XGBoost

Part 3: Boosted trees

By: Noureddin Sadawi, PhD

University of London

Gradient boosting

- The gradient descent algorithm is used for optimisation.
- Each new model takes a step in the direction that minimises prediction error.
- The central idea here is to correct the errors in the predictions of the previous learner(s).
- It has three main components:
  - loss function that needs to be optimised
  - weak learner
  - a method to optimise the loss function.

Gradient boosting

- The gradient descent algorithm is used for optimisation.
- AdaBoost was extended by Friedman who called this a gradient boosting machine.
- Friedman’s idea was to cast boosting as a numerical optimisation problem.
- Here a gradient-descent-like procedure is used to minimise the loss of the model by adding weak learners.
- This class of algorithms can be thought of as a stage-wise additive model.
- This is an iterative algorithm and in each iteration a new weak learner is added (while not changing existing weak learners in the model).

Boosted trees

The loss function

- The type of problem being solved plays a role in selecting a suitable loss function.
- One essential characteristic of loss function is that it must be differentiable.
- Several loss functions exist (you can define your own).
- For example, regression may use a squared error and classification may use logarithmic loss.
- In general, any differentiable loss function can be used (this makes gradient boosting a generic framework).

Boosted trees

The weak learner

- Gradient boosting employs decision trees as its weak learner.
- Decision trees can be used for regression and classification.
- By now we know how a tree is greedily built.
- The weak learners are sometimes constrained (e.g. by capping the number of nodes, splits and other parameters).
- The constraint(s) help in making sure the learners do not become stronger (i.e. they remain weak).

Boosted trees

The additive model

- We mentioned before that a new tree is added in each iteration (while freezing trees that are already in the model) and the loss when adding trees is minimised using a gradient descent procedure.
- In order to use gradient descent to minimise the loss of adding a new tree, the tree is parameterised and its parameters are modified so it becomes possible to move in the direction that reduces the loss.
- This is known as functional gradient descent.
- In order to improve the overall final outcome of the model, the output of the added tree is combined with the output of existing trees in the mode.
- The addition of new trees stops when a stop criterion is reached (such as reaching the maximum number of trees or the loss no longer improves).

What is XGBoost?

XGBoost is a popular implementation of gradient boosting:

https://xgboost.readthedocs.io/