Supplementary Text

Posterior Distribution for the Global Importance Scores in the GOALS

In this subsection, we derive the full joint distribution for the sample means of the GOALS operator to conduct posterior inference on the global importance of features in a nonlinear model. As was done in the main text, consider a data set with N individuals. We have a vector response variable $\mathbf{y}$ of length N and an $N \times J$ design matrix $\mathbf{X}$ with J denoting the number of features. For consistency, we will demonstrate the properties of GOALS using a weight-space Gaussian process regression model

$$\mathbf{y} = \boldsymbol{f} + \boldsymbol{\varepsilon}, \qquad \boldsymbol{f} \sim \mathcal{N}(\mathbf{0}, \mathbf{K}), \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}) \tag{S1}$$

where $\boldsymbol{f} = [f(\mathbf{x}_1), \ldots, f(\mathbf{x}_N)]$ is a normally distributed random variable with mean vector $\mathbf{0}$ and a covariance matrix $\mathbf{K}$ defined by some nonlinear kernel function.

In the main text, we defined a set of perturbed features $\mathbf{X} + \boldsymbol{\Xi}^{(j)}$, where $\boldsymbol{\Xi}^{(j)}$ is an $N \times J$ matrix with rows $\boldsymbol{\xi}^{(j)}$ equal to all zeros except for the j-th element which we set to be a vector of some positive constant ξ. We then defined a length N vector $\mathbf{g}^{(j)} = [f(\mathbf{x}_1 + \boldsymbol{\xi}^{(j)}), \ldots, f(\mathbf{x}_N + \boldsymbol{\xi}^{(j)})]$ where we showed that the joint distribution for the GOALS operator $\boldsymbol{\delta}^{(j)} = \boldsymbol{f} - \mathbf{g}^{(j)}$ (conditional on the data) can be written as the following

$$\begin{bmatrix} \boldsymbol{\delta}^{(1)} \\ \vdots \\ \boldsymbol{\delta}^{(J)} \end{bmatrix} \Bigg| \, \mathbf{y} \sim \mathcal{N}\left(\begin{bmatrix} \left[\mathbf{K} - \left(\mathbf{B}^{(1)}\right)^\intercal\right] \mathbf{A}^{-1}\mathbf{y} \\ \vdots \\ \left[\mathbf{K} - \left(\mathbf{B}^{(J)}\right)^\intercal\right] \mathbf{A}^{-1}\mathbf{y} \end{bmatrix}, \begin{bmatrix} \boldsymbol{\Sigma}^{(1)} & \cdots & \boldsymbol{\Sigma}^{(1,J)} \\ \vdots & \ddots & \vdots \\ \boldsymbol{\Sigma}^{(J,1)} & \cdots & \boldsymbol{\Sigma}^{(J)} \end{bmatrix} \right) \tag{S2}$$

where, in addition to previous notation, $\mathbf{A} = \mathbf{K} + \sigma^2\mathbf{I}$ is the marginal variance of the response vector $\mathbf{y}$; $\mathbf{B}^{(j)}$ is the covariance between $\boldsymbol{f}$ and $\mathbf{g}^{(j)}$ using the original matrix $\mathbf{X}$ and the perturbed matrix $\mathbf{X}+\boldsymbol{\Xi}^{(j)}$; $\mathbf{C}^{(j)}$ is the variance of $\mathbf{g}^{(j)}$ using the perturbed matrix $\mathbf{X} + \boldsymbol{\Xi}^{(j)}$; and $\mathbf{D}^{(j,l)}$ is the covariance between $\mathbf{g}^{(j)}$ and $\mathbf{g}^{(l)}$ having perturbed the j-th and l-th feature, respectively. Furthermore, we define

$$\boldsymbol{\Sigma}^{(j)} = \mathbf{K}\mathbf{A}^{-1}\mathbf{K} - \left(\mathbf{B}^{(j)}\right)^\intercal\mathbf{A}^{-1}\mathbf{B}^{(j)} - \left[\left(\mathbf{B}^{(j)}\right)^\intercal - \left(\mathbf{B}^{(j)}\right)^\intercal\mathbf{A}^{-1}\mathbf{K} + \mathbf{B}^{(j)} - \mathbf{K}\mathbf{A}^{-1}\mathbf{B}^{(j)}\right]$$

$$\boldsymbol{\Sigma}^{(j,l)} = \mathbf{K} - \mathbf{K}\mathbf{A}^{-1}\mathbf{K} + \mathbf{D}^{(j,l)} - \left(\mathbf{B}^{(j)}\right)^\intercal\mathbf{A}^{-1}\mathbf{B}^{(l)} - \left[\left(\mathbf{B}^{(j)}\right)^\intercal - \left(\mathbf{B}^{(j)}\right)^\intercal\mathbf{A}^{-1}\mathbf{K} + \mathbf{B}^{(l)} - \mathbf{K}\mathbf{A}^{-1}\mathbf{B}^{(l)}\right].$$

Altogether, the above represents a joint conditional distribution from which one can sample estimates of each $\boldsymbol{\delta}^{(j)}$ and obtain local interpretability. To investigate the global interpretability of each feature, one can use the sample mean across the local explanations for all observations where $\bar{\delta}^{(j)} = \mathbf{1}^\intercal\boldsymbol{\delta}^{(j)}/N$ with $\mathbf{1}$ being a length N vector of ones. These global interpretability scores have the following joint distribution

$$\begin{bmatrix} \bar{\delta}^{(1)} \\ \vdots \\ \bar{\delta}^{(J)} \end{bmatrix} \Bigg| \, \mathbf{y} \sim \mathcal{N}\left(\begin{bmatrix} \mathbf{1}^\intercal\left[\mathbf{K} - \left(\mathbf{B}^{(1)}\right)^\intercal\right] \mathbf{A}^{-1}\mathbf{y}/N \\ \vdots \\ \mathbf{1}^\intercal\left[\mathbf{K} - \left(\mathbf{B}^{(J)}\right)^\intercal\right] \mathbf{A}^{-1}\mathbf{y}/N \end{bmatrix}, \begin{bmatrix} \mathbf{1}^\intercal\boldsymbol{\Sigma}^{(1)}\mathbf{1}/N^2 & \cdots & \mathbf{1}^\intercal\boldsymbol{\Sigma}^{(1,J)}\mathbf{1}/N^2 \\ \vdots & \ddots & \vdots \\ \mathbf{1}^\intercal\boldsymbol{\Sigma}^{(J,1)}\mathbf{1}/N^2 & \cdots & \mathbf{1}^\intercal\boldsymbol{\Sigma}^{(J)}\mathbf{1}/N^2 \end{bmatrix} \right). \tag{S3}$$

