ПРАВИТЕЛЬСТВО РОССИЙСКОЙ ФЕДЕРАЦИИ

Федеральное государственное автономное образовательное учреждение высшего образования

Национальный исследовательский университет «Высшая школа экономики»

Факультет экономических наук
Образовательная программа «Экономика»

КУРСОВАЯ РАБОТА

«Методы интерпретации моделей машинного обучения»

Студентка группы БЭК171
Махнева Елизавета Александровна

Научный руководитель:
Соколов Евгений Андреевич

Москва 2020

Содержание

1 Введение

Во введении я хочу сказать про то что существует интерпретируемые и нет модели. Про то что нам хотелось бы интерпретировать и привести аргументы за (как минимум те 3 из презенташки). Наверное стоит объединить с блоком интерпретации, так как здесь особо больше нечего писать (если только много не получится), а воду лить не хочется

1. Интерпретируемые и нет модели. Желательно показать примеры, почему не интерпретируются. 2. Интерпретация нужна! Потому что... 3. Плюсы интерпретации и небольшие минусы 4. Что и как хотелось бы интерпретировать. Основы интерпретации – какой она должна быть 5. Кратко перечислить методы, сказать какие они бывают

2 Методы интерпретации

2.1 PDP

PDP (Partial Dependence Plot, график частичной зависимости) – график, который показывает зависимость прогноза модели от значения отдельного признака. С его помощью мы можем понять, как некоторый признак влияет на предсказание. Данный график можно изобразить для двух либо трех признаков из имеющихся.

2.1.1 Идея

Визуализация – это отличный способ интерпретации. Если мы хотим понять, как признаки влияют на результат, можно посмотреть, как меняется прогноз от изменения одного признака при прочих равных. В идеальной ситуации мы бы построили график зависимости результата от всех признаков и меняли бы только один. Однако мы сталкиваемся с проблемой: если признаков больше двух, построить график не получится. Поэтому чтобы сохранить возможность визуализации, можно анализировать зависимость результата от одного признака без учета влияния остальных, построив график зависимости от одного признака. Аналогично можно изучать влияние одновременно двух признаков, построив трехмерный график.

<Пример 2мерного и 3мерного графиков>

2.1.2 Принцип работы

Пусть $x = (x_1, \dots, x_d)$ – вектор признаков объекта. У нас есть модель $f(x) : \mathbb{R}^d \to \mathbb{R}$. Мы хотим понять, как признаки x_1 и x_2 влияют на предсказание модели. Обозначим $x_r = (x_3, \dots, x_d)$ за вектор остальных признаков.

Нам нужно получить функцию зависимости предсказания от одного-двух признаков при зафиксированных остальных: $g(x_1, x_2) = f(x_1, x_2 | \, x_r)$. Но если x_1 и/или x_2 зависимы с признаками из x_r, то возникает проблема. При изменении анализируемого признака меняется и зависимый с ним, который мы не рассматриваем – мы не сможем рассмотреть чистый предельный эффект одного признака, на него всегда будет наложен эффект другого предиктора. Поэтому одной из предпосылок метода является независимость исследуемых признаков от остальных.

Но даже с предпосылкой о независимости признаков функция $g(x_1, x_2)$ не будет показывать точный результат, так как предельные эффекты предикторов разные для разных объектов выборки. Поэтому мы рассмотрим, как влияют анализируемые признаки на среднее предсказание. То есть найдем матожидание предсказания модели при фиксированных исследуемых признаках (как констант с точки зрения матожидания):

$$\bar{g}(x_1, x_2) = \mathbb{E}(f(x_1, x_2, x_r) \, | x_r)$$

Таким образом, мы получим функцию, которая показывает предельные эффекты признаков для среднего предсказания. Но чтобы найти матожидание, мы должны знать истинные распределения признаков. Поскольку нам недоступна данная информация, можно воспользоваться методом Монте-Карло, чтобы примерно оценить искомую функцию:

$$\hat{g}(x_1, x_2) = \frac{1}{n} \sum_{i}^{n} f(x_1, x_2, X_r^{(i)}),$$

где $X_r^{(i)}$ – i строка матрицы X_r, содержащей признаки x_r для всех объектов выборки.

