CLAIMS

What is claimed is:

1. A model integration method, the method comprising:

determining obtaining an initial integrated model, the integrated model having 1 one integrated output value and taking the a plurality of input values, the plurality of input values corresponding to a plurality of output values of at least twoa plurality of independent models as input values;

performing iteration processing with following steps one or more iterations of an optimizing process until a preset iteration stop condition is satisfied, the iteration stop condition at least comprising that: an index of a current integrated model meets a preset performance requirement;wherein the optimizing process comprises:

acquiring a prediction result output by the current integrated model for based on a preset test event set;

, and counting the prediction results to acquire at least onedetermining an index value of the integrated model based on the prediction output, the index value being configured to evaluateindicating the a performance evaluation of the integrated model;

if the index value fails to meet a preset performance requirement, performing optimizing ation processing on the integrated model according tobased on a preset optimization training event set different from the preset test event set by performing at least the following adjustments, the optimization processing comprising:

refitting an the integrated model ion method of by adjusting a plurality of integration weights in the integrated model respectively corresponding to a the plurality of independent models,;

or separately refitting one or more of the plurality of independent models by performing operations, comprising:

adjusting one or more inputs associated with each of the one or more independent models;

sorting the plurality of independent models based on the plurality of corresponding integration weights;

selecting for refitting a first independent model of the sorted plurality of independent models that has not been refitted and has a highest integration weight;

after selecting the first independent model for refitting, refitting the first

independent model; and

after refitting the first independent model, retraining the first independent model; and

after the preset iteration stop condition is ~~stopped, if the index value of the current integrated model meets the performance requirement~~satisfied, determining the ~~current~~ integrated model as ~~an available integrated model~~acceptable.

2. The method according to claim 1, wherein the preset iteration stop condition comprises a requirement that the index value satisfies the preset performance requirement, or a number of the one or more iterations reaches a preset threshold.

~~2~~3. The method according to claim 1, wherein ~~the step of performing~~ optimizing ~~ation processing on~~ the integrated model comprises:

refitting the integrated model in ~~the~~ a first iteration ~~process, refitting the~~by adjusting the plurality of integration weights respectively corresponding to ~~method of~~ the plurality of independent models ~~first~~, and

if the preset performance requirement is ~~may~~ not ~~be met~~satisfied~~,~~ :

separately refitting one or more of the plurality of independent models in ~~the~~ one or more subsequent ~~rounds of~~ iterations after the first iteration by adjusting the one or more inputs associated with each of the one or more independent models; and

retraining one or more of the plurality of independent models in one or more subsequent iterations after the first iteration to acquire one or more new independent models after removing input features from or adding input features to each of the one or more independent models.

~~3~~4. The method according to claim 1, wherein optimizing the integrated model further comprises:~~wherein the step of refitting the integration method of the plurality of independent models comprises~~:

~~refitting integration weights of the independent models in the integrated model without changing the independent models participating in the integration;~~

~~or~~

~~increasing or decreasing a quantity of the independent models configured to provide the input values of the integrated model, and then~~ retraining the integrated ~~independent~~ model~~s~~ to acquire a new integrated model after removing one or more independent models from the plurality of independent models to provide inputs to the integrated model, or after adding one or more new independent models to the plurality of independent models to provide inputs to the integrated model.

4. The method according to claim 1, wherein the step of separately refitting the independent models comprises:

in the iteration process, based on the integration weight of each independent model in the integrated model, refitting the independent models with relatively high integration weights first; if the performance requirement is not met, refitting the independent models with relatively low integration weights in subsequent rounds of iterations.

5. The method according to claim 1, wherein the step of separately refitting the independent models comprises:

refitting parameters of the independent models without changing input features of the independent models;

or

increasing or decreasing a quantity of the input features of the independent models, and then retraining the independent models to acquire new independent models.

6. The method according to claim 1, wherein the test event set for counting the index values and the optimization training event set for optimizing the integrated model are a test set and a verification set from a same event set.