Therefore, to simulate from the posterior distribution of the sample means, one simply needs to compute the following closed form equations for the first and second moments

$$\begin{aligned} \mathbb{E}\left[\bar{\delta}^{(j)}\right] &= \mathbf{1}^\intercal\left[\mathbf{K} - \left(\mathbf{B}^{(j)}\right)^\intercal\right] \mathbf{A}^{-1}\mathbf{y}/N \\ \mathbb{V}\left[\bar{\delta}^{(j)}\right] &= \left(\lambda + \alpha_{jj} - 2\psi_j\right)/N^2 \\ \mathbb{V}\left[\bar{\delta}^{(j)}, \bar{\delta}^{(l)}\right] &= \left(\lambda + \alpha_{jl} - \psi_j - \psi_l\right)/N^2 \end{aligned} \tag{S4}$$

where $\lambda = \mathbf{1}^\intercal\mathbf{K}\mathbf{1} - \mathbf{1}^\intercal\mathbf{K}\mathbf{A}^{-1}\mathbf{K}\mathbf{1}$; $\alpha_{jl} = \mathbf{1}^\intercal\mathbf{D}^{(j,l)}\mathbf{1} - \mathbf{1}^\intercal\left(\mathbf{B}^{(j)}\right)^\intercal\mathbf{A}^{-1}\mathbf{B}^{(l)}\mathbf{1}$; and $\psi_j = \mathbf{1}^\intercal\mathbf{B}^{(j)}\mathbf{1} - \mathbf{1}^\intercal\mathbf{K}\mathbf{A}^{-1}\mathbf{B}^{(j)}\mathbf{1}$, respectively.

Extension of the GOALS to Probabilistic Neural Networks

In this section, we show how the "GlObal And Local Score" (GOALS) operator can be used to determine global and local interpretability in probabilistic neural networks. In contrast to a "standard" neural network, which uses maximum likelihood point-estimates for its parameters, we will assume a model architecture that places a prior distribution over its weights. During training, we will use a learned posterior probability over these weights to compute the posterior predictive distribution. Once again, we consider a general data application where we are given with a response variable $\mathbf{y}$ and an $N \times J$ design matrix $\mathbf{X}$ with J covariates. For this problem, we assume the following hierarchical network architecture to learn the predicted response in the data

$$\mathbf{y} = r^{-1}(\boldsymbol{f}), \qquad \boldsymbol{f} = \mathbf{H}(\boldsymbol{\vartheta})\mathbf{w}, \qquad \mathbf{w} \sim \pi\,, \tag{S5}$$

where $r(\bullet)$ is a link function (which we will assume to be the identity for regression-based tasks), $\boldsymbol{\vartheta}$ is a vector of inner layer weights, and $\boldsymbol{f}$ is a vector of smooth latent values or "functions" that need to be estimated. Here, we use $\mathbf{H}(\boldsymbol{\vartheta}) = h(\mathbf{X}\boldsymbol{\vartheta})$ to denote an $N \times L$ matrix of activations from the penultimate layer (which are fixed given a predetermined activation function $h(\bullet)$, a set of features $\mathbf{X}$, and point estimates for the inner layer weights $\boldsymbol{\vartheta}$), and $\mathbf{w} \sim \pi$ is a L-dimensional vector of weights at the output layer assumed to follow prior distribution π. For simplicity, we omit the bias term in Eq. (S5) that is produced during the training phase. Also note that the structure of the hidden layers in the model above can be of any size or type, provided that we have access to draws of the posterior predictive distribution for the response variables.

The structure of Equation (S5) is motivated by the fact that we are most interested in the posterior distribution of the latent variables $\boldsymbol{f}$. To this end, we follow previous work (Ish-Horowicz et al., 2019) and split the network architecture into three key components: *(i)* an input layer of the original features $\mathbf{X}$, *(ii)* hidden layers $\mathbf{H}(\boldsymbol{\vartheta})$ where parameters are deterministically computed, and *(iii)* the outer layer where the parameters and activations are treated as random variables. As the size of datasets in many application areas continues to grow, it has become common to train neural networks with algorithms that are based on variational Bayes and the stochastic optimization of a variational lower bound (Barber and Bishop, 1998; Graves, 2011; Hinton and Van Camp, 1993). Here, the variational Bayes framework has the additional benefit of providing closed-form expressions for the posterior distribution of the weights in the outer layer $\mathbf{w}$ and, subsequently, the functions $\boldsymbol{f}$.

We will begin by first specifying a prior $\pi(\mathbf{w})$ over the weights and replace the intractable true posterior $p(\mathbf{w} \,|\, \mathbf{y}) \propto p(\mathbf{y} \,|\, \mathbf{w})\pi(\mathbf{w})$ with an approximating family of distributions $q_{\boldsymbol{\phi}}(\mathbf{w})$ where $\boldsymbol{\phi}$ denotes a collection of free parameters. The overall goal of variational inference is to minimize the Kullback-Leibler divergence between the exact and approximate posterior distributions, respectively. This is equivalent to maximizing the so-called evidence lower bound where all parameters can be optimized jointly as follows

$$\underset{\boldsymbol{\phi},\boldsymbol{\vartheta}}{\arg\max}\; \mathbb{E}_{q_{\boldsymbol{\phi}}(\mathbf{w})}\left[\log p(\mathbf{y} \,|\, \mathbf{w}, \boldsymbol{\vartheta})\right] - \eta\, \mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{w}) \,\|\, \pi(\mathbf{w})). \tag{S6}$$

