Некоторые идеи:

1. Preprocessing:

- разбивка на выборки: размеры, стацифицирование
- поработать с категориальными признаками
- что делать с NaN? (уже есть замена средним, попробовать медиану), нули
- что лучше - нормировка или шкалирование на максимум? можно ли обойтись без нормировок?
- посмотреть какие столбцы влияют сильнее, если что - разумно выкинуть ненужные (предположить)
- добавить новых признаков (обосновать)
- (опционально) выделить главные компоненты, проанализировать значения. ~~Если разумно - использовать для визуализации~~

2. Baseline и простые алгоритмы:

- уже есть в стартовой реализации: RandomForest, но без подбора параметров.
- Добавить расчёт целевой метрики. Определять эффективность изменений в пункте 1 на данном этапе

- (опционально, но желательно) посмотреть еще какие-нибудь простые алгоритмы: ~~NaiveBayes, LogReg,~~ SVM (выбрать ядро), ... Подобрать параметры. Построить графики зависимости целевой метрики от сложности модели (то же и для размера ансмабля в случае RandomForest)
- (опционально) ансамблирование простых моделей из прошлого пункта, попробовать Blending, Stacking

3. Boosting:

- посмореть работу на разных библиотеках: lightgbm, catboost (говорят, дает высокие результаты в данной задаче), xgboost. Подобрать параметры
- (опционально) построить графики зависимости целевой метрики от сложности модели
- (опционально) ансамбли лучших моделей boosting-а (см. аналогичный предыдущий пункт)

4. Анализ результатов:g

- (опционально, но желательно) оценить важность признаков по соответствующим графикам (можно использовать shap value)

К чему пришел Слава:

Исключить MRG (для всех одинаков)
Исключить REGION
Исключить TOP_PACK
AUC
0. Baseline: RF (из starter notebook-а): 0.8974371957198335 (в стартовой реализации данные теста были слиты заменой Nan)
1. Baseline: RF (из starter notebook-а): 0.8961048043598108 (правильная работа с Nan)
2. Baseline: RF (из starter notebook-а): 0.8958074686549761 (правильная работа с TENURE)

3. Baseline: RF (из starter notebook-a): 0.8969799790913013 (без данных TENURE)
Исключить TENURE

4. Baseline: RF (из starter notebook-a): 0.8977189279904491 (с медианой вместо NaN, вместо среднего)
Ипользовать медиану
5. Baseline: RF (из starter notebook-a): 0.8981660830846924 (изменил размеры выборок, тест – 0.3 * train (остальное обучающая), валидация – 0.2 от обучающей на предыдущем этапе)
6. Baseline: RF (из starter notebook-a): 0.898166019844284 (без нормировки)
7. Baseline: RF (из starter notebook-a): 0.898168 7192856287 (MinMaxScaler)
Можно использовать MinMaxScaler (но в целом без разницы)
8. Baseline: RF (из starter notebook-a): 0.8976 01632014879 (нули вместо Nan)

9. Новые признаки:
 # Варианты:
 # общее число звонков ['ON_NET'] + ['ORANGE'] + ['TIGO'] + ['ZONE1'] + ['ZONE2']
 # MONTANT / FREQUENCE_RECH (средняя сумма пополнения)
*# ARPU_SEGMENT * 3 - MONTANT (сколько оставалось после пополнений)*
 # DATA_VOLUME / REGULARITY (плотность активности, т.е. число подкючений на число входов в сеть)
 # общее число звонков / REGULARITY: 0.8988783864258681

10. RF (с подборкой параметров, скоринг по AUC): 0.8990601322636138
```
{'max_depth': 7, 'max_samples': 0.6, 'n_estimators': 300}
```

11. Взять 10 лучших моделей RF и усреднить предсказания: AUC: 0.8988 01781483697

12. Baseline: RF (из starter notebook-a): работа с Nan (выборочно нули, где-то – медина), добавил новый признак: 0.8975787933559122