Lecture 10: Ensemble methods

Meelis Kull
meelis.kull@ut.ee
Fall 2019
Minimisation of average loss
Compositional models
Feed-forward neural networks
Example network
Back-propagation algorithm
Universal approximation theorem
Key challenges in deep learning
Choosing the structure
Regularisation
Optimisation
Multi-class classification and softmax
Convolutional neural networks and summary
Lecture 10 – Ensemble methods

• Why do we need ensemble methods?
• Bagging
• Random forest
• Weighted averaging
• Boosting
  – Intuitive explanation
  – AdaBoost algorithm
  – Alternative formulations
  – Interpretations
Acknowledgements

• This lecture is partly inspired by:
  • Gavin Brown, University of Manchester
    – INIT/AERFAI Summer School on Machine Learning
      3 lectures on Multiple Classifier Systems (2013)
  • Raivo Kolde, University of Tartu
    – Machine Learning course,
      1 lecture on Basics of Ensemble Methods (2012)
  • Mari-Liis Allikivi and Ardi Tampuu, University of Tartu
    – Seminar presentation on boosting within the
      Special Course in Machine Learning: Ensemble
      methods (2017)
• Some slides have been reused as indicated on the slides
Lecture 10 – Ensemble methods

• Why do we need ensemble methods?
• Bagging
• Random forest
• Weighted averaging
• Boosting
  – Intuitive explanation
  – AdaBoost algorithm
  – Alternative formulations
  – Interpretations
What are Ensemble Methods?

• Ensemble method:
  – Get predictions from multiple models (ensemble) and aggregate the predictions
Aggregation of predictions

Input instance → Model 1 → Prediction 1

Input instance → Model 2 → Prediction 2

Input instance → Model 3 → Prediction 3

... (Model m)

Aggregater → Aggregated prediction

Prediction 1 → Prediction 2 → Prediction 3 → Prediction m → Aggregated prediction

Model 1 → Model 2 → Model 3 → Model m
What are Ensemble Methods?

• Ensemble method:
  – Get predictions from multiple models (ensemble)
    and aggregate the predictions

• Key questions in creating an ensemble method:
  – How to get multiple models?
  – How to aggregate the predictions?
Example of an ensemble method

• County fair in Cornwall, England in 1906
• Competition: Guess the weight of the cow
• 787 participants
• Correct answer:
  – 1198 lb ~ 543 kg
• Sir Francis Galton recorded the results and published in Nature
### Distribution of the estimates of the dressed weight of a particular living ox, made by 787 different persons.

