Ensembling

Remember Our Old Friend the Random Forest

- The power of random forest was using multiple models.
- Why should we limit ourselves to just partition trees?
- Different models give us different pictures.
  - Together, they can give us a more complete picture.

The Simple Ensemble: Voting

Model A	1	0	1	1	0	1	1	0	0	0	80% accuracy
Model B	1	1	0	1	1	1	0	1	1	0	60% accuracy
Model C	1	0	0	0	1	1	1	1	0	1	70% accuracy
Vote winner	1	0	0	1	1	1	1	1	0	0	90% accuracy
True values	1	0	0	1	0	1	1	1	0	0	

Continuous Outputs: Average

Model A	71.2	55.1	22.7	71.5	10.2
Model B	71.8	58.0	21.8	71.2	10.3
Model C	70.9	59.2	23.2	71.7	15.0
Average	71.3	57.4	22.6	71.5	11.83
True values	71.0	57.9	22.2	72.0	12.0

What If Not All Models Are Equal?

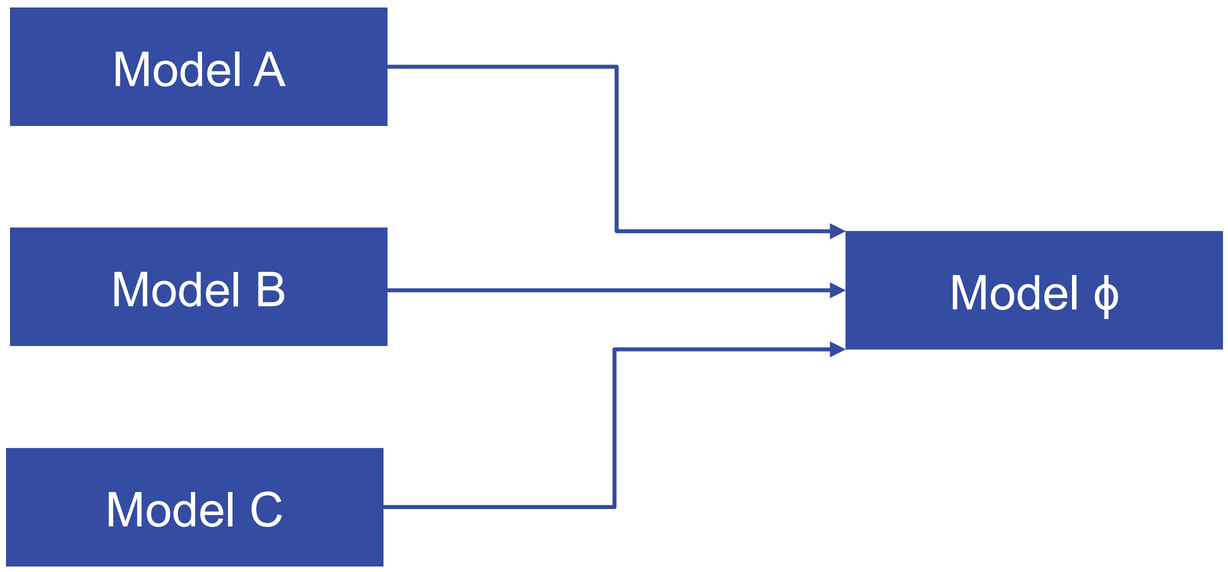
- What if one model is really good, and a few are really, really bad—couldn’t we weight their inputs?
- How would we determine their weights?
- Why don’t we **model** their weights!

The Heart of Ensembling

Use the predictions of models as **inputs** to a new set of models and let the new models determine how to combine all of the outputs.

- **Key idea:** The predictions **must be** out of fold predictions.
- “In fold” predictions, or predictions on data used to build the model, will be biased.

Ensemble Diagram



- The original training data is used to create and train Models A, B, and C.
- Then the out of fold predictions of the training data are used as “new” training data to train Model ϕ.

- In theory, a train/test split could be used and then the predictions for the test set would be used to build Model ϕ. However, that means different sizes (and possible representations!) of the data are used to build different models. Out-of-fold (OOF) is the standard way.

Notice Anything?

- The model types were not specified.
  - Models A, B, C, and ϕ can all be different types of models!
  - That means we can use all of our tools from the course
- The amount of ensemble models was not specified. There is nothing that says we cannot have multiple levels of ensembling!
  - In practice, 3 is usually the maximum

DataScience@SMU

Benefits of Ensembling

What Level of Improvement?

- Ensembling gives a small boost in performance.
- Ensembling is the **last** step and generally the smallest improvement.
- Feature creation always is the best improvement.
- Well-tuned models give the next best level of improvement.
- Ensembling is the smallest improvement (and generally the biggest effort).

The Effect Is Real Though

The leaderboard of top score for a neural network challenge as of November 2020. The best single model was in position 7 with an EM score of 89.551.

Leaderboard

SQuAD2.0 tests the ability of a system to not only answer reading comprehension questions, but also abstain when presented with a question that cannot be answered based on the provided paragraph.

Rank	Model	EM	F1
	Human Performance *Stanford University* (Rajpurkar & Jia et al. '18)	86.831	89.452
1 Apr 06, 2020	SA-Net on Albert (ensemble) *QIANXIN*	**90.724**	**93.011**
2 May 05, 2020	SA-Net-V2 (ensemble) *QIANXIN*	90.679	92.948
2 Apr 05, 2020	Retro-Reader (ensemble) *Shanghai Jiao Tong University* http://arxiv.org/abs/2001.09694	90.578	92.978
3 Jul 31, 2020	ATRLP+PV (ensemble) *Hithink RoyalFlush*	90.442	92.877
3 May 04, 2020	ELECTRA+ALBERT+EntitySpanFocus (ensemble) *SRCB_DML*	90.442	92.839
4 Jun 21, 2020	ELECTRA+ALBERT+EntitySpanFocus (ensemble) *SRCB_DML*	90.420	92.799
4 Sep 11, 2020	EntitySpanFocus+AT (ensemble) *RICOH_SRCB_DML*	90.454	92.748

Takeaways

- Even if a model doesn’t perform well, save the out-of-fold predictions for an ensemble.
- Ensembling is the final “squeeze” to get the last bit of information.
- Be aware of the effort-reward trade-off.
  - Ensembling is a lot of effort for a small reward
  - Sometimes that extra boost is important
  - Sometimes it is not

DataScience@SMU