Машинное обучение, ФКН ВШЭ
Семинар №25
Интерпретируемость моделей

1 Введение

Ранее в курсе мы рассматривали различные модели машинного обучения, основной задачей которых является сделать предсказание, «хорошее» с точки зрения некоторых функционалов качества. Чаще всего модели не могут сделать идеальных предсказаний (из-за шума в данных или отсутствия реальной функциональной зависимости между признаками и целевой переменной). В некоторых задачах ошибка не так критична (например, в ранжировании фильмов в рекомендациях) и может вызвать лишь небольшой негатив у пользователя. Однако есть задачи, в которых цена — человеческие жизни или хотя бы большие суммы денег.

В случае, когда от модели зависит выбор способа лечения при диагностировании заболевания, становится важным понимать, почему модель приняла то или иное решение. По законам некоторых стран нельзя принимать решение о выдаче кредита на основании модели, объяснить которую невозможно.

Также иногда в процессе разработки модели может быть полезно лучше понимать, на какие данные она «обращает внимание». Так можно найти ошибки в реализации подсчёта признаков (признаки не используются или используются не так, как ожидалось) или найти ошибки в данных (например, утечки целевой переменной).

На этом семинаре рассмотрим несколько подходов к интерпретации моделей.

2 Интерпретация, основанная на особенностях моделей

Некоторые модели из-за своих особенностей позволяют понять, какие признаки вносят наибольший вклад при построении предсказаний. Рассмотрим несколько из них.

§2.1 Линейные модели

Естественной важностью признаков в линейных моделях являются веса признаков (после маштабирования). По абсолютному значению можно судить о силе влияния на предсказание, а по знаку — на направление. Однако в случае большого количества признаков или при наличии взаимосвязи между признаками могут

быть искажения (например, два скоррелированных признака разделят между собой важность, а иногда один из них может иметь противоположный знак).

§2.2 Решающие деревья

Исключительной особенностью деревьев является простота их интерпретации (кроме того, их можно еще и визуализировать), однако при переходе к композициям над решающими деревьями (например, случайному лесу или бустингу) интерпретировать и визуализировать тысячи деревьев уже не представляется возможным.

§2.3 Нейронные сети

Для нейронных сетей, строящих предсказания для изображений, можно строить карты уверенности предсказаний. Для этого «закрывается» некоторый элемент исходного изображения и сравнивается уверенность предсказания по сравнению с исходным изображением — чем сильнее она уменьшилась, тем важнее для предсказания пиксели из «закрытой» части изображения, поэтому разность уверенностей можно использовать в качестве важностей различных элементов исходного изображения. На основании этих разностей можно строить тепловую карту важности пикселей для предсказания (пример на рис. 1). В качестве «закрывания» части изображения можно использовать её размытие или замену пикселей из этой части на однотонные.

Рис. 1. Пример построения тепловой карты на изображении для нейронной сети

3 LIME

Рассмотрим метод **LIME** (*Local Interpretable Model-Agnostic Explanations*), основная идея которого, как следует из названия, заключается в том, чтобы интерпретировать предсказания некоторой **объясняемой** модели $a(x)$ для заданного объекта x^* в его окрестности.

Предполагается, что полученная интерпретация может использоваться широким кругом лиц для принятия решений (например, докторами для принятия решений о лечении пациентов), поэтому для каждого объекта наряду с признаковым описанием x, используемым в модели $a(x)$, вводится его *интерпретируемое* представление $\bar{x}$, а результатом работы метода является **объясняющая** модель $\bar{a}(\bar{x})$, строящая предсказания именно для этих интерпретируемых представлений (а не для исходных признаковых описаний объекта). В связи с этим в качестве интерпретируемых представлений обычно рассматривают достаточно простые бинарные признаковые описания исходных объектов, например:

- для текстовых данных можно использовать «мешок слов» с ограничением на количество слов или N-грамм, используемых в представлении;
- для изображений используется аналогичное представление, но вместо слов выступают *суперпиксели* — непрерывные области «похожих» пикселей, которые могут быть найдены на изображении x при помощи любого стандартного метода сегментации изображений.

