Pokročilé metódy analýzy dát 6

úvod do hlbokého učenia

Peter Bednár

Vysvetľovanie modelov

Prečo je interpretovateľnosť modelov dôležitá

- Odhad agregovaných štatistík na testovacích dátach (presnosť, návratnosť, atď.) nemusí stačiť na testovanie spoľahlivosti modelu
- Získanie nových znalostí o svete z modelu
- Zlepšenie sociálnej akceptovateľnosti modelov
- Užitočné aj pri učení a ladení modelov, detegovaní *biasu*

Kedy nie je interpretovateľnosť potrebná

- Pri nekritických aplikáciách
- Ak máme dobré teoretické porozumenie samotného problému
- Interpretovateľnosť umožňuje ľahšiu manipuláciu s predikciami modelu

Rozdelenie metód interpretovateľnosti

- Interpretovateľné modely
- Podľa závislosti na modeloch:
  - Metódy nezávislé na modeloch
  - Metódy určené pre určitý typ modelov (napr. neurónové siete, alebo rozhodovacie stromy)
- Metódy určujúce dôležitosť atribútov
- Metódy založené na inštanciách

Interpretovateľné modely (1)

- Lineárne modely
  - Lineárna/logistická regresia
  - Parametre modelu – váhy, ktoré pre každý atribút priamo vyjadrujú jeho dôležitosť pre predikciu

$$f(\boldsymbol{x}) = \beta_0 + \beta_1 x_1 + \cdots + \beta_m x_m$$

  - Pri učení je potrebné normovať hodnoty atribútov aby sa dali parametre priamo porovnávať
  - Dajú sa štatisticky interpretovať – ale predpoklady sa ťažko validujú pre reálne dáta
  - LASSO regularizácia pre ohraničenie parametrov

Interpretovateľné modely (2)

- Rozhodovacie stromy a pravidlá
  - Dajú sa interpretovať ako logické Ak-Potom pravidlá
  - Podmienky priamo ohraničujúce hodnoty atribútov
  - Ohraničenie/orezanie stromov/pravidiel pre zlepšenie interpretovateľnosti
  - Podľa kritéria pre výber podmienok/delenie (informačný zisk, GINI index, atď.) môžeme priamo ohodnotiť dôležitosť jednotlivých atribútov

Interpretovateľné modely (3)

- Naivný Bayesov klasifikátor
  - Podmienené pravdepodobnosti pre každý atribút umožňujú vypočítať príspevok voči predikcii (podobne ako pri lineárnych modeloch)

$$P(c|x) = \frac{1}{Z}P(c)P(x|c) = \frac{1}{Z}P(c)\prod_{i=1}^{m}P(x_i|c)$$

- Učenie založená na inštanciách – k-NN
  - Lokálne vysvetlenie pre jeden predikovaný príklad – k najpodobnejších príkladov
  - Musíme vedieť interpretovať jednotlivé príklady/inštancie

