:

1. Definir os hiperparametros a serem configurados de cada classificador
2. Escolher os valores dos hiperparametros através de GridSearch com validação cruzada de 3 *folds*, utilizando a acurácia.
3. Calcular a acurácia de validação cruzada do modelo nos dados de treinamento com 3 *folds*.
4. Treinar o modelo nos dados de treinamento.
5. Testar o modelo nos dados de teste, calcular a acurácia e matriz de confusão do teste.

Durante o refinamento dos modelos, os seguintes hiperparametros foram testados pelo GridSearch (etapa 2 acima) para cada modelo:

KNN	
n_neighbors	3 a 20 (step=2)
weights	distance ; uniform
p	1 e 2

RF	
criterion	gini; entropy
max_depth	3 a 15 (step=2)
min_samples_leaf	3 a 15 (step=2)
min_samples_split	10 a 50 (step=5)
n_estimators	100 a 500 (step=50)

XGboost	
colsample_bytree	0.6 a 1 (step=0.1)
gamma	0 a 1 (step=0.1)
learning_rate	0.1 / 0.01
max_depth	3 a 20 (step=2)
min_child_weight	1 a 10 (step=2)
n_estimators	20 a 500 (step=1) / 480 a 2030 (step=10)
reg_alpha	0, 1e-5, 1e-2, 0.1, 1 e 100
subsample	0.6 a 1 (step=0.1)

Os modelos SVC e *Stacking* não foram submetidos ao GridSearch.

Após estas etapas os resultados dos três modelos com as maiores acurácias nos dados de teste foram agrupados para criar modelo de stacking, utilizando o StackingCVClassifier, do pacote mlxtend, com intuito de obter uma melhor acurácia para o classificador *stacking* do que para os demais classificadores separados.

