Bias-variance decomposition Gradient boosting

Radoslav Neychev

girafe ai

YSDA, Fall 2024

Outline

1. Intuitions
2. Gradient boosting theory
3. Examples
4. Libraries
5. Feature importances
6. Hyperparameter optimization

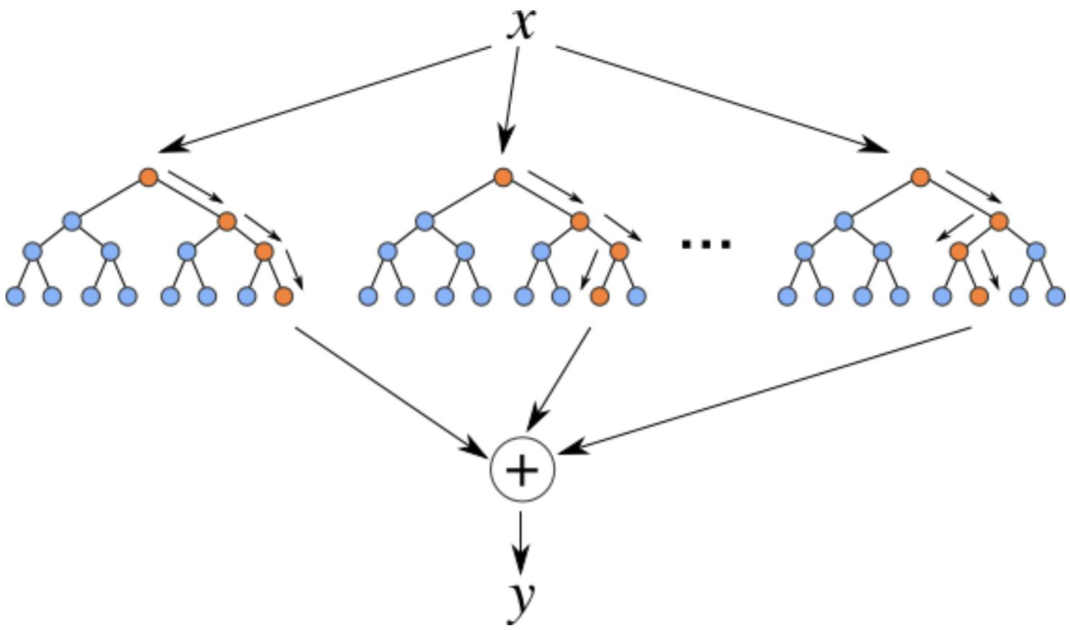
Ensembling recap

girafe
ai

01

Random Forest

Bagging + RSM = Random Forest



Random Forest

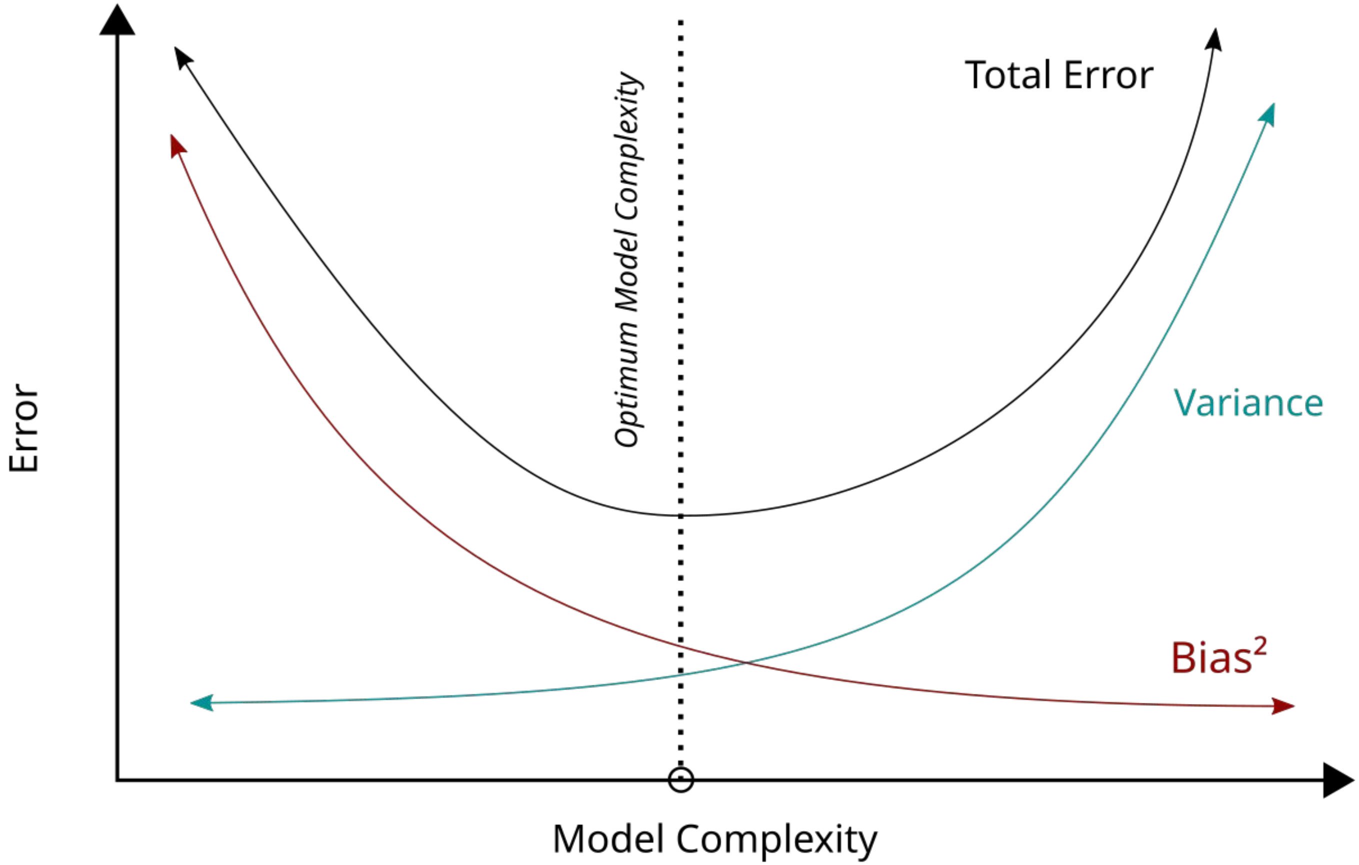
- One of the greatest “universal” models
- There are some modifications: Extremely Randomized Trees, Isolation Forest, etc.

Bias-variance tradeoff

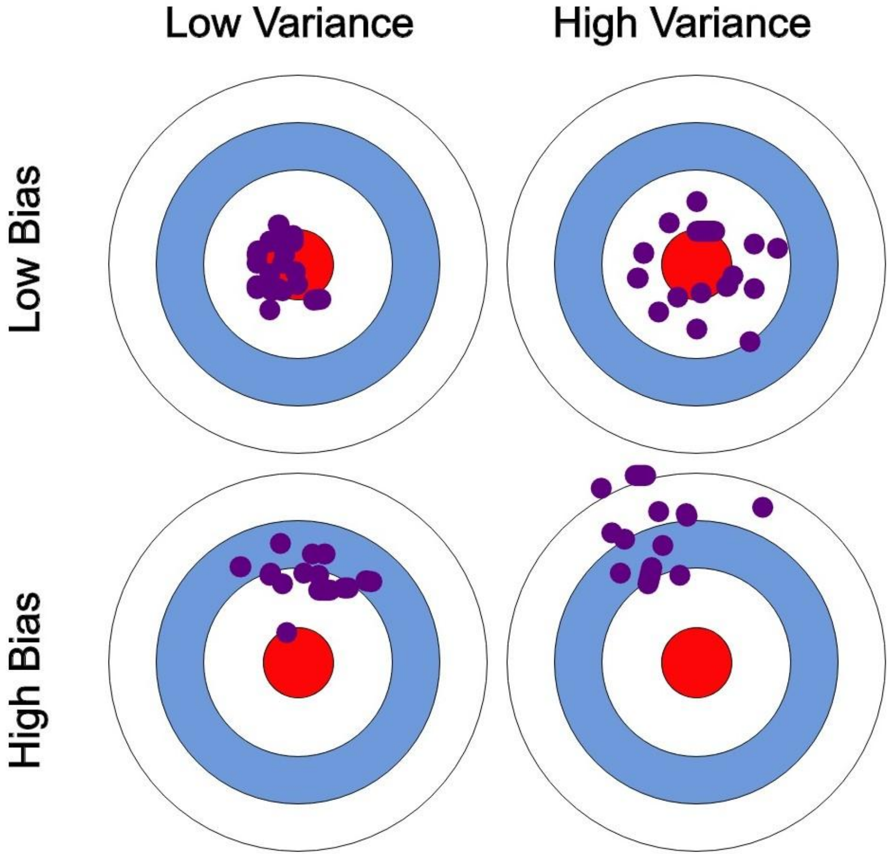
girafe
ai

00

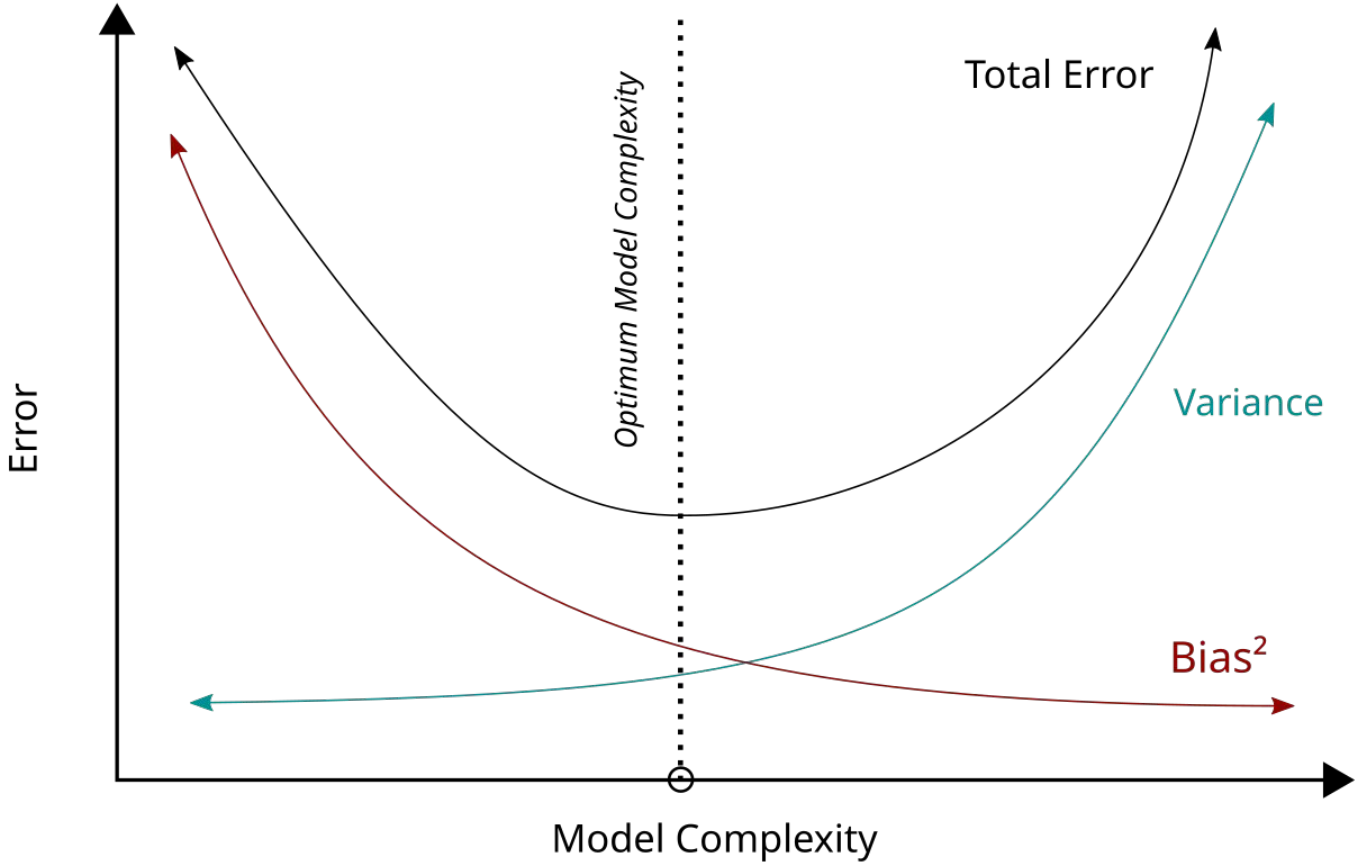
Bias-variance tradeoff



Bias-variance tradeoff



Bias-variance tradeoff



Boosting intuition

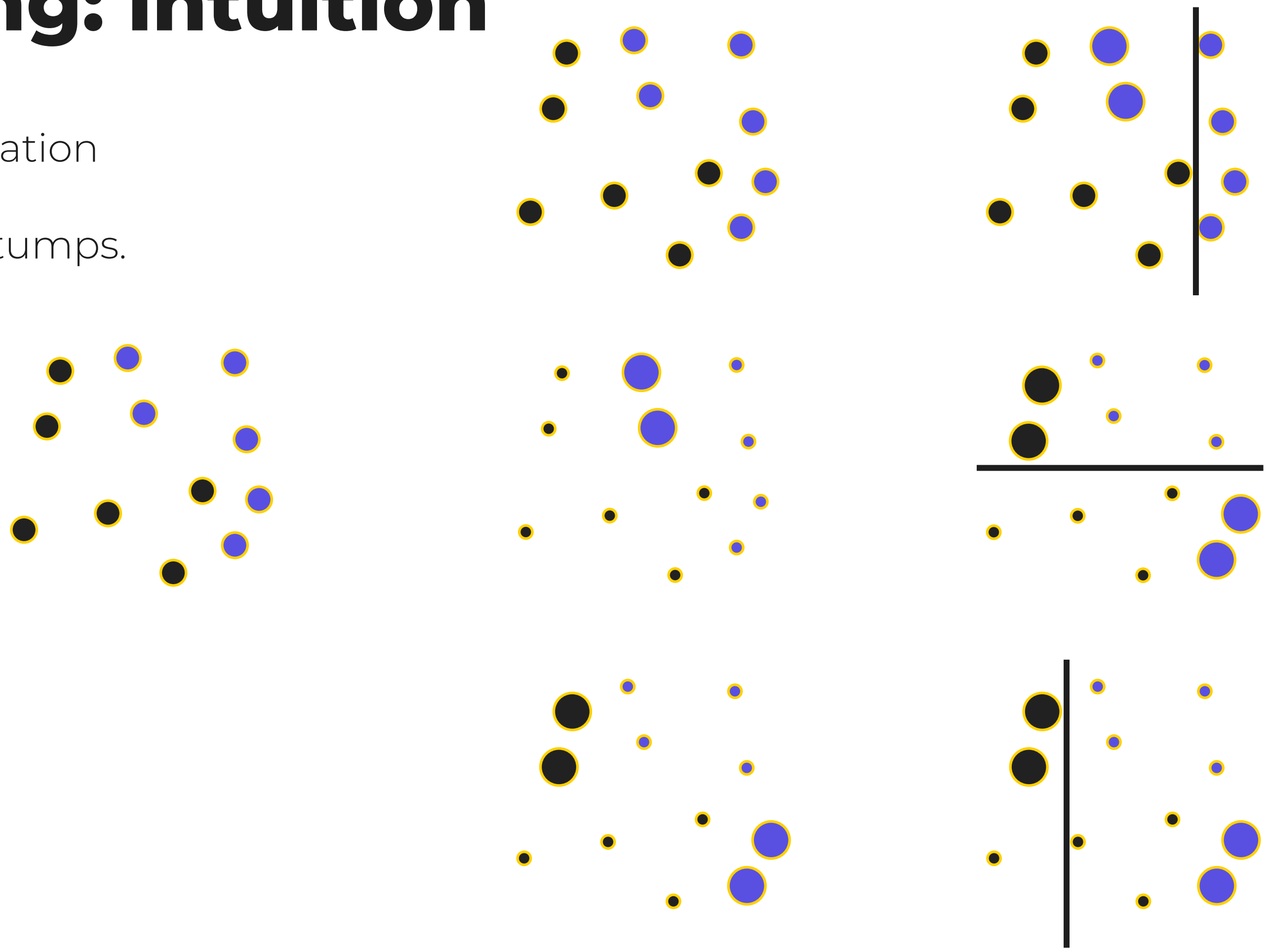
girafe
ai

01

Boosting: intuition

Binary classification

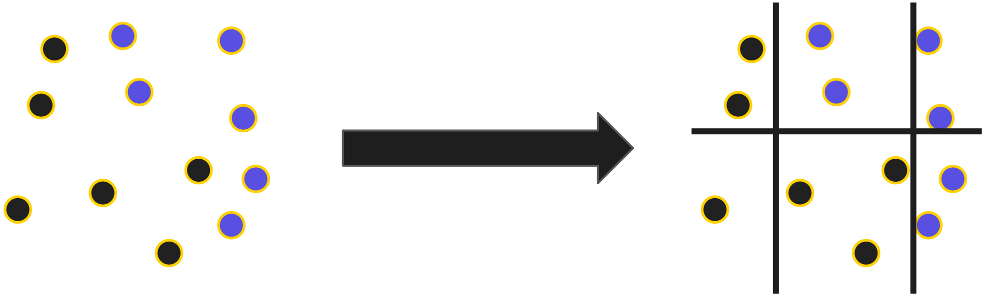
Use decision stumps.



Boosting: intuition

Binary classification

Use decision stumps.



Ensembles computation comparison

	Training	Inference
Bagging	parallel	parallel
Boosting	sequential	parallel

Gradient boosting theory

girafe
ai

02

Gradient boosting: theory

Denote dataset $\{(x_i, y_i)\}_{i=1,\dots,n}$, loss function $L(y, f)$.

Optimal model:

$$\hat{f}(x) = \underset{f(x)}{\arg\min}\ L(y, f(x)) = \underset{f(x)}{\arg\min}\ \mathbb{E}_{x,y}[L(y, f(x))]$$

Let it be from parametric family:

$$\hat{f}(x) = f(x, \hat{\theta}),$$

$$\hat{\theta} = \underset{\theta}{\arg\min}\ \mathbb{E}_{x,y}[L(y, f(x, \theta))]$$

Gradient boosting: theory

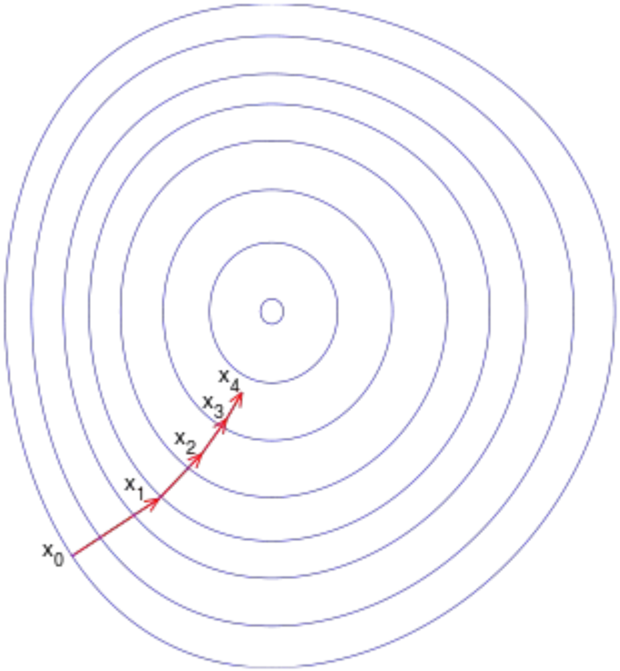
$$\hat{f}(x) = \sum_{i=0}^{t-1} \hat{f}_i(x),$$

$$(\rho_t, \theta_t) = \arg\min_{\rho,\theta} \mathbb{E}_{x,y}[L(y, \hat{f}(x) + \rho \cdot h(x, \theta))],$$

$$\hat{f}_t(x) = \rho_t \cdot h(x, \theta_t)$$

What if we could use gradient descent in space of our models?

Gradient boosting: theory



What if we could use gradient descent in space of our models?

Gradient boosting: theory

$$\hat{f}(x) = \sum_{i=0}^{t-1} \hat{f}_i(x),$$

$$r_{it} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=\hat{f}(x)}, \quad \text{for } i = 1, \ldots, n,$$

$$\theta_t = \arg\min_{\theta} \sum_{i=1}^{n} (r_{it} - h(x_i, \theta))^2,$$

$$\rho_t = \arg\min_{\rho} \sum_{i=1}^{n} L(y_i, \hat{f}(x_i) + \rho \cdot h(x_i, \theta_t))$$

Gradient boosting: theory

In linear regression case with MSE loss:

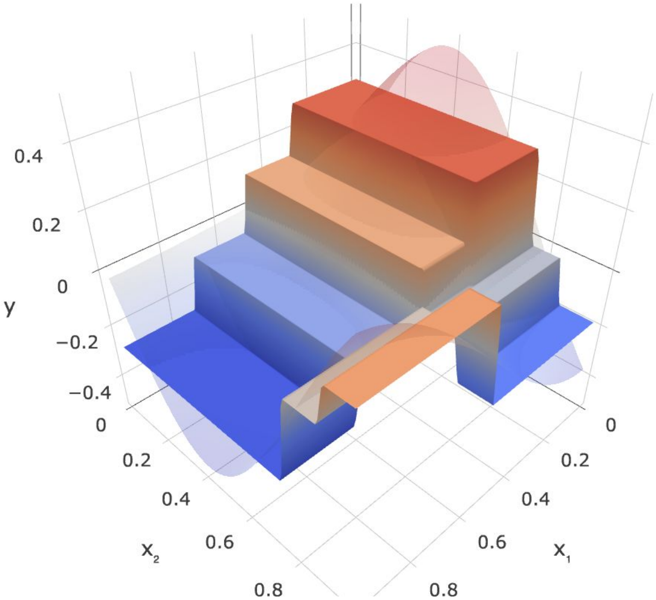
$$r_{it} = -\left[\frac{\partial L(y_i, f(x_i))}{\partial f(x_i)}\right]_{f(x)=\hat{f}(x)} = -2(\hat{y}_i - y_i) \propto \hat{y}_i - y_i$$

GB examples

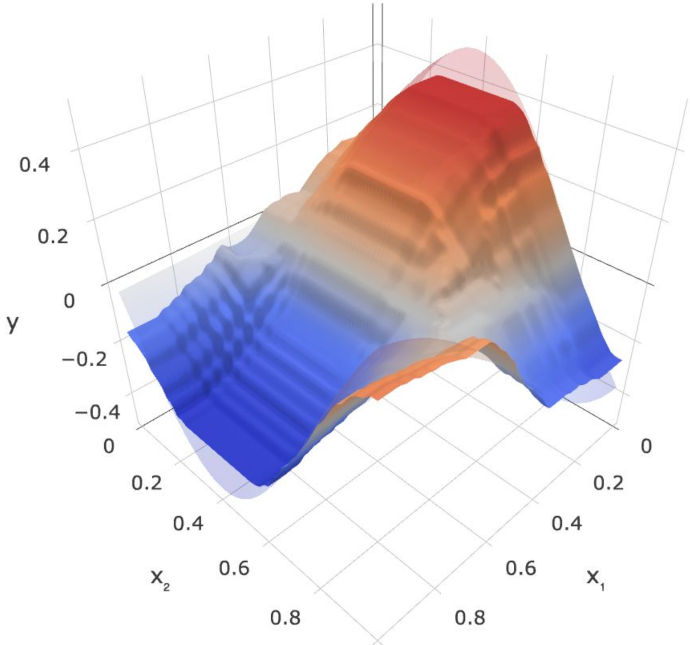
girafe ai

03

Beautiful demo



One tree



Boosting

http://arogozhnikov.github.io/2016/06/24/gradient_boosting_explained.html

Gradient boosting

What we need:

- Data
- Loss function and its gradient
- Family of algorithms (with constraints if necessary)
- Number of iterations M
- Initial value (GBM by Friedman): constant

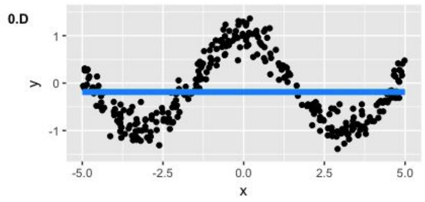
Gradient boosting: example

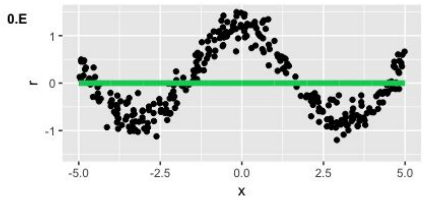
What we need:

- Data: toy dataset $y = cos(x) + \epsilon, \epsilon \sim \mathcal{N}(0, \frac{1}{5}), x \in [-5, 5]$
- Loss function: MSE
- Family of algorithms: decision trees with depth 2
- Number of iterations M = 3
- Initial value: just mean valu

https://habr.com/ru/company/ods/blog/327250/

Gradient boosting: example



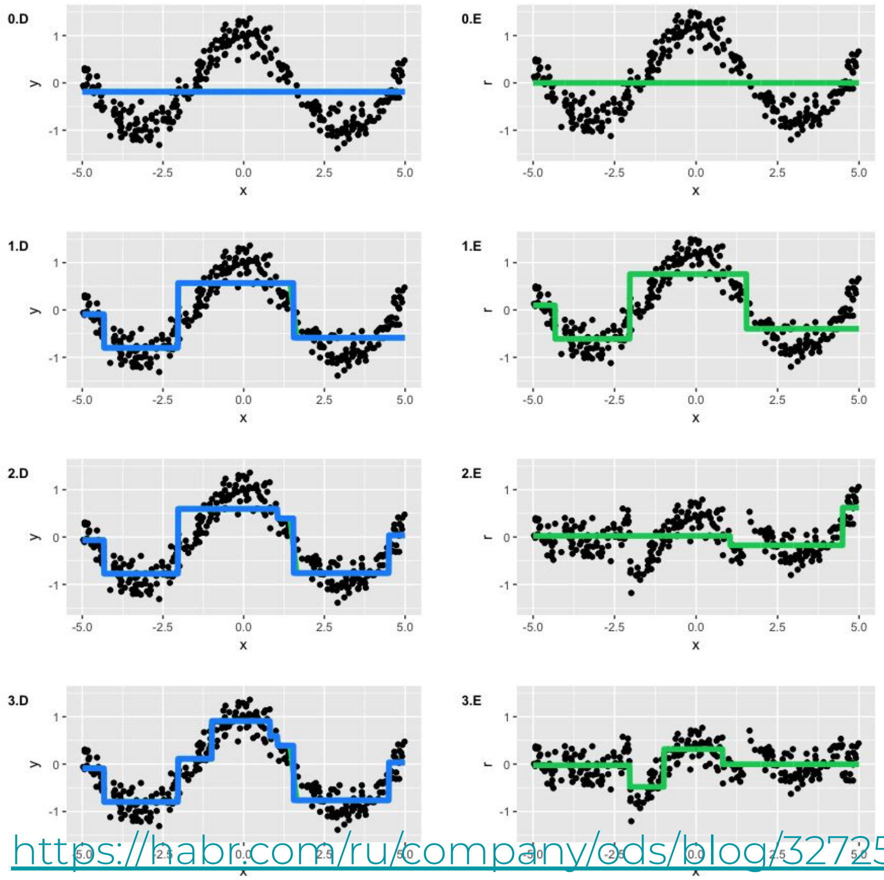


Left: full ensemble on each step.

Right: additional tree decisions.

https://habr.com/ru/company/ods/blog/327250/

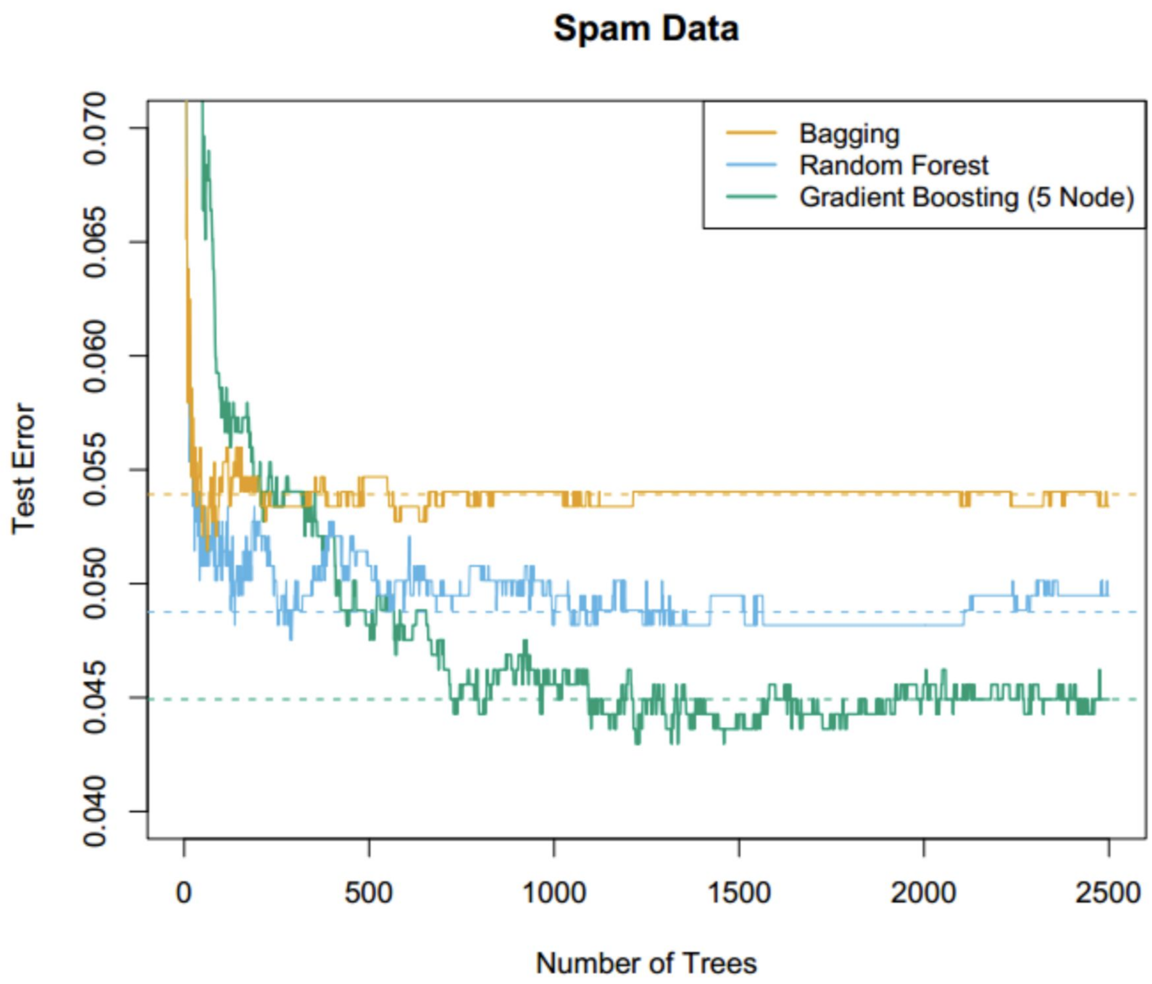
Gradient boosting: example

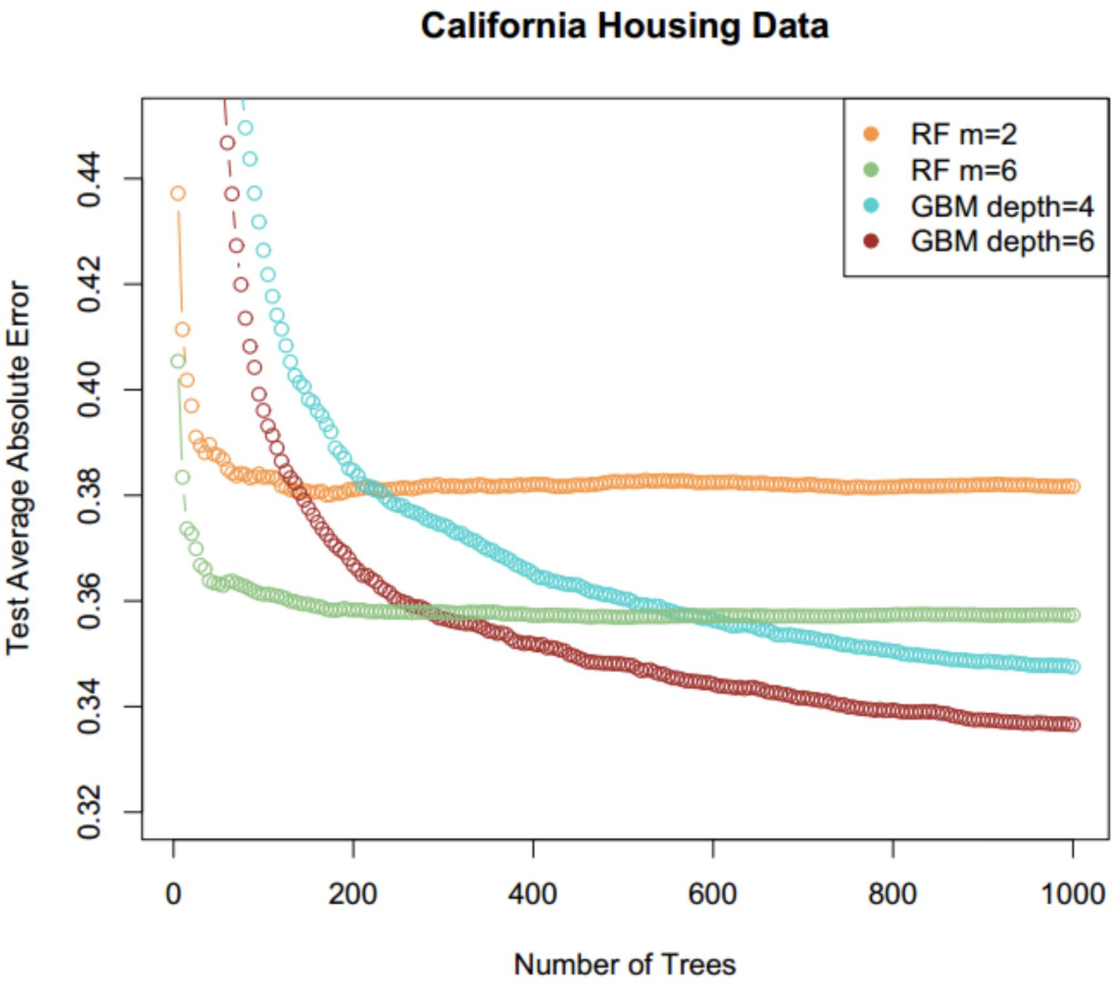


Left: full ensemble on each step.

Right: additional tree decisions

https://habr.com/ru/company/ods/blog/327250/


Spam Data
Bagging
Random Forest
Gradient Boosting (5 Node)
Test Error
0.040
0.045
0.050
0.055
0.060
0.065
0.070
0
500
1000
1500
2000
2500
Number of Trees


California Housing Data
RF m=2
RF m=6
GBM depth=4
GBM depth=6
Test Average Absolute Error
0.32
0.34
0.36
0.38
0.40
0.42
0.44
0
200
400
600
800
1000
Number of Trees

Parallelization

Which of the ensembling methods could be parallelized?

- Random Forest: parallel on the forest level (all trees are independent)
- Gradient boosting: parallel on one tree level

Libraries for GB

girafe
ai

04

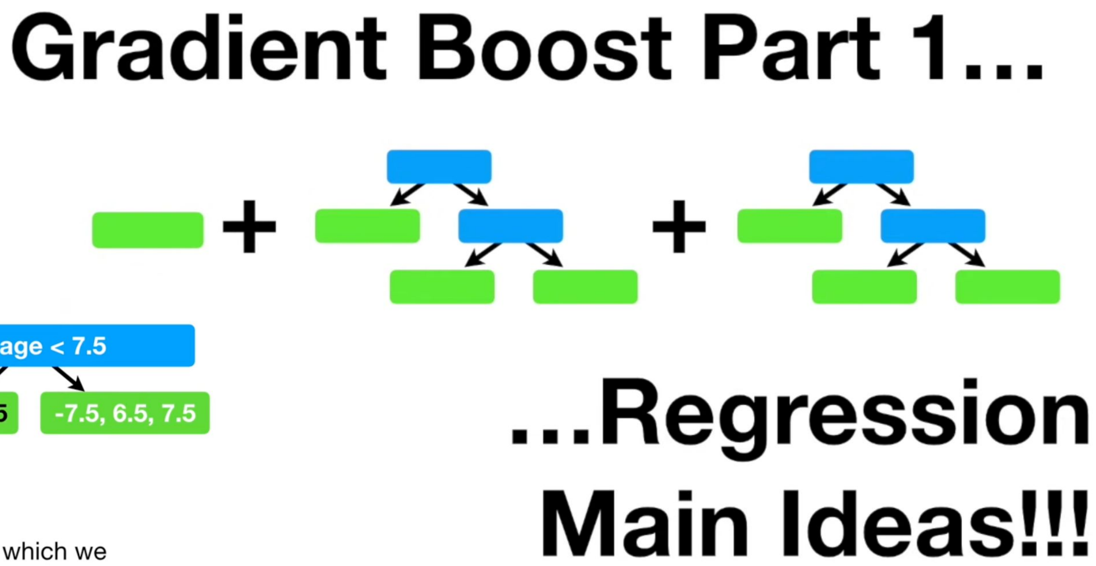
Main contemporary instruments

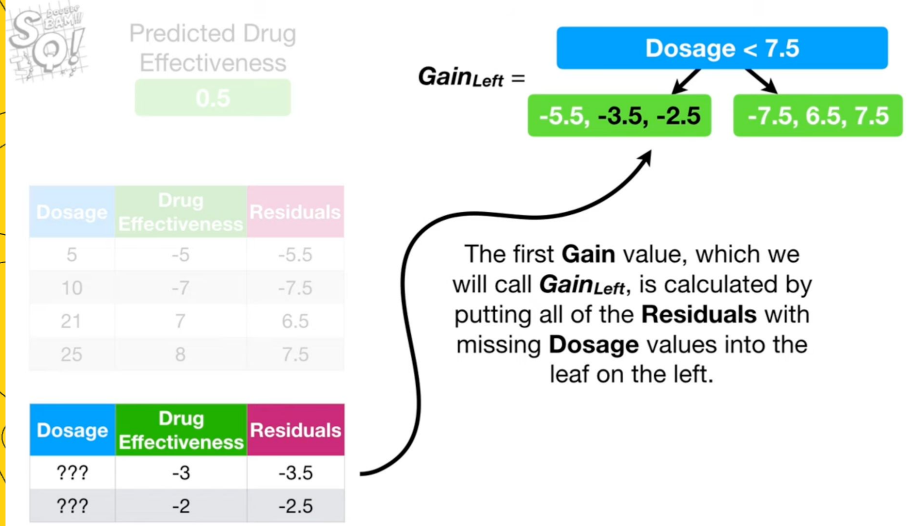
1. Catboost by Yandex
   https://catboost.ai/
   a. Explained by core developer for girafe-ai slides
2. LightGBM by Microsoft
   https://lightgbm.readthedocs.io/en/latest/index.html
3. XGboost by the community
   https://xgboost.readthedocs.io/en/stable/

Definitely not sklearn!

Boosting explained in verse!

1. [Boosting explained]()
2. [XGBoost expained]()





More on boosting

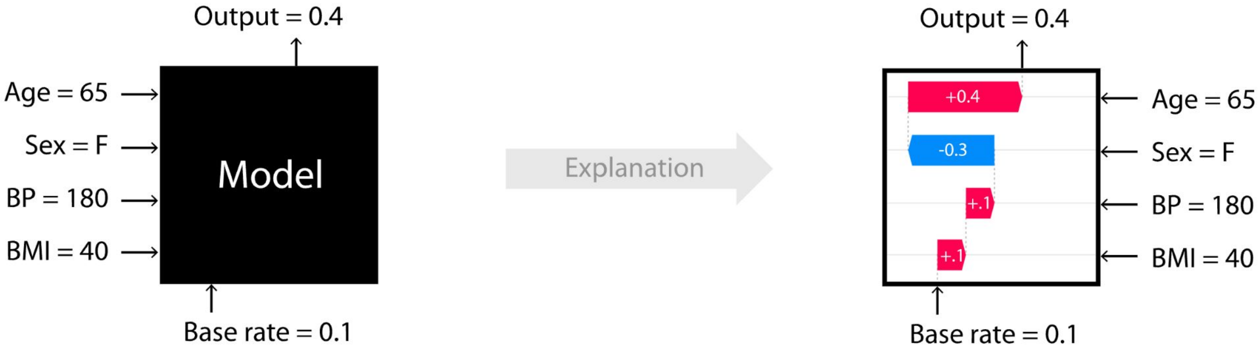
- https://habr.com/ru/companies/ods/articles/645887/
- https://neptune.ai/blog/when-to-choose-catboost-over-xgboost-or-lightgbm
- https://towardsdatascience.com/catboost-vs-lightgbm-vs-xgboost-c80f40662924
- https://www.springboard.com/blog/data-science/xgboost-random-forest-catboost-lightgbm/
- https://towardsdatascience.com/performance-comparison-catboost-vs-xgboost-and-catboost-vs-lightgbm-886c1c96db64

Peak at feature importances!

girafe
ai

05

Shap values



https://github.com/shap/shap

Hyperparameter optimization

girafe
ai

06

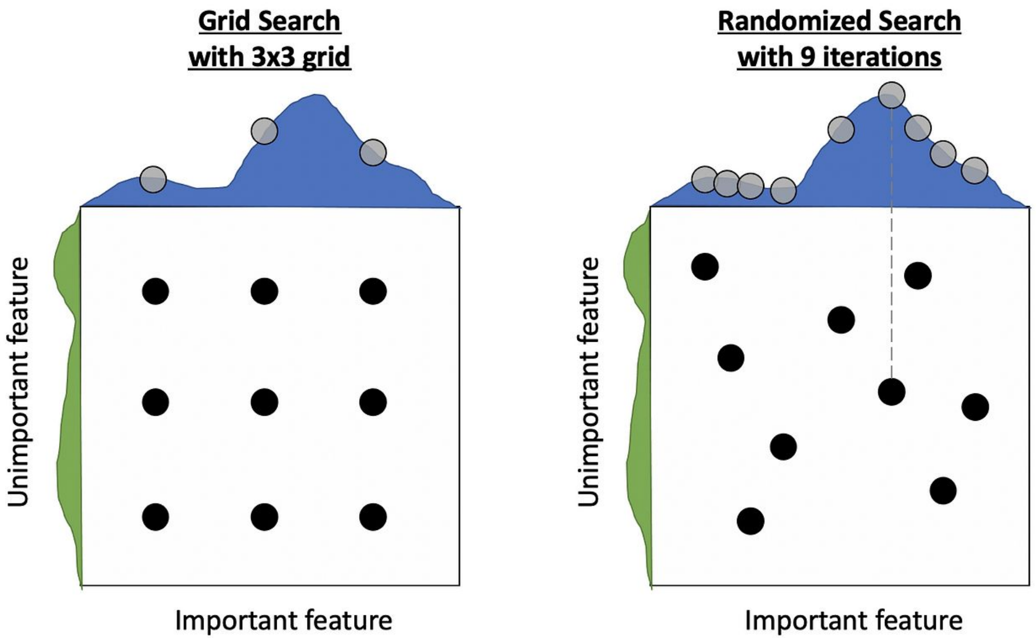
Optimization note

In optimization theory methods are associated with the order of derivatives they use. Main ones are:

- first order optimization
  - use gradient of optimized function e.g. SGD which we discussed
- second order optimization
  - use Hessian matrix. They are qute slow
- zero order
  - don’t need gradient, only values of optimized function
  - that’s what we are interested in today

0-order optimization approaches

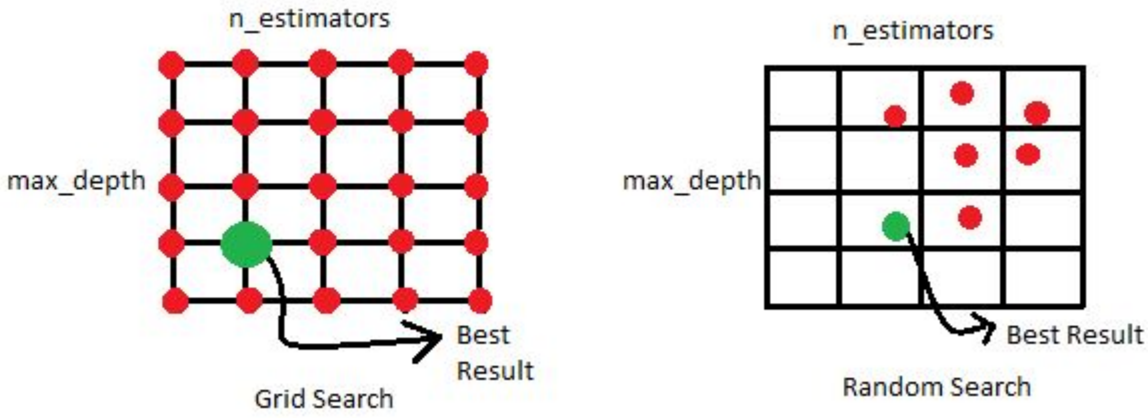
1. Manual trials
2. Grid search
3. Random search



In theory

0-order optimization approaches

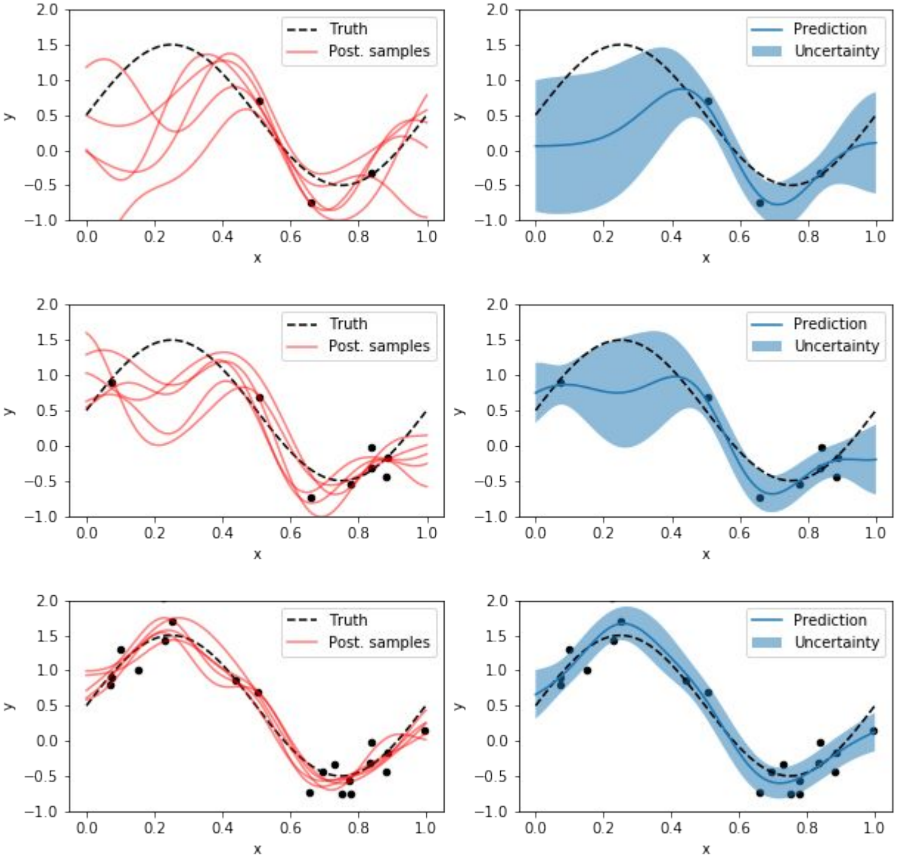
1. Manual trials
2. Grid search
3. Random search



In practice

0-order optimization approaches

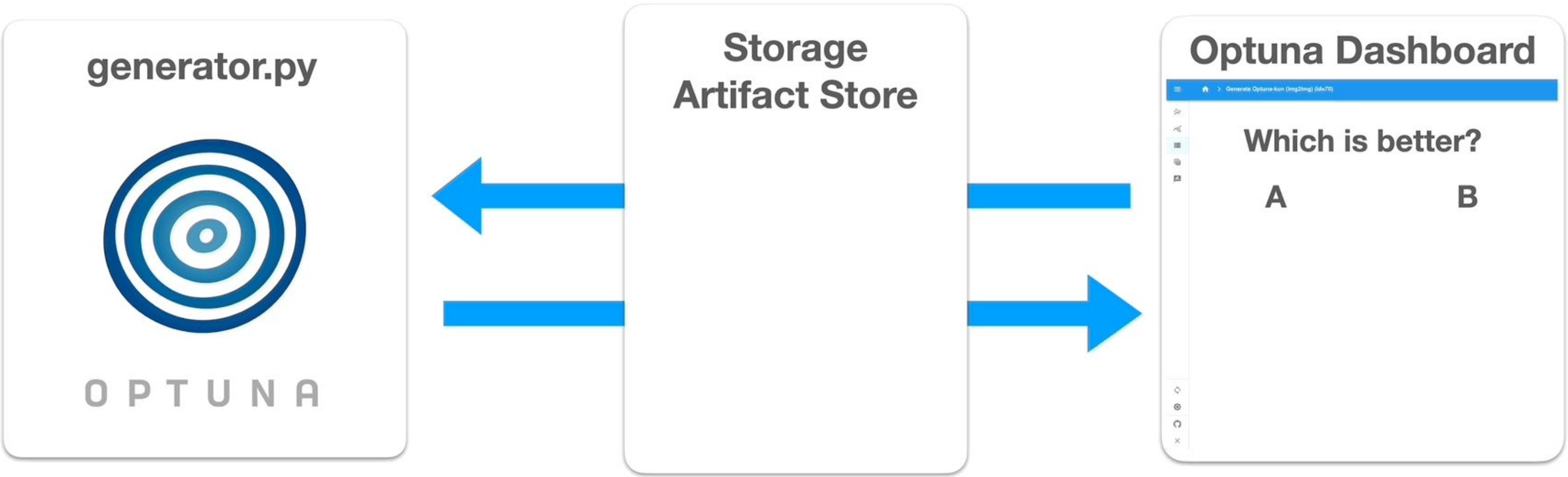
1. Manual trials
2. Grid search
3. Random search
4. Bayesian methods
5. Evolutionary methods



Main libraries

- [Hyperopt]
- [Optuna]

Black box or 0 order optimization



https://optuna.org/ and http://hyperopt.github.io/hyperopt/

Revise

1. Intuitions
2. Gradient boosting theory
3. Examples
4. Libraries
5. Feature importances
6. Hyperparameter optimization

Thanks for attention!

Questions?

girafe
ai