<table>
<thead>
<tr>
<th>Degrees of the length of Array $q^3 - 100$</th>
<th>Estimates in lbs.</th>
<th>$q_1$ Centiles</th>
<th>$q_3$ Centiles</th>
<th>Excess of Observed over Normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>Estimate</td>
<td>Deviates from 1207 lbs.</td>
<td>Observed</td>
<td>Normal</td>
<td></td>
</tr>
<tr>
<td>5</td>
<td>1074</td>
<td>-133</td>
<td>-90</td>
<td>+43</td>
</tr>
<tr>
<td>10</td>
<td>1109</td>
<td>-98</td>
<td>-70</td>
<td>+28</td>
</tr>
<tr>
<td>15</td>
<td>1126</td>
<td>-81</td>
<td>-57</td>
<td>+24</td>
</tr>
<tr>
<td>20</td>
<td>1148</td>
<td>-59</td>
<td>-46</td>
<td>+13</td>
</tr>
<tr>
<td>$q_1$ 25</td>
<td>1162</td>
<td>-45</td>
<td>-37</td>
<td>+8</td>
</tr>
<tr>
<td>30</td>
<td>1174</td>
<td>-33</td>
<td>-29</td>
<td>+4</td>
</tr>
<tr>
<td>35</td>
<td>1181</td>
<td>-26</td>
<td>-21</td>
<td>+5</td>
</tr>
<tr>
<td>40</td>
<td>1188</td>
<td>-19</td>
<td>-14</td>
<td>+5</td>
</tr>
<tr>
<td>45</td>
<td>1197</td>
<td>-10</td>
<td>-7</td>
<td>+3</td>
</tr>
<tr>
<td>$m$ 50</td>
<td>1207</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>55</td>
<td>1214</td>
<td>+7</td>
<td>+7</td>
<td>0</td>
</tr>
<tr>
<td>60</td>
<td>1219</td>
<td>+12</td>
<td>+14</td>
<td>-2</td>
</tr>
<tr>
<td>65</td>
<td>1225</td>
<td>+18</td>
<td>+21</td>
<td>-3</td>
</tr>
<tr>
<td>70</td>
<td>1230</td>
<td>+23</td>
<td>+29</td>
<td>-6</td>
</tr>
<tr>
<td>$q_3$ 75</td>
<td>1236</td>
<td>+29</td>
<td>+37</td>
<td>-8</td>
</tr>
<tr>
<td>80</td>
<td>1243</td>
<td>+36</td>
<td>+46</td>
<td>-10</td>
</tr>
<tr>
<td>85</td>
<td>1254</td>
<td>+47</td>
<td>+57</td>
<td>-10</td>
</tr>
<tr>
<td>90</td>
<td>1267</td>
<td>+52</td>
<td>+70</td>
<td>-18</td>
</tr>
<tr>
<td>95</td>
<td>1293</td>
<td>+86</td>
<td>+90</td>
<td>-4</td>
</tr>
</tbody>
</table>

$q_1$, $q_3$, the first and third quartiles, stand at $25^\circ$ and $75^\circ$ respectively. $m$, the median or middlemost value, stands at $50^\circ$. The dressed weight proved to be 1198 lbs.

---

**Sir Francis Galton**

*VOX POPULI*

*Nature (1907), No. 1949, Vol. 75, 450-451*

<table>
<thead>
<tr>
<th>Degrees of the length of Array 0°–100</th>
<th>Estimates in lbs.</th>
<th>* Centiles</th>
<th>Excess of Observed over Normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1074</td>
<td>-133</td>
<td>-90</td>
</tr>
<tr>
<td>10</td>
<td>1109</td>
<td>-98</td>
<td>-70</td>
</tr>
<tr>
<td>15</td>
<td>1126</td>
<td>-81</td>
<td>-57</td>
</tr>
<tr>
<td>20</td>
<td>1148</td>
<td>-59</td>
<td>-46</td>
</tr>
<tr>
<td>$q_1$ 25</td>
<td><strong>1162</strong></td>
<td>-45</td>
<td>-37</td>
</tr>
<tr>
<td>30</td>
<td>1174</td>
<td>-33</td>
<td>-29</td>
</tr>
<tr>
<td>35</td>
<td>1181</td>
<td>-26</td>
<td>-21</td>
</tr>
<tr>
<td>40</td>
<td>1188</td>
<td>-19</td>
<td>-14</td>
</tr>
<tr>
<td>45</td>
<td>1197</td>
<td>-10</td>
<td>-7</td>
</tr>
<tr>
<td>$m$ 50</td>
<td><strong>1207</strong></td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>55</td>
<td>1214</td>
<td>+7</td>
<td>+7</td>
</tr>
<tr>
<td>60</td>
<td>1219</td>
<td>+12</td>
<td>+14</td>
</tr>
<tr>
<td>65</td>
<td>1225</td>
<td>+18</td>
<td>+21</td>
</tr>
<tr>
<td>70</td>
<td>1230</td>
<td>+23</td>
<td>+29</td>
</tr>
<tr>
<td>$q_3$ 75</td>
<td><strong>1236</strong></td>
<td>+29</td>
<td>+37</td>
</tr>
<tr>
<td>80</td>
<td>1243</td>
<td>+36</td>
<td>+46</td>
</tr>
<tr>
<td>85</td>
<td>1254</td>
<td>+47</td>
<td>+57</td>
</tr>
<tr>
<td>90</td>
<td>1267</td>
<td>+52</td>
<td>+70</td>
</tr>
<tr>
<td>95</td>
<td>1293</td>
<td>+86</td>
<td>+90</td>
</tr>
</tbody>
</table>

- **Truth:** 1198
- **Percentiles:**
  - 25th : 1162
  - 50th : 1207
  - 75th : 1236
### Distribution of the estimates of the dressed weight of a particular living ox, made by 787 different persons.

