## Objectives

Ensemble models are a very popular technique as they can assist your models be more resistant to outliers and have better chances at generalizing with future data. They also gained popularity after several ensembles helped people win prediction competitions. Recently, stochastic gradient boosting became a go-to candidate model for many data scientists.This model walks you through the theory behind ensemble models and popular tree-based ensembles.

* Identify, use, and interpret common ensemble models for classification, including bagging, boosting, stacking, and random forest.
* Build ensemble models with sklearn, including bagging, boosting, stacking, and random forest.
* Identify common supervised machine learning algorithms.

## Ensemble Based Methods and Bagging

Improvements to decision trees: use many trees

Aggregate results: trees vote on or average results for each data point

Bagging = bootstrap aggregating

How many trees to fit

* Bagging performance improvements increase with more trees
* Maximum improvement generally reached ~50 trees

Bagging error calculations

Same as decision trees:

* Easy to interpret and implement
* Heterogeneous input data allowed, no preprocessing required

Specfic to bagging:

* Less variabiity than decision trees
* Can grow trees in parallel

From sklearn.ensemble import BaggingClassifier

BC = BaggingClassifier(n\_estimators=50)

Random Forest

Reduction in variance due to bagging

* For n independent trees, each with variance o^2, the bagging variance is : o^2/n
* However bootstrap samples are correlated (p)

Introducting more randomness

* Solution:
  + Further de-correlate trees
* Use random subset of features
  + Classification
  + Regression
* Called “random forest”

How many random forest trees

* Errors are further reduced for random forest relative to bagging
* Grow enough trees until error settles down
* Additional trees won’t improve results

From sklearn.ensemble import RandomForestClassifier

Introducing even more randomness

* Sometimes additional randomness is desired beyon random forests
* Solution: select features randomly and create splits randomly don’t choose greedily
* Called “Extra Random trees”

From sklearn.ensemble import ExtraTreesClassifier

Baggin notebook pt. 1

Correlation between different features

Examing target and preprocessing

Value\_counts(normalize=True) ????

[X for x in data.columns if x != target] ????

Bagging notebook Pt. 2

Oob\_score = true

N\_jobs = -1

Iterate though up to 400 trees

Baggin notebook pt. 3

Roc-auc

Review of Bagging

Decision stump: boosting base learner

Fit

Adjust weights of points

Find new decision stump to current residuals

Find new decision stump to fit weighted residuals

Combine to form a single classifier

Result is weighted sum of all classifiers

Successive classifiers are weighted by learning rate

Learning rate <1.0 prevent overfitting

Boosting specifics

* Boosting utilizes diff. Loss functions
* At each stage, the margin is determined for each point.
* Margin is positive for correctly classified points and negative for misclassifications
* Value of loss function is calculated from margin

0-1 loss function

- the 0-1 loss multiplies misclassified points by 1

- correctly classified points are ignored.

- Theoretical “ideal” loss function

- difficult to optimize

- non-smooth and non-convex

AdaBoost loss function

* Adaboost = adaptive boosting
* Loss function is expoential
* Makes adaboost more sensitive to outliers than other types of boosting

Gradient boosting loss function

* Generalized boosting method that can use different loss functions
* Common implementation uses binomial log likelihood loss function (deviance)
* More robust to outliers than adaboost

Bagging:

* Bootstrapped samples
* Base trees created independetly
* Only data points considered
* No weights used
* Excess trees will not overfit

Boosting

* Fit entire data set
* Base trees created successively
* Use residuals from previous models
* Up-weight misclassified points
* Beware of overfitting

Boosting is additive so possible to overfit

Use cross validation

Features to use:

* Learning rate
* Subsample
* Max\_features

Gradientboostingclassifier: syntax

From sklearn.ensemble import GradientBoostingClassifier

From sklearn.tree import DecisionTreeClassifier

Stacking: combining classifiers

* Models of any kind can be combined to create a stacked model.
* Like bagging but not limited to decision trees
* Output of base learners creates features, can recombine with data
* Output of base learners can be combined via majority vote or weighted
* Additional hold-out data needed if meta learner paramters are used.
* Be aware of increasing model complexity
* The final prediction can be done by vvoting or with another model

Base learners and meta learners

From sklearn.ensemble import VotingClassifier

Boosting Notebook Pt. 1

Check max value for every column

.all() iare all values true

Split data into train and test set

Boosting can take a long time to run

Feature columns

X var feature\_columns

Boosting noteook pt. 2

Question 3.

Gradient boosting models

Deminision error

Question 4 grid search with gradient boosted classifier

Add regularization to model

Pickle file

Subsample a term of regulatization

Check confusion method to see what went wrong

Now do the same thing for adaboost

Part 3.

Questino 6 – begin solution block of code - took 45 sec to a min to run?

Combining two classificer and use voting classifier

Voting=’soft’ means we are get probabilities

## End of module review: Ensemble Models

### **Ensemble Based Methods and Bagging**

Tree ensembles have been found to generalize well when scoring new data. Some useful and popular tree ensembles are bagging, boosting, and random forests. Bagging, which combines decision trees by using bootstrap aggregated samples. An advantage specific to bagging is that this method can be multithreaded or computed in parallel. Most of these ensembles are assessed using out-of-bag error.

