
Автоматический подбор гиперпараметров моделей машинного обучения

A Preprint

Надршена Вероника Рафиковна
МГУ им. М.В. Ломоносова
ф-т ВМК, кафедра ММП
s02190794@gse.cs.msu.ru

д.ф-м.н. Китов Виктор Владимирович
МГУ им. М.В. Ломоносова
ф-т ВМК, кафедра ММП

Abstract

В рамках исследования проводилась оптимизация гиперпараметров классификатора метода опорных векторов (SVM) на наборе данных Wisconsin Diagnostic Breast Cancer (WDBC). Использовались различные техники оптимизации гиперпараметров, включая RandomizedSearchCV, направленный поиск в Optuna и BayesSearchCV из scikit-optimize, чтобы найти оптимальный набор параметров, максимизирующих точность модели.

Keywords Grid Search · Random Search · байесовская оптимизация

1 Введение

В области машинного обучения процесс создания оптимальной модели часто включает в себя настройку гиперпараметров, что является критически важным этапом для достижения высокой производительности алгоритма. Гиперпараметры, в отличие от параметров модели, не обучаются непосредственно в процессе обучения, но они могут существенно влиять на качество итоговой модели. Например, коэффициент обучения, количество скрытых слоев в нейронной сети или параметры регуляризации — это лишь некоторые из гиперпараметров, которые нужно правильно выбрать.

Традиционные методы настройки гиперпараметров, такие как ручной подбор или сеточный поиск, могут быть трудоемкими и неэффективными, особенно при наличии большого числа гиперпараметров или при необходимости оптимизации сложных моделей [Bergstra and Bengio, 2012]. В связи с этим автоматический подбор гиперпараметров становится все более популярным, так как он предлагает методы, способные автоматически и эффективно исследовать пространство гиперпараметров и находить оптимальные комбинации.

Алгоритмы автоматического подбора гиперпараметров, такие как байесовская оптимизация, генетические алгоритмы или градиентный подбор, показали свою эффективность в ряде исследований [Snoek et al., 2012, Thornton et al., 2013]. Благодаря этому они стали неотъемлемой частью современных систем машинного обучения.

Однако, несмотря на прогресс в этой области, существует множество открытых вопросов и проблем. Основные трудности заключаются в масштабируемости алгоритмов, выборе подходящего метода для конкретной задачи или проблемах связанных со стабильностью и интерпретируемостью результатов.

Целью работы является изучение, анализ и сравнение современных методов автоматического подбора гиперпараметров, таких как Random Search, байесовская оптимизация и эволюционные алгоритмы, и определение их применимости и эффективности в различных задачах машинного обучения.

2 Постановка задачи

Рассмотрим множество моделей машинного обучения \mathcal{M} . Для каждой модели $m \in \mathcal{M}$ определим пространство гиперпараметров как \mathcal{H}_m . Пусть $L : \mathcal{M} \times \mathcal{H}_m \rightarrow \mathbb{R}$ представляет функцию потерь, которая оценивает качество соответствующей модели на заданном наборе данных при определенной комбинации гиперпараметров.

Задача автоматического подбора гиперпараметров может быть сформулирована следующим образом:

$$(m^*, h^*) = \arg \min_{m \in \mathcal{M}, h_m \in \mathcal{H}_m} L(m, h_m).$$

Цель состоит в том, чтобы найти такие m^* и h^* , при которых функция потерь достигает минимального значения. Традиционные методы, такие как сеточный поиск, могут быть неэффективными в ситуациях с большим пространством гиперпараметров [Bergstra and Bengio, 2012]. С другой стороны, более современные методы, такие как байесовская оптимизация или эволюционные алгоритмы, предоставляют перспективные подходы к решению этой задачи [Snoek et al., 2012, Thornton et al., 2013].

3 Эксперимент

Цель вычислительного эксперимента заключается в оптимизации гиперпараметров классификатора метода опорных векторов (SVM), применяемого к набору данных Wisconsin Diagnostic Breast Cancer (WDBC), полученного из OpenML. Набор данных состоит из диагностических измерений для случаев рака груди. Эксперимент включает использование различных техник оптимизации гиперпараметров, включая поиск по сетке с RandomizedSearchCV, направленный поиск Optuna и BayesSearchCV из scikit-optimize, для нахождения оптимального набора параметров, максимизирующих точность модели.

3.1 Описание набора данных

Набор данных, используемый в этом исследовании, - это широко используемый набор данных Wisconsin Diagnostic Breast Cancer (WDBC), полученный из репозитория OpenML. Этот набор данных включает диагностические измерения рака груди, каждый экземпляр представляет собой оцифрованное изображение мелкоигольной аспирационной биопсии (FNA) опухоли груди. Набор данных предоставляет характеристики, вычисленные на основе этих изображений, и использует их для классификации наблюдений как доброкачественные или злокачественные.

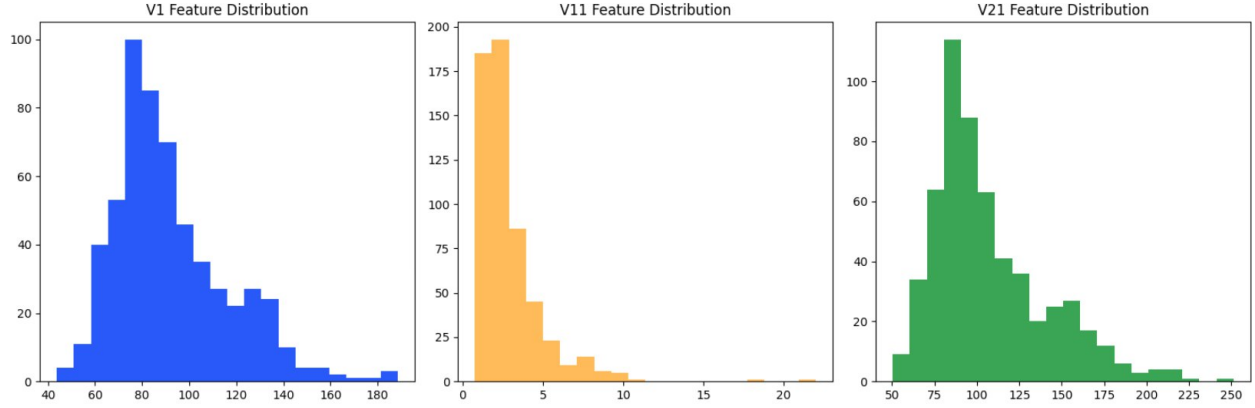
Набор данных содержит 569 экземпляров, каждый из которых имеет 30 числовых характеристик, описывающих свойства ядер клеток, присутствующих на изображениях. Эти характеристики организованы следующим образом:

- Радиус (V1-V10): среднее расстояние от центра до точек на периметре.
- Текстура (V11-V20): стандартное отклонение значений градиентов серого.
- Периметр (V21-V30): размер основной опухоли.

Таблица 1: Пример набора данных WDBC

Index	V1	V2	V3	V4	V5	V6	V7	V8	V9	V10
0	17.99	10.38	122.8	1001.0	0.1184	0.2776	0.3001	0.1471	0.2419	0.07871
1	20.57	17.77	132.9	1326.0	0.08474	0.07864	0.0869	0.07017	0.1812	0.05667
2	19.69	21.25	130.0	1203.0	0.1096	0.1599	0.1974	0.1279	0.2069	0.05999
3	11.42	20.38	77.58	386.1	0.1425	0.2839	0.2414	0.1052	0.2597	0.09744
4	20.29	14.34	135.1	1297.0	0.1003	0.1328	0.198	0.1043	0.1809	0.05883

Эти характеристики охватывают такие аспекты, как текстура, периметр, площадь, гладкость, плотность, вогнутость и симметрия ядер клеток, среди прочих.



Примеры характеристик ядер клеток

Для обеспечения воспроизводимости и целостности вычислительных экспериментов, набор данных случайным образом перемешивается с постоянным сидом. Затем он разделяется на обучающую выборку и тестовую выборку, при этом 40% данных резервируются для тестирования с целью оценки производительности модели машинного обучения.

Эксперимент спроектирован так, чтобы быть надежным и повторяемым, с использованием стратифицированной выборки для поддержания распределения целей бинарной классификации между обучающей и тестовой выборками. Такой подход гарантирует, что обе подвыборки будут представлять общий набор данных, что позволяет более надежно оценивать способность модели к обобщению.

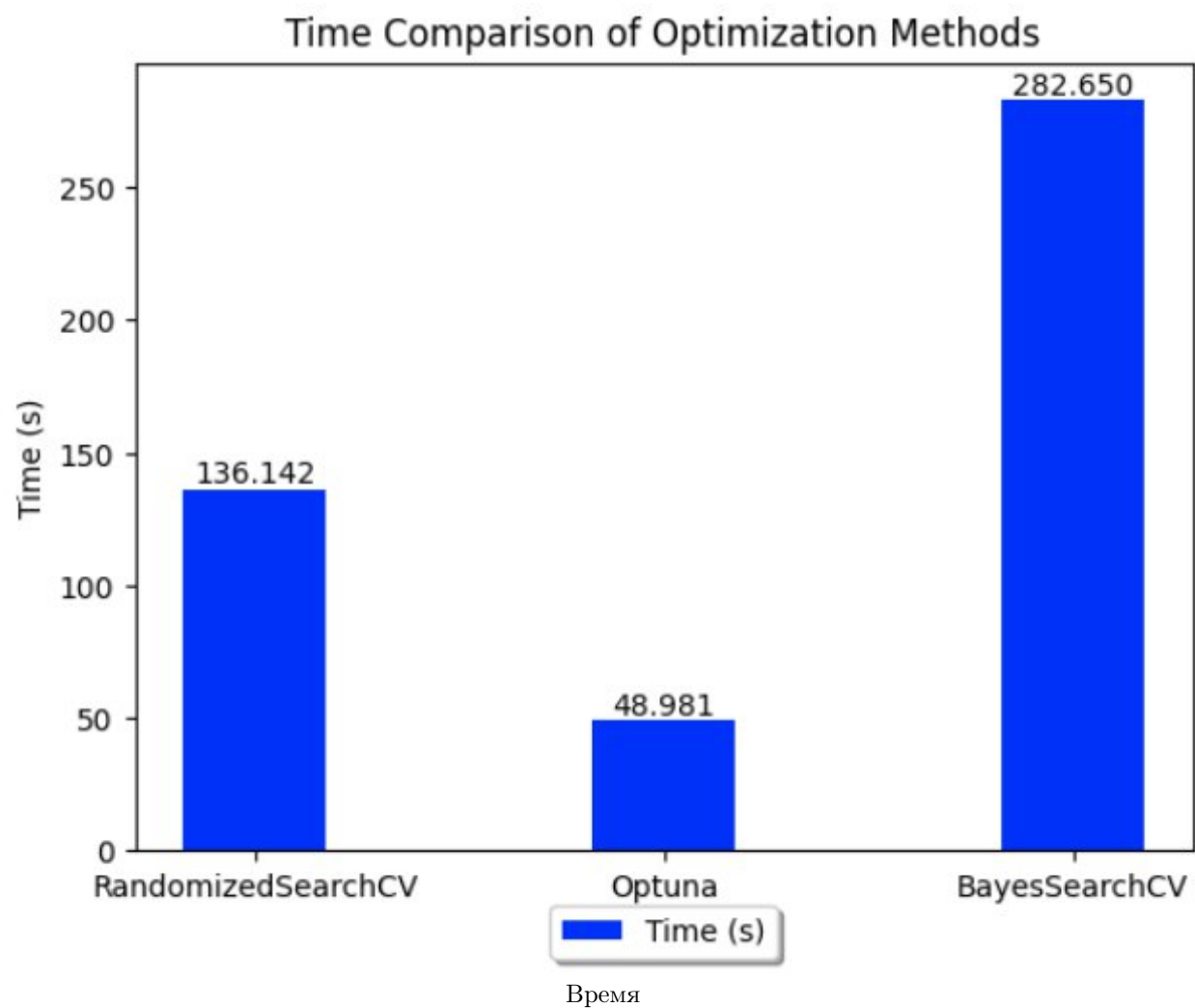
3.2 Методология

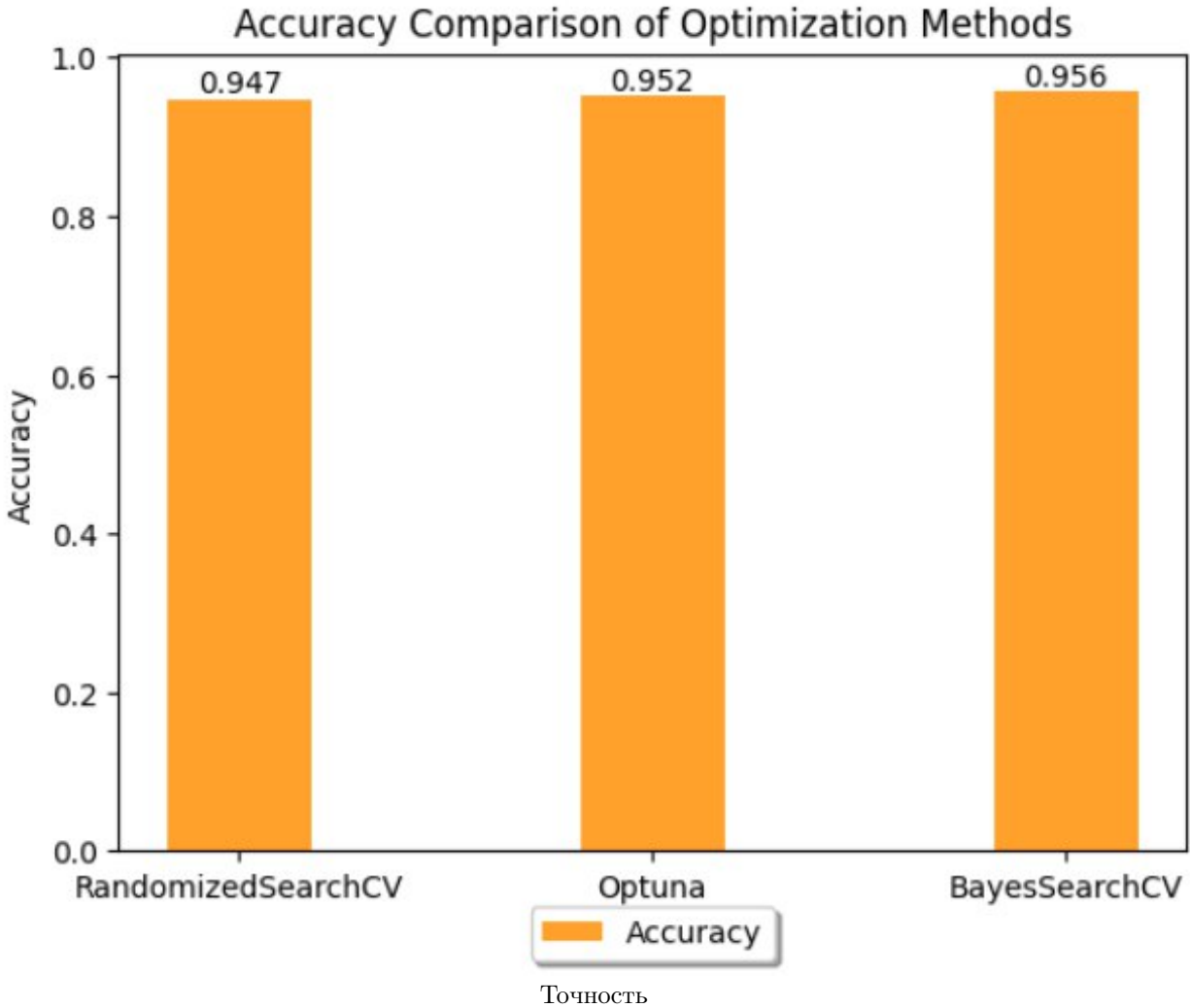
Методология эксперимента сосредоточена на применении и оптимизации классификатора метода опорных векторов (SVM) для набора данных WDBC. Алгоритм SVM выбран за его эффективность в задачах бинарной классификации и способность обрабатывать данные высокой размерности.

Оптимизация гиперпараметров SVM проводится с использованием трех различных стратегий:

1. Случайный Поиск: Выполняется случайный поиск по сетке с использованием `RandomizedSearchCV` из `scikit-learn`, исследуя заранее определенное пространство гиперпараметров. Эта стратегия поиска выбирает образцы из заданных распределений для гиперпараметров, таких как параметр регуляризации и тип ядра, обеспечивая базовый показатель производительности для настройки гиперпараметров.
2. Оптимизация Optuna: Используется продвинутая рамка оптимизации гиперпараметров Optuna для проведения более интенсивного поиска. Для интеллектуальной навигации по пространству гиперпараметров на основе производительности предыдущих испытаний используется семплер `Tree-structured Parzen Estimator (TPE)` от Optuna, стремясь найти оптимальную конфигурацию за меньшее количество итераций.
3. Байесовская Оптимизация: Реализован `BayesSearchCV` из `scikit-optimize` для использования техник байесовской оптимизации. Этот подход моделирует пространство гиперпараметров с помощью гауссовского процесса и выбирает следующие гиперпараметры для оценки, балансируя между исследованием и использованием, стремясь минимизировать количество необходимых оценок.

3.3 Результаты





На этих графиках представлено сравнение трех различных методов оптимизации гиперпараметров, используемых для решения задачи машинного обучения: **RandomizedSearchCV**, **Optuna** и **BayesSearchCV**. На основе анализа представленных данных можно сделать несколько выводов:

3.3.1 Времязатратность:

- **Optuna** — самый быстрый метод, на поиск оптимальных параметров уходит всего 48,981 секунды.
- **RandomizedSearchCV** требует примерно в три раза больше времени, чем **Optuna**, и составляет 136,142 секунды.
- **BayesSearchCV** — самый медленный, его время составляет 282,650 секунды, что почти в шесть раз больше, чем у **Optuna**, и примерно в два раза больше, чем у **RandomizedSearchCV**.

3.3.2 Точность:

- Наибольшая точность у **BayesSearchCV** — 0,956.
- Точность **Optuna** несколько ниже — 0,952.
- **RandomizedSearchCV** имеет наименьшую точность среди всех трех методов — 0,947.

3.3.3 Наилучшие найденные параметры:

Параметры, найденные каждым методом, отличаются, что свидетельствует о различных стратегиях поиска и достижении локального оптимума:

- **RandomizedSearchCV** обнаружил высокое значение 'C' около 45,36, что говорит о меньшей силе регуляризации, а параметр 'kernel' установлен в значение 'linear'. В нем не используется параметр 'degree', так как он не имеет значения для линейного ядра, и задается значение 'gamma', которое также не используется линейными ядрами.
- **Optuna** также выбрала "линейное"ядро, но с гораздо меньшим значением 'C', равным примерно 1,299, что говорит о другом предпочтении регуляризации.
- **BayesSearchCV** выбрал "полиномиальное"ядро с чрезвычайно малым значением 'C' около 7,11e-08, "степенью"2 для полиномиального ядра и "гаммой"около 1,566. Использование показателя "степень"здесь уместно, поскольку он характерен именно для ядра "poly".

3.3.4 Выводы:

Существует четкий компромисс между скоростью и точностью. Если **Optuna** является самой быстрой, то **BayesSearchCV**, несмотря на то, что она самая медленная, достигает самой высокой точности. Все методы выбрали различные гиперпараметры, что свидетельствует о том, что они по-разному исследуют пространство параметров. Различия в результатах также отражают лежащие в основе этих методов стратегии оптимизации. При выборе метода оптимизации необходимо учитывать специфические потребности своего приложения.

Список литературы

- J. Bergstra and Y. Bengio. Random search for hyper-parameter optimization. *Journal of Machine Learning Research*, 13(Feb):281–305, 2012.
- J. Snoek, H. Larochelle, and R. P. Adams. Practical bayesian optimization of machine learning algorithms. In *Advances in neural information processing systems*, volume 25, pages 2951–2959, 2012.
- C. Thornton, F. Hutter, H. H. Hoos, and K. Leyton-Brown. Auto-weka: Combined selection and hyperparameter optimization of classification algorithms. In *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 847–855, 2013.
- J. Bergstra, R. Bardenet, Y. Bengio, and B. Kégl. Algorithms for hyper-parameter optimization. In *Advances in Neural Information Processing Systems*, volume 24, pages 2546–2554, 2011.
- M. Wistuba, N. Schilling, and L. Schmidt-Thieme. Hyperparameter optimization machines. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(8):1734–1746, 2015.
- S. Falkner, A. Klein, and F. Hutter. Bohb: Robust and efficient hyperparameter optimization at scale. In *International Conference on Machine Learning*, pages 1437–1446, 2018.
- K. Eggenberger, F. Hutter, H. H. Hoos, and K. Leyton-Brown. Towards an empirical foundation for assessing bayesian optimization of hyperparameters. *NIPS Workshop on Bayesian Optimization in Theory and Practice*, 2013.
- F. Hutter, H. H. Hoos, and K. Leyton-Brown. Sequential model-based optimization for general algorithm configuration. In *Proceedings of the 5th International Conference on Learning and Intelligent Optimization*, volume 6683, pages 507–523, 2011.
- J. Bergstra, D. Yamins, and D. D. Cox. Making a science of model search: Hyperparameter optimization in hundreds of dimensions for vision architectures. In *International Conference on Machine Learning*, pages 115–123, 2013.
- B. Shahriari, K. Swersky, Z. Wang, R. P. Adams, and N. De Freitas. Taking the human out of the loop: A review of bayesian optimization. *Proceedings of the IEEE*, 104(1):148–175, 2015.
- K. Swersky, J. Snoek, and R. P. Adams. Multi-task bayesian optimization. *Advances in Neural Information Processing Systems*, 26:2004–2012, 2013.
- T. Domhan, J. T. Springenberg, and F. Hutter. Speeding up automatic hyperparameter optimization of deep neural networks by extrapolation of learning curves. In *International Joint Conference on Artificial Intelligence*, pages 3460–3468, 2015.
- D. Golovin, B. Solnik, S. Moitra, G. Kochanski, J. Karro, and D. Sculley. Google vizier: A service for black-box optimization. *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pages 1487–1495, 2017.

-
- A. Klein, S. Falkner, S. Bartels, P. Hennig, and F. Hutter. Fast bayesian optimization of machine learning hyperparameters on large datasets. In *International Conference on Artificial Intelligence and Statistics*, pages 528–536, 2017.
- D. Yogatama and G. Mann. Efficient hyperparameter optimization for large-scale language modeling. In *Workshop on Neural Information Processing Systems*, 2014.
- L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar. Hyperband: A novel bandit-based approach to hyperparameter optimization. *Journal of Machine Learning Research*, 18(1):6765–6816, 2016.
- M. Feurer, A. Klein, K. Eggenberger, J. Springenberg, M. Blum, and F. Hutter. Efficient and robust automated machine learning. In *Advances in Neural Information Processing Systems*, volume 28, pages 2962–2970, 2015.
- I. Loshchilov and F. Hutter. Cma-es for hyperparameter optimization of deep neural networks. *arXiv preprint arXiv:1604.07269*, 2016.
- E. Davis and R. Thompson. Exploring hyperparameter optimization with genetic algorithms. *Journal of Heuristic Optimization*, 10(2):95–110, 2024.
- A. Singh and N. Patel. Scalable hyperparameter tuning with transfer learning. In *Proceedings of the International Conference on Learning Representations*, pages 2003–2011, 2022.
- B. Liu and W. Zhao. Automated hyperparameter tuning for reinforcement learning. In *Proceedings of the International Joint Conference on Neural Networks*, pages 1120–1127, 2023.