

• Прикладные модели машинного обучения • Интерпретируемость и объяснимость

Воронцов Константин Вячеславович

`k.v.vorontsov@phystech.edu`

`http://www.MachineLearning.ru/wiki?title=User:Vokov`

Этот курс доступен на странице вики-ресурса

`http://www.MachineLearning.ru/wiki`

«Машинное обучение (курс лекций, К.В.Воронцов)»

МФТИ • 24 апреля 2026

1 Интерпретируемость и объяснимость

- Цели, задачи, основные понятия
- Интерпретируемые модели
- Визуальные методы интерпретации

2 Интерпретация в пространстве признаков

- Методы оценивания важности признаков
- Методы LIME и Anchors
- Методы SHAP и SAGE

3 Интерпретация в пространстве объектов

- Вектор Шепли для объектов
- Метод Gradient Shapley
- Контрфактическое объяснение

Объяснимость (XAI, eXplainable Artificial Intelligence)

Interpretability — пассивная интерпретируемость устройства модели или предсказания на объекте

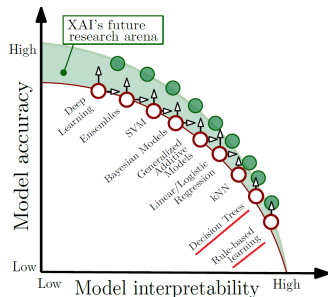
Explainability — активная генерация объяснений как дополнительных выходных данных для объекта

Comprehensibility — возможность представить выученные закономерности в виде понятного людям знания

Understandability, Transparency — прозрачность всего процесса создания модели, её частей, промежуточных результатов

“Do you want an interpretable model, or the one that works?”

[Yann LeCun, NIPS'17]



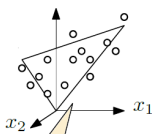
Объяснимость — кому, зачем и какие нужны объяснения?

- **Кому:** экспертам предметной области
Зачем: доверие к моделям, получение знаний из данных
- **Кому:** конечным пользователям
Зачем: понимание причин принимаемых решений
- **Кому:** регуляторам
Зачем: аудит соответствия моделей стандартам и нормам
- **Кому:** исследователям, разработчикам
Зачем: понимание свойств моделей, продуктов и сервисов
- **Кому:** бенефициарам, менеджерам
Зачем: понимание влияния моделей на бизнес-процессы

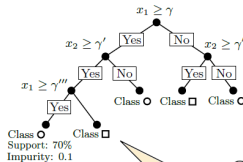
A.B.Arrieta et al. Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI. 2019.

Интерпретируемые модели машинного обучения

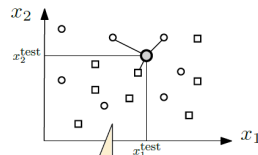
$$y = w_1x_1 + w_2x_2 + w_0$$



линейные модели:
вес показывает, на сколько
изменится y при $x_j + 1$



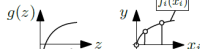
решающие деревья:
путь из корня объясняет,
почему принято такое решение



метрические классификаторы:
ближайшие соседи объясняют,
почему принято такое решение

$$g(\mathbb{E}(y)) = w_1f_1(x_1) + w_2f_2(x_2)$$

$\mathbb{E}(y)$: expected value



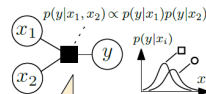
обобщённые линейные
модели и LR: объяснение
изменения вероятности $p(y)$



Training
dataset

- If x_1 is high then $y = \circ$
- If x_1 is low and x_2 is high then $y = \circ$
- If x_2 is low then $y = \square$

индукция правил:
объяснение решения
на естественном языке



байесовские сети и NB:
объяснение зависимостей
между переменными

A.B.Arrieta et al. Explainable Artificial Intelligence (XAI): Concepts, Taxonomies, Opportunities and Challenges toward Responsible AI. 2019.

Напоминание. Многомерная линейная регрессия

Модель линейной регрессии на n признаках $f_1(x), \dots, f_n(x)$:

$$f(x, \alpha) = \sum_{j=1}^n \alpha_j f_j(x), \quad \alpha \in \mathbb{R}^n$$

Метод наименьших квадратов, обучение по выборке $(x_i, y_i)_{i=1}^\ell$:

$$Q(\alpha) = \sum_{i=1}^{\ell} (f(x_i, \alpha) - y_i)^2 = \|F\alpha - y\|^2 \rightarrow \min_{\alpha}$$

$\alpha^* = (F^T F)^{-1} F^T y$ — решение задачи НК, $F = (f_j(x_i))_{\ell \times n}$

