SUPPLEMENTAL MATERIAL

Data S1.

Supplemental Methods and Results

eXtreme Gradient Boosting (XGBoost)

XGBoost is a highly effective scalable machine learning system for tree boosting. This transforms several weak classifiers into a strong classifier for a better performance through an iterative computation of weak classifiers. The scalability of XGBoost is due to several important systems and algorithmic optimizations including a novel tree learning algorithm, a theoretically justified weighted quantile sketch procedure and parallel and distributed computing (1,2). Tree boosting, an effective ensemble learning algorithm, can transform several weak classifiers into a strong classifier for better performance.

Let $D = \{(x_i, y_i)\}(|D| = n, x_i \in \mathbb{R}^m, y_i \in \mathbb{R}^n)$ represents a database with n examples and m features. A tree boosting model output $\hat{y}_i$ with K trees is defined as follows:

$$\hat{\mathrm{y}}_{\mathrm{i}} = \sum_{k=1}^{K} f_k(x_i), f_k \in F \qquad [1]$$

where $F = \{f(x) = \omega_q(x)\}(q: \mathbb{R}^m \to T, \omega \in \mathbb{R}^T)$ is the space of regression or classification trees (also known as CART). Each $f_k$divides a tree into structure part q and leaf weights part ω. Here T denotes the number of leaves in the tree. The set of function f_k in the tree model can be learned by minimizing the following objective function:

$$O = \sum_i l(\hat{y}_i, y_i) + \sum_k \Omega(f_k) \quad [2]$$

The first term l in Eq. [2] is a training loss function which measures the distance between the prediction $\hat{y}_i$ and the object y_i. The second term Ω in Eq. [2] represents the penalty term of the tree model complexity. Tree boosting model whose objective function is Eq. [2] cannot be optimized through traditional optimization methods in Euclidean space. Gradient Tree Boosting is an improved version of tree boosting by training tree model in an additive manner, which means the prediction of the *t*-th iteration $\hat{y}^{(t)} = \hat{y}^{(t-1)} + f_t(x)$. The objective function in *t*-th iteration is changed as:

$$O^{(t)} = \sum_{i=1}^{n} l\left(y_i, \hat{y}_i^{(t-1)} + f_t(x_i)\right) + \Omega(f_t) \quad [3]$$

XGBoost approximates Eq. [3] by utilizing the second order Taylor expansion and the final objective function at step *t* can be rewritten as:

$$O^{(t)} \simeq \tilde{O}^{(t)} = \sum_{i=1}^{n} \left[l\left(y_i, \hat{y}_i^{(t-1)}\right) + g_i f_t(x_i) + \frac{1}{2} h_i f_t^2(x_i) \right] + \Omega(f_t) \quad [4]$$

where g_i and h_i are first and second order gradient statistics on the loss function, and $\Omega(f) = \gamma T + \frac{1}{2}\lambda\|\omega\|^2$ in XGBoost.

Denote $I_j = \{i|q(x_i) = j\}$ as the instance set of leaf j, after removing the constant terms and expanding Ω, Eq. [4] can be simplified as:

$$\tilde{O}^{(t)} = \sum_{j=1}^{T} \left[\left(\sum_{i \in I_j} g_i \right) \omega_j + \frac{1}{2} \left(\sum_{i \in I_j} h_i + \lambda \right) \omega_j^2 \right] + \gamma T \quad [5]$$

The solution weight ω_j^* of leaf j for a fixed tree structure $q(x)$ can be obtained by applying the following equation:

$$\omega_j^* = -\frac{\sum_{i\in I_j} g_i}{\sum_{i\in I_j} h_i + \lambda} \quad [6]$$

After substituting ω_j^* into Eq. [5], there exists:

$$\tilde{O}(q) = -\frac{1}{2}\sum_{j=1}^{T} \frac{\left(\sum_{i\in I_j} g_i\right)^2}{\sum_{i\in I_j} h_i + \lambda} + \gamma T \quad [7]$$

Define Eq. [7] as a scoring function to evaluate the tree structure $q(x)$ and find the optimal tree structures for classification. However, it is impossible to search the whole possible tree structures q in practice. A greedy algorithm starts from a single leaf, and iteratively adds branches to grow the tree structure. Whether adding a split to the existing tree structure can be decided by the following function (1,2):

$$O_{split} = \frac{1}{2}\left[\frac{\left(\sum_{i\in I_L} g_i\right)^2}{\sum_{i\in I_L} h_i + \lambda} + \frac{\left(\sum_{i\in I_R} g_i\right)^2}{\sum_{i\in I_R} h_i + \lambda} - \frac{(\sum_{i\in \mathrm{I}} g_i)^2}{\sum_{i\in \mathrm{I}} h_i + \lambda}\right] - \gamma \quad [8]$$

Where I_L and I_R are the instance sets of left and right nodes after the split and $I = I_L \cup I_R$. XGBoost is a fast implementation of GB algorithm, which has the advantages of fast speed and high accuracy.

