УНИВЕРЗИТЕТ У БЕОГРАДУ
МАТЕМАТИЧКИ ФАКУЛТЕТ

Урош Миленковић

ИНТЕРПРЕТАБИЛНОСТ ПРЕНОСА НАУЧЕНОГ ЗНАЊА У ЛИНЕАРНОЈ РЕГРЕСИЈИ

мастер рад

Београд, 2026.

Ментор:

др Бојана Милошевић, ванредни професор
Универзитет у Београду, Математички факултет

Чланови комисије:

др Марија Цупарић, доцент
Универзитет у Београду, Математички факултет

др Марко Обрадовић, ванредни професор
Универзитет у Београду, Математички факултет

Датум одбране: _____

Тати и деди

Наслов мастер рада: Интерпретабилност преноса наученог знања у линеарној регресији

Резиме: У овом раду проучавамо технику преноса наученог знања у контексту линеарне регресије, са посебним нагласком на интерпретабилност добијених коефицијената. Након прегледа теоријске основе линеарне регресије и мотивације за пренос знања, разматрамо различите методе преноса који се могу свести на матричну комбинацију изворног и циљног модела. Изводимо израз за добитак остварен преносом и анализирамо његово понашање у сопственој бази Грамове матрице. Представљамо статистички тест за процену корисности преноса заснован на Фишеровој расподели, као и Бајесовску интерпретацију добијене оцене. Затим уводимо фамилију губитака усмерених ка извору и изводимо трајекторије градијентног спуста за њих, које као специјалне случајеве обухватају чисто фино подешавање и дестилацију знања. Развијена методологија илустрована је на симулираним подацима, као и на реалним подацима о хемијском саставу и квалитету вина.

Кључне речи: пренос наученог знања, линеарна регресија, интерпретабилност, статистички тест, фино подешавање, дестилација знања, градијентни спуст, регуларизација, гребена регресија

Садржај

Глава 1

Увод

Често у машинском учењу подразумевамо да имамо довољно података за тренирање модела, као и да су подаци коришћени за тренирање из исте расподеле као и подаци над којима ће се вршити предвиђање.

Наравно, у пракси, ове две претпоставке су ретко тачне (подаци еволуирају кроз време, зависе од локације, начина мерења, и мноштва других ствари), па су нам неопходне методе помоћу којих можемо да их надоместимо. Узмимо као пример ресторан који жели да предвиди шта гости желе на јеловнику на основу података о омиљеним врстама хране испитаника. Засигурно се ови подаци разликују у Србији данас у односу на податке током последњих 20 година, као и од истих података од данас из Немачке. Међутим, јасно је и да подаци од раније и из других места нису небитни, то јест, очекујемо да ће преклапања у жељама бити.

Управо то преклапање је предмет бављења *преноса наученог знања*[1]. Пре свега, наиван начин да искористимо претходно научено је помоћу стручњака који ће моћи да раздвоји битно и небитно из ранијих података, а затим оно битно употреби на новим подацима. Други наиван начин је комбиновање података, придодајући различите тежине старим и новим. У примеру који смо навели, очекујемо да ће нових података бити далеко мање него старих (упоређујемо или нове податке са подацима током дугог временског периода, или податке на нивоу Србије са подацима на нивоу Немачке). Уколико имамо модел на великом скупу података, тешко да ће нам било какво наивно поновно тренирање где ови подаци чине огромну већину донети боља предвиђања у односу на модел који користи искључиво нове податке, јер ће свако тренирање трајати

[1]енгл. transfer learning

дуго.

Дакле, уколико имамо стари модел, потребно нам је да њега надоградимо, уместо да правимо нови испочетка, услед уштеде ресурса и времена, али и вероватно бољих резултата. Управо овај принцип је основа рада. Иако ћемо се у раду задржати на линеарној регресији, вреди напоменути да су примене шире од најосновнијих модела. Посебно, један од модернијих концепата је фино подешавање[2]. Оно управо подразумева коришћење огромних модела (попут великих језичких модела), и њихово специјализовање за конкретне задатке. Већ имамо трениран модел, практично на свим могућим (легално и илегално набављеним) подацима са интернета, а рецимо да ми желимо модел који генерише реченице које се користе на разгласу на железничким станицама. Претпоставимо и да имамо неки скуп реченица који одговара нашим потребама. Обзиром да је ово дефинитивно мали скуп, тренирање новог модела би готово сигурно довело до преприлагођавања[3] подацима. Међутим, користећи постојећи, већи модел, можемо да додамо суптилне промене у ове реченице, тако да задрже значење, али да искористимо пуну моћ језика, уместо његовог малог подскупа.

У контексту великих језичких модела[4], имамо још једну познатију примену где се заправо користи пренос наученог знања. Основни корак при тренирању је репрезентација речи у неком огромном векторском простору. Приликом тренирања, ове репрезентације се мењају, и заправо кроз њихове промене и међусобне односе добијамо модел који предвиђа наредну реч. Ове репрезентације је могуће користити у другим контекстима (на пример, желимо да упростимо језик), што представља својеврсни пренос наученог знања.

Поставка проблема који ћемо решавати је следећа: постоји и нама је доступан модел линеарне регресије (то јест, његови коефицијенти) на великом скупу података, као и модел на малом, али релативно сличном скупу података. Циљ је одредити модел који ће представљати комбинацију постојећих модела, а који ће давати боље резултате од њих на мањем скупу. Уз то, желимо да одговоримо и на још два питања: када има смисла преносити знање, као и да ли (у ситуацијама када то можемо да дискутујемо) умемо да интерпретирамо коефицијенте добијене при преносу знања.

Рад је организован на следећи начин. У глави 2 излажемо математичку основу: подсећамо се модела линеарне регресије, разматрамо ситуацију у којој циљ-

[2]енгл. fine-tuning
[3]енгл. overfitting
[4]енгл. large language models, LLMs

ни скуп података није довољан за поуздано оцењивање, и мотивишемо потребу за преносом знања преко линеарне комбинације изворне и циљне оцене. Ту такође уводимо и општу дефиницију преноса знања. У глави 3 концентришемо се на случај линеарне регресије и изводимо различите врсте оцена. Представљамо статистички тест за одлучивање да ли је пренос користан на конкретном пару скупова података. У глави 4 дискутујемо резултате на конкретним примерима: најпре на вештачким, а затим и на реалним подацима о хемијском саставу и квалитету вина [2].

Глава 2

Математичка основа

У овој глави подсећамо се модела линеарне регресије, уводимо нотацију и поставку рада, а затим уводимо сам појам преноса наученог знања, за почетак као општу дефиницију, а потом и кроз мотивацију најједноставнијом могућом идејом: конвексном комбинацијом изворне и циљне оцене, где ћемо демонстрирати компромис између дисперзије и пристрасности.

2.1 Осврт на линеарну регресију

Подсетимо се модела линеарне регресије. Основна претпоставка је да наши подаци $\{X, \mathbf{Y}\}$, где је $\mathbf{Y} = (y_1, y_2, \ldots, y_N)$, $X \in \mathbb{R}^{N \times D}$, N величина скупа података и D број предиктора, задовољавају:

$$\mathbf{Y} = X\beta + \varepsilon. \tag{2.1}$$

Овде вектор β представља непознате коефицијенте линеарног модела, а вектор ε грешке модела (за које важе услови Гаус-Маркова: $\mathbb{E}(\varepsilon) = 0$, $\mathbb{D}(\varepsilon) = \text{const}$, као и да су ови шумови некорелисани за све опсервације). Наравно, циљ нам је да пронађемо онакво $\hat{\beta}$ које, у неком контексту, минимизује грешку приближног модела $\hat{\varepsilon} = \mathbf{Y} - X\hat{\beta}$, односно оно $\hat{\beta}$ које минимизује израз:

$$L(\beta) = \|\mathbf{Y} - X\beta\|_2^2 = (\mathbf{Y} - X\beta)^\top(\mathbf{Y} - X\beta). \tag{2.2}$$

Рачунамо градијент ове функције:

$$\begin{aligned}
\nabla L &= \nabla(\mathbf{Y}^\top\mathbf{Y} - \mathbf{Y}^\top X\beta - \beta^\top X^\top\mathbf{Y} + \beta^\top X^\top X\beta) \\
&= \nabla(-2\beta^\top X^\top\mathbf{Y} + \beta^\top X^\top X\beta) \\
&= -2X^\top\mathbf{Y} + 2X^\top X\beta.
\end{aligned}$$

Претпостављамо још да је $X^\top X$ инвертибилна. Тада, да бисмо одредили екстремум функције, постављамо градијент на 0, чиме лако изводимо да је:

$$\hat{\beta} = (X^\top X)^{-1} X^\top \mathbf{Y}. \tag{2.3}$$

Да је ово минимум знамо на основу Хесијана[1] $\nabla^2 L = 2X^\top X$, што је позитивно дефинитна матрица (под претпоставком да X има пун ранг по колонама), па је L конвексна, и самим тим пронађено $\hat{\beta}$ мора бити минимум.

Знамо да је β вектор величине D, чије координате представљају допринос моделу појединачних предиктора. То значи да, при упоређивању важности различитих предиктора, можемо да посматрамо однос апсолутних вредности одговарајућих координата, грубо речено (под претпоставком да су предиктори стандардизовани). Иако нећемо превише формално дискутовати о овоме, видећемо да се предиктори "различите важности" сасвим природно различито понашају при (још увек недефинисаном) преносу знања.

Одредимо и очекивање и дисперзију оцене $\hat{\beta}$. Претпоставимо да за грешку модела важи $\varepsilon \sim \mathcal{N}(0, \sigma^2 I_N)$. Заменом $\mathbf{Y} = X\beta + \varepsilon$ у (2.3) добијамо алгебарски израз за грешку оцене:

$$\hat{\beta} - \beta = (X^\top X)^{-1} X^\top (X\beta + \varepsilon) - \beta = (X^\top X)^{-1} X^\top \varepsilon,$$

те је, будући да је $\mathbb{E}[\varepsilon] = 0$, оцена непристрасна ($\mathbb{E}[\hat{\beta}] = \beta$). Коваријациона матрица се тада рачуна директно:

$$\begin{aligned}
\operatorname{Cov}(\hat{\beta}) &= \mathbb{E}\Big[(\hat{\beta} - \beta)(\hat{\beta} - \beta)^\top\Big] = \mathbb{E}\big[(X^\top X)^{-1} X^\top \varepsilon\varepsilon^\top X (X^\top X)^{-1}\big] \\
&= (X^\top X)^{-1} X^\top \,\mathbb{E}[\varepsilon\varepsilon^\top]\, X (X^\top X)^{-1} = \sigma^2 (X^\top X)^{-1} X^\top X (X^\top X)^{-1} \\
&= \sigma^2 (X^\top X)^{-1},
\end{aligned}$$

где смо искористили $\mathbb{E}[\varepsilon\varepsilon^\top] = \sigma^2 I_N$. Одатле, средња квадратна грешка оцене коефицијената једнака је трагу коваријационе матрице:

$$\mathbb{E}\Big[\|\hat{\beta} - \beta\|_2^2\Big] = \operatorname{tr}\Big(\operatorname{Cov}(\hat{\beta})\Big) = \sigma^2 \operatorname{tr}\big((X^\top X)^{-1}\big)\,. \tag{2.4}$$

Наведимо и облик коваријационе матрице линеарне трансформације оцене, који ће нам бити користан у наставку. Ако је $A \in \mathbb{R}^{D\times D}$ произвољна (неслучајна) матрица, тада је $\operatorname{Cov}(A\hat{\beta}) = A \operatorname{Cov}(\hat{\beta}) A^\top = \sigma^2 A (X^\top X)^{-1} A^\top$, па је средња квадратна грешка при линеарној трансформацији оцене

$$\mathbb{E}\Big[\|A\hat{\beta} - A\beta\|_2^2\Big] = \sigma^2 \operatorname{tr}\big(A(X^\top X)^{-1} A^\top\big)\,. \tag{2.5}$$

[1] Ludwig Otto Hesse (1811–1874)

Будући да ће нам у главама које следе бити потребна двојака поставка, изворни и циљни домен, уведимо је овде, како бисмо сав каснији материјал могли изражавати заједничком нотацијом. Претпостављамо да су нам дата два модела:

$$\mathbf{Y}_S = f_S(X_S, \varepsilon_S), \qquad \mathbf{Y}_T = f_T(X_T, \varepsilon_T), \tag{2.6}$$

где индекси S и T означавају редом изворни[2] и циљни[3] домен. Матрице $X_S \in \mathbb{R}^{N_S \times D}$ и $X_T \in \mathbb{R}^{N_T \times D}$ имају исти број колона D, али, по правилу, $N_S \gg N_T$ у изворном домену располажемо великим бројем примера, док је циљни узорак мали. Шумови ε_S и ε_T су независни, а функције f_S и f_T различите, а њихова разлика говори колико се два домена суштински разликују.