Коэффициент детерминации $R^2 \in [0, 1]$, чем выше, тем лучше:

$$\begin{aligned} R^2 &= 1 - \frac{\min_{\alpha} \|F\alpha - y\|^2}{\min_c \|c - y\|^2} = 1 - \frac{\|F\alpha^* - y\|^2}{\|\bar{y} - y\|^2} = \frac{y^T F\alpha^* - n\bar{y}^2}{y^T y - n\bar{y}^2} \\ &= y^T F\alpha^* = \sum_{j=1}^n \alpha_j^* (f_j^T y) \quad \text{при } y^T y = 1, \bar{y} = 0 \end{aligned}$$

Оценки значимости признаков в линейной регрессии

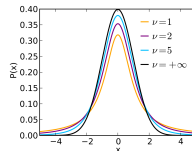
- Коэффициент α_j^* равен изменению f при увеличении f_j на 1
 - не учитывается масштаб, сдвиг, дисперсия, корреляции, мультиколлинеарность признаков (источник переобучения)
- *t-статистика значимости признака (feature importance)*
 - учитывает дисперсию оценки α_j^* :

$$T_j = \frac{\alpha_j^*}{\hat{\sigma} \sqrt{(F^T F)^{-1}_{jj}}} \sim t_{\ell-n}, \quad \hat{\sigma}^2 = \frac{Q(\alpha^*)}{\ell-n}$$

- позволяет проверять гипотезу $\alpha_j^* = 0$,
- вычислять p-value для этой гипотезы,
- доверительные интервалы для α_j^* .

- Чистый эффект (net effect) NEF_j признака в разложении R^2 :

$$R^2 = y^T F \alpha^* = \sum_{j=1}^n \alpha_j^* (f_j^T y) = \sum_{j=1}^n \text{NEF}_j \quad (\text{при } y^T y = 1, \bar{y} = 0)$$



t-распределение
Стьюдента с $\nu = \ell - n$
степенями свободы

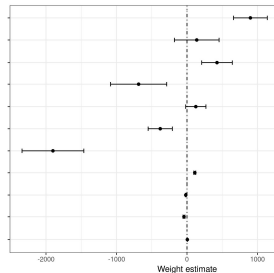
Пример. Задача прогнозирования аренды велосипедов

x_i — дата, y_i — число арендованных велосипедов

Weight = α_j^* ; Standard Error SE = $\hat{\sigma}\sqrt{(F^T F)_{jj}^{-1}}$; t = $|T_j|$

Intercept — свободный член, коэффициент при признаке $f_1 = 1$

	Weight	SE	t
(Intercept)	2399.4	238.3	10.1
season SUMMER	899.3	122.3	7.4
season FALL	138.2	161.7	0.9
season WINTER	425.6	110.8	3.8
holiday	-686.1	203.3	3.4
workingday	124.9	73.3	1.7
weathersit MISTY	-379.4	87.6	4.3
weathersit RAIN/SNOW/STORM	-1901.5	223.6	8.5
temp	110.7	7.0	15.7
hum	-17.4	3.2	5.5
windspeed	-42.5	6.9	6.2
days_since_2011	4.9	0.2	28.5



UCI ML Repo: <http://archive.ics.uci.edu/ml/datasets/Bike+Sharing+Dataset>

Christoph Molnar. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. 2019

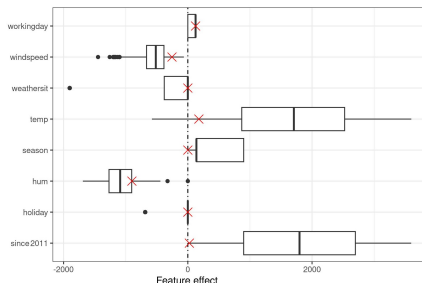
Оценки значимости признаков для фиксированного объекта

- Важность признака (вклад, effect) $\text{eff}_j(x) = \alpha_j f_j(x)$
 - учитывается масштаб, не учитываются сдвиг и корреляции
- Ситуативная важность $\text{situational importance} = \alpha_j (f_j(x) - \bar{f}_j)$
 - учитывается масштаб и сдвиг, не учитываются корреляции

Боксы (boxplot) показывают распределения $\{\text{eff}_j(x_i)\}_{i=1}^{\ell}$

✗ — эффекты на конкретном выбранном объекте x ;

График объясняет, какие признаки обусловили низкий прогноз на данном объекте



Christoph Molnar. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. 2019

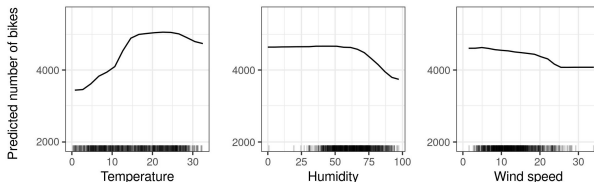
Графики частичной зависимости (Partial Dependence Plot, PDP)

Как модель $f(x)$ зависит от части признаков $S \subseteq \{f_1, \dots, f_n\}$?
 $x = (x_S, \tilde{x})$, x_S — признаки из S , $\tilde{x}$ — остальные признаки.

