Правительство Российской Федерации

Федеральное государственное автономное образовательное учреждение высшего образования

Национальный исследовательский университет

«Высшая школа экономики»

Проект по машинному обучению в экономике:

«Влияние рекламной интеграции на количество минут просмотра видео»

Выполнили:

Алексанян Алек Давидович БЭК213

Григорьева Наталья Алексеевна БЭК216

Дмитриева Полина Дмитриевна БЭК216

Семинарист:

Погорелова Полина Вячеславовна

Москва, 2024

1. Обоснование темы

1.1. Придумайте непрерывную зависимую (целевую) переменную (например, заработная плата или прибыль) и бинарную переменную воздействия (например, образование или факт занятий спортом).

В данном исследовании изучается, насколько показ рекламы в случайном моменте в видео снижает желание смотреть это видео. Для этого используются следующие целевая переменная и переменная воздействия.

Целевая переменная: $Minutes_i$ - Количество минут, проведенных за просмотром видео .
Переменная воздействия: $Advert_i$ - Просмотр рекламной интеграции в случайном моменте в видео (1 - был просмотр, 0 - нет просмотра).

1.2. Опишите, для чего может быть полезно изучение влияния переменной воздействия на зависимую переменную. В частности, укажите, как эта информация может быть использована бизнесом или государственными органами.

Изучение представленной темы может быть интересна в первую бизнесу, особенно, видеохостингам и стриминговым сервисам по типу YouTube, Netflix, Okko и иным сервисам, размешающим видео. С помощью такого исследования можно снижение интереса к видео при рекламной интеграции в ней. Если будет выявлено значимое снижение продолжительности просмотра видео из-за просмотра рекламы, то стриминговым сервисам не следует делать длинные рекламной интеграции в видео, так как "убивает" интерес к видео и, тем самым, снижает лояльность пользователя к сервису. В ином случае, если количество минут просмотренного видео не снижается из-за просмотра рекламы, то компания может смело увеличивать количество рекламы в своём видео без потери пользователей.

1.3. Обоснуйте наличие причинно-следственной связи между зависимой переменной и переменной воздействия. Приведите не менее 2-х источников из научной литературы, подкрепляющих ваши предположения.

В работе анализируется причинно-следственная связь между просмотром рекламы и дальнейшим поведением пользователя, вызванное соответственно этим самым просмотром. Данная связь очевидно вытекает из обычной практики взаимодействия среднестатического пользователя с рекламой: реклама сильно надоедает клиента и забирает у него слишком много времени, и поэтому просмотр рекламу влияет на поведение пользователя. В случае с видеохостингами и стриминговыми сервисами негативное изменение поведение пользователя, в первую очередь, означает снижение количества просматриваемого видео.

1.4.Кратко опишите результаты предшествовавших исследований по схожей тематике и критически оцените методологию этих работ с точки зрения гибкости (жесткости предпосылок) использовавшихся методов эконометрического анализа.

1.5. Придумайте хотя бы 3 контрольные переменные, по крайней мере одна из которых должна быть бинарной и хотя бы одна – непрерывной. Кратко обоснуйте выбор каждой из них.

Контрольные переменные:

1) Age_i - Возраст пользователя (непрерывная переменная (количество полных лет)). Возраст влияет на отношение человека к рекламе и к количеству просматриваемого контента. Есть

предположение, что с возрастом пользователи становятся все более терпимы к рекламе, что нивелирует снижение интереса к видео из-за просмотра рекламы

2) Job_i - Наличие работы (бинарная переменная (0 – нет работы, 1 – есть работа)). Наличие работы у индивида должно снижать его терпимость к рекламе из-за меньшего свободного времени и большей раздраженности.
3) $Children_i$ - Наличие детей (бинарная переменная (0 – нет детей, 1 – есть хотя бы 1 ребенок)). С одной стороны, родитель может дать свой телефон ребенку, которому в принципе все равно и на наличие рекламы, и на видео, что очевидно увеличивает количество просмотренных минут при просмотре рекламе. С другой стороны, если видео смотрит сам родитель, то здесь скорее всего сыграет большая раздраженность и усталость родителя, которые повлияют на его терпимость к рекламе.

1.6. Придумайте бинарную инструментальную переменную и обоснуйте, почему она удовлетворяет необходимым условиям.

Инструментальная переменная: $Block_i$ (1 - есть блокировщик рекламы на устройстве, 0 - нет блокировщика) - Наличие у пользователя блокировщика рекламы на телефоне. Данная инструментальная переменная удовлетворяет нужным условиям, так как сам факт наличия блокировщика на телефоне характеризует уровень терпимости пользователя к рекламе, то есть данная инструментальная переменная очевидна сильно связана с фактом просмотра рекламы индивидом. К тому же, сам факт наличия блокировщика никак не должен повлиять на количество минут просмотра видео, ведь блокировщик никак не влияет на интерес человека к видео.

Здесь нужно сделать примечание, по какому принципу работает блокировщик в сгенерированном мире. Перед показом рекламного видео он делает предупреждение о наличии рекламы в данном видео, а далее пользователь сам решает смотреть ли ему видео или же выключать из присутствия раздрающей рекламы в ней.

2. Генерация данных и предварительная обработка данных

2.1. Опишите математически предполагаемый вами процесс генерации данных.

- Возраст (Age_i)

  $Age_i \sim \mathrm{N}(45, 10^2)$

Генерировали как нормальное, также ограничили данную переменную сверху 75 годами и снизу 18 годами из разумных соображений.

- Наличие работы (Job_i)

  $Job_i \sim \mathrm{Ber}(0.7)$

Генерировали как распределение Бернулли, предположив, что у 70% пользователей есть работа.

- Наличие детей ($Children_i$)
  $Children_i \sim \mathrm{Ber}(0.8)$

Генерировали как распределение Бернулли. Предположительно с вероятностью 0,8 у человека, просматривающего видео, будут дети.

- Инструментальная переменная ($Block_i$)

Генерировали индекс для данной переменной, который выглядит следующим образом:

$$\mathrm{P}\left(\mathrm{Block}_i = 1 | \mathrm{Age}_i, \mathrm{Job}_i, \mathrm{Children}_i\right) = \Phi\left(\frac{2 * (\mathrm{Children}_i + \mathrm{Job}_i)}{(1 + \mathrm{Age}_i)} - 1.1 * \mathrm{Children}_i^3 * \mathrm{Age}_i^{\frac{1}{4}} + \mathrm{Job}_i \times \mathrm{Children}_i\right)$$

- Ненаблюдаемая переменная, порождающая эндогенность ($Interest_i$)

В качестве данной переменной мы решили использовать меру интереса к видео. Логика: если видео очень интересное то рекламы ты посмотришь, так как нужно видео смотреть. Ну и очевидно если видео очень интересное то и само видео ты больше дольше смотреть.

$Interest_i \sim$ T(10)

Предположили, что данная переменная имеет распределение Стьюдента (похоже на нормальное, но с более тяжелыми хвостами), поскольку в середине видео интерес к ролику наиболее велик, чем в начале или в конце просмотра.

- Переменная воздействия ($Advert_i$)

Сгенерировали часть индекса, независящую от наличия блокировщика. Затем симулировали факты просмотра рекламы в случае, когда у человека нет блокировщика и когда он есть. По нашему предположению, наличие блокировщика не может гарантировать отсутствие просмотра рекламной интеграции в видео (например, блокировщик может вылететь из-за какой-то ошибки и тогда уже сам человек решает смотреть ему рекламу или нет).

Также в этом пункте мы рассмотрели корреляции факта просмотра рекламной интеграции с мерой интереса в видео и наличием блокировщика и получили отрицательные значения. Это полностью соответствует логике: наличие блокировщика отрицательно сказывается на просмотре рекламы, а просмотр рекламы раздражает пользователя и ведет к снижению меры интереса к просматриваемому видео.