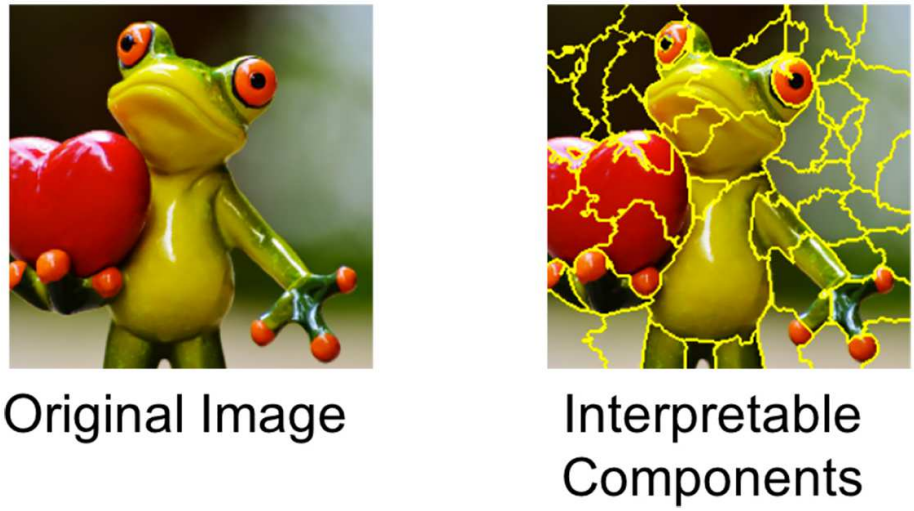

Рис. 2. Пример разбиения изображения на суперпиксели

Для построения объясняющей модели $\bar{a}(\bar{x})$ составляется «суррогатная» выборка $X^{\ell}_{x^*} = \{(\bar{x}_i, y_i)\}\}$ следующим образом: создадим объект $\bar{x}_i$ путем случайного обнуления случайного количества единиц (все случайности — согласно равномерным распределениям) в интерпретируемом представлении $\bar{x}^*$ объекта, для которого строится интерпретация, после чего перейдем от представления $\bar{x}_i$ нового объекта к признаковому описанию x_i в исходном признаковом пространстве и положим $y_i = a(x_i)$. Таким образом, мы создали искусственную выборку в окрестности интерпретируемого представления $\bar{x}^*$, после чего для каждого объекта вычислили таргет как прогноз интерпретируемой модели $a(x)$ в исходном признаковом пространстве (в случае многоклассовой классификации интерпретация чаще всего строится независимо для каждого класса).

Обратим внимание на переход от интерпретируемого представления $\bar{x}_i$ к исходному признаковому описанию x_i: для текстов под «обнулением» элементов представления понимается удаление соответствующих слов или N-грамм из исходного текста и вычисление нового признакового описания для изменённого текста; для изображений аналогично подразумевается «закрытие» соответствующих суперпикселей изображения.