Оценивание интеграла методом Монте-Карло:

$$g(x_S) = E_{\tilde{x}} f(x_S, \tilde{x}) = \int f(x_S, \tilde{x}) dP(\tilde{x}) \text{ или } \dots dP(\tilde{x} | x_S)$$

$$\hat{g}(x_S) = \frac{1}{\ell} \sum_{i=1}^{\ell} f(x_S, \tilde{x}_i) \text{ или } \hat{g}(x_S) = \frac{\sum_i K(x_S, x_{Si}) f(x_S, \tilde{x}_i)}{\sum_i K(x_S, x_{Si})}$$

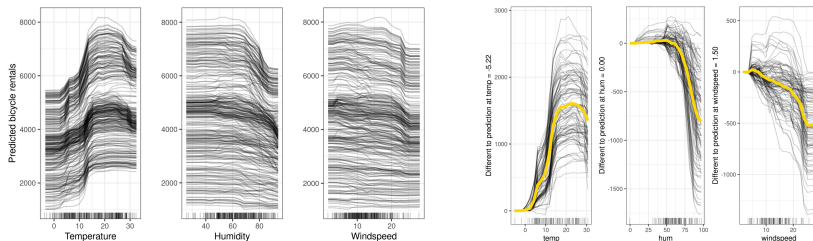


Christoph Molnar. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. 2019

Графики индивидуальных условных зависимостей (ICE)

Individual Conditional Expectation: PDP по отдельным объектам
 $f(x) = f(x_S, \tilde{x})$, x_S — признаки из S , $\tilde{x}$ — остальные признаки.

График зависимости $g_i(x_S) = f(x_S, \tilde{x}_i)$ для каждого $x_i = (x_{Si}, \tilde{x}_i)$:



Показывает, как изменится предсказание модели на объекте, если изменять значение выбранного признака x_S , $|S|=1$.

Christoph Molnar. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. 2019

Перестановочные оценки значимости признаков

Перестановочная оценка PFI (permutational feature importance)

$$PFI_j = Q^j / Q \text{ или } Q^j - Q$$

потери на исходной выборке:

$$Q = \sum_i \mathcal{L}(f(x_i), y_i)$$

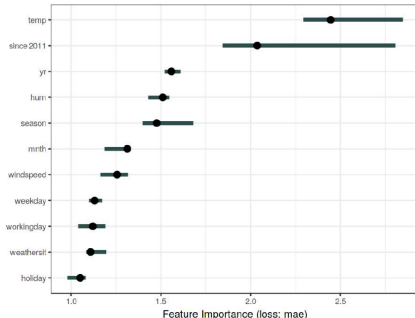
потери после перемешивания:

$$Q^j = \sum_i \mathcal{L}(f(\tilde{x}_i^j), y_i)$$

где $f(x)$ — обученная модель,

$\mathcal{L}(f, y)$ — функция потерь,

$\tilde{x}_i^j$ = замена ($f_j(x_i) \rightarrow f_j(x_{\text{rand}})$).



⊕ любая модель ⊕ однократное обучение ⊕ учёт корреляций
⊖ перемешивание может порождать нереалистичные объекты

Christoph Molnar. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. 2019

Вектор Шепли (из теории кооперативных игр)

Признаки $F = \{f_1, \dots, f_n\}$ играют в кооперативную игру
 $V(S)$ — совместный выигрыш коалиции $S \subseteq F$, $V(\emptyset) = 0$

Игроки вступают в S по очереди, задаваемой перестановкой π
 $\Delta(j, S) = V(S \cup j) - V(S)$ — полезность игрока f_j в коалиции S
 $S_{\pi j} \subset F$ — множество игроков, идущих перед f_j в перестановке π

Вектор Шепли ϕ — справедливый делёж общего выигрыша:

$$\phi_j = \frac{1}{n!} \sum_{\pi} \Delta(j, S_{\pi j}) = \sum_S \frac{|S|! (n - |S| - 1)!}{n!} \Delta(j, S)$$

$|S|!$ — число способов образовать коалицию S

$(n - |S| - 1)!$ — число способов продолжить образование коалиции после присоединения f_j к S

$n!$ — число перестановок π множества n игроков

Lloyd Stowell Shapley. A value for n-person games. 1952

Свойства вектора Шепли

Теорема

Это единственный способ делёжа, удовлетворяющий аксиомам:

- ① эффективность:

$$\sum_{j=1}^n \phi_j = V(F)$$

- ② симметричность (анонимность игроков):

$$\forall S, j, k \Delta(j, S) = \Delta(k, S) \Rightarrow \phi_j = \phi_k$$

- ③ невозможность халявы для «болвана»:

$$\forall S, j \Delta(j, S) = 0 \Rightarrow \phi_j = 0$$

- ④ состоятельность (Δ_1, Δ_2 — две функции полезности):

$$\forall S, j \Delta_1(j, S) \leq \Delta_2(j, S) \Rightarrow \phi_{1j} \leq \phi_{2j}$$