### 

### **Random Forest**

Random forest is a tree ensemble that has a similar approach to bagging. Their main characteristic is that they add randomness by only using a subset of features to train each split of the trees it trains. Extra Random Trees is an implementation that adds randomness by creating splits at random, instead of using a greedy search to find split variables and split points.

### 

### **Boosting**

Boosting methods are additive in the sense that they sequentially retrain decision trees using the observations with the highest residuals on the previous tree. To do so, observations with a high residual are assigned a higher weight.

### 

### **Gradient Boosting**

The main loss functions for boosting algorithms are:

* 0-1 loss function, which ignores observations that were correctly classified. The shape of this loss function makes it difficult to optimize.
* Adaptive boosting loss function, which has an exponential nature. The shape of this function is more sensitive to outliers.
* Gradient boosting loss function. The most common gradient boosting implementation uses a binomial log-likelihood loss function called deviance. It tends to be more robust to outliers than AdaBoost.

The additive nature of gradient boosting makes it prone to overfitting. This can be addressed using cross validation or fine tuning the number of boosting iterations. Other hyperparameters to fine tune are:

* learning rate (shrinkage)
* subsample
* number of features

### 

### **Stacking**

Stacking is an ensemble method that combines any type of model by combining the predicted probabilities of classes. In that sense, it is a generalized case of bagging. The two most common ways to combine the predicted probabilities in stacking are: using a majority vote or using weights for each predicted probability.

Quiz

|  |  |
| --- | --- |
|  | Congratulations, you earned a passing score. |

### Question 1

Correct

1.00 points out of 1.00

Flag question

#### Question text

The term Baggingstands for bootstrap aggregating.

Select one:

True

False

#### Feedback

Correct! You can find more information in the lesson: Ensemble Based Methods and Bagging.

The correct answer is 'True'.

### Question 2

Correct

1.00 points out of 1.00

Flag question

#### Question text

This is the best way to choose the number of trees to build on a Bagging ensemble.

Select one:

A.

Tune number of treesas a hyperparameter that needs to be optimized

B.

Choose a large number of trees, typically above 100

C.

Choose a number of trees past the point of diminishing returns

D.

Prioratize training error metrics over out of bag sample

#### Feedback

Correct! You can find more information in the lesson: Ensemble Based Methods and Bagging.

The correct answer is: Tune number of treesas a hyperparameter that needs to be optimized

### Question 3

Correct

1.00 points out of 1.00

Flag question

#### Question text

Which type of Ensemble modeling approach is NOT a special case of model averaging?

Select one:

A.

The Bagging method of Bootstrap aggregation

B.

The Pasting method of Bootstrap aggregation

C.

Boosting methods

D.

Random Forest methods

#### Feedback

Correct! You can find more information in the lesson Overview of Boosting.

The correct answer is: Boosting methods

### Question 4

Correct

1.00 points out of 1.00

Flag question

#### Question text

What is an ensemble model that needs you to look at out of bag error?

Select one:

A.

Out of Bag Regression

B.

Stacking

C.

Logistic Regression.

D.

Random Forest

#### Feedback

Correct! You can find more information in the lesson Random Forest.

The correct answer is: Random Forest

### Question 5

Correct

1.00 points out of 1.00

Flag question

#### Question text

What is the main condition to use stacking as ensemble method?

Select one:

A.

Models need to output residual values for each class

B.

Models need to be parametric

C.

Models need to output predicted probabilities

D.

Models need to be nonparametric

#### Feedback

Correct! You can find more information in the lesson Stacking.

The correct answer is: Models need to output predicted probabilities

### Question 6

Correct

1.00 points out of 1.00

Flag question

#### Question text

This tree ensemble method only uses a subset of the features for each tree:

Select one:

A.

Adaboost

B.

Bagging

C.

Stacking

D.

Random Forest

Correct! This tree ensemble only uses a subset of the features for each tree. For more information, please review the Random Forest lesson.

#### Feedback

Your answer is correct.

The correct answer is: Random Forest

### Question 7

Correct

1.00 points out of 1.00

Flag question

#### Question text

Order these tree ensembles in order of most randomness to least randomness:

Select one:

A.

Bagging, Random Forest, Random Trees

B.

Random Forest, Bagging, Random Trees

C.

Random Trees, Random Forest, Bagging

Correct! Random Trees add one more degree of randomness than Random Forests and two more than Bagging. You can find more information in the Random Forest lesson.

D.

Random Forest, Random Trees, Bagging

#### Feedback

Your answer is correct.

The correct answer is: Random Trees, Random Forest, Bagging

### Question 8

Correct

1.00 points out of 1.00

Flag question

#### Question text

This is an ensemble model that does not use bootstrapped samples to fit the base trees, takes residuals into account, and fits the base trees iteratively:

Select one:

A.

Random Forest

B.

Boosting

C.

Bagging

D.

Random Trees

#### Feedback

Correct! These are all characteristics of boosting algorithms. You can find more information in the Boosting lesson.

The correct answer is: Boosting