Metódy nezávislé na modeloch

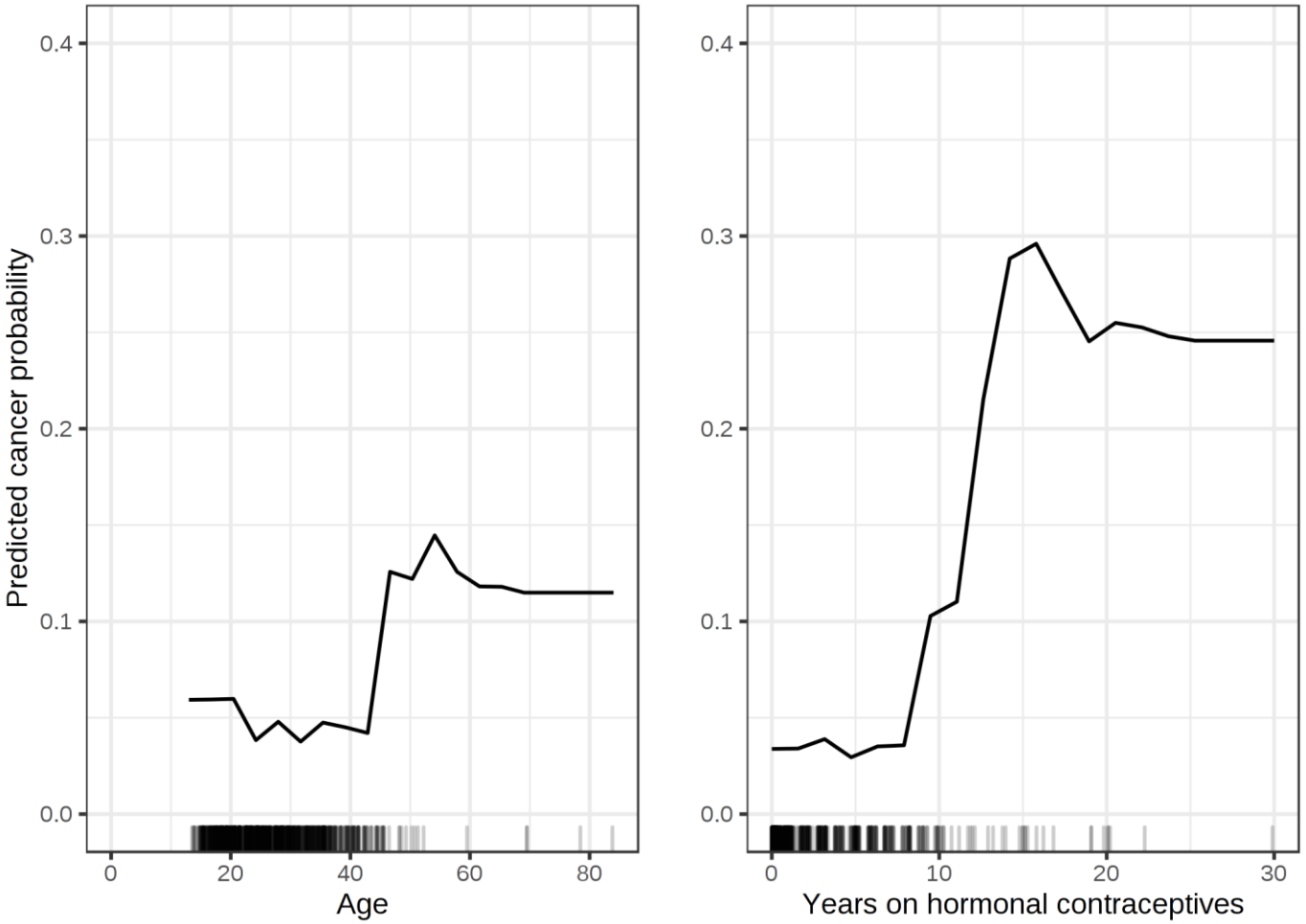
Partial Dependence Plot – PDP (1)

- Pre daný model zobrazuje závislosť medzi jedným/dvoma vstupnými atribútmi a cieľovým atribútom

1. Postupne meníme hodnotu sledovaného atribútu x_S
