Advanced Machine Learning Subsidary Notes

Lecture 5: Ensemble Methods

Adam Prügel-Bennett

January 19, 2024

1 Keywords

- Decision Trees, Averaging, Bagging

2 Main Points

2.1 Decision Trees

- Decision trees are binary tree where a set of data is partitioned into two at each node of the tree
- Rule for partition depends on a single feature
- Trees are grown greedily
- All possible rules are tried to find the one that maximise the purity of the leaves
- Gini, entropy or variance used to measure purity
- Decision trees can handle categorical or numerical data
- They can handle missing data
  - after deciding on a new rule (ignoring missing data), a rule to send the missing data right or left is selected to maximise purity
- Decision trees are useful for understanding new data sets
  - By examining the rules at the top of the tree we get to see the most important variables
- Decisions trees are often not competitive
  - they can have high bias because they can only split the data on single variables (limits their expressiveness)
  - they can also have high variance because a small change to the data set that changes a rule at the top of the tree can lead to very different predictions

2.2 Ensembling

- This is used to reduces variance by averaging many different machines
- This only works if the machines make weakly correlated classifications
- We ensemble machines by coming to a consensus
  - Vote on a class in classification
  - Average for regression

2.2.1 Estimating the Mean

- Suppose we want to estimate the mean of a distribution, $f_X(x)$ from random samples of the distribution. We assume the distribution has mean $\mu = \mathbb{E}[X]$ and variance $\sigma^2 = \mathbb{E}\left[(X-\mu)^2\right]$

- Our estimated mean is
$$\hat{\mu} = \frac{1}{n}\sum_{i=1}^{n} X_i$$
where $X_i \sim f_X(x)$ (i.e. the X_i are drawn at random form the distribution)

- In expectation
$$\mathbb{E}[\hat{\mu}] = \frac{1}{n}\sum_{i=1}^{n}\mathbb{E}[X_i] = \frac{1}{n}\sum_{i=1}^{n}\mu = \mu$$

- However, each $\hat{\mu}$ will differ from the true μ (some will e greater and some less than μ)

- We want to measure the variance in $\hat{\mu}-\mu$
$$\hat{\mu}-\mu = \left(\frac{1}{n}\sum_{i=1}^{n} X_i\right) - \mu = \frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)$$

- Thus
$$(\hat{\mu}-\mu)^2 = \left(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)\right)^2 = \left(\frac{1}{n}\sum_{i=1}^{n}(X_i-\mu)\right)\left(\frac{1}{n}\sum_{j=1}^{n}(X_j-\mu)\right)$$
  - Note that i and j are just dummy indices (it doesn't matter what they are called but they are different)

- Or
$$(\hat{\mu}-\mu)^2 = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}(X_i-\mu)(X_j-\mu)$$
  - Now there are times when $i=j$ and times when this isn't true so we separate these two out
$$(\hat{\mu}-\mu)^2 = \frac{1}{n^2}\sum_{i=1}^{n}(X_i-\mu)^2 + \sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n}(X_i-\mu)(X_j-\mu)$$

- Taking expectations
$$\mathbb{E}\left[(\hat{\mu}-\mu)^2\right] = \frac{1}{n^2}\sum_{i=1}^{n}\mathbb{E}\left[(X_i-\mu)^2\right] + \sum_{i=1}^{n}\sum_{\substack{j=1\\ j\neq i}}^{n}\mathbb{E}[(X_i-\mu)(X_j-\mu)]$$
  - where I have used $\mathbb{E}[A+B] = \mathbb{E}[A]+\mathbb{E}[B]$ (so that $\mathbb{E}[\sum_i A_i] = \sum_i \mathbb{E}[A_i]$)

- Now because X_i and X_j are by assumption independently chosen
$$\mathbb{E}[(X_i-\mu)(X_j-\mu)] = \mathbb{E}[X_i-\mu]\ \mathbb{E}[X_j-\mu]$$
but $\mathbb{E}[X_i-\mu] = \mathbb{E}[X_i]-\mu = \mu-\mu = 0$ so
$$\mathbb{E}[(X_i-\mu)(X_j-\mu)] = 0$$
and
$$\mathbb{E}\left[(\hat{\mu}-\mu)^2\right] = \frac{1}{n^2}\sum_{i=1}^{n}\mathbb{E}\left[(X_i-\mu)^2\right] = \frac{1}{n^2}\sum_{i=1}^{n}\sigma^2 = \frac{\sigma^2}{n}$$

2.2.2 Bagging

- Uses bootstrap aggregation
  - that is we sample the training set with replacement to obtain slightly different datasets
- Random Forest
  - Uses decision trees with bootstrap aggregation
  - But also use different random subsets of features
  - Often gives state-of-the-art performance

3 Exercises

3.1 Compute expected error in mean for correlated variables

- We look at estimating the mean by sampling n random variables

$$\hat{\mu} = \frac{1}{n} \sum_{i=1}^{n} X_i$$

- In expectation $\mathbb{E}[X_i] = \mu$
- Thus in expectation $\mathbb{E}[\hat{\mu}] = \mu$
- But there will typically be fluctuations from the mean which we can compute using

$$\sigma_{\hat{\mu}}^2 = \mathbb{E}\left[\left(\hat{\mu} - \mu\right)^2\right]$$

- **Compute this** using

$$\mathbb{E}[(X_i - \mu)^2] = \sigma^2 \qquad\qquad \mathbb{E}[(X_i - \mu)\,(X_j - \mu)] = \rho\,\sigma^2$$

- Answer given in lecture notes
- Note that the estimated error in the mean is $\sigma_{\hat{\mu}}$ (this is what you use when you compute error-bars)

4 Experiments

4.1 Visualise a decision tree

```python
from sklearn.datasets import load_iris
from sklearn import tree
X, y = load_iris(return_X_y=True)
clf = tree.DecisionTreeClassifier()
clf = clf.fit(X, y)

tree.plot_tree(clf.fit(iris.data, iris.target))
```