Montagem de ambiente para classificação de solos usando ScikitLearn

Setting up of an environment for soil classification using ScikitLearn

Gabriel Teston Vasconcelos[1]
Kleber Xavier Sampaio de Souza[2]
Stanley Robson de Medeiros Oliveira[3]
João Camargo Neto[4]

Resumo – Técnicas de Mineração de Dados e Modelagem preditiva são cada vez mais usadas para automação de tarefas nos mais diversos campos do conhecimento. O da agricultura é um deles, existindo diversos modelos para predição de eventos climáticos, ocorrências de pragas e produtividade. A classificação de solos é uma das tarefas dentro dessa área que ainda não possui um sistema computacional satisfatório. Este trabalho tem como objetivo a criação de um sistema para a classificação automática de solos, a partir de dados previamente classificados segundo o método descrito no Sistema Brasileiro de Classificação de Solos (SiBCS). A modelagem para o sistema de classificação aqui proposto tem como base algoritmos de Aprendizado de Máquina. O trabalho ainda está em andamento e os resultados obtidos até agora indicam que a abordagem é promissora.

Termos para indexação: árvores de decisão, floresta aleatória, SVM, KNN, mineração de dados, atributos de solos.

Abstract – Data mining techniques and predictive modeling are increasingly being used for automation of several tasks in diverse knowledge fields. Agriculture is one of these fields, for which there are models for predicting climatic events, occurrence of pests and productivity. Soil classification is one of the tasks within this area that does not yet have a satisfactory computational system. . This work aims to create a system for the automatic classification of soils, based on data previously classified according to the method described in the Brazilian Soil Classification System. The modeling for the classification system proposed here is based on Machine Learning algorithms. Work is still ongoing and results so far indicate that the approach is promising.

Index terms: decision tree, random forest, SVM, KNN, data mining, soil attributes.

1 Estudante de Engenharia da Computação, bolsista da Embrapa Informática Agropecuária

2 Doutor em Engenharia Elétrica, pesquisador da Embrapa Informática Agropecuária

3 Doutor em Ciência da Computação, pesquisador da Embrapa Informática Agropecuária

4 Doutor em Engenharia Agrícola, analista da Embrapa Informática Agropecuária

Introdução

A adequada classificação de um solo permite estabelecer correlações com sua gênese e evolução, assim como com fatores ambientais e econômicos relativos à sua ocupação, manejo, aptidão agrícola, entre outros (Oliveira et al., 1992).
O SiBCS está organizado em níveis categóricos, levando em consideração diversos atributos físicos, químicos, morfológicos e mineralógicos de um perfil de solo. Contudo, esse sistema está em constante evolução, sendo possível a adição de novas classes, assim como a reestruturação das existentes.
Devido à complexidade do sistema de classificação, essa tarefa é realizada por especialistas no assunto, que detêm grande conhecimento e experiência sobre os solos brasileiros. Logo a criação de um sistema de classificação automática de solos é altamente desejável, pois o SiBCS não possui um software que auxilie nessa tarefa.
Idealmente, esse software deveria trazer como vantagens: a) classificação de novos perfis de solo; b) revisão das classificações já realizadas; c) análise de todo o conjunto de dados para auxiliar nas revisões do próprio SiBCS; e d) utilização como material didático.
Uma das abordagens para a criação de um software dessa espécie é a utilização de algoritmos de aprendizado de máquina. Nessa abordagem, os algoritmos têm a capacidade de aprender padrões ao analisar o conjunto de dados e sugerir como os perfis de solo podem ser classificados, permitindo a tomada de decisões.
Em uma primeira abordagem, utilizou-se os algoritmos contidos no software Weka para realizar a classificação (Vasconcelos; Oliveira, 2018). Esta abordagem, entretanto, apresentou alguns problemas de escalabilidade para o desenvolvimento de um sistema em produção, sendo necessários diversos passos para gerar uma predição pelos modelos treinados. Decidiu-se, então, pela migração para a linguagem Python e sua plataforma SciKit Learn (Pedregosa et al., 2011). Este trabalho visa relatar os resultados obtidos até o momento durante a construção desta nova versão do sistema de classificação automática, agora utilizando a plataforma SciKit Learn.

Material e Métodos

Origem dos Dados

Todo sistema de classificação que utiliza algoritmos de aprendizado de máquina requer um conjunto de dados para seu treinamento. Os dados de solos analisados para a construção do sistema de classificação foram obtidos do Mapeamento de Recursos Naturais do Brasil, disponíveis no Instituto Brasileiro de Geografia e Estatística (IBGE) (IBGE, 2018). Em particular, foram considerados os atributos de solos relacionados à pedologia. Cada perfil apresenta um ou mais horizontes de solos, perfazendo um total de 23.534 horizontes (instâncias). De cada perfil foram considerados dados de local, posição no relevo, declividade, altitude, litologia, relevo local, erosão, drenagem e uso. Dos horizontes foram considerados dados referentes aos atributos morfológicos, físicos, químicos e mineralógicos. O conjunto de dados original é composto de 23.534 instâncias e 95 atributos.

Tratamento dos Dados

Foram removidos, do conjunto de dados originais, os perfis sem classificação e seus respectivos horizontes, pois estes não são úteis nem na classificação nem na validação. Além disso, foram removidos atributos com mais 80% dos valores faltantes. Com estas supressões, o conjunto de dados final foi constituído de 17.796 instâncias (horizontes) e 58 atributos de solos, sendo um deles a classificação no primeiro nível categórico do SiBCS.

Nesta nova abordagem, com o SciKit Learn, foi necessário o tratamento dos dados para alimentar os algoritmos, sendo o conjunto de dados completo com a média e a moda para atributos numéricos e categóricos, respectivamente, uma vez que os algoritmos trabalham apenas com valores reais não nulos. A média e a moda são tradicionalmente usadas para preenchimento de valores numéricos e categóricos, respectivamente.Também foi necessária a conversão dos atributos categóricos para atributos binários, seguindo a técnica de one hot-bit encoding. Por exemplo, um atributo X que possua três possíveis valores (a, b e c) passa a ser representado por três atributos binários (Xa, Xb e Xc) que recebem valor zero ou um.

Algoritmos de Aprendizado de Máquina

Foram utilizados algoritmos do pacote Scikit-learn implementados em python para gerar os modelos preditivos. Os algoritmos utilizados para a geração dos modelos são listados a seguir, com suas respectivas identificações de classes, dentro do pacote Scikit-learn:

Árvore de Decisão (sklearn.tree.DecisionTreeClassifier):
Método baseado no conceito de entropia que gera uma árvore de condições baseada em um conjunto de dados fornecido, particionando o conjunto de dados a cada teste, tendo nas extremidades desse grafo as respectivas classes de cada partição (Breiman et al., 1984).
O nome do algoritmo vem da representação gráfica que pode ser interpretada como uma árvore invertida, sendo um caminho da raiz até uma folha a sequência de testes a que uma instância deve ser bem-sucedida ou a característica dessa instância para que faça parte da classe representada na folha. Na Figura 1 podemos visualizar parte da árvore de decisão gerada sobre o conjunto de dados Iris usando o algoritmo C4.5 (Quinlan, 1986). Nesta árvore, caso a largura de pétala (petalwidth) estiver entre 0,6 e 1,7, e o comprimento da pétala (petallenght) for menor ou igual a 4,9, classifica-se a planta como Iris versicolor.

