BI-ML1.21 přednáška 10 handout 27. listopadu 2024
Problémy, návrhy apod. hlaste v GitLabu.
Verze souboru: 27. listopadu 2024 10:25.

Co bude v dnešní přednášce

- Výběr příznaků obecně
- Filtrovací metody
- Obalové metody
- Vestavěné metody - Lasso

38 Výběr příznaků

Redukce počtu vysvětlujících proměnných

- V dnešní přednášce se budeme věnovat metodám, které mají za úkol snížit počet vysvětlujících proměnných.
- V zásadě jde o to, vybrat ze všech dostupných příznaků (které mohly být i extra napočítány) jistou podmnožinu, kterou použijeme pro trénování a predikci modelu.
- Obecně v této situaci mluvíme o *výběru příznaků* (angl. *feature selection* nebo také *subset selection*).
- Problematika výběru příznaků spadá do části předzpracování dat. Dneska se podíváme jenom na některé základní metody a podrobněji se touto problematikou budete zabývat v magisterském kurzu *NI-PDD*.
- Z jistého pohledu se jedná o podoblast redukce dimenzionality (angl. dimensionality reduction). Do té ale spadají především metody, které napočítávají (lineárně nebo nelineárně) kompletně nové příznaky. Proto se redukce dimenzionality často uvádí jako separátní oblast.

Účel výběru příznaků

Výběr příznaků může být výhodný z mnoha různých důvodů. Uveďme si některé z nich.

- Zahozením nerelevantních a redundantních příznaků můžeme významně zlepšit schopnost generalizace modelu.
  - Příznaky, které jsou nerelevantní a nesouvisí s vysvětlovanou proměnnou, škodí typicky i modelům, které by si s nimi teoreticky dokázali poradit - jako např. lineární regrese (může tam použít koeficient 0) nebo rozhodovací strom (nepoužije tento příznak do žádného dělícího kritéria).
  - Redundantní příznaky jsou takové, které nesou stejnou informaci. Kromě toho, že zbytečně zvyšují dimenzi mohou některým modelům vyloženě škodit, jako např. problém kolinearity u lineární regrese.
- Pomáhá s prokletím dimenzionality. Jak víme, tak pro některé modely je velká dimenze prostoru příznaků problematická (např. KNN).
- Zlepšuje vysvětlitelnost modelu. U modelů, který využívají menší počet příznaků bývá jednodušší porozumět tomu, na základě čeho se rozhodují.
- Snižuje výpočetní nároky pro trénování a použití výsledného modelu.

39 Metody výběru příznaků

Filtrační metody

Filtrační metody (angl. *filter methods*) mohou být supervizované i nesupervizované.

Typicky koukají, jak hodně informace může být v daném příznaku a případně jak ta informace souvisí s vysvětlovanou proměnnou.

Pokud nemáme nebo nechceme využít vysvětlovanou proměnnou, můžeme:

- Vyhodit příznaky, které mají příliš nízký rozptyl a jsou tedy téměř konstantní.

- Vyhodit příznaky, které mají příliš chybějících hodnot.
- Vyhodit některé z příznaků, které spolu hodně korelují a jsou tedy redundantní.

Pokud chceme využít vysvětlovanou proměnnou, můžeme:

- Vyhodit příznaky, které mají nízkou korelaci s vysvětlovanou proměnnou.
- U binárních příznaků rozdělit vysvětlovanou na dvě populace a udělat test hypotézy o rovnosti středních hodnot (dvouvýběrový t-test). Pokud vyjde velká p hodnota, můžeme tento příznak vyhodit.
- U diskrétních příznaků i diskrétní vysvětlované proměnné udělat test hypotézy o nezávislosti těchto dvou diskrétních veličin (např. chí kvadrát test nezávislosti). Opět, pokud vyjde velká p hodnota, je tento příznak kandidátem na vyhození.

Obalové metody (1/2)

- *Obalové metody* (angl. *wrapper methods*) využívají nějaký pomocný model pro ohodnocování podmnožin příznaků.
- Cílem je vybrat takovou podmnožinu, pro kterou je výkonnost modelu největší.
- Pro každou kandidátní podmnožinu se model natrénuje a změří jeho výkonnost na validační množině (nebo křížovou validací).
- Jako finální podmnožina příznaků je použita taková, pro kterou je model nejvýkonnější.
- Výhodou obalových metod je, že preferují podmnožiny příznaků, které jsou pro daný model výhodné.
- To je ale zároveň nevýhoda, protože může snadno dojít k přeučení a také protože takto vybrané podmnožiny nemusí být vhodné pro jiné modely (např. pro ten finální, který chceme použít).
- Nevýhodou obalových metod je jejich výpočetní náročnost. I při použití jednoduchého pomocného modelu (jako např. lineární regrese) může být výpočetní čas značný.