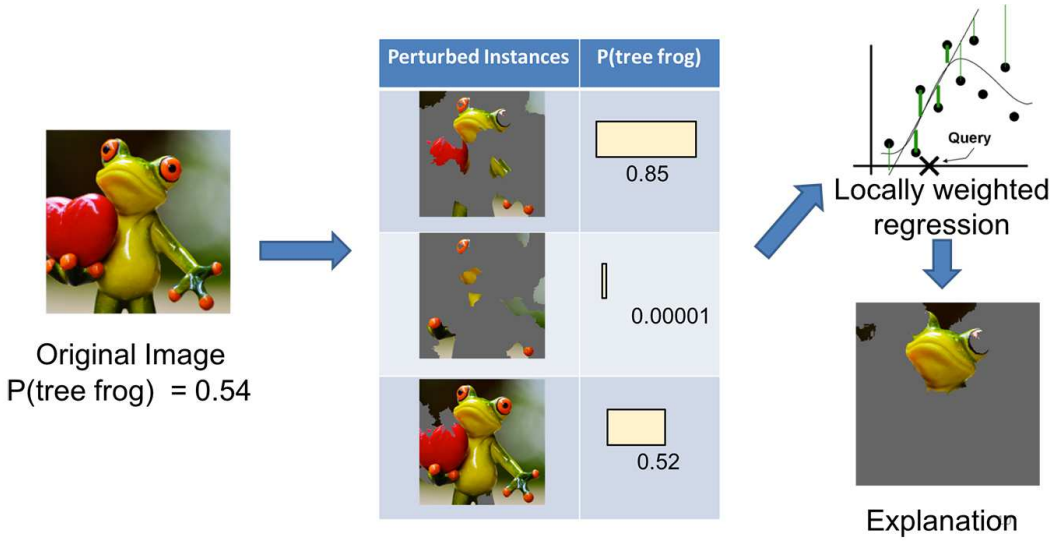

Рис. 3. Пример генерации объектов для суррогатной выборки: из исходного изображения случайным образом удаляется часть суперпикселей, для полученного изображения вычисляется прогноз модели, после чего на итоговой взвешенной выборке интерпретируемых бинарных представлений обучается объясняющая модель

Поскольку итоговая модель $\bar{a}(\bar{x})$ должна, во-первых, интерпретировать предсказания в окрестности исходного объекта, и, во-вторых, быть достаточно простой, то она может быть найдена как решение, например, следующей задачи:

$$\bar{a} = \arg\min_{b \in B} \sum_{\bar{x}_i \in X^{\ell}_{x^*}} \pi_{x^*}(x_i)(a(x_i) - b(\bar{x}_i))^2 + \Omega(b),$$

где x^* — объект, для которого происходит поиск интерпретации,

B — семейство возможных объясняющих моделей (ещё раз обратим внимание на то, что эти модели строят предсказания для интерпретируемых представлений, а не для исходных признаковых описаний),

$\pi_{x^*}(x_i)$ — вес объекта в функционале ошибки (часто в качестве $\pi_{x^*}(x)$ выбирают некоторое ядро в центром в объекте x^*),

$\Omega(b)$ — сложность объясняющей модели.

В качестве семейства объясняющих моделей B можно выбирать любое семейство достаточно простых моделей (например, линейные модели, решающие деревья или решающие списки), в качестве функции потерь можно также выбирать любую другую вместо квадратичной, используемой в формуле выше. Частный случай LIME

при использовании квадратичной функции потерь с экспоненциальным ядром в качестве функции $\pi_{x^*}(x)$, а также семейства B линейных моделей и $\Omega(b) = \infty[\|w_b\|_0 > K]$, где w_b — вектор весов для модели b (то есть с ограничением количества ненулевых весов модели), называется методом *разреженных линейных представлений* (*Sparse Linear Explanations*) и на практике используется чаще всего.

После нахождения объясняющей модели $\bar{a}(\bar{x})$ в силу того, что она строит прогнозы для бинарных векторов, а также того, что является достаточно простой в силу выбора семейства и наличия регуляризации $\Omega(b)$, она может быть легко интерпретирована — например, в случае метода Sparse Linear Explanations в качестве интерпретации достаточно предъявить список признаков с ненулевыми весами (в случае текстов и изображений эти признаки говорят о наличии слов/N-грамм и суперпикселей соответственно).