<table>
<thead>
<tr>
<th>Degrees of the length of Array 0°—100</th>
<th>Estimates in lbs.</th>
<th>Observed deviates from 1207 lbs.</th>
<th>Normal p.e = 37</th>
<th>Excess of Observed over Normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1074</td>
<td>-133</td>
<td>-90</td>
<td>+43</td>
</tr>
<tr>
<td>10</td>
<td>1109</td>
<td>-98</td>
<td>-70</td>
<td>+28</td>
</tr>
<tr>
<td>15</td>
<td>1126</td>
<td>-81</td>
<td>-57</td>
<td>+24</td>
</tr>
<tr>
<td>20</td>
<td>1148</td>
<td>-59</td>
<td>-46</td>
<td>+13</td>
</tr>
<tr>
<td>25</td>
<td>1162</td>
<td>-45</td>
<td>-37</td>
<td>+8</td>
</tr>
<tr>
<td>30</td>
<td>1174</td>
<td>-33</td>
<td>-29</td>
<td>+4</td>
</tr>
<tr>
<td>35</td>
<td>1181</td>
<td>-26</td>
<td>-21</td>
<td>+5</td>
</tr>
<tr>
<td>40</td>
<td>1188</td>
<td>-19</td>
<td>-14</td>
<td>+5</td>
</tr>
<tr>
<td>45</td>
<td>1197</td>
<td>-10</td>
<td>-7</td>
<td>+3</td>
</tr>
<tr>
<td>50</td>
<td>1207</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>55</td>
<td>1214</td>
<td>+7</td>
<td>+7</td>
<td>0</td>
</tr>
<tr>
<td>60</td>
<td>1219</td>
<td>+12</td>
<td>+14</td>
<td>-2</td>
</tr>
<tr>
<td>65</td>
<td>1225</td>
<td>+18</td>
<td>+21</td>
<td>-3</td>
</tr>
<tr>
<td>70</td>
<td>1230</td>
<td>+23</td>
<td>+29</td>
<td>-6</td>
</tr>
<tr>
<td>75</td>
<td>1236</td>
<td>+29</td>
<td>+37</td>
<td>-8</td>
</tr>
<tr>
<td>80</td>
<td>1243</td>
<td>+36</td>
<td>+46</td>
<td>-10</td>
</tr>
<tr>
<td>85</td>
<td>1254</td>
<td>+47</td>
<td>+57</td>
<td>-10</td>
</tr>
<tr>
<td>90</td>
<td>1267</td>
<td>+52</td>
<td>+70</td>
<td>-18</td>
</tr>
<tr>
<td>95</td>
<td>1293</td>
<td>+86</td>
<td>+90</td>
<td>-4</td>
</tr>
</tbody>
</table>

- **Truth:** 1198
- **Percentiles:**
  - 25th : 1162
  - 50th : 1207
  - 75th : 1236
- **Mean:** 1197
<table>
<thead>
<tr>
<th>Degrees of the length of Array 0°—100</th>
<th>Estimates in lbs.</th>
<th>Observed deviates from 1207 lbs.</th>
<th>Normal p.e ( \pm 37 )</th>
<th>Excess of Observed over Normal</th>
</tr>
</thead>
<tbody>
<tr>
<td>5</td>
<td>1074</td>
<td>-133</td>
<td>90</td>
<td>+43</td>
</tr>
</tbody>
</table>

- **Truth:** 1198
- **Percentiles:**
  - 25th: 1162
  - 50th: 1207
  - 75th: 1236
- **Mean:** 1197

**Ensemble method:** average of predictors

The dressed weight proved to be 1198 lbs.

