

# ПРОГНОЗИРОВАНИЕ ПЛАТЯЩЕГО ИГРОКА НА ОСНОВЕ ПЕРВОГО ДНЯ ЖИЗНИ

Выполнила: Бахитова Алина Асылановна

Руководитель: к.э.н. доцент Заграновская Анна Васильевна

# Контекст

- **Игра:** бизнес-модель free-to-play (F2P) - можно скачать и играть бесплатно
- **Доход в игре:**
  - покупки внутриигровых товаров за деньги
  - просмотр игроками рекламы
- **Проблема:** наличие рекламы снижает кол-во покупок за реальные деньги
- **Цель бизнеса:** максимизация дохода

Для максимизации дохода нужно как можно раньше определять платящего игрока и не показывать ему рекламу

# Цель и задачи работы

## Цель:

- Выбор и настройка модели прогнозирования платящего игрока по данным за первый день жизни

## Задачи:

- Сбор данных об игровом поведении за первый день жизни и факта платежа к 7 дню жизни
- Обработка собранных данных для использования в прогнозной модели
- Разработка метрики оценки качества прогнозной модели
- Автоматизированный подбор наилучшей прогнозной модели
- Настройка выбранной прогнозной модели

# Исходные данные

- Мобильная игра в жанре управления временем (time-management)
- В игре нужно проходить уровни на скорость
- 1,361,667 игроков, которые установили игру в течение трех месяцев
- 63 факторных признака (10 бинарных, 53 числовых) + 1 целевая метрика

# Исходные данные

## Целевая метрика:

- заплатил ли игрок к 7 дню жизни или нет

## Проблема:

- 1% платящих игроков (несбалансированность)

# Факторные признаки

**Общие показатели** - переменные, относящиеся к общему игровому опыту

- например, была ли игра установлена ранее игроком или авторизовался ли игрок через сторонние сервисы (Facebook или Google Play);

**Показатели вовлеченности** - переменные, связанные с активным участием игрока в игре

- например, количество стартов уровней и время, проведенное в игре;

**Показатели эффективности прохождения уровней** – переменные, относящиеся к игровому прогрессу и используемым стратегиям прохождения уровней

- например, доля побед или использование бустеров на уровнях;

**Социальные показатели** – переменные, относящихся к социальному взаимодействию

- например, кол-во друзей в игре или вступил ли он гильдию;

**Технические показатели** – технические настройки в игре

- например, процент входов в игру, которые сопровождались сбоем игры или размер экрана устройства игрока

Всего 63 факторных признака

# Факторные признаки

**Общие показатели** - переменные, относящиеся к общему игровому опыту

- например, была ли игра установлена ранее игроком или авторизовался ли игрок через сторонние сервисы (Facebook или Google Play);

**Показатели вовлеченности** - переменные, связанные с активным участием игрока в игре

- например, количество стартов уровней и время, проведенное в игре;

**Показатели эффективности прохождения уровней** – переменные, относящиеся к игровому прогрессу и используемым стратегиям прохождения уровней

- например, доля побед или использование бустеров на уровнях;

**Социальные показатели** – переменные, относящихся к социальному взаимодействию

- например, кол-во друзей в игре или вступил ли он гильдию;

**Технические показатели** – технические настройки в игре

- например, процент входов в игру, которые сопровождались сбоем игры или размер экрана устройства игрока

Всего 63 факторных признака

# Факторные признаки

**Общие показатели** - переменные, относящиеся к общему игровому опыту

- например, была ли игра установлена ранее игроком или авторизовался ли игрок через сторонние сервисы (Facebook или Google Play);

**Показатели вовлеченности** - переменные, связанные с активным участием игрока в игре

- например, количество стартов уровней и время, проведенное в игре;

**Показатели эффективности прохождения уровней** – переменные, относящиеся к игровому прогрессу и используемым стратегиям прохождения уровней

- например, доля побед или использование бустеров на уровнях;

**Социальные показатели** – переменные, относящихся к социальному взаимодействию

- например, кол-во друзей в игре или вступил ли он гильдию;

**Технические показатели** – технические настройки в игре

- например, процент входов в игру, которые сопровождались сбоем игры или размер экрана устройства игрока

Всего 63 факторных признака

# Факторные признаки

**Общие показатели** - переменные, относящиеся к общему игровому опыту

- например, была ли игра установлена ранее игроком или авторизовался ли игрок через сторонние сервисы (Facebook или Google Play);

**Показатели вовлеченности** - переменные, связанные с активным участием игрока в игре

- например, количество стартов уровней и время, проведенное в игре;

**Показатели эффективности прохождения уровней** – переменные, относящиеся к игровому прогрессу и используемым стратегиям прохождения уровней

- например, доля побед или использование бустеров на уровнях;

**Социальные показатели** – переменные, относящихся к социальному взаимодействию

- например, кол-во друзей в игре или вступил ли он в гильдию;

**Технические показатели** – технические настройки в игре

- например, процент входов в игру, которые сопровождались сбоем игры или размер экрана устройства игрока

Всего 63 факторных признака

# Факторные признаки

**Общие показатели** - переменные, относящиеся к общему игровому опыту

- например, была ли игра установлена ранее игроком или авторизовался ли игрок через сторонние сервисы (Facebook или Google Play);

**Показатели вовлеченности** - переменные, связанные с активным участием игрока в игре

- например, количество стартов уровней и время, проведенное в игре;

**Показатели эффективности прохождения уровней** – переменные, относящиеся к игровому прогрессу и используемым стратегиям прохождения уровней

- например, доля побед или использование бустеров на уровнях;

**Социальные показатели** – переменные, относящихся к социальному взаимодействию

- например, кол-во друзей в игре или вступил ли он гильдию;

**Технические показатели** – технические настройки в игре

- например, процент входов в игру, которые сопровождались сбоем игры или размер экрана устройства игрока

Всего 63 факторных признака

# Факторные признаки

**Общие показатели** - переменные, относящиеся к общему игровому опыту

- например, была ли игра установлена ранее игроком или авторизовался ли игрок через сторонние сервисы (Facebook или Google Play);

**Показатели вовлеченности** - переменные, связанные с активным участием игрока в игре

- например, количество стартов уровней и время, проведенное в игре;

**Показатели эффективности прохождения уровней** – переменные, относящиеся к игровому прогрессу и используемым стратегиям прохождения уровней

- например, доля побед или использование бустеров на уровнях;

**Социальные показатели** – переменные, относящихся к социальному взаимодействию

- например, кол-во друзей в игре или вступил ли он гильдию;

