

# No Free Lunch

Machine Learning and Data Mining

---

Maxim Borisyak

National Research University Higher School of Economics (HSE)

September 18, 2018

No free lunch

—

## IQ test: try to learn yourself!

---

First question from MENSA website:

*Following the pattern shown in the number sequence below, what is the missing number?*

1, 8, 27, ?, 125, 216

Possible answers:

- 36
- 45
- 46
- 64
- 99

## IQ test: try to learn yourself!

---

First question from MENSA website:

*Following the pattern shown in the number sequence below, what is the missing number?*

$X_{\text{train}}$	1	2	3	5	6
$y_{\text{train}}$	1	8	27	125	216

$$X_{\text{test}} = (4, )$$

## IQ test: try to learn yourself!

---

My solution:

$$y = \frac{1}{12}(91x^5 - 1519x^4 + 9449x^3 - 26705x^2 + 33588x - 14940)$$

- fits perfectly!

My answer:

- 99

Why solution:

$$y = x^3$$

seems much more suitable than

$$y = \frac{1}{12}(91x^5 - 1519x^4 + 9449x^3 - 26705x^2 + 33588x - 14940)?$$

# Terminology

---

Machine Learning is about learning algorithms  $A$  that:

- defined on sample set  $\mathcal{X}$  (e.g.  $\{0, \dots, 255\}^{28 \times 28}$ ) and targets  $\mathcal{Y}$  (e.g.  $\{0, 1, \dots, 9\}$ ):
  - $\mathcal{X}$  and  $\mathcal{Y}$  are discrete;
- take a problem (dataset)  $D = (X, y) \subseteq \mathcal{X} \times \mathcal{Y}$ ;
- learn relation between  $\mathcal{X}$  and  $\mathcal{Y}$ ;
- and return prediction function  $h$  (hypothesis):

$$A(D) = h$$

$$h : \mathcal{X} \rightarrow \mathcal{Y}$$

By this definition, e.g. XGBoost is a **family** of algorithms.

## Off-training-set error

---

$$\text{Err}(f, h, d) = \frac{\sum_{x \in d_X} \pi(x) \mathbb{I}[f(x) \neq h(x)]}{\sum_{x \in d_X} \pi(x)}$$

where:

- $h = A(d)$ ;
- $f$  - true dependency;

*In most practical cases,  $\text{Err} \approx \text{accuracy}$ .*



## No free lunch, strictly

---

For any two learning algorithms  $A_1$  and  $A_2$ :

1. uniformly averaged over all  $f$ , for any  $n$ :

$$\mathbb{E}(\text{Err} \mid f, |D| = n, A_1) - \mathbb{E}(\text{Err} \mid f, |D| = n, A_2) = 0$$

2. uniformly averaged over all  $d$ , for any  $f$ :

$$\mathbb{E}(\text{Err} \mid f, d, A_1) - \mathbb{E}(\text{Err} \mid f, d, A_2) = 0$$

3. uniformly averaged over all priors  $P(f)$ :

$$\mathbb{E}(\text{Err} \mid |D| = n, A_1) - \mathbb{E}(\text{Err} \mid |D| = n, A_2) = 0$$

4. uniformly averaged over all priors  $P(f)$ , for any  $d$ :

$$\mathbb{E}(\text{Err} \mid d, A_1) - \mathbb{E}(\text{Err} \mid d, A_2) = 0$$

## No free lunch theorem

---

No free lunch theorem states that **on average by all datasets** all learning algorithms are equally bad at learning.

Examples:

- crazy algorithm:

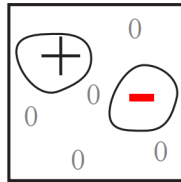
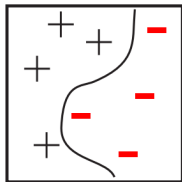
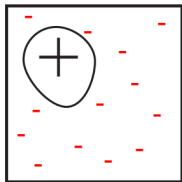
$$f(x) = \left[ \left( \left[ \sum_i x_i + \theta \right] \bmod 17 + 1027 \right)^\pi \right] \bmod 2$$

- any configuration of SVM

perform equally well **on average**.

# No free lunch theorem

---



Possible learning algorithm behaviours in **problem space**:

- + - better than the average;
- — - worse than the average.

# Is Machine Learning useless?

---

Is Machine Learning useless?

---

**No.**

## Assumptions and algorithms

---

# Is Machine Learning useless?

---

No Free Lunch theorem applies to:

- one learning algorithm;
- against all possible problems.

In real world we have:

- **data scientist** with prior knowledge of the world;
- problem description;
- data description;
- a set of standard algorithms.

# Is Machine Learning useless?

---

Real world problems often behave nicely:

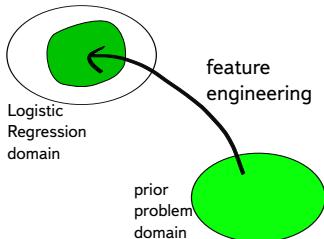
- data is collected by humans (features are determined by humans);
  - algorithms with human-bias dominate (e.g. XGboost);
- problems are posed by humans;
- a lot of assumptions behind the data can be quickly identified from the problem domain.



## Traditional ML (simplified)

---

- analyze the problem and make assumptions;
- pick an algorithm from a toolkit (e.g. logistic regression);
- provide assumptions suitable for the algorithm (**feature engineering**).



## Discussion

---

- this approach works well for traditional datasets with a small number of features:
- e.g. Titanic dataset:

passenger class	name	gender	age	fare	...
-----------------	------	--------	-----	------	-----

Essentially, performance of the algorithm depends on:

- knowledge of the domain;
- feature engineering skills;
- understanding of assumptions behind standard algorithms.

What are the assumptions behind:

- logistic regression,
- decision trees,
- linear SVM,
- SVM with RBF kernel?

# Discussion

---

What about:

- cross-validation-based algorithm selection?

What if we consider accuracy instead of off-training-set accuracy?

What if we consider accuracy instead of off-training-set accuracy?

$$\text{accuracy} = \text{generalization} + \text{memorization}$$

What about other losses?

## Representation matters

---

$x_1$	$x_2$	$x_3$	$y$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0



# Representation matters

---

$x_1$	$x_2$	$x_3$	$y$
0	0	0	0
0	0	1	0
0	1	0	1
0	1	1	0
1	0	0	0
1	0	1	1
1	1	0	0
1	1	1	0

$$x = 4 \cdot x_1 + 2 \cdot x_2 + x_3$$

$x$	$y$
0	0
1	0
2	1
3	0
4	0
5	1
6	0
7	0

$$y = \begin{cases} 1, & x \bmod 3 = 2; \\ 0, & \text{otherwise} \end{cases}$$

Solve with a descent algorithm:

$$(x - 8)^2 \rightarrow \min$$

where:  $x \in \{0, 1, \dots, 15\}$

- $\text{neighbors}(x) = x \pm 1$ ;
- $\text{neighbors}(x) = \{z \mid \|\text{binary}(x) - \text{binary}(z)\|_1 = 1\}$

# Algorithms

---

What makes a good family of learning algorithms (ML library)?

## Corollary from No-Free-Lunch

---

A good machine learning family of algorithms/framework:

- has clear relation between hyperparameters and set of problems each algorithm covers.

A great machine learning family/frameworks:

- covers a wide range of problems;
- but each algorithm covers a small set of problems;
- i.e. a lot of sensitive and well-defined hyperparameters.

*Here feature engineering/selection/generation is a part of the algorithm.*



[Website](#) | [Documentation](#) | [Installation](#)

build passing pypi package 0.1.1.9

CatBoost is a machine learning method based on gradient boosting over

Main advantages of CatBoost:

- Superior quality when compared with other libraries.
- Support for both numerical and categorical features.

## Summary

---

## No-Free-Lunch:

- learning is impossible without prior knowledge;
- there is no silver bullet for learning;
- every learning algorithm has assumptions behind it;
- **data scientist's** job is to select/make an algorithm to match the assumptions.



## References

---

### No-Free-Lunch theorem:

- Schaffer, Cullen. "A conservation law for generalization performance." Proceedings of the 11th international conference on machine learning. 1994.
- Wolpert, David H. "The supervised learning no-free-lunch theorems." Soft computing and industry. Springer London, 2002. 25-42.
- Wolpert, David H., and William G. Macready. "No free lunch theorems for optimization." IEEE transactions on evolutionary computation 1.1 (1997): 67-82.