1. The first and third quartiles, stand at 25° and 75° respectively.
2. \( m \), the median or middlemost value, stands at 50°.
3. The mean of the Truth for month year. Both years, bulletin contains station results. In stations, estimates are employed in mean, approximate and applicable. Estimates are also obtained by averaging across all predictors.

) The absolute rain was Most tender, not being J. D.
Aggregation of predictions

Input instance → Human 1 → Prediction 1 → Averaging
Input instance → Human 2 → Prediction 2 → Averaging
Input instance → Human 3 → Prediction 3 → Averaging
... → Human m → Prediction m → Averaging

Aggregated prediction
A theoretical justification to averaging

• $t$ - truth; $x_1, x_2, \ldots, x_M$ - predictions

• $x_{rand}$ - randomly selecting one of these

$$
\epsilon_{rand} = \mathbb{E}[(x_{rand} - t)^2] = \frac{1}{M} \sum_{i=1}^{M} (x_i - t)^2
$$

• $\bar{x} = \frac{1}{M} \sum_{i=1}^{M} x_i$ - average prediction

$$
\epsilon_{aver} = (\bar{x} - t)^2
$$

• Well-known “ambiguity” decomposition:

$$
\epsilon_{rand} = \epsilon_{aver} + \frac{1}{M} \sum_{i=1}^{M} (x_i - \bar{x})^2
$$

$$
\implies \epsilon_{aver} \leq \epsilon_{rand} !!!!!!
$$
Reasons to use ensembles

- **Reasons** (From the 2004 book by Ludmila Kuncheva, *Combining pattern classifiers: methods and algorithms*)
  - **Statistical:**
    - Hopefully the ensemble generalises better than a single chosen model
  - **Computational**
    - Averaging can sometimes be a fast way of reaching closer to the optimal than direct optimisation
  - **Representational**
    - Averaging models of some model class can sometimes take you outside of that model class
In which supervised learning tasks can ensemble models be used?

A. Only classification
B. Only regression
C. Classification and regression
D. All supervised learning tasks
E. I don’t know

Response Counter:
- Only classification
- Only regression
- Classification and regression
- All supervised learning tasks
- I don’t know
Types of tasks in supervised learning where ensembles can be used

- Ensembles can be used in pretty much all supervised learning tasks
  - Classification
  - Regression
  - Structured output prediction
Theory in favour of voting (binary task)

- Marquis de Condorcet 1785: Essay on the Application of Analysis to the Probability of Majority Decisions
Voting in binary classification

Input instance

Voter 1

Voter 2

Voter 3

...  

Voter m

Vote 1

Vote 2

Vote 3

Vote m

Which class has more votes?

Aggregated prediction
Theory in favour of voting (binary task)

- Marquis de Condorcet 1785: Essay on the Application of Analysis to the Probability of Majority Decisions

- \( M \) voters, independent errors, individual error probability \( \epsilon \)

- Majority vote is wrong with probability:

\[
\sum_{k \geq \left\lceil \frac{M + 1}{2} \right\rceil} \binom{M}{k} \epsilon^k (1 - \epsilon)^{M-k}
\]
\[ p(\text{majority vote error}) = \sum_{k \geq \lceil \frac{M}{2} \rceil} \binom{M}{k} \epsilon^k (1 - \epsilon)^{M-k} \]

11 classifiers. Individual error probability = 0.3
Probability of voted ensemble error = 0.078225

Figure 4: Probability of voting errors.

Slide adapted from Gavin Brown
$p(\text{majority vote error}) = \sum_{k \geq \lceil \frac{M+1}{2} \rceil} \binom{M}{k} \epsilon^k (1 - \epsilon)^{M-k}$

21 classifiers. Individual error probability = 0.3
Probability of voted ensemble error = 0.02639

Slide adapted from Gavin Brown
\[ p(\text{majority vote error}) = \sum_{k \geq \lceil \frac{M+1}{2} \rceil} \binom{M}{k} \epsilon^k (1 - \epsilon)^{M-k} \]

Virtually ZERO error by \( M = 50 \) !!