Fig. 6 *Histogramas dos atributos utilizados para o treinamento.*

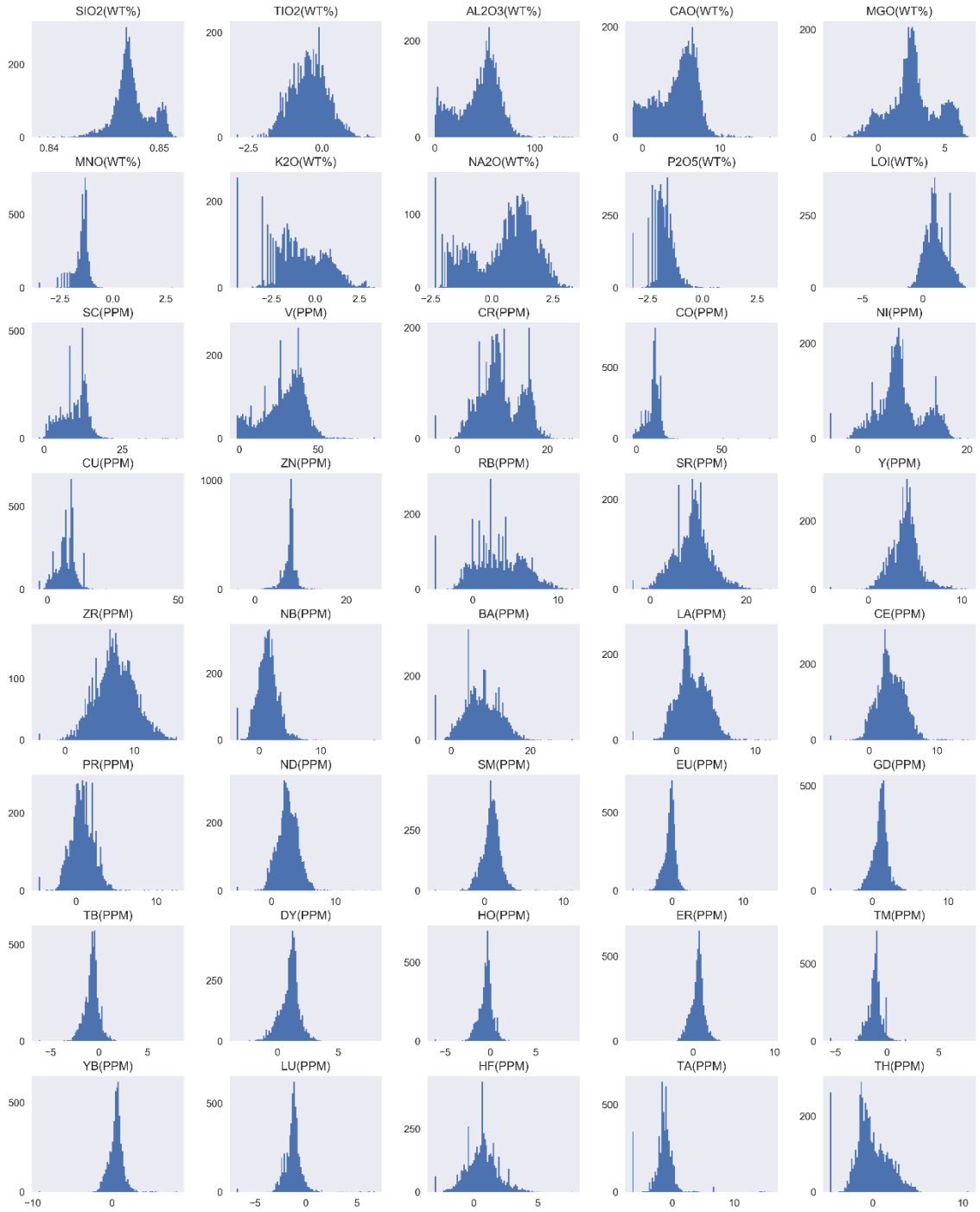

Fig. 7 *Histogramas após transformação box-cox dos atributos utilizados para o treinamento.*

Resultados

Avaliação e Validação do Modelo

Os hiperparametros utilizados em cada um dos modelos foram escolhidos pois obtiveram uma melhor acurácia nos dados de treinamento. A seguir é listado os hiperparametros finais de cada m dos modelos utilizados:

SVC		KNN		RF	
C	1	algorithm	auto	criterion	gini
decision_function_shape	ovr	leaf_size	30	max_depth	13
degree	3	metric	minkowski	max_features	auto
gamma	auto	n_neighbors	3	min_samples_leaf	3
kernel	rbf	p	1	min_samples_split	10
tol	0.001	weights	distance	n_estimators	400

XGboost		Stacking	
base_score	0.5	classifier	xgboost, rf, knn
booster	gbtree	cv	3
colsample_bylevel	1	meta_classifier	LogisticRegression – default
Colsample_bytree	0.9	shuffle	True
gamma	0	stratify	True
learning_rate	0.01	use_probas	True
max_delta_step	0	use_features_in_secondary	False
max_depth	3		
min_child_weight	1		
n_estimators	1840		
reg_alpha	0		
reg_lambda	1		
subsample	0.8		

Os resultados de acurácia destes modelos nos dados de teste, assim como o tempo de treinamento dos modelos se encontram na tabela a seguir:

	SVC	KNN	RF	XGB	STACKING
Acurácia	0.69	0.81	0.85	0.93	0.95
Tempo de Treinamento (segundos)	13.57	0.01	5.49	437.90	1323.61

Para verificar melhor o desempenho dos modelos, a matriz de confusão de cada modelo foi analisada (Fig. 8, 9, 10, 11, 12).