2.2 Пренос наученог знања

Сада можемо да уведемо и сам појам преноса наученог знања. Као и код огромног броја других појмова машинског учења, није захвално покушати да нађемо неку превише формалну дефиницију, јер желимо нешто што обухвата огроман број различитих приступа и метода, а који сви имају за циљ да пренесу научено из једног на други модел, ма шта тај модел био, иако се у раду задржавамо на линеарној регресији.

Дефиниција 1. Дати су изворни скуп података $\{X_S, \mathbf{Y}_S\}$ и циљни скуп података $\{X_T, \mathbf{Y}_T\}$. Претпоставимо да имамо модел $\mathcal{M}_S$, трениран искључиво на изворном скупу података, као и модел $\mathcal{M}_T$ трениран искључиво на циљном скупу података. *Пренос наученог знања* је било који метод који, користећи моделе $\mathcal{M}_S$ и $\mathcal{M}_T$, дефинише нови модел $\mathcal{M}$, који, за унапред изабрану функцију ризика $\mathcal{R}$, којом оцењујемо квалитет модела, даје боље резултате у односу на модел $\mathcal{M}_T$, то јест, важи $\mathcal{R}(\mathcal{M}) < \mathcal{R}(\mathcal{M}_T)$.

Функција $\mathcal{R}$ из претходне дефиниције може, на пример, представљати очекивану квадратну грешку предвиђања на тест скупу повезаном са скупом података $\{X_T, \mathbf{Y}_T\}$. Разлог овакве општости дефиниције је тај што не захтевамо ни да $\mathcal{M}_S$ и $\mathcal{M}_T$ буду модели истог типа, нити да функција евалуације има неки посебан облик, па чак ни да имамо исти број предиктора у скупу података. Све-

[2] енгл. source
[3] енгл. target

обухватан преглед приступа преносу наученог знања у различитим доменима даје [8].

Оно на чему ћемо се ипак задржати у раду је следеће:

1. Подаци из изворног и циљног скупа имају једнак број димензија.
2. Изворни скуп података је далеко већи од циљног скупа.
3. Сви модели $\mathcal{M}_T$, $\mathcal{M}_S$ и $\mathcal{M}$ су истог типа (у раду су то модели линеарне регресије).

Уз помоћ ових претпоставки ћемо доћи до великог броја различитих приступа преносу знања, који, иако ће бити дискутовани из угла линеарне регресије, се могу приметити и на многе друге врсте модела.

Пример 1. Нека нам је доступан модел о предвиђању температуре на основу метеоролошких података из периода $1950 - 1980.$ године, $\mathcal{M}_S$, као и модел из 2026. године, $\mathcal{M}_T$. Рецимо да примећујемо да модел трениран само на доступним подацима из ове године не показује најбоља предвиђања, а да модел од раније даје добра предвиђања данима када концентрација штетних материја у ваздуху није велика. Можемо да закључимо да старији модел самим тим не узима у обзир на адекватан начин овај податак, и да зато греши у ситуацијама када наиђе на, за тај период, некарактеристичну концентрацију ових честица. Међутим, видимо и да модел заправо и даље ради како треба када то није проблем. То значи да можемо да искористимо $\mathcal{M}_S$, али тако да додамо члан који ће се односити само на концентрацију штетних честица, а који ћемо добити тренирањем над скупом података из ове године.

Овај пример нам је важан из два разлога. Прво, интуитивно, он показује да је смислен пренос знања *локална* операција у простору коефицијената: желимо да променимо оно што треба да се промени, а да задржимо оно што и даље важи. Друго, он већ наговештава улогу интерпретабилности: ако смо у стању да покажемо да је, у оцени који смо добили преносом знања, коефицијент уз концентрацију штетних честица знатно већи у односу на изворни, а остали углавном слични, добили смо не само боље предвиђање него и смислен опис онога што се променило између два домена.

Овим је општа слика преноса наученог знања уведена. Питање зашто би овакво повезивање два модела уопште било корисно, и како конкретно га треба извести у линеарној регресији, тема је наредног одељка.

2.3 Зашто преносити знање?

Вратимо се на двојну поставку из (2.6): велик изворни скуп $\{X_S, \mathbf{Y}_S\}$ и мали цильни скуп $\{X_T, \mathbf{Y}_T\}$, описани на неки начин правим коефицијентима β_S и β_T који су различити. Нека су непристрасне оцене ових коефицијената $\hat{\beta}_S$ и $\hat{\beta}_T$. При малом N_T, грешка цильне оцене може бити неприхватљиво велика, док изворна оцена $\hat{\beta}_S$, услед велике величине узорка, има веома малу дисперзију, али је, гледано као оцена непознатог цильног вектора β_T, пристрасна, са пристрасношћу једнаком $\mathbb{E}(\hat{\beta}_S) - \beta_T = \beta_S - \beta_T$. Две оцене, дакле, греше на супротне начине, један кроз дисперзију, други кроз пристрасност. Природно је покушати компромис, преко линеарне комбинације ове две оцене. Размотримо најједноставнији случај, конвексну комбинацију

$$\hat{\beta}(\omega) = \omega\hat{\beta}_T + (1-\omega)\hat{\beta}_S, \qquad \omega \in [0,1]. \tag{2.7}$$

За $\omega = 1$ добијамо цильну оцену, а за $\omega = 0$ изворну. Зарад прегледности, у наставку овог одељка третирамо изворну оцену као детерминистичку, једнаку β_S, што је смислена апроксимација у граничном случају $N_S \to \infty$, јер тада дисперзија изворне оцене нестаје. Навешћемо и шта ако то није случај.

Под овом претпоставком, тј. замењујући $\hat{\beta}_S$ са β_S у (2.7), добијамо $\hat{\beta}(\omega) = \omega\hat{\beta}_T + (1-\omega)\beta_S$. Грешку у односу на β_T растављамо на случајни и детерминистички део:

$$\hat{\beta}(\omega) - \beta_T = \omega\hat{\beta}_T + (1-\omega)\beta_S - \beta_T = \omega(\hat{\beta}_T - \beta_T) + (1-\omega)(\beta_S - \beta_T).$$

Први сабирак је случајна величина са $\mathbb{E}[\hat{\beta}_T - \beta_T] = 0$; други је детерминистички. Развијањем квадрата норме, мешовити члан чије очекивање рачунамо постаје нула,

$$\mathbb{E}\left[\|\hat{\beta}(\omega) - \beta_T\|_2^2\right] = \omega^2\,\mathbb{E}\left[\|\hat{\beta}_T - \beta_T\|_2^2\right] + (1-\omega)^2\,\|\beta_S - \beta_T\|_2^2,$$

где ниједан члан заправо не зависи од конкретног избора ω, већ су својства оцена, односно података. Уводећи ознаке

$$A := \mathbb{E}\left[\|\hat{\beta}_T - \beta_T\|_2^2\right], \qquad C := \|\beta_S - \beta_T\|_2^2,$$

добијамо

$$\mathcal{R}(\omega) := \mathbb{E}\left[\|\hat{\beta}(\omega) - \beta_T\|_2^2\right] = \omega^2 A + (1-\omega)^2 C. \tag{2.8}$$

Минимизација по ω. $\mathcal{R}(\omega)$ је квадратна функција по ω са позитивним водећим коефицијентом $A + C$, па је строго конвексна и има јединствен минимум.

Из услова

$$\frac{d}{d\omega}\mathcal{R}(\omega) = 2A\omega - 2C(1-\omega) = 0$$

добијамо $\omega(A+C) = C$, тј.

$$\omega^\star = \frac{C}{A+C} = \frac{\|\beta_S - \beta_T\|_2^2}{\sigma_T^2 \operatorname{tr}\big((X_T^\top X_T)^{-1}\big) + \|\beta_S - \beta_T\|_2^2} \in [0,1]. \tag{2.9}$$

Заменом у (2.8), вредност грешке у оптимуму је

$$\mathcal{R}(\omega^\star) = \left(\tfrac{C}{A+C}\right)^2 A + \left(\tfrac{A}{A+C}\right)^2 C = \frac{AC(C+A)}{(A+C)^2} = \frac{A \cdot C}{A+C},$$

одакле, коришћењем неједнакости $\frac{AC}{A+C} \leq \min(A,C)$ такође добијамо:

$$\mathcal{R}(\omega^\star) \leq \min(A,C). \tag{2.10}$$

Из израза (2.9) и (2.10) читамо пар битних ствари, које смо интуитивно и могли да претпоставимо. Прво, оптимална мешана оцена увек постиже грешку строго мању од грешке циљне оцене A (осим у граничном случају када $C \to \infty$). Друго, када је циљни узорак мали (A велико), или када су циљни и изворни подаци блиски (C мало), треба да се у већој мери ослонимо на изворни модел. Тек када A постане мало (имамо довољно циљних података) или C велико (изворни модел је неупотребљив), $\omega^\star$ тежи јединици и враћамо се чистом циљном моделу.

Овим смо мотивисали основну идеју преноса наученог знања: ако имамо стабилну, али помало погрешну информацију из изворног модела, можемо је користити да умањимо дисперзију циљног модела по цену увођења контролисане пристрасности, баш у духу класичне регуларизације [4]. Тежински коефицијент ω је, наравно, најгрубља могућа контрола јер она третира све правце у простору коефицијената подједнако.

У глави 3 видећемо како различите методе преноса у случају линеарне регресије сви дају оцене облика

$$\hat{\beta}_{TL} = W\hat{\beta}_T + (I - W)\hat{\beta}_S,$$

за неку матрицу $W \in \mathbb{R}^{D\times D}$, у складу са управо изведеном интуицијом. Преласком са скалара ω на матрицу W управо прелазимо са *једнаког* третмана свих праваца на њихово разликовање, јер, као што смо у примеру видели, могуће је да су нам изворни коефицијенти корисни у једним правцима (у којима је пристрасност $\hat{\beta}_S - \hat{\beta}_T$ мала), а мање корисни у другим (у којима је велика).