Результат: функция показывает, как исследуемые признаки в среднем влияют на результат работы модели. Мы можем построить ее график, чтобы более наглядно посмотреть на влияние предикторов на предсказание.

2.1.3 Реализация

2.2 LIME

LIME (Local Interpretable Model-Agnostic Explanations) – метод, показывающий вклад признаков в отдельное предсказание, работающий с любой моделью.

2.2.1 Идея

Результаты некоторых моделей легко интерпретировать. Например, в линейной регрессии можно посмотреть на веса. Они показывают, насколько изменится предсказание при изменении признаков. Так для каждого конкретного предсказания можно понять, почему модель выдала именно такой результат – виден непосредственный вклад каждого признака.

Но не все модели легко интерпретировать. Например, некоторые архитектуры нейронных сетей. Они зачастую значительно превосходят линейные модели, но при этом сама структура модели представляет собой «черный ящик» – непонятно, как именно модель сформировала предсказание, какие признаки сильнее повлияли на решение нейронной сети.

Идея состоит в том, чтобы перенести свойство интерпретируемости простых моделей на более сложные. Мы можем обучить интерпретируемую модель по выборке, где ответами являются предсказания сложной модели. В процессе обучения модель анализирует зависимости непосредственно между признаками и предсказаниями сложной модели. Тогда мы сможем интерпретировать результаты простой модели, которые являются аппроксимацией предсказаний сложной модели.

Возникает проблема: сложная модель выявляет зависимости, которые, например, линейная модель может не уловить. Но мы можем воспользоваться свойством, что дифференцируемые функции можно линеаризовать в окрестности заданной точки. То есть, если мы будем рассматривать одно предсказание, то в его небольшой окрестности мы можем считать простую модель аппроксимацией более сложной.

<Свой пример для чиселок>

2.2.2 Принцип работы

У нас есть модель $f(x) : \mathbb{R}^d \to \mathbb{R}$ и объект $x \in \mathbb{R}^d$, предсказание для которого нужно интерпретировать. Мы хотим найти модель g из класса интерпретируемых моделей G, чтобы получить из нее объяснение результата более сложной модели (например, посмотреть на веса признаков).

Сложность моделей обычно обратно зависит от ее интерпретируемости. Например, линейную модель с 2-3 предикторами гораздо проще интерпретировать, чем модель с 10 и более предикторами. Поэтому нам нужно не просто использовать более простую модель, но и преобразовать исходное пространство признаков: $x \to x'$. Стоит также помнить, что мы хотим получить интерпретацию на исходных предикторах. Поэтому нам не подходят

методы снижения размерности, которые представляют признаки в уже неинтерпретируемом виде (PCA, UMAP и пр.). В данной задаче можно уменьшить количество признаков и преобразовать.

Чтобы уменьшить количество признаков, мы можем случайно выбирать некоторые из них. Либо комбинировать их между собой, при этом обращая внимание на интерпретируемость новых признаков. Для каждого типа данных подобное преобразование может проходить по-разному: для изображений – несколько пикселей могут объединяться в один суперпиксель, для текстов – символы объединяться в токены (например, слова – они хорошо интерпретируются).

Для преобразования признаков существует большое количество методов. Например, приведение непрерывных переменных к дискретному виду. Основная цель подобных преобразований: уменьшить множество значений признаков, после чего авторы алгоритма предлагают приводить предикторы к еще более упрощенному виду: использовать вместо признака дамми-переменную его наличия. Именно поэтому наша простая модель $g(x') : \{0,1\}^{d'} \to \mathbb{R}$ работает с интерпретируемым представлением предсказания $x' \in \{0,1\}^{d'}$, где обычно $d' << d$. Дополнительно в задаче вводится мера сложности $\Omega(g)$ искомой модели как регуляризация.