- ⑤ аддитивность (по источникам выигрыша):

$$\forall S \ V(S) = \alpha_1 V_1(S) + \alpha_2 V_2(S) \Rightarrow \forall j \ \phi_j = \alpha_1 \phi_{1j} + \alpha_2 \phi_{2j}$$

Lloyd Stowell Shapley. A value for n-person games. 1952

Эмпирическое оценивание вектора Шепли

Несмещённая оценка вектора Шепли методом Монте-Карло:
 Π — случайное подмножество перестановок; для каждой $\pi \in \Pi$
в модель инкрементно добавляются признаки $\pi(j)$, $j = 1, \dots, n$:

$$\hat{\phi}_j = \frac{1}{|\Pi|} \sum_{\pi \in \Pi} \Delta(j, S_{\pi j})$$

Что считать выигрышем $V(S)$ признаков $S \subseteq \{f_1, \dots, f_n\}$:

- Коэффициент детерминации $V(S) = R_S^2$,
модель дообучается при добавлении каждого признака
- *Shapley regression value* $V(S) = f_S(x)$ на объекте x ,
где модель f_S обучена только на признаках из S
- *Shapley sampling value* $V(S) = E_{\tilde{x}} f(x_S, \tilde{x})$, где $x = (x_S, \tilde{x})$
 $E_{\tilde{x}}$ — среднее по объектам выборки $x_i = (x_{Si}, \tilde{x}_i)$: $x_{Si} \approx x_S$

E.Štrumbelj, I. Kononenko. Explaining prediction models and individual predictions with feature contributions. 2014

Вектор Шепли для признаков в линейной регрессии

$\alpha = (F^T F)^{-1} F^T y$ — решение задачи $\|F\alpha - y\|^2 \rightarrow \min_{\alpha}$.

Разложение коэффициента детерминации на *чистые эффекты*:

$$R^2 = \sum_{j=1}^n \alpha_j (f_j^T y) \quad (\text{при } y^T y = 1, \bar{y} = 0)$$

Разложение R^2 по значениям Шепли признаков, при $V(S) = R_S^2$:

$$R^2 = \sum_{j=1}^n \phi_j$$

Приравнивая эффекты, $\phi_j = \alpha_j (f_j^T y)$, получаем $\alpha_j = \phi_j / (f_j^T y)$

Преимущество оценок Шепли α_j для линейной регрессии:

- не подвержены мультиколлинеарности, более устойчивы
- коэффициенты интерпретируемы по знаку и величине
- могут иметь смещение, но оно незначительно

Суррогатное моделирование в окрестности объекта x

$(x_i, y_i)_{i=1}^{\ell}$ — обучающая выборка, $\mathcal{L}(f, y)$ — функция потерь
 $f(x, \alpha)$ — неинтерпретируемая модель, обученная по выборке:

$$\sum_{i=1}^{\ell} \mathcal{L}(f(x_i, \alpha), y_i) \rightarrow \min_{\alpha}$$

$g_x(z, \beta)$ — интерпретируемая *суррогатная модель* для аппроксимации f в окрестности объясняемого объекта x :

$$\sum_{i=1}^k w_{xi} \mathcal{L}(g_x(z_i, \beta), f(z_i, \alpha)) + \Omega(\beta) \rightarrow \min_{\beta}$$

$(z_i)_{i=1}^k \sim \pi(K_h(z, x))$ — *суррогатные объекты*, сэмплируемые из радиального распределения с центром в x и радиусом h

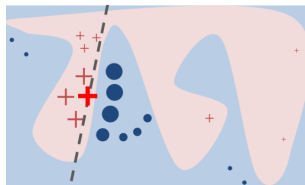
$w_{xi} = K_h(z, x)$ — веса объектов в h -окрестности объекта x

$K_h(z, x)$ — функция близости (kernel) радиуса h

$\Omega(\beta)$ — регуляризатор, штраф за сложность модели $g_x(z, \beta)$

Метод LIME (Local Interpretable Model-agnostic Explanations)

$g_x(z, \beta) = \sum_{j=1}^m \beta_j b_j(z)$ — локальная линейная аппроксимация



- 1 фиксируется **объект x**, для которого требуется объяснение
- 2 синтезируются *суррогатные объекты* z_i в его окрестности
- 3 на них вычисляются значения основной модели $f(z_i, \alpha)$
- 4 строится локальная аппроксимация *суррогатной моделью*
- 5 для объекта x строится объяснение и его визуализация