7. The method according to claim 1, wherein the step of determining the initial integrated model comprises:

acquiring an event data set from a historical event record, each piece of event data comprising: a predicted output value of each independent model for an event and an integrated label value of the event;

by taking the acquired event data set as a sample, training the sample with a supervised learning algorithm to acquire the integrated model.

8. A model integration method, the method comprising:

determining an initial integrated model, the integrated model having 1 integrated output value and taking the output values of at least two independent models as input values;

acquiring a prediction result output by a current integrated model for a preset test event set, and counting the prediction results to acquire at least one index value, the index value being configured to evaluate the performance of the integrated model;

if the index value does not meet a performance requirement, performing optimization processing on the integrated model according to a preset optimization training event set, the optimization processing comprising: refitting an integration method of a plurality of independent models, or separately refitting the independent models;

after the optimization processing, if the index value of the current integrated model

meets the performance requirement, determining the current integrated model as an available integrated model.

9. A model integration device, the device comprising:

an initial integrated model determination module, configured to determine an initial integrated model, the integrated model having 1 integrated output value and taking the output values of at least two independent models as input values;

an optimization processing module, configured to perform iteration processing with following submodules until a preset iteration stop condition is satisfied, the iteration stop condition at least comprising that: an index of the current integrated model meets a preset performance requirement;

a performance evaluation submodule, configured to acquire a prediction result output by the current integrated model for a preset test event set, and count the prediction results to acquire at least one index value of the integrated model, the index value being configured to evaluate the performance of the integrated model;

an optimization processing submodule, configured to perform optimization processing on the integrated model according to a preset optimization training event set, the optimization processing comprising: refitting an integration method of a plurality of independent models, or separately refitting the independent models;

an output module, configured to determine the current integrated model as an available integrated model after the iteration is stopped and if the index value of the current integrated model meets the performance requirements.

10. The device according to claim 9, wherein the optimization processing submodule is specifically configured to:

in the iteration process, refit the integration method of the plurality of independent models first, and if the performance requirement may not be met, separately refit the independent models in the subsequent rounds of iterations.

11. The device according to claim 9, wherein the optimization processing submodule is specifically configured to refit the integration method of the plurality of independent models with following method:

refitting integration weights of the independent models in the integrated model without changing the independent models participating in the integration;

or

increasing or decreasing a quantity of the independent models configured to provide the input values of the integrated model, and then retraining the independent models to acquire a

new integrated model.

12. The device according to claim 9, wherein the optimization processing submodule is specifically configured to separately refit the independent models by using the following method:

in the iteration process, based on the integration weight of each independent model in the integrated model, refitting the independent models with relatively high integration weights first; if the performance requirement is not met, refitting the independent models with relatively low integration weights in subsequent rounds of iterations.

13. The device according to claim 9, wherein the optimization processing submodule is specifically configured to separately refit the independent models by using the following method:

refitting parameters of the independent models without changing input features of the independent models;

or

increasing or decreasing a quantity of the input features of the independent models, and then retraining the independent models to acquire new independent models.

14. The device according to claim 9, wherein the test event set for counting the index values and the optimization training event set for optimizing the integrated model are a test set and a verification set from a same event set.

15. The device according to claim 9, wherein the initial integrated model determination module is specifically configured to:

acquiring an event data set from a historical event record, each piece of event data comprising: a predicted output value of each independent model for an event and an integrated label value of the event;

by taking the acquired event data set as a sample, training the sample with a supervised learning algorithm to acquire the integrated model.

16. A model integration device, the device comprising:

an initial integrated model determination module, configured to determine an initial integrated model, the integrated model having 1 integrated output value and taking the output values of at least two independent models as input values;

an optimization processing module, configured to acquire a prediction result output by a current integrated model for a preset test event set, count the prediction results to acquire at least one index value, the index value being configured to evaluate the performance of the integrated model, and perform optimization processing on the integrated model according to

~~a preset optimization training event set if the index value does not meet a performance requirement, the optimization processing comprising: refitting an integration method of a plurality of independent models, or separately refitting the independent models;~~

~~an output module, configured to determine the current integrated model as an available integrated model after the optimization and if the index value of the current integrated model meets the performance requirement.~~

~~17. A computer apparatus, comprising a memory, a processor and computer program stored on the memory and capable of being operated on the processor, wherein when the processor executes the program, the processor implements the method according to any one of claims 1 to 8.~~

5. The method according to claim 1, wherein retraining the first independent model comprises:

removing input features from the first independent model or adding input features to the first independent model.