Depending on the chosen variational family, the gradients of the minimized $\mathrm{KL}(q_{\boldsymbol{\phi}}(\mathbf{w}) \,\|\, \pi(\mathbf{w}))$ may be available in closed-form, while gradients of the log-likelihood $\log p(\mathbf{y} \,|\, \mathbf{w}, \boldsymbol{\vartheta})$ are evaluated using Monte Carlo samples and the local reparameterization trick (Kingma et al., 2015). Following this procedure, we obtain an optimal set of parameters for $q_{\boldsymbol{\phi}}(\mathbf{w})$, with which we can sample posterior draws for the outer layer. For simplicity, we will assume isotropic Gaussians as the family of approximating distributions

$$q_{\boldsymbol{\phi}}(\mathbf{w}) = \mathcal{N}(\mathbf{0}, \mathbf{V}), \tag{S7}$$

where $\mathbf{0}$ is vector of zeros and $\mathbf{V}$ is a diagonal covariance matrix. Using Equations (S5) and (S7), we may derive the implied distribution over the latent function values using the affine transformation

$$\boldsymbol{f} \sim \mathcal{N}(\mathbf{0}, \mathbf{H}(\boldsymbol{\vartheta})\mathbf{V}\mathbf{H}(\boldsymbol{\vartheta})^{\intercal}). \tag{S8}$$

While the elements of $\mathbf{w}$ are independent, dependencies in the input data (via the deterministic hidden activations $\mathbf{H}(\boldsymbol{\vartheta}) = h(\mathbf{X}\boldsymbol{\vartheta})$) induce a non-diagonal covariance $\mathbf{K} = \mathbf{H}(\boldsymbol{\vartheta})\mathbf{V}\mathbf{H}(\boldsymbol{\vartheta})^\intercal$ between the elements of the latent function $\boldsymbol{f}$.

Similar to what was shown with Gaussian process regression, to perform variable importance with the GOALS measure, we can define perturbed features $\mathbf{X} + \boldsymbol{\Xi}^{(j)}$, where $\boldsymbol{\Xi}^{(j)}$ is an $N \times J$ matrix with rows $\boldsymbol{\xi}^{(j)}$ equal to all zeros except for the j-th element which we set to be a vector of some positive constant ξ, and we can also define $\mathbf{g}^{(j)} = [f(\mathbf{x}_1 + \boldsymbol{\xi}^{(j)}), \ldots, f(\mathbf{x}_N + \boldsymbol{\xi}^{(j)})]$. An analogous way to think about variable importance is to consider the expected change in the mean response given a ξ-unit increase in the corresponding covariate (holding all else constant). This again leads to the natural quantity to understand the importance of each variable by examining $\boldsymbol{\delta}^{(j)} = \boldsymbol{f} - \mathbf{g}^{(j)}$. Using Eq. (S8), the posterior mean of $\boldsymbol{\delta}^{(j)}$ to perform local variable importance in neural networks also takes on the general form

$$\mathbb{E}\left[\boldsymbol{\delta}^{(j)} \,|\, \mathbf{y}\right] = \left[\mathbf{K} - \left(\mathbf{B}^{(j)}\right)^\intercal\right] \mathbf{A}^{-1}\mathbf{y}. \tag{S9}$$

There are a two main differences in this formulation when working with neural networks. First, the marginal variance $\mathbf{A} = \mathbf{K} + \sigma^2\mathbf{I}$ can be estimated by using $\sigma^2 \approx \mathbb{V}[\mathbf{y} - \mathbf{H}(\boldsymbol{\vartheta})\mathbf{w}]$ which approximates the variance of residual training error in the penultimate layer (e.g., Demetci et al., 2021). Second, we must find the covariance between $\boldsymbol{f}$ and $\mathbf{g}^{(j)}$ using the original matrix $\mathbf{X}$ and some perturbed matrix $\mathbf{X} + \boldsymbol{\Xi}^{(j)}$. To do so, note that using the perturbed matrix as the input to an already trained neural network (i.e., meaning model weights have already been estimated and frozen) allows us to directly estimate new hidden neurons $\mathbf{H}^{(j)}(\boldsymbol{\vartheta}) = h[(\mathbf{X} + \boldsymbol{\Xi}^{(j)})\boldsymbol{\vartheta}]$. This implies that the covariance between $\boldsymbol{f}$ and $\mathbf{g}^{(j)}$ can be written as a function of $\mathbf{H}(\boldsymbol{\vartheta})$ and $\mathbf{H}^{(j)}(\boldsymbol{\vartheta})^\intercal$, respectively, where $\mathbf{B}^{(j)} = \mathbf{H}(\boldsymbol{\vartheta})\mathbf{V}\mathbf{H}^{(j)}(\boldsymbol{\vartheta})^\intercal$. Lastly, as we did in the main text, one can take the sample means of the local importance values to get a measurement of global importance.

Scalable Computation for GOALS in Linear Regression

In this section, we show that the "GlObal And Local Score" (GOALS) operator can also be efficiently computed in a linear regression framework. As in the previous sections, we will assume that we have access to a length N vector response variable $\mathbf{y}$ and an $N \times J$ design matrix $\mathbf{X}$ with J denoting the number of features. Next, consider a standard linear model

$$\mathbf{y} = \boldsymbol{f} + \boldsymbol{\varepsilon}, \qquad \boldsymbol{f} = \mathbf{X}\boldsymbol{\beta}, \qquad \boldsymbol{\varepsilon} \sim \mathcal{N}(\mathbf{0}, \sigma^2\mathbf{I}) \tag{S10}$$

where the function to be estimated $\boldsymbol{f}$ is assumed to be a linear combination of features in $\mathbf{X}$ and their respective effects denoted by the J-dimensional vector $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_J)$ additive coefficients, $\boldsymbol{\varepsilon}$ is a normally distributed error term with mean zero and scaled variance term σ^2, and $\mathbf{I}$ denotes an $N \times N$ identity matrix. For convenience, we will assume that the outcome variable $\mathbf{y}$ has been mean-centered and standardized. The key identity in this section is that we can equivalently represent the regression in Eq. (S10) as a Gaussian process model with a linear gram kernel where the covariance matrix is written as $\mathbf{K} = \mathbf{X}\mathbf{X}^\intercal$. Once again, we will work with the posterior mean of $\boldsymbol{\delta}^{(j)}$ of the GOALS measure which again takes on the same general form presented in Eq. (S9). Since we are working within the context of linear regression, the covariance between $\boldsymbol{f}$ and $\mathbf{g}^{(j)}$ simplifies to the following

$$\mathbf{B}^{(j)} = k(\mathbf{X}, \mathbf{X} + \boldsymbol{\Xi}^{(j)}) = \mathbf{X}(\mathbf{X} + \boldsymbol{\Xi}^{(j)})^\intercal = \mathbf{K} + \mathbf{X}\boldsymbol{\Xi}^{(j)\intercal}. \tag{S11}$$

