**Лабораторная работа №4**

АНСАМБЛЕВЫЕ МЕТОДЫ

По дисциплине «Машинное обучение»

Выполнил студент

группы 3530904/80102: Шерман М.Л.

Преподаватель: Селин И.А.

Оглавление

[Задачи 3](#_Toc71488987)

[Пункт 1 4](#_Toc71488988)

[Пункт 2 5](#_Toc71488989)

[Пункт 3 5](#_Toc71488990)

[Вывод 7](#_Toc71488991)

[Приложение 8](#_Toc71488992)

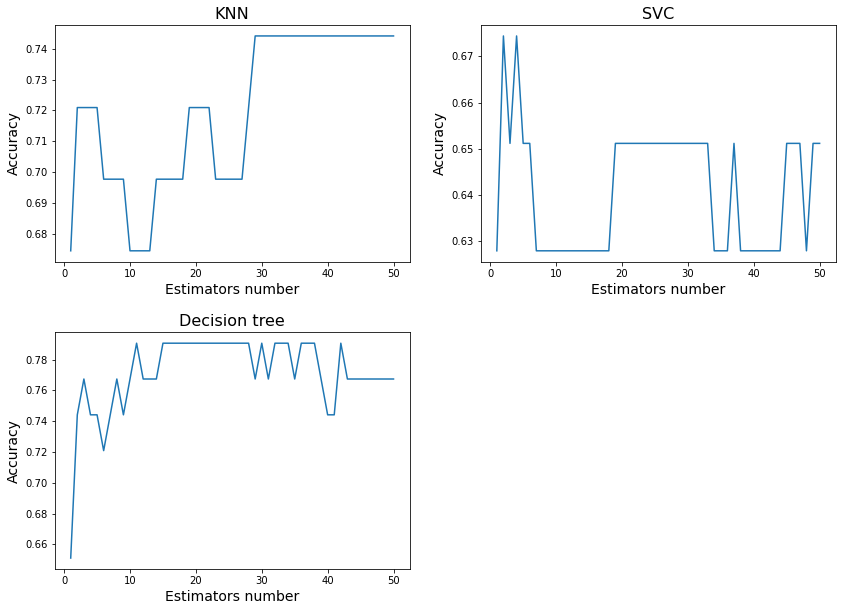
# Задачи

1. Исследуйте зависимость качества классификации от количества классификаторов в ансамбле для алгоритмов бэггинга на наборе данных glass.csv с различными базовыми классификаторами. Постройте графики зависимости качества классификации при различном числе классификаторов, объясните полученные результаты.
2. Исследуйте зависимость качества классификации от количества классификаторов в ансамбле для алгоритма бустинга (например, AdaBoost) на наборе данных vehicle.csv с различными базовыми классификаторами. Постройте графики зависимости качества классификации при различном числе классификаторов, объясните полученные результаты.
3. Постройте мета-классификатор для набора данных titanic\_train.csv используя стекинг и оцените качество классификации на titanic\_train.csv

# Пункт 1

В данной лабораторной работе мы будем рассматривать ансамблевые методы: бэггинг, бустинг и стекинг. Ансамблевые методы — это парадигма машинного обучения, где несколько моделей (часто называемых «слабыми учениками») обучаются для решения одной и той же проблемы и объединяются для получения лучших результатов. Основная гипотеза состоит в том, что при правильном сочетании слабых моделей мы можем получить более точные и/или надежные модели. В первом пункте лабораторной рассмотрим применение бэггинга.

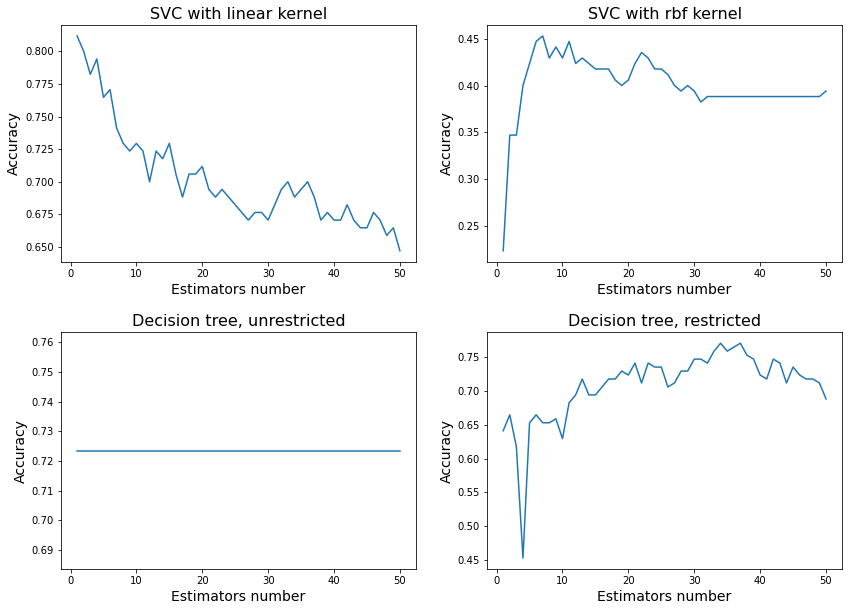
Суть данного метода состоит в том, что мы будем рассматривать несколько однородных моделей (например svm). Все модели будут обучаться параллельно и независимо друг от друга, а затем объединим их, следуя некоторому детерминированному процессу усреднения (зависит от реализации).

Рассмотрим применение данного метода, взяв в качестве моделей k ближайших соседей, svm с линейным ядром и дерево решений. Также рассмотрим зависимость точности классификации от числа моделей.

По графикам видно, что благодаря бэггингу удалось улучшить точность предсказания, однако это справедливо только для определённого числа моделей. Так, для при использовании моделей, на основе метода k ближайших соседей потребовалось 30 экземпляров, чтобы достичь наилучшего результата, примерно 21 для svm (конечно, наилучший результат был достигнут примерно с 3-4 экземплярами, но это скорее всего из-за случайности разбиения выборки) и около 20 при использовании деревьев решений. Самый большой рост точности был при использовании деревьев решений.

# Пункт 2

Перейдём к следующему ансамблевому методу, а именно к бустингу. В данном случае однородные модели обучаются последовательно, причём каждая последующая зависит от результатов предыдущих. Каждая новая модель фокусирует свои усилия на самых сложных объектах выборки при обучении предыдущих моделей, чтобы мы получили в конце процесса «сильную» модель с более малым смещением.

Аналогично предыдущему пункту возьмём три модели, однако заменим k ближайших соседей на svm с ядром rbf, а также рассмотрим дерево решений с ограничением по глубине и без. Также воспользуемся реализацией бустинга AdaBoost библиотеки scikit-learn.

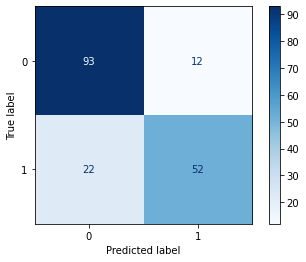
К сожалению, бустинг не смог показать хороших результатов с алгоритмом svm. В методе опорных векторов с линейным ядром при увеличении числа моделей качество ухудшалось. С ядром rbf ситуация противоположна, но при использовании данной модели даже не удалось достичь точности в 50%. Необычно обстоит ситуация при использовании дерева решения без ограничения глубины. Данная ситуация происходит из-за переобучения данной модели. Если мы поставим ограничение глубины дерева (в примере использована глубина 3), то на промежутке от 30 до 35 экземпляров мы получим наилучший результат.

# Пункт 3

Третьим ансамблевым методом является стекинг. Идея данного метода состоит в использовании разнородных моделей и их объединении для получения метамодели.

Построим метаклассификатор на основе случайного леса и svm (с линейным ядром и rbf), однако до этого рассмотрим предоставленные данные.

В данном пункте мы будем обучать модель на известном датасете, содержащим информацию о пассажирах Титаника. Взглянув на данные, можно увидеть признаки, которые для рассматриваемых нами моделей вряд ли понадобятся, в силу своей уникальности (имена пассажиров, номер билета и кабины). От этих данных мы избавимся. У оставшихся признаков найдём пропущенные значения. Признак Age имеет 177 пропущенных значений, а Embarked 2. Второй пока оставим, а пустые значения возраста заполним медианой оставшихся возрастов пассажиров. Попробуем найти категориальные признаки. При первом взгляде на данные такими признаками являются Sex и Embarked, попробуем отыскать «скрытые» признаки по количеству уникальных значений. В результате категориальными признаками оказались Pclass, Sex, SibSp, Parch, Embarked. Признак Survived мы будем предсказывать, поэтому оставим в исходном состоянии. К остальным категориальным признакам применим алгоритм One-Hot Encoding, для улучшения работы модели.

После обработки данных создадим метаклассификатор и обучим его. В результате получим точность 81%. Матрица ошибок:

Как мы видим, полученный метаклассификатор хорошо справляется с определением выживших и погибших, хоть и с небольшими огрехами.

# Вывод

В результате проделанной работы удалось познакомиться с ансамблевыми методами – бэггингом, бустингом и стекингом. Все три метода преследуют одну цель – улучшить качество предсказания, путём использования множества обычных моделей, однако каждый из них предлагает свой, уникальный вариант решения проблемы. В данной работе мы лишь начали изучение этих важнейших методов, однако с полученной базой не составит труда и дальше углубиться в изучение данных подходов.

# Приложение

Весь код и графики можно найти в следующем репозитории:

<https://github.com/Mark-Sherman-SE/ML-Labs>