2. Pre každú hodnotu dosadíme do príkladu kombináciu hodnôt ostatných atribútov $x_C^{(i)}$ a spriemerníme predikciu modelu

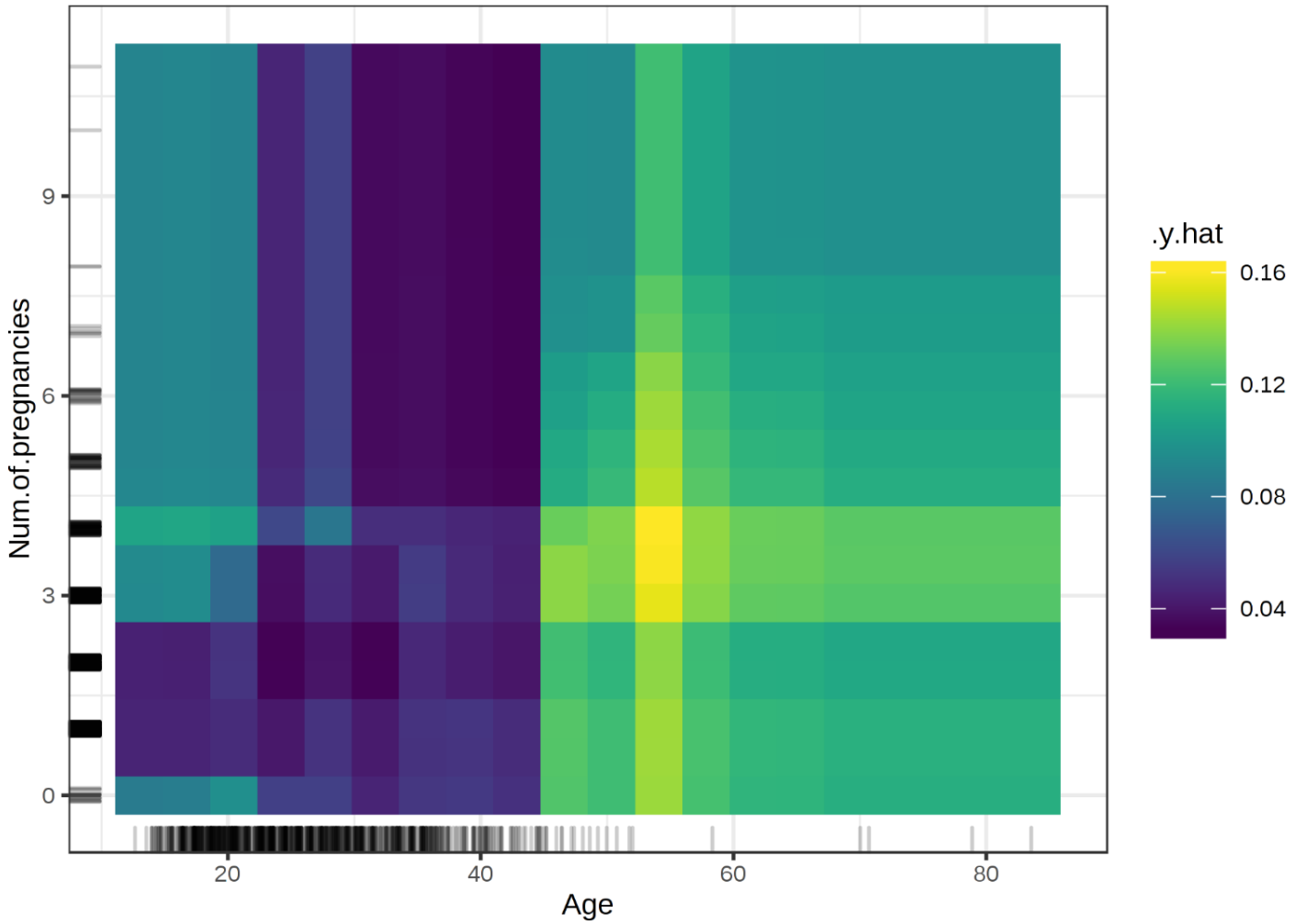
$$PD_f(x_s) = \frac{1}{n}\sum_{i=0}^{n} f\left(x_S, x_C^{(i)}\right)$$