Obalové metody (2/2)

Z důvodů výpočetní náročnosti nejčastěji používané metody vybranou množinu příznaků konstruují iterativně nějakým „hladovým“ přístupem.

- *Dopředný výběr* (angl. *forward selection*) - začínáme s prázdnou množinou příznaků a postupně po jednom přidáváme vždy tak, aby ten jeden přidaný nejvíce zlepšil výkonnost modelu v dané iteraci.

  Skončíme když máme požadovaný počet příznaků, případně když už nedochází ke zlepšování výkonnosti.
- *Zpětný výběr* (angl. *backward selection*) - začneme se všemi příznaky a postupně po jednom odebíráme tak, aby ten odebraný příznak vedl na nejvýkonnější model v dané iteraci.

  Skončíme když máme požadovaný počet příznaků, případně když už nedochází ke zlepšování výkonnosti.
- Dopředný i zpětný výběr jsou v knihovně `sklearn` implementovány pomocí `sklearn.feature_selection.SequentialFea`

Obalové metody (3/3)

- *Rekurzivní odebírání příznaků* (angl. *recursive feature elimination*) - probíhá podobně jako zpětný výběr, ale model se využívá trochu jiným způsobem.

  Konkrétně se k vybírání příznaků používá vnitřní ohodnocení důležitosti jednotlivých příznaků modelem (který toho tedy musí být schopen).

  U lineární (logistické) regrese to je například absolutní hodnota koeficientu u daného příznaku. U rozhodovacího stromu to může být informační zisk daného příznaku.

  Nejprve model natrénujeme pro všechny příznaky. Potom najdeme příznak, který má nejmenší důležitost a ten vyhodíme. Pak model znovu natrénujeme na redukované množině a postup opakujeme.
- Rekurzivní odebírání příznaků je v knihovně `sklearn` implementováno pomocí `sklearn.feature_selection.RF`

Vestavěné metody

- *Vestavěné metody* (angl. *embedded methods*) využívají model, který se trénuje pouze jednou na celých datech a při tom implicitně provede výběr příznaků.
- Tento implicitní výběr se projeví tak, že se model naučí některé příznaky vůbec nevyužívat.
- Např. u lineární regrese jsou příslušné koeficienty odhadnuty jako 0.
- Modelem, který je nejpoužívanější, je L_1 regularizovaná lineární regrese (Lasso). V případě klasifikace to je pak L_1 regularizovaná logistická regrese.
- Jako vybraná podmnožina příznaků se vezmou příznaky, u který jsou příslušné koeficienty nenulové.

40 Lasso

Model lineární regrese

Nejprve si připomeňme model lineární regrese.

- Model pro vysvětlovanou proměnnou Y v bodě $\boldsymbol{x}$ je

$$Y = \boldsymbol{w}^T\boldsymbol{x} + \varepsilon = \boldsymbol{x}^T\boldsymbol{w} + \varepsilon = w_0 + w_1x_1 + \dots w_px_p + \varepsilon.$$

- Model pro trénovací množinu tvořenou N páry $(Y_i, \boldsymbol{x}_i)$ je $Y_i = \boldsymbol{x}_i^T\boldsymbol{w} + \varepsilon_i$.
- Toto zapisujeme maticově jako $\boldsymbol{Y} = \mathbf{X}\boldsymbol{w} + \boldsymbol{\varepsilon}$, kde

$$\mathbf{X} = \begin{pmatrix} \boldsymbol{x}_1^T \\ \vdots \\ \boldsymbol{x}_N^T \end{pmatrix} = \begin{pmatrix} 1 & x_{1;1} & \cdots & x_{1;p} \\ \vdots & \vdots & \ddots & \vdots \\ 1 & x_{N;1} & \cdots & x_{N;p} \end{pmatrix}, \quad \boldsymbol{Y} = \begin{pmatrix} Y_1 \\ \vdots \\ Y_N \end{pmatrix}, \quad \boldsymbol{\varepsilon} = \begin{pmatrix} \varepsilon_1 \\ \vdots \\ \varepsilon_N \end{pmatrix}.$$

- Při trénování metodou nejmenších čtverců minimalizujeme residuální součet čtverců

$$\mathrm{RSS}(\boldsymbol{w}) = \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{w}\|^2.$$

- Za předpokladu, že je matice $\mathbf{X}^T\mathbf{X}$ regulární, existuje jediné řešení minimalizující $\mathrm{RSS}(\boldsymbol{w})$,

$$\hat{\boldsymbol{w}}^{\mathrm{OLS}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y}.$$

Lasso - úvod

Lasso (zkr. angl. *Least Absolute Shrinkage and Selection Operator*) [Tibshirani (1996)] nebo taky L_1 regularizace zavádí penalizační člen úměrný součtu absolutních hodnot složek vektoru koeficientů $\boldsymbol{w}$ s vynecháním interceptu.