M. Ribeiro, S. Singh, C. Guestrin. "Why should I trust you?" Explaining the predictions of any classifier. 2016

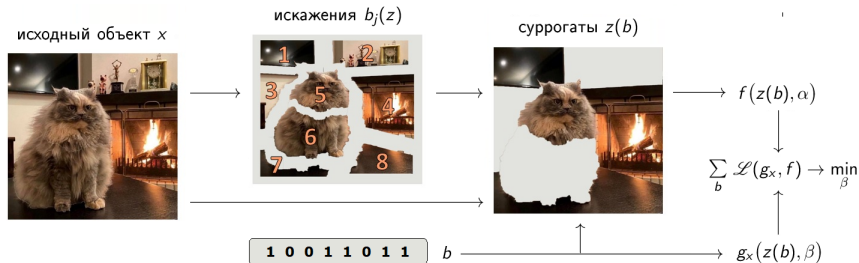
Метод LIME: синтез суррогатных объектов

$g_x(z, \beta) = \sum_{j=1}^m \beta_j b_j(z)$ — локальная линейная аппроксимация

Признаки $b_j(z) = [j\text{-го искажения объекта } x \text{ в суррогате } z \text{ нет}]$

Синтез суррогата $z(b) = \text{применить к } x \text{ все искажения } j: b_j = 0$

Синтез выборки суррогатов $(z_i)_{i=1}^k$ — по случайным $b_j \in \{0, 1\}$



Олег Седухин. Интерпретация моделей и диагностика сдвига данных: LIME, SHAP и Shapley Flow. 2022-01-13. <https://habr.com/ru/companies/ods/articles/599573>

Метод LIME: интерпретируемость суррогатной модели

$g_x(z, \beta) = \sum_{j=1}^m \beta_j b_j(z)$ — локальная линейная аппроксимация

Примеры интерпретируемых искажений объекта x :

- замена j -го признака в x на пропуск, среднее значение или 0;
- замена части объекта x частью другого объекта;
- выбрасывание j -го слова из текста x , и т.п.

Интерпретируемость линейной модели $g_x(z, \beta)$:

- вес β_j равен изменению g при устранении искажения b_j
- число m не должно быть слишком большим
- не должно быть мультиколлинеарности (регуляризация!)

$\mathcal{L}(g, f) = (g - f)^2$ — квадратичная функция потерь

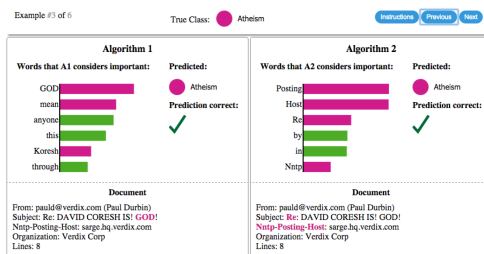
$K_h(z, x) = \exp(-\frac{1}{h^2} \rho^2(z, x))$, где ρ евклидова или косинусная

M. Ribeiro, S. Singh, C. Guestrin. "Why should I trust you?" Explaining the predictions of any classifier. 2016

Пример LIME. Задача классификации текстов (20NewsGroups)

Признаки $b_j(z) = [\text{наличие слова } j \text{ из текста } x \text{ в тексте } z]$

Гистограмма весов β_j : важности слов j для исходного текста x



Модель классификации SVM-RBF имеет точность 94% на тесте, но при различении классов «christianity» и «atheism» считает важными мусорные слова «Posting», «Host», «Re».

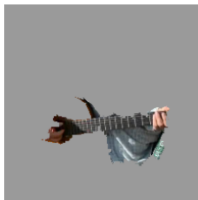
Ясно, в чём проблема, и как её исправлять (фильтровать слова)

Пример LIME. Задача классификации изображений

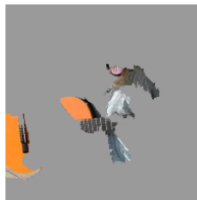
Признаки $b_j(z)_{j=1}^m$ — сегменты (super-pixel) из изображения x
 $m = 10$, признаки конструируются под объясняемый объект x



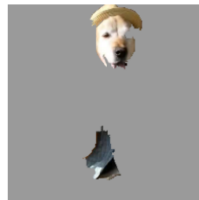
(a) Original Image



(b) Explaining *Electric guitar*



(c) Explaining *Acoustic guitar*



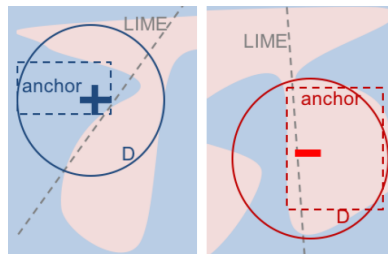
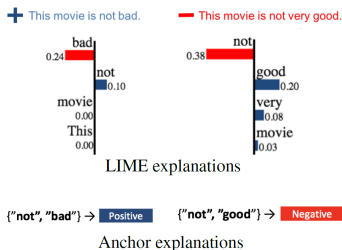
(d) Explaining *Labrador*