6. The method according to claim 1, wherein obtaining the integrated model comprises:

obtaining a set of historical event data records, each comprising a plurality of predicted values generated by the plurality of independent models and an integrated label value, wherein each of the plurality of predicted values is within a predicted value range associated with the corresponding independent model;

dividing the predicted value range associated with the corresponding independent model into a plurality of subintervals, wherein each of the plurality of predicted values falls into one of the plurality of subintervals;

converting, for each of the set of historical event data records, the plurality of predicted values respectively to a plurality of encoded values;

obtaining a training data set based on the set of historical event data records, wherein each training data of the training data set is associated with a plurality of eigenvalues determined based on the plurality of corresponding encoded values; and

training the integrated model by a supervised learning algorithm based on the training data set.

7. The method according to claim 6, wherein converting, for each of the set of historical event data records, the plurality of predicted values respectively to a plurality of encoded values comprises:

converting the plurality of predicted values respectively to a plurality of One-Hot

encoded values, wherein each of the plurality of One-Hot encoded values comprises a plurality of bits, and a quantity of the plurality of bits equals to a quantity of the plurality of subintervals.

8. The method according to claim 6, wherein converting, for each of the set of historical event data records, the plurality of predicted values respectively to a plurality of encoded values comprises:

for each independent model of the plurality of independent models, determining a plurality of weight of evidence (WOE) scores respectively for the plurality of subintervals; and

converting each predicted value from the independent model to a WOE score corresponding to a subinterval of the plurality of subintervals in which the each predicted value belongs.

9. The method of claim 1, wherein the integrated model is a risk control model, and each of the plurality of independent models is a risk control model corresponding to a risk type.

10. A model integration system, comprising one or more processors and one or more non-transitory computer-readable memories coupled to the one or more processors and configured with instructions executable by the one or more processors to cause the system to perform operations comprising:

obtaining an integrated model, the integrated model having one integrated output value and a plurality of input values, the plurality of input values corresponding to a plurality of output values of a plurality of independent models;

performing one or more iterations of an optimizing process until a preset iteration stop condition is satisfied, wherein the optimizing process comprises:

acquiring a prediction output by the integrated model based on a preset test event set;

determining an index value of the integrated model based on the prediction output, the index value indicating a performance evaluation of the integrated model;

if the index value fails to meet a preset performance requirement, optimizing the integrated model based on a preset optimization event set different from the preset test event set by performing at least the following adjustments:

refitting the integrated model by adjusting a plurality of integration weights in the integrated model respectively corresponding to the plurality of independent models;

separately refitting one or more of the plurality of independent models, wherein the separately refitting comprises:

adjusting one or more inputs associated with each of the one or more independent models;

sorting the plurality of independent models based on the plurality of corresponding integration weights;

selecting for refitting a first independent model of the sorted plurality of independent models that has not been refitted and has a highest integration weight; and

after selecting the first independent model for refitting, refitting the first independent model; and

after refitting the first independent model, retraining the first independent model;

after the preset iteration stop condition is satisfied, determining the integrated model as acceptable.

11. The system according to claim 10, wherein the preset iteration stop condition comprises a requirement that the index value satisfies the preset performance requirement, or a number of the one or more iterations reaches a preset threshold.

12. The system according to claim 10, wherein optimizing the integrated model comprises:

refitting the integrated model in a first iteration by adjusting the plurality of integration weights respectively corresponding to the plurality of independent models; and

if the preset performance requirement is not satisfied:

separately refitting one or more of the plurality of independent models in one or more subsequent iterations after the first iteration by adjusting the one or more inputs associated with each of the one or more independent models; and

retraining one or more of the plurality of independent models in one or more subsequent iterations after the first iteration to acquire one or more new independent models after removing input features from or adding input features to each of the one or more independent models.