Slide adapted from Gavin Brown
Why does this theory not work in practice?
Why does this theory not work in practice?

A. Too few training data
B. Individual models are not good enough
C. Errors of individual models are not independent
D. Not enough different learning algorithms exist
E. I don’t know

Response Counter:
- Too few training data: 1
- Individual models are not good enough: 1
- Errors of individual models are not independent: 1
- Not enough different learning algorithms exist: 1
- I don’t know: 1
Why does this theory not work in practice?

Errors of voters are not independent
Key challenge of ensemble learning

- The key challenge of ensemble learning is to obtain models that are:
  - reasonably accurate
  - as independent as possible
Why do we need ensemble methods?

- **Bagging**
- Random forest
- Weighted averaging
- Boosting
  - Intuitive explanation
  - AdaBoost algorithm
  - Alternative formulations
  - Interpretations
How to achieve independence of models?

- **Idea 1: Split+Train**
  - Randomly split training data into M disjoint groups of instances, train a separate model on each group.
How to achieve independence of models?

- **Idea 1: Split+Train**
  - Randomly split training data into M disjoint groups of instances, train a separate model on each group
How to achieve independence of models?

• **Idea 1: Split+Train**
  
  – Randomly split training data into M disjoint groups of instances, train a separate model on each group
  
  – Bad because groups get small and the learned models will have poor prediction quality
How to achieve independence of models?

• Idea 2: Split by features+Train
  – Randomly split training data into $M$ disjoint groups of features, train a separate model on each group
How to achieve independence of models?

- Idea 2: Split by features + Train
  - Randomly split training data into M disjoint groups of features, train a separate model on each group
  - Lack of good features can be even worse than lack of instances
How to achieve independence of models?

- Idea 3: Overlapping subsets of instances + Train
  - Randomly sample M overlapping groups of instances, train a separate model on each group
How to achieve independence of models?

- Idea 3: Overlapping subsets of instances + Train
  - Randomly sample $M$ overlapping groups of instances, train a separate model on each group
  - Not too bad, but still smaller training sets
How to achieve independence of models?

- **Idea 4: Bootstrap-sampled instances + Train**
  - Sample with replacement M overlapping groups of instances with same size as original (bootstrapping), train a separate model on each group

```
  instances  features  label
```

37
Generating a new dataset by “Bootstrapping”

- **Bootstrapping:**

  - Sample N items with replacement from the original N training instances

<table>
<thead>
<tr>
<th>id</th>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.18</td>
<td>0.45</td>
<td>0.8</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.11</td>
<td>0.82</td>
<td>0.07</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0.87</td>
<td>0.3</td>
<td>0.21</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0.34</td>
<td>0.49</td>
<td>0.18</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0.95</td>
<td>0.64</td>
<td>0.63</td>
<td>0</td>
</tr>
<tr>
<td>6</td>
<td>0.03</td>
<td>0.59</td>
<td>0.15</td>
<td>1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>id</th>
<th>x1</th>
<th>x2</th>
<th>x3</th>
<th>y</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.18</td>
<td>0.45</td>
<td>0.8</td>
<td>0</td>
</tr>
<tr>
<td>1</td>
<td>0.18</td>
<td>0.45</td>
<td>0.8</td>
<td>0</td>
</tr>
<tr>
<td>3</td>
<td>0.87</td>
<td>0.3</td>
<td>0.21</td>
<td>1</td>
</tr>
<tr>
<td>4</td>
<td>0.34</td>
<td>0.49</td>
<td>0.18</td>
<td>1</td>
</tr>
<tr>
<td>5</td>
<td>0.95</td>
<td>0.64</td>
<td>0.63</td>
<td>0</td>
</tr>
<tr>
<td>5</td>
<td>0.95</td>
<td>0.64</td>
<td>0.63</td>
<td>0</td>
</tr>
</tbody>
</table>

Original data

Bootstrap 1

Bootstrap 2

Slide adapted from Gavin Brown
Is it possible that a particular training instance is not included in the bootstrap sample?