Модель классификации — глубокая нейросеть Google Inception
Три наиболее вероятных класса: «electric guitar» ($p = 0.32$),
«acoustic guitar» ($p = 0.24$), «labrador» ($p = 0.21$)

Ясно, почему модель перепутала «acoustic» с «electric»
— из-за грифа, см. рис. (b)

Метод якорей (Anchors): аппроксимация конъюнкциями

$g_x(z, \beta) = \bigwedge_{j \in J} b_j(z)$ — правила-конъюнкции, образуемые небольшим числом бинарных признаков или пороговых условий



Правило стремится покрыть как можно большую область объектов, относящихся моделью f к тому же классу, что x

Обходит LIME по точности, покрытию, качеству объяснений

M.Ribeiro, S.Singh, C.Guestrin. Anchors: high-precision model-agnostic explanations. 2018

Метод SHAP (SHapley Additive exPlanations)

$g_x(z, \beta) = \sum_{j=1}^m \beta_j b_j(z)$ — локальная линейная аппроксимация

Признаки $b_j(z) = [j\text{-го искажения объекта } x \text{ в суррогате } z \text{ нет}]$

Синтез суррогата $z(b) = \text{применить к } x \text{ все искажения } j: b_j = 0$

Приращение $f(z, \alpha)$, если в суррогате $z(b)$ убрать искажение j :

$$\Delta(j, b) = V(b|_{b_j=1}) - V(b|_{b_j=0}), \quad V(b) = f(z(b), \alpha)$$

Три желательных свойства локальной модели $g_x(z, \beta)$

- 1 локальная согласованность аппроксимации в точке x :
 $\forall j \ b_j(z) = 1 \Rightarrow g_x(z, \beta) = f(x, \alpha)$
- 2 бесполезность болвана — признака b_j , пропущенного в x :
 $\forall j \ b_j(x) = 0 \Rightarrow \beta_j = 0$
- 3 состоятельность: с ростом приращения $\Delta(j, b)$ растёт β_j ,
 $\forall b, j \ \Delta_1(j, b) \leq \Delta_2(j, b) \Rightarrow \beta_{1j} \leq \beta_{2j}$

Метод SHAP: теоретическое обоснование

Теорема 1

Единственным распределением весов β_j , удовлетворяющим свойствам ❶ ❷ ❸ является вектор Шепли:

$$\beta_j = \sum_{b \in \{0,1\}^m} \frac{|b|! (m - |b| - 1)!}{m!} \Delta(j, b)$$

где $|b| = \{j: b_j = 1\}$ — число единиц в векторе b .

Теорема 2 (метод Shapley Kernel)

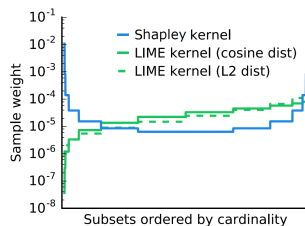
Вектор Шепли (β_j) является решением задачи НК с весами:

$$\sum_{b \in \{0,1\}^m} w_b (g_x(z(b), \beta) - f(z(b), \alpha))^2 \rightarrow \min_{\beta}$$

где w_b — веса 2^m суррогатов, $w_b = \frac{|b|! (m - |b| - 1)!}{m!} \frac{m-1}{|b|} = \frac{1}{m C_{m-2}^{|b|-1}}$

Метод Shapley Kernel: вариант реализации SHAP

- ⊕ Вектор Шепли (β_j) вычисляется взвешенной линейной регрессией
- ⊕ LIME решает ту же задачу, но веса суррогатов w_b задаются эвристически, неоптимально
- ⊕ При больших 2^m векторы b можно сэмплировать из распределения w_b
- ⊕ SHAP лучше LIME в экспериментах, где они сравнивались с тем, как эксперты объясняют решения моделей
- ⊖ Значимость признаков оценивается по нереалистичным (out-of-distribution) суррогатным объектам



Scott Lundberg, Su-In Lee. A unified approach to interpreting model predictions. 2017
E.Kumar, S.Venkatasubramanian, C.Scheidegger, S.A.Friedler. Problems with Shapley-value-based explanations as feature importance measures. 2020

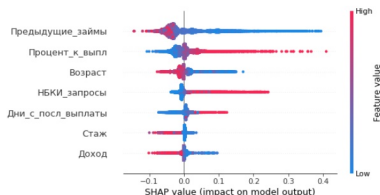
Метод SHAP: пример визуализации

Модель вероятности дефолта $f(x)$, градиентный бустинг

Индивидуальное объяснение для x : $f(x) = 19\%$ при $\bar{y} = 6\%$
Значения Шепли показываются цветом: $\beta_j(x) < 0$, $\beta_j(x) > 0$