13. The system according to claim 10, wherein obtaining the integrated model comprises:

obtaining a set of historical event data records, each comprising a plurality of predicted values generated by the plurality of independent models and an integrated label value, wherein each of the plurality of predicted values is within a predicted value range associated with the corresponding independent model;

dividing the predicted value range associated with the corresponding independent model into a plurality of subintervals, wherein each of the plurality of predicted values falls into one

of the plurality of subintervals;

converting, for each of the set of historical event data records, the plurality of predicted values respectively to a plurality of encoded values;

obtaining a training data set based on the set of historical event data records, wherein each training data of the training data set is associated with a plurality of eigenvalues determined based on the plurality of corresponding encoded values; and

training the integrated model by a supervised learning algorithm based on the training data set.

14. The system of claim 10, wherein the integrated model is a risk control model, and each of the plurality of independent models is a risk control model corresponding to a risk type.

15. A non-transitory computer-readable storage medium for model integration configured with instructions executable by one or more processors to cause the one or more processors to perform operations comprising:

obtaining an integrated model, the integrated model having one integrated output value and a plurality of input values, the plurality of input values corresponding to a plurality of output values of a plurality of independent models;

performing one or more iterations of an optimizing process until a preset iteration stop condition is satisfied, wherein the optimizing process comprises:

acquiring a prediction output by the integrated model based on a preset test event set;

determining an index value of the integrated model based on the prediction output, the index value indicating a performance evaluation of the integrated model;

if the index value fails to meet a preset performance requirement, optimizing the integrated model based on a preset optimization event set different from the preset test event set by performing at least the following adjustments:

refitting the integrated model by adjusting a plurality of integration weights in the integrated model respectively corresponding to the plurality of independent models;

separately refitting one or more of the plurality of independent models, wherein the separately refitting comprises:

adjusting one or more inputs associated with each of the one or more independent models;

sorting the plurality of independent models based on the plurality of corresponding integration weights;

selecting for refitting a first independent model of the sorted plurality of

independent models that has not been refitted and has a highest integration weight; and

after selecting the first independent model for refitting, refitting the first independent model; and

after refitting the first independent model, retraining the first independent model; and

after the preset iteration stop condition is satisfied, determining the integrated model as acceptable.

16. The non-transitory computer-readable storage medium according to claim 15, wherein the preset iteration stop condition comprises a requirement that the index value satisfies the preset performance requirement, or a number of the one or more iterations reaches a preset threshold.

17. The non-transitory computer-readable storage medium according to claim 15, wherein optimizing the integrated model comprises:

refitting the integrated model in a first iteration by adjusting the plurality of integration weights respectively corresponding to the plurality of independent models; and

if the preset performance requirement is not satisfied:

separately refitting one or more of the plurality of independent models in one or more subsequent iterations after the first iteration by adjusting the one or more inputs associated with each of the one or more independent models; and

retraining one or more of the plurality of independent models in one or more subsequent iterations after the first iteration to acquire one or more new independent models after removing input features from or adding input features to each of the one or more independent models.

18. The non-transitory computer-readable storage medium according to claim 15, wherein retraining the first independent model comprises:

removing input features from the first independent model or adding input features to the first independent model.

19. The non-transitory computer-readable storage medium according to claim 15, wherein obtaining the integrated model comprises:

obtaining a set of historical event data records, each comprising a plurality of predicted values generated by the plurality of independent models and an integrated label value, wherein each of the plurality of predicted values is within a predicted value range associated with the corresponding independent model;

dividing the predicted value range associated with the corresponding independent model into a plurality of subintervals, wherein each of the plurality of predicted values falls into one

of the plurality of subintervals;

converting, for each of the set of historical event data records, the plurality of predicted values respectively to a plurality of encoded values;

obtaining a training data set based on the set of historical event data records, wherein each training data of the training data set is associated with a plurality of eigenvalues determined based on the plurality of corresponding encoded values; and

training the integrated model by a supervised learning algorithm based on the training data set.

20. The non-transitory computer-readable storage medium of claim 15, wherein the integrated model is a risk control model, and each of the plurality of independent models is a risk control model corresponding to a risk type.