### Supplementary Methods

In this section, we formally describe the tested machine learning models.

**Linear regression (LR)** LR is one of the simplest parametric methods to model the relationship between the input variables (or explanatory variables) and the dependent variable (or response variable). The main assumptions are that there is a linear relationship between the input and output variables and homoscedasticity, i.e. the residuals (or error terms) must possess a constant variance, regardless of the independent variable. Such a linear relationship is represented in a linear equation as follows:

,

where *n* is the number of input variables, is the i-th feature, and is the i-th model parameter. The coefficients can be estimated in an efficient way using Least Squares method.

**Gaussian Process Regression (GPR)** GPR, also known as Kriging, can be thought of as a generalization of the Gaussian probability distribution to infinitely many variables. A Gaussian Process (GP) is a Gaussian random function, and is fully specified by a mean function and a covariance function , as follows:

.

To define an individual GP, one needs to choose a form for and . In most applications there is no prior knowledge about the mean function, , so, by simplicity, and because GPs are, by definition, a linear combination of random variables with Normal Distribution, this is commonly assumed to be zero. The covariance function, , can be in general any function that takes any two arguments, such that k(x, x0 ) generates a nonnegative definite covariance matrix *K*. There are many possible covariance functions, but one that is most frequently used is the squared exponential covariance function, also known as Radial Basis Function (RBF):

.

It is easy to see that for such an equation, the covariance between any two inputs is really close to one if the inputs are close to each other, and decreases exponentially as the distance between the inputs increases. Here, and *l* are the hyperparameters of the kernel which are dataset-dependent and are estimated during the training phase.

**Support Vector Regression (SVR)** In the SVR, the purpose is to find a function f(x) that has at most ε deviation from any output for all the training data, and simple as possible. In its linear formulation, a SVR model is similar to a linear model, but the coefficients are estimated as follows:

subject to

,

,

, ,

where , are slack variables to assure the feasibility of the problem and determines the trade-off between the flatness of *f* and the amount up to which deviations larger than ε are tolerated. To cope with non-linear effect, it is possible to introduce a non-linear kernel function , where is a non-linear map from the original space of the input variables and a new feature space . In this case, the learning function becomes

.

**K-nearest Neighbors regression (KNR)** KNR belongs to the family of instance-based learning algorithms, where the training instances are stored in memory without explicitly learning a model. The training instances are processed in the prediction phase. For each new data instance, a query is made and, in its simplest case, the query returns the k nearest data points to the new instance, based on some distance or similarity metric. In KNR, the output is determined by averaging the outputs of the k-neighboring data points:

.

**Regression Trees.** Regression trees are supervised learning models, where the main idea consists of breaking down a dataset into smaller and smaller subsets while at the same time an associated decision tree is incrementally developed. The final result is a tree with decision nodes and leaf nodes. A decision node has two or more branches, each representing values for the attribute tested. Leaf node represents a decision on the numerical target. Usually, a decision tree is built top-down from a root node and involves the partitioning the data into subsets (e.g. rectangles) that contain instances with similar values, and fitting a simple model (like a constant) in each one. The corresponding regression model becomes:

,

where , if and 0 otherwise, and is just the average of in region :

.

The partition of the data is usually done iteratively with a greedy algorithm, in which, at each step, the best pair *(j,s)* is chosen, where *j* is the splitting variable*,* and *s* is the splitting point*.*

**Random Forest (RF)** is an ensemble learning method, where multiple Regression Trees (Forest) are trained in parallel without any interaction between each other. Each Tree is usually trained in a different bootstrap version of the original dataset, selecting a subset of variables at random from the original variables. The output of the model results from the average of the predictions of the various trees.

**Gradient Boosting** (GB) GB is another ensemble method, in which multiple Regression Trees are built. Unlike random forests, the decision trees in gradient boosting are built additively; i.e. each decision tree is built one after another. Each new tree is built to improve the previous trees (boosting). The gradient part of gradient boosting comes from minimizing the gradient of the loss function as the algorithm builds each tree.