- Целевая переменная ($Minutes_i$)

Для генерации переменной, отвечающей за время просмотра видео, мы разбили ее на две составляющие. Первая часть определяется контрольными переменными (g_obs) и ненаблюдаемой мерой интереса к видео (g_unobs), а вторая часть случайными ошибками (error). Также составили разные функции для контрольных переменных и ненаблюдаемой переменной в двух случаях (когда человек посмотрел рекламу и когда не посмотрел), поскольку от самого факта просмотра/не просмотра рекламы может зависеть то, какой вклад переменные будут вносить в модель. Например, предположили, что контрольные переменные будут вносить больший вклад в случае, когда человек смотрит рекламу.

2.2. Кратко обоснуйте предполагаемые направления связей зависимой переменной и переменной воздействия с контрольными переменными.

Чтобы понять, какие можно построить зависимости, то нужно определять, что это за человек, который не смотрит рекламу. Мы полагаем, что рекламу смотрят менее раздраженные люди, которым все в целом интересно послушать. Например, взрослые люди без работы.

1) **Если клиент посмотрел рекламу**

Если человек смотрит рекламу, то он в целом просто хочет смотреть видео, так как ему очень интересно, тем самым, переменная, создающая эндогенность, должна иметь больший коэффициент при генерации количества часов просмотра видео. По контрольным переменным можно предположить, что они немного сильнее влияют, чем у тех, кто не посмотрел рекламу, так как пользователь не смотрит видео на репите, и все характеристики начинают иметь большую роль.

2) **Если клиент не посмотрел рекламу**

Сформированная нетерпимость к рекламе ведет к перенасыщению просмотром видео, то есть пользователь смотрит просто чтобы смотреть, отсюда коэффициент при эндогенной переменной конечно положительный, но меньше чем в первом случае. Аналогичная логика и для контрольных переменных, так как это все уже имеет меньшее значение, если ты очень часто смотришь видео.

2.3.Симулируйте данные в соответствии с предполагаемым вами процессом и приведите корреляционную матрицу, а также таблицу со следующими описательными статистиками:

• Для непрерывных переменных: выборочное среднее, выборочное стандартное отклонение, медиана, минимум и максимум.

• Для бинарных переменных: доля и количество единиц.

Рис.1 Корреляционная матрица переменных

	minutes	age
count	10000.000000	10000.000000
mean	36.937811	44.962100
std	14.530368	9.938729
min	0.000000	18.000000
25%	26.867179	38.000000
50%	36.240676	45.000000
75%	45.732478	52.000000
max	115.593779	75.000000

Таблица 1. Описательные статистики для непрерывных переменных

	Количество единиц	Доля единиц
advert	5207	0.5207
job	6983	0.6983
children	8009	0.8009
block	1273	0.1273

Таблица 2. Описательные статистики для бинарных переменных

2.4. Разделите выборку на обучающую и тестовую. Тестовая выборка должна включать от 20% до 30% наблюдений.

Мы разделили выборку в отношении 75%(train) на 25%(test).

3. Классификация

В каждом из заданий, если не сказано иного, необходимо использовать хотя бы 3 (на ваш выбор) из следующих методов: наивный Байесовский классификатор, метод ближайших соседей, случайный лес, градиентный бустинг и логистическая регрессия.

Мы выбрали следующие модели: логистическая регрессия, случайный лес, градиентный бустинг.

3.1. Отберите признаки, которые могут быть полезны при прогнозировании переменной воздействия и кратко обоснуйте выбор каждой из них. Не включайте в число этих признаков целевую переменную.

Для прогнозирования переменной воздействия $Advert_i$ мы отобрали следующие переменные:

$Age_i, Children_i, Job_i, Block_i$.

Age_i – молодые люди чаще обладают дофаминовой зависимостью от просмотра видео или сериалов, а потому менее терпимы к рекламе. К тому же люди взрослого возраста и пожилые люди зачастую хуже разбираются в интерфейсах приложений, в том числе и стриминговых сервисов, поэтому склонны смотреть рекламу, а не искать кнопку для того, чтобы ее пропустить или закрыть.

$Children_i$ – пользователи с детьми могут давать смотреть, например, мультики своим детям. Дети не всегда способны сами контролировать процесс просмотра рекламы, поэтому чаще всего будут ее смотреть.

Job_i – работающие люди располагают меньшим свободным временем, соответственно ценят его сильнее, чем безработные. Исходя из этого предположения, работающий человек более склонен пропускать рекламу, чем не работающий.

$Block_i$ – само наличие блокировщика рекламы свидетельствует о нежелании пользователя просматривать надоедливую рекламу, значит, с большой вероятностью, он и не будет ее смотреть.

3.2. Выберите произвольные значения гиперпараметров, а затем оцените и сравните (между методами) точность прогнозов:
• на обучающей выборке.
• на тестовой выборке.
• с помощью кросс-валидации (используйте только обучающую выборку).

	ACC train	ACC test	ACC cross val
Логистическая регрессия	0.804400	0.8200	0.803200
Случайный лес	0.806267	0.8184	0.801733
Градиентный бустинг	0.806267	0.8184	0.801200

Таблица 3. Классификация. Точность прогнозов на произвольных гиперпараметрах

Мы обучали модели на дефолтных параметрах, качество оценивали с помощью accuracy, то есть доли правильных предсказаний, и на кросс валидации на обучающей выборке, наиболее высокое качество показала модель Логистической регрессии, на обоих метриках, в то время как Градиентный бустинг и Случайный лес показали почти одинаковый результат, чуть менее высокий.

3.3. Для каждого метода с помощью кросс-валидации на обучающей выборке подберите оптимальные значения гиперпараметров (тюнинг). В качестве критерия качества используйте точность ACC. Результат представьте в форме таблицы, в которой для каждого метода должны быть указаны:
• изначальные и подобранные значения гиперпараметров.
• кросс-валидационная точность на обучающей выборке с исходными и подобранными значениями гиперпараметров.
• точность на тестовой выборке с исходными и подобранными значениями гиперпараметров.
Проинтерпретируйте полученные результаты и далее используйте методы с подобранными значениями гиперпараметров.

	default_lr	best_lr	default_rf	best_rf	default_gb	best_gb
learning_rate	-	-	-	-	0.1	0.3
max_depth	-	-	-	10	3.0	1.0
max_features	-	-	sqrt	sqrt	-	-
max_samples	-	-	-	600	-	-
n_estimator	-	-	-	-	100.0	10.0
n_estimators	-	-	100	100	-	-
solver	lbfgs	liblinear	-	-	-	-
Градиентный бустинг cv ACC	-	-	-	-	0.8012	0.8012
Градиентный бустинг тест ACC	-	-	-	-	0.8184	0.8208
Логистическая регрессия cv ACC	0.8032	0.803333	-	-	-	-
Логистическая регрессия тест ACC	0.82	0.8208	-	-	-	-
Случайный лес cv ACC	-	-	0.801733	0.802267	-	-
Случайный лес тест ACC	-	-	0.8184	0.8056	-	-

Таблица 4. Классификация. Сравнение моделей с произвольными и подобранными гиперпараметрами относительно accuracy

Изначально нами были использованы дефолтные значения гиперпараметров. Как мы можем видеть из таблицы, точность прогнозов у методов с подобранными гиперпараметрами выше практически во всех случаях. На таблице 4 представлены результаты.

Для Логистической регрессии мы подбирали параметр 'solver', то есть метод оптимизации, дефолтное значение гиперпараметра ‘lbfgs’, наиболее оптимальный для наших данных ‘liblinear’, который использует стохастический градиентный спуск и подходит для небольших объемов данных. После подбора качество незначительно выросло.

Для Случайного леса подбирались параметры ‘max_depth’ – глубина дерева, 'n_estimators’ – количество деревьев в ансамбле, 'max_samples’ – количество выборок бутстрапа для каждого базового алгоритма. После подбора параметров, дерево получилось достаточно глубокое, при том количестве признаков, которое у нас есть (4), обучался лес на 100 деревьях и для каждого дерева использовалось 600 выборок бутсрапа. Качество на кросс валидации с участием только тренировочных данных выросло, но на тесте упало, что может свидетельствовать о переобучении.

Для Градиентного бустинга подбирались параметры 'learning rate’- скорость сходимости, ‘max_depth’ – глубина и ‘n_estimator’ – количество базовых моделей. В результате подбора параметров Градиентного бустинга модель оказала, что лучшее качество достигается на более простых параметрах, то есть с более больших шагом обучения, меньшей глубиной и количеством базовых алгоритмов. Тем не менее, Градиентный бустинг показал меньшую точность по сравнению с логистической регрессией, на тесте обогнал случайный лес, но на кросс валидации показал результат хуже.

Таким образом, можно предварительно сделать вывод, что для наших данных подходят менее сложные модели.

Повышенная сложность: подберите на обучающей выборке оптимальные значения гиперпараметров случайного леса ориентируясь на значение OOB (out-of-bag) ошибки. Сопоставьте гиперпараметры и точность на тестовой выборке для случайного леса в зависимости от того, используется кросс-валидация или OOB ошибка. Объясните преимущество OOB ошибки по сравнению с кроссвалидацей.

	OOB	CV
max_depth	18.000000	10.000000
max_samples	500.000000	600.000000
n_estimators	100.000000	100.000000
Случайный лес тест ACC	0.805733	0.805600
Случайный лес cv ACC	0.801600	0.802267

Таблица 5. Классификация. Сравнение модели случайного леса с гиперпараметрами, подобранными OOB и кросс-валидацией

Можно заметить, на таблице 5 что качество на тестовой выборке выросло после подбора гиперпараметров методом OOB по сравнению с кросс- валидацией. Преимуществами out-of-bag метода считаются скорость, простота, а также оценка почти не смещена , а дисперсия достаточно низкая за счет усреднения по множеству моделей.

3.4 Повторите предыдущий пункт, используя любой альтернативный критерий качества модели. Обоснуйте возможные преимущества и недостатки этого альтернативного критерия.

	default_lr	best_lr	default_rf	best_rf	default_gb	best_gb
learning_rate	-	-	-	-	0.1	0.3
max_depth	-	-	-	12.0	3.0	1.0
max_samples	-	-	-	400.0	-	-
n_estimator	-	-	-	-	100.0	10.0
n_estimators	-	-	100.0	100.0	-	-
solver	lbfgs	liblinear	-	-	-	-
Градиентный бустинг cv f1	-	-	-	-	0.809622	0.811783
Градиентный бустинг тест f1	-	-	-	-	0.82224	0.8208
Логистическая регрессия cv f1	0.813328	0.815605	-	-	-	-
Логистическая регрессия тест f1	0.828244	0.826966	-	-	-	-
Случайный лес cv f1	-	-	0.810167	0.811113	-	-
Случайный лес тест f1	-	-	0.82224	0.82224	-	-

Таблица 6. Классификация. Сравнение моделей с произвольными и подобранными гиперпараметрами относительно f1

В качестве альтернативного критерия нами была выбрана метрика f1. f1-score является гармоническим средним между точностью (precision) и полнотой (recall), что позволяет бороться с проблемой дисбаланса классов, также это позволяет более точно сравнивать разные модели машинного обучения и данная метрика сильно чувствительна к ложноположительным и ложноотрицательным результатам, что важно в нашей задаче. Из недостатков можно отметить зависимость от выбранного порога, неприменимость к многоклассовой классификации.

В результате подбора гиперпараметров лучшее качество все еще лучшее качество показывает Логистическая регрессия , но качество на случайном лесе и градиентном бустинге выросло, что может свидетельствовать о наличии небольшого дисбаланса классов.Результаты представлены в таблице 6.

Повышенная сложность: дополнительно самостоятельно запрограммируйте не представленный в стандартных библиотеках критерий качества и используйте его для тюнинга гиперпараметров. Сравните результат стандартного и вашего критериев.

В качестве альтернативного критерия будем использовать критерий Gini, который считается преобразованием площадью под кривой AUC. Мы сравнивали качество моделей, то есть accuracy на моделях, с гиперпараметрами подобранными на accuracy и критерии Джини

	Accuracy_lr	Gini_lr	Accuracy_rf	Gini_rf	Accuracy_gb	Gini_gb
learning_rate	-	-	-	-	0.3	0.1
max_depth	-	-	10	10	1.0	1.0
max_features	-	-	sqrt	sqrt	-	-
max_samples	-	-	600	400	-	-
n_estimator	-	-	-	-	10.0	10.0
n_estimators	-	-	100	50	-	-
solver	liblinear	lbfgs	-	-	-	-
Градиентный бустинг cv ACC	-	-	-	-	0.8012	0.8012
Градиентный бустинг тест ACC	-	-	-	-	0.8208	0.8076
Логистическая регрессия cv	0.803333	0.8032	-	-	-	-
Логистическая регрессия тест	0.8208	0.82	-	-	-	-
Случайный лес cv ACC	-	-	0.802267	0.801733	-	-
Случайный лес тест ACC	-	-	0.8056	0.8052	-	-

Таблица 7. Классификация. Сравнение моделей относительно accuracy с произвольными и подобранными гиперпараметрами на критерии Gini

Можно заметить в таблице 7, что качество моделей после подбора гиперпараметров критерием Джини, по сравнению с accuracy.

3.5 Постройте ROC-кривую для ваших моделей и сравните их по AUC на тестовой выборке.

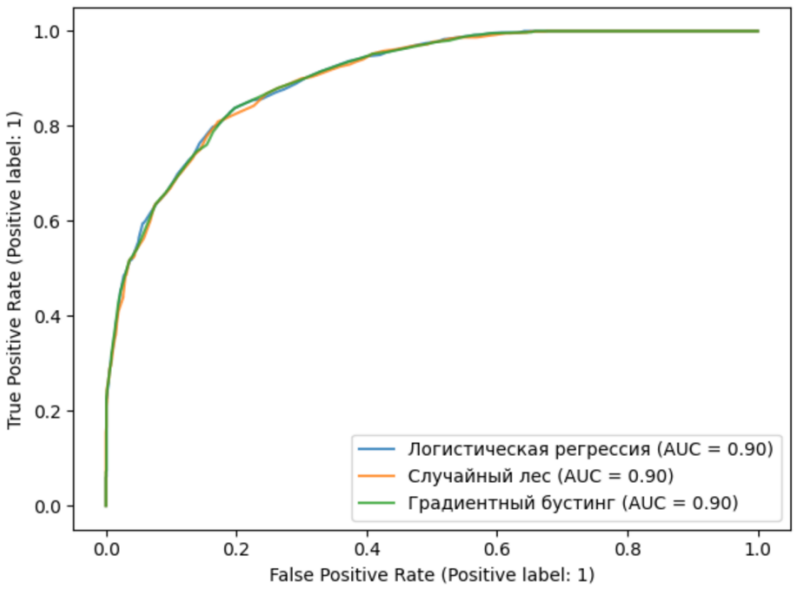

Таблица 8. Классификация. Сравнение моделей ROC AUC

Модели почти не отличаются по метрике AUC.

Повышенная сложность: дополнительно выполните это задание для Байесовской сети.

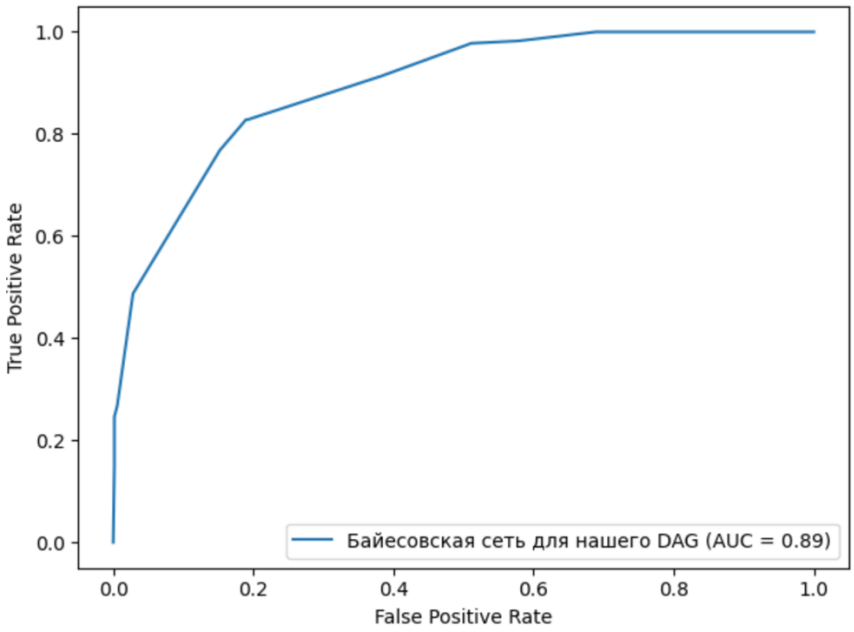

Таблица 9. Классификация. ROC AUC для Байесовской сети

3.6 Постройте матрицу путаницы и предположите цены различных видов прогнозов. Исходя из критерия максимизации прибыли на обучающей выборке подберите оптимальный порог прогнозирования для каждого из методов и сравните прибыли на тестовой выборке при соответствующих порогах. Результат представьте в форме таблицы, в которой должны быть указаны как AUC, так и прибыли (на тестовой выборке). Проинтерпретируйте полученный результат.

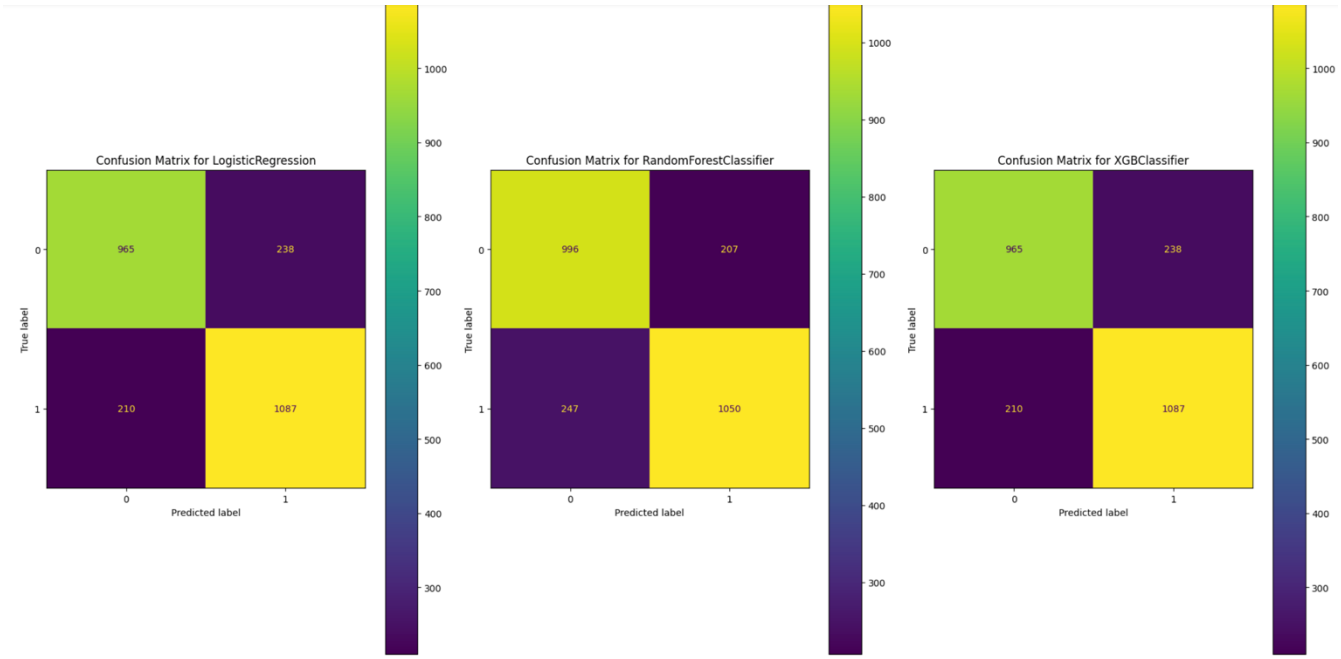

Таблица 10. Классификация. Матрица путаницы

	AUC	profit
LogisticRegression	0.903339	258360
RandomForestClassifier	0.900412	256380
XGBClassifier	0.902871	260740

Таблица 11. Классификация. Сравнение прибылей и AUC по моделям

Исходя из здравого смысла нами были выдвинуты следующие предположения о прибыли относительного каждого сектора матрицы:

TN_prof = 0

TP_prof = 500

FP_prof = -1000

FN_prof = -170

То есть для нашей модели неправильно классифицировать факт просмотра рекламы человеком наиболее неблагоприятный исход, а правильная классификация факта отсутствия просмотра не имеет значение, то есть наша модель должна фокусироваться на максимизации TP, таблица 11.

3.7 Опишите предполагаемые связи между переменными в форме ориентированного ациклического графа (DAG). Обучите структуру Байесовской сети на обучающей выборке и сравните точность прогнозов вашего и обученного DAG на тестовой выборке.

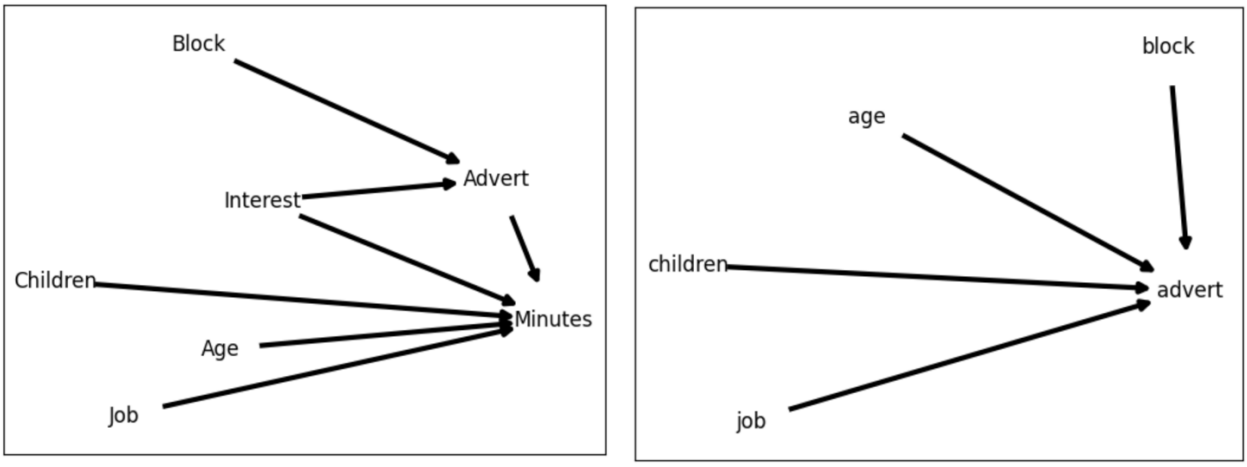

Рис. 12. Классификация. DAG

У нас огромная проблема с предсказаниями, так как переменная возраст - непрерывная, библиотека итдуфкт по дефолту такое не поддерживает, поэтому так как в тестовых данных есть люди с возрастом которого нет в тренировочных, предсказания ломаются. Для решения этой проблемы мы объединили людей в возрастные группы (25, 35, 45, 55, 65, 75).

	ACC-test
Байесовская сеть с исходным DAG	0.8196
Байесовская сеть с обученным DAG	0.8196

Таблица. 13. Классификация. Сравнение Байесовской сети с данным DAG и с обученным DAG

Исходя из результатов качество на Байесовской сети с обученным и данным DAG не отливаются, таблица 13.

3.9 **Повышенная сложность:** включите в анализ дополнительный метод классификации, не рассматривавшийся в курсе и не представленный в библиотеке scikit-learn. Опишите данный метод (принцип работы, преимущества и недостатки) и осуществите тюнинг гиперпараметров. Сопоставьте его точность на тестовой выборке с точностью лучшего из обученных вами ранее методов.

В качестве альтернативной модели мы использовали CatBoost Classifier – модель, разработанная Yandex, основанная на градиентном бустинге.

Принципы работы :

1) Использование симметричных случайных деревьев, что значительно ускоряет процесс обучения
2) Минимизация переобучения и улучшение качество моделирования за счет использования нового подхода к построению деревьев, при котором данные для обучения и валидации разделяются по-разному для различных итераций обучения
3) Автоматическая обработка категориальных признаков

Преимущества:

1) Скорость обучения
2) Устойчивость к переобучению
3) Легкость и понятность

Недостатки:

1) При большом объеме данных относительно трудозатратно настраивать параметры
2) При больших объемах данных скорость может снижаться
3) Сложность настройки параметров, несмотря на то что модель хорошо работает с дефолтными

Для наших данных модель CatBoost показала почти наилучшее качество, accuracy = 0.8208, несмотря на то, что на некоторых прогонах кода Логистическая регрессия показывала результаты выше, воспроизводимость и устойчивость результатов Логистической регрессии меньше, чем у CatBoosting, поэтому мы решили, что для наших данных наиболее удачной моделью остается имеенно она.

3.8 На основании проделанного анализа выберите лучший и худший из обученных классификаторов. Обоснуйте сделанный выбор.

Как было упомянуто выше, самое высокое качество accuracy показала модель Логистической регрессии, что может быть связано, что преобладанием линейных связей в наших данных, наихудшее качество показал Случайны лес, что скорее всего связано с переобучением модели, так как мы видели, что наибольшее качество достигается на большой глубине, но оно не сохраняется между метриками, то есть разница между качеством на кросс-валидации и тесте. Исходя из AUC наиболее подходящими также являются логистическая регрессия и градиентный бустинг. Байесовская сеть дала хороший результат, лучший, чем случайный лес, но чуть хуже, чем два вышеперечисленных лидера. Стоит отметить, что после подбора гиперпараметров для Байесовской сети и CatBoostinga качество на тесте не поменялось, что говорит о том, что наши данные достаточно простые, но в них, скорее всего, нет явного разделения классов, поэтому очень сложные модели одинаково хорошо работают на дефолтных и на подобранных гиперпараметрах. Несмотря на то, что Логистическая регрессия показала очень высокие результаты, они менее устойчивы к воспроизведению, в отличие от CatBoosting, который дал такое качество на тесте.

4. Регрессия

В каждом из заданий, если не сказано иного, необходимо использовать хотя бы 3 (на ваш выбор) из следующих методов: случайный лес, метод наименьших квадратов, метод ближайших соседей и градиентный бустинг.

Мы выбрали следующие модели: метод ближайших соседей, случайный лес, градиентный бустинг.

4.1. Отберите признаки, которые могут быть полезны при прогнозировании целевой (зависимой) переменной. Не включайте в число этих признаков переменную воздействия.

Для прогнозирования целевой переменной $Minutes_i$ мы отобрали следующие переменные:

$Age_i, Children_i, Job_i$.

Age_i – предположительно, молодые люди в целом больше любят смотреть видео, а некоторые даже имеют дофаминовую зависимость от просмотра роликов. В связи с этим они могут за раз посмотреть длинное видео, в то время как взрослые люди могут растягивать просмотр.

$Children_i$ – мы предполагаем, что пользователи с детьми могут давать смотреть, например, мультики своим детям. Детям зачастую тяжело самостоятельно закончить просмотр видео, так как они очень увлечены процессом. Соответственно, фактор наличия детей

Job_i – работающие люди располагают меньшим свободным временем, чем безработные. Вследствие чего безработные могут за один раз посмотреть видео целиком, а работающий поставить на паузу и вернуться к просмотру позже.

4.2. Выберите произвольные значения гиперпараметров, а затем оцените и сравните (между методами) точность прогнозов с помощью RMSE и MAPE:
• на обучающей выборке.
• на тестовой выборке.
• с помощью кросс-валидации (используйте только обучающую выборку).

Опять же мы брали дефолтные значения гиперпараметров. Лучшее значение RMSE модели показали на обучающей выборке, худшие – на тесте. Значение MAPE во всех случаях оказалось запредельным, так как на наших данных присутствуют очень маленькие (факт-прогноз) значения, поэтому ориентироваться на данную метрику мы не можем.

4.3. Для каждого метода с помощью кросс-валидации на обучающей выборке подберите оптимальные значения гиперпараметров (тюнинг). В качестве критерия качества используйте RMSE. Результат представьте в форме таблицы, в которой для каждого метода должны быть указаны:

• изначальные и подобранные значения гиперпараметров.

• кросс-валидационное значение RMSE на обучающей выборке с исходными и подобранными значениями гиперпараметров.

• значение RMSE на тестовой выборке с исходными и подобранными значениями гиперпараметров.

Подбираемые гиперпараметры

Градиентный бустинг

- max_depth - максимальная глубина дерева, определяет, как далеко дерево может разветвиться в каждом узле при построении модели
- learning rate – скорость обучения, задает величину шага, на которую обновляются предсказания на каждой итерации в градиентном бустинге
- n_estimators - определяет количество деревьев, которые будут добавлены к ансамблю в процессе обучения градиентного бустинга. Каждое новое дерево выучивает дополнительную информацию об остатках (разнице между предсказанными значениями и фактическими значениями) модели, улучшая качество предсказаний

Метод ближайших соседей

- n_neighbors - определяет количество ближайших соседей, которые будут использоваться для выполнения регрессии

Случайный лес

- max_depth – определяет максимальную глубину деревьев, которые будут использоваться в случайном лесе
- max_samples – определяет максимальное количество образцов (наблюдений), которые будут использоваться при обучении каждого дерева в случайном лесе. Если устанавливается в значение None, то все образцы будут использоваться при обучении каждого дерева. Если устанавливается в значение меньше 1.0 (например, 0.7), то каждое дерево будет обучаться на случайной подвыборке, состоящей из указанной доли образцов от общего числа.
- n_estimators – отвечает за количество деревьев, которые будут использоваться в случайном лесе. Каждый дерево в случайном лесе строится независимо от других деревьев, что позволяет модели работать на основе ансамбля деревьев.

	default_knn	best_knn	default_rf	best_rf	default_gb	best_gb
learning_rate	-	-	-	-	0.1	0.1
max_depth	-	-	3.0	6.0	3.0	3.0
max_samples	-	-	-	400.0	-	-
n_estimators	-	-	100.0	1000.0	100.0	100.0
n_neighbors	5.0	24.0	-	-	-	-
Градиентный бустинг cv RMSE	-	-	-	-	11.421752	11.421752
Градиентный бустинг тест RMSE	-	-	-	-	11.155561	11.155561
Метод ближайших соседей cv RMSE	12.571647	11.71365	-	-	-	-
Метод ближайших соседей тест RMSE	12.180639	11.414637	-	-	-	-
Случайный лес cv RMSE	-	-	11.570353	11.418848	-	-
Случайный лес тест RMSE	-	-	11.281641	11.154417	-	-

Таблица 14. Регрессия. Сравнение моделей с произвольными и подобранными гиперпараметрами

Повышенная сложность: подберите на обучающей выборке оптимальные значения гиперпараметров градиентного бустинга ориентируясь на значение OOB (out-of-bag) ошибки. Сопоставьте гиперпараметры и точность на тестовой выборке для градиентного бустинга в зависимости от того, используется кросс-валидация или OOB ошибка.

	OOB	CV
max_depth	3.000000	3.000000
learning_rate	0.010000	0.010000
n_estimators	100.000000	1000.000000
Градиентный бустинг тест RMSE	11.890290	11.271049
Градиентный бустинг cv RMSE	11.959098	11.418281

Таблица 15. Регрессия. OOB и кросс-валидация

Подобранные гиперпараметры в случае OOB ошибки несколько отличаются (n_estimators = 100) от параметров, подобранных с использованием кросс-валидации (n_estimators = 1000). Качество модели у кросс-валидации оказалось лучше.

4.4. На основании проделанного анализа выберите лучший и худший из обученных классификаторов. Обоснуйте сделанный выбор.

Основываясь на Таблице 15. На тесте лучшее качество после подбора гиперпараметров показал случайный лес, худшее качество - метод ближайших соседей.

На кросс-валидации по обучающей выборке с подобранными гиперпараметрами наименьшее значение RMSE у случайного леса, а наибольшее у метода ближайших соседей.

Можно сделать вывод, что наилучшим образом себя показали градиентный бустинг и случайный лес, хуже всего - метод ближайших соседей. Хотя стоит отметить, что после подбора гиперпараметров качество градиентного бустинга не улучшилось, по сравнению со случайным лесом и KNN.

4.5. **Повышенная сложность:** включите в анализ дополнительный метод регрессии, не рассматривавшийся в курсе и не представленный в библиотеке scikitlearn. Опишите данный метод (принцип работы, преимущества и недостатки) и осуществите тюнинг гиперпараметров. Сопоставьте его точность на тестовой выборке с точностью лучшего из обученных вами ранее методов.

Мы использовали CatBoostRegressor() из библиотеки catboost.

Принцип работы метода CatBoostRegressor() заключается в использовании алгоритма градиентного бустинга для решения задач регрессии. CatBoost использует алгоритм градиентного бустинга, который обучает ансамбль деревьев решений последовательно. Каждое новое дерево добавляется с учетом ошибок, сделанных предыдущими деревьями.

CatBoost позволяет автоматически обрабатывать категориальные признаки, что является одним из его основных преимуществ. Кроме того, он имеет много гиперпараметров для настройки, которые могут улучшить производительность модели. Также он имеет высокую производительность и скорость обучения модели.

Недостатками метода CatBoostRegressor() могут быть:
1. Требовательность к вычислительным ресурсам из-за своей сложности.
2. Возможное переобучение модели из-за большого количества деревьев в ансамбле.

	default_knn	best_knn	default_rf	best_rf	default_gb	best_gb	default_cat	best_cat
CatBoost cv RMSE	-	-	-	-	-	-	11.523714	11.389818
CatBoost тест RMSE	-	-	-	-	-	-	11.247864	11.141676
depth	-	-	-	-	-	-	6.0	4.0
iterations	-	-	-	-	-	-	1000.0	50.0
learning_rate	-	-	-	-	0.1	0.1	0.03	0.1
max_depth	-	-	3.0	6.0	3.0	3.0	-	-
max_samples	-	-	-	400.0	-	-	-	-
n_estimators	-	-	100.0	1000.0	100.0	100.0	-	-
n_neighbors	5.0	24.0	-	-	-	-	-	-
Градиентный бустинг cv RMSE	-	-	-	-	11.421752	11.421752	-	-
Градиентный бустинг тест RMSE	-	-	-	-	11.155561	11.155561	-	-
Метод ближайших соседей cv RMSE	12.571647	11.71365	-	-	-	-	-	-
Метод ближайших соседей тест RMSE	12.180639	11.414637	-	-	-	-	-	-
Случайный лес cv RMSE	-	-	11.570353	11.418848	-	-	-	-
Случайный лес тест RMSE	-	-	11.281641	11.154417	-	-	-	-

Таблица 16. Регрессия. Сравнение моделей и catboost

Как можно заметить, CatBoost показал лучшее качество как на тестовой выборке, так и на кросс-валидации по обучающей выборке.

5. Эффекты воздействия

5.1. Математически запишите и содержательно проинтерпретируйте потенциальные исходы целевой переменной. Объясните, как они связаны с наблюдаемыми значениями целевой переменной.

Сформируем представления о поведении пользователя при просмотре видео в зависимости от факта просмотра рекламы.

Основная идея заключается в том, что интерес к видео, возраст, наличие детей и работы сильнее влияют на количество просматриваемого контента, когда индивид смотрит предпочитает смотреть рекламу, ведь, как уже упоминали выше, пользователь, которые спойкойно относятся к рекламе, не пресыщены общим количеством просмотренного контента и, таким образом, все факторы начинают иметь большую отдачу.

Уравнение потенционального исхода целевой переменной при просмотре рекламы:

$$\text{Minutes}_{1i} = \underbrace{\underbrace{0.7 \times \text{Interest}_i}_{g_1^{\text{unobs}}} + \underbrace{14 \times \frac{\text{Children}_i}{2 - \text{Age}_i - \text{Job}_i} + \frac{\text{Age}_i \times \text{Job}_i}{3}}_{g_1^{\text{obs}}}}_{g_1} + \varepsilon_{1i}, \text{ где } \varepsilon_{1i} \sim (\text{EXP}(0.1) - 10)$$

Уравнение потенционального исхода целевой переменной при отсутствии просмотра рекламы:

$$\text{Minutes}_{0i} = \underbrace{\underbrace{0.5 \times \text{Interest}_i}_{g_0^{\text{unobs}}} + \underbrace{12 \times \frac{\text{Children}_i}{5 - \text{Age}_i - \text{Job}_i} + \frac{\text{Age}_i \times \text{Job}_i}{4}}_{g_0^{\text{obs}}}}_{g_0} + \varepsilon_{0i}, \text{ где } \varepsilon_{0i} \sim (8 \times t(15))$$

Как известно, для одного и то же наблюдения одновременно не могут выполняться оба потенциальных исхода. Существует наблюдаемое значение целевой переменной. В данном случае наблюдаемое значения количества просмотренных минут Minutes_i равно Minutes_{0i}, если не было просмотра рекламы, и Minutes_{1i} в ином случае. Заметим, что ранее эти значения были так сгенерированы, что $\text{Minutes}_{1i} >= \text{Minutes}_{0i}$.

Таким образом, наблюдаемое значение целевой переменной можно записать следующим образом.

$$\text{Minutes}_i = \begin{cases} \text{Minutes}_{1i}, \text{ если } \text{Advert}_i = 1 \\ \text{Minutes}_{0i}, \text{ если } \text{Advert}_i = 0 \end{cases} == \text{Minutes}_{1i} \times \text{Advert}_i + \text{Minutes}_{0i} \times (1 - \text{Advert}_i)$$

5.2. Используя симулированные вами, но недоступные в реальных данных потенциальные исходы (гипотетические значения), получите оценки среднего эффекта воздействия, условных средних эффектов воздействия и локального среднего эффекта воздействия. Результаты представьте в форме таблицы.

Настоящие эффекты воздействия:

$$\text{TE}_i = \text{Minutes}_{1i} - \text{Minutes}_{0i}$$

TE = [22.51766363, 25.33091959, 1.77667212, 0, 0]

Средний эффект воздействия:

$$\text{ATE} = \text{E}\left(\text{Minutes}_{1i} - \text{Minutes}_{0i}\right)$$

Если бы у нас были данные о потенциальных исходах Minutes_{1i} и Minutes_{0i}, то мы могли бы точно оценить ATE как:

$$\widehat{\text{ATE}} = \frac{1}{n}\sum_{i=1}^{n} \text{Minutes}_{1i} - \text{Minutes}_{0i}$$

ATE = 13.161

Локальный средний эффект воздействия:

$$\text{LATE} = \text{E}(\text{Hours}_{1i} - \text{Hours}_{0i} | \text{Advert}_{1i} > \text{Advert}_{0i})$$

Здесь необходимо объяснить, кого в наших данных мы считаем за Always taker, Never taker, Complier и Denier.

- Always taker - пользователь, который смотрит рекламу, независимо от того, у него есть блокировщик рекламы на устройстве или нет
- Never taker - пользователь, который не смотрит рекламу, независимо от того, у него есть блокировщик рекламы на устройстве или нет
- Complier - пользователь, который смотрит рекламу, если у него нет блокировщика рекламы, и не смотрит рекламу, если у него есть блокировщик
- Denier - пользователь, который смотрит рекламу, если у него есть блокировщика рекламы, и не смотрит рекламу, если у него нет блокировщика

Метрика LATE считается для Compliers

LATE = 12.817

Условный средний эффект воздействия:

$$\mathrm{CATE}_i = \mathrm{E}\left(\mathrm{Minutes}_{1i}|X_i\right) - \mathrm{E}\left(\mathrm{Minutes}_{0i}|X_i\right) = g_1(X_i) - g_0(X_i)$$

CATE: [11.50246047 19.40789708 11.91314347 10.67766154 10.05094339]

Финальные результаты:

```
       Значение
ATE   13.161334
LATE  12.817489

          TE       CATE
0  22.517664  11.502460
1  25.330920  19.407897
2   1.776672  11.913143
3   0.000000  10.677662
4   0.000000  10.050943
```

5.3 Оцените средний эффект воздействия как разницу в средних по выборкам тех, кто получил и не получил воздействие. Опишите недостатки соответствующего подхода с учетом специфики рассматриваемой вами экономической проблемы.

Наивный подход предполагает оценивание ATE как средней разницы в зарплатах людей с высшим образованием и без высшего образования.

$$\widehat{\mathrm{ATE}}_{\mathrm{naive}} = \frac{1}{n_1} \sum_{i:\mathrm{Advert}_i=1} \mathrm{Minutes}_{1i} - \frac{1}{n_0} \sum_{i:\mathrm{Advert}_i=0} \mathrm{Minutes}_{0i}$$

Где n_1 и n_0 - число людей с высшим образованием и без высшего образования соответственно.

	Оценка
ATE	13.161334
ATE naive	11.781931

Как видим, оценка наивным образом все-таки отличается от настоящего ATE. Это свидетельствует о том, что не выполнено допущение о независимости:

$$\mathrm{E}(\mathrm{Minutes}_{1i}|\mathrm{Advert}_i = 1) = \mathrm{E}(\mathrm{Minutes}_{1i}) \qquad \mathrm{E}(\mathrm{Minutes}_{0i}|\mathrm{Advert}_i = 0) = \mathrm{E}(\mathrm{Minutes}_{0i})$$

В этом и заключается основной недостаток подобного способа оценки. Так происходит, поскольку в нашей задаче контрольные переменные Age_i, Job_i и $\mathrm{Children}_i$ одновременно связаны и с тем, будет ли пользователь смотреть рекламу Advert_i, и с количеством минут, сколько видео будет посмотрено $\mathrm{Minutes}_i$. Для получения более точной оценки среднего эффекта воздействия в следующих пунктах воспользуемся другими методами

5.4. Используя оценки, полученные лучшими из обученных ранее классификационных и регрессионных моделей, оцените средний эффект воздействия с помощью:

а) метода наименьших квадратов.

б) условных математических ожиданий.

в) взвешивания на обратные вероятности (в случае возникновения ошибок убедитесь в отсутствии оценок вероятностей, равных 0 или 1 и при необходимости измените метод оценивания).

г) метода, обладающего двойной устойчивостью.

д) двойного машинного обучения.

Сравните результаты и назовите ключевую предпосылку этих методов. Содержательно обсудите причины, по которым она может соблюдаться или нарушаться в вашем случае. Приведите содержательную экономическую интерпретацию оценки среднего эффекта воздействия.

а) Метод наименьших квадратов

Для МНК оценики воспользуемся допущением об условной независимости:

$$\mathrm{E}(\mathrm{Minutes}_{1i}|\mathrm{Advert}_i = 1, X_i) = \mathrm{E}(\mathrm{Minutes}_{1i}|X_i) \qquad \mathrm{E}(\mathrm{Minutes}_{0i}|\mathrm{Advert}_i = 0, X_i) = \mathrm{E}(\mathrm{Minutes}_{0i}|X_i)$$

Попробуем оценить ATE, рассмотрев среднюю разницу в оценках минут, проведённых за просмотром видео, полученных с помощью МНК отдельно сперва по пользователям, которые посмотрели рекламу, а затем по пользователям, которые этого не сделали.

$$\widehat{\mathrm{ATE}}_{\mathrm{OLS}} = \frac{1}{n}\sum_{i=1}^{n} \underbrace{\hat{\mathrm{E}}\left(\mathrm{Minutes}_{1i}|X_i\right) - \hat{\mathrm{E}}\left(\mathrm{Minutes}_{0i}|X_i\right)}_{\widehat{\mathrm{CATE}}_i}$$

Где:

- $\hat{\mathrm{E}}\left(\mathrm{Minutes}_{1i}|X_i\right)$ - оценка, полученная с использованием МНК оценок регрессионных коэффициентов β, полученных по выборке из пользователей, просмотревших рекламу $\mathrm{Advert}_i = 1$.
- $\hat{\mathrm{E}}\left(\mathrm{Minutes}_{0i}|X_i\right)$ - оценка, полученная с использованием МНК оценок регрессионных коэффициентов β, полученных по выборке из пользователей, не просмотревших рекламу $\mathrm{Advert}_i = 0$.

Результат:

	Оценка
ATE	13.161334
ATE naive	11.781931
ATE ols	13.338258

Вывод: Оценка получилась еще хуже, чем в наивном методе. Значит все равно есть нарушение предпосылки об условной независимости

б) Условные математические ожидания

Оценки условных математических ожиданий можно получить двумя разными способами: S-learner - оценка $\hat{\mathrm{E}}(\mathrm{Minutes}_i|\mathrm{Advert}_i, X_i)$ по всей выборке, или T-learner - оценка отдельно $\hat{\mathrm{E}}(\mathrm{Minutes}_i|\mathrm{Advert}_i = 1, X_i)$ и $\hat{\mathrm{E}}(\mathrm{Minutes}_i|\mathrm{Advert}_i = 0, X_i)$ по группе воздействия и по контрольной группе соответственно. Заметим, что для обоих способов снова требуется предположения об условной независимости. Воспользуемся обоими способами и затем сравним результаты

T-learner

В случае с T-learner эту оценку можно записать как:

$$\widehat{\mathrm{ATE}}^{\text{T-learner}} = \frac{1}{n}\sum_{i=1}^{n} \hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 1\right) - \hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 0\right)$$

Где:

- $\hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 1\right)$ - оценка, полученная с использованием метода машинного обучения по выборке из пользователей, просмотревших рекламу $\mathrm{Advert}_i = 1$.
- $\hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 0\right)$ - оценка, полученная с использованием метода машинного обучения по выборке из пользователей, не просмотревших рекламу $\mathrm{Advert}_i = 0$.

S-learner

Оценка такая же, как в T-learner. Отличаются только методы оценки условных математических ожиданий.

$$\widehat{\mathrm{ATE}}^{\text{S-learner}} = \frac{1}{n}\sum_{i=1}^{n} \hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 1\right) - \hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 0\right)$$

Где:

- $\hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 1\right)$ - оценка, полученная с использованием метода машинного обучения по всей выборке.
- $\hat{\mathrm{E}}\left(\mathrm{Minutes}_i|X_i, \mathrm{Advert}_i = 0\right)$ - оценка, полученная с использованием метода машинного обучения по всей выборке.

Результат:

	Оценка
ATE	13.161334
ATE naive	11.781931
ATE ols	13.338258
ATE T-learner	13.357342
ATE S-learner	13.013749

Вывод: S-learner демонстрирует результат ближе к реальному эффекту среднего воздействия, чем T-learner. Однако обе оценки дали хуже результат, чем наивный метод и метод наименьших квадратов

в) Взвешивания на обратные вероятности

Метод взвешивания на обратные вероятности можно использовать, когда выполнена предпосылка об условной независимости. Оценка, получаемая с помощью взвешивания на обратные вероятности $\mathbf{IPW}$, имеет вид:

$$\widehat{\mathrm{ATE}}^{\mathrm{IPW}} = \frac{1}{n}\sum_{i=1}^{n}\frac{\mathrm{Advert}_i \times \mathrm{Minutes}_i}{\hat{P}\left(\mathrm{Advert}_i = 1|X_i\right)} - \frac{(1-\mathrm{Advert}_i) \times \mathrm{Minutes}_i}{1-\hat{P}\left(\mathrm{Advert}_i = 1|X_i\right)}$$

Где условные вероятности $\hat{P}\left(\mathrm{Advert}_i = 1|X_i\right)$ оценивается с помощью методов классификации. В данном случае воспользуемся градиентным бустингом

Результат:

	Оценка
ATE	13.161334
ATE naive	11.781931
ATE ols	13.338258
ATE T-learner	13.357342
ATE S-learner	13.013749
ATE IPW	4.793951

Вывод: Оценка с помощью взвешивания на обратные вероятности показывает ужасные результаты

г) Метод, обладающий двойной устойчивостью

Данный метод объединяет методы взвешивания на обратные вероятности метод оценки через условные математические ожидания, проблемой которых являлось то, точность оценок каждого из этих способов зависит от точности оценок соответствующих условных математических ожиданий или вероятностей. Если они оценены неточно, то и итоговая оценка ATE также будет неточной. Поэтому метод, обладающий двойной устойчивостью, совмещает оба способа, чтобы оценка ATE оказывалась состоятельной, если по крайней мере один из них дает состоятельную оценку

Результат:

	Оценка
ATE	13.161334
ATE naive	11.781931
ATE ols	13.338258
ATE T-learner	13.357342
ATE S-learner	13.013749
ATE IPW	4.793951
ATE DR	13.502459

Вывод: Оценка методом, обладающим двойной устойчивости, обладает достойным результатом

Д) Двойное машинное обучение

Результаты:

	Оценка
ATE	13.161334
ATE naive	11.781931
ATE ols	13.338258
ATE T-learner	13.357342
ATE S-learner	13.013749
ATE IPW	4.793951
ATE DR	13.502459
ATE dml standard	13.969030

Вывод: ДМО справляется с оценкой среднего эффекта воздействия хуже других способов, кроме наивного и IPW

Вывод: С помощью данных способов мы получили значения ATE более близкие к реальному значению, чем в наивном подходе. Это подтверждает то, что допущение об условной независимости, в отличие от допущения о полной независимости, может выполняться, так как по сути пользователь, при условии конкретных личных характеристик, уже будет независимо принимать решение о просмотре или не просмотре рекламы.

Содержательная интепретация ATE: в нашей постановке задачи средний эффект воздействия показывает то, насколько в среднем те, кто решает посмотреть рекламу, смотрят ее дольше, чем те, кто отключает видео, как только видит предупреждение о рекламе. Так, если значение оказывается больше (скажем, >10), то можно предположить, что просмотр рекламы не снижает интерес к видео и у пользователей не появляется желание сразу же выключить видео из-за надоедливой рекламы

5.5 Оцените локальный условный эффект воздействия с помощью:

• двойного машинного обучения без инструментальной переменной.

• двойного машинного обучения с инструментальной переменной.

Сопоставьте результаты и объясните, в чем в вашем случае будет заключаться различие между средним эффектом воздействия и локальным средним эффектом воздействия. Приведите содержательную экономическую интерпретацию оценки локального среднего эффекта воздействия.

Результат:

	Оценка
ATE	13.161334
LATE	12.817489
LATE dml standard2	4.595290
LATE dml iv	13.176224

Содержательная интерпретация: LATE показывает среднюю разницу для группы, просмотревших рекламу, между тем, сколько минут видео они посмотрели и сколько они бы посмотрели, если не посмотрели бы

5.6. Оцените условные средние эффекты воздействия с помощью:

• метода наименьших квадратов.

• S-learner.

• T-learner.

• метода трансформации классов.

• X-learner.

Сравните результаты и обсудите, насколько в вашем случае мотивированы применение метода X-learner. Опишите, как можно было бы использовать полученные вами оценки в бизнесе или при реализации государственных программ.

Результат:

	True	OLS	T-learner	S-learner	X-learner	IPW
0	11.502460	11.332502	10.624161	11.183665	11.044042	6.470140
1	19.407897	19.632423	16.837683	16.144752	16.920762	-43.208403
2	11.913143	11.172412	15.515763	15.678734	12.493540	-31.259472
3	10.677662	10.297499	14.971588	14.777771	15.217200	11.220185
4	10.050943	10.379752	12.128994	11.301191	12.505081	10.110409
...	...	...	...	...	...	...
9995	8.357293	8.202819	8.865980	10.787889	8.374528	-8.426242
9996	8.020951	8.095455	8.416621	9.058485	9.262200	9.871421
9997	16.599687	17.390148	14.563858	15.951202	16.040523	-36.217885
9998	8.554052	8.740831	8.469425	8.960287	11.098705	24.952892
9999	10.564984	10.429720	12.033763	11.063701	11.226498	3.304411

Содержательная интерпретация: Стриминговые сервисы могут использовать эти значения в качестве аргумента о том, что реклама в видео не снижает интерес к видео, и поэтому можно безболезненно добавлять рекламную интеграции такого формата для увеличения прибыли компании

5.7. Выберите лучшую модель оценивания условных средних эффектов воздействия,используя:

• истинные значения условных средних эффектов воздействия.

• прогнозную точность моделей.

• псевдоисходы.

Проинтерпретируйте различия в результатах различных подходов.

Сравним точность оценок по среднеквадратической ошибке (недоступно на реальных данных):

$$\mathrm{MSE}_0 = \frac{1}{n}\sum_{i=1}^{n}\left(\mathrm{CATE}_i - \widehat{\mathrm{CATE}}_i\right)^2$$

Результат

	MSE0
OLS	1.685991
T-learner	4.655986
S-learner	2.952090
X-learner	2.906315
CT	580.562310

Среднеквадратическая ошибка показала, что OLS и S-learner точнее остальных методов. Сравним прогнозную точность этих моделей по среднеквадратической ошибке

$$\mathrm{MSE} = \frac{1}{n}\sum_{i=1}^{n}(\mathrm{Minutes}_i - \hat{\mathrm{E}}(\mathrm{Minutes}_i|X_i, T_i))^2$$

Результат

	MSE1
OLS	87.526923
S-learner	82.989468

Сравним точность оценок с помощью псевдоисходов:

$$\mathrm{MSE}^* = \frac{1}{n}\sum_{i=1}^{n}\left(\mathrm{Minutes}_i^* - \widehat{\mathrm{CATE}}_i\right)^2$$

Результат

	MSE2
OLS	7962.046998
T-learner	7964.628816
S-learner	7979.405132
X-learner	7950.713271
CT	7280.435688

Вывод: по реальной среднеквадратической ошибке лучший результат демонстрирует OLS можель, тогда как прогнозной точность модели выше у S-learner. Тем не менее, по псевдоисходам лучший результат демонстрирует CT, который очень плох по среднеквадратической ошибке. Полагаю, так происходит из-за того, что CT обучается как раз на псевдоисходах, а не отличных реальных значений целевой переменной

5.8. Оцените средние эффекты воздействия и локальные средние эффекты воздействия используя худшие из обученных классификационных и регрессионных моделей. Сопоставьте результаты с теми, что были получены с помощью лучших моделей. Сдеайте вывод об устойчивости результатов к качеству используемых методов машинного обучения.

Двойное машинное обучение подходит и для того оценки средних эффектов воздействия и для оценки локальаных средних эффектов воздействия. Попробуем использовать его для сравнения и здесь

Результаты для ATE:

	Оценка
ATE	13.161334
ATE naive	11.781931
ATE dml best ols	13.782537
ATE dml best worst	11.826012

Результаты для LATE:

	Оценка
ATE	13.161334
LATE	12.817489
LATE dml iv best	13.176224
LATE dml iv worst	14.249288

Вывод: Результаты оказываются сильно хуже, если брать плохие модели машинного обучения, особенно это справедливо для локального среднего эффекта воздействия.