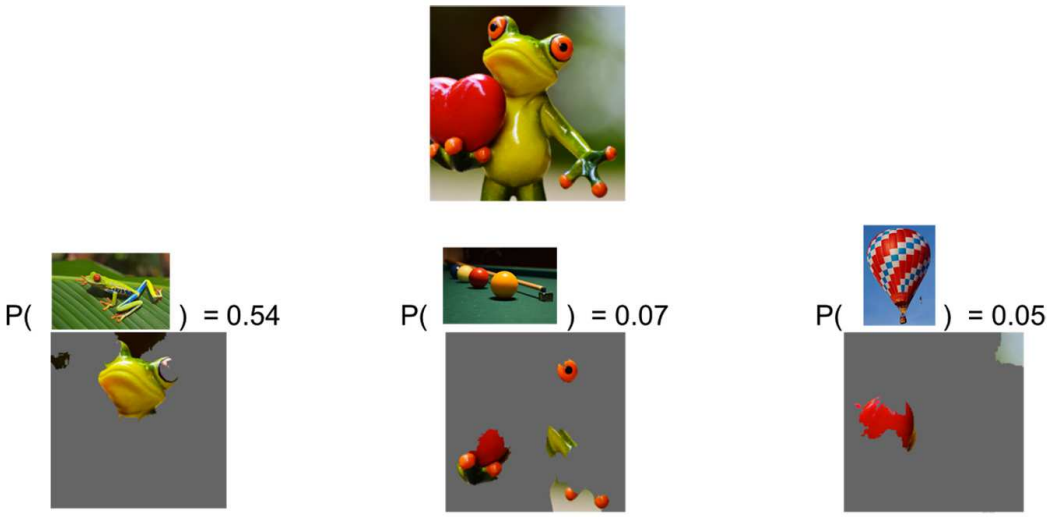

Рис. 4. Пример работы LIME для описания 3 классов

4 Influential Instances (влиятельность объектов)

Альтернативой подходу с оценкой влияния признаков и их изменения на предсказания объясняемой модели является подход с оценкой влияния объектов обучающей выборки на построенную модель. Существует два подхода для оценки влияния объектов: через удаление объекта из обучающей выборки и через оценку влияния на функцию потерь.

Оценка влияния объектов на модель помогает ответить на вопросы об устойчивости моделей (хорошая модель не должна сильно зависеть от отдельных объектов) и найти особенности в данных, которые могут мешать настройке модели (модели, плохо устойчивые к выбросам, сильно изменяются при их удалении).

§4.1 Диагностика через удаление

Для оценки влиятельности объекта обучим модель 2 раза: на полной обучающей выборке и на обучающей выборке без этого объекта, после чего оценим, насколько полученные модели отличаются друг от друга. Если у модели есть явный вектор параметров, то можно посчитать отличия в векторах параметров (например, как норму разности). Более универсальный подход заключается в сравнении предсказаний двух моделей, например, с помощью *расстояния Кука* (*Cook's distance*). Для задачи регрессии влияние объекта x_i обучающей выборки X^ℓ можно оценить следующим образом:

$$D_i = \frac{\sum_{j=1}^{\ell}(a(x_j) - a^{-i}(x_j))^2}{d \times \text{MSE}(a, X^\ell)},$$

где $a(x)$ — модель, обученная на полной обучающей выборке X^ℓ, $a^{-i}(x)$ — модель, обученная на выборке $X^\ell \setminus \{(x_i, y_i)\}$, d — количество признаков, $\text{MSE}(a, X^\ell)$ — исходная квадратичная ошибка модели $a(x)$ на выборке X^ℓ.

На практике поиск влиятельных объектов может быть малоинформативным, поскольку сложно интерпретировать таблицу с признаками десяти самых влиятельных объектов. Однако можно построить простую модель (например, решающее дерево), которая будет детектировать эти влиятельные объекты относительно всех остальных объектов. Так можно будет увидеть, чем эти влиятельные объекты отличаются от остальной выборки.

Также можно оценивать влиятельность конкретного объекта на предсказание для другого объекта, оценивая разность от предсказаний двух моделей.

§4.2 Функции влияния (influence functions)

Недостатком подхода с удалением является его сложность — на практике обучить модели по количеству объектов в обучающей выборке не представляется возможным. Альтернативный вариант предлагает для моделей с дифференцируемой функцией потерь исследовать влияние изменения веса для конкретного примера обучающей выборки на параметры модели через влияние на функцию потерь.