Note that, because $\boldsymbol{\Xi}^{(j)\intercal}$ is a matrix of all zeros except for the j-th column, we can use Eq. (S11) to simplify the form of Eq. (S9) as the following

$$\mathbb{E}\left[\boldsymbol{\delta}^{(j)} \,|\, \mathbf{y}\right] = -\mathbf{X}\boldsymbol{\Xi}^{(j)\intercal}\mathbf{A}^{-1}\mathbf{y} = \xi\mathbf{x}_{\bullet j}\mathbf{1}^\intercal\mathbf{A}^{-1}\mathbf{y} \tag{S12}$$

where $\mathbf{1}$ is a length N vector of ones and $\mathbf{x}_{\bullet j}$ is the j-th column in the design matrix $\mathbf{X}$. The main summary is twofold. First, the magnitude of the GOALS operator is directly dependent on the value of ξ. Here, computing $\boldsymbol{\delta}^{(j)}$ with $\xi = 2$ will lead to importance scores that are twice as large as when the operator is computed with $\xi = 1$. The second main takeaway is that the computation of Eq. (S9) only relies on linear operations after an initial pre-computation of the term $\mathbf{1}^\intercal \mathbf{A}^{-1}\mathbf{y}$ which can be sped up using matrix decompositions.

Supplementary Figures

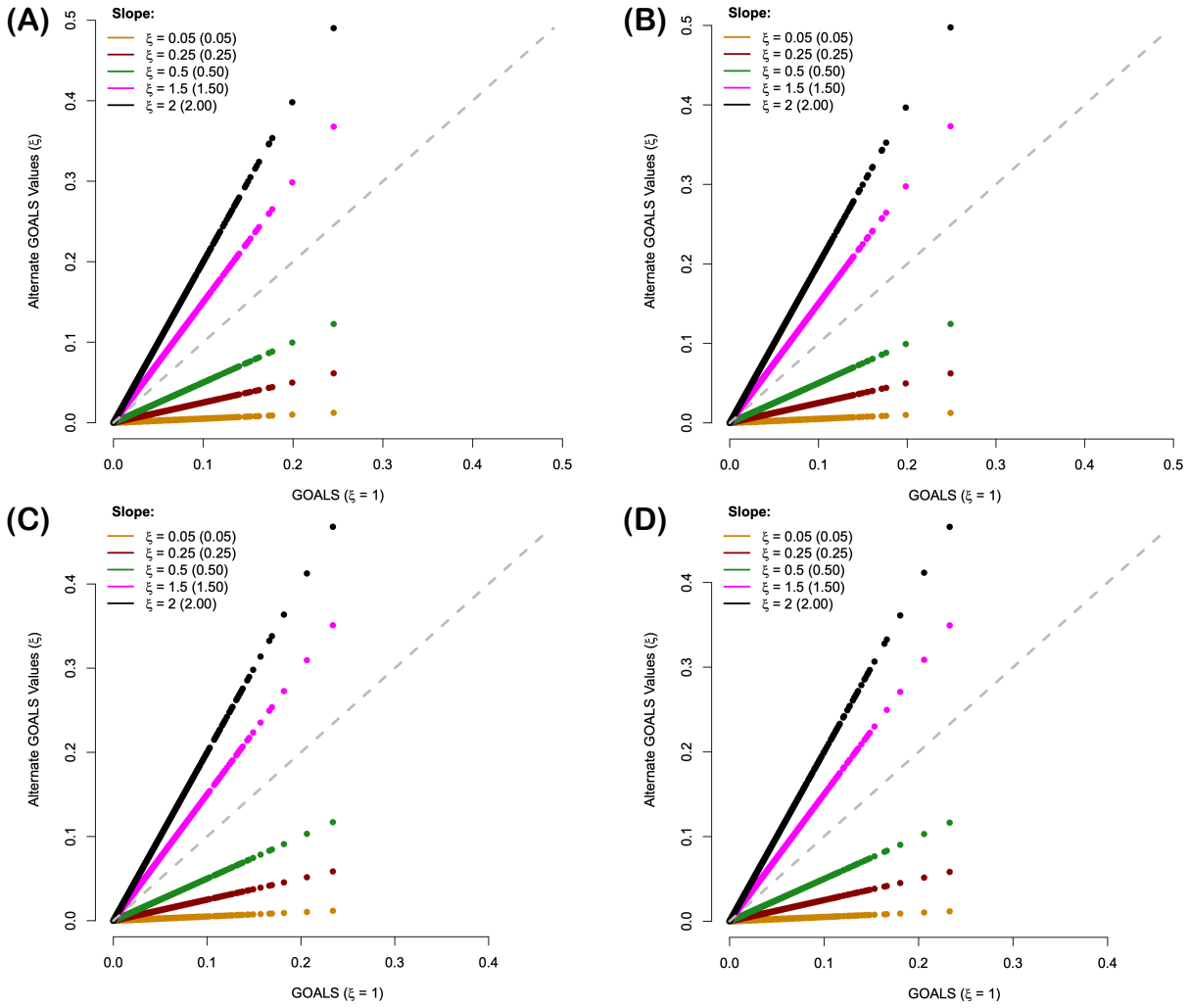

Figure S1. Scatterplots assessing how the value of the parameter ξ affects the magnitude of the GOALS operator in simulations. Here, synthetic responses are simulated to have a signal-to-noise ratio equal to $v^2 = 0.6$ with only additive effects in panels **(A)** and **(B)**, and a combination of additive and pairwise interaction effects in panels **(C)** and **(D)**. This is controlled by a free parameter $\rho = \{0.5, 1\}$ which was used to determine the proportion of signal that is contributed by additivity. The response variables simulated in panels **(B)** and **(D)** also have the additional complexity of having population stratification effects. Values of GOALS computed with $\xi = 1$ are given on the x-axis, and values of GOALS computed with $\xi = \{0.05, 0.25, 0.5, 1.5, 2\}$ are given on the y-axis. Slopes of each line are used to compare the differences between estimates, and the dotted grey line represents the values at which estimates from both approaches are the same. The main takeaway is that while the magnitude of GOALS is affected by ξ, its relative ranking of features remains the same (hence, the robust performance in terms of true and false positive rates in Figure 3). All results are based on 100 simulated replicates.

