Machine Learning and Data Mining

Meta-algorithms

Maxim Borisyak
National Research University Higher School of Economics (HSE)

September 19, 2017

In the last episode

```python
def data_science(problem_description,
                 domain_expertise=None,
                 *args, **kwargs):
  if problem_description is None:
    raise Exception('Learning is impossible!')

  prior_on_algorithms = \
    data_scientist.world_knowledge.infer(
      problem_description,
      domain_expertise,
      *args, **kwargs
    )

  return prior_on_algorithms
```

Making algorithms

Constructing learning algorithms from scratch is hard:

- it is the reason people use machine learning instead of classical statistical approach.

- producing tons of simple, rude algorithms is quite easy;
- fitting all-powerfull zero-bias classifier is easy.

Can an good algorithm be assembled from a set of simple ones?

Bootstrap

Settings

Suppose we have a quite good learning algorithm $f(x, D)$ where:

- D is a dataset,
- x is a point of interest,

with **high variance** and **low bias**.

What is the most common way of decreasing variance of mean estimate of a random variable?

Bootstrap

Let's consider average over multiple datasets:

$$F(x) = \frac{1}{n}\sum_i f(x, D_i) \approx \mathop{\mathbb{E}}_{D \sim P^n(X,Y)} f(x, D) = \hat{F}(x)$$

If D_i are i.i.d:

- $F(x)$ would reduce variance.

If D_i are correlated (via $f(x, D_i)$):

$$\mathbb{D}\left[\frac{1}{n}\sum_i Z_i\right] = \frac{\sigma^2}{n}\left(1 + (n-1)\rho\right) \rightarrow_{n \to \infty} \rho$$

where:

- $\mathbb{D}\left[Z_i\right] = \sigma^2$, $\rho = \mathrm{corr}(Z_i, Z_j)$ $(i \neq j)$.

Non-parametric bootstrap

Let's approximate $P(X, Y)$ by $\mathbb{U}\{D\}$:

- consider $D_i = \{(x^i_j, y^i_j)\}_{j=1}^m$ drawn i.i.d from D with replacement:

$$F(x) = \sum_{D_i \sim \mathbb{U}^m\{D\}} f(x, D_i)$$

- it will reduce variance.

Seems like model's variance was reduced for 'free', where is the catch?

Parametric bootstrap

If we have a sacred knowledge then we can:

- **using D produce more accurate $\hat{P}(X, Y)$ than $\mathbb{U}^n \{D\}$**

E.g. for regression:

$$D_i = \{(x_i, y_i + \varepsilon)\}_{i=1}^{N}$$

where:

- $\varepsilon \sim \mathcal{N}(0, \sigma_\varepsilon)$

Parametric bootstrap

> *...the bootstrap mean is approximately a posterior average ...*

For details:
Hastie, T., Tibshirani, R. and Friedman, J., 2001. The elements of statistical learning, ser., chapter 8

Bootstrap: a note

Sometimes we can produce more diverse $\{f(x, D_i)\}_i$ by training on feature subsets.

Stacking: settings

Bayesian averaging:

- ζ - variable of our interest (e.g. f(x));
- $\mathcal{M}_m$, $m = 1, \ldots, M$ - a candidate models;
- D - training dataset.

$$\mathbb{E}(\zeta \mid D) = \sum_m \mathbb{E}(\zeta \mid \mathcal{M}_m, D) P(\mathcal{M}_m \mid D) = \sum_m w_m \, \mathbb{E}(\zeta \mid \mathcal{M}_m, D)$$

$$w_m = P(\mathcal{M}_m \mid D)$$

Stacking: BIC

$$P(\mathcal{M}_m \mid D) \sim P(\mathcal{M}_m)P(D \mid \mathcal{M}_m)$$