Partial Dependence Plot – PDP (2)



Zdroj: https://christophm.github.io/interpretable-ml-book/pdp.html

Partial Dependence Plot – PDP (3)



Zdroj: https://christophm.github.io/interpretable-ml-book/pdp.html

Dôležitosť atribútov podľa permutácie hodnôt

- Pre daný atribút vytvoríme z trénovacích dát novú množinu príkladov permutáciou jeho hodnôt
  - Narušíme interakcie v modely medzi vstupnými atribútmi
- Atribút je dôležitejší, ak sa výrazne po permutácii zníži presnosť modelu
- Zoradíme atribúty podľa dôležitosti

Globálne zástupné modely

- Predikciu zložitého modelu aproximujeme interpretovateľným modelom

1. Zvolíme si dátovú množinu (môžu to byť trénovacie dáta použité na vytvorenie vysvetľovaného modelu, ale aj iná množina príkladov z tej istej domény)
2. Na zvolenej množine vypočítame predikciu vysvetľovaného modelu
3. Naučíme interpretovateľný zástupný model na zvolených príkladoch s predikovanými hodnotami cieľového atribútu

- Chyba zástupného modelu určuje, koľko variancie vysvetľovaného modelu sme pokryli

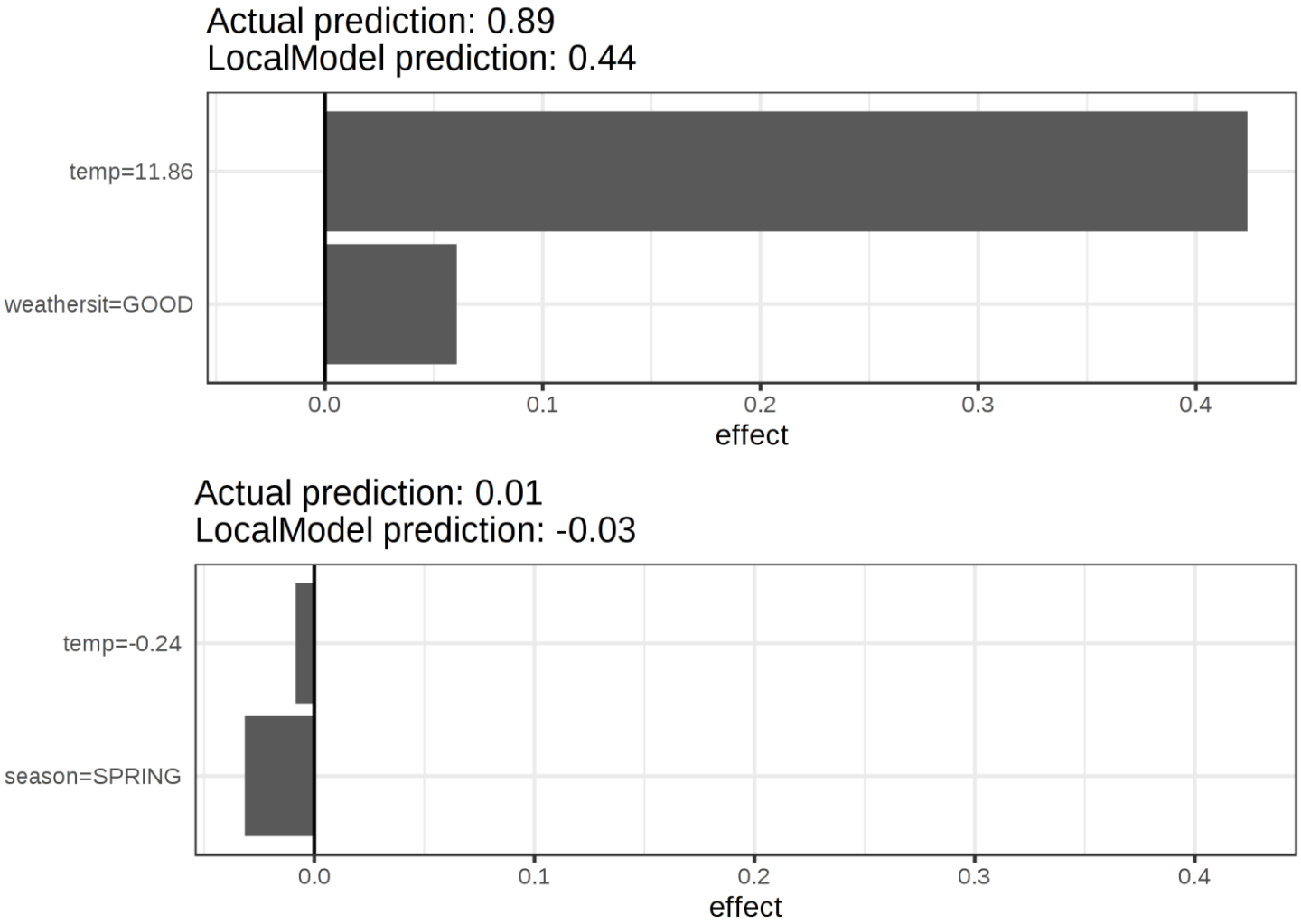
Lokálne zástupné modely - LIME

- Pre zložité problémy nedokážeme globálne pre všetky dáta aproximovať zložitý model jednoduchším modelom dostatočne presne

- LIME - *Local Interpretable Model-agnostic Explanations*

- Vysvetlenie pre jeden príklad
  1. Náhodnými perturbáciami si vytvoríme množinu dát v okolí vysvetľovaného príkladu, cieľový atribút vypočítame ako predikciu vysvetľovaného modelu
  2. Naučíme lokálny interpretovateľný zástupný model

LIME s LASSO lineárnym modelom



Zdroj: https://christophm.github.io/interpretable-ml-book/lime.html

Metódy založené na inštanciách

Príklady pre alternatívne scenáre (1)

- Pre príklad pre ktorý chceme nájsť vysvetlenie skonštruujeme alternatívny príklad, pre ktorý bude mať model zvolenú alternatívnu hodnotu predikcie
- Musí platiť:
  - Predikcia modelu pre alternatívny príklad je čo najbližšie k požadovanému alternatívnemu výstupu
  - Alternatívny príklad je čo najviac podobný vysvetľovanému príkladu

Príklady pre alternatívne scenáre (2)

- Napr. pre regresiu hľadáme x', ktoré minimalizuje:

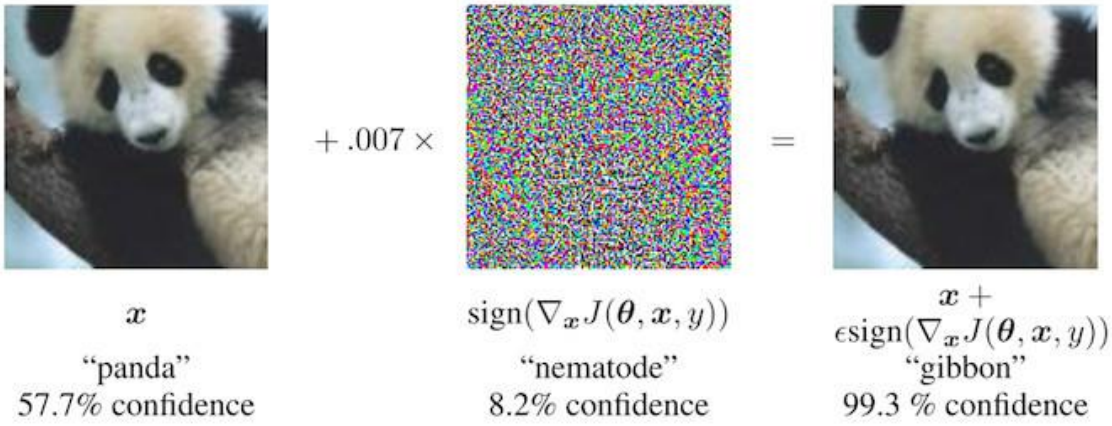
$$L(x, x', y') = \lambda(f(x') - y')^2 + d(x, x')$$

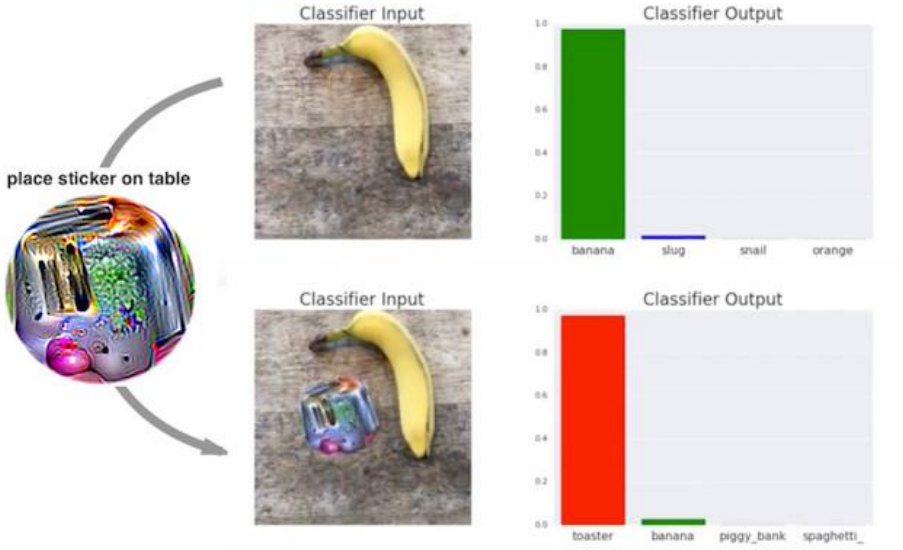
- Kde λ váži vplyv medzi dvoma kritériami a $d(x, x')$ meria podobnosť medzi dvoma príkladmi (napr. Manhattan vzdialenosť)

Kontradiktórne príklady (1)

- Cielene sa snažíme nájsť príklady, ktoré model pomýlia pri predikcii
- Napr. pri obrázkoch vychádzame zo zdrojového obrázka a:
  - Meníme hodnoty všetkých pixelov s malými človekom nerozoznateľnými zmenami, ktoré celkovo spôsobia chybnú klasifikáciu
  - Alebo zmeníme výraznejšie malý počet pixelov – v extrémnom prípade 1 bod
- Bezpečnostná zraniteľnosť metód

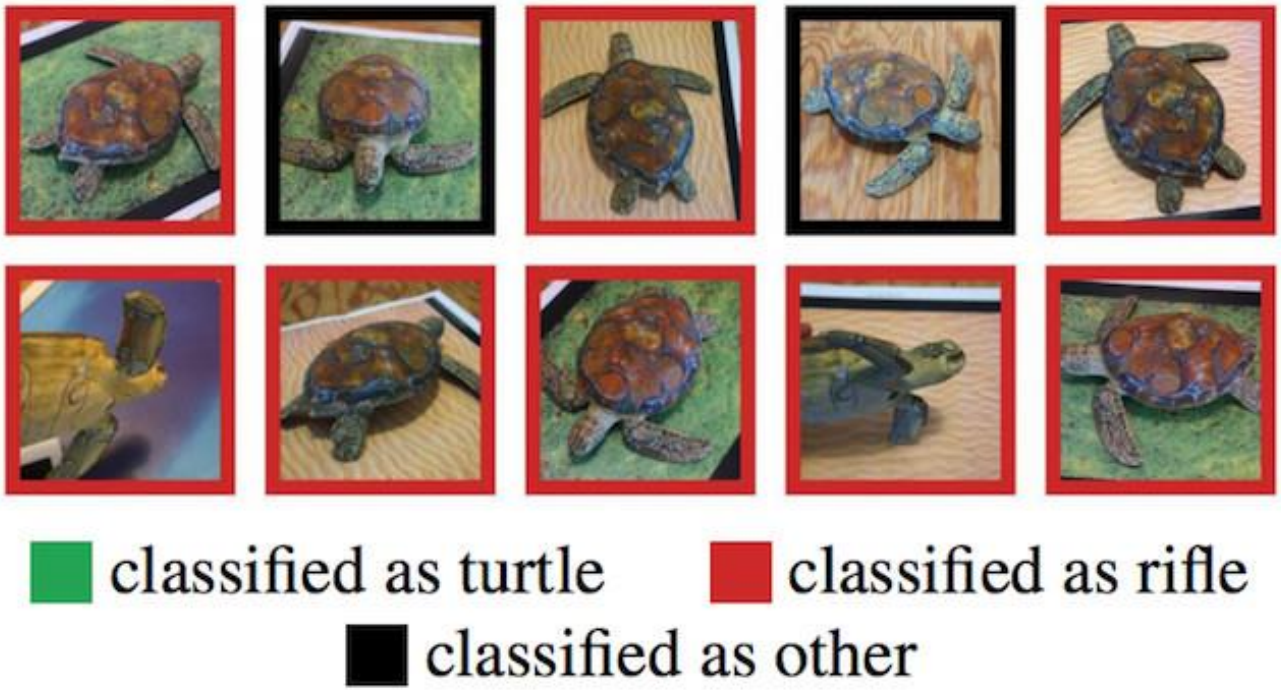
Kontradiktórne príklady (2)





Zdroj: https://christophm.github.io/interpretable-ml-book/adversarial.html

Kontradiktórne príklady (3)



Zdroj: https://christophm.github.io/interpretable-ml-book/adversarial.html

Prototypy a kritické príklady

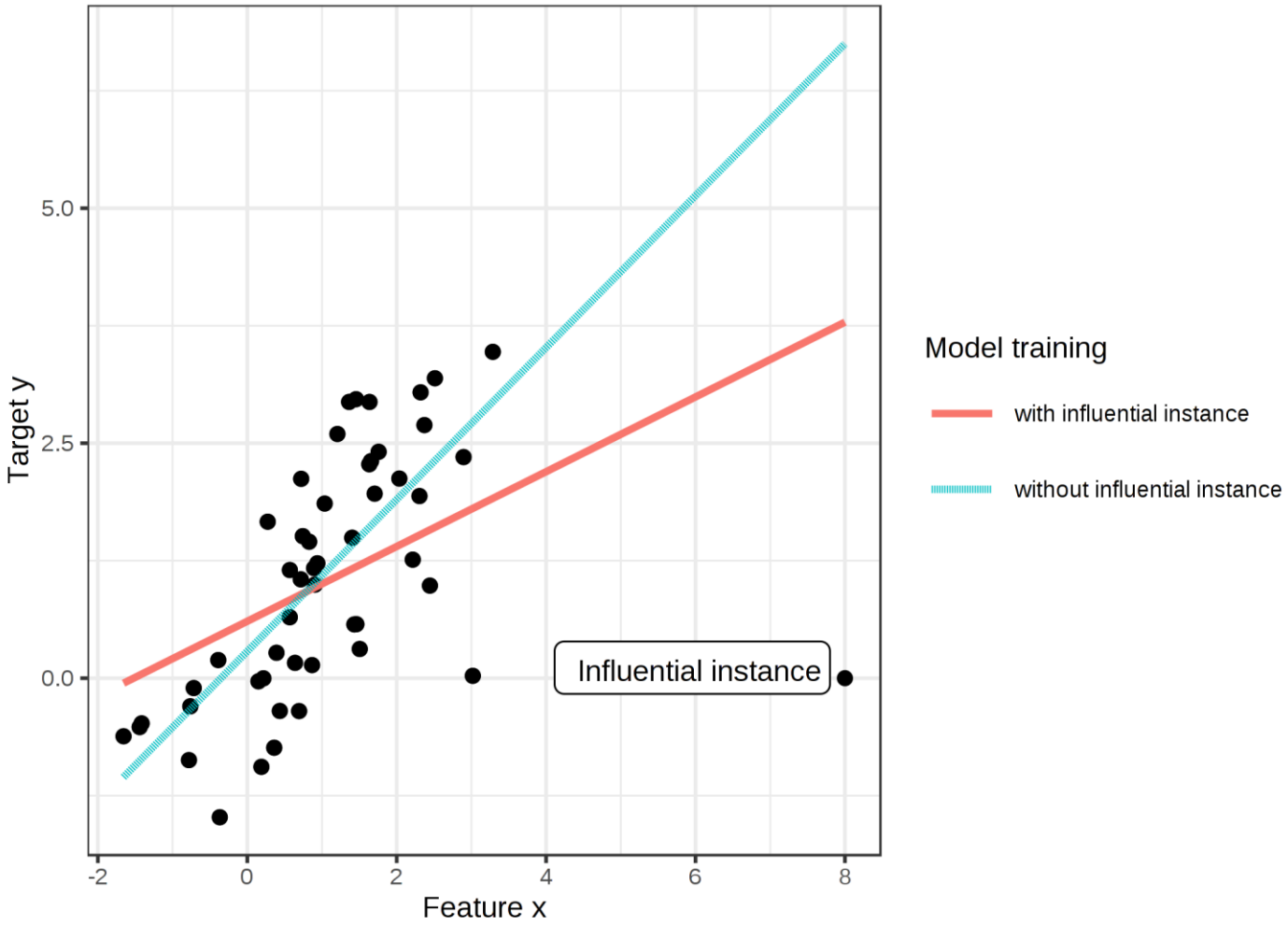
- Prototypy sú reprezentatívne príklady, ktoré pokrývajú všetky trénovacie dáta
- Kritické príklady sú príklady, ktoré sa nedajú dobre reprezentovať niektorým z prototypov

- Pre identifikovanie prototypov a kritických príkladov je možné použiť metódy zhlukovania (napr. *k-means*, alebo *k-medoids*)

Najvplyvnejšie príklady (1)

- Najvplyvnejšie trénovacie príklady spôsobia po ich odstránení z trénovacej množiny:
  1. Veľkú zmenu parametrov modelu, alebo
  2. Veľkú zmenu v predikcii modelu
     - Porovnanie dvoch predikcií pomocou absolútnej chybovej funkcie
- Príklady je potrebné interpretovať
  - Napr. diskriminačne interpretovateľnými modelmi

Najvplyvnejšie príklady (2)



Zdroj: https://christophm.github.io/interpretable-ml-book/influential.html

Najvplyvnejšie príklady (3)

- Detegovanie najvplyvnejších príkladov vyžaduje učenie veľkého počtu modelov
- Pre metódy založené na gradiente:
  - Namiesto odstránenia príkladu nepatrne zväčšíme jeho váhu pri učení a sledujeme ako sa zmenia hodnoty parametrov
  - Efektívny algoritmus, ktorý nevyžaduje viacnásobné učenie modelov – musíme mať pre daný príklad prístup ku gradientu chybovej funkcie a k jej druhej derivácii (Hessovej matici)

Vizualizácia neurónových sietí

Vizualizácia naučených príznakov (1)

- Pre zvolené neuróny vieme spätne vypočítať a vizualizovať vstupné hodnoty, ktoré vedú k maximálnej aktivácii daného neurónu
  - Optimalizačná úloha: váhy naučenej neurónovej siete sa nemenia, menia sa vstupné atribúty a hľadáme takú kombináciu vstupných hodnôt ktorá maximalizuje aktiváciu zvoleného neurónu
  - Môžeme vizualizovať vstupné dáta pre neurón na výstupnej, alebo ľubovoľnej skrytej vrstve

Vizualizácia naučených príznakov (2)

- Pre konvolučné siete môžeme naraz vizualizovať aktiváciu jednotlivých filtrov pre všetky kanály naraz
- Výsledok je vstupný obrázok, pre ktorý je daný filter najviac aktivovaný – zobrazí sa naučený vzor
  - Pre výstupný neurón je možné takýmto spôsobom rekonštruovať najcharakteristickejší príklad pre danú triedu – daný výstupný neurón – bezpečnostné riziko
- Gradientová metóda pre optimalizáciu – na začiatku je vstupný obrázok vygenerovaný náhodne

Vizualizácia naučených príznakov (3)

hrany textury vzory časti objekty

Zdroj: https://distill.pub/2017/feature-visualization/

Disekcia sietí (1)

- Máme konvolučnú sieť naučenú pre sémantické segmentovanie obrázkov, tzn. sieť ktorá dokáže o každom pixely obrázka rozhodnúť akému typu objektu patrí

- Hypotéza: Existuje v sieti priamo jednotka (konvolučný filter), ktorá reprezentuje jednotlivé typy sémantických konceptov – tzn. objektov, alebo pomenovaných vzorov?
  - Základná otázka je, či sa sieť naučí priamo reprezentovať koncepty jednotlivými jednotkami, alebo či je pre rozpoznávanie konceptov potrebné kombinovať výstup z viacerých jednotiek

Disekcia sietí (2)

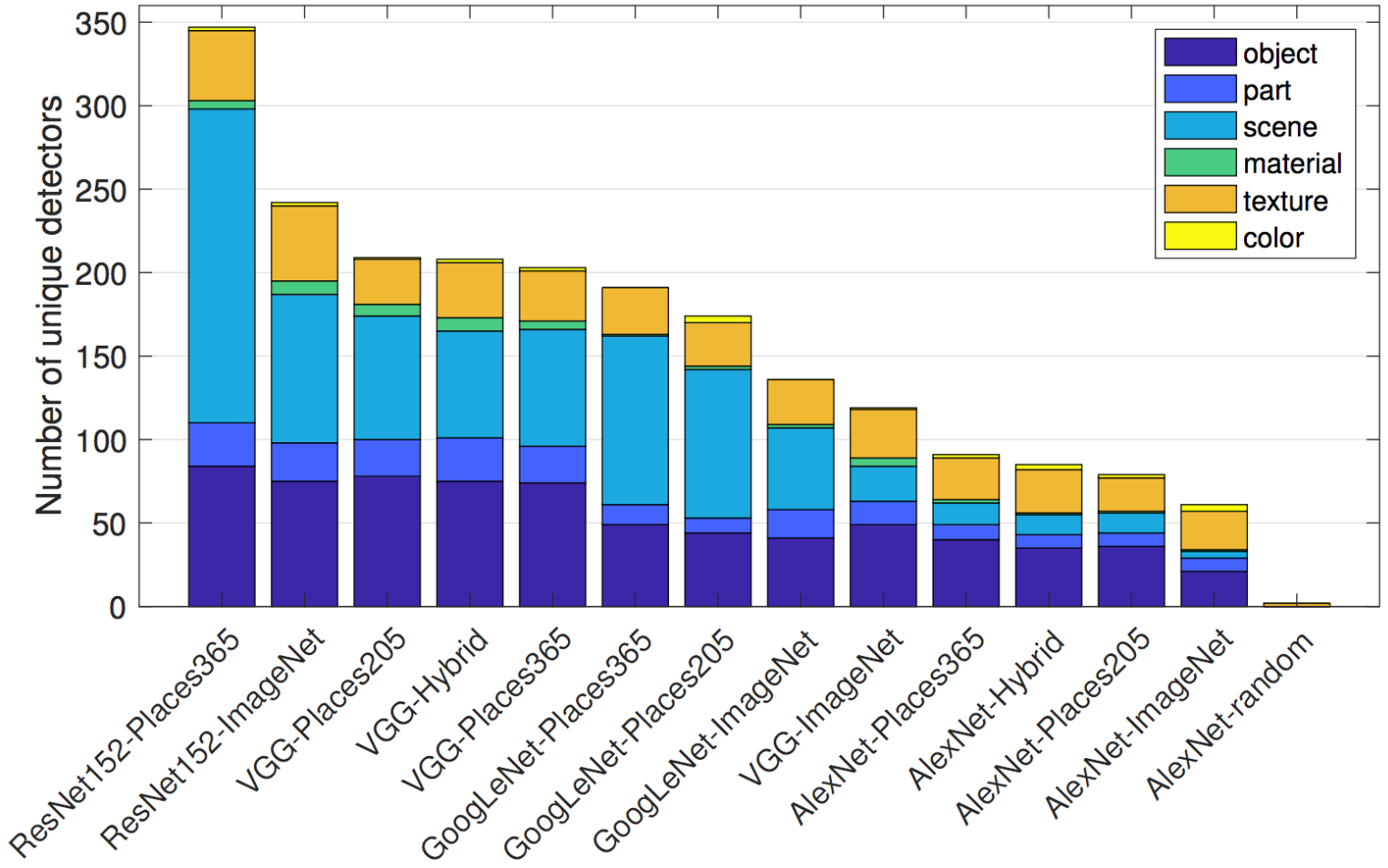
- Testovacie dáta: dátová množina Broden (*Broadly and Densely labeled data*), 60 000 obrázkov, každý pixel je zaradený do viacerých konceptov od farieb, vzorov až po objekty a komplexné scény

Disekcia sietí (3)

1. Každý testovací obrázok premietneme do vrstvy, ktorá obsahuje testovaný konvolučný filter – získame aktivačnú masku filtra pre každý bod na danej vrstve
2. Vypočítame distribúciu aktivácii pre všetky obrázky
3. Pre daný obrázok prahovaním získame binárnu aktivačnú masku na danej vrstve – prah 99.5 % percentil
4. Porovnáme binárnu aktivačnú masku s maskou pre daný koncept na danom obrázku a vypočítame ich prienik

Ak sa masky často výrazne zhodujú, daný filter dobre reprezentuje daný koncept

Disekcia sietí (4)



Zdroj: http://netdissect.csail.mit.edu