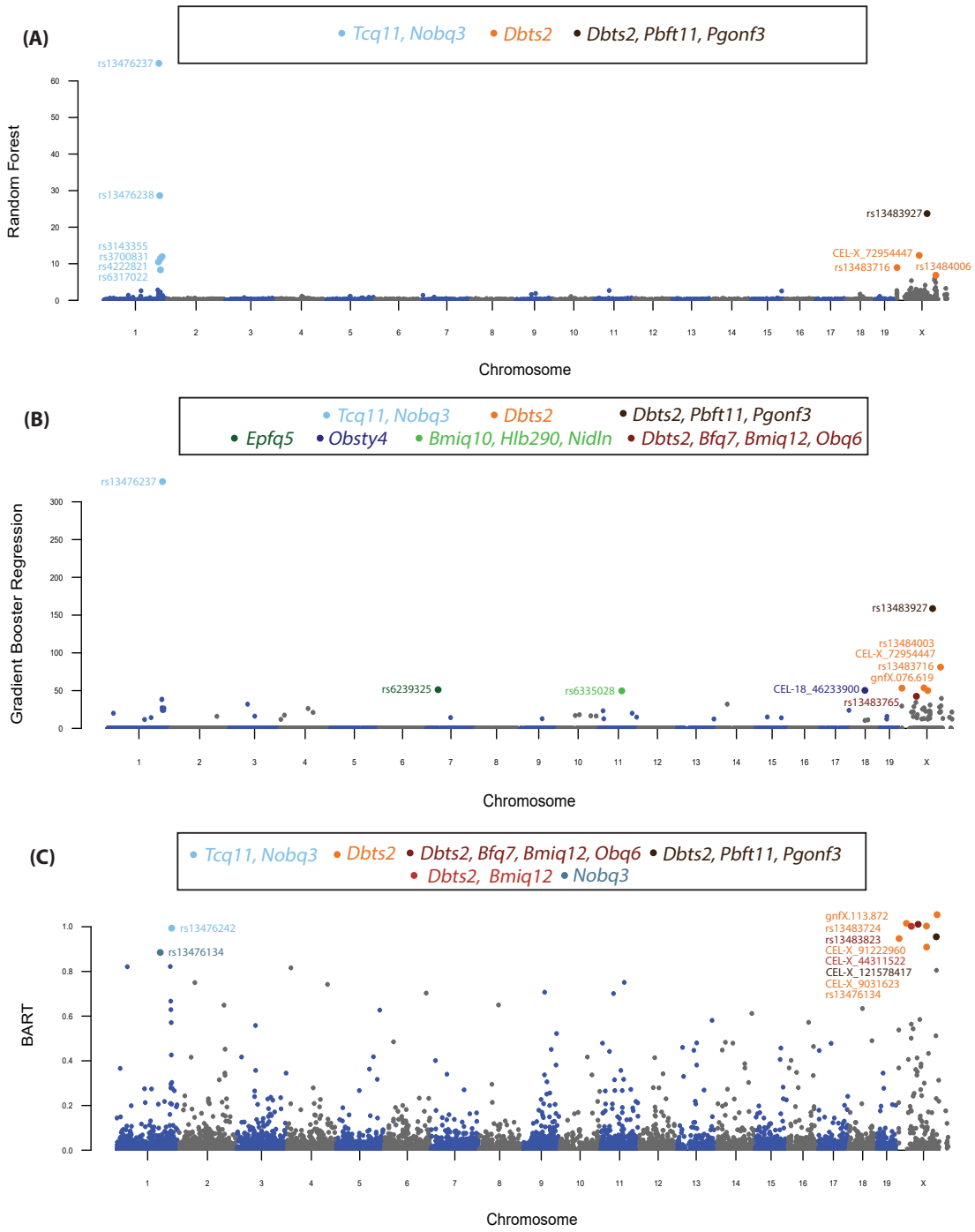

Figure S2. Manhattan plot of variant-level association mapping results for high-density lipoprotein (HDL) content in the heterogeneous stock of mice data set from the Wellcome Trust Centre of Human Genetics (Valdar et al., 2006a,b) using competing global variable importance approaches. Panel **(A)** depicts the global importance for each SNP plotted against their genomic positions after running a random forest (RF) with 500 trees (Ishwaran and Lu, 2019). Here, genetic features are ranked by assessing their relative influence which is computed by taking the average total decrease in the residual sum of squares after splitting on each variable. As a direct comparison, we also include results after implementing **(B)** a gradient boosting machine (GBM) (Friedman, 2001) with 100 trees and **(C)** a Bayesian additive regression tree (BART) (Chipman et al., 2010) with 200 trees and 1000 MCMC iterations on the same quantitative trait. In the GBM, global importance is also determined by computing the relative influence of each SNP; while, in BART, features are ranked by the average number of times that they are used in decisions for each tree. In this figure, chromosomes are shown in alternating colors for clarity. The top 10 highest ranked SNPs by each method are labeled and color coded based on their nearest mapped gene(s) as cited by the Mouse Genome Informatics database (`http://www.informatics.jax.org/`) (Bult et al., 2019). These annotated genes are listed in the legends of each panel. A complete list of the values for all SNPs can be found in Table S2.

