Uncover-ML: a machine learning pipeline for geoscience data analysis

Article · July 2020

DOI: 10.11636/134466

7 authors, including:

Sudipta Basak
Geoscience Australia
14 PUBLICATIONS **667** CITATIONS
SEE PROFILE

Rakib Hassan
Geoscience Australia
46 PUBLICATIONS **622** CITATIONS
SEE PROFILE

Australian Government
Geoscience Australia

Exploring for the Future
minerals | energy | groundwater

Uncover-ML: a machine learning pipeline for geoscience data analysis

J. Wilford[1], S. Basak[1], R. Hassan[1], B. Moushall[1], L. McCalman[2], D. Steinberg[2] and F. Zhang[1]

[1]Geoscience Australia, [2]Gradient Institute, Canberra

The geosciences are a data-rich domain where Earth materials and processes are analysed from local to global scales. However, often we only have discrete measurements at specific locations, and a limited understanding of how these features vary across the landscape. Earth system processes are inherently complex, and trans-disciplinary science will likely become increasingly important in finding solutions to future challenges associated with the environment, mineral/petroleum resources and food security. Machine learning is an important approach to synthesise the increasing complexity and sheer volume of Earth science data, and is now widely used in prediction across many scientific disciplines. In this context, we have built a machine learning pipeline, called Uncover-ML, for both supervised and unsupervised learning, prediction and classification. The Uncover-ML pipeline was developed from a partnership between CSIRO and Geoscience Australia, and is largely built around the Python scikit-learn machine learning libraries. In this paper, we briefly describe the architecture and components of Uncover-ML for feature extraction, data scaling, sample selection, predictive mapping, estimating model performance, model optimisation and estimating model uncertainties. Links to download the source code and information on how to implement the algorithms are also provided.

A key driver for development of the Uncover-ML code base was to address the issue of data sparsity. This is a common problem in the geosciences and many other science disciplines. Data sparsity relates to the fact that, although we typically have a rich resource of geospatial information (e.g. geophysical data, satellite imagery, terrain attributes), we invariably have, in comparison, very few field samples that are accurately interpreted or quantitatively measured. The former either occur everywhere or have a large spatial extent—for example, satellite imagery or airborne survey datasets—whereas the latter typically occur at specific sites (e.g. drillholes, soil or rock geochemical samples).

Machine learning provides a mechanism to learn relationships between site observations and geospatial datasets, and to use these relationships to generate predictive maps. These maps represent spatial predictions of the target value with the same resolution and extent as the geospatial datasets, and have distinct advantages over common geostatistical methods such as krigging and inverse distance weighting (Wilford et al., 2016). This type of predictive mapping is a departure from traditional approaches that use empirical knowledge of experts to build compilations and interpretive maps. In contrast, the machine learning approach explores correlations between the site attribute and thematic datasets with statistical measures of model performance and uncertainly.

Machine learning has been criticised as being ‘black box’, where the generation of a model prediction is disconnected from the domain expert. We have therefore designed Uncover-ML as a processing pipeline, where the domain expert can influence many stages of the modelling process. The domain expert can assess the quality of the model through data diagnostics, performance indicators and estimates of uncertainty. Domain experts also assess the model to determine which covariates should be used to produce a model that makes geological sense. As part of the Exploring for the Future (EFTF) program, Uncover-ML has been used to detect Cu anomalies in stream sediments, and to predict weathering intensity (Wilford, 2020) and thickness of sedimentary cover (Bonnardot et al., 2020).

Architecture and components

Uncover-ML is a machine learning software package designed to address classification and regression problems in the geosciences. It consists of a series of interconnected workflows in the form of a processing pipeline. The components in the pipeline define the overall architecture of Uncover-ML (Figure 1). The design of the pipeline has been strongly influenced by the development of, and methodologies used in, digital soil mapping (McBratney et al., 2003; Hartemink et al., 2008; Hengl, 2009). The main algorithms available through Uncover-ML include support vector regression, Gaussian process, random forest, cubist and boosted trees. Many of these codes are sourced from the scikit-learn machine learning resource (https://scikit-learn.org/stable; Pedregosa et al., 2011) and scalable Bayesian linear models (https://github.com/NICTA/revrand).

Inputs and scaling

Targets and covariates

The pipeline has two inputs: targets (also called labels) and covariates (also called predictive datasets). Targets can be either integer or floating-point values, and represent the attribute you want to predict. A target might be a geological class, geophysical measurement, geochemical concentration or measurements down a drillhole.

Covariates are used in prediction; they include primary datasets such as magnetic intensity, terrain elevation and gravity, and secondary derivatives from these primary data types—for example, in the case of elevation, slope, aspect and relief. These primary datasets and secondary derivatives form ‘features’ that machine learning uses in learning and prediction. All covariates for Uncover-ML are input in GeoTIFF format with the same cell size, spatial extent and projection. Covariates can be either continuous (e.g. imagery) or categorical (e.g. geological/soil map units). Feature

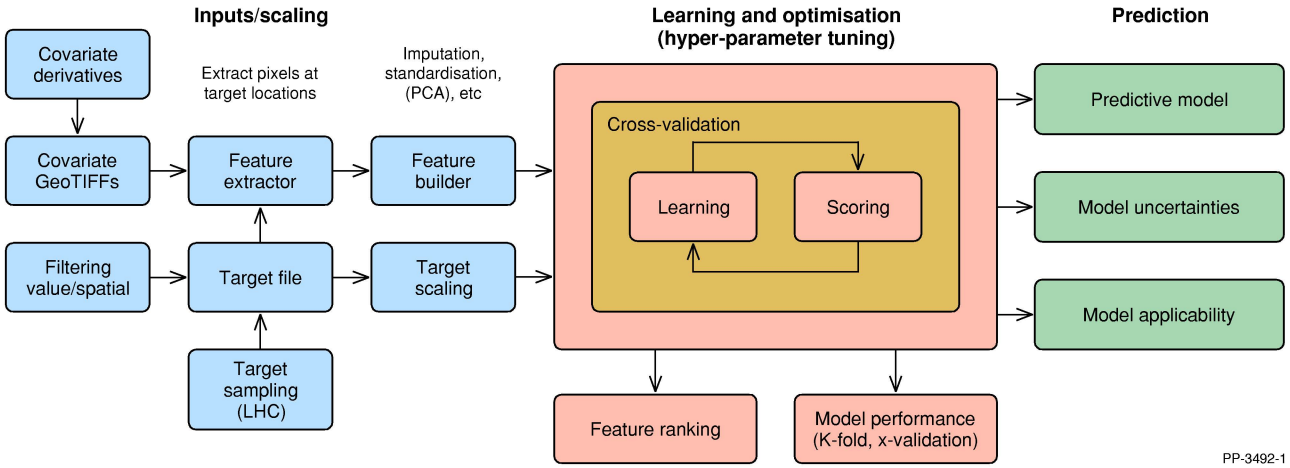

Figure 1 The Uncover-ML pipeline, consisting of three interconnected custom-built workflow components: 1) data inputs, filtering, scaling, transforming and imputation of missing or no-data covariate values; 2) learning, optimisation, feature ranking and model performance assessment; and 3) model prediction, uncertainty and applicability. LHC = Latin Hyper-Cube; PCA = principal component analysis.

generation and feature selection is an important component of machine learning. Adding too many covariates to a predictive model can lead to model overfitting. In addition, large numbers of covariates increase training and prediction compute time, which can be significant for very large, complex national models. Feature generation can also refine the questions posed of machine learning—for example, bare-earth Landsat imagery where most seasonal green vegetation has been removed using time-series analysis (Roberts et al., 2019). The bare-earth bands provide a more powerful set of covariates for machine prediction of soil and bedrock (Wilford and Roberts, 2020). Multiscale transformations of topography (Wilford et al., 2020) can be used in tree-based machine learning that is otherwise poor in capturing neighbourhood relationships.

Simpler models using a smaller, selective set of non-correlated covariates are likely to give better results and scale more reliably. Uncover-ML supports feature ranking, which allows the user to identify and remove uninformative covariates. Feature ranking can also be used to better understand the geological factors that control the distribution of the target variable (Figure 2).

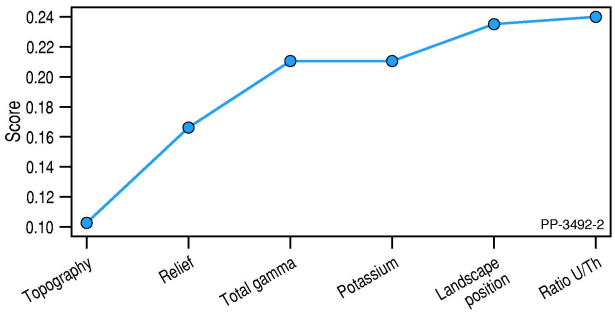

Figure 2 Feature ranking in Uncover-ML, which allows listing of the importance of covariates (x axis) in the model (score = r^2). This can be useful in understanding the factors or environmental/geological controls that influence the distribution of a target variable. In this context, feature ranking can be used to facilitate knowledge discovery.

Imputation

If values are missing in the one or more covariates where a target is located, the missing values can be imputed. This is crucial when you only have a small training dataset. Uncover-ML offers several options to deal with missing values, including:

- impute using the mean of the covariate
- impute by interpolation, using a Gaussian distribution fit to the data
- represent *K* covariates with a *K*-dimensional tree and impute using the average value of *n* nearest neighbours (where *n* is a user parameter).

Data scaling, transformation and reduction

Many machine learning algorithms work best when the input data are scaled consistently. For example, performance and speed increase when features are similarly scaled and approximately normally distributed. Both the selection of covariates and their transformation allow the expert user to pose specific questions about implicit relationships between the targets and covariates to achieve the best results.

Several options are available to scale the input data to a common level of magnitude. Common transformations—including centring, square root, natural logarithm and standard-normal transformations—can be applied to either input data type. Target datasets can also be transformed by kernel density estimation, logistic or rank transformations. For covariates, an important transformation is principal component analysis (PCA, or whitening). PCA is a decorrelation technique that uses orthogonal transformations to split correlated data into linearly uncorrelated variables. By selecting a small number of principal components, PCA can be used to decrease the dimensionality and reduce noise in the input covariates that are used in prediction. This enhances the performance of modelling by reducing the non-uniqueness in the ranking of covariates. However, we recommend that the covariates are scaled before PCA.

Multiscaling covariates

Many landscape and geophysical datasets (e.g. topography, drainage networks, magnetic intensity, earthquake epicentres) exhibit fractal patterns (Turcotte, 1992), which show the same statistical properties at many different scales. However, machine learning algorithms are typically unable to exploit the self-similarity of input data sets at long wavelengths. Targets and the corresponding covariate values used for training are point measurements that invariably do not take into account spatial relationships. We can capture these relationships by generating several multiscale versions of each covariate using 2D wavelet reconstructions

(Kalbermatten et al., 2012). By including these multiscale versions of each covariate, we enable the machine learning algorithms to embed these relationships during training.

We use PyWavelets (Gregory et al., 2019), an open-source Python package, for decomposing and reconstructing raster data based on dyadic wavelet transforms. We decompose and reconstruct each raster into progressively longer wavelength representations, while preserving their original pixel resolution, which is an essential requirement for the machine learning pipeline. This multiscaling functionality has been incorporated into the pre-processing module of Uncover-ML, which allows us to selectively apply it on continuous, non-categorical raster data. Preliminary prediction results obtained by including multiscaled covariates in the training phase show significant predictability improvements (up to ~40%) compared with standard models.

Target selection

Since machine learning algorithms find predictive relationships between the target value and covariates, we can use the distribution of the covariates to sample the 'feature space' when acquiring new training data. Feature space sampling or stratified random sampling is implemented using the conditioned Latin hypercube (Minasny and McBratney, 2006). The Latin hypercube is particularly useful for locating new sites for training by reducing redundant sampling of the features, supplementing existing models with new site observations and prioritising existing training datasets for interpretation or analysis (Figure 3).

Learning and optimisation

Model training and performance

The performance or robustness of the machine-generated predictive models reflects the strength of the correlation between the targets and covariates. Model learning and validation is achieved by splitting targets into training and testing subsets. The most common cross-validation (CV) method is called K-Fold CV. In K-Fold CV, we split our training targets into K number of subsets, called folds. We train our model K times by successively training on K – 1 folds and validating on the left-out fold. By systematically shifting the validation fold, we are able to use all the data for both training and validation. Model performance metrics are based on an average of all the folds, and the predictive model output is based on all the training data.

Uncover-ML provides a variety of metrics for evaluating either supervised or unsupervised predictive models. These metrics are derived from CV using training and validation subsets of the target variable described earlier. Scatter plots of actual vs predicted values are automatically generated (cf. Wilford and Roberts, 2020). Importantly, however, good statistical performance does not guarantee a good model. Predictions that score well from CV may reflect overly simplistic or complicated models that scale poorly. Therefore, critical evaluation of the prediction by the discipline expert and field checking are strongly recommended.

Model optimisation

Model optimisation typically involves experimenting with the various algorithm settings to improve prediction performance (Figure 1). This process, also called hyper-parameter tuning, can involve a lengthy trial-and-error process. Evaluating different combinations of tuning parameters can involve many hundreds of different combinations. To assist, Uncover-ML implements random- and grid-search routines (https://scikit-learn.org/stable), which work in combination to narrow down and define the optimum configuration of hyper-parameters for the model.

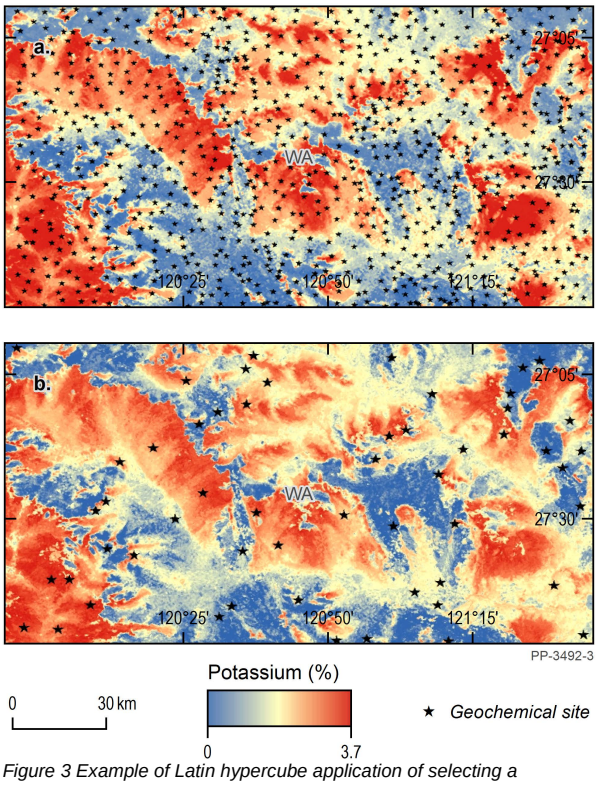

Figure 3 Example of Latin hypercube application of selecting a representative set of 64 targets from a total of 1026 surface geochemical sites. Random forest was trained on 1026 sites in (a) and 64 sites in (b) to generate a prediction map of potassium concentration. Applying the same testing subset, the cross-validation r^2 is 0.712 for model (a) and a respectable 0.610 for model (b), despite training on less than 10% of the original geochemical sites.

Prediction

Uncover-ML generates raster predictions in GeoTIFF format. In the case of probabilistic models, an ensemble of predictions can be calculated, and output raster values can represent the mean, median, variance or standard deviation of all the model iterations. These summary statistics provide metrics on the performance and uncertainty of the model prediction. The larger the standard deviation, the less certain we are of the prediction (i.e. the greater the spread of prediction values for a given location). Upper and lower quantiles can be defined by the user, and corresponding GeoTIFFs are generated based on the specified quantile range.

Machine learning only trains on the targets and associated covariates it 'sees'. It is important to assess how representative the training datasets are to the study area as a whole. The applicability of the model to the area of investigation is assessed using covariate drift. Covariate drift provides a spatial classification of where the model is likely to capture the variations seen in the covariates and where it has not adequately learnt these relationships. Covariate drift gives the user an indication of where the model can be extrapolated (Figure 4).

Download and implementation

To support the wider geoscientific community, we make Uncover-ML freely available as an open-source package through our GitHub repository (https://github.com/GeoscienceAustralia/Uncover-ML). We

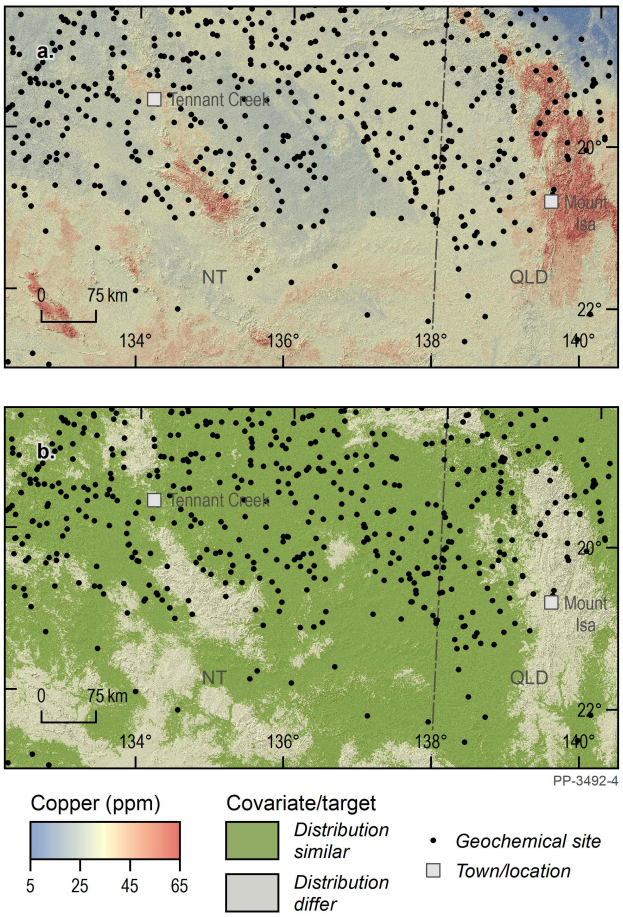

Figure 4 Example of covariate drift assessment overlain on a digital terrain model. (a) Machine learning predictive model using the random forest algorithm of copper concentration based on Cenozoic age soil samples. (b) Covariate shift classified image: green = model representatively samples the covariates; pale yellow = model does not sample the covariates well, and the model should be interpreted with caution. Note correspondence of high values in (a) with undersampled regions in (b).

provide installation instructions for both Linux and high-performance computing environments. Uncover-ML has been developed in Python. The easiest way to run and interact with the code is through Jupyter notebooks, YAML configuration files or command line arguments.

Implementation of Uncover-ML using YAML does not require Python programming experience, and provides an easy way to change and tune parameters across an integrated workflow. YAML scripts and associated files can be used as a metadata record linked to model predictions. These metadata records capture the inputs and processing steps used to generate a predictive model, and therefore enable reproducibility.

Future development of the pipeline will include improving online documentation and simplifying the steps required to run the code. A simple-to-use graphic interface to access Uncover-ML functionality is in development for the EFTF portal (https://portal.ga.gov.au/persona/eftf). This interface will allow users to train and generate predictive models using the extensive library of geospatial datasets accessible via the EFTF portal.

In terms of new functionality, two new modelling architectures are planned: super learners and deep convolutional neural networks (CNNs). Super learners are ensembles of algorithms that are combined to improve the final model prediction. In general, super learners perform either the same or better than using individual algorithms in isolation. The CNN approach has been shown to improve predictive accuracy compared with more commonly used decision tree–based modelling approaches (Padarian et al., 2019). The convolution component of a CNN deals explicitly with neighbourhood relationships; this makes CNNs particularly attractive in the geosciences, where scale relationships are often fundamental in interpreting and understanding processes.

Acknowledgements

This research was undertaken with the assistance of resources from the National Computational Infrastructure (NCI Australia). We thank Philip Main and Marcus Haynes for peer review of this abstract. The authors also wish to thank the overall support and guidance from Karol Czarnota and Richard Blewett. This abstract is published with the permission of the CEO of Geoscience Australia.

eCat: 134466, doi: 10.11636/134466

References

Bonnardot M.-A., et al., 2020. Mapping the cover in northern Australia: towards a unified national 3D geological model. In: Czarnota K., et al. (eds.), *Exploring for the Future: extended abstracts*, Geoscience Australia, Canberra, 1–4.

Gregory L., et al., 2019. PyWavelets: a Python package for wavelet analysis. *Journal of Open Source Software* 4:1237.

Hartemink A. E., McBratney A. & Mendonça-Santos M. L. (eds.), 2008. *Digital soil mapping with limited data*, Springer.

Hengl T., 2009. *A practical guide to geostatistical mapping*, EUR 22904 EN Scientific and Technical Research Series, Office for Official Publications of the European Communities, Luxembourg.

Kalbermatten M., et al., 2012. Multiscale analysis of geomorphological and geological features in high resolution digital elevation models using the wavelet transform. *Geomorphology* 138:352–63.

McBratney A. B., Mendonça-Santos M. L. & Minasny B., 2003. On digital soil mapping. *Geoderma* 117:3–52.

Minasny B. & McBratney A. B., 2006. A conditioned Latin hypercube method for sampling in the presence of ancillary information. *Computers & Geosciences* 32:1378–88.

Padarian J., Minasny B. & McBratney A. B., 2019. Using deep learning for digital soil mapping. *SOIL* 5:79–89.

Pedregosa F., et al., 2011. Scikit-learn: machine learning in Python, *JMLR* 12:2825–30.

Roberts D., Wilford J. & Ghattas O., 2019. Exposed soil and mineral map of the Australian continent revealing the land at its barest. *Nature Communications* 10:5297.

Turcotte D. L., 1992. Fractals, chaos, self-organized criticality and tectonics. *Terra Nova* 4:4–12.

Wilford J., 2020. Revised weathering intensity model of Australia. In: Czarnota K., et al. (eds.), *Exploring for the Future: extended abstracts*, Geoscience Australia, Canberra, 1–4.

Wilford J. & Roberts D., 2020. Enhanced bare earth Landsat imagery for soil and lithological modelling. In: Czarnota K., et al. (eds.), *Exploring for the Future: extended abstracts*, Geoscience Australia, Canberra, 1–6.

Wilford J., Caritat de P. & Bui E., 2016. Predictive geochemical mapping using environmental correlation. *Applied Geochemistry* 66:275–88.

Wilford J., Basak S. & Lindsay J., 2020. Multiscale topographic position image of the Australian continent. In: Czarnota K., et al. (eds.), *Exploring for the Future: extended abstracts*, Geoscience Australia, Canberra, 1–4.