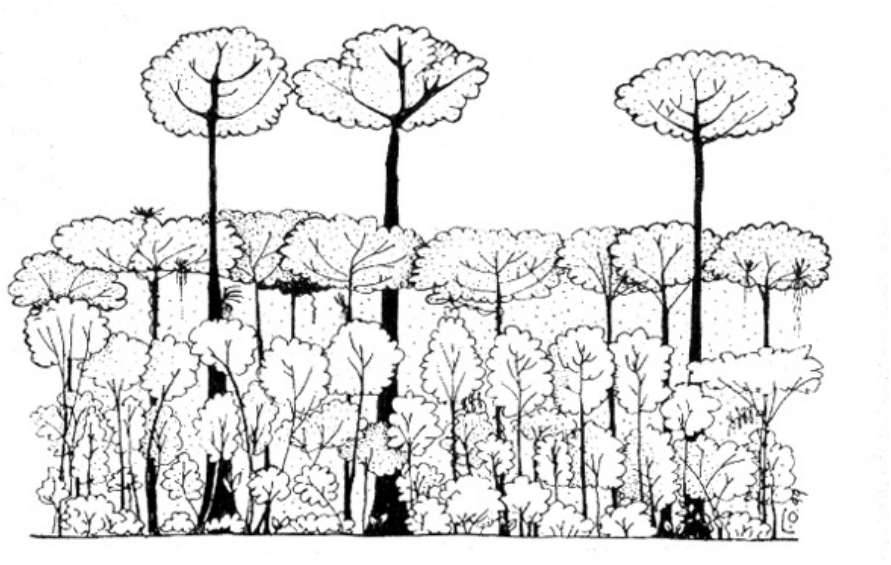
Лабораторная 2: Спартанские леса



Недавно в Спарте придумали блоги. Вам дается датасет BlogFeedback
https://archive.ics.uci.edu/ml/datasets/BlogFeedback

Каждый объект выборки — пост в спартанском блоге. Он описывается различными признаками: длина текста поста, наличие наиболее частотных слов, день недели, количество комментариев за последние 24 часа и т.п., а так же целевым признаком — количеством комментариев к посту. Полный список и описание находятся на странице датасета.

Разбейте датасет на обучающую и тестовую выборку в соотношении 1 к 3, обучающая меньше тестовой. Рассматривается задача регрессии с метрикой MSE (средний квадрат ошибки). Для выполнения потребуются библиотеки `scikitlearn` и `xgboost`.

Дедлайн и технические детали появятся тут в течении нескольких дней.

Задания:

1. **(1 балл)** Обучите Random Forest по обучающей выборке. Постройте и сравните графики зависимости ошибки от количества деревьев в композиции: на обучающей выборке, на тестовой выборке, используя ошибку out-of-bag (`oob_prediction_`). Сделайте вывод.

2. **(1 балл)** Постройте график распределения важности признаков (обозначим этот набор $\mathcal{F}$). Отберите признаки, которые дают порядка 95% качества, (обозначим этот набор $\mathcal{F}_1$). Постройте на одном графике зависимость ошибки прогноза в зависимости от количества деревьев для выборки по всем признакам и по выбранным. Сделайте вывод.

3. **(3 балла)** Подберите оптимальные значения параметров `max_features` и `min_samples_split`. Для этого возьмите сетку по этим параметрам (не менее пяти значений каждого параметра) и для каждой пары параметров обучите Random Forest с этими значениями параметров до тех пор, пока ошибка на обучении не стабилизируется (график будет почти постоянным). Для некоторых параметров постройте графики ошибки прогноза на обучающей и на тестовой выборке в зависимости от количества деревьев. Визуализируйте так же матрицу ошибок (как в первой лабораторной).

Момент стабилизации графика можно отслеживать глазами, либо автоматически. В последнем случае точкой стабилизации можно считать момент, для которого на протяжении некоторого количества предыдущих итераций (не меньше 10) качество изменяется не сильно. Если вы обучили композицию, а деревьев все равно не достаточно, не нужно делать обучение заново, можно дообучить модель. Для этого нужно воспользоваться методом `set_params` у уже обученной модели, в который передать новое количество деревьев, и снова вызвать `fit`.

Какие параметры получились оптимальными? Сделайте выводы.

4. **(4 балла)** Проведите аналогичные исследования для градиентного бустинга для реализаций из библиотек `scikitlearn` и `xgboost`. Сделайте выводы.

5. **(1 балл)** В пунктах 2-3 вы должны получить три лучших модели для разных методов и их реализаций. Сравните их между собой по качеству и по времени работы. Обучите эти модели с такими же параметрами на обучающей выборке с множеством признаков $\mathcal{F}_1$. Сделайте выводы.

6. **(1 балл)** Обучите три лучшие модели на большом количестве деревьев (не менее 1000). Постройте графики ошибки на обучении и на тесте. Удалось ли достигнуть переобучения?

7. **(1 балл)** Возьмите лучшую композицию из трех предыдущих в качестве начального приближения для градиентного бустинга, то есть обучайте бустинг на ошибки, которые дает композиция. Аналогичную операцию проделайте для начального приближения линейной регрессией (перед этим промасштабируйте данные). Дали ли подобные подходы прирост качества по сравнению с обычным градиентным бустингом и почему?

8. **(1 балл)** Обучите линейную регрессию (LR), Random Forest (RF) и градиентный бустинг (GB). Подберите оптимальные α для модели $\alpha RF + (1-\alpha)GB$ и $\alpha LR + (1-\alpha)GB$. Сравните качество новых моделей между собой и моделями из предыдущего пункта.