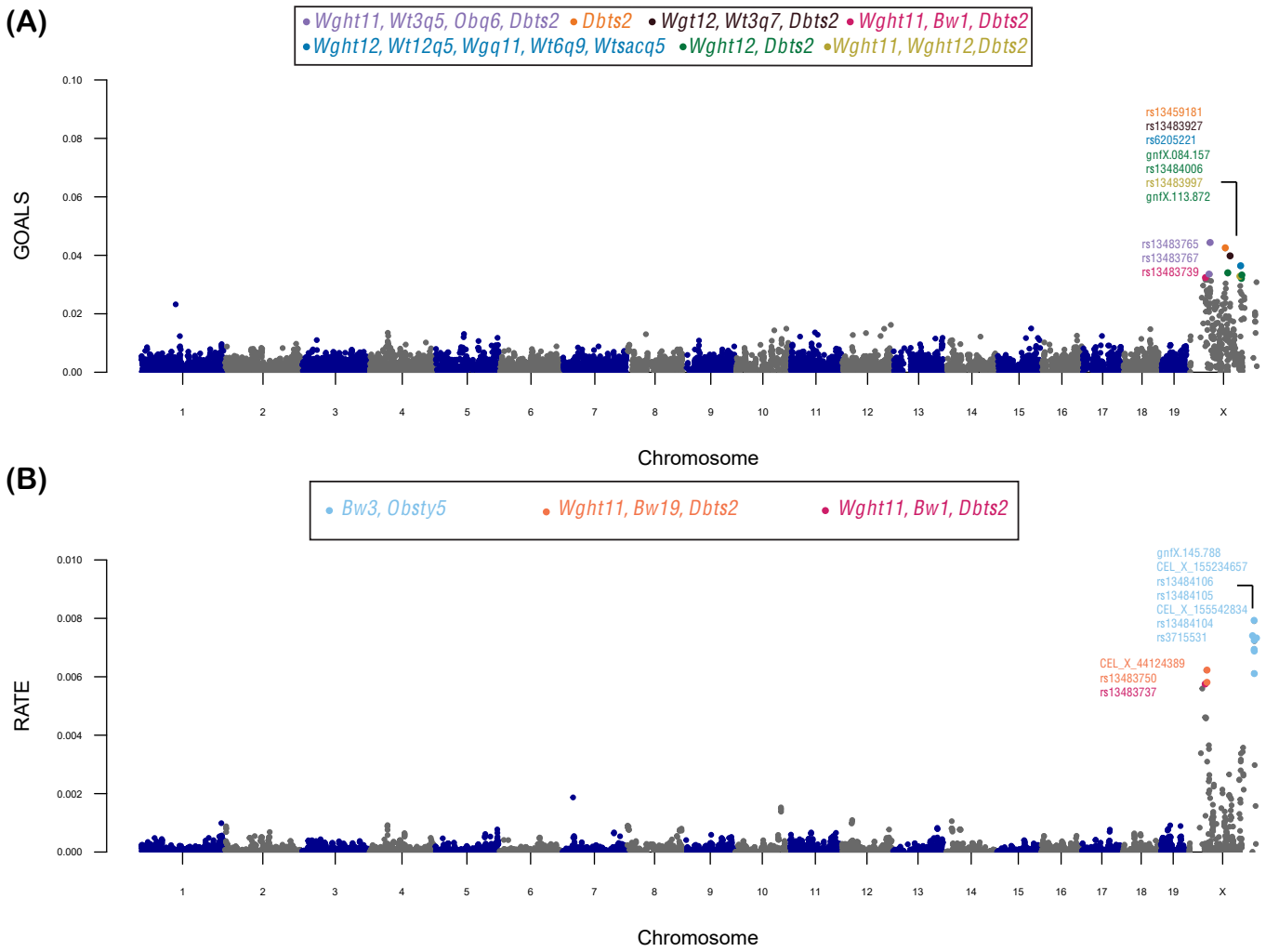

Figure S3. Manhattan plot of variant-level association mapping results for body weight in the heterogeneous stock of mice data set from the Wellcome Trust Centre of Human Genetics (Valdar et al., 2006a,b). Panel **(A)** depicts the global GOALS measure (with $\xi = 1$) of quality-control-positive SNPs plotted against their genomic positions after running a Bayesian Gaussian process (GP) regression on the quantitative trait. As a direct comparison, in panel **(B)**, we also include results after implementing RATE on the same fitted GP model. In this figure, chromosomes are shown in alternating colors for clarity. The top 10 highest ranked SNPs by GOALS and RATE, respectively, are labeled and color coded based on their nearest mapped gene(s) as cited by the Mouse Genome Informatics database (`http://www.informatics.jax.org/`) (Bult et al., 2019). These annotated genes are listed in the legends of each panel. A complete list of the GOALS and RATE values for all SNPs can be found in Table S2.