Обозначим за $\hat{\theta}$ вектор параметров модели, обученной на выборке X^ℓ, за $\hat{\theta}_{\varepsilon,x_i}$ — вектор параметров модели при увеличении веса объекта x_i на ε:

$$\hat{\theta}_{\varepsilon,x_i} = \arg\min_{\theta} \frac{1}{\ell}\sum_{i=1}^{\ell} L(x_i, \theta) + \varepsilon L(x_i, \theta)$$

Заметим, что $\hat{\theta}_{0,x_i} = \hat{\theta}$. Тогда влияние увеличения веса объекта x_i на ошибку на новом объекте x можно вычислить следующим образом:

$$\begin{aligned} I_{\text{up,loss}}(x_i, x) = \frac{\partial L(x, \hat{\theta}_{\varepsilon,x_i})}{\partial \varepsilon}\bigg|_{\varepsilon=0} &= \nabla_\theta L(x, \hat{\theta}_{0,x_i})^\top \frac{d\hat{\theta}_{\varepsilon,x_i}}{d\varepsilon}\bigg|_{\varepsilon=0} = \\ &= -\nabla_\theta L(x, \hat{\theta})^\top H_{\hat{\theta}}^{-1} \nabla_\theta L(x_i, \hat{\theta}), \end{aligned} \tag{1}$$

где $H_{\hat{\theta}} = \frac{1}{\ell}\sum_1^\ell \nabla^2_\theta L(x_i, \hat{\theta})$ — гессиан функции потерь L, который может быть вычислен приближённо. Кроме того, в последнем переходе мы воспользовались классическим результатом теории функций влияния о влиянии веса объекта на параметры модели, полное доказательство которого можно найти в [1]:

$$\left.\frac{d\hat{\theta}_{\varepsilon,x_i}}{d\varepsilon}\right|_{\varepsilon=0} = -H^{-1}_{\hat{\theta}} \nabla_\theta L(x_i, \hat{\theta}).$$

Интуиция за этой формулой 1 следующая. Представим гессиан равным единичной матрице, тогда положительное значение $I_{\text{up,loss}}(x_i, x)$ (т.е. ухудшение качества предсказания на объекте x) означает противоположные направления градиентов функции потерь для объектов x_i и x (то есть объект x_i «мешает» построению хорошего прогноза на x).

Использовать функции влияния можно несколькими способами:

1. Сравнивать модели между собой. Как в примере с изображением, может получиться так, что одна из моделей ищет более сложные паттерны на изображении по сравнению с другой. Для этого нужно сравнивать между собой наиболее влиятельные изображения для некоторого примера выборки.

2. Детектирование несовпадений доменов между обучающим и тестовым множеством. Можно найти ложное срабатывание модели и изучить влиятельные объекты для этого ложного предсказания. Так можно выяснить паттерны в данных, которые мешают корректной работе алгоритма на новом домене данных (данные из несколько другого распределения).

3. Коррекция обучающих данных. Если у нас есть возможность перепроверить корректность разметки небольшого числа объектов обучающей выборки, то эффективнее сделать это на наиболее влиятельных объектах, так как именно они влияют на нашу модель сильнее всего.

5 Состязательные атаки

Кроме интерпретации модели для оценки её устойчивости через признаки или объекты обучающей выборки, можно оценивать корректность модели через её устойчивость к состязательным (*adversarial*) объектам. Состязательными объектами называют такие объекты, предсказания на которых радикально меняются при малом изменении исходных данных (например, после наложения шума на корректно классифицируемое изображение панды нейросеть может уверенно относить новое изображение к совершенно другому классу).

Методы генерации таких изображений называются состязательными атаками (*adversarial attacks*). Целью атак является изменение заданного объекта таким образом, чтобы изменение было не слишком «заметным» и чтобы при этом изменить предсказание заданной модели для него. Чтобы контролировать «заметность» изменения, исходное и измененное изображение обычно сравнивают по некоторой метрике

