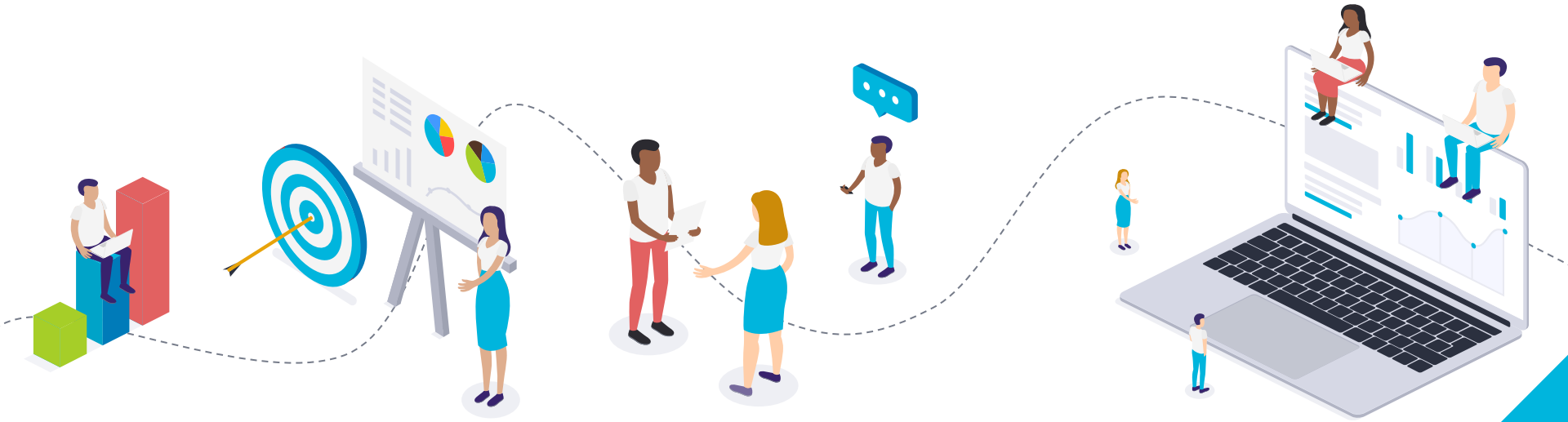
МИНИСТЕРСТВО НАУКИ И ВЫСШЕГО ОБРАЗОВАНИЯ РОССИЙСКОЙ ФЕДЕРАЦИИ
федеральное государственное бюджетное образовательное учреждение высшего образования
«АЛТАЙСКИЙ ГОСУДАРСТВЕННЫЙ УНИВЕРСИТЕТ»

Программа профессиональной переподготовки
«Технологии искусственного интеллекта, визуализации и анализа данных»



Ансамблевые методы

- Задача ансамблевых методов состоит в том, чтобы объединять несколько моделей (слабых учеников, базовых моделей) в одну, чья обобщающая способность будет лучше, чем у каждой модели в отдельности.

  Для сбора ансамблей применяют следующие подходы:
  - Stacking (стекинг)
  - Bagging (беггинг)
  - Boosting (бустинг)

Stacking

- Позволяет объединять разнородные модели (например, KNN, SVM и логистическую регрессию)
- Наименее популярный

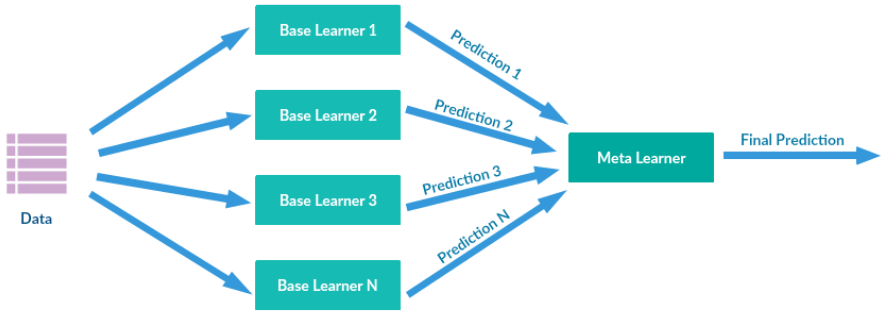
Алгоритм:

• Выборка данных делится на *k* частей (блоков).

• Для каждого объекта *k*-ой части делается предсказание слабыми алгоритмами, которые были обучены на *k*-1 частей. Этот процесс итеративен и происходит для каждой части выборки.

• Создается набор прогнозов слабых алгоритмов для каждого объекта выборки.

• На сформированных прогнозах обучается метамодель.



Stacking

Для классификации: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.StackingClassifier.html
Параметры:

- **estimators** – базовые (слабые) алгоритмы.
- **final_estimator** – итоговый классификатор – метамодель (по умолчанию LogisticRegression).
- **cv** – количество частей для разбиения выборки.

Для регрессии: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.StackingRegressor.html
(**final_estimator** по умолчанию RidgeCV).

Bagging

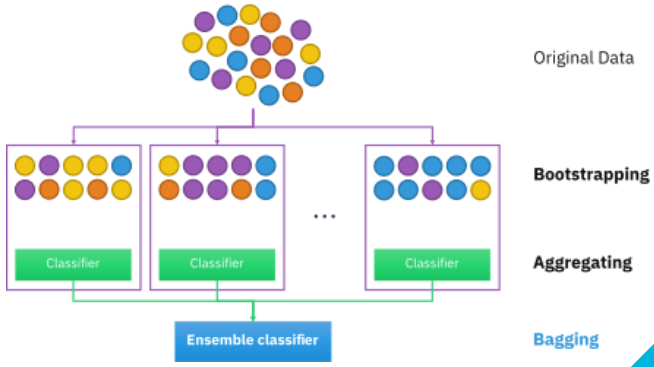
- Более популярный метод
- Позволяет объединять только однородные модели

Алгоритм:

• Выборка данных делится на *бутстрэп*-выборки.

При использовании бутстрэпа из исходной выборки размером *t* берется случайный объект и записывается в обучающую выборку. Следующий объект тоже берется случайным образом из исходной выборки размером *t*. Так повторяется *n* раз, где *n* – желаемый размер обучающей выборки. Таким образом, должны сформироваться *m* обучающих выборок для *m* слабых алгоритмов (как правило, деревьев решений).

- На каждой выборке обучается одна модель.
- В задаче классификации класс для нового объекта определяется мажоритарным голосованием. В задаче регрессии ответы моделей усредняются.



Bagging

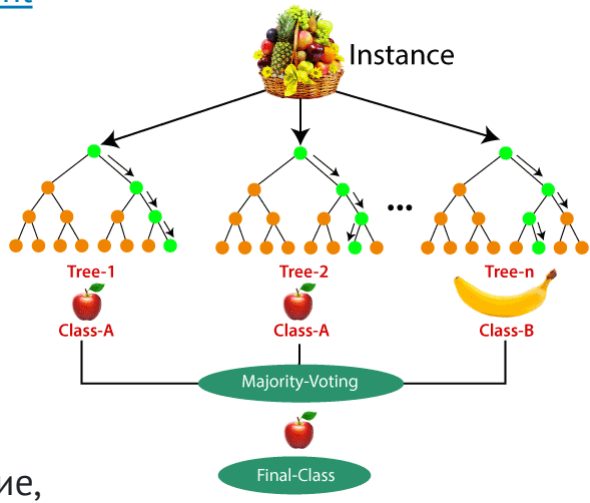
Для классификации: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingClassifier.html
Параметры:
base_estimator – базовый алгоритм
n_estimators – количество базовых алгоритмов в ансамбле

Для регрессии: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.BaggingRegressor.html

Случайный лес (Random Forest)

- Частный случай бэггинга
- Используется метод случайных подпространств: базовые алгоритмы обучаются на различных подмножествах признакового описания, которые также выделяются случайным образом.

Для классификации: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestClassifier.html
Параметры:
n_estimators (по умолчанию 100) – количество деревьев в лесу.
criterion – критерий (по умолчанию gini).
max_depth – максимальная глубина деревьев (по умолчанию None, т.е. полное дерево).
max_features – количество случайно выбранных признаков, рассматриваемых для расщепления
bootstrap (по умолчанию True, если установить False, то будет использоваться исходная выборка).
max_samples – количество объектов, которые будут извлечены из выборки с помощью бутстрэпа, если bootstrap = True (по умолчанию используется значение, равное количеству элементов в исходной выборке).
Для регрессии: https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.RandomForestRegressor.html



Boosting

В данном подходе, модели не обучаются отдельно друг от друга, а каждая следующая старается исправить ошибки предыдущей. Если один слабый ученик не смог выявить какую-либо закономерность в данных, так как это было для него сложно, то следующий должен сделать это.

Исходная процедура бустинга может быть резюмирована в четырех ключевых шагах:

1) извлечь случайное подмножество обучающих объектов d_1 из исходного набора D (без повторений) для тренировки слабого ученика C_1 .
2) извлечь второе случайное подмножество d_2 из исходного набора и добавить 50% ранее ошибочно классифицированных объектов для тренировки слабого ученика C_2 .
3) найти в наборе данных D объекты d_3, по которым предсказания C_1 и C_2 расходятся, для тренировки третьего слабого ученика C_3.
4) объединить слабых учеников посредством мажоритарного голосования.

Boosting

Градиентный бустинг:

- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingClassifier.html
- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.GradientBoostingRegressor.html

Адаптивный бустинг:

- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostClassifier.html
- https://scikit-learn.org/stable/modules/generated/sklearn.ensemble.AdaBoostRegressor.html