A. Not possible
B. Possible, but only when the training set is small
C. Possible, happens with less than 50% probability
D. Possible, happens with more than 50% probability
E. I don’t know

Response Counter

- Not possible: 1
- Possible, but only when the training set is small: 1
- Possible, happens with less than 50% probability: 1
- Possible, happens with more than 50% probability: 1
- I don’t know: 1
Bagging: Bootstrap AGGregating (Breiman 1996)

• Bootstrap sampling:
  – Sample with replacement $M$ overlapping groups of instances with the same size as original
  – That is, each original instance will have 0, 1, or more copies in such a group
  – On average, each sample will have $\frac{1}{e} \approx 63.2\%$ of original instances represented in the group

• Bagging (Bootstrap AGGregating = BAGG):
  – Train a model separately on each bootstrapped dataset and then aggregate the results
Is it possible that a particular training instance is not included in any of the M bootstrapped datasets?

A. Not possible
B. Possible only when M is small
C. Possible only when the training set size N is small
D. Possible, but happens less and less frequently as M grows
E. Possible, but happens less and less frequently as N grows
F. I don’t know

Response Counter

- Possible only when M is small: 1
- Possible only when the training set size N is small: 1
- Possible, but happens less and less frequently as M grows: 1
- Possible, but happens less and less frequently as N grows: 1
- I don’t know: 1
\[ \text{Prob( excluding an example from the whole ensemble )} = \]
The same principle can be used to plot theoretical voting error as the number of voters increases. Figure 5 shows this, for different values of $\varepsilon$. Note that the error approaches zero as the number of voters increases — though always under the assumption that the voters make statistically independent errors.

Figure 5: Theoretical majority voting error as we increase the ensemble size (number of voters), assuming errors are statistically independent.

If each model has only 20% error rate ($\varepsilon = 0.2$), the ensemble will have < 1% error if we combine 15 of them, or < 0.1% if we combine 20 of them.

Whilst we could continue our historical journey even further back to Plato and the study of democracy in ancient Greece, we should turn our attention to machine learning classifiers. An interesting question is therefore, how realistic is this independence assumption — if we generate classifiers, will they be independent? Obviously if we train two classifiers from identical training sets, the only difference between them will be their training algorithms. If the training is the same, then we get identical models. So, what if we took our training set, and divided it amongst the classifiers? If we had for 2000 examples, we could make two classifiers and give them 1000 examples each. We know the usual assumption that our training data is independent and identically distributed — so surely this will result in perfectly independent classifiers? We can extend this idea – dividing the 2000 examples amongst 3, 4, or more classifiers. The results of this process are plotted below, using two different types of classifier.

Slide adapted from Gavin Brown
Independence test between errors of bagged decision trees.

\( (\chi^2 \text{ test, } \alpha = 0.05) \)

White pixel indicates significant correlation between errors of 2 individual decision trees in the ensemble.

Slide adapted from Gavin Brown
When is bagging useful?

- Bagging is bad if models are very similar (not independent enough)
- This happens if the learning algorithm is stable
  - That is, model does not usually change much after changing a few instances

SVM?

Very unstable = High variance
Very stable = Low variance
When is bagging useful?

- Bagging is bad if models are very similar (not independent enough)
- This happens if the learning algorithm is stable
  - That is, model does not usually change much after changing a few instances

KNN? SVM

Very unstable = High variance

Very stable = Low variance
When is bagging useful?

• Bagging is bad if models are very similar (not independent enough)
• This happens if the learning algorithm is stable
  – That is, model does not usually change much after changing a few instances

Neural nets? KNN SVM

Very unstable = Very stable =
High variance = Low variance

Meelis Kull - Fall 2019 - MTAT.03.227 - Machine Learning - Lecture 10
When is bagging useful?