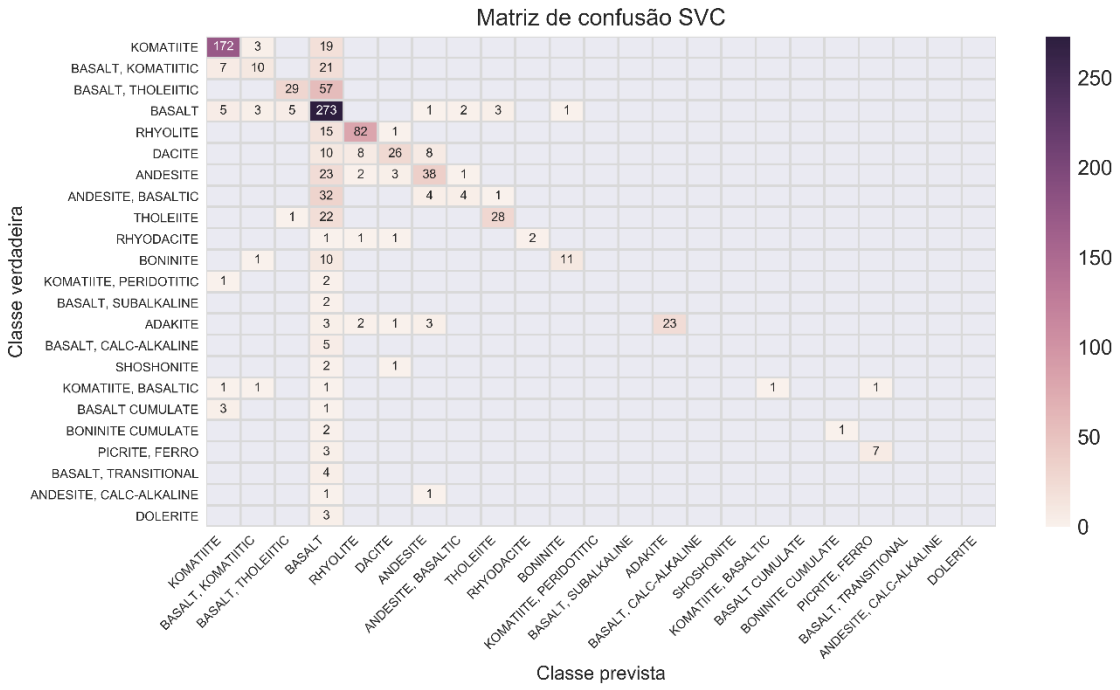

Fig. 8 *Matriz de confusão para o classificador SVC, gerada com os resultados de classificação nos dados de teste.*