**Технические показатели** – технические настройки в игре

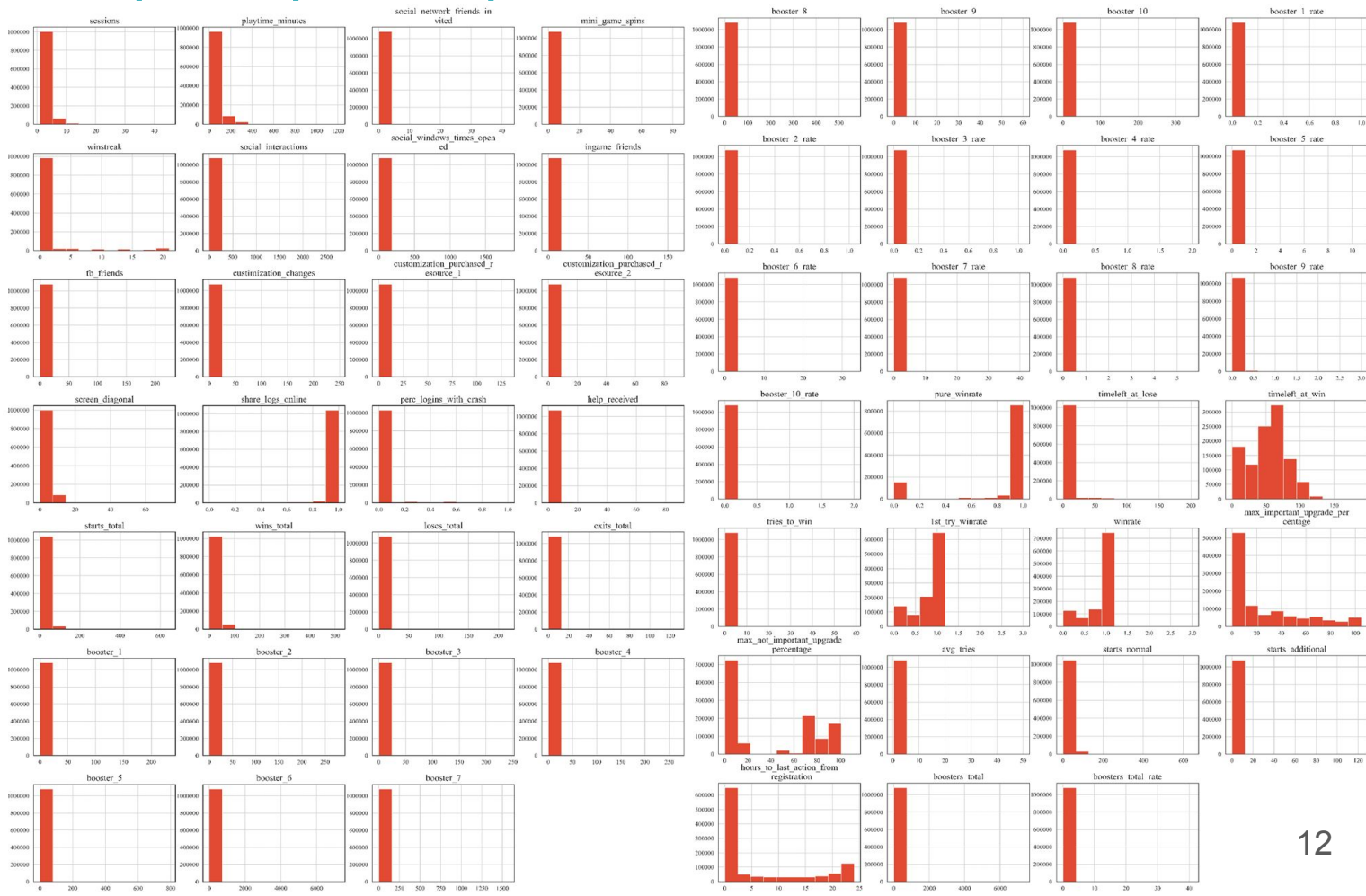
- например, процент входов в игру, которые сопровождались сбоем игры или размер экрана устройства игрока

Всего 63 факторных признака

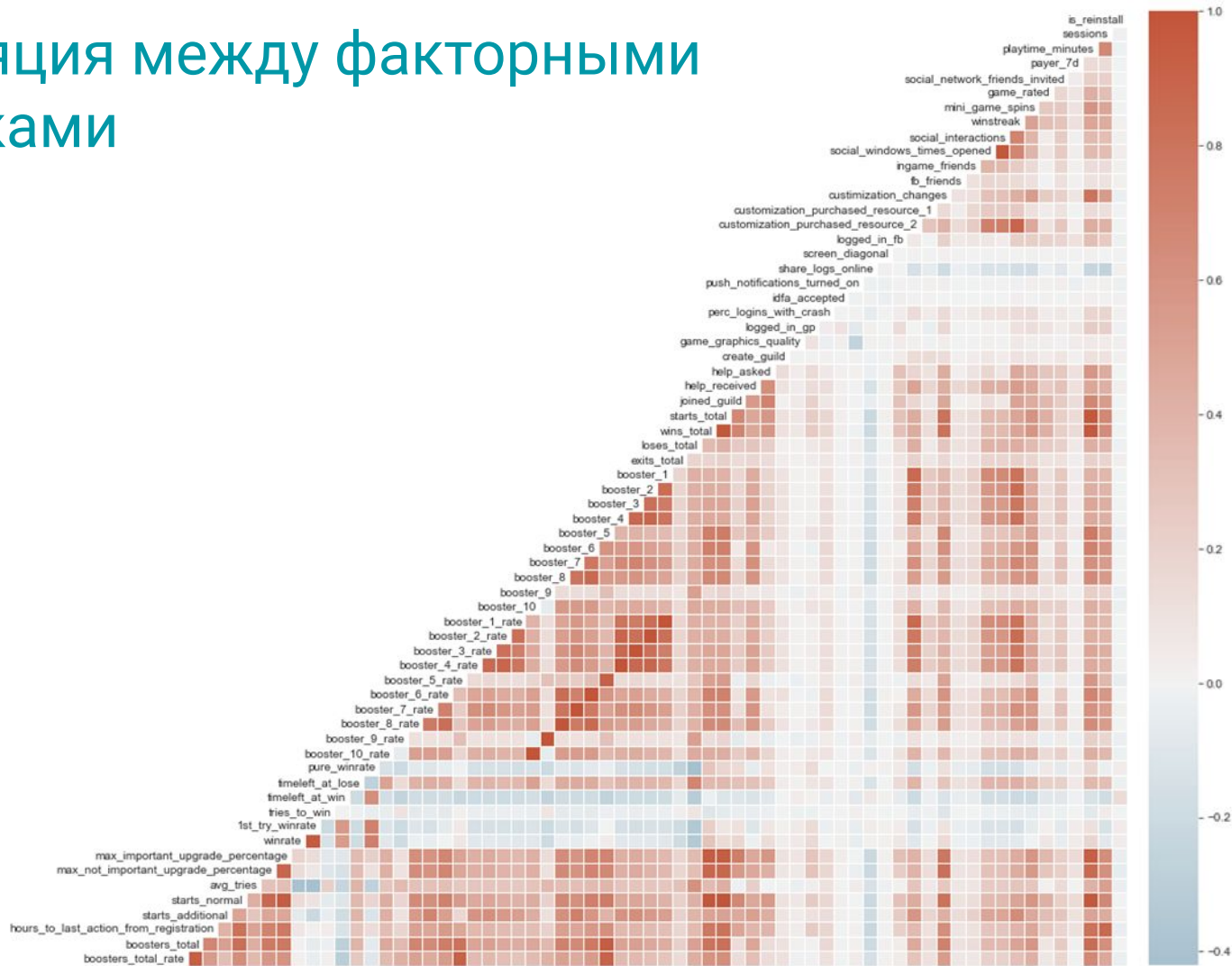
# Распределение факторных признаков

1. Ненормальное распределение

2. Сильная позитивная скошенность (positive skew)