Агрегированные объяснения по всей выборке $\{\beta_j(x_i)\}$:



ось X: $\beta_j(x_i)$

ось Y: признаки j

цвет точки: значение признака $f_j(x_i)$

ширина линии $\propto$ число точек

<https://rb.ru/opinion/uzhe-ne-black-box>

Метод SAGE (Shapley Additive Global importanceE)

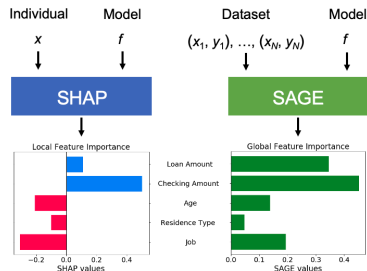
SHAP:

каковы вклады признаков f_j
в предсказание $f(x)$

SAGE:

как качество модели в целом
зависит от признаков f_j

Модификация SHAP → SAGE:



$V(S) = -\mathcal{L}(E_{\tilde{x}} f(x_S, \tilde{x}))$ — раскладываются потери (LossSHAP)

$\phi_j = \frac{1}{\ell} \sum_{i=1}^{\ell} \phi_j(x_i)$ — значения Шепли усредняются по выборке

$\phi_j = \frac{1}{|X|} \sum_{x_i \in X} \phi_j(x_i)$ — или по случайной подвыборке, если долго

Ian C. Covert, Scott Lundberg, Su-In Lee. Understanding Global Feature Contributions With Additive Importance Measures. 2020

Вектор Шепли для объектов: инкрементное обучение

Теперь обучающие объекты играют в кооперативную игру:

$f_S(x)$ — модель, обученная на подвыборке $S \subseteq \{x_1, \dots, x_\ell\}$

$V(S) = -\sum_x \mathcal{L}(f_S(x))$ на тестовых объектах x (hold-out)

$\Delta(i, S) = V(S \cup i) - V(S)$ — полезность обучающего объекта x_i

$\phi_i = \frac{1}{|\Pi|} \sum_{\pi \in \Pi} \Delta(i, S_{\pi i})$ — несмещённая оценка Монте-Карло

для всех перестановок $\pi_t \in \Pi$, $t = 1, \dots, |\Pi|$:

$S := \emptyset$; $v_0 := V(\emptyset)$;

для всех $i = \pi_t(1), \dots, \pi_t(\ell)$:

$S := S \cup \{x_i\}$;

обновить модель $f_S(x)$, дообучив её на объекте x_i ;

оценить модель $v_i := V(S)$;

$\phi_i := \frac{t-1}{t} \phi_i + \frac{1}{t} (v_i - v_{i-1})$;

Встраивание оценок Шепли в онлайнный градиентный спуск

Градиентная минимизация аддитивного критерия:

$$\sum_{i=1}^{\ell} \mathcal{L}(f(x_i, \alpha), y_i) \rightarrow \min_{\alpha}$$

Алгоритм инкрементного обучения Online Gradient Descent:

для всех перестановок $\pi_t \in \Pi$, $t = 1, \dots, |\Pi|$:

$S := \emptyset$; $v_0 := V(\emptyset)$; **инициализировать** α_0 ;

для всех $i = \pi_t(1), \dots, \pi_t(\ell)$:

$S := S \cup \{x_i\}$;

обновить модель $\alpha_i := \alpha_{i-1} - \eta_i \nabla_{\alpha} \mathcal{L}(f(x_i, \alpha_{i-1}), y_i)$;

оценить модель $v_i := V(S)$;

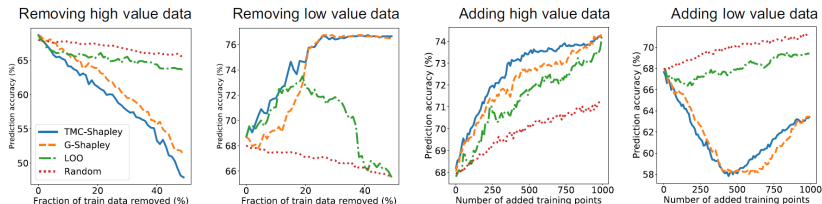
$\phi_i := \frac{t-1}{t} \phi_i + \frac{1}{t} (v_i - v_{i-1})$;

A. Ghorbani, J. Zou. Data Shapley: equitable valuation of data for machine learning. 2019

M. Zinkevich. Online convex programming and generalized infinitesimal gradient ascent. 2003.

Интерпретация объектов с помощью значений Шепли

- низкое ϕ_i — выбросы, такие x_i можно удалять из выборки
- высокое ϕ_i — опорные, пограничные, таких x_i не хватает
- более устойчивая оценка по сравнению с leave-one-out



Задача UCI:BreastCancer

- (1) изъятие из обучения лучших объектов, по убыванию ϕ_i
- (2) изъятие из обучения худших объектов, по возрастанию ϕ_i
- (3) добавление объектов, похожих на лучшие, по убыванию ϕ_i
- (4) добавление объектов, похожих на худшие, по возрастанию ϕ_i

A. Ghorbani, J. Zou. Data Shapley: equitable valuation of data for machine learning. 2019

Задача поиска контрфактов

Контрфакт x' — объект, схожий с x , но существенно отличающийся предсказанием модели $f(x', \alpha^*)$.

- Модель кредитного скоринга выдала отказ.
Какие изменения признаков поменяют решение модели?
(закрыть другие кредиты? переехать в другой город?
сменить работу? изменить структуру расходов?)
- Модель оценила для собственника стоимость аренды.
Какие факторы способны увеличить оценку стоимости?
(улучшить ремонт? разрешить домашних животных?)

Важно: находить реализуемые изменения признаков:

- минимально изменять минимальное число признаков
- выбирать из множества разнообразных контрфактов

Riccardo Guidotti. Counterfactual explanations and how to find them: literature review and benchmarking. 2022

Метод поиска контрфактов (Counterfactual explanations)

Контрфакт x' — объект, минимально отличающийся от x и существенно отличающийся предсказанием модели $f(x', \alpha)$.

В психологии: мысленный альтернативный вариант развития событий.

Оптимизационная задача поиска контрфактов x' с заданным y' :

$$\mathcal{L}(f(x', \alpha), y') + \lambda \|x - x'\|_1 \rightarrow \min_{x'} \min_{\lambda}$$

L_1 -регуляризатор обеспечивает разреженность решения — чем больше λ , тем больше совпадений признаков $x_j = x'_j$:

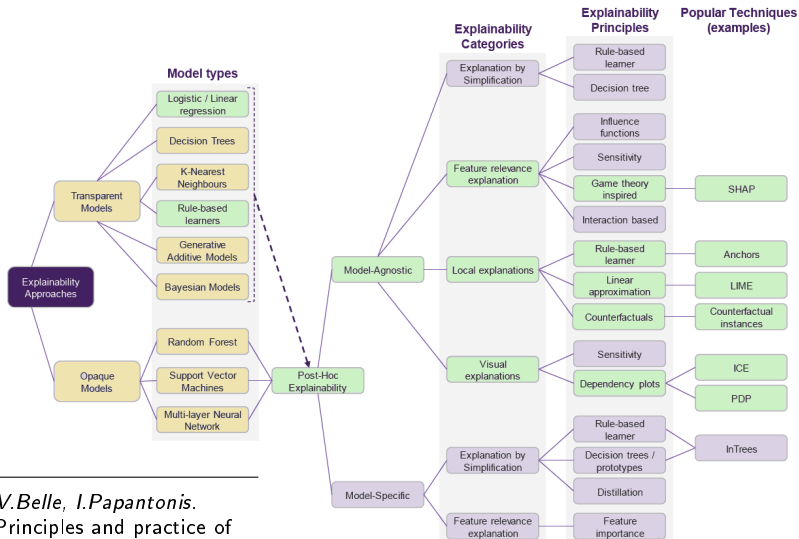
$$\|x - x'\|_1 = \sum_{j=1}^n \frac{|x_j - x'_j|}{\text{MAD}_j},$$

Median Absolute Deviation $\text{MAD}_j = \text{med}_i |x_{ij} - M_j|$, $M_j = \text{med}_i x_{ij}$

Постепенно уменьшая λ , подгоняем $f(x', \alpha) \rightarrow y'$.

S. Wachter, B. Mittelstadt, C. Russell. Counterfactual explanations without opening the black box: Automated decisions and the GDPR. 2017

Подходы к объяснимости моделей



V.Belle, I.Papantonis.

Principles and practice of
explainable machine learning. 2020

- *Интерпретируемость* — прозрачность строения модели, либо понятность её результата на объекте
- Интерпретируемых моделей не много: линейные (MVLР, LR, GAM, GLM), логические (DT, RI), метрические (kNN, PW, RBF), байесовские (NB, BN)
- *Объяснимость* решения на объекте — как правило, с помощью интерпретируемой *суррогатной модели*
- *Вектор Шепли* оценивает вклады «игроков» в общий результат по данным об успешности их коалиций; «игроки» это признаки, но идея применима и к объектам
- SHAP, SAGE — наиболее продвинутые методы объяснения

P.Linardatos, V.Papastefanopoulos, S.Kotsiantis. Explainable AI: A Review of Machine Learning Interpretability Methods. 2021

Zachary C. Lipton. The Mythos of Model Interpretability. 2018

Christoph Molnar. Interpretable Machine Learning: A Guide for Making Black Box Models Explainable. 2019

Ribana Roscher et al. Explainable Machine Learning for Scientific Insights and Discoveries. 2020