Глава 3

Пренос наученог знања у линеарној регресији

Циљ ове главе је да размотримо мноштво различитих метода помоћу којих можемо да спроведемо пренос знања, где ћемо на крају видети да све те методе воде ка сличним резултатима. Показаћемо да се све ове методе своде на матричну линеарну комбинацију неких оцена (које су углавном циљна и изворна, али видећемо и друге случајеве).

Конкретизујемо дефиницију 1 на оно што је нама у раду значајно, а пре тога уведимо и нотацију.

Претпостављамо да су нам дата два линеарна модела:

$$\mathbf{Y}_S = X_S\beta_S + \varepsilon_S, \qquad \mathbf{Y}_T = X_T\beta_T + \varepsilon_T, \tag{3.1}$$

где индекси S и T означавају редом изворни (*source*) и циљни (*target*) домен. Матрице $X_S \in \mathbb{R}^{N_S\times D}$ и $X_T \in \mathbb{R}^{N_T\times D}$ имају исти број колона D, али, по правилу, $N_S \gg N_T$, то јест, у изворном домену располажемо великим узорком, док је циљни узорак мали. Шумови $\varepsilon_S \sim \mathcal{N}(0, \sigma_S^2 I_{N_S})$ и $\varepsilon_T \sim \mathcal{N}(0, \sigma_T^2 I_{N_T})$ су наравно независни, а вектори коефицијената $\beta_S, \beta_T \in \mathbb{R}^D$ су различити. Одговарајуће оцене непознатих коефицијената су

$$\hat{\beta}_S = (X_S^\top X_S)^{-1} X_S^\top \mathbf{Y}_S, \qquad \hat{\beta}_T = (X_T^\top X_T)^{-1} X_T^\top \mathbf{Y}_T,$$

и њихове особине следе директно из (2.3)–(2.5) примењених на одговарајући домен.

Дефиниција 2. Дати су изворни скуп података $\{X_S, \mathbf{Y}_S\}$ и циљни скуп података $\{X_T, \mathbf{Y}_T\}$, са линеарним моделима, и оценама њихових коефицијената

$\hat{\beta}_S$ и $\hat{\beta}_T$. *Пренос научености знања у линеарној регресији* је било који метод којим долазимо до коефицијената $\hat{\beta}_{TL}$ тако да за предвиђање модела $\mathcal{M}_{TL}$ важи $\mathcal{R}(\mathcal{M}_{TL}) < \mathcal{R}(\mathcal{M}_T)$, где је $\mathcal{R}$ унапред изабрана функција ризика. Стандардно ћемо узимати $\mathbb{E}[(y-\hat{y})^2]$; а тамо где ће нам интерпретација бити важнија од предикције, посматраћемо и грешку саме процене коефицијената, $\mathbb{E}[\|\hat{\beta}_{TL} - \beta_T\|_2^2]$.

Формална дефиниција не прецизира облик тражене оцене. Као што смо навели у глави 2, у овом раду држаћемо се матричног облика

$$\hat{\beta}_{TL} = W\hat{\beta}_T + (I - W)\hat{\beta}_S, \qquad W \in \mathbb{R}^{D\times D}. \tag{3.2}$$

Овај облик је природан, али, оно што је важније, испоставиће се да неколико независних мотивација, које ћемо редом обрадити, воде до оваквог облика оцене. Најближе полазиште је тражење $\hat{\beta}_{TL}$ као минимизацију функције губитка која кажњава одступање од изворног модела, то јест генерализација гребене[1] регуларизације разматрана и код [1]. Овај чисто оптимизациони приступ ћемо, у наставку, размотрити и из бајесовског[2] угла, а затим и у оквиру финог подешавања помоћу алгоритма опадајућег градијента[3]. Коначно, поменућемо и дестилацију знања, која је далеко моћнија у генералном случају машинског учења (када се не ослањамо искључиво на линеарну регресију).

3.1 Извођење оцена

Први приступ подсећа на извођење оптималних коефицијената у случају класичне линеарне регресије, али у израз који желимо да минимизујемо додајемо и члан који се односи на изворни модел, и то на следећи начин [6]:

$$L(\beta) = \tfrac{1}{2}\|\mathbf{Y}_T - X_T\beta\|_2^2 + \tfrac{\lambda}{2}\|\beta - \hat{\beta}_S\|_2^2. \tag{3.3}$$

Параметар $\lambda \geq 0$ је регуларизациони, и као што смо рекли, подсећа на гребену регуларизацију [4]. Минимум тражимо анулирањем градијента. Први члан је онај из извођења стандардне оцене (2.2)–(2.3),

$$\nabla_\beta \tfrac{1}{2}\|\mathbf{Y}_T - X_T\beta\|_2^2 = -X_T^\top \mathbf{Y}_T + X_T^\top X_T\beta,$$

[1] енгл. ridge
[2] Thomas Bayes (1702–1761)
[3] енгл. gradient descent

а други члан рачунамо директно. Како је $\frac{1}{2}\|\beta - \hat{\beta}_S\|_2^2 = \frac{1}{2}(\beta - \hat{\beta}_S)^\top(\beta - \hat{\beta}_S)$, квадратна форма са јединичном матрицом I_D, опште правило $\nabla_\beta \frac{1}{2}(\beta-a)^\top A(\beta-a) = A(\beta - a)$ (за симетрично A) даје

$$\nabla_\beta \tfrac{1}{2}\|\beta - \hat{\beta}_S\|_2^2 = \beta - \hat{\beta}_S.$$

Даље, сабирањем два градијента,

$$\nabla_\beta L(\beta) = -X_T^\top \mathbf{Y}_T + X_T^\top X_T \beta + \lambda(\beta - \hat{\beta}_S) = (X_T^\top X_T + \lambda I)\beta - (X_T^\top \mathbf{Y}_T + \lambda \hat{\beta}_S),$$

те, будући да је Хесијан $\nabla^2 L = X_T^\top X_T + \lambda I$ позитивно дефинитан за $\lambda > 0$, функција је строго конвексна и јединствен минимум се добија баш у нули:

$$\boxed{\hat{\beta}_\lambda = (X_T^\top X_T + \lambda I)^{-1}(X_T^\top \mathbf{Y}_T + \lambda \hat{\beta}_S).} \tag{3.4}$$

Матрични облик. Ако у изразу (3.4) циљну оцену запишемо преко $\hat{\beta}_T = (X_T^\top X_T)^{-1} X_T^\top \mathbf{Y}_T$, добијамо

$$\hat{\beta}_\lambda = H_\lambda \hat{\beta}_T + (I - H_\lambda)\hat{\beta}_S, \quad H_\lambda := (X_T^\top X_T + \lambda I)^{-1} X_T^\top X_T. \tag{3.5}$$

Матрица H_λ игра управо улогу коју смо у одељку 2.3 најавили, а то је да нам она говори колику тежину дајемо циљној оцени. Међутим сада та тежина зависи од правца у простору коефицијената. Преко сопствене базе матрице $S_T := X_T^\top X_T$ ћемо у одељку 3.3 дискутовати особине ове оцене, а, пре тога, размотрићемо (3.4) из бајесовског угла, да бисмо се уверили да ово није произвољан избор.

3.2 Бајесовска интерпретација

Основна идеја је да је оцена (3.4) апостериорно очекивање под природном априорном претпоставком, да су циљни коефицијенти, до на шум, једнаки изворним [6]. Формализујмо:

$$\beta \sim \mathcal{N}(\beta_S, \tau^2 I_D), \qquad \mathbf{Y}_T \mid \beta \sim \mathcal{N}(X_T\beta, \sigma_T^2 I_{N_T}). \tag{3.6}$$

Извођење апостериорне расподеле. По Бајесовој теореми, апостериорна густина је пропорционална производу густина из (3.6), па:

$$p(\beta \mid \mathbf{Y}_T) \propto p(\mathbf{Y}_T \mid \beta)\, p(\beta) \propto \exp\Big(-\tfrac{1}{2\sigma_T^2}\|\mathbf{Y}_T - X_T\beta\|_2^2 - \tfrac{1}{2\tau^2}\|\beta - \beta_S\|_2^2\Big).$$

Означимо експонент са $-\frac{1}{2}Q(\beta)$, где је

$$Q(\beta) = \tfrac{1}{\sigma_T^2}\|\mathbf{Y}_T - X_T\beta\|_2^2 + \tfrac{1}{\tau^2}\|\beta - \beta_S\|_2^2.$$

Пошто је $Q(\beta)$ строго конвексна квадратна форма по β, апостериорна расподела је Гаусова[4], $\beta \mid \mathbf{Y}_T \sim \mathcal{N}(\mu_{\text{post}}, \Sigma_{\text{post}})$, где се μ_{post} и Σ_{post} читају из разлагања $Q(\beta) = (\beta - \mu_{\text{post}})^\top \Sigma_{\text{post}}^{-1}(\beta - \mu_{\text{post}}) + \text{const}$. Развијањем сваког члана по β, и третирањем $\hat{\beta}_S$ као фиксног (водећи се истом логиком као у одељку 2.3),

$$\begin{aligned}\tfrac{1}{\sigma_T^2}\|\mathbf{Y}_T - X_T\beta\|_2^2 &= \tfrac{1}{\sigma_T^2}\left(\beta^\top X_T^\top X_T\beta - 2\beta^\top X_T^\top \mathbf{Y}_T\right) + \text{const},\\ \tfrac{1}{\tau^2}\|\beta - \hat{\beta}_S\|_2^2 &= \tfrac{1}{\tau^2}\left(\beta^\top\beta - 2\beta^\top\hat{\beta}_S\right) + \text{const}.\end{aligned}$$

Сабирањем и груписањем по степену β,

$$Q(\beta) = \beta^\top\left(\tfrac{1}{\sigma_T^2}X_T^\top X_T + \tfrac{1}{\tau^2}I\right)\beta - 2\beta^\top\left(\tfrac{1}{\sigma_T^2}X_T^\top\mathbf{Y}_T + \tfrac{1}{\tau^2}\hat{\beta}_S\right) + \text{const},$$

и поређењем са општим обликом $\beta^\top\Sigma_{\text{post}}^{-1}\beta - 2\beta^\top\Sigma_{\text{post}}^{-1}\mu_{\text{post}}$ следи

$$\Sigma_{\text{post}}^{-1} = \tfrac{1}{\sigma_T^2}X_T^\top X_T + \tfrac{1}{\tau^2}I, \tag{3.7}$$

$$\mathbb{E}[\beta \mid \mathbf{Y}_T] = \mu_{\text{post}} = \Sigma_{\text{post}}\left(\tfrac{1}{\sigma_T^2}X_T^\top\mathbf{Y}_T + \tfrac{1}{\tau^2}\hat{\beta}_S\right). \tag{3.8}$$

Множењем бројиоца и имениоца унутар (3.8) са σ_T^2 добијамо

$$\mathbb{E}[\beta \mid \mathbf{Y}_T] = \left(X_T^\top X_T + \tfrac{\sigma_T^2}{\tau^2}I\right)^{-1}\left(X_T^\top\mathbf{Y}_T + \tfrac{\sigma_T^2}{\tau^2}\hat{\beta}_S\right),$$

што је, упоређујући са (3.4), тачно $\hat{\beta}_\lambda$ за

$$\lambda = \frac{\sigma_T^2}{\tau^2}. \tag{3.9}$$

Дакле, λ мери однос шума података и априорне несигурности: што више верујемо у изворни модел (мало τ^2), веће је λ, и тиме јаче повлачење ка β_S.