5-fold cross validation tests. The 5-fold cross validation scheme divided the training

sample into non-overlapped five partitions (Online Figure 1). Each partition was rotated to be the test set and the rests are used as training data. The accuracy was calculated by averaging the accuracies over five tests. To reduce variability, multiple rounds of cross-validation were performed and averaged.

Figure S1. 5-fold cross validation.

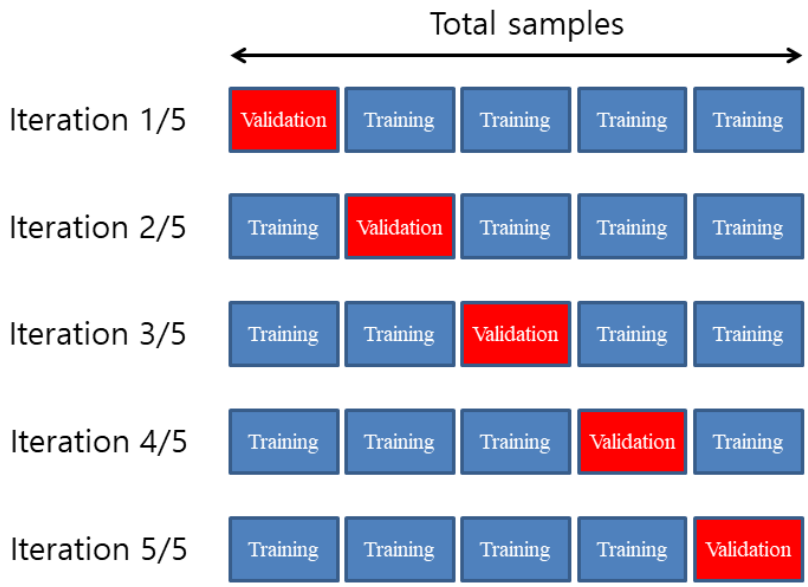
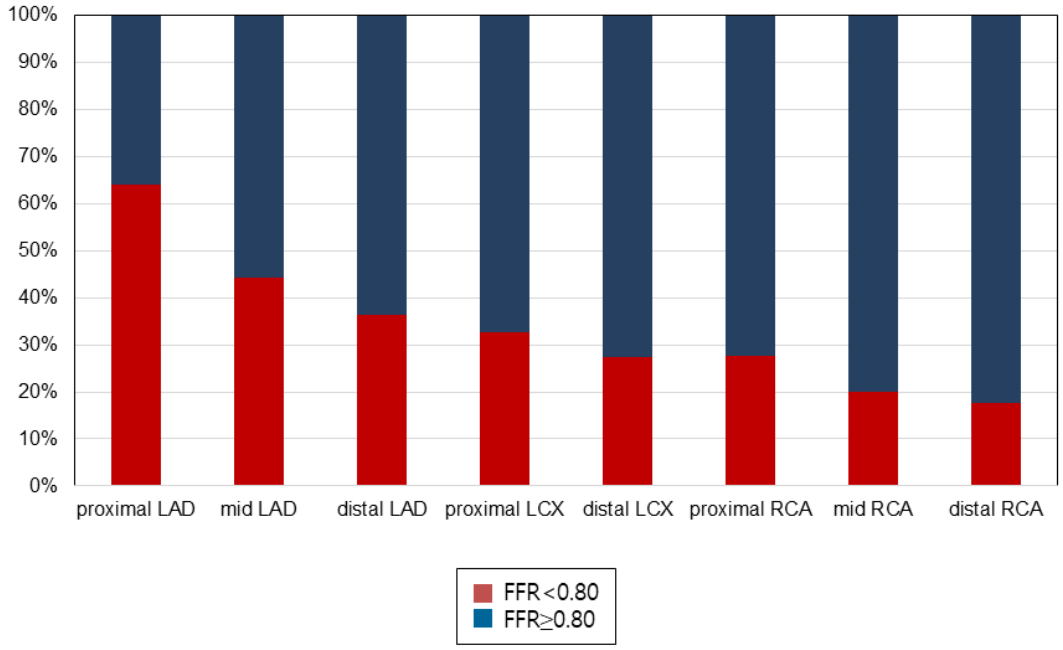


Figure S2. Frequency of FFR<0.80 according to the involved segment.



Supplemental References:

1. Chen T, Guestrin C. Xgboost: a scalable tree boosting system. In: Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 785–794. ACM (2016).
2. Xudie R, Haonan G, Shenghong L. A Novel Image Classification Method with CNN-XGBoost Model. Digital Forensics and Watermarking. 2017;378-90.