# Корреляция между факторными признаками



# Применимость факторного анализа

**Критерий сферичности Бартлетта** ( $\chi^2 = 127814017.68$ ,  $p = 0.00$ ) → можно применять

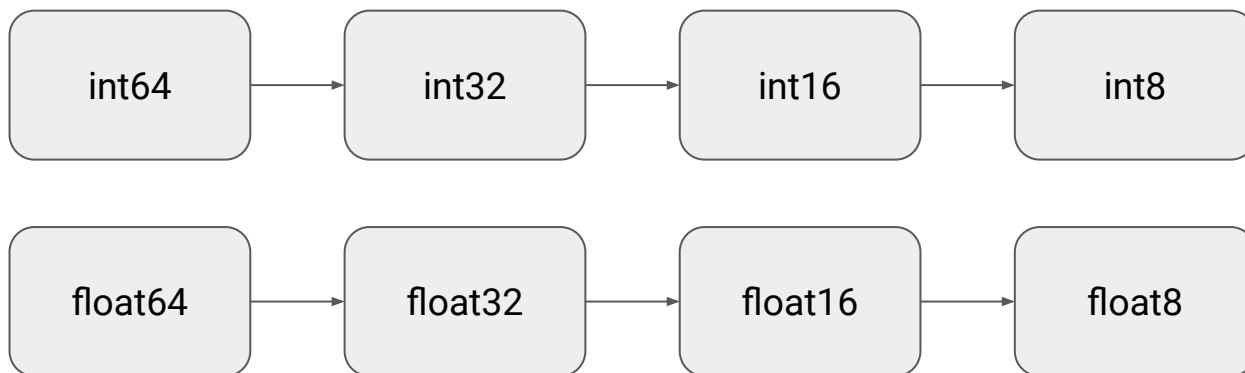
**Тест Кейзера-Мейера-Олкина** ( $KMO = 0.49$ ) → нет смысла применять

**Решение:** проводить обучение моделей на исходных данных

# Обработка данных

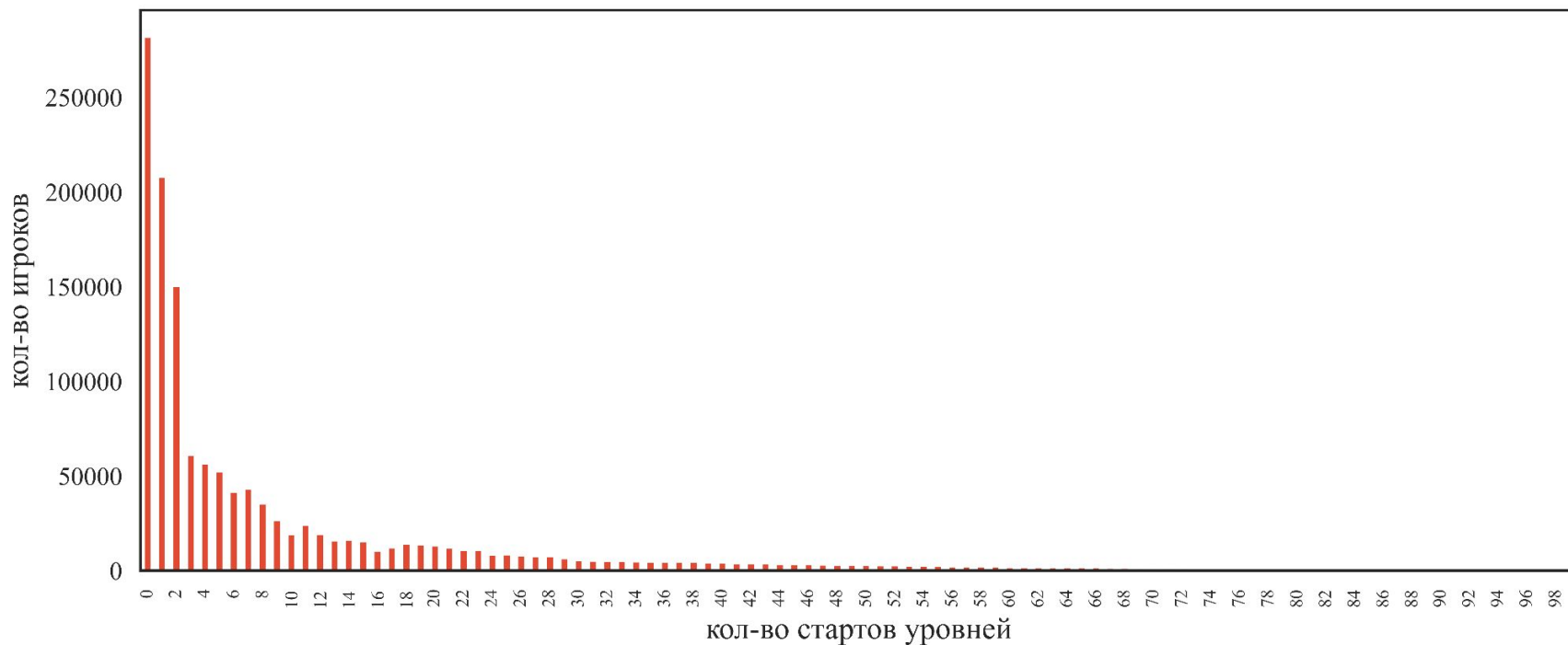
- **Нисходящее преобразование (downcasting) факторных признаков**
  - 670 MB → 189 MB занимаемой таблицей памяти

Пример работы нисходящего преобразования:



# Обработка данных

- **Убраны игроки, не прошедшие ни одного уровня в игре (20% игроков)**
  - Убраны пропущенные значения в факторных признаках
  - Затронуло всего 0.01% платящих пользователей



# Метрика оценки качества модели классификации

Пусть:

- 450 руб. - доход с платящего игрока, если не показываем ему рекламу
- 10% - ожидаемая просадка в доходе платящего из-за рекламы
- 1 руб. - доход с рекламы с одного игрока

Прогнозная модель будет использоваться для принятия решения о показе рекламы

Тогда ожидаемый доход от одного игрока считается по матрице ошибок:

		Реальное значение	
		Не заплатит	Заплатит
Предсказанное значение	Не заплатит	+ 1 р.	+ 406 р.
	Заплатит	0 р.	+ 450 р.

Далее считаем среднее значение ожидаемого дохода на одного игрока

# Критерии отбора моделей классификации

- Есть реализация модели в библиотеке sklearn
- Работает с задачей классификации для двух классов
  - Не подходят: `OneVsOneClassifier()`, `OneVsRestClassifier()`, `OutputCodeClassifier()`, `ClassifierChain()`, `MultiOutputClassifier()`
- Не требует дополнительной настройки
  - Не подходят: `VotingClassifier()`, `StackingClassifier()`
- Не слишком ресурсоемкая - ограничение в 51 GB оперативной памяти в Google Colab Pro+
  - Не подходят: `GaussianProcessClassifier()`
- Обучается не более чем за 12 часов работы:
  - Не подходят: `NuSVC()`

# Критерии отбора моделей классификации

- Есть реализация модели в библиотеке sklearn
- Работает с задачей классификации для двух классов
  - Не подходят: `OneVsOneClassifier()`, `OneVsRestClassifier()`, `OutputCodeClassifier()`, `ClassifierChain()`, `MultiOutputClassifier()`
- Не требует дополнительной настройки
  - Не подходят: `VotingClassifier()`, `StackingClassifier()`
- Не слишком ресурсоемкая - ограничение в 51 GB оперативной памяти в Google Colab Pro+
  - Не подходят: `GaussianProcessClassifier()`
- Обучается не более чем за 12 часов работы:
  - Не подходят: `NuSVC()`

