Жамков Никита

Метамодели

**Начало**

Я провел небольшое исследование на простом датасете (Customer Telecom Churn from **Kaggle**), для которого решается задача *бинарной классификации* (1 или 0). Ради задания я не проводил тяжелый анализ или обработку данных, так как, логично, что на необработанных данных, что на обработанных данных – точность будет изменяться прямо пропорционально. К примеру:

Если на необработанных данных у первой модели точность – 0.75, у второй 0.60, а на обработанных данных у первой – 0.89, у второй – 0.74, то пропорция увеличения точностей осталась прежней.

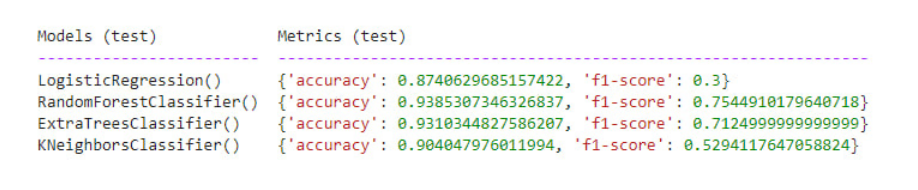
(*если изменения и будут наблюдаться – то незначительные*)

**Исследование**

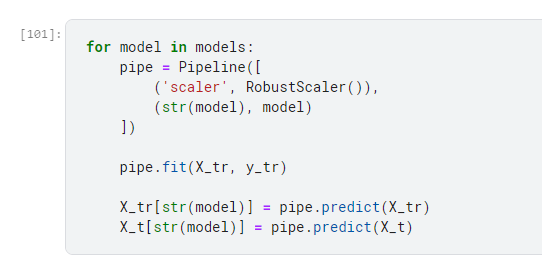
Я решил разделить датасет на 20% к 80% и применить RobustScaler к данным для обучения (X). Для проверки точности каждой выбранной модели из списка:

* LogisticRegression
* RandomForestClassifier
* ExtraTreesClassifier
* KNN

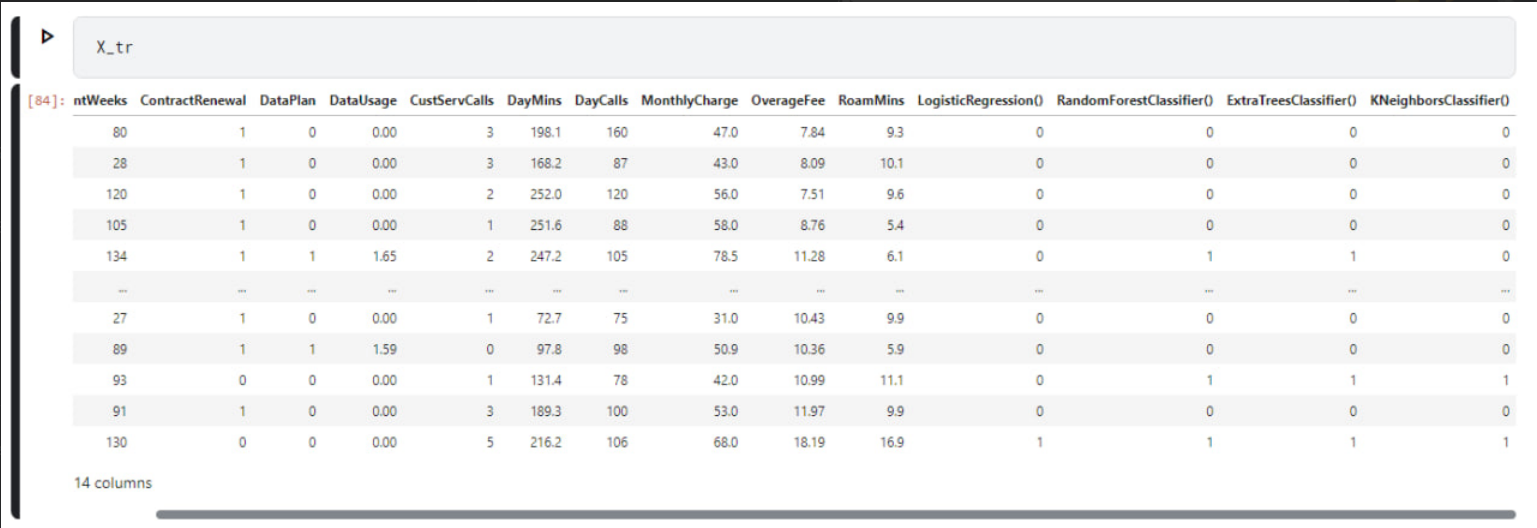
Вставил каждую модель в свой модуль для определения точностей ([GMS](https://github.com/plugg1N/gms-module)) каждой из модели на валидационной выборке, результаты оказались следующими:



Далее, я решил создать pipeline (*sklearn.pipeline.Pipeline*) для каждой модели, чтобы автоматически применять, сначала, RobustScaler, а потом тренировать модель и/или получать predict.