• Bagging is bad if models are very similar (not independent enough)

• This happens if the learning algorithm is stable
  – That is, model does not usually change much after changing a few instances

<table>
<thead>
<tr>
<th>Very unstable</th>
<th>Very stable</th>
</tr>
</thead>
<tbody>
<tr>
<td>= High variance</td>
<td>= Low variance</td>
</tr>
</tbody>
</table>

Decision trees: SVM, KNN, Neural nets.
When is bagging useful?

• Bagging is bad if models are very similar (not independent enough)
• This happens if the learning algorithm is stable
  – That is, model does not usually change much after changing a few instances

Bagging?

Very unstable = High variance
Very stable = Low variance

SVM
KNN
Neural nets
Decision trees

Meelis Kull - Fall 2019 - MTAT.03.227 - Machine Learning - Lecture 10
When is bagging useful?

- Bagging is bad if models are very similar (not independent enough)
- This happens if the learning algorithm is stable
  - That is, model does not usually change much after changing a few instances
Bagging is strongly affected by the quality of individual models.

Bagged decision stumps
(trees with 1 decision node)

Figure by Raivo Kolde, https://courses.cs.ut.ee/2012/ml/uploads/Main/lecture-18.pdf
Summary of Bagging

• Individual models trained on bootstrap-sampled instances, predictions are aggregated

• Bagging is useful when the algorithm to learn individual models is:
  – Relatively accurate
  – Relatively unstable (high variance)

• The aggregated model is then usually better than the original model trained on full dataset
Why do we need ensemble methods?

Bagging

- Random forest

- Weighted averaging

- Boosting
  - Intuitive explanation
  - AdaBoost algorithm
  - Alternative formulations
  - Interpretations
Random forests (Breiman 2000):

- Random forests: similar to bagged decision trees but different in using features

- **In each recursive step** of learning decision trees:
  - Randomly select $F$ features out of all $P$ given features
  - Find the best split among these features

- Parameter $F$ is usually fixed to be
  - $F=\sqrt{P}$ for classification
  - $F=P/3$ for regression
Figure 13: Bagging (LEFT) vs Random Forests (RIGHT) on the Splice dataset.

What do you notice about the starting point, for a single bagged/RF’d tree?

Slide adapted from Gavin Brown
Real-time human pose recognition in parts from single depth images

Computer Vision and Pattern Recognition 2011
Shotton et al, Microsoft Research

- Basis of Kinect controller
- Features are simple image properties
- Test phase: 200 frames per sec on GPU
- Train phase more complex but still parallel

To keep the training times down we employ a distributed implementation. Training 3 trees to depth 20 from 1 million images takes about a day on a 1000 core cluster.

Figure 4. Randomized Decision Forests. A forest is an ensemble of trees. Each tree consists of split nodes (blue) and leaf nodes (green). The red arrows indicate the different paths that might be taken by different trees for a particular input.

3.3. Randomized decision forests

Randomized decision trees and forests [35, 30, 2, 8] have proven fast and effective multi-class classifiers for many tasks [20, 23, 36], and can be implemented efficiently on the GPU [34]. As illustrated in Fig. 4, a forest is an ensemble of $T$ decision trees, each consisting of split and leaf nodes.
How many trees in a forest?
How many trees in a forest?

- The more the better!
- How do I know there are enough?
- Out-of-bag (OOB) error
  - For each training instance make a prediction using trees that do not use that instance and evaluate
  - Stabilisation of OOB error suggests that there are enough trees
Typically values for \( m \) are \( \sqrt{p} \) or even as low as 1. After \( B \) such trees \( \{ T(x; \Theta_b) \} \) are grown, the random forest (regression) predictor is

\[
\hat{f}_{\text{RF}}(x) = \frac{1}{B} \sum_{b=1}^{B} T(x; \Theta_b).
\]

As in Section 10.9 (page 356), \( \Theta_b \) characterizes the \( b \)th random forest tree in terms of split variables, cutpoints at each node, and terminal-node values.