Minimalizujeme tedy *regularizovaný reziduální součet čtverců*

$$\mathrm{RSS}_\lambda^{\mathrm{Lasso}}(\boldsymbol{w}) = \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{w}\|^2 + \lambda\sum_{i=1}^{p}|w_i|,$$

který závisí na parametru $\lambda \geq 0$.

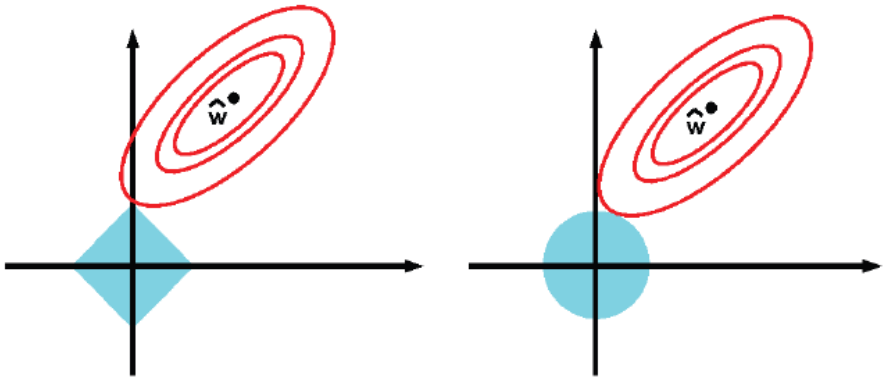
- Lasso jakožto jiná forma regularizace sdílí některé obecné vlastnosti s modelem hřebenové regrese.
- Pro $\lambda = 0$ dostáváme $\mathrm{RSS}_0^{\mathrm{Lasso}}(\boldsymbol{w}) = \mathrm{RSS}(\boldsymbol{w})$ a máme tedy obyčejnou metodu nejmenších čtverců.
- Pro $\lambda > 0$ je vidět, že v minimu se bude cílit na takové vektory $\boldsymbol{w}$, které mají co nejmenší složky.
- Hodnotu w_0 interceptu nijak nepenalizujeme. Jedná se pouze o vertikální posun, který zajišťuje předpoklad $\mathrm{E}\,\varepsilon = 0$ modelu a je tedy vhodné ho neomezovat.

Lasso - pokračování

- Na rozdíl od modelu hřebenové regrese není regularizační člen $\sum_{i=1}^{p} |w_i|$ diferencovatelný v bodech kde $w_j = 0$.
- V tomto případě není možné nalézt explicitní řešení a existují pouze iterativní metody, které najdou řešení
$$\hat{\boldsymbol{w}}_\lambda^{\text{Lasso}} = \arg\min_{\boldsymbol{w}} \text{RSS}_\lambda^{\text{Lasso}}(\boldsymbol{w}).$$
- Výhoda modelu Lasso je, že řešení je tzv. *řídké* (angl. *sparse solution*).
- To znamená, že odhady $\hat{\boldsymbol{w}}_{\lambda;j}^{\text{Lasso}}$ některých složek $\boldsymbol{w}$ jsou rovny 0.
  Samozřejmě tím častěji, čím je λ větší.
- Tím se lasso zásadně liší od hřebenové regrese, kde tento efekt v podstatě nikdy nenastane.
- Formálně to dokázat není jednoduché (používá se teorie subgradientů), ale ukážeme si alespoň přesvědčivou ilustraci.

Lasso - ilustrace fungování

Podívejme se na „vrstevnice“ funkcí, které se snažíme minimalizovat u lasso (vlevo) a hřebenové regrese (vpravo).



Červeně jsou znázorněny vrstevnice odpovídající neregularizované části $\|\boldsymbol{Y} - \mathbf{X}\boldsymbol{w}\|^2$, což je v zásadě parabolická jáma.

Modře je pak znázorněna oblast, která odpovídá regularizačnímu členu.

Vidíme, že u lasso existuje velká šance, že se vrstevnice dotýká některého z rohů, což znamená, že je jedna ze složek $\boldsymbol{w}$ rovna 0.

U hřebenové regrese se to samozřejmě také může stát, ale evidentně to je velmi nepravděpodobné.

Lasso a hřebenová regrese - vázané optimalizace

Fakt z předchozí ilustrace může být ještě podpořen následující ekvivaletní formulací.

Lze ukázat, že úloha minimalizace

$$\text{RSS}_\lambda^{\text{Lasso}}(\boldsymbol{w}) = \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{w}\|^2 + \lambda \sum_{i=1}^{p} |w_i|,$$

je ekvivalentní úloze vázané minimalizace

$$\text{RSS}(\boldsymbol{w}) = \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{w}\|^2,$$

za podmínky

$$\sum_{i=1}^{p} |w_i| \leq B, \quad \text{pro nějaké } B > 0.$$

Analogicky pro hřebenovou regresi platí, že minimalizace

$$\text{RSS}_\lambda^{\text{Ridge}}(\boldsymbol{w}) = \|\boldsymbol{Y} - \mathbf{X}\boldsymbol{w}\|^2 + \lambda \sum_{i=1}^{p} w_i^2,$$

je ekvivalentní úloze vázané minimalizace RSS($\boldsymbol{w}$) za podmínky

$$\sum_{i=1}^{p} w_i^2 \leq B, \quad \text{pro nějaké } B > 0.$$

Porovnání nejmenších čtverců, hřebenové regrese a lasso

Pro porovnání metod regularizace se zaměřme na situaci, kdy jsou příznaky ortonormální. Tj. když $\mathbf{X}^T\mathbf{X} = \mathbf{I}$.

- Metoda nejmenších čtverců v takovém případě vychází

$$\hat{\boldsymbol{w}}^{\text{OLS}} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T\boldsymbol{Y} = \mathbf{X}^T\boldsymbol{Y}.$$

- Hřebenová regrese vychází

$$\hat{\boldsymbol{w}}_\lambda^{\text{Ridge}} = (\mathbf{X}^T\mathbf{X} + \lambda\mathbf{I}')^{-1}\mathbf{X}^T\boldsymbol{Y} = (\mathbf{I} + \lambda\mathbf{I}')^{-1}\mathbf{X}^T\boldsymbol{Y}.$$

  Protože

$$(\mathbf{I} + \lambda\mathbf{I}')^{-1} = \begin{pmatrix} 1 & 0 & \cdots & 0 \\ 0 & \frac{1}{1+\lambda} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & \frac{1}{1+\lambda} \end{pmatrix},$$

  platí pro jednotlivé složky

$$\hat{\boldsymbol{w}}_{\lambda;0}^{\text{Ridge}} = \hat{\boldsymbol{w}}_0^{\text{OLS}} \quad \text{a} \quad \hat{\boldsymbol{w}}_{\lambda;j}^{\text{Ridge}} = \frac{1}{1+\lambda}\hat{\boldsymbol{w}}_j^{\text{OLS}}.$$

- Pro Lasso lze ukázat

$$\hat{\boldsymbol{w}}_{\lambda;0}^{\text{Lasso}} = \hat{\boldsymbol{w}}_0^{\text{OLS}} \quad \text{a} \quad \hat{\boldsymbol{w}}_{\lambda;j}^{\text{Lasso}} = \operatorname{sgn}\left(\hat{\boldsymbol{w}}_j^{\text{OLS}}\right) \max\left(0, |\hat{\boldsymbol{w}}_j^{\text{OLS}}| - \tfrac{\lambda}{2}\right).$$

  Tomuto se říká soft thresholding.

Závěrečné poznámky

- Lasso je v knihovně `sklearn` implementováno pomocí `sklearn.linear_model.Lasso`.
- Při využití pro výběr příznaků se dá použít implementace `sklearn.feature_selection.SelectFromModel`.
- U logistické regrese bychom regularizační člen přidali k mínus logaritmu věrohodnostní funkce (k binární relativní entropii) a pak bychom prováděli minimalizaci.
- Jednou z nevýhod lasso proti hřebenové regresi je, že v případě kolineárních příznaků má tendenci vybrat pouze některé z nich. To se v některých případech při predikci na nových datech ukazuje jako jistá nevýhoda oproti hřebenové regresi, která typicky využije všechny příznaky.
- Existuje tedy také model, který obsahuje oba způsoby regularizace.
- Tomuto modelu se říká *elastic net* a kombinuje výhody obou přístupů.

ChangeLog

Verze	Datum	Autor	Log
1.0	27.11.2024	DV	Výchozí verze pro rok 2024/2025.
1.0	7.12.2023	DV	Výchozí verze pro rok 2023/2024.
1.0	1.12.2022	DV	Výchozí verze pro rok 2022/2023.