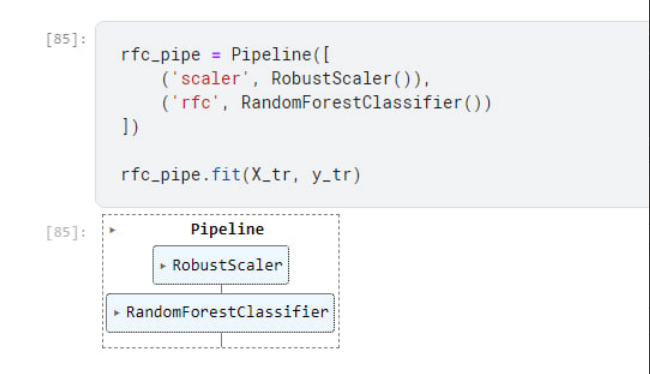


После применения pipeline-ов, наш датасет выглядит следующим образом:



Можно заметить, что появились колонки справа, которые принимают значения либо 0, либо 1. Это результат predict-ов моделей из списка, указанного раннее.

Далее, я решил создать очередной pipeline, только моделью будет являться лучшая модель, проявившая себя на нашем датасете – это RandomForestClassifier, с точностью *0.75449101 F1-score.*



Результаты оценки точности получились следующими:

* Accuracy: 0.938530734
* F1-score: 0.751515151

Еще раз обратим внимание на таблицу из оценок точности КАЖДОЙ модели, проверенной в модуле GMS. Можно понять, что точность F1-score незначительно ухудшилась.

Появляется гипотеза: *«…если мы используем RandomForestClassifier, как главную модель для оценки, конечно же эта модель* *будет иметь такую же точность, как RandomForestClassifier отдельно!*»

Проверим гипотезу на следующей странице…

Я решил, в качестве модели для независимой оценки, использовать модель, которой не было в **списке.** Этой моделью стал:

Light Gradient Boosting Machine Classifier (*или* LGBMClassifier)

Проделав те же самые операции и построив новый pipeline с новой моделью независимой оценки, получаем следующий результат:

* Accuracy: 0.938530734
* F1-score: 0.751515151

Точность такая же абсолютно, как и в прошлый раз.

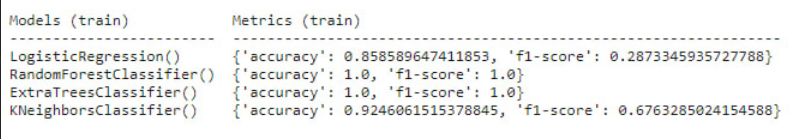
Прошлый pipeline:



Новый pipeline:



После некоторого времени рассуждений, можно понять, почему так получается. GMS модуль проанализировал не только валидационную выборку, но и тренировочную, и собрал точности каждой модели во время тренировки.



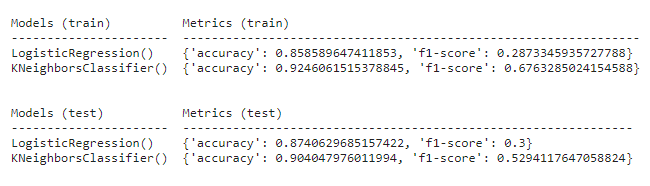
Можно заметить, что лучшая модель RandomForestClassifier переобучилась на тренировке. Так как *Accuracy и F1-score* точны на 100%, так как это практически невозможно.

Теперь можно рассмотреть ситуацию со стороны LGBM Classifier. В теории, у каждой колонки есть свой параметр “Importance” (с англ. *важность*). Модель, опираясь на важность каждой колонки, делает вывод.

LGBM Classifier видит и сравнивает ответы RandomForestClassifier и колонки “Churn”, которую LGBM и пытается предсказать. Так как RandomForestClassifier отвечает с точностью 100% на тренировочном датасете, независимой модели нет смысла проводить собственный анализ, и модель начинает просто перенимать ответы с RandomForestClassifier.

Появляется новая гипотеза: «…*раз RandomForestClassifier и подобные ему модели переобучаются на тренировке, давайте возьмем модели, которые не подвержены переобучению на тренировке*»

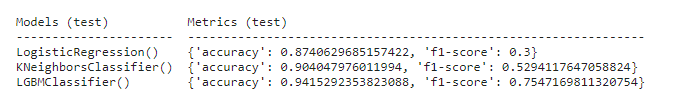
Проверка: я убрал две модели, которые переобучились на тренировке. Остались две:



Добавил ответы только двух этих моделей в таблицу и обучил независимую модель LGBM Classifier на новой таблице. Результаты получились следующие:

* Accuracy: 0.9295352323838081
* F1-score: 0.7044025157232705

Точность явно повысилась, однако если обучить LGBM Classifier отдельно без других моделей, то точность LGBM Classifier получится следующей:



Можно увидеть, что точность LGBM Classifier, отдельно больше, причем Accuracy увеличилась на 0.02 пункта, а F1-score на 0.05 пунктов.

**Заключение**

Применение метамодели не увеличило точность предсказаний. Проверяя первую гипотезу, изменения были незначительны, проверяя вторую гипотезу, точность оказалась не самой оптимальной из всех возможных.