Model Creation and Validation for the Social Vulnerability Index
Training and Building Traditional Random Forest Models

Thesis for a Master of Public Health, Epidemiology

Nathan Garcia-Diaz

Brown University, School of Public Health

August 30, 2024

Contents

Statement of Purpose **3**
- Defining Hyperparameters 3
- Defining: `SpatialML` 3
- Defining: Out of Bag Mean Error Rate 3
- Defining: Partially Spatial Nest-Cross Validation Method 4

Outline of Model Building Process **5**

Note: the table of contents acts as in-document hyperlinks

Statement of Purpose

The purpose of the file is to build is to build a multiple traditional random forest model (RF) and determine the best preforming model. The hyperparameters of the best model will then be used in a geographically weighted random forest model (GWFRF), which is preformed in the upsequent file. The following two sentence provide a overarching description of the two models. In a RF model, each tree in the forest is built from a different bootstrap sample of the training data, and at each node, a random subset of predictors (features) is considered for splitting, rather than the full set of predictors. A GWRF model expands on this concept by incorporating spatial information by weighting the training samples based on their geographic proximity to the prediction location. The splitting process in a RF model is determined by the mean squared error and in a GWRF is influenced by the spatial weights (i.e., weighted mean squared error), which adjust the contribution of each sample based on its geographic distance. Lastly, the feature importance plots will be generated for the final, and local feature importance plots will also be created.

Defining Hyperparameters

In James et al 2021, Ch 8.2.2 Random Forests, James et al 2023, Ch 15.2 Definition of Random Forests and Garson 2021, Ch 5 Random Forest, the hyperparameters that are shared between the traditional RF and the geographically-weighted RF models include:

- **Number of randomly selected predictors**: This is the number of predictors (p) considered for splitting at each node. It controls the diversity among the trees. A smaller m leads to greater diversity, while a larger m can make the trees more similar to each other.
  - for regression this defaults to $p/3$, where p is the total of predictor variables
- **Number of trees**: This is the total number of decision trees in the forest (m). More trees generally lead to a more stable and accurate model, but at the cost of increased computational resources and time.
  - for the `randomForest::randomForest()`, this defaults to 500

Additionally, GWRF involves an extra tuning spatial parameters:

- **Bandwidth parameter**: This controls the influence of spatial weights, determining how quickly the weight decreases with distance. A smaller bandwidth means only very close samples have significant influence, while a larger bandwidth allows more distant samples to also contribute to the model.

Defining: `SpatialML`

Georganos et al (2019) created the `package(SpatialML)`, and subsequently the tuning is made possible by the `SpatialML::grf.bw()` function. The function uses an exhaustive approach (i.e., it tests sequential nearest neighbor bandwidths within a range and with a user defined step, and returns a list of goodness of fit statistics).

Defining: Out of Bag Mean Error Rate

In Garson 2021, Ch 5 Random Forest, Garson teaches Random Forest Models by using `randomForest::randomForest()`, and in chapter 5.5.9 (pg. 267), he provides methods for tuning both of these parameters simultaneously using the Out of Bag MSE Error Rates. This value is a measure of the prediction error for data points that were not used in training

each tree, hence this value is unique to ensemble methods. It is mathematically expressed as OOB Error Rate $= \frac{1}{n}\Sigma_{i=1}^{N}(y_i - \hat{y}_i^{\text{OOB}})^2$. $\hat{y}_i^{\text{OOB}}$ is the OOB prediction for the i-th observation, which is obtained by averaging the predictions from only those trees that did not include i in their bootstrap sample. To provide a high-level summary, since each tree in a Random Forest is trained on a bootstrap sample (a random sample with replacement) of the data, approximately one-third of the data is not used for training each tree. This subset of data is referred to as the "out-of-bag" data for that tree, and this value is calculated using the data points that were not included in the bootstrap sample used to build each tree. The code in this file has been modified so that cross validation is implemented to ensure consistency across the models, and as such the only difference across models is the metric and the type of nested cross validation being used.

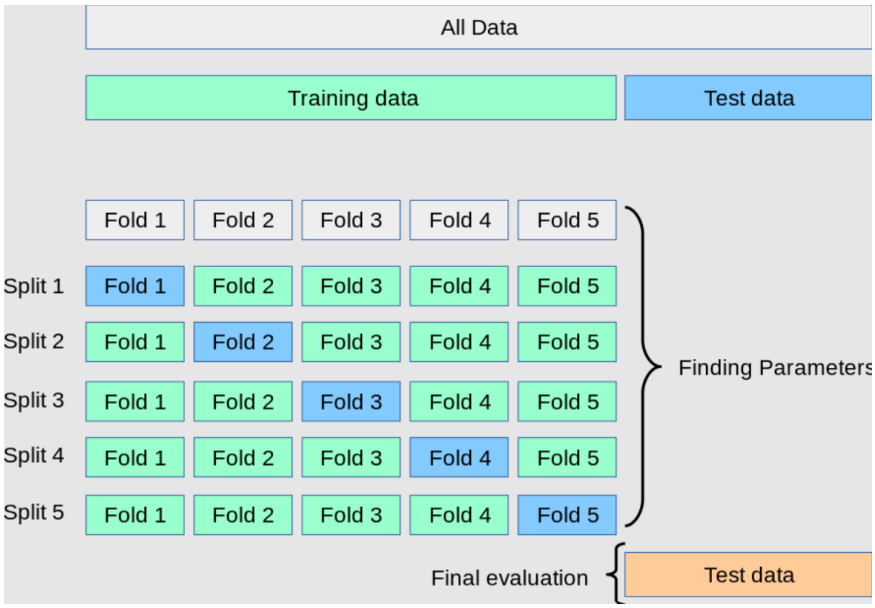
Defining: Partially Spatial Nest-Cross Validation Method

All models will be validated and tuned with a nested cross-validation, a technique used to assess the performance of a model and tuning hyperparameters. It helps to avoid over fitting and provides an unbiased estimate of model performance. A spatial nested cross-validation is a two-level cross-validation procedure designed to evaluate a model's performance and tune its hyperparameters simultaneously. A nested cross-validation is a method that revolves around an outer and liner loop. An example of the workflow include:

- Split the data into "outer_k" folds defined by spatial hierarchical clustering.
- For each fold in the outer loop:
  - Use "outer_k - 1" folds for training.
  - Apply the inner cross-validation on this training set to tune hyperparameters.
  - Evaluate the performance of the model with the selected hyperparameters on the held-out test fold.
- Average the performance metrics across all outer folds to get an overall estimate

---

A visual description of the method, which can be in Jian et al (2022) - Rapid Analysis of Cylindrical Bypass Flow Field Based on Deep Learning Model.



- Outer Cross-Validation Loop:
  - Purpose: To estimate the model's performance on unseen data and provide a more reliable measure of how well the model generalizes to new data.
  - Procedure: The data set is divided into several folds (e.g., 5 or 10). In each iteration, one fold is used as the test set, and the remaining folds are used for training and hyper parameter tuning. Folds are defined by hierarchical clustering. This process is repeated for each fold, ensuring that every data point is used for testing exactly once.
- Inner Cross-Validation Loop:
  - Purpose: To select the best hyperparameters for the model.
  - Procedure: Within each training set from the outer loop, a further cross-validation is performed. This involves splitting the training data into additional folds (e.g., 3 or 5). The model is trained with various hyper parameter combinations on these inner folds, and the performance is evaluated to choose the optimal set of hyperparameters.

Outline of Model Building Process

5 RF models will be built, and they differ based on the different hyperparameters: (1) default settings; (2) Exhaustive Grid Search with RMSE as Metric and Traditional Nested Cross Validation, (3) Exhaustive Grid Search with RMSE as Metric and Partially Spatial Nested Cross Validation, (4)Iterative Grid with Out of Bag Mean Squared Error as Metric and Traditional Nested Cross Validation (i.e., Modified Code from Garson 2021), (5) Iterative Search with Out of Bag Mean Squared Error as Metric and Partially Spatial Nested Cross Validation. For each model, MAE, RMSE, and R^2 will be calculated and the hyperparameters of the best model will continue onto the GWRF. To provide points of comparison in the GWRF, two additional models will be created. Thus, two GWRF models will be created: (1) default *mtry* and *ntrees* with optimized *bandwidth parameter*, and (2) using the previously defined best hyperparameters. The same model evaluation metrics will be compared in addition to calculating the residual autocorrelation.