Gravity Adaptive Model (GAM)

Henri Vuorinen

July 5, 2025

1 Core concept

The GAM is build with idea to use two phases for the algorithm.

1.1 Gravity-Based clustering (phase 1

This aims to identify the natural clusters within the data by simulating a gravitational attraction between data points.

1.1.1 Data Points

Let $X = \{\ x_1, x_2, x_3, ..., x_n\}$ be a set of N input data points, where each $x_i \in \mathbb{R}^D$, where $\mathbb{R}^D$ is D-dimentional feature vector.

1.1.2 Gravitational force

This is a theoretical force or influence exerted by point x_j on x_i is moduled using a Gaussian (Radial Basis Function) kernel, which ensures highly localized interaction

$$I(x_i, x_j) = \exp(-\gamma ||x_i - x_j||^2)$$

where $||x_i - x_j||^2$ is the squared Euclidean distance between x_i and x_j, and $\gamma > 0$ is the bandwith parameter, which is used to control the spread of the influence. A larger γ means more localized force, and vice versa.

1.1.3 Iterative movement

Points iteratively move towards regions of higher density. For each point x_i, its new position is calculated as the weighted average of all other points' current positions, where weights are determined by the gravitational influence:

$$x_i^{new} = \frac{\sum_{j \neq i} I(x_i^{current}, x_j^{current}) * x_j^{current}}{\sum_{j \neq i} I(x_i^{current}, x_j^{current})}$$

This iterative process allows all points to converge towards modes of density in the feature space

1.1.4 Cluster Identification

After k iteration, points that have converged to sufficient close locations are grouped into cluster using Agglomerative Clustering.

1.2 Adaptive Local Modeling (phase 2)

When clusters $C_1, C_2, ..., C_M$ are identified, a specialized regression model is trained for each cluster.

1.2.1 Local model training

For each cluster C_m, a subset of the original data $D_m = \{(x_i, y_i) \mid x_i \in C_m\}$ is used to train a local model f_m.

Each f_m is a polynomial regression model. This is achieved by first transforming the input features x using 'PolynomialFeatures' to generate higher-order and interaction terms, resulting in $\phi(x)$.

To prevent overfitting and wild extrapolation, the "Ridge" regularization is applied to these polynomial regressions. The local model $f_m(x)$ then takes the form:

$$f_m(x) = w_m^T \phi(x)$$

The coefficients w_m are learned by minimizing the Ridge objective function:

$$\min_{w_m} ||Y_m - \Phi_m w_m||^2 + \lambda ||w_m||^2$$

Here, Y_m is the vector of target values for data points in cluster C_m, Φ_m is the design matrix where each row corresponds to the polynomial features $\phi(x_i)$ for $x_i \in C_m$, and λ is the regularization strength ($\lambda \geq 0$), a hyperparameter that controls the balance between fitting the training data well and keeping the model weights small.

1.3 Smooth Blending of Predictions

For a new, unseen data points x_{new}, the final prediction is a weighed sum of the predictions from all local models.

1.3.1 Cluster Centers

Each cluster C_m is represented by a center c_m, typically the mean of the original data points within that cluster.

1.3.2 Blending Weights

The contribution of each local model f_m to the final prediction is determined by a blending weights $B_m(x_{new})$, which measures the proximity of x_{new} to the cluster center. This also uses Gaussian kernel:

$$\bar{B}_m(x_{new}) = \exp(-\alpha ||x_{new} - c_m||^2)$$

where $\alpha > 0$, and it represents a blending bandwith parameter.

1.3.3 Normilzed Blending Weights

These weights are normalized across all clusters to sum to 1:

$$\bar{B}_m(x_{new}) = \frac{\bar{B}_m(x_{new})}{\sum_{k=1}^{M} B_k(x_{new})}$$

1.3.4 final Prediction

The overall predictions $\hat{y}(x_{new})$ is the blended sum:

$$\hat{y}(x_{new}) = \sum_{m=1}^{M} \bar{B}(x_{new}) * f_m(x_{new})$$

This ensures a continuous and smooth blending prediction surface that adaptive responds to the local data structure.