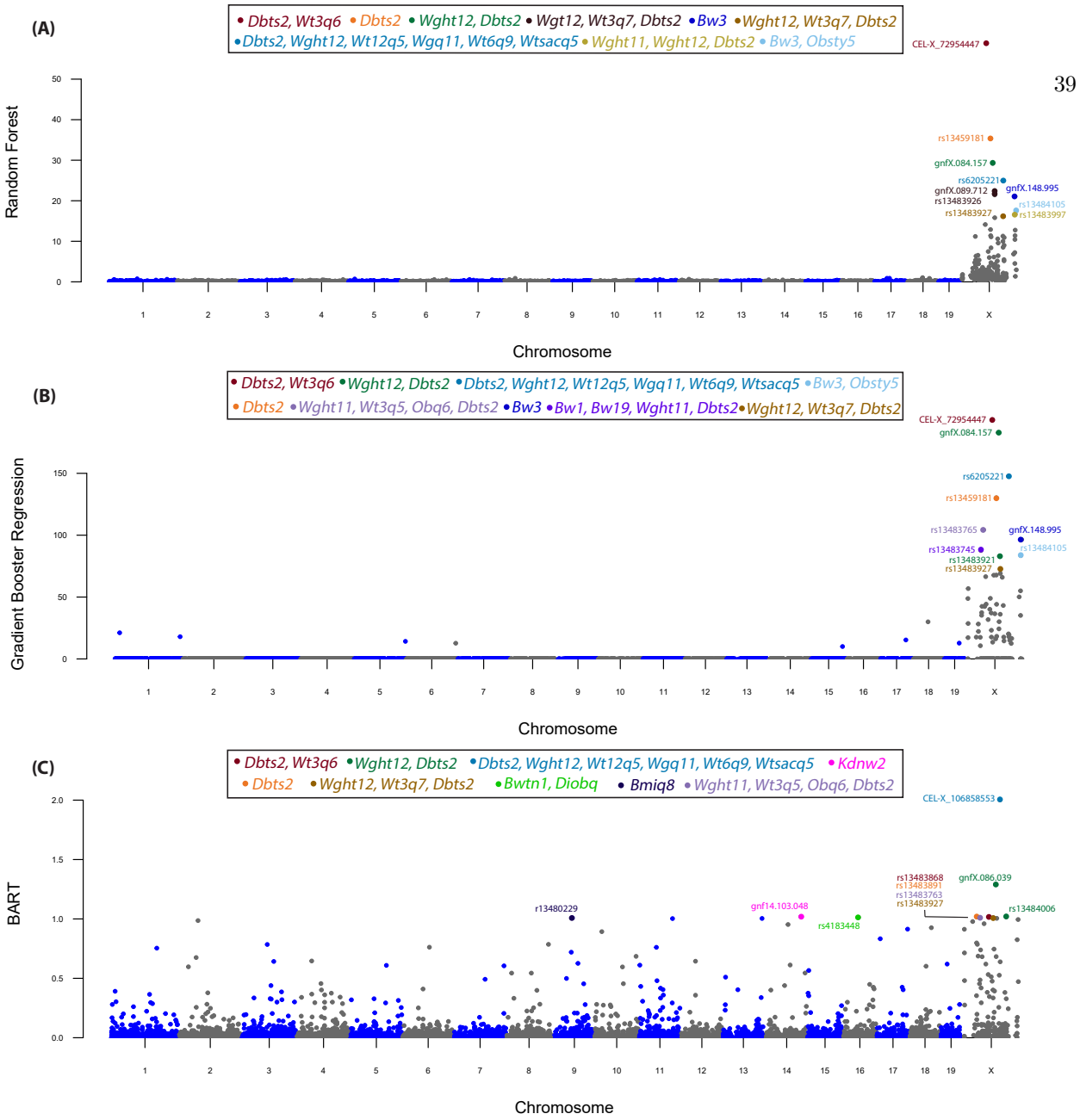

Figure S4. Manhattan plot of variant-level association mapping results for body weight in the heterogeneous stock of mice data set from the Wellcome Trust Centre of Human Genetics (Valdar et al., 2006a,b) using competing global variable importance approaches. Panel **(A)** depicts the global importance for each SNP plotted against their genomic positions after running a random forest (RF) with 500 trees (Ishwaran and Lu, 2019). Here, genetic features are ranked by assessing their relative influence which is computed by taking the average total decrease in the residual sum of squares after splitting on each variable. As a direct comparison, we also include results after implementing **(B)** a gradient boosting machine (GBM) (Friedman, 2001) with 100 trees and **(C)** a Bayesian additive regression tree (BART) (Chipman et al., 2010) with 200 trees and 1000 MCMC iterations on the same quantitative trait. In the GBM, global importance is also determined by computing the relative influence of each SNP; while, in BART, features are ranked by the average number of times that they are used in decisions for each tree. In this figure, chromosomes are shown in alternating colors for clarity. The top 10 highest ranked SNPs by each method are labeled and color coded based on their nearest mapped gene(s) as cited by the Mouse Genome Informatics database (`http://www.informatics.jax.org/`) (Bult et al., 2019). These annotated genes are listed in the legends of each panel. A complete list of the values for all SNPs can be found in Table S2.

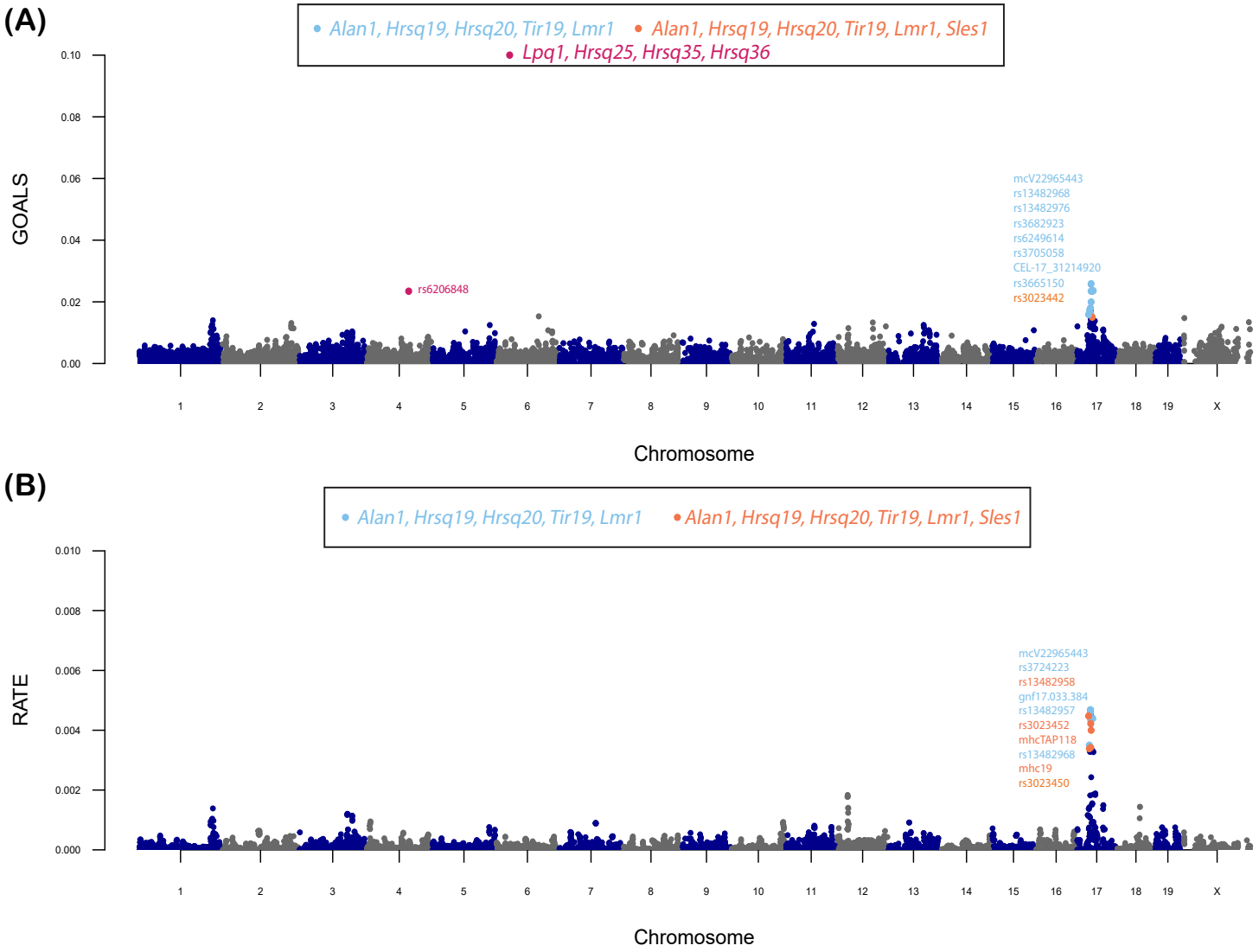

Figure S5. Manhattan plot of variant-level association mapping results for the percentage of CD8+ cells in the heterogeneous stock of mice data set from the Wellcome Trust Centre of Human Genetics (Valdar et al., 2006a,b). Panel **(A)** depicts the global GOALS measure (with $\xi = 1$) of quality-control-positive SNPs plotted against their genomic positions after running a Bayesian Gaussian process (GP) regression on the quantitative trait. As a direct comparison, in panel **(B)**, we also include results after implementing RATE on the same fitted GP model. In this figure, chromosomes are shown in alternating colors for clarity. The top 10 highest ranked SNPs by GOALS and RATE, respectively, are labeled and color coded based on their nearest mapped gene(s) as cited by the Mouse Genome Informatics database (`http://www.informatics.jax.org/`) (Bult et al., 2019). These annotated genes are listed in the legends of each panel. A complete list of the GOALS and RATE values for all SNPs can be found in Table S3.

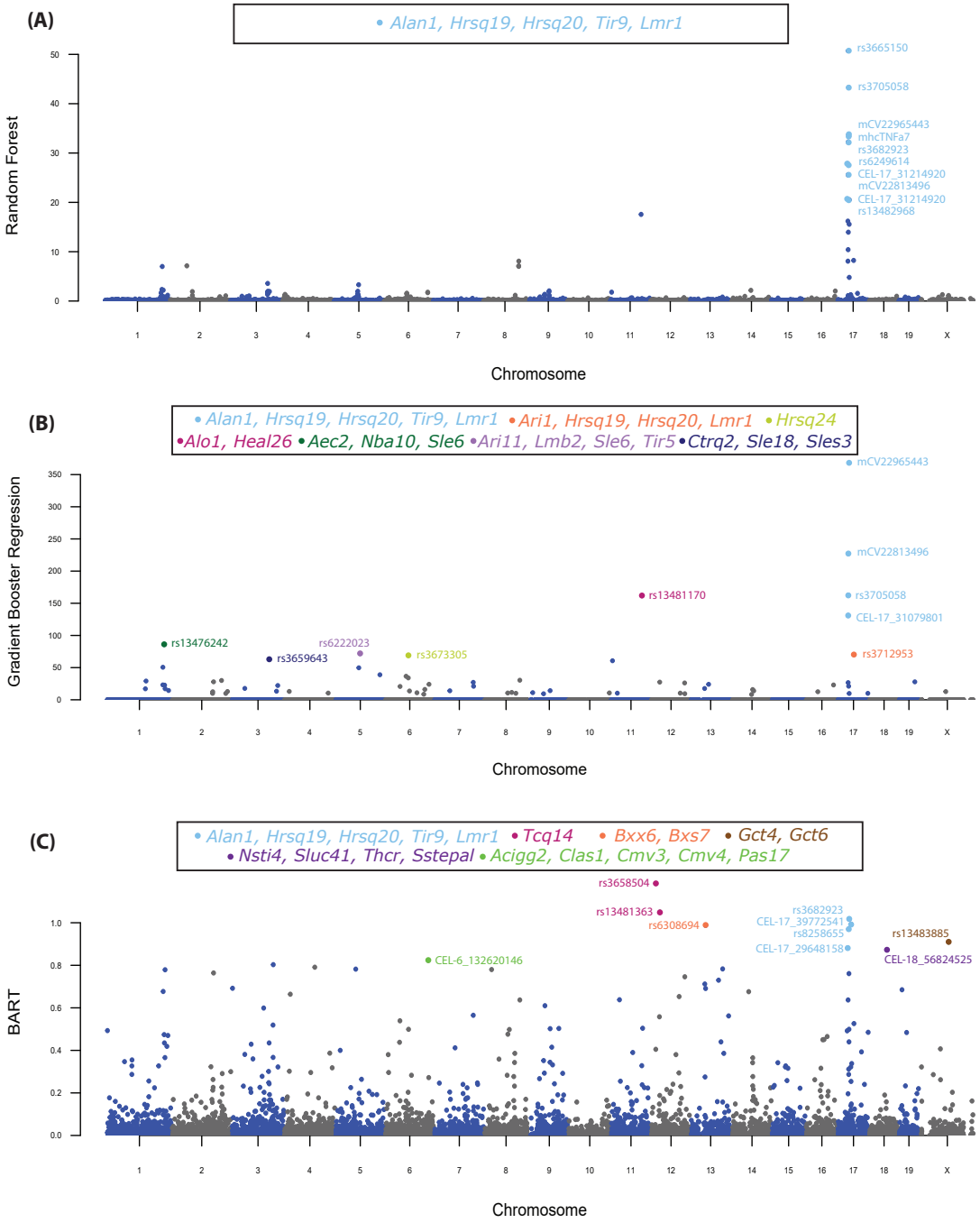

Figure S6. Manhattan plot of variant-level association mapping results for the percentage of CD8+ cells in the heterogeneous stock of mice data set from the Wellcome Trust Centre of Human Genetics (Valdar et al., 2006a,b) using competing global variable importance approaches. Panel **(A)** depicts the global importance for each SNP plotted against their genomic positions after running a random forest (RF) with 500 trees (Ishwaran and Lu, 2019). Here, genetic features are ranked by assessing their relative influence which is computed by taking the average total decrease in the residual sum of squares after splitting on each variable. As a direct comparison, we also include results after implementing **(B)** a gradient boosting machine (GBM) (Friedman, 2001) with 100 trees and **(C)** a Bayesian additive regression tree (BART) (Chipman et al., 2010) with 200 trees and 1000 MCMC iterations on the same quantitative trait. In the GBM, global importance is also determined by computing the relative influence of each SNP; while, in BART, features are ranked by the average number of times that they are used in decisions for each tree. In this figure, chromosomes are shown in alternating colors for clarity. The top 10 highest ranked SNPs by each method are labeled and color coded based on their nearest mapped gene(s) as cited by the Mouse Genome Informatics database (`http://www.informatics.jax.org/`) (Bult et al., 2019). These annotated genes are listed in the legends of each panel. A complete list of the values for all SNPs can be found in Table S2.