# Критерии отбора моделей классификации

- Есть реализация модели в библиотеке sklearn
- Работает с задачей классификации для двух классов
  - Не подходят: `OneVsOneClassifier()`, `OneVsRestClassifier()`, `OutputCodeClassifier()`, `ClassifierChain()`, `MultiOutputClassifier()`
- Не требует дополнительной настройки
  - Не подходят: `VotingClassifier()`, `StackingClassifier()`
- Не слишком ресурсоемкая - ограничение в 51 GB оперативной памяти в Google Colab Pro+
- Обучается не более чем за 12 часов работы:
  - Не подходят: `NuSVC()`

# Критерии отбора моделей классификации

- Есть реализация модели в библиотеке sklearn
- Работает с задачей классификации для двух классов
  - Не подходят: `OneVsOneClassifier()`, `OneVsRestClassifier()`, `OutputCodeClassifier()`, `ClassifierChain()`, `MultiOutputClassifier()`
- Не требует дополнительной настройки
  - Не подходят: `VotingClassifier()`, `StackingClassifier()`
- Не слишком ресурсоемкая - ограничение в 51 GB оперативной памяти в Google Colab Pro+
  - Не подходят: `GaussianProcessClassifier()`
- Обучается не более чем за 12 часов работы:
  - Не подходят: `NuSVC()`

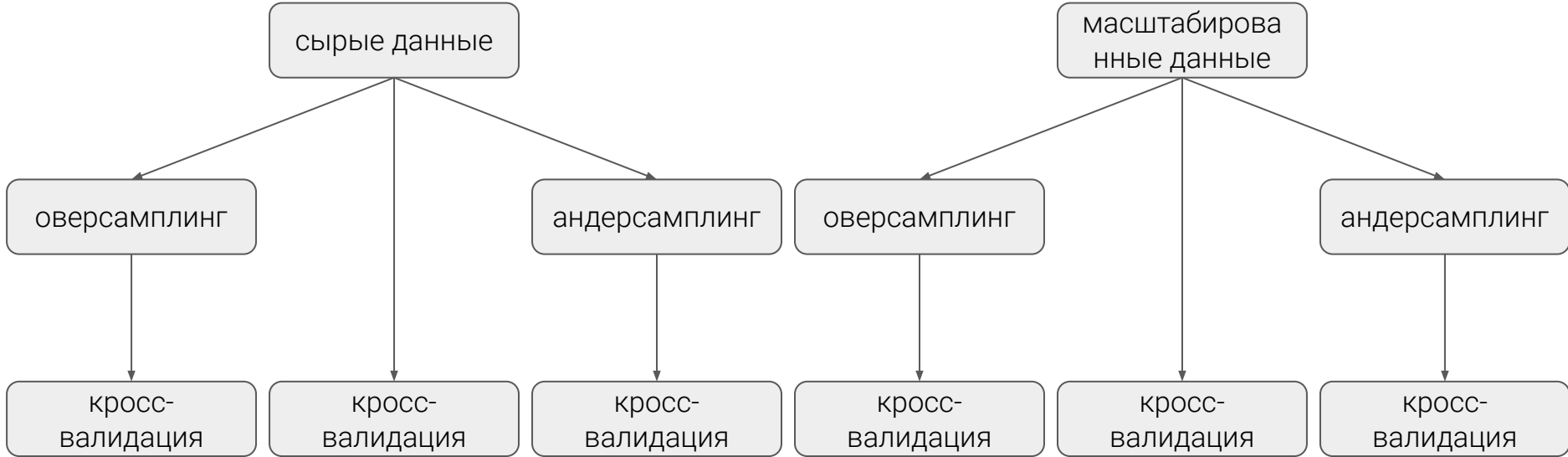
# Критерии отбора моделей классификации

- Есть реализация модели в библиотеке sklearn
- Работает с задачей классификации для двух классов
  - Не подходят: `OneVsOneClassifier()`, `OneVsRestClassifier()`, `OutputCodeClassifier()`, `ClassifierChain()`, `MultiOutputClassifier()`
- Не требует дополнительной настройки
  - Не подходят: `VotingClassifier()`, `StackingClassifier()`
- Не слишком ресурсоемкая - ограничение в 51 GB оперативной памяти в Google Colab Pro+
  - Не подходят: `GaussianProcessClassifier()`
- Обучается не более чем за 12 часов работы:
  - Не подходят: `NuSVC()`

# Итоговый список моделей, прошедших отбор

- **Ансамбли деревьев:** Бутстрэп-агрегирование (бэггинг), Дополнительные деревья, Случайный лес, Градиентный бустинг, Градиентный бустинг с применением гистограмм, Адаптивный бустинг (AdaBoost)
- **Базовый (dummy):** Базовый классификатор, предсказывающий преобладающий класс
- **Деревья решений:** Дерево решений, Классификатор дополнительных деревьев
- **Дискриминантный анализ:** Линейный дискриминантный анализ, Квадратичный дискриминантный анализ
- **Калибровка:** Логистическая регрессия с калибровкой и с кросс-валидацией
- **Линейные модели:** Логистическая регрессия, Логистическая регрессия с кросс-валидацией, Пассивно-агрессивный классификатор, Персептрон, Логистическая регрессия с регуляризацией Ridge, Логистическая регрессия с регуляризацией Ridge и кросс-валидацией, Стохастический градиентный спуск
- **Наивный Байес:** Бернулли наивный Байес, Дополнительный алгоритм наивного байеса, Мультиномиальный наивный Байес
- **Нейронные сети:** Многослойный персептрон
- **Опорные вектора:** Линейный метод опорных векторов, Метод опорных векторов
- **Ближайшие соседи:** Подбор ближайших K соседей, Метод ближайшего центроида

# Виды моделей



# Топ-5 моделей по ожидаемому доходу

Модель	Преобразование данных	Время обучения модели (сек.)	Качество модели на тестовых данных				Ожидаемый доход
			Доля правильных ответов (accuracy)	F-мера (F-score)	Полнота (recall)	Точность (precision)	
<b>Градиентный бустинг</b>	<b>оверсэмплинг</b>	<b>98.71</b>	<b>0.82</b>	<b>0.08</b>	<b>0.79</b>	<b>0.05</b>	<b>5.493</b>
Логистическая регрессия	масштабирование и оверсэмплинг	3.01	0.81	0.08	0.80	0.04	5.490
Логистическая регрессия с калибровкой и подбором гиперпараметров с кросс-валидацией	оверсэмплинг	358.11	0.81	0.08	0.80	0.04	5.489
Градиентный бустинг	масштабирование и оверсэмплинг	93.06	0.80	0.08	0.82	0.04	5.487
Логистическая регрессия с подбором гиперпараметров с кросс-валидацией	масштабирование и оверсэмплинг	91.24	0.80	0.08	0.81	0.04	5.487

# Топ моделей по остальным метрикам качества

Метрика	Модель	Преобразование данных	Время обучения модели (сек.)	Качество модели на тестовых данных				
				Доля правильных ответов (accuracy)	F-мера (F-score)	Полнота (recall)	Точность (precision)	Ожидаемый доход
Ожидаемый доход	Градиентный бустинг	оверсэмплинг	98.71	0.82	0.08	0.79	0.05	5.493
Доля правильных ответов (accuracy)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Точность (precision)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Полнота (recall)	Квадратичный дискриминантный анализ	масштабирование и оверсэмплинг	0.74	0.29	0.03	0.98	0.01	5.050
F-мера (F-score)	Линейный дискриминантный анализ	сырые данные	0.93	0.97	0.14	0.19	0.11	5.376

# Топ моделей по остальным метрикам качества

Метрика	Модель	Преобразование данных	Время обучения модели (сек.)	Качество модели на тестовых данных				
				Доля правильных ответов (accuracy)	F-мера (F-score)	Полнота (recall)	Точность (precision)	Ожидаемый доход
Ожидаемый доход	Градиентный бустинг	оверсэмплинг	98.71	0.82	0.08	0.79	0.05	5.493
Доля правильных ответов (accuracy)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Точность (precision)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Полнота (recall)	Квадратичный дискриминантный анализ	масштабирование и оверсэмплинг	0.74	0.29	0.03	0.98	0.01	5.050
F-мера (F-score)	Линейный дискриминантный анализ	сырые данные	0.93	0.97	0.14	0.19	0.11	5.376

# Топ моделей по остальным метрикам качества

Метрика	Модель	Преобразование данных	Время обучения модели (сек.)	Качество модели на тестовых данных				
				Доля правильных ответов (accuracy)	F-мера (F-score)	Полнота (recall)	Точность (precision)	Ожидаемый доход
Ожидаемый доход	Градиентный бустинг	оверсэмплинг	98.71	0.82	0.08	0.79	0.05	5.493
Доля правильных ответов (accuracy)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Точность (precision)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Полнота (recall)	Квадратичный дискриминантный анализ	масштабирование и оверсэмплинг	0.74	0.29	0.03	0.98	0.01	5.050
F-мера (F-score)	Линейный дискриминантный анализ	сырые данные	0.93	0.97	0.14	0.19	0.11	5.376

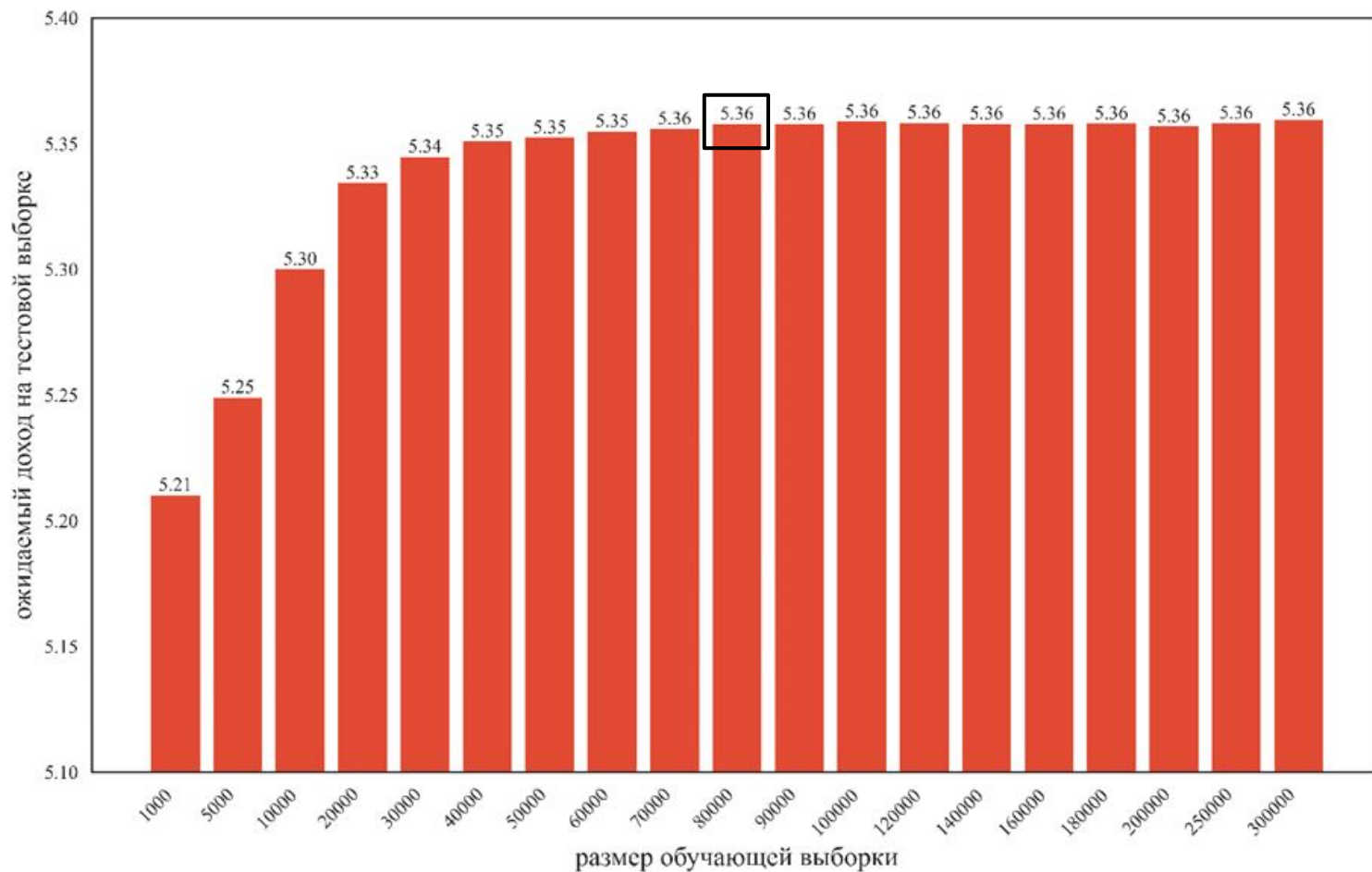
# Топ моделей по остальным метрикам качества

Метрика	Модель	Преобразование данных	Время обучения модели (сек.)	Качество модели на тестовых данных				
				Доля правильных ответов (accuracy)	F-мера (F-score)	Полнота (recall)	Точность (precision)	Ожидаемый доход
Ожидаемый доход	Градиентный бустинг	оверсэмплинг	98.71	0.82	0.08	0.79	0.05	5.493
Доля правильных ответов (accuracy)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Точность (precision)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Полнота (recall)	Квадратичный дискриминантный анализ	масштабирование и оверсэмплинг	0.74	0.29	0.03	0.98	0.01	5.050
F-мера (F-score)	Линейный дискриминантный анализ	сырые данные	0.93	0.97	0.14	0.19	0.11	5.376

# Топ моделей по остальным метрикам качества

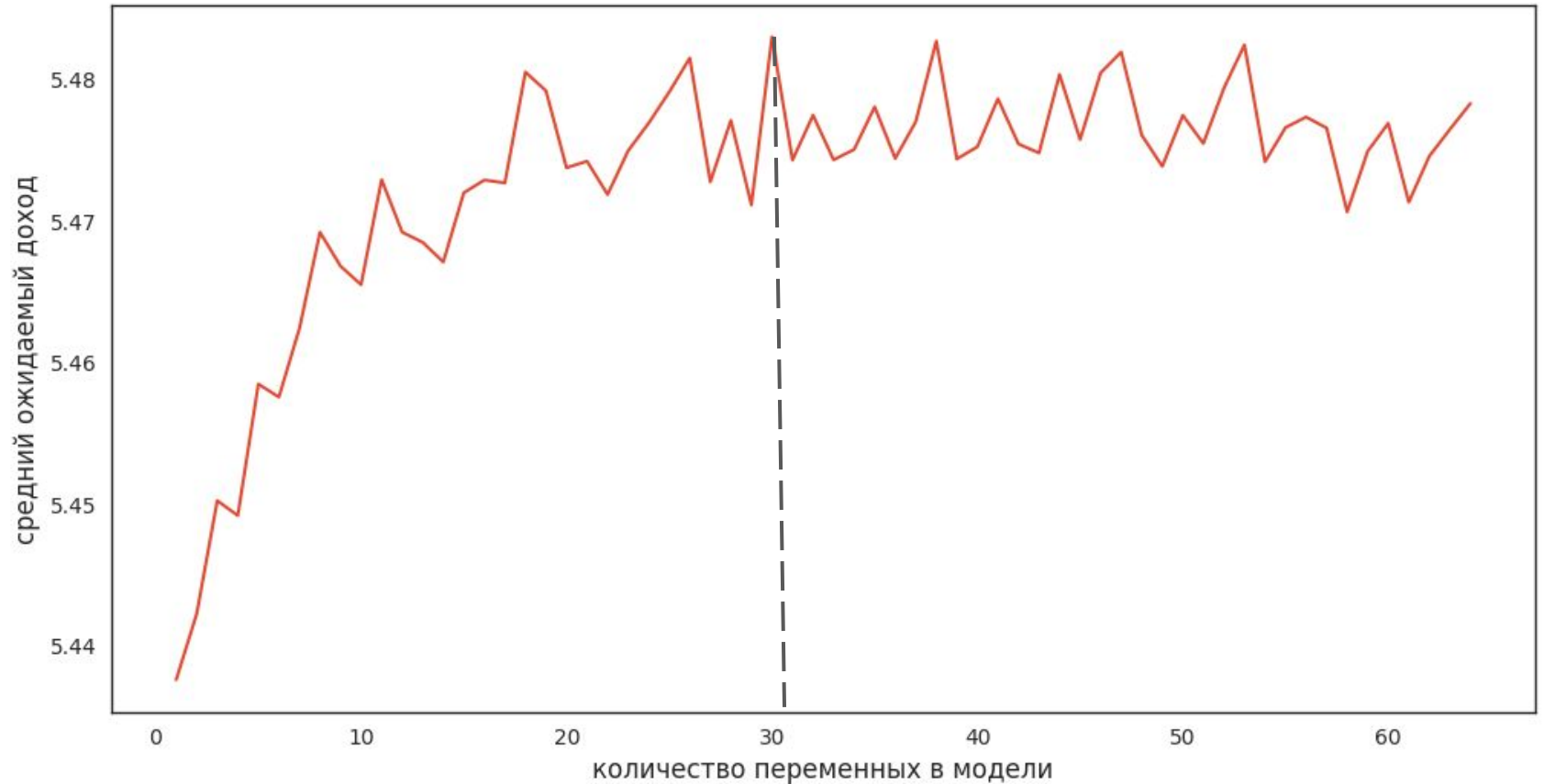
Метрика	Модель	Преобразование данных	Время обучения модели (сек.)	Качество модели на тестовых данных				
				Доля правильных ответов (accuracy)	F-мера (F-score)	Полнота (recall)	Точность (precision)	Ожидаемый доход
Ожидаемый доход	Градиентный бустинг	оверсэмплинг	98.71	0.82	0.08	0.79	0.05	5.493
Доля правильных ответов (accuracy)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Точность (precision)	Адаптивный бустинг (AdaBoost)	сырые данные	8.57	0.99	0.00	0.00	0.20	5.306
Полнота (recall)	Квадратичный дискриминантный анализ	масштабирование и оверсэмплинг	0.74	0.29	0.03	0.98	0.01	5.050
F-мера (F-score)	Линейный дискриминантный анализ	сырые данные	0.93	0.97	0.14	0.19	0.11	5.376

# Градиентный бустинг: размер обучающей выборки



# Градиентный бустинг: отбор переменных

Метод рекурсивного отбора признаков с кросс-валидацией (RFECV)



# Градиентный бустинг: подбор гиперпараметров

- функция потерь, которую нужно минимизировать (**loss**): 'log\_loss',
- коэффициент скорости обучения (**learning\_rate**): 0.5,
- минимальное количество наблюдений для разбивки узла дерева решений (**min\_samples\_split**): 6,
- минимальное количество наблюдений в листе дерева (**min\_samples\_leaf**): 3,
- максимальная глубина деревьев (**max\_depth**): 8,
- максимальное количество признаков, учитываемых при поиске лучшего разделения (**max\_features**): 8,
- максимальное количество листьев в дереве (**max\_leaf\_nodes**): None,
- критерий выбора разделения в узле (**criterion**): 'squared\_error',
- минимальная доля наблюдений, которая должна быть в листе после разбивки от всех наблюдений до разбивки (**min\_weight\_fraction\_leaf**): 0.0,
- доля выборки, используемая для обучения одной отдельной базовой модели внутри ансамбля (**subsample**): 1.0,
- количество деревьев в ансамбле (**n\_estimators**): 200,
- каким способом делается первое предсказание класса, которое после будет улучшать бустинг (**init**): 'zero',
- использовать ли значения ранее обученной модели (**warm\_start**): True,
- минимальное уменьшение индекса Джини (impurity index), необходимое для разбивки (**min\_impurity\_decrease**): 0

# Настройка градиентного бустинга: итог

	Ожидаемый доход	Прирост по сравнению с предыдущим вариантом
1. Игрокам не показывается реклама	4.783	
2. Реклама показывается всем (базовая модель)	5.305	+10,9%
3. Используется базовая версия <b>градиентного бустинга</b> (GradientBoostingClassifier) для прогноза платящего	<b>5.472</b>	<b>+3,1%</b>
4. Используется <b>градиентный бустинг</b> с отбором переменных и подбором гиперпараметров для прогноза платящего	<b>5.481</b>	<b>+0,2%</b>
5. Идеальная модель, если бы все предсказания делались корректно	5.772	+5,3%

# Настройка градиентного бустинга: итог

	Ожидаемый доход	Прирост по сравнению с предыдущим вариантом
1. Игрокам не показывается реклама	4.783	
2. Реклама показывается всем (базовая модель)	5.305	+10,9%
3. Используется базовая версия <b>градиентного бустинга</b> (GradientBoostingClassifier) для прогноза платящего	<b>5.472</b>	<b>+3,1%</b>
4. Используется <b>градиентный бустинг</b> с отбором переменных и подбором гиперпараметров для прогноза платящего	<b>5.481</b>	<b>+0,2%</b>
5. Идеальная модель, если бы все предсказания делались корректно	5.772	+5,3%

# Настройка градиентного бустинга: итог

	Ожидаемый доход	Прирост по сравнению с предыдущим вариантом
1. Игрокам не показывается реклама	4.783	
2. Реклама показывается всем (базовая модель)	5.305	+10,9%
3. Используется базовая версия <b>градиентного бустинга</b> (GradientBoostingClassifier) для прогноза платящего	<b>5.472</b>	<b>+3,1%</b>
4. Используется <b>градиентный бустинг</b> с отбором переменных и подбором гиперпараметров для прогноза платящего	<b>5.481</b>	<b>+0,2%</b>
5. Идеальная модель, если бы все предсказания делались корректно	5.772	+5,3%

# Настройка градиентного бустинга: итог

	Ожидаемый доход	Прирост по сравнению с предыдущим вариантом
1. Игрокам не показывается реклама	4.783	
2. Реклама показывается всем (базовая модель)	5.305	+10,9%
3. Используется базовая версия <b>градиентного бустинга</b> (GradientBoostingClassifier) для прогноза платящего	<b>5.472</b>	<b>+3,1%</b>
4. Используется <b>градиентный бустинг</b> с отбором переменных и подбором гиперпараметров для прогноза платящего	<b>5.481</b>	<b>+0,2%</b>
5. Идеальная модель, если бы все предсказания делались корректно	5.772	+5,3%

# Настройка градиентного бустинга: итог

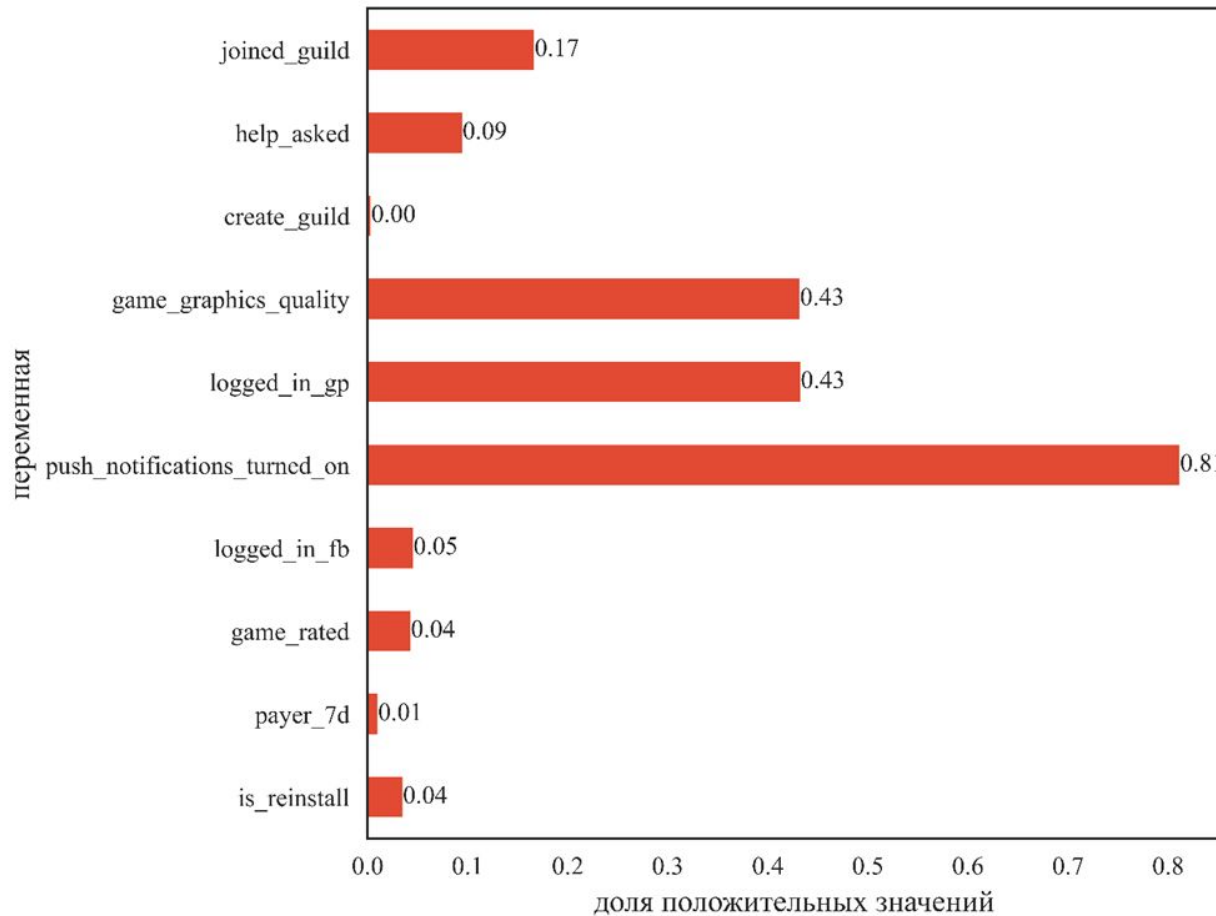
	Ожидаемый доход	Прирост по сравнению с предыдущим вариантом
1. Игрокам не показывается реклама	4.783	
2. Реклама показывается всем (базовая модель)	5.305	+10,9%
3. Используется базовая версия <b>градиентного бустинга</b> (GradientBoostingClassifier) для прогноза платящего	<b>5.472</b>	<b>+3,1%</b>
4. Используется <b>градиентный бустинг</b> с отбором переменных и подбором гиперпараметров для прогноза платящего	<b>5.481</b>	<b>+0,2%</b>
5. Идеальная модель, если бы все предсказания делались корректно	5.772	+5,3%

# Выводы

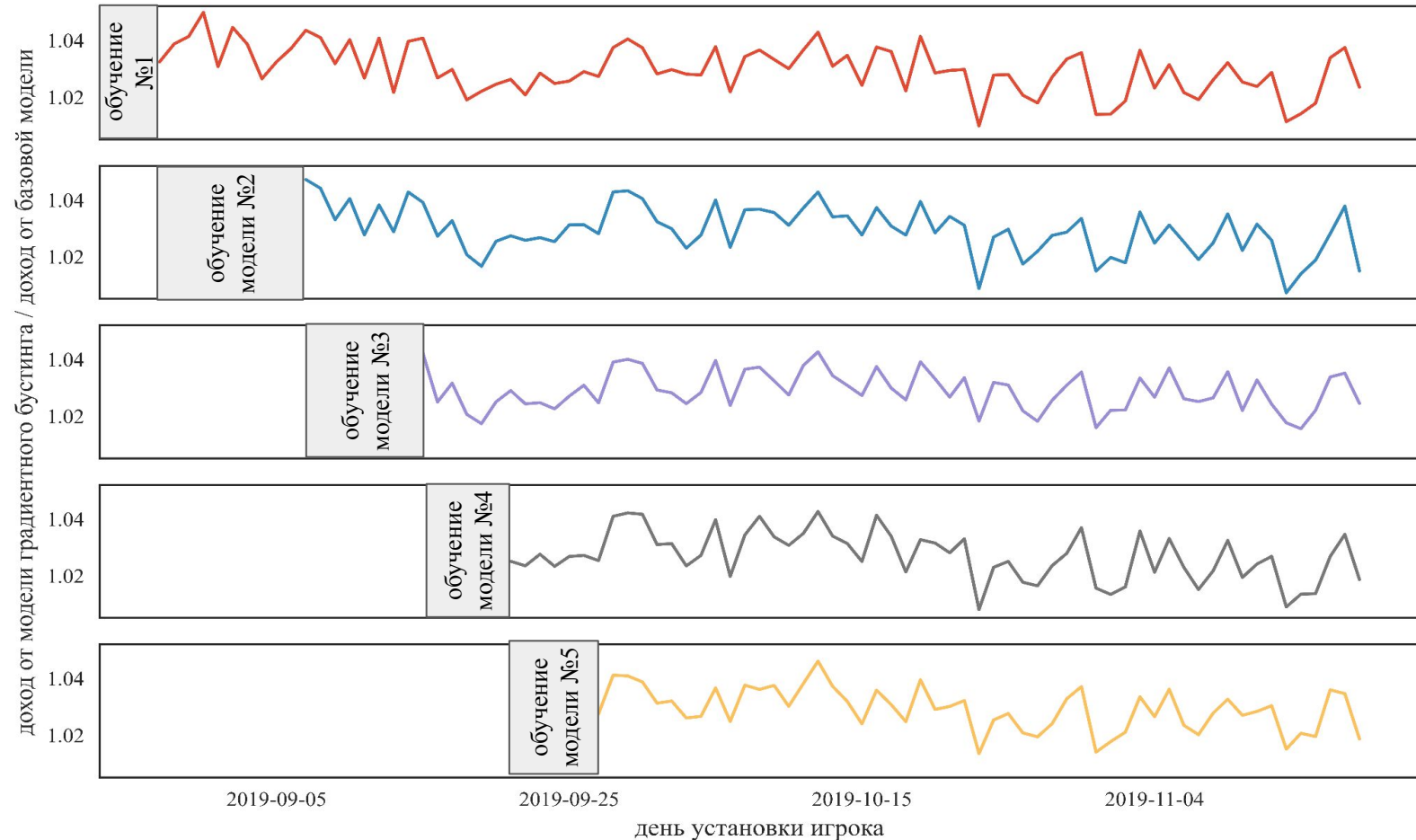
- Собраны и обработаны данные об игровом поведении за первый день жизни и факта платежа к 7 дню жизни
- Разработана метрика для оценки качества прогнозной модели на основе ожидаемого дохода
- Наилучшая прогнозная модель - Градиентный Бустинг (GradientBoostingClassifier)
- Модель дает +3% дохода к базовой модели
- Настройка модели дает дополнительные 0.2% дохода
- Предлагается настроить систему уведомлений для определения момента, когда качество модели начнет ухудшаться, чтобы можно было оперативно отключить модель
- На основе модели градиентного бустинга предполагается в дальнейшем принимать решение о показе игроку рекламы для максимизации прибыли



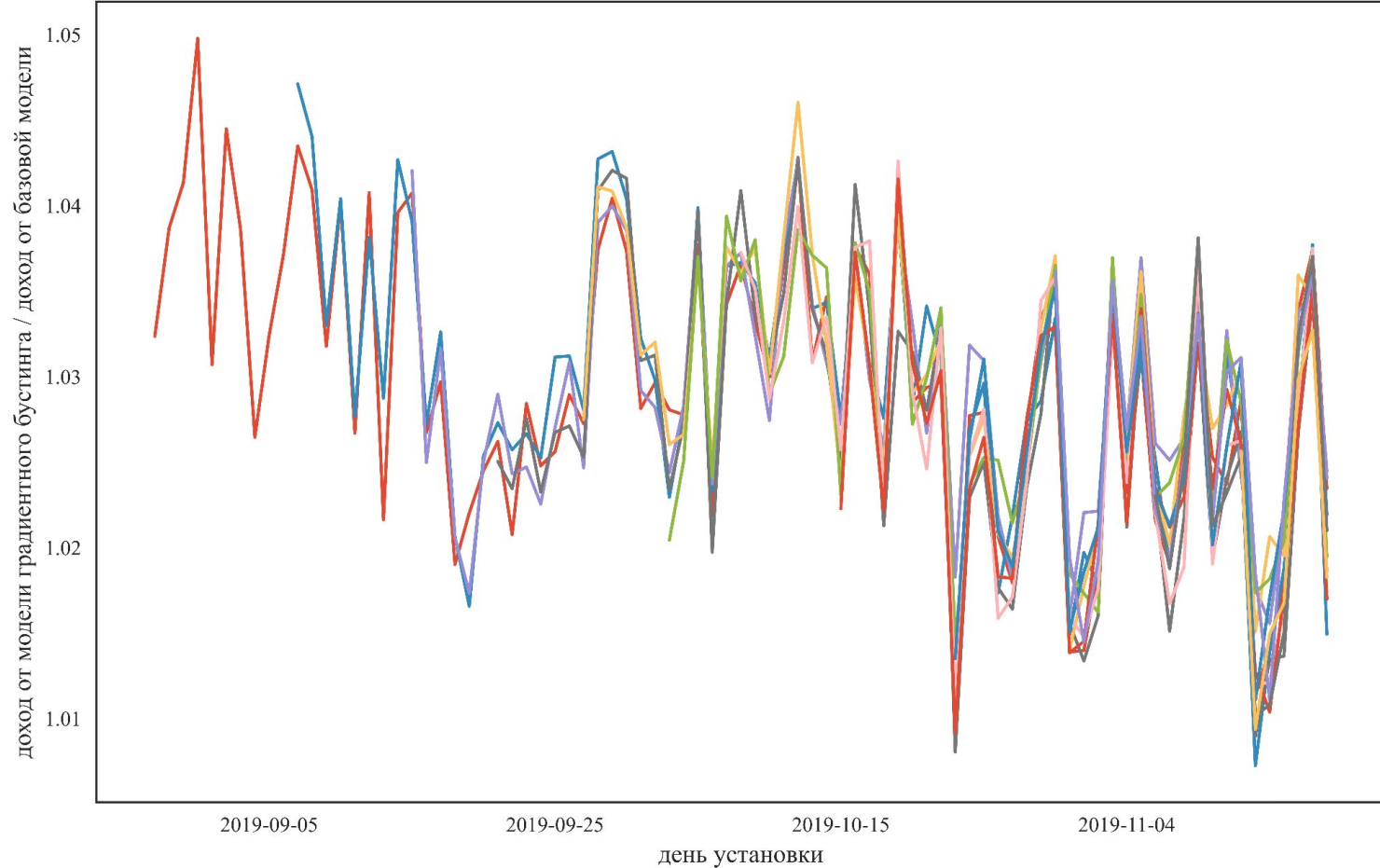
# Распределение бинарных переменных



# Как часто нужно будет переобучать модель



# Как часто нужно будет переобучать модель



# Формулы корреляции

Коэффициент корреляции Спирмана

$$\rho = 1 - \frac{6 \sum d_i^2}{n(n^2 - 1)}$$

$n$  - объем выборки

$d$  - разность между рангами по двум переменным для каждого испытуемого

Точечно-бисериальный коэффициент корреляции (Point-Biserial)

$$r_{pb} = \frac{X_1 - X_0}{s_n} \sqrt{\frac{n_1 n_0}{n^2}}$$

$X_1$  и  $X_0$  - средние значения численной переменной  $X$  со значением 1 или 0 у бинарной переменной  $Y$

$s_n$  - стандартное отклонение численной переменной,

$n_1$  и  $n_0$  - количество значений  $X$  с 1 или 0 по  $Y$ ,

$n$  - общий размер выборки

Коэффициент корреляции Фи (Phi коэффициент)

$$\phi = \frac{n_{11} * n_{00} - n_{10} * n_{01}}{\sqrt{n_{1o} * n_{o1} * n_{o0} * n_{0o}}}$$

	<b>Y = 1</b>	<b>Y = 0</b>	<b>Total</b>
X = 1	$n_{11}$	$n_{10}$	$n_{1o}$
X = 0	$n_{01}$	$n_{00}$	$n_{0o}$
Total	$n_{o1}$	$n_{o0}$	$n$

# Нисходящее преобразование данных (downcasting)

Пример: преобразования столбца со значениями от 0 до 100 из типа данных int64 в тип uint8

int64 - целые числа (integer) от -9223372036854775808 до 9223372036854775807

uint8 - беззнаковые целые числа (unsigned integer) от 0 до 255