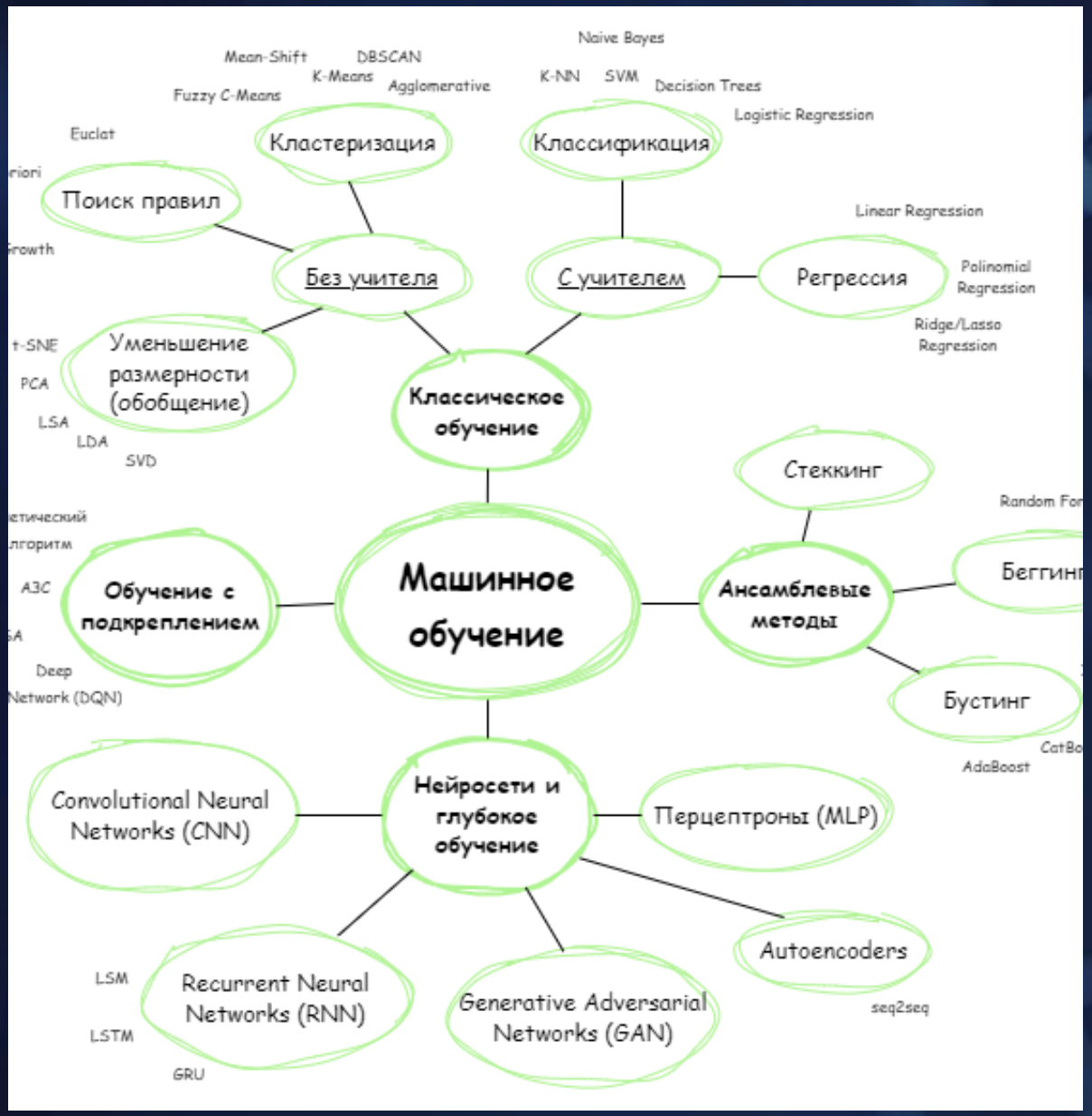
Ансамблевые методы машинного обучения

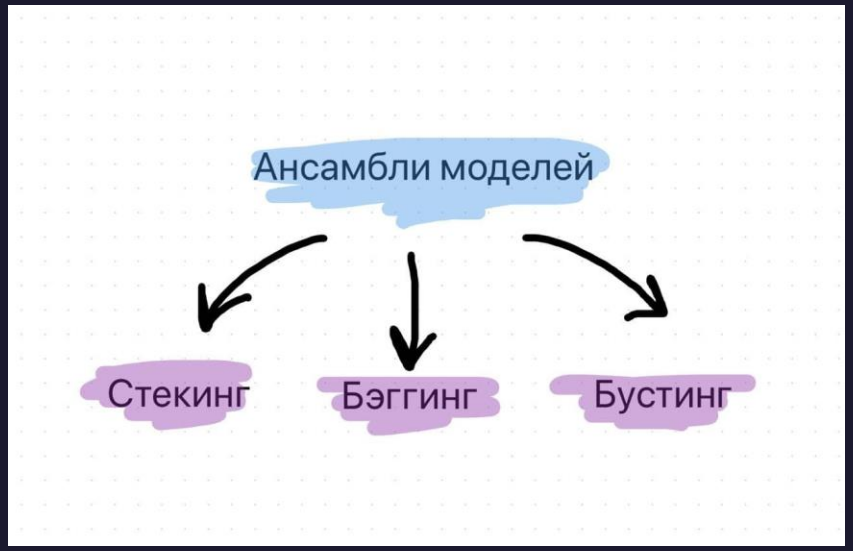


Что такое ансамбль?

Метод машинного обучения, где несколько моделей обучаются для решения одной и той же проблемы и объединяются для получения лучших результатов называется ансамблевым методом. Основная предпосылка заключается в том, что результат работы нескольких моделей будет более точен, чем результат только одной модели.

Когда говорится об ансамблях, то вводится понятие слабого ученика (обычные модели вроде линейной регрессии или дерева решений). Множество слабых учеников являются строительными блоками для более сложных моделей. Объединение слабых учеников для улучшения качества модели, уменьшения смещения или разброса, называется сильным учеником.