Intuitively, reducing \( m \) will reduce the correlation between any pair of trees in the ensemble, and hence by (15.1) reduce the variance of the average.

FIGURE 15.1. Bagging, random forest, and gradient boosting, applied to the spam data. For boosting, 5-node trees were used, and the number of trees were chosen by 10-fold cross-validation (2500 trees). Each “step” in the figure corresponds to a change in a single misclassification (in a test set of 1536).

Not all estimators can be improved by shaking up the data like this. It seems that highly nonlinear estimators, such as trees, benefit the most. For bootstrapped trees, \( \rho \) is typically small (0.05 or lower is typical; see Figure 15.9), while \( \sigma^2 \) is not much larger than the variance for the original tree. On the other hand, bagging does not change linear estimates, such as the sample mean (hence its variance either); the pairwise correlation between bootstrapped means is about 50% (Exercise 15.4).
Why do we need ensemble methods?

- Bagging
- Random forest

- **Weighted averaging**
- Boosting
  - Intuitive explanation
  - AdaBoost algorithm
  - Alternative formulations
  - Interpretations
Homogeneous and Heterogeneous ensembles

• Homogeneous – all individual models are obtained with the same learning algorithm, on slightly different datasets

• Heterogeneous – individual models are obtained with different algorithms
Aggregation of predictions

• Classification:
  – Voting
  – Weighted voting
Weighted voting

Model 1 → Prediction 1
Model 2 → Prediction 2
Model 3 → Prediction 3
Model m → Prediction m

Which class has the most total weight

Aggregated prediction

$w_1, \ldots, w_m$ - weights for votes / predictions
Aggregation of predictions

• Classification:
  – Voting
  – Weighted voting

• Regression:
  – Averaging
  – Weighted averaging

• Better models should have higher weights

• How to obtain weights?
Bayesian model averaging (BMA)

• Suppose we know that one of the M models is the true model but we do not know which

• \(X\) – data; \(T\) – index of true model; \(Y\) – true class; \(\hat{y}_1, \ldots, \hat{y}_M\) – predictions of M models

\[
P(Y|X) = \sum_{t=1}^{M} P(Y, T = t|X) = \sum_{t=1}^{M} P(Y|T = t, X)P(T = t|X) = \sum_{t=1}^{M} w_t \hat{y}_t
\]

• This is weighted averaging of model-specific posterior class probabilities \(\hat{y}_t = P(Y|T = t, X)\)

• Assuming uniform prior over models, the weights are likelihoods of models:

\[
w_t = P(T = t|X) = \frac{P(X|T = t)P(T = t)}{P(X)} \propto P(X|T = t)
\]
Bayesian model averaging (BMA)

- Suppose we know that one of the M models is the true model but we do not know which
- $X$ – data; $T$ – index of true model; $Y$ – true class; $\hat{y}_1, \ldots, \hat{y}_M$ – predictions of M models

An example and further intuition will be given in the lecture on Bayesian machine learning

- This is weighted averaging of model-specific posterior class probabilities $\hat{y}_t = P(Y|T = t, X)$
- Assuming uniform prior over models, the weights are likelihoods of models:

$$w_t = P(T = t|X) = \frac{P(X|T = t)P(T = t)}{P(X)} \propto P(X|T = t)$$
Stacking

• Often we have no way of estimating model likelihoods reliably
• Instead, we can learn the weights in a linear classification task where the individual model outputs are treated as features
• This is known as stacking, because we stack one classifier on top of many individual classifiers
Stacking

Input instance → Model 1 → Prediction 1
Input instance → Model 2 → Prediction 2
Input instance → Model 3 → Prediction 3
... → Model m → Prediction m

Predictions are aggregated by an aggregating model to produce an aggregated prediction.
Lecture 10 – Ensemble methods

✓ Why do we need ensemble methods?
✓ Bagging
✓ Random forest
✓ Weighted averaging

• **Boosting**
  – Intuitive explanation
  – AdaBoost algorithm
  – Alternative formulations
  – Interpretations