Чтобы определить окрестность рядом с x, внутри которой мы можем использовать простую модель, введем меру близости $\pi_x(z)$ между объектом x и его соседом z. И наконец определим нашу функцию потерь, которую мы будем оптимизировать: $L(f, g, \pi_x)$ – разница между моделями f и g в окрестности, заданной π_x. Тогда в целом задача алгоритма выглядит следующим образом:

$$explanation(x) = \xi(x) = \underset{g \in G}{\operatorname{argmin}}(L(f, g, \pi_x) + \Omega(g))$$

Одной из особенностей алгоритма является его независимость от модели, которую необходимо интерпретировать. Поэтому мы не можем приписывать модели f никакие свойства. Вместо этого мы будем аппроксимировать ее, искусственно создавая объекты в окрестности x', переводя их в исходное пространство признаков и получая для них предсказания из f.

Понятие окрестности носит абстрактный характер, поэтому чтобы учесть расстояние между объектами на практике, мы будем использовать меру близости $\pi_x(z)$ как веса: чем ближе объект к x, тем больший вклад он вносит в функцию потерь.

Гиперпараметры в задаче:

- $\diamond$ G – класс интерпретируемых моделей: линейные модели, решающие деревья и др.
- $\diamond$ d' – количество признаков в новом пространстве
- $\diamond$ $\pi_x(z)$ – мера близости: например, ядра (гауссово, логистическое и др.)
- $\diamond$ $D(x, z)$ – расстояние между объектами, используемое при расчете меры близости: евклидова метрика, косинусное расстояние и др.
- $\diamond$ L – функция потерь: MSE, MAE и др.

Результат: мы получаем алгоритм, который изучает работу более сложной модели и интерпретирует ее с некоторой погрешностью – можно изучать влияние признаков на отдельные предсказания.

2.2.3 Реализация

2.3 SHAP

SHAP (SHapley Additive exPlanations) – метод, оценивающий вклад признаков в предсказания модели на основе значений Шэпли.

2.3.1 Идея

Предсказание модели формируется на основе признаков объектов. Если мы хотим узнать влияние отдельного признака, мы можем построить предсказание модели без него и с ним и посмотреть как меняется результат. Но модель может быть слишком сложной, чтобы мы могли оценить влияние предиктора по одному объекту, регулируя один признак при фиксированных остальных. Правильнее рассмотреть все возможные комбинации всех признаков: их разные значения, наличие/отсутствие, чтобы понять, как в каждом из перечисленных случаев добавление и исключение признака влияет на предсказание. Но рассматривая влияние в каждом конкретном случае, мы получаем огромное количество предельных эффектов, что тяжело интерпретируется. Поэтому можно рассмотреть, какой в среднем оказывает эффект включение признака в модель.

Мысли, которые нужно включить сюда:

1. Как считается value

2. Есть value, есть вклад, есть прогноз – надо понять как вклад связан с value

3. Есть интерпретация – можно вывести из предпосылок формулу value

4. Я не хочу выводить этот алгоритм через аддитивные модели – я хочу попроще написать, а потом указать что она аддитивная и показать, что это

5. По возможности расписать как их находить и что с ними потом делать для интерпретации

6. Расписать разницу SHAP и Shaply values – пока что не очень понятно

2.3.2 Shapley values (значения Шэпли)

2.3.3 Принцип работы

2.3.4 Реализация

3 Данные и модели

3.1 Данные

3.2 Модели

3.3 Попытка интерпретации

4 Анализ результатов

5 Заключение

Список литературы

[1] Interpretable Machine Learning | Christoph Molnar | Christoph Molnar | 2020 | all pages

[2] “Why Should I Trust You?” Explaining the Predictions of Any Classifier | ...