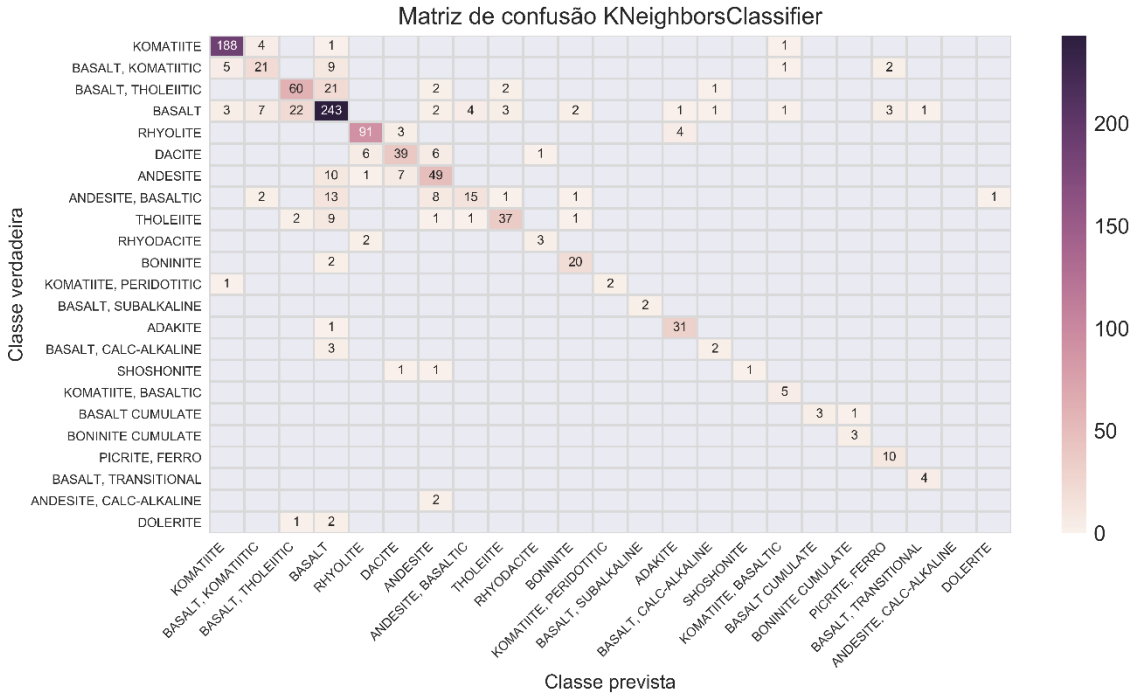

Fig. 9 *Matriz de confusão para o classificador KNN, gerada com os resultados de classificação nos dados de teste.*

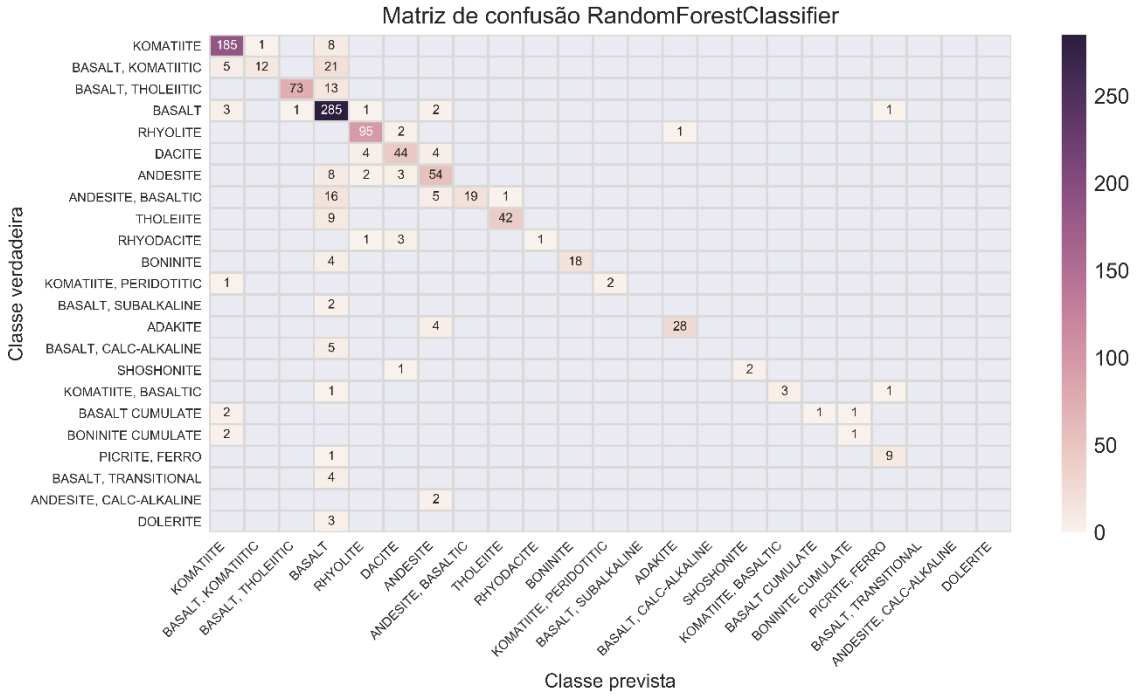

Fig. 10 *Matriz de confusão para o classificador RF, gerada com os resultados de classificação nos dados de teste.*

Os resultados das matrizes de classificação mostram que os modelos de maneira geral sobreajustaram na classe BASALT, pois essa foi a classe com a maior frequência de erros de classificação. O que pode ser explicado pela grande diferença de amostras pertencentes a essa classe quando comparada com as demais classes (Fig. 5). As classes BASALT KOMATIITIC, BASALT THOLEITIC, e ANDESITE BASALTIC, foram as que o modelo teve a maior dificuldade de diferenciar da classe BASALT. O que pode ser explicado pela similaridade química dessas rochas. As

classes com menor quantidade de amostras foram as com maior taxa de erros, sendo o modelo SVC o que teve pior desempenho na classificação destas classes.

Justificativa

Os resultados obtidos para o classificador com a maior acurácia ficaram próximos dos obtidos pelo modelo de *benchmark*. No modelo de *benchmark*, foi utilizado um classificador SVC, que obteve acurácia de 98% nos dados de teste (Fig. 12). Neste trabalho se obteve acurácia de 95% para o classificador STACKING. Entretanto, a acurácia para o classificador SVC neste trabalho foi a mais baixa, 69%. Apesar da discrepância nos valores, a comparação entre o modelo *benchmark* e o modelo desenvolvido neste projeto deve ser realizada com cautela, pois a tarefa de classificação proposta pelo modelo *benchmark* é diferente da proposta neste trabalho, apensar de utilizar o mesmo banco de dados.

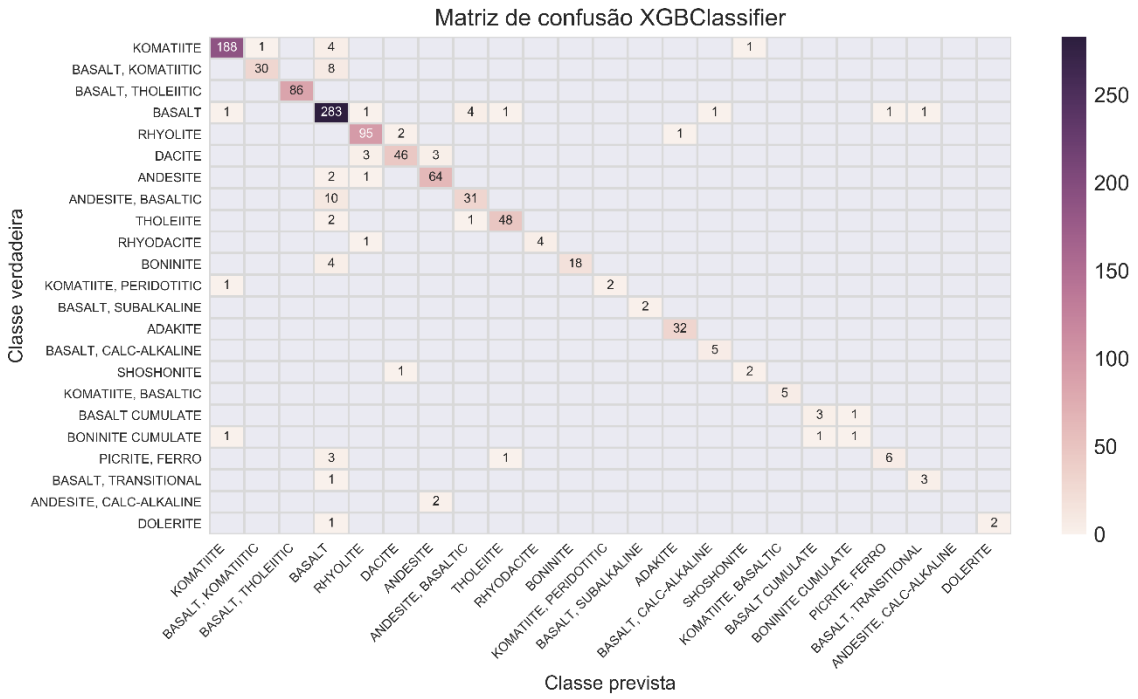

Fig. 11 *Matriz de confusão para o classificador XGboost, gerada com os resultados de classificação nos dados de teste.*

Conclusão

Visualização Free-Form

A figura 12 mostra a matriz de confusão do modelo final (STACKING), que obteve a maior acurácia (95%). Observando a matriz é possível visualizar que o maior erro do modelo é um sobreajuste da classe BASALT, provavelmente provocado pelo discrepante número de amostras dessa classe quando comparada com as demais. A matriz também mostra que as classes com menor quantidade de amostras, ANDESITE CALC-ALKALINE, DOLERITE, foram as que apresentaram a maior quantidade de erros, proporcionalmente.

Os resultados obtidos para o modelo STACKING possibilita a aplicação, na prática, do modelo para a automatização da identificação do nome de rochas vulcânicas.

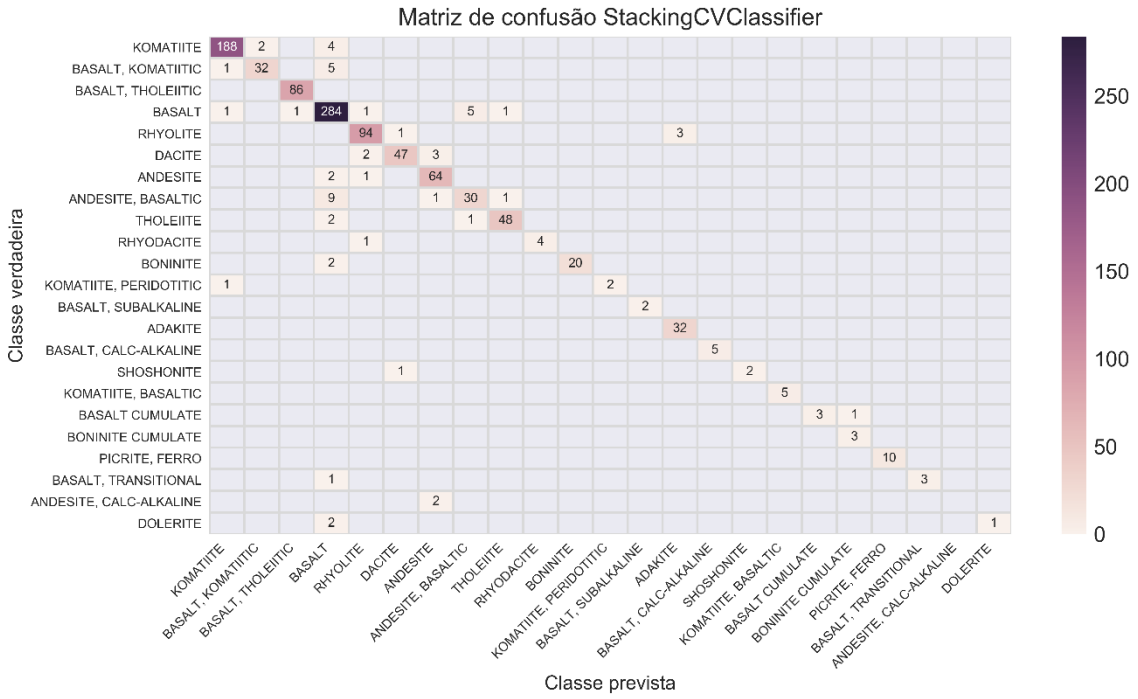

Fig. 12 *Matriz de confusão para o classificador STACKING, gerada com os resultados de classificação nos dados de teste.*

Reflexão

Os processos executados neste projeto podem ser sumarizados nas seguintes etapas:

1. Baixar os dados do banco GEOROC.
2. Processar os dados para o treinamento. Esta etapa consiste na remoção de amostras classificadas como alteradas. Normalização dos dados analíticos utilizando Box-Cox. Remoção dos atributos indesejados para a classificação litológica, como grau de alteração, localização geográfica, dados isotópicos, e elementos com muitas análises faltantes.
3. Separar os dados em dados de treino, validação e teste.
4. Treinar os dados utilizando os seguintes classificadores: SVM, KNN, RF, XGboost.
5. Calcular os hiperparametros com maior acurácia para cada modelo.
6. Analisar a performance dos modelos gerados por cada classificador utilizando a acurácia.
7. Treinar um modelo STACKING com os três melhores classificadores.
8. Indicar o modelo que obteve a melhor pontuação, e avaliar a pontuação obtida.
9. Verificar se a pontuação obtida para o melhor modelo é aceitável para a aplicação prática do modelo.

A etapa 2 foi a mais complicada, pois nesta etapa foi preciso gastar um bom tempo para entender a formatação do dado e também para visualizar todos os possíveis problemas nos dados que poderiam impactar nos modelos.

A parte mais interessante do projeto foi o treinamento do modelo Xgboost, que obteve uma performance muito melhor que os demais algoritmos, apesar de ser o que levou o maior tempo para o treinamento.

Melhorias

O modelo final do projeto pode ser melhorado executando as seguintes sugestões:

1. Treinar o modelo com um banco de dados menos desbalanceado, para que o modelo diminua seu sobreajuste na classe BASALT, e aumente sua acurácia.
2. Treinar o modelo com um banco de dados que contenha todas as rochas vulcânicas existentes, permitindo maior poder de generalização do modelo.
3. Remover os atributos com correlação elevada com os demais (como por exemplo as análises de alguns elementos terras raras), visando diminuir a dimensão dos dados, o que pode diminuir o tempo de treinamento assim como diminuir o sobreajuste.

O modelo final pode ser melhorado significativamente com as sugestões acima.

Bibliografia

BOX, G.E.P., COX, D.R., 1964. AN ANALYSIS OF TRANSFORMATIONS. J. R. STAT. SOC. SER. B 26, 211–252. DOI:10.2307/2287791

CRACKNELL M. J., COWOOD A. L. (2016) CONSTRUCTION AND ANALYSIS OF HYDROGEOLOGICAL LANDSCAPE UNITS USING SELF-ORGANISING MAPS. SOIL RESEARCH 54, 328-345.

CRACKNELL, M.J. & READING, A.M., 2014. GEOLOGICAL MAPPING USING REMOTE SENSING DATA: A COMPARISON OF FIVE MACHINE LEARNING ALGORITHMS, THEIR RESPONSE TO VARIATIONS IN THE SPATIAL DISTRIBUTION OF TRAINING DATA AND THE USE OF EXPLICIT SPATIAL INFORMATION, COMPUTERS & GEOSCIENCES, 63, 22-33.

CRACKNELL, M.J., READING, A.M. & DE CARITAT, P., 2015. MULTIPLE INFLUENCES ON REGOLITH CHARACTERISTICS FROM CONTINENTAL-SCALE GEOPHYSICAL AND MINERALOGICAL REMOTE SENSING DATA USING SELF-ORGANIZING MAPS, REMOTE SENSING OF ENVIRONMENT, 165, 86-99.

HILL, E.J., ROBERTSON, J. & UVAROVA, Y., 2015. MULTISCALE HIERARCHICAL DOMAINING AND COMPRESSION OF DRILL HOLE DATA, COMPUTERS & GEOSCIENCES, 79, 47-57.

MERDITH, A.S., LANDGREBE, T.C.W. & MULLER, R.D., 2015. PROSPECTIVITY OF WESTERN AUSTRALIAN IRON ORE FROM GEOPHYSICAL DATA USING A REJECT OPTION CLASSIFIER, ORE GEOLOGY REVIEWS, 71, 761-776.

Petrelli, M., Perugini, D. (2016) – Solving petrological problems through machine learning: the study case of tectonic discrimination using geochemical and isotopic data. Contributions to Mineralogy and Petrology,171 (10), 81.

Reading, A.M. & Gallagher, K., 2013. Transdimensional change-point modeling as a tool to investigate uncertainty in applied geophysical inference: An example using borehole geophysical logs, Geophysics, 78, WB89-WB99.