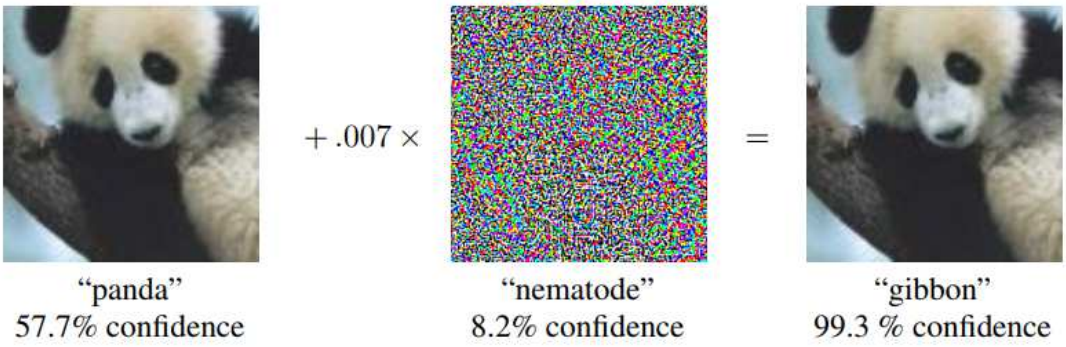

Рис. 5. Пример состязательного изображения

(чаще всего в качестве метрики выбирают l_∞, то есть учитывается максимальное по всем пикселям изменение исходного изображения).

В основном все методы делятся на две группы: *white-box* (с доступом к градиенту модели) и *black-box* (как мог догадаться смышлёный читатель, без доступа к градиенту модели). Кроме того, атаки делятся на ненаправленные (*non-targeted*), целью которых является заставить модель строить *какой угодно* некорректный прогноз для измененного объекта (например, отнести изображение кошки к любому классу, отличному от кошек), и направленные (*targeted*), целью которых является заставить модель строить некий *определенный* некорректный прогноз (например, отнести изображение к конкретному классу «собака»).

В рамках атаки, можно, например, можно генерировать объект, решая следующую оптимизационную задачу:

$$L(a(x^{adv}), y_{\text{target}}) + \lambda \|x^{adv} - x\| \to \min_{x^{adv}},$$

где x — атакуемое изображение, x^{adv} — состязательный объект, который мы хотим построить на основе x, L — функция потерь нашей модели, $a(x)$ — модель, y_{target} — желаемая метка объекта x^{adv}.

Пусть x — исходное изображение, для которого производится атака, y_{true} — его истинная метка. Разберем несколько широко используемых состязательных white-box атак.

1. **Fast gradient sign method** (FGSM)
   Ненаправленный метод атак, идея которого заключается в том, чтобы изменить объект в сторону градиента функции потерь L для истинной метки объекта:

   $$x^{adv} = x + \varepsilon \operatorname{sign}(\nabla_x L(x, y_{\text{true}})).$$

   Заметим, что при таком способе изменения исходного изображения l_∞-норма их разности не будет превышать ε, поэтому мощность атаки может регулироваться этим гиперпараметром. Данный метод является ненаправленным, а также относится к т.н. одношаговым (*one-shot*) методам.

2. **Targeted fast gradient sign method** (T-FGSM)
   Направленная версия *FGSM*, в которой объект изменяется в сторону антиградиента функции потерь для целевой метки атаки y_{target} :

   $$x^{adv} = x - \varepsilon \operatorname{sign}(\nabla_x L(x, y_{\text{target}})).$$

3. **Iterative fast gradient sign method** (I-FGSM)
   В отличие от предыдущих методов, являющихся одношаговыми, в данном вместо одного шага длины ε делается T шагов длины $\alpha = \frac{\varepsilon}{T}$:

   $$x_0^{adv} = x,$$
   $$x_{t+1}^{adv} = x_t^{adv} + \alpha \operatorname{sign}(\nabla_x L(x, y_{\text{true}}))$$

В случае с black-box атаками необходимо обучить суррогатную модель на выборке, таргетами в которой являются предсказания атакуемой модели («чёрного ящика»), после чего для полученной модели проводится атака white-box методами.

Список литературы

[1] R Dennis Cook and Sanford Weisberg. *Residuals and influence in regression*. New York: Chapman and Hall, 1982.

[2] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. "why should i trust you?"explaining the predictions of any classifier. In *Proceedings of the 22nd ACM SIGKDD international conference on knowledge discovery and data mining*, pages 1135–1144, 2016.