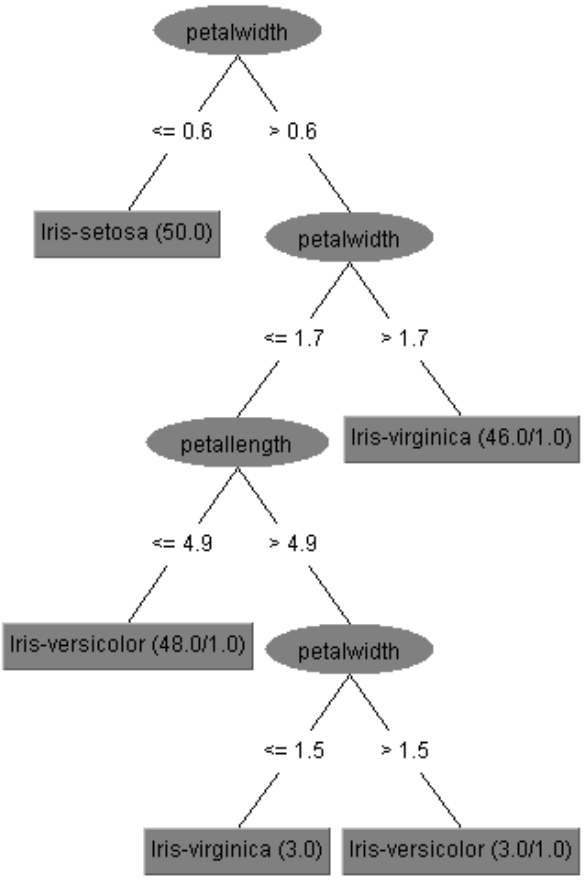

Figura 1: Árvore de decisão usando o algoritmo C4.5 com o conjunto de dados Iris.

Floresta Aleatória (sklearn.ensemble.RandomForestClassifier):
Semelhante ao algoritmo anterior, este algoritmo gera diversas Árvores de Decisão, cada uma com uma parcela dos dados fornecidos, o que dificulta a ocorrência de overfitting e aumenta a acurácia do modelo (Breiman, 2001). Para a classificação de uma instância, existe uma votação entre as diversas árvores da floresta, levando em consideração a possibilidade de acerto de cada uma.

KNN (sklearn.neighbors.KNeighborsClassifier):
Diferentemente dos algoritmos acima citados, o K-Nearest Neighbors (KNN), leva em consideração a distribuição das instâncias de treino em um espaço vetorial (Aha et al., 1991). Sendo a classificação de uma instância a moda entre a classificação dos K vizinhos mais próximos nesse espaço, conforme Figura 2.

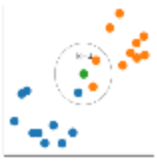

Figura 2. Exemplo de funcionamento KNN.

SVM (sklearn.svm.LinearSVC):
Os modelos gerados pelo Support Vector Machines (SVM) (Alex; Bernhard, 2004) se baseiam na criação de hiperplano de separação ótima, maximizando a margem de separação (linhas pontilhadas, na Figura 3) entre as classes no espaço gerado pelos dados de entrada. Embora a Figura 3 esteja no espaço bidimensional, na realidade os planos de separação ocorrem em um espaço com alta dimensionalidade (Vapnik, 1995).

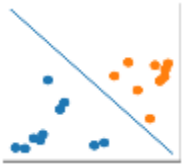

Figura 3: Exemplo de hiperplano de separação ótima.

Validação dos Modelos Preditivos

Para cada algoritmo foi gerado um modelo preditivo, visando maximizar o acerto por horizonte. O conjunto de dados também foi separado em dois, sendo 70% dos perfis para treino e 30% para teste. Note-se que a separação levou em conta todos os horizontes de um mesmo perfil, de modo que eles não fossem separados. Além disso, foi mantida a representatividade de cada classe nos conjuntos de teste e treino.
Após verificadas as acurácias dos algoritmos por horizonte, foi gerado um sistema de comitê, que primeiramente verifica a classificação de todos os horizontes de um perfil para cada modelo e, subsequentemente, a classificação do perfil entre os modelos, obtendo-se assim a precisão para perfis do sistema como um todo.

3. Resultados e Discussão

Para a criação tanto da Árvore de Decisão quanto para a Floresta Aleatória foram utilizados os hiperparâmetros padrão de cada algoritmo.
Para o modelo gerado pelo KNN foi utilizado o valor de K = 4, valor esse obtido após uma análise exploratória dos diversos valores de K, conforme mostrado na Figura 4.
.

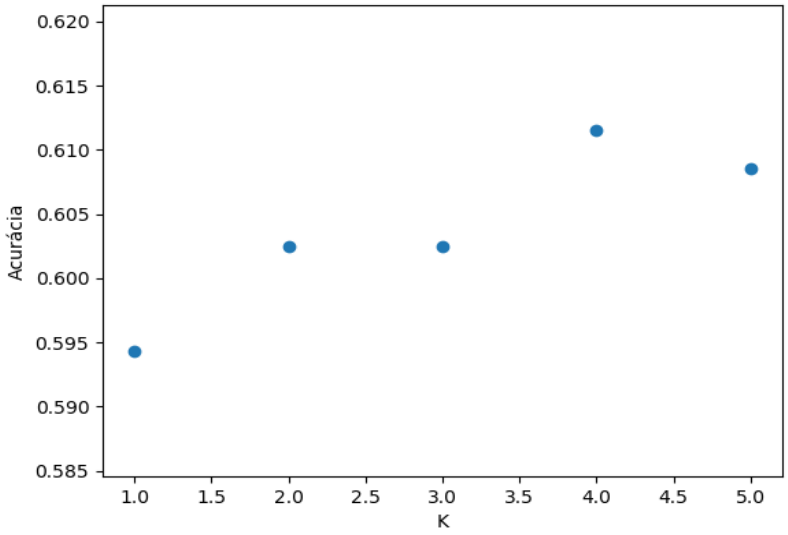

Figura 4: Ajuste do valor de K, sendo k = 4 o valor que maximiza a acurácia.

Para o modelo gerado pelo SVM foi utilizada a implementação específica para kernel linear (LinearSVC) após uma análise exploratória dos outros possíveis kernels.

A Tabela 1 apresenta os resultados obtidos até o momento usando o ambiente Scikit-Learn, que já apresenta a escalabilidade desejada. Ainda estão sendo buscadas alternativas para melhorar o pré-processamento de dados (preenchimento de valores faltantes, por exemplo) visando aumentar a precisão dos modelos.

Tabela 1. Resultados das métricas de avaliação dos modelos preditivos. Os algoritmos estão identificados com seus respectivos nomes em inglês.

	Decision Tree	Random Forest	KNN	SVM
Latossolo	0.75891583	0.7382311	0.74108417	0.79029957
Argissolo	0.73836608	0.85418821	0.74560496	0.71716649
Neossolo	0.58629442	0.53807107	0.51269036	0.78680203
Nitossolo	0.16949153	0.18644068	0.31638418	0.28813559
Plintossolo	0.39130435	0.33201581	0.32015810	0.57312253
Chernossolo	0.41269841	0.4047619	0.34126984	0.66666667
Gleissolo	0.77692308	0.74615385	0.72307692	0.78076923
Espodossolo	0.34246575	0.43835616	0.52054795	0.38356164
Cambissolo	0.38211382	0.22222222	0.22222222	0.49864499
Planossolo	0.42331288	0.3803681	0.28220859	0.52147239
Luvissolo	0.19417476	0	0.03883495	0.23300971
Vertissolo	0.42857143	0.07142857	0.35714286	0.5
Organossolo	1	0.25	0.5	0.75
Total	0.63913	0.650473	0.611531	0.686011

A acurácia do sistema, levando em consideração todos os classificadores foi 0.68829.

Conclusão

Neste trabalho foi possível a criação de um protótipo de sistema de classificação automática de solos para o primeiro nível categórico do SiBCS, baseado em quatro algoritmos de Aprendizado de Máquina, utilizando a plataforma Scikit Learn.
O sistema proposto necessita de algumas melhorias, pois mesmo tendo um aumento de escalabilidade, quanto ao volume de dados suportados, os modelos gerados neste estudo ainda não apresentam resultados similares àqueles disponíveis na literatura. Essa diferença pode ser devida ao tratamento de dados na fase de pré-processamento.
Este trabalho ainda está em andamento e os próximos passos para a sua continuação serão: a) evoluir os procedimentos para tratamento dos dados e reavaliação dos resultados; b) aprimorar os modelos preditivos através de seleção de atributos; e c) expandir os resultados para os próximos níveis de classificação do SiBCS.

Agradecimentos

Os autores agradecem ao programa CNPq/PIBIC pela concessão da bolsa de Iniciação Científica, processo Nº106600/2018-4 para o aluno Gabriel Teston Vasconcelos, à equipe do projeto SmartSolos da Empresa Brasileira de Pesquisa Agropecuária (Embrapa) pelo apoio oferecido durante o desenvolvimento e ao professor Ricardo Coelho do Instituto Agronômico de Campinas (IAC) pelos valiosos comentários sobre o processo de classificação de solos.

Referências

AHA, D.; KIBLER, D.; ALBERT, M. K. Instance-based learning algorithms. **Machine Learning**, v. 6, n. 1, p. 37-66, 1991.

ALEX, J. S.; BERNHARD, S. A tutorial on support vector regression, **Statistics and Computing**, v. 14, n. 3, p. 199-222, Aug. 2004.

BREIMAN, L. Random forests. **Machine Learning**, v. 45, n.1, p. 5-32, Oct. 2001.

BREIMAN, L.; FRIEDMAN, J.; OLSHEN, R.; STONE, C. **Classification and regression trees**. Belmont: Wadsworth International Group,1984. 358 p. il.

IBGE. Mapeamento de recursos naturais do Brasil: Escala 1:250.000. Rio de Janeiro, 2018. (IBGE. Documentação técnica geral). Disponível em: http://geoftp.ibge.gov.br/informacoes_ambientais/vegetacao/vetores/escala_250_mil/DOCUMENTACAO_TECNICA_MRN.pdf>. Acesso em: 5 maio de 2018.

OLIVEIRA, J. B.; JACOMINE, P. K. T.; CAMARGO, M. N. **Classes gerais de solos do Brasil**: guia auxiliar para seu reconhecimento. 2 ed. Jaboticabal: Funep, 1992. 201 p.

PEDREGOSA, F.; VAROQUAUX, G.; GRAMFORT, A.; MICHEL, V.; THIRION, B.; GRISEL, O.; BLONDEL, M.; PRETTENHOFER, P.; WEISS, R.; DUBOURG, V.; VANDERPLAS, J.; PASSOS, A.; COURNAPEAU, D.; BRUCHER, M.; PERROT, M.; DUCHESNAY, E. **Scikit-learn**: Machine Learning in Python. Journal of Machine Learning Research, v. 12, p. 2825-2830, 2011.

QUINLAN, J. R. Induction of decision trees. **Machine Learning**, v. 1, p. 81–106, 1986.

VAPNIK, V. N. **The nature of statistical learning theory**. 2nd. New York: Springer-Verlag, 1995. 188 p. il.

VASCONCELOS, G. T.; OLIVEIRA, S. R. de M. Avaliação da eficiência de algoritmos de aprendizado de máquina para classificação automática de solos. In: CONGRESSO INTERINSTITUCIONAL DE INICIAÇÃO CIENTÍFICA, 12., 2018, Campinas. Anais... [S.l: s.n], 2018. Não paginado. CIIC 2018. Disponível em: <http://ainfo.cnptia.embrapa.br/digital/bitstream/item/183363/1/18603.pdf>.