Ensembling of Models

* Neural networks are non-linear models which learn complex relationships via a stochastic training algorithm
* This stochastic nature makes the network sensitive to the specifics of the training data and it may find a different set of weights each time it is trained
* This results in different predictions each time and creates difficulty in obtaining a reliable and reproducible prediction from the final model
* In order to reduce this high variance, one way is to train multiple models and combine their predictions
* The idea is to combine the predictions from multiple good but different models
* A good model has skill, meaning that its predictions are better than random chance. Importantly, the models must be good in different ways; they must make different prediction errors.
* Since each model makes a different error, the correlation of error between different models is very low. Thus when these models are combined, the overall error normalize and the ensemble model is more geenralized
* How to ensemble?
  + 3 elements of the model can be varied when creating ensemble
    - Training data – vary the choice of data used to train each model
    - Ensemble models- vary the choice of the models used in the ensemble
    - Combinations – vary the choice of the ways the outcomes are combined
  + Vary training data
    - K-fold CV
    - Training dataset with replacement – called bootstrap aggregation or bagging
    - Random training subset ensemble
  + Vary models
    - Multiple training run ensemble
    - Hyperparameter tuning ensemble, eg learning rate, number of epochs
    - Snapshot ensemble
    - Horizontal voting ensemble
  + Vary combinations
    - Model averaging: Average of predictions from the ensemble members
    - Model Blending: Weighted average of predictions
    - Stacked generalization (Stacking): Using a new model to best combine the predictions from each ensemble member.
    - Boosting: A more sophisticated method for stacking models where members are added one at a time in order to correct the mistakes of the prior model. Due to its complexity, not much used with large neural networks
    - Model weight averaging: Averaging the weights of multiple neural networks with the same structure.
* References

<https://machinelearningmastery.com/ensemble-methods-for-deep-learning-neural-networks/>

Ensemble Learning techniques

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Complexity | Method | Key point | Use-case | Simplest Code |
| Basic | Max Voting | The prediction with maximum number of votes is selected | Classification where output is a class | finalpred = np.append(finalpred, mode([pred1[i], pred2[i], pred3[i]])) |
|  | Averaging | Predictions of all models are averaged | Regression  Classification where output is a probability | finalpred=(pred1+pred2+pred3)/3 |
|  | Weighted Average | Weighted average of predictions based on importance of respective models | Regression  Classification where output is a probability | finalpred=(pred1\*0.3+pred2\*0.3+pred3\*0.4) |
| Advanced | Stacking |  |  |  |
|  | Blending |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |
|  |  |  |  |  |