Виды ансамблевых методов



Стекинг. Могут рассматриваться разнородные отдельно взятые модели. Существует мета-модель, которой на вход подаются базовые модели, а выходом является итоговый прогноз.

Бэггинг. Рассматриваются однородные модели, которые обучаются независимо и параллельно, а затем их результаты просто усредняются. Ярким представителем данного метода является случайный лес.

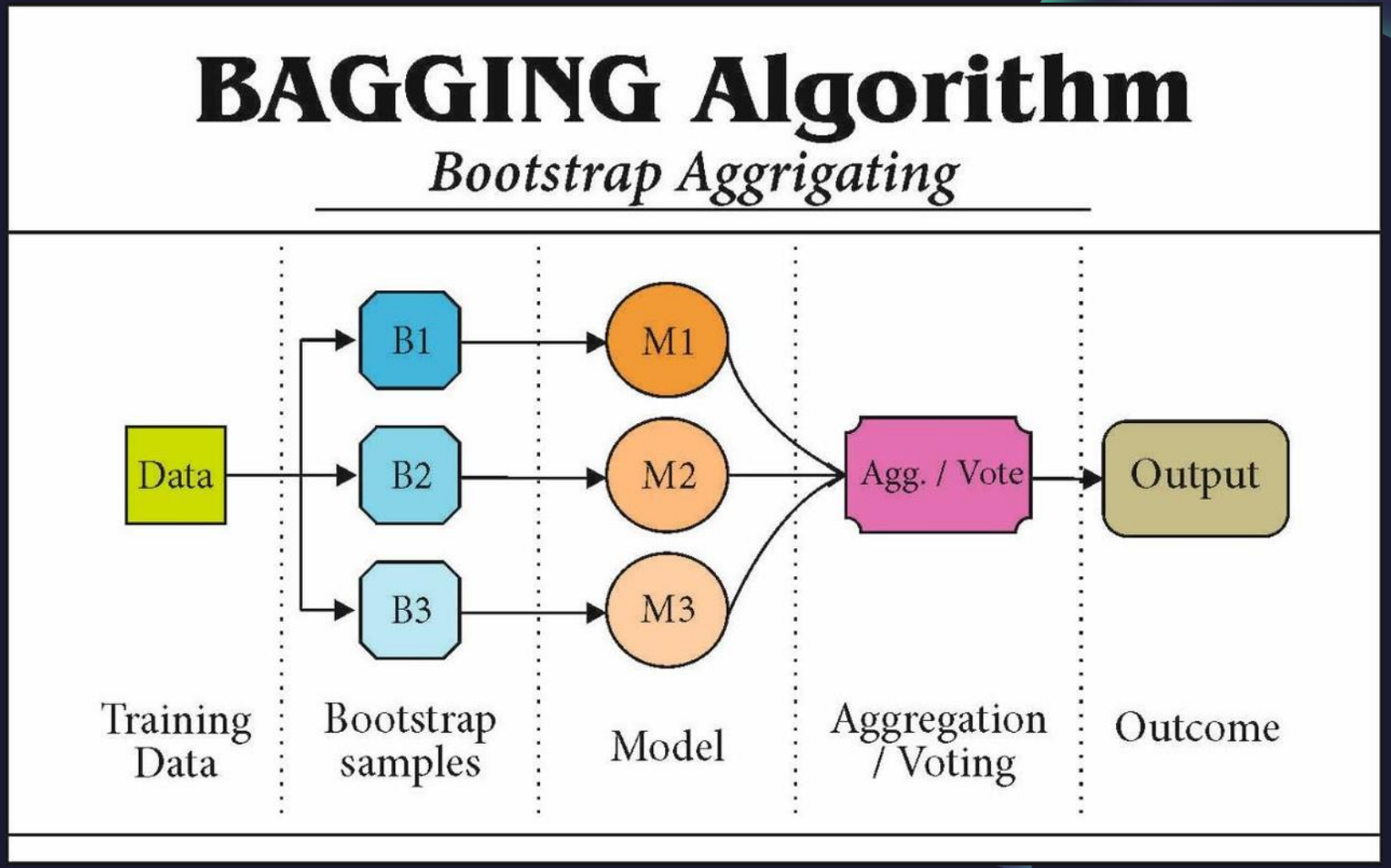
Бустинг. Рассматриваются однородные модели, которые обучаются последовательно, причем последующая модель должна исправлять ошибки предыдущей. Конечно, в качестве примера здесь сразу приходит на ум градиентный бустинг.

Беггинг

Беггинг

Бэггинг (bootstrap aggregating) в машинном обучении — это ансамблевый метод, который объединяет прогнозы из нескольких методов обучения вместе, чтобы предсказывать более точно, чем любая отдельная модель.

Идея бэгинга заключается в том, что каждый базовый алгоритм обучается на случайном подмножестве обучающей выборки. В этом случае, даже используя одну модель алгоритмов, получаются различные базовые алгоритмы.



Беггинг

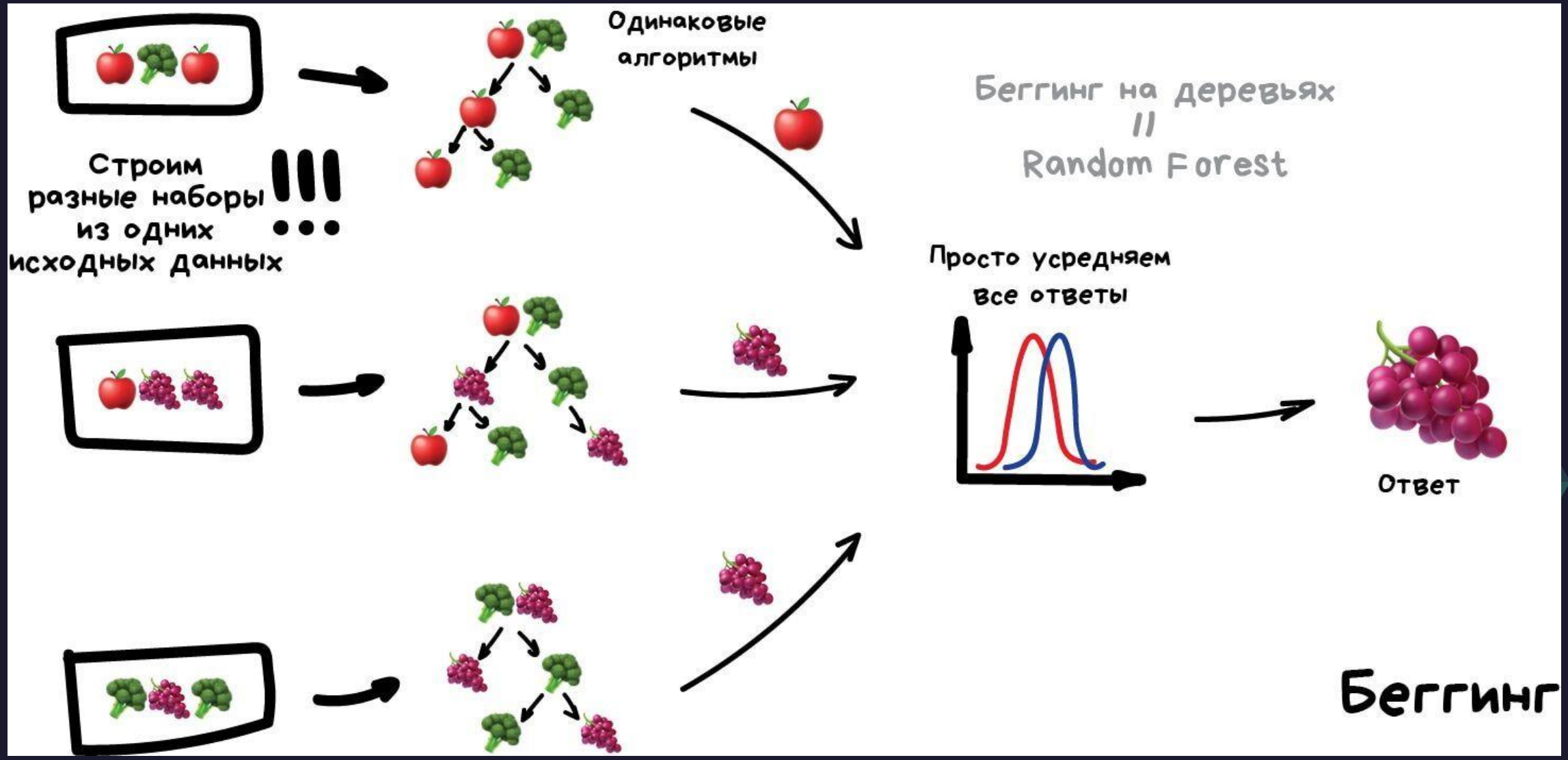
Плюсы Random Forest:

- Снижает переобучение и улучшает точность.
- Может использоваться для решения задач классификации и регрессии.
- Хорошо работает с категориальными и непрерывными значениями.
- Автоматически обрабатывает пропущенные значения.
- Нормализация данных не требуется.
- Может обрабатывать большие объемы данных с высокой размерностью и определять наиболее значимые переменные.
- Имеет методы для балансировки ошибок в несбалансированных данных.
- Позволяет использовать внебазовые выборки для тестирования, что экономит время и ресурсы.

Минусы Random Forest:

- Не дает точных непрерывных прогнозов в задачах регрессии.
- Может казаться "черным ящиком" для статистических моделей, так как управление моделью практически отсутствует.
- Требует много вычислительной мощности и ресурсов для построения множества деревьев решений.
- Требует много времени для обучения модели.

Беггинг


Одинаковые алгоритмы
Строим разные наборы из одних исходных данных
Беггинг на деревьях
//
Random Forest
Просто усредняем все ответы
Ответ
Беггинг

Беггинг

Основная цель: Уменьшение дисперсии (variance) модели. Бэггинг отлично справляется с переобучением (overfitting), особенно для алгоритмов с высокой дисперсией, таких как деревья решений.

Бэггинг позволяет снизить дисперсию (разброс, variance) результатов обучаемых моделей. Это возможно если модели ошибаются независимо друг от друга: одна ошибется в плюс, а другая в минус - в среднем получится то, что надо. Но это же условие является и ограничением беггинга - модели должны быть независимы (статистически) и ошибаться по-разному. Если же все модели имеют общую ошибку, то беггинг никак не поможет.

Самый известный пример: Случайный лес (Random Forest)

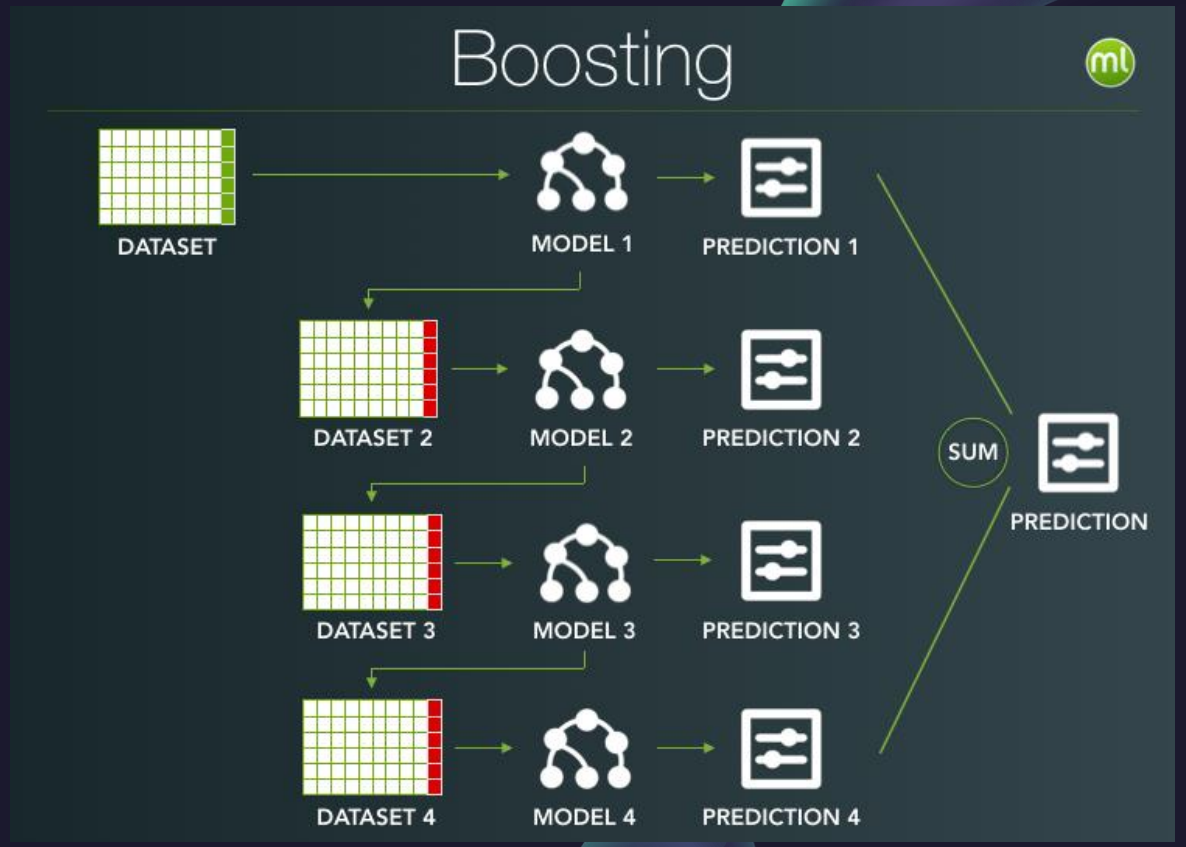
В библиотеке sklearn в модуле ensemble есть реализация BaggingRegressor для беггинга в задачах регрессии и BaggingClassifier для беггинга в задачах классификации.

Бустинг

Бустинг

Бустинг в машинном обучении — это мета-алгоритм, который позволяет создать сильный обучающий алгоритм из коллекции слабых.

Основная идея бустинга — комбинирование слабых функций, которые строятся в ходе итеративного процесса, где на каждом шаге новая модель обучается с использованием данных об ошибках предыдущих.

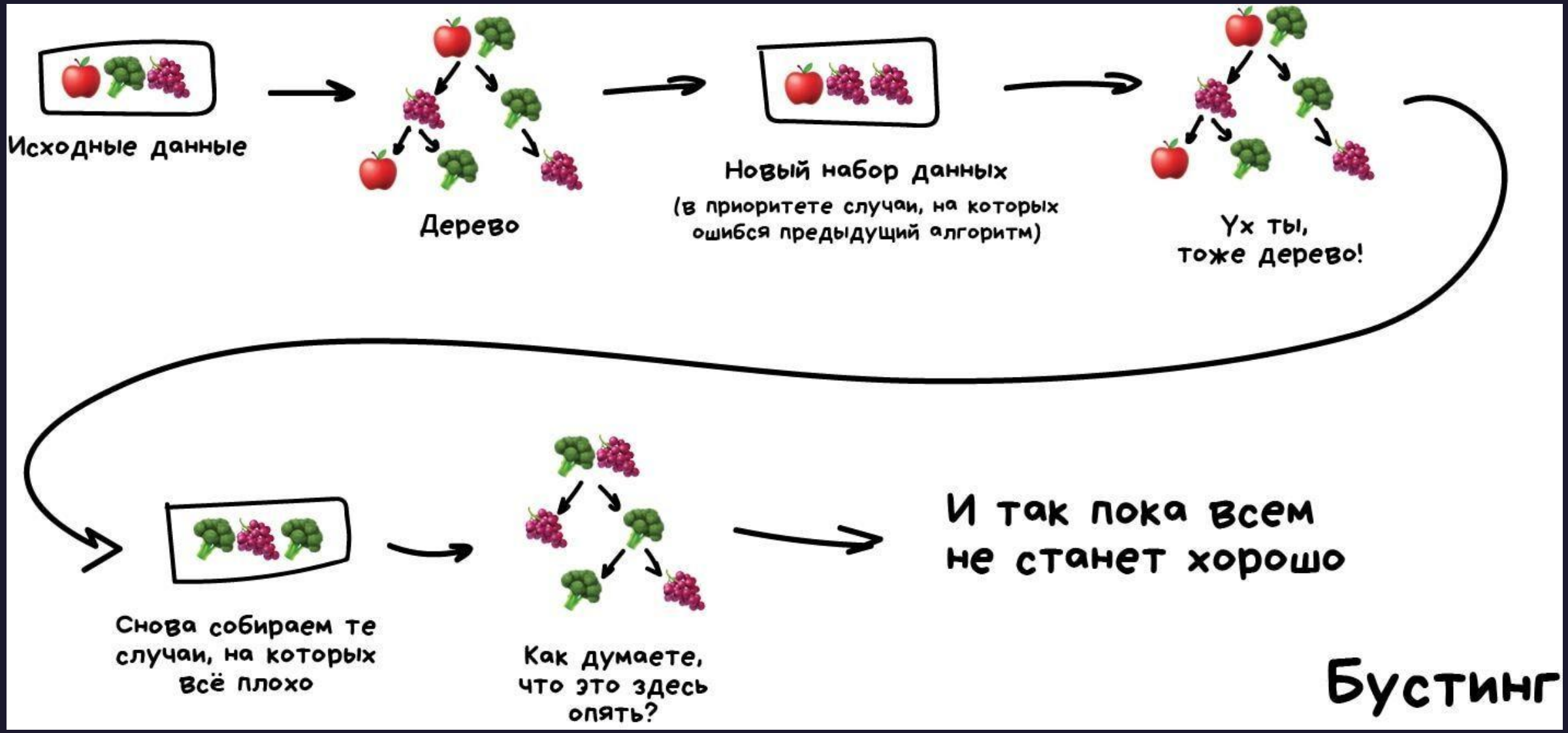


Бустинг

Некоторые преимущества бустинга:

- Повышение точности. Бустинг объединяет точности нескольких слабых моделей и усредняет их для регрессии или проводит голосование для классификации, что увеличивает точность финальной модели.
- Устойчивость к переобучению. Бустинг снижает риск переобучения, перевзвешивая входные данные, которые классифицированы неправильно.
- Лучшая обработка несбалансированных данных. Бустинг фокусируется больше на неправильно классифицированных точках данных.
- Повышение интерпретируемости модели. Бустинг увеличивает интерпретируемость модели, разбивая процесс принятия решения модели на несколько процессов.

Бустинг


Исходные данные
Дерево
Новый набор данных
(в приоритете случаи, на которых ошибся предыдущий алгоритм)
Ух ты, тоже дерево!
И так пока всем не станет хорошо
Снова собираем те случаи, на которых всё плохо
Как думаете, что это здесь опять?
Бустинг

Бустинг

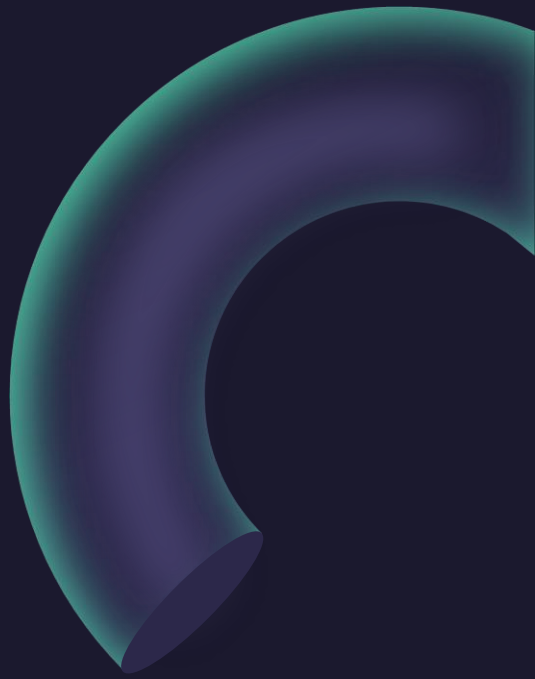
Основная цель: Уменьшение смещения (bias). Бустинг превращает множество "слабых учеников" (например, маленьких пней решений — decision stumps) в один очень сильный алгоритм.

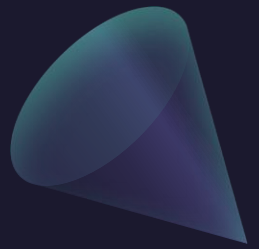
Самые известные примеры:

AdaBoost (Adaptive Boosting): Классический алгоритм, который динамически назначает веса объектам на каждом шаге.

Gradient Boosting Machine (GBM): Более общий и мощный алгоритм, где последующие модели обучаются на градиенте функции потерь (а не прямо на ошибках). Это как спускаться вниз по градиенту ошибки, чтобы найти минимум.

XGBoost, LightGBM, CatBoost: Современные, чрезвычайно эффективные реализации градиентного бустинга, которые являются золотым стандартом на многих соревнованиях по машинному обучению.





Стекинг

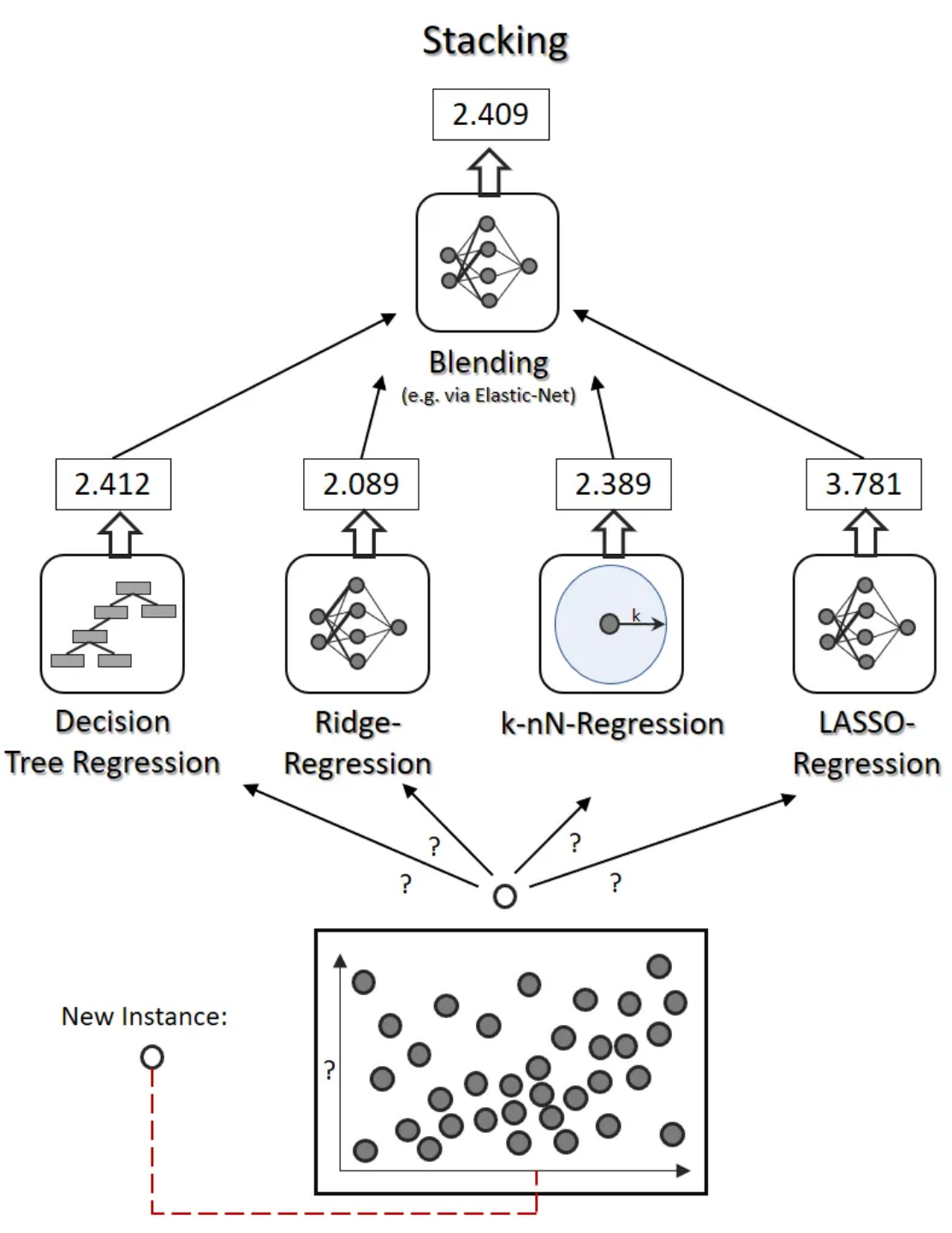
Стекинг

Стекинг (Stacking) — это метод ансамблирования, в котором несколько моделей комбинируются на уровне метамодели. Основная идея стекинга заключается в использовании предсказаний нескольких моделей в качестве входных данных для новой модели (метамодели), которая обучается на этих предсказаниях для улучшения точности итогового предсказания.

Процесс стекинга включает следующие этапы:

- Несколько базовых моделей обучаются на исходных данных, и их предсказания сохраняются.
- На основе предсказаний базовых моделей обучается метамодель.
- Итоговое предсказание формируется на основе метамодели, которая может учитывать зависимость между предсказаниями базовых моделей.

.



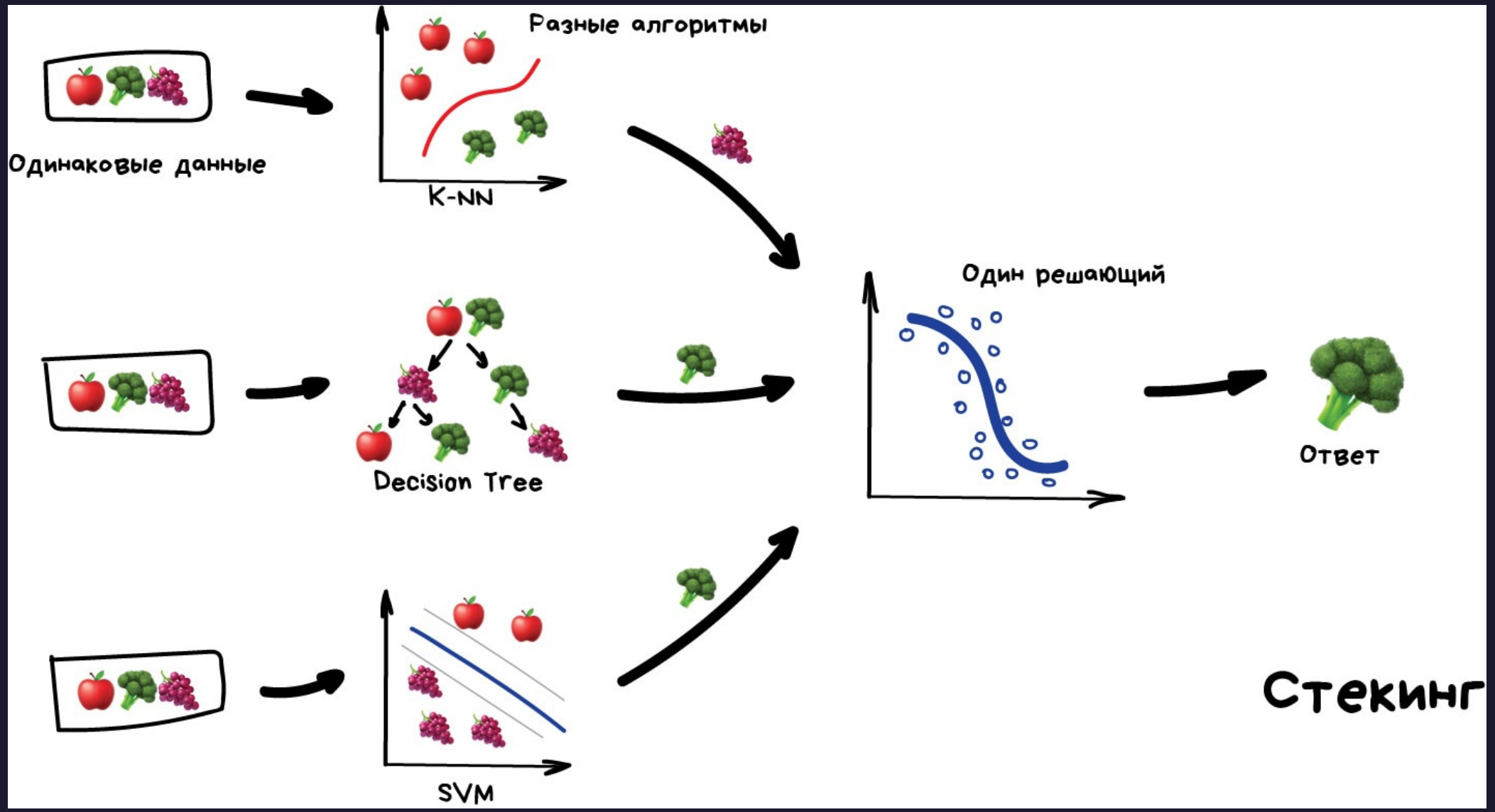
Стекинг

Преимущества стекинга:

- позволяет комбинировать модели с разными подходами, что может значительно повысить точность предсказания;
- даёт гибкость в выборе базовых моделей и метамодели. Часто метамодель представляет собой простую модель, такую как логистическая регрессия или линейная регрессия, которая усредняет предсказания базовых моделей.

Стекинг находит применение в задачах, где требуется высокая точность, таких как соревнования по анализу данных (например, Kaggle), где используется комбинация различных моделей для достижения лучших результатов.

Стекинг


Разные алгоритмы
Одинаковые данные
K-NN
Один решающий
Decision Tree
Ответ
SVM
Стекинг

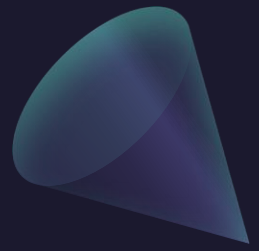
Стекинг

Основная цель: Максимально использовать сильные стороны разных алгоритмов. Модель-объединитель учится, какой алгоритм в какой ситуации доверять больше.

Еще один вариант ансамблей - стекинг - когда объединяются результаты разнородных моделей с помощью еще одной модели. В sklearn реализованы:

StackingClassifier - классификатор

StackingRegressor - регрессор



Сравнительная таблица

Характеристика	Бэггинг (Bagging)	Бустинг (Boosting)	Стекинг (Stacking)
Основная цель	Уменьшение **дисперсии**	Уменьшение **смещения**	Повышение **общей точности**
Обучение моделей	Параллельное	Последовательное	Параллельное (на 1-м уровне)
Взаимодействие	Модели независимы	Новые модели исправляют ошибки старых	Модели объединяет мета-алгоритм
Вес объектов	Равный вес, выборка с возвращением	Вес ошибок увеличивается	Используются прогнозы моделей
Склонность к переобучению	Устойчив, уменьшает переобучение	Высокая, требует аккуратной настройки	Высокая, требует аккуратной настройки
Примеры	Random Forest	AdaBoost, XGBoost, LightGBM	Часто кастомные ансамбли

Выбор между этими методами зависит от задачи, данных и вычислительных ресурсов. Random Forest — отличный выбор для быстрого построения надежной модели. Градиентный бустинг (XGBoost, etc.) часто дает максимальную точность, но требует больше времени для настройки. Стекинг — это мощный инструмент для соревнований и сложных задач, где нужно выжать максимум из данных.

Спасибо