Ensemble methods

Streams processing

Bagging (Bootstrap aggregating)

- Wisdom of crowds
- Leverages independent weak models
- Commonly used to reduce variance within a noisy dataset
- Batch version proposed by Leo Breiman
  - Bootstrapping
  - Parallel training
  - Aggregation

Stage one: Bootstrapping

- Bagging leverages a bootstrapping sampling technique to create diverse samples.
- This resampling method generates different subsets of the training dataset by selecting data points at random and with replacement.
- This means that each time you select a data point from the training dataset, you are able to select the same instance multiple times.
- As a result, a value/instance repeated twice (or more) in a sample.

Example: Bootstrapping the training data

- Dataset

$$\mathcal{D} = \{A, B, C, D\}$$

- Provide six bootstrapped datasets from D.

Stage two: parallel training

- These bootstrap samples are then trained independently and in parallel with each other using weak or base learners.

- Why can we run training in parallel?

Stage three: Aggregation

- Finally, depending on the task (i.e. regression or classification), an average or a majority of the predictions are taken to compute a more accurate estimate.

- In the case of regression, an average is taken of all the outputs predicted by the individual classifiers; this is known as soft voting.

- For classification problems, the class with the highest majority of votes is accepted; this is known as hard voting or majority voting.

Aggregation for regression

$$f(x) = \frac{1}{N} \sum_{n=1}^{N} f_n(x)$$

- Example: Compute prediction f from four regressors

$$\{f_1(x_0) = 4, f_2(x_0) = 3.5, f_3(x_0) = 3, f_4(x_0) = 3.2\}$$

Aggregation for classification

- Aggregated (binary) classification

$$f(x) = \text{sign}\left(\sum_{n=1}^{N} f_n(x)\right)$$

  - Example: Compute prediction f from four classifiers

$$\{f_1(x_0) = 0.3, f_2(x_0) = -0.01, f_3(x_0) = -0.04, f_4(x_0) = -0.03\}$$

- Majority vote

$$f(x) = \text{sign}\left(\sum_{n=1}^{N} \text{sign}\left(f_n(x)\right)\right)$$

  - Example: Compute prediction f from four classifiers

Bagging in a nutshell (batch)

- Given a training dataset
- Sample T sets of n elements from D (with replacement)
- Train a weak learner on each dataset and obtain a sequence of T outputs
- Prediction by aggregation
  - Regression
  - Classification

Online bagging for streams

- How to draw a sample with replacement from a stream?
- What happens to the loss when we sample with replacement?

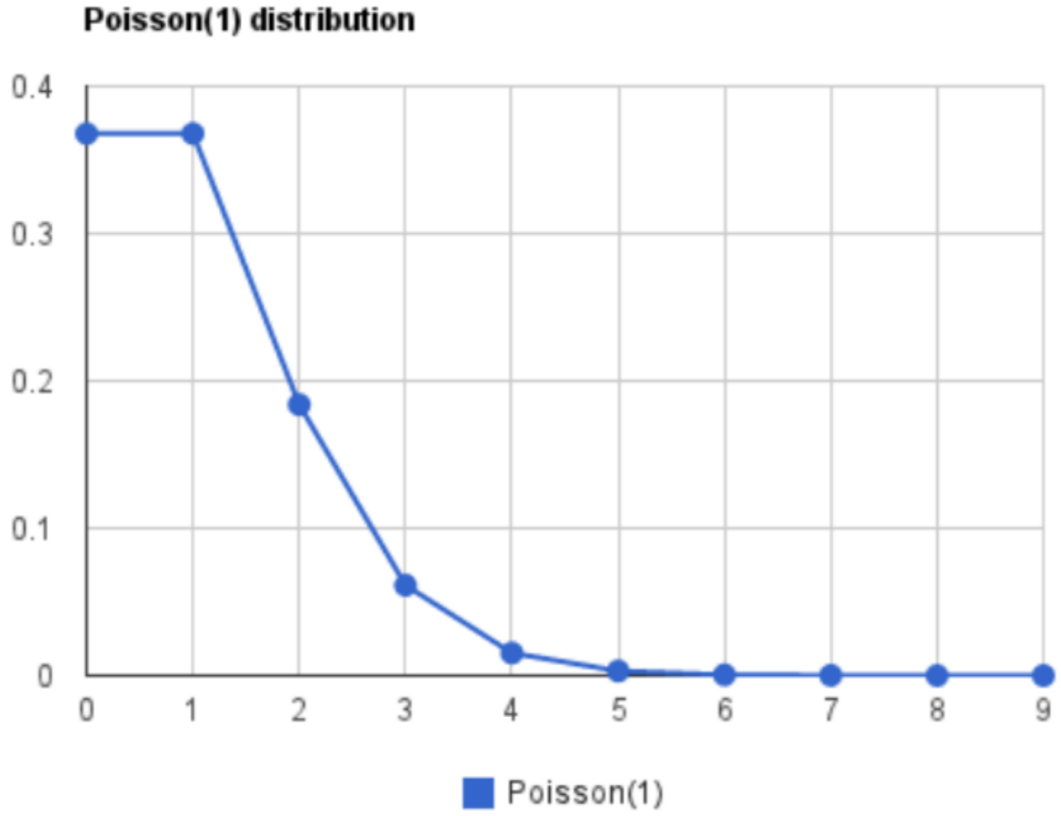
- In bootstrapping, the number of repeated samples K in the dataset with N data points follows a binomial distribution

$$P(K = k) = \binom{N}{k} \frac{1}{N}^{k} \left(1 - \frac{1}{N}\right)^{N-k}$$

- For large N, the binomial tends to the Poisson(1)

Simulating sampling with replacement

- Instead of resampling with replacement, we will weigh each new point in the stream according to the Poisson(1)



The algorithm

1: Initialize base models h_m for all $m \in \{1, 2, ..., M\}$
2: **for all** training examples **do**
3: **for** $m = 1, 2, ..., M$ **do**
4: Set $w =$ *Poisson*(1)
5: Update h_m with the current example with weight w

6: **anytime output:**
7: **return** hypothesis: $h_{fin}(x) = \arg\max_{y \in Y} \sum_{t=1}^{T} I(h_t(x) = y)$

In river: BaggingClassifier

Bagging with a change detector: ADWIN bagging

- Uses M instances of ADWIN
- When one of the M ADWIN detects a change
  - The worst classifier is removed
  - A new one is trained

- In river, implemented as ADWINBaggingClassifier

Adaptive Random Forest

- Effective resampling method
- Adaptive operators that can cope with different types of concept drift
- The parallel implementation
- Shows no degradation in terms of classification performance compared to a serial implementation, since trees and adaptive operators are independent of one another.

- In river: AdaptiveRandomForestClassifier

Regression in river

- BaggingRegressor
- AdaptiveRandomForestRegressor
- EWARegressor