Дијагонална априорна коваријација. Претпоставка (3.6) додељује свакој координати вектора β исту меру априорне несигурности τ^2. Ако верујемо да се неки коефицијенти драстичније мењају између домена од других, природна генерализација је да дозволимо дијагоналну априорну коваријациону матрицу са

$$\beta \sim \mathcal{N}(\hat{\beta}_S, \Sigma_0), \qquad \Sigma_0 = \operatorname{diag}(\tau_1^2, \ldots, \tau_D^2), \quad \tau_j^2 > 0,$$

[4]Carl Friedrich Gauss (1777–1855)

где τ_j^2 мери априори несигурност у j-тој координати. Понављањем истог комплетирања квадрата, али са $\Sigma_0^{-1} = \mathrm{diag}(1/\tau_1^2, \ldots, 1/\tau_D^2)$ уместо $\frac{1}{\tau^2} I$, добијамо

$$\Sigma_{\text{post}}^{-1} = \tfrac{1}{\sigma_T^2} X_T^\top X_T + \Sigma_0^{-1},$$
$$\mathbb{E}[\beta \mid \mathbf{Y}_T] = \Sigma_{\text{post}} \left(\tfrac{1}{\sigma_T^2} X_T^\top \mathbf{Y}_T + \Sigma_0^{-1} \hat{\beta}_S \right).$$

Множењем са σ_T^2 добијамо аналогну једнакост са (3.4) али са матричним регуларизатором $\Lambda_0 := \sigma_T^2 \Sigma_0^{-1} = \mathrm{diag}(\sigma_T^2/\tau_1^2, \ldots, \sigma_T^2/\tau_D^2)$:

$$\hat{\beta}_{\Lambda_0} = (X_T^\top X_T + \Lambda_0)^{-1} (X_T^\top \mathbf{Y}_T + \Lambda_0 \hat{\beta}_S). \tag{3.10}$$

Скаларни случај (3.4) одговара избору $\tau_j^2 \equiv \tau^2$ за све j. Мање τ_j^2 (веће $[\Lambda_0]_{jj}$) значи већу априорну увереност да је j-ти циљни коефицијент близак изворном, па ће оцена у том правцу јаче повући ка $[\hat{\beta}_S]_j$. У пракси Σ_0 може доћи из претходне анализе варијабилности појединих предиктора између домена, или из коваријационе матрице изворне оцене (тамо где је извор мање поуздан, бирамо веће τ_j^2).

3.3 Сопствена база Грамове[5] матрице

Да бисмо разумели шта матрица H_λ из (3.5) заправо ради, дијагонализујмо Грамову матрицу $S_T = X_T^\top X_T$. Како је она симетрична и позитивно семидефинитна, постоји ортогонална матрица U и дијагонална матрица $\Lambda = \mathrm{diag}(d_1, \ldots, d_D)$, $d_j > 0$, тако да је

$$S_T = U \Lambda U^\top.$$

Колоне $u_1, \ldots, u_D$ матрице U су сопствени вектори S_T, а бројеви d_j сопствене вредности. Вектори u_j су правци у простору предиктора, а d_j мере колико у том правцу имамо података. Велико d_j значи много информације (много варијације предиктора у том правцу), мало d_j значи да у том правцу немамо шта да оценимо, и оцена у том правцу постаје бесмислено велика.

У овој бази, лако рачунамо

$$H_\lambda = U(\Lambda + \lambda I)^{-1} \Lambda \, U^\top = U \, \mathrm{diag}\left(\frac{d_j}{d_j + \lambda} \right) U^\top,$$

[5] Jørgen Pedersen Gram (1850–1916)

па је, по координатама:

$$[\hat{\beta}_\lambda]_j = \frac{d_j}{d_j+\lambda}[\hat{\beta}_T]_j + \frac{\lambda}{d_j+\lambda}[\hat{\beta}_S]_j, \quad j = 1, \ldots, D, \tag{3.11}$$

где $[\cdot]_j$ означава j-ту координату у односу на базу $\{u_j\}$.

Ова формула нам говори:

- У правцима у којима имамо *пуно* циљних података (велико d_j), количник $d_j/(d_j+\lambda)$ је близу јединице, па оцена у том правцу практично игнорише извор и користи циљну оцену добијену методом најмањих квадрата. Ово и желимо, јер ако смо у стању да у том правцу поуздано оценимо коефицијент из циљних података, немамо разлога да се ослањамо на нешто друго.

- У правцима у којима имамо *мало* циљних података (мало d_j, а λ веће од d_j), количник $\lambda/(d_j+\lambda)$ је близу јединице, па оцена у том правцу практично игнорише циљ и користи изворни коефицијент. И ово је пожељно, у правцима где не знамо ништа, боље је ослонити се на извор, макар и уз пристрасност.

- Средњи правци се третирају континуирано између ове две крајности, у складу са односом d_j и λ.

Сада видимо да је пренос знања на овај начин локална трансформација, јер се дешава баш ту где нам је и потребан (у слабо оцењеним правцима), а не свугде. Зато је оваква оцена много боља од просте конвексне комбинације (2.7), скаларна тежина ω би истовремено ослабила и правце у којима је циљна оцена $\hat{\beta}_S$ сасвим у реду.

Пример 2. Вратимо се примеру 1. Претпоставимо да је концентрација штетних честица био предиктор који се раније практично није мењао, па је у изворним подацима одговарајући правац у S_S имао малу сопствену вредност. У циљним подацима, тај предиктор има већу варијацију (пошто загађеност значајно осцилује), па је одговарајуће d_j у S_T веће. Нека је λ фиксирано.

Тада из (3.11) читамо да је удео циљне оцене значајан у правцима у којима је $d_j > \lambda$, па се коефицијент уз концентрацију штетних честица мења у односу на изворни, наравно, за погодно одабрано λ. Да смо изабрали "превелико" λ, ефекат привлачења циљне оцене би остао, али у мањој мери, па морамо пажљиво да бирамо овај коефицијент.

3.4 Добитак остварен преносом

Након што смо разумели структуру оцене, желимо да одговоримо на питање: када нам она заиста доноси добитак у односу на цильну оцену? Односно, за које λ важи

$$\mathbb{E}[\|\hat{\beta}_\lambda - \beta_T\|_2^2] < \mathbb{E}[\|\hat{\beta}_T - \beta_T\|_2^2] = \sigma_T^2 \operatorname{tr}(S_T^{-1})?$$

Да бисмо на ово одговорили, поновимо декомпозицију пристрасност–дисперзија за $\hat{\beta}_\lambda$. Ради јасноће, третираћемо изворну оцену као детерминистичку, $\hat{\beta}_S = \beta_S$; објашњење шта да радимо када $\hat{\beta}_S$ носи и свој шум даћемо на крају овог одељка.

Из (3.4) лако читамо очекивање

$$\mathbb{E}[\hat{\beta}_\lambda] = (S_T + \lambda I)^{-1}(X_T^\top X_T \beta_T + \lambda \beta_S) = H_\lambda \beta_T + (I - H_\lambda)\beta_S,$$

па је пристрасност

$$\mathbb{E}[\hat{\beta}_\lambda] - \beta_T = -(I - H_\lambda)(\beta_T - \beta_S) = -(I - H_\lambda)\,\delta, \quad \delta := \beta_T - \beta_S.$$

За дисперзију $\hat{\beta}_\lambda$ крећемо од еквивалентног записа (3.5), $\hat{\beta}_\lambda = H_\lambda \hat{\beta}_T + (I - H_\lambda)\beta_S$. Како за множење константном матрицом важи $\operatorname{Cov}(A\xi) = A \operatorname{Cov}(\xi) A^\top$, следи

$$\operatorname{Cov}(\hat{\beta}_\lambda) = \operatorname{Cov}(H_\lambda \hat{\beta}_T) = H_\lambda \operatorname{Cov}(\hat{\beta}_T)\, H_\lambda^\top.$$

Из (2.5) знамо $\operatorname{Cov}(\hat{\beta}_T) = \sigma_T^2 S_T^{-1}$, а из (3.5) је $H_\lambda = (S_T + \lambda I)^{-1} S_T$, па

$$\operatorname{Cov}(\hat{\beta}_\lambda) = \sigma_T^2 H_\lambda S_T^{-1} H_\lambda^\top = \sigma_T^2 (S_T + \lambda I)^{-2} S_T,$$

где смо у последњем кораку користили симетричност H_λ (што је последица чињенице да и S_T и $S_T + \lambda I$ комутирају, па је $H_\lambda = S_T(S_T + \lambda I)^{-1}$) и скратили S_T^{-1} са S_T-ом у бројиоцу. Одатле је средња квадратна грешка

$$\operatorname{MSE}(\hat{\beta}_\lambda) = \underbrace{\|(I - H_\lambda)\delta\|_2^2}_{\text{пристрасност}} + \underbrace{\sigma_T^2 \operatorname{tr}\big((S_T + \lambda I)^{-2} S_T\big)}_{\text{дисперзија}}. \tag{3.12}$$

Сада ћемо видети и како овај израз изгледа у сопственој бази матрице S_T.

Тврђење 3.1 (Средња квадратна грешка у сопственој бази)**.** *Нека је* $\tilde{\delta} := U^\top \delta$ *(односно* $\tilde{\delta}_j := u_j^\top \delta$*, где је, као и раније,* $S_T = U \Lambda U^\top$*). Тада важи*

$$\operatorname{MSE}(\hat{\beta}_\lambda) = \sum_{j=1}^{D} \left[\left(\frac{\lambda}{d_j + \lambda} \right)^2 \tilde{\delta}_j^2 + \sigma_T^2 \, \frac{d_j}{(d_j + \lambda)^2} \right]. \tag{3.13}$$

Доказ. Како је $(S_T + \lambda I) = U(D + \lambda I)U^\top$, следи:

$$(S_T + \lambda I)^{-1} = U \operatorname{diag}\Big(\tfrac{1}{d_j+\lambda}\Big) U^\top,$$

$$H_\lambda = (S_T + \lambda I)^{-1} S_T = U \operatorname{diag}\Big(\tfrac{d_j}{d_j+\lambda}\Big) U^\top,$$

$$I - H_\lambda = U \operatorname{diag}\Big(\tfrac{\lambda}{d_j+\lambda}\Big) U^\top.$$

Даље, како је U ортонормирана матрица, за пристрасност важи: ($\|Ux\|_2 = \|x\|_2$ јер је $U^\top U = I$):

$$\begin{aligned}
\|(I - H_\lambda)\delta\|_2^2 &= \left\| U \operatorname{diag}\Big(\tfrac{\lambda}{d_j+\lambda}\Big) U^\top \delta \right\|_2^2 \\
&= \left\| U \operatorname{diag}\Big(\tfrac{\lambda}{d_j+\lambda}\Big) \tilde{\delta} \right\|_2^2 \\
&= \left\| \operatorname{diag}\Big(\tfrac{\lambda}{d_j+\lambda}\Big) \tilde{\delta} \right\|_2^2 \\
&= \sum_{j=1}^{D} \Big(\tfrac{\lambda}{d_j+\lambda}\Big)^2 \tilde{\delta}_j^2.
\end{aligned}$$

Коначно, за дисперзију важи:

$$\begin{aligned}
(S_T + \lambda I)^{-2} S_T &= U \operatorname{diag}\Big(\tfrac{1}{d_j+\lambda}\Big)^2 U^\top U \Lambda U^\top \\
&= U \operatorname{diag}\Big(\tfrac{1}{d_j+\lambda}\Big)^2 \Lambda U^\top \\
&= U \operatorname{diag}\Big(\tfrac{d_j}{(d_j+\lambda)^2}\Big) U^\top.
\end{aligned}$$

и затим (јер матрице комутирају приликом рачунања трага производа)

$$\begin{aligned}
\operatorname{tr}\big((S_T + \lambda I)^{-2} S_T\big) &= \operatorname{tr}\Big(U \operatorname{diag}\Big(\tfrac{d_j}{(d_j+\lambda)^2}\Big) U^\top\Big) \\
&= \operatorname{tr}\Big(\operatorname{diag}\Big(\tfrac{d_j}{(d_j+\lambda)^2}\Big)\Big) \\
&= \sum_{j=1}^{D} \tfrac{d_j}{(d_j+\lambda)^2}.
\end{aligned}$$

Дакле, убацивањем у (3.12) следи:

$$\operatorname{MSE}(\hat{\beta}_\lambda) = \sum_{j=1}^{D} \left[\Big(\tfrac{\lambda}{d_j+\lambda}\Big)^2 \tilde{\delta}_j^2 + \sigma_T^2 \tfrac{d_j}{(d_j+\lambda)^2} \right].$$

□

Ово је погодна репрезентација одакле лако читамо да ли и када пренос доноси добитак. За $\lambda = 0$ имамо $\mathrm{MSE}(\hat{\beta}_{\lambda=0}) = \mathrm{MSE}(\hat{\beta}_T) = \sigma_T^2 \sum_j 1/d_j$, док је извод у нули, по λ:

$$\left.\frac{d\ \mathrm{MSE}(\hat{\beta}_\lambda)}{d\lambda}\right|_{\lambda=0} = -2\sigma_T^2 \sum_{j=1}^{D} \frac{1}{d_j^2} < 0. \tag{3.14}$$

Дакле, извод грешке у нули је *строго негативан*, што значи да за било које довољно мало $\lambda > 0$ важи $\mathrm{MSE}(\hat{\beta}_\lambda) < \mathrm{MSE}(\hat{\beta}_T)$. Другим речима, макар минимално померање ка изворном моделу увек побољшава оцену у смислу средње квадратне грешке. Ово је формална потврда идеје коју смо мотивисали у одељку 2.3: постоји нетривијална корист од преноса знања, независна од квалитета извора.

За веће λ, први члан у (3.13) тежи $\sum_j \tilde{\delta}_j^2 = \|\delta\|_2^2$, што одражава чињеницу да у граници потпуно верујемо извору и наслеђујемо његову пристрасност. Ако је та пристрасност велика (тј. извор је далеко од циља), можемо направити грешку већу од оне коју би направила сама циљна оцена. Ова појава је позната као *негативан пренос* [8], па нам је потребан начин за оцењивање корисности преноса за конкретан избор параметара. То ће бити тема једног од наредних одељака.

Оптимално $\lambda^\star$ које минимизује (3.13) налазимо диференцирањем и изједначавањем са нулом:

$$\sum_j \frac{2\lambda d_j \tilde{\delta}_j^2 - 2\sigma_T^2 d_j}{(d_j + \lambda)^3} = 0.$$

Напоменимо и шта када не тврдимо да је $\hat{\beta}_S = \beta_S$. Ако $\hat{\beta}_S$ има коваријациону матрицу Σ_S и независан је од ε_T, лако се проверава да очекивање у (3.12) остаје исто (уз замену $\beta_S \mapsto \mathbb{E}[\hat{\beta}_S] = \beta_S$, ако је извор непристрасно оцењен), а дисперзија добија додатни члан

$$\mathrm{tr}\big((I - H_\lambda)\Sigma_S(I - H_\lambda)^\top\big),$$

који тежи нули уколико је извор постојано оцењен ($N_S \to \infty$ повлачи $\Sigma_S \to 0$). У свим експериментима у глави 4 биће $N_S \gg N_T$, што овај члан чини занемарљивим и оправдава да у главном тексту користимо детерминистичку варијанту.

3.5 Фино подешавање и алгоритам опадајућег градијента

До сада смо оцену $\hat{\beta}_\lambda$ разматрали у затвореном облику. Када D расте, или када је линеарни модел део шире мреже, природан начин тренирања је градијентни спуст. У том контексту, [7] су разматрали фино подешавање, са функцијом губитка датом са

$$J_T(\beta) = \tfrac{1}{2}\|\mathbf{Y}_T - X_T\beta\|_2^2,$$

која је накнадно минимизована методом градијентног спуста, док је за почетну оцену узета изворна $\hat{\beta}_S$, а за корак α. Градијент је дат са $\nabla J_T(\beta) = S_T\beta - X_T^\top \mathbf{Y}_T = S_T(\beta - \hat{\beta}_T)$, па се накнадне итерације добијају из

$$\beta_{k+1} = \beta_k - \alpha S_T(\beta_k - \hat{\beta}_T) = A\beta_k + (I - A)\hat{\beta}_T, \qquad A := I - \alpha S_T.$$

Размотавањем ове формуле до $\beta_0 = \hat{\beta}_S$ добијамо

$$\hat{\beta}_k^{\mathrm{FT}} = A^k\hat{\beta}_S + (I - A^k)\hat{\beta}_T, \qquad A = I - \alpha S_T. \tag{3.15}$$

Конвергенција ове итерације зависи од понашања матричних степена A^k. Подсетимо: *спектрални радијус* матрице A је $\rho(A) := \max_j |\mu_j(A)|$, где су $\mu_j(A)$ њене сопствене вредности. Познато је да $A^k \to 0$ када $k \to \infty$ ако и само ако је $\rho(A) < 1$ [5, видети Теорему 5.6.12, стр. 348]. За нашу $A = I - \alpha S_T$, пошто је S_T симетрична са сопственим вредностима $d_j > 0$, сопствене вредности матрице A су $1 - \alpha d_j$, па $\rho(A) < 1$ се своди на $|1 - \alpha d_j| < 1$ за свако j, тј. $0 < \alpha < 2/d_{\max}$, где је $d_{\max}$ највећа сопствена вредност S_T-а. У свему што следи претпостављамо овај избор корака учења.

Идеја (3.15) је очигледна: у почетним итерацијама, оцена је близу извора, а како k расте, приближава се чистој циљној оцени. Одатле видимо да рано заустављање игра улогу регуларизатора, у смислу да контролише количину преноса.

Природно уопштење које избегава ову слабост је да извор укључимо и у сам губитак, не само у иницијализацију. Нека је $P \succeq 0$ симетрична позитивно-полудефинитна матрица и $\beta_0 \in \mathbb{R}^D$ произвољан референтни вектор. Разматрамо функцију губитка

$$J(\beta) = \tfrac{1}{2}\|\mathbf{Y}_T - X_T\beta\|_2^2 + \tfrac{\lambda}{2}(\beta - \beta_0)^\top P(\beta - \beta_0), \qquad \lambda \geq 0, \tag{3.16}$$

што подсећа на гребену регуларизацију са почетком у β_0 са матрицом P (видети [4] и [1]). Имамо три занимљива специјална случаја:

- $P = I$, $\beta_0 = \hat{\beta}_S$, када регуларизациони члан постаје $\frac{\lambda}{2}\|\beta - \hat{\beta}_S\|_2^2$ и поклапа се са (3.3).
- $P = S_T$, $\beta_0 = \hat{\beta}_S$ када регуларизациони члан постаје $\frac{\lambda}{2}\|X_T(\beta - \hat{\beta}_S)\|_2^2$, који контролише разлику у предвиђањима на циљном скупу.
- $\lambda = 0$, које нас враћа на (3.15).

Сада изводимо затворени облик за k-ту итерацију градијентног спуста над (3.16) са иницијализацијом $\hat{\beta}_S$.

Тврђење 3.2 (Итерација уопштеног финог подешавања)**.** *Нека је $M = S_T + \lambda P$ и $A_M = I - \alpha M$. Претпоставимо $\rho(A_M) < 1$. Тада градијентни спуст на функцији губитка* (3.16)*, са кораком α и иницијализацијом $\hat{\beta}_S$, у k-тој итерацији даје*

$$\hat{\beta}_k = A_M^k \hat{\beta}_S + (I - A_M^k)\beta^\star, \qquad \beta^\star = M^{-1}(X_T^\top \mathbf{Y}_T + \lambda P \beta_0). \tag{3.17}$$

Доказ. Градијент (3.16) је

$$\nabla J(\beta) = X_T^\top (X_T\beta - \mathbf{Y}_T) + \lambda P(\beta - \beta_0) = (S_T + \lambda P)\beta - (X_T^\top \mathbf{Y}_T + \lambda P \beta_0).$$

Када у горњем изразу уведемо $M = S_T + \lambda P$ и $\beta^\star = M^{-1}(X_T^\top \mathbf{Y}_T + \lambda P \beta_0)$ (минимум губитка (3.16), добијен из $\nabla J(\beta) = 0$), градијент постаје

$$\nabla J(\beta) = M\beta - M\beta^\star = M(\beta - \beta^\star).$$

Одатле, за једну итерацију финог подешавања са кораком α добијамо:

$$\beta_{k+1} = \beta_k - \alpha M(\beta_k - \beta^\star) = (I - \alpha M)\beta_k + \alpha M \beta^\star = A_M \beta_k + (I - A_M)\beta^\star,$$

где смо у последњој једнакости користили $\alpha M = I - A_M$. Посматрамо $e_k := \beta_k - \beta^\star$, за шта важи

$$e_{k+1} = \beta_{k+1} - \beta^\star = A_M \beta_k + (I - A_M)\beta^\star - \beta^\star = A_M(\beta_k - \beta^\star) = A_M e_k.$$

Како је још $e_0 = Ie_0 = A_M^0 e_0$, индукцијом по k добијамо $e_k = A_M^k e_0$, па са иницијализацијом $\beta_0 = \hat{\beta}_S$ имамо $e_0 = \hat{\beta}_S - \beta^\star$ и

$$\hat{\beta}_k = \beta^\star + e_k = \beta^\star + A_M^k(\hat{\beta}_S - \beta^\star) = A_M^k \hat{\beta}_S + (I - A_M^k)\beta^\star,$$

што је управо (3.17). Корак α бирамо тако да је $\rho(A_M) < 1$, што гарантује конвергенцију $\hat{\beta}_k \to \beta^\star$; пошто је $M \succ 0$ са највећом сопственом вредношћу $m_{\max}$, довољно је $0 < \alpha < 2/m_{\max}$. □

За $\lambda = 0$ имамо $M = S_T$, $\beta^\star = \hat{\beta}_T$, па (3.17) даје (3.15). Прегледамо и преостала два случаја.

Последица 3.1. *За $P = I$ и $\beta_0 = \hat{\beta}_S$ важи $M = S_T + \lambda I$, па је $A_M = I - \alpha(S_T + \lambda I)$ и*

$$\beta^\star = (S_T + \lambda I)^{-1}(X_T^\top \mathbf{Y}_T + \lambda \hat{\beta}_S) = \hat{\beta}_\lambda,$$

затворена оцена из (3.4)*. Према томе,*

$$\hat{\beta}_k = A_M^k \hat{\beta}_S + (I - A_M^k)\hat{\beta}_\lambda,$$

што конвергира ка $\hat{\beta}_\lambda$ када $k \to \infty$.

Последица 3.2. *За $P = S_T$ и $\beta_0 = \hat{\beta}_S$ важи $M = (1 + \lambda)S_T$, па је $A_M = I - \alpha(1 + \lambda)S_T$ и*

$$\beta^\star = M^{-1}(X_T^\top \mathbf{Y}_T + \lambda S_T \hat{\beta}_S) = \tfrac{1}{1+\lambda}\hat{\beta}_T + \tfrac{\lambda}{1+\lambda}\hat{\beta}_S,$$

скаларна конвексна комбинација $\hat{\beta}_T$ и $\hat{\beta}_S$. Према томе,

$$\hat{\beta}_k = A_M^k \hat{\beta}_S + (I - A_M^k)\left(\tfrac{1}{1+\lambda}\hat{\beta}_T + \tfrac{\lambda}{1+\lambda}\hat{\beta}_S\right),$$

што конвергира ка $\frac{1}{1+\lambda}\hat{\beta}_T + \frac{\lambda}{1+\lambda}\hat{\beta}_S$ када $k \to \infty$.

3.6 Дестилација[6] наученог знања

Дестилација знања подразумева тренирање великог модела – *учитеља* – а затим и мањег модела – *ученика* – који у исто време погађа праве вредности као и вредности које је модел учитеља предвидео над подацима. Два основна разлога оваквог моделирања су стабилизација резултата (јер ће учитељ, на пример код бинарног погађања, имати ужу расподелу од стварне расподеле података), као и могућност тренирања мањих модела који ће се понашати слично као и већи, али ће услед мањег броја параметара бити ефикаснији. Дакле, ова метода се користи као регуларизација и компресија, у исто време [3].

[6]енгл. knowledge distillation

Специјализовано за линеарну регресију, учитељ је фиксирана линеарна оцена $\hat{\beta}_S \in \mathbb{R}^D$ са предвиђањима $\tilde{\mathbf{Y}}_T = X_T\hat{\beta}_S$ а ученик минимизује

$$J_T^{\mathrm{KD}}(\beta) = \tfrac{1-\lambda}{2}\|\mathbf{Y}_T - X_T\beta\|_2^2 + \tfrac{\lambda}{2}\|\tilde{\mathbf{Y}}_T - X_T\beta\|_2^2, \qquad \lambda \in [0,1]. \tag{3.18}$$

За $\lambda = 0$, ученик је просто циљна оцена; за $\lambda = 1$, ученик је једнак $\hat{\beta}_S$; за међувредности имамо интерполацију.

Али (3.18) је управо функција губитка дата са (3.16), са $P = S_T$, $\beta_0 = \hat{\beta}_S$ и репараметризацијом $\lambda_{\mathrm{KD}} = \lambda/(1-\lambda)$. Величина корака градијентног спуста се такође мења репараметризацијом, јер је $J_T^{KD} = (1-\lambda)J$, одакле је одговарајући корак сада $\alpha_{KD} = (1-\lambda)\alpha$. Трајекторија градијентног спуста онда следи из последице 3.2:

$$\hat{\beta}_k^{\mathrm{KD}} = A^k\hat{\beta}_S + (I - A^k)\big[(1-\lambda)\hat{\beta}_T + \lambda\hat{\beta}_S\big], \qquad A = I - \alpha_{KD}S_T. \tag{3.19}$$

Дакле, можемо посматрати дестилацију знања као специјалан случај финог подешавања, у ком тежина дестилације λ игра улогу јачине референце.

Уколико пак желимо да директно минимизујемо функцију губитка, напишимо оба члана као једну блок матрицу:

$$\tilde{X} = \begin{pmatrix} \sqrt{1-\lambda}\, X_T \\ \sqrt{\lambda}\, X_T \end{pmatrix}, \qquad \tilde{\mathbf{Y}} = \begin{pmatrix} \sqrt{1-\lambda}\, \mathbf{Y}_T \\ \sqrt{\lambda}\, \tilde{\mathbf{Y}}_T \end{pmatrix}.$$

Тада је

$$J_T^{\mathrm{KD}}(\beta) = \tfrac{1}{2}\|\tilde{\mathbf{Y}} - \tilde{X}\beta\|_2^2,$$

и стандардно решење $\hat{\beta}_{KD} = (\tilde{X}^\top\tilde{X})^{-1}\tilde{X}^\top\tilde{\mathbf{Y}}$ даје:

$$\tilde{X}^\top\tilde{X} = (1-\lambda)X_T^\top X_T + \lambda X_T^\top X_T = X_T^\top X_T,$$

$$\tilde{X}^\top\tilde{Y} = (1-\lambda)X_T^\top\mathbf{Y}_T + \lambda X_T^\top\tilde{\mathbf{Y}}_T,$$

па следи да је

$$\hat{\beta}^{\mathrm{KD}} = (X_T^\top X_T)^{-1}X_T^\top\big[(1-\lambda)\mathbf{Y}_T + \lambda X_T\hat{\beta}_S\big].$$

односно

$$\hat{\beta}^{\mathrm{KD}} = (1-\lambda)\,\hat{\beta}_T + \lambda\,\hat{\beta}_S,$$

што само по себи и није претерано занимљиво. Ипак, као што смо и рекли, корист ове методе управо и јесте у простој примени за компресију и регуларизацију [3], што овде свакако не можемо да приметимо.

3.7 Статистичко тестирање добитка оствареног преносом

Формула (3.13) нам говори све о томе када пренос помаже, али у пракси не знамо β_T ни σ_T^2, па ни δ. Не знамо ни β_S, али, као и до сада, сматраћемо да је $\beta_S = \hat{\beta}_S$ (јер је изворни скуп велики). Стога желимо да имамо статистички тест који, из података, одлучује да ли је пренос користан. Природна нулта хипотеза је да пренос не доноси разлику у односу на чисту циљну оцену, а алтернативна је да је пренос знања користан. Прецизније:

$$H_0 : \ \mathrm{MSE}(\hat{\beta}_\lambda) \geq \mathrm{MSE}(\hat{\beta}_T), \quad H_1 : \ \mathrm{MSE}(\hat{\beta}_\lambda) < \mathrm{MSE}(\hat{\beta}_T).$$

Да бисмо конструисали тест, посматрамо резидуалне суме квадрата две оцене:

$$\mathrm{RSS}_T = \|\mathbf{Y}_T - X_T\hat{\beta}_T\|_2^2, \quad \mathrm{RSS}_\lambda = \|\mathbf{Y}_T - X_T\hat{\beta}_\lambda\|_2^2.$$

Означимо $P = X_T(X_T^\top X_T)^{-1}X_T^\top$. Ово је пројектор, што је лако проверљиво: $P^2 = X_T(X_T^\top X_T)^{-1}X_T^\top X_T(X_T^\top X_T)^{-1}X_T^\top = X_T(X_T^\top X_T)^{-1}X_T^\top = P$, и $P^\top = (X_T(X_T^\top X_T)^{-1}X_T^\top)^\top = X_T(X_T^\top X_T)^{-1}X_T^\top = P$.

Из $\hat{\beta}_T = (X_T^\top X_T)^{-1}X_T^\top\mathbf{Y}_T$ следи $X_T\hat{\beta}_T = P\mathbf{Y}_T$, па је

$$\mathbf{Y}_T - X_T\hat{\beta}_T = (I-P)\mathbf{Y}_T = (I-P)(X_T\beta_T + \varepsilon_T).$$

Како је $PX_T = X_T(X_T^\top X_T)^{-1}X_T^\top X_T = X_T$, имамо $(I-P)X_T = 0$, те

$$\mathbf{Y}_T - X_T\hat{\beta}_T = (I-P)\varepsilon_T.$$

Даље, важи:

- $(I-P)^\top = I-P$;
- $(I-P)^2 = I - 2P + P^2 = I-P$;
- $\mathrm{rang}(P) = D$ (пројекција на $\mathrm{col}(X_T)$), па $\mathrm{rang}(I-P) = N_T - D$.

Ако је A симетрична и идемпотентна матрица ранга r, и $\varepsilon \sim \mathcal{N}(0, \sigma^2 I)$, тада

$$\varepsilon_T^\top A\varepsilon_T/\sigma^2 \sim \chi^2(r).$$

Применом на $A = I-P$ и коришћењем $\|(I-P)\varepsilon_T\|^2 = \varepsilon_T^\top(I-P)^\top(I-P)\varepsilon_T = \varepsilon_T^\top(I-P)\varepsilon_T$, закључујемо

$$\mathrm{RSS}_T = \|(I-P)\varepsilon_T\|^2$$

$$\mathrm{RSS}_T/\sigma_T^2 \sim \chi^2(N_T - D).$$

Даље, нека је

$$r_T = \mathbf{Y}_T - X_T\hat{\beta}_T, \quad r_\lambda = \mathbf{Y}_T - X_T\hat{\beta}_\lambda;$$
$$\mathrm{RSS}_T = \|r_T\|^2, \quad \mathrm{RSS}_\lambda = \|r_\lambda\|^2.$$

Тада

$$r_\lambda = r_T + X_T(\hat{\beta}_T - \hat{\beta}_\lambda),$$

па важи

$$\|r_\lambda\|^2 = \|r_T\|^2 + 2\, r_T^\top X_T(\hat{\beta}_T - \hat{\beta}_\lambda) + \|X_T(\hat{\beta}_T - \hat{\beta}_\lambda)\|^2.$$

Средњи сабирак је заправо нула, јер

$$r_T^\top X_T = \mathbf{Y}_T^\top X_T - \hat{\beta}_T^\top X_T^\top X_T = \mathbf{Y}_T^\top X_T - \mathbf{Y}_T^\top X_T (X_T^\top X_T)^{-1} X_T^\top X_T = 0$$

па следи да је

$$\mathrm{RSS}_\lambda = \mathrm{RSS}_T + \|X_T(\hat{\beta}_T - \hat{\beta}_\lambda)\|^2,$$

односно:

$$\mathrm{RSS}_\lambda - \mathrm{RSS}_T = \|X_T(\hat{\beta}_T - \hat{\beta}_\lambda)\|^2.$$

Даље, одузимањем (3.13) и $\mathrm{MSE}(\hat{\beta}_T) = \sigma_T^2 \sum_j 1/d_j$ добијамо

$$\mathrm{MSE}(\hat{\beta}_\lambda) - \mathrm{MSE}(\hat{\beta}_T) = \sum_j \left(\tfrac{\lambda}{d_j+\lambda}\right)^2 \tilde{\delta}_j^2 - \sigma_T^2 \sum_j \left[\tfrac{1}{d_j} - \tfrac{d_j}{(d_j+\lambda)^2}\right].$$

Дакле, уз алгебарски идентитет

$$\tfrac{1}{d_j} - \tfrac{d_j}{(d_j+\lambda)^2} = \tfrac{(d_j+\lambda)^2 - d_j^2}{d_j(d_j+\lambda)^2} = \tfrac{\lambda(2d_j+\lambda)}{d_j(d_j+\lambda)^2},$$

неједнакост из H_0 добија еквивалентан облик

$$H_0 \iff \sum_j \left(\tfrac{\lambda}{d_j+\lambda}\right)^2 \tilde{\delta}_j^2 \geq \sigma_T^2 c_\lambda, \tag{3.20}$$

$$c_\lambda := \sum_j \tfrac{\lambda(2d_j+\lambda)}{d_j(d_j+\lambda)^2},$$

где је c_λ функција само од λ и спектра S_T, дакле, у потпуности позната из података.

Како је

$$\begin{aligned}
\hat{\beta}_T - \hat{\beta}_\lambda &= (S_T+\lambda I)^{-1}(S_T+\lambda I)\hat{\beta}_T - (S_T+\lambda I)^{-1}(X_T^\top \mathbf{Y}_T + \lambda\hat{\beta}_S) \\
&= (S_T+\lambda I)^{-1}\big[(S_T\hat{\beta}_T+\lambda\hat{\beta}_T) - (X_T^\top \mathbf{Y}_T + \lambda\hat{\beta}_S)\big] \\
&= (S_T+\lambda I)^{-1}\big[(X_T^\top \mathbf{Y}_T+\lambda\hat{\beta}_T) - (X_T^\top \mathbf{Y}_T + \lambda\hat{\beta}_S)\big] \\
&= \lambda(S_T+\lambda I)^{-1}(\hat{\beta}_T - \hat{\beta}_S) \\
&= \lambda(S_T+\lambda I)^{-1}\big[(\beta_T + S_T^{-1}X_T^\top\varepsilon_T) - \hat{\beta}_S\big] \\
&= \lambda(S_T+\lambda I)^{-1}\big[S_T^{-1}X_T^\top\varepsilon_T + \delta\big].
\end{aligned}$$

онда сменом $z := U^\top X_T^\top \varepsilon_T/\sigma_T$ добијамо

$$\begin{aligned}
\mathrm{Cov}(z) &= \mathrm{Cov}\Big(\tfrac{U^\top X_T^\top \varepsilon_T}{\sigma_T}\Big) \\
&= \tfrac{1}{\sigma_T^2}\, U^\top X_T^\top \,\mathrm{Cov}(\varepsilon_T)\, X_T U \\
&= \tfrac{1}{\sigma_T^2}\, U^\top X_T^\top (\sigma_T^2 I_{N_T})\, X_T U \\
&= U^\top X_T^\top X_T U \\
&= U^\top S_T U \\
&= U^\top (U\Lambda U^\top) U = \Lambda,
\end{aligned}$$

тј. $z_j = \sqrt{d_j} Z_j$ уз $Z_j \sim \mathcal{N}(0,1)$, где су све Z_j међусобно независне. Пројектовано у сопствену базу,

$$[U^\top(\hat{\beta}_T - \hat{\beta}_\lambda)]_j = \tfrac{\lambda}{d_j+\lambda}\left[\tfrac{\sigma_T Z_j}{\sqrt{d_j}} + \tilde{\delta}_j\right],$$

а како је $\|X_T v\|^2 = v^\top S_T v = \sum_j d_j (U^\top v)_j^2$, следи

$$\frac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\sigma_T^2} = \sum_j \Big(\tfrac{\lambda}{d_j+\lambda}\Big)^2 \Big(Z_j + \tfrac{\sqrt{d_j}\,\tilde{\delta}_j}{\sigma_T}\Big)^2.$$

Из $\mathbb{E}[(Z_j+m_j)^2] = 1 + m_j^2$ уз $m_j = \sqrt{d_j}\,\tilde{\delta}_j/\sigma_T$ добија се

$$\mathbb{E}\Big[\tfrac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\sigma_T^2}\Big] = a_\lambda + b_\lambda(\tilde{\delta}), \qquad a_\lambda := \sum_j \Big(\tfrac{\lambda}{d_j+\lambda}\Big)^2, \quad b_\lambda(\tilde{\delta}) := \sum_j \Big(\tfrac{\lambda}{d_j+\lambda}\Big)^2 \tfrac{d_j\tilde{\delta}_j^2}{\sigma_T^2}.$$

Означавајући $\alpha_j := (\lambda/(d_j+\lambda))^2 \tilde{\delta}_j^2/\sigma_T^2 \geq 0$, еквиваленција (3.20) постаје $\sum_j \alpha_j \geq c_\lambda$, док је $b_\lambda = \sum_j d_j\alpha_j \geq d_{\min}\sum_j \alpha_j \geq d_{\min} c_\lambda$.

Сада дефинишемо и тест статистику, за коју смо до сада припремали терен, и то са

$$T := \frac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\hat{\sigma}_T^2 \cdot s_\lambda}, \qquad \hat{\sigma}_T^2 = \tfrac{\mathrm{RSS}_T}{N_T - D}, \qquad s_\lambda = \inf_{H_0} \mathbb{E}\Big[\tfrac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\sigma_T^2}\Big]. \tag{3.21}$$

Расподела ове статистике зависи од положаја b_λ у целом простору. Предлажемо калибрацију теста у тачки где је најмања вредност очекивања $\mathbb{E}\left[\frac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\sigma_T^2}\right]$ (са идејом да је тада одбацивање најлакше, и да желимо да контрола нивоа α важи свуда). Како ћемо нулту хипотезу одбацивати у случају да $T < c$, за погодно изабрано c, под нашом претпоставком ћемо за b_λ узети најмању могућу вредност, односно да је $s_\lambda = a_\lambda + d_{\min} c_\lambda$, и тада ће важити $\mathbb{E}(T) \geq 1$, док ваљаност овог приступа проверавамо симулацијама.

Остаје још да одредимо расподелу, јер све величине које учествују у тест статистици већ знамо. Такође, овде нам је важна претпоставка да су грешке нормалне, јер једино тада важе све наредне изведене расподеле случајних величина. Имамо да је

$$T = \frac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\hat{\sigma}_T^2 \cdot s_\lambda} = \frac{\frac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\sigma_T^2 s_\lambda}}{\frac{\hat{\sigma}_T^2}{\sigma_T^2}} = \frac{U}{V},$$

где важи $V = \frac{\frac{\mathrm{RSS}_T}{N_T - D}}{\sigma_T^2} = \frac{\frac{\mathrm{RSS}_T}{\sigma_T^2}}{N_T - D} \sim \frac{\chi^2(N_T - D)}{N_T - D}$, уз конвенцију да $W \sim \frac{F}{r}$ значи исто што и $rW \sim F$, као и да је $\chi^2(k)$ хи-квадратна расподела са k степени слободе.

Приметимо и да су U и V независне, јер

$$X_T^\top \mathbf{Y}_T = X_T^\top X_T \beta_T + X_T^\top P \varepsilon_T, \qquad X_T^\top (I - P) = 0,$$

па важи

$$U = f(P\varepsilon_T, \hat{\beta}_S).$$

С друге стране,

$$\mathrm{RSS}_T = \varepsilon_T^\top (I - P)\varepsilon_T, \qquad V = g((I - P)\varepsilon_T).$$

Како је

$$\mathrm{Cov}(P\varepsilon_T, (I - P)\varepsilon_T) = \sigma_T^2 P(I - P) = 0,$$

нормалност грешака и независност изворног и циљног узорка дају независност U и V.

Даље, U је тежинска средина померених χ^2 расподела, која нема расподелу која се пријатно може записати у затвореној форми. Међутим, послужићемо се апроксимацијом Сатервајта[7], тако што ћемо расподелу свакако апроксимирати χ^2 расподелом са одговарајућим бројем степени слободе, тако што ћемо изједначити прва два момента [9]. Ова апроксимација демонстрирано добро ради у

[7] Franklin Eves Satterthwaite (1914–1989)

случају када имамо тежинску средину стандардних χ^2 расподела, међутим, за веће σ_T (што очекујемо), померање ових расподела није толико значајно, па апроксимација свакако има смисла.

У додатку А изводимо ову апроксимацију, па добијамо:

$$k := \frac{s_\lambda^2}{\sum_j \left(\frac{\lambda}{d_j+\lambda}\right)^4 + 2\left(\frac{\lambda}{d_{j_{\min}}+\lambda}\right)^4 m^{\star 2}}, \qquad m^\star := \frac{\sqrt{d_{j_{\min}}}\tilde{\delta}^\star_{j_{\min}}}{\sigma_T},$$

односно $U \sim \frac{\chi^2(k)}{k}$. Коначно, следи:

$$T = \frac{U}{V} \overset{\text{approx}}{\sim} \frac{\frac{\chi^2(k)}{k}}{\frac{\chi^2(N_T-D)}{N_T-D}} \overset{\text{approx}}{\sim} F(k, N_T - D)$$

на најнеповољнијој тачки H_0, где је са F означена Фишерова[8] расподела, док са $Z(k) \overset{\text{approx}}{\sim} G$ означавамо да случајна величина $Z(k)$ има приближно функцију расподеле G, односно да $Z(k) \xrightarrow{d} Z \sim G$ за $k \to \infty$.

Дакле, нулту хипотезу одбацујемо са нивоом α у случају да важи

$$T < F_\alpha(k, N_T - D),$$

где је F_α доњи α-квантил Фишерове расподеле. Као што смо и рекли, све величине d_j, a_λ, c_λ, $d_{\min}$, s_λ, k, RSS_T, RSS_λ, $\hat{\sigma}_T^2$ су функције X_T, $\mathbf{Y}_T$, $\hat{\beta}_S$ и λ; нигде не претпостављамо познавање δ или σ_T^2.

[8]Ronald Aylmer Fisher (1890–1962)

Глава 4

Примена

У овој глави примењујемо развијену методологију, најпре на симулираним подацима, код којих су прави коефицијенти познати па се теорија може проверити егзактно, а затим и на реалне податке о хемијском саставу и квалитету вина из *UCI Wine Quality* скупа [2].

4.1 Имплементација

Све оцене изведене у глави 3 директно су израчунљиве. Оцена (3.4) захтева решавање једног линеарног система, тест из одељка 3.7 спектралну декомпозицију Грамове матрице S_T, а фино подешавање непосредну примену итерације из тврђења 3.2. Псеудокод сва три алгоритма дат је у прилогу Б, а комплетна имплементација у програмском језику *Python* доступна је у ауторовом репозиторијуму[1].

4.2 Симулација

Дизајн вештачких података

Узимамо $D = 8$ предиктора, $N_S = 5000$ изворних и свега $N_T = 40$ циљних опсервација. Врсте матрица X_S и X_T извлачимо независно, са $x_{ij} \sim \mathcal{N}(0, s_j^2)$ и опадајућим скалама $(s_1, \ldots, s_8) = (3,\ 2.2,\ 1.6,\ 1.2,\ 0.8,\ 0.5,\ 0.3,\ 0.15)$, па последњи предиктори у малом циљном узорку носе врло мало информације. Пошто су предиктори генерисани независно, елементи матрице изван главне дијагонале

[1] `https://github.com/urosm561/transfer-learning-linear-regression/`

S_T су занемарљиво мали, па је она приближно дијагонална и њени сопствени вектори су приближно координатне осе, са сопственим вредностима $d_j \approx N_T s_j^2$. Захваљујући томе, у овом одељку о j-тој координати коефицијената можемо да говоримо као о j-том сопственом правцу (на реалним подацима, где су предиктори корелисани, то више не важи). У коришћеној реализацији сопствене вредности S_T-а износе $(d_1, \ldots, d_8) \approx (358.1,\ 152.4,\ 125.7,\ 76.8,\ 24.1,\ 9.1,\ 2.2,\ 0.6)$, са односом највеће и најмање око 600.

Прави коефицијенти су $\beta_S = (1.5,\ -1,\ 0.8,\ 1.2,\ -0.6,\ 0.5,\ 1,\ -0.8)$ и $\beta_T = \beta_S + \delta$, где је разлика међу доменима ретка: $\delta_2 = 0.8$ (у добро опаженом правцу) и $\delta_6 = 0.5$ (у умерено опаженом), а остале координате су једнаке. Шум је Гаусов са $\sigma_S = \sigma_T = 1$. Како је $\|\delta\|_2^2 = 0.89$ мање од $\mathrm{MSE}(\hat{\beta}_T) = \sigma_T^2 \operatorname{tr}(S_T^{-1}) \approx 2.443$, извор је ближи циљној истини него што мали циљни узорак сам може да досегне, што је баш режим у коме пренос треба да се исплати. Генерисани подаци сачувани су као табеле (`synthetic_source.csv`, `synthetic_target_train.csv`, `synthetic_target_test.csv`, уз праве параметре у `synthetic_meta.json`), а на овој реализацији је $\|\hat{\beta}_S - \beta_S\|_2^2 \approx 0.023$ (што оправдава третирање извора као детерминистичког) и $\|\hat{\beta}_T - \beta_T\|_2^2 \approx 1.36$.

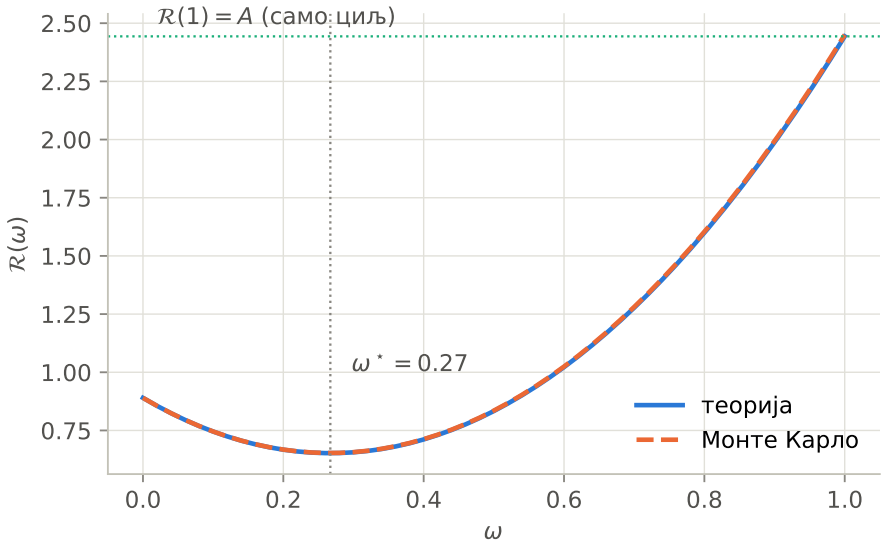
Конвексна комбинација

Са $A = \mathrm{MSE}(\hat{\beta}_T) \approx 2.443$ и $C = \|\delta\|_2^2 = 0.890$, формула (2.9) даје $\omega^\star \approx 0.267$ и $\mathcal{R}(\omega^\star) \approx 0.652$. Слика 4.1 пореди теоријску криву (2.8) са Монте Карло оценом уз помоћ 3000 понављања. Криве се поклапају, а већ и овај најгрубљи облик преноса постиже отприлике четвртину грешке чисте циљне оцене.

Избор параметра λ и одговарајућа оцена

Слика 4.2 приказује средњу квадратну грешку оцене $\hat{\beta}_\lambda$ у зависности од λ. Видимо да се теоријска декомпозиција (3.13) и Монте Карло оцена поклапају. Крива опада у $\lambda = 0$, у складу са (3.14), минимум је у $\lambda^\star \approx 10.6$ са грешком ≈ 0.169, смањење од 93% у односу на $\mathrm{MSE}(\hat{\beta}_T)$ и знатно боље и од оптималне скаларне комбинације (0.652). За $\lambda \to \infty$ грешка тежи ка $\|\delta\|_2^2$, то јест наслеђујемо пристрасност извора.

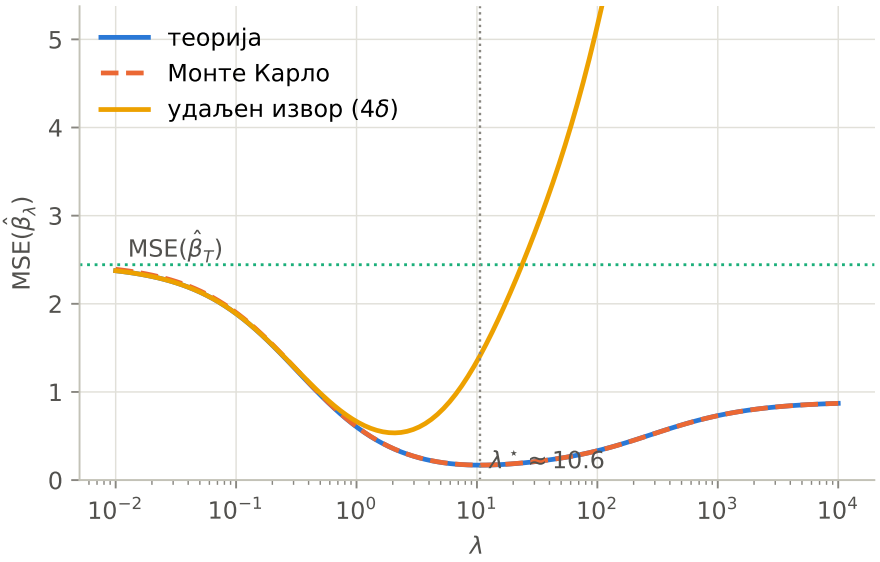
Иста слика приказује и криву за удаљен извор, са разликом 4δ. Ту за $\lambda \gtrsim 24$ грешка премашује $\mathrm{MSE}(\hat{\beta}_T)$, што је негативан пренос из одељка 3.4. Избор λ стога није безазлен, што мотивише унакрсну валидацију и статистички тест.



Слика 4.1: Ризик конвексне комбинације $\hat{\beta}(\omega)$: теорија (2.8) и Монте Карло оцена (3000 понављања). Тачкаста линија је ризик чисте циљне оцене.

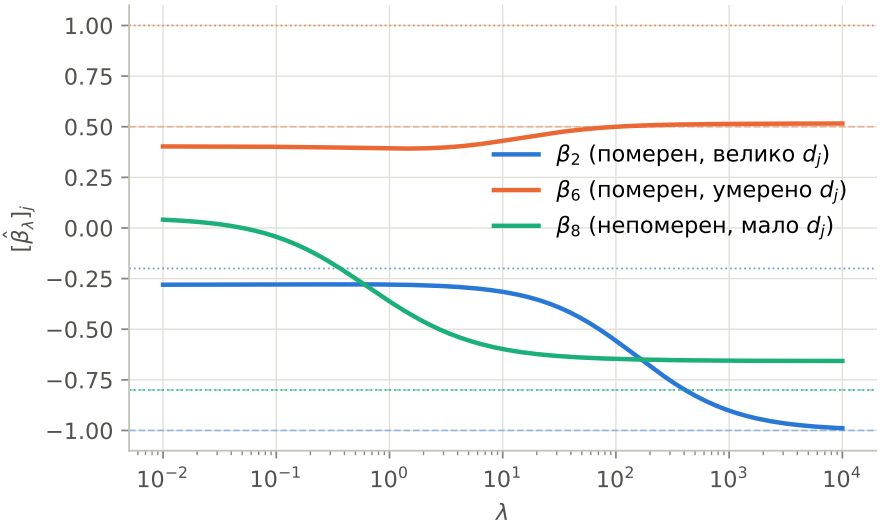
Путање коефицијената

Слика 4.3 прати изабране координате оцене $\hat{\beta}_\lambda$ док λ расте од 0 (чиста циљна оцена) ка ∞ (чист изворни), у складу са анализом из одељка 3.3. Коефицијент



Слика 4.2: MSE($\hat{\beta}_\lambda$) у зависности од λ: теорија (3.13), Монте Карло оцена и крива за четири пута удаљенији извор. Тачкаста линија је MSE($\hat{\beta}_T$).

β_2, различит међу доменима и у правцу велике сопствене вредности ($d \approx 152$), остаје на циљној вредности на целом практично релевантном опсегу λ: подаци су ту довољно богати да одбране промену. Коефицијент β_8 једнак је у оба домена али налази се у најслабијем правцу ($d \approx 0.6$), цильна оцена оцењује бесмислено ($+0.05$ уместо -0.8), а већ умерено λ повлачи га на изворну вредност, то јест, пренос поправља баш правац у коме цильни узорак не зна ништа. Коефицијент β_6 који се налази у правцу који је умерено заступљен, и различит је међу доменима, интерполира између ова два режима. Дакле, разлика $\hat{\beta}_\lambda - \hat{\beta}_S$ концентрише се на координатама које су се заиста промениле, под условом да је цильни узорак у њима информативан.



Слика 4.3: Путање $\lambda \mapsto [\hat{\beta}_\lambda]_j$ за три карактеристичне координате. Тачкасте линије су праве цильне вредности, испрекидане изворне.

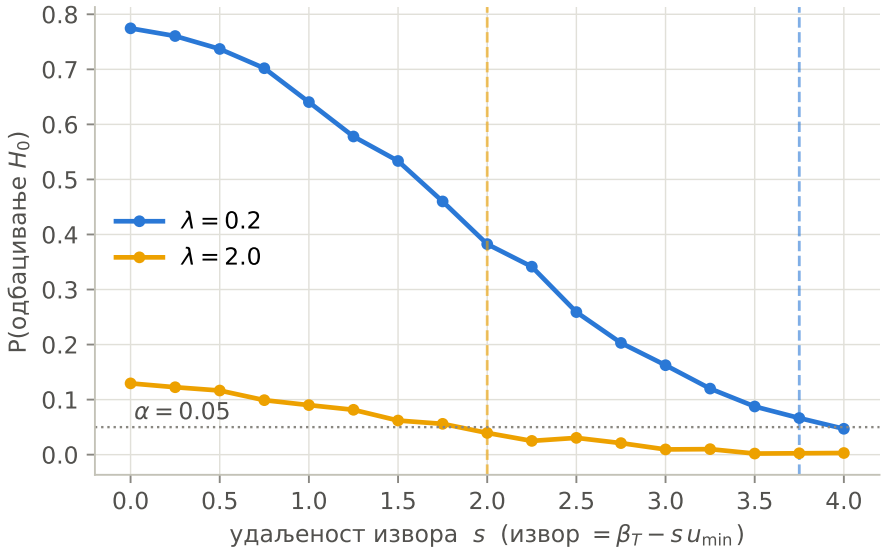
Тест корисности преноса

Тест из одељка 3.7 најпре примењујемо на генерисани узорак, са изворном оценом $\hat{\beta}_S$ у улози правих коефицијената:

λ	T	k	$F_{0.05}(k, N_T - D)$	p
0.1	0.028	7.31	0.306	$< 10^{-4}$
0.2	0.049	4.42	0.196	0.003
0.5	0.094	2.75	0.100	0.046
10.6	0.618	2.91	0.111	0.396

За мале вредности λ тест одбацује H_0 и потврђује корисност преноса, у $\lambda^\star$ не одбацује, иако пренос ту највише помаже, јер константа s_λ из калибрације у најнеповољнијој тачки расте са λ. У пракси то сугерише два корака: тест са малим λ служи као потврда да је пренос уопште користан, а λ које ћемо користити се бира унакрсном валидацијом.

Ниво и моћ испитујемо удаљавањем извора дуж најслабијег сопственог правца $u_{\min}$, са $\beta_S = \beta_T - s\,u_{\min}$, $s \in [0, 4]$. Граница H_0 се тада рачуна егзактно из (3.13). Слика 4.4 приказује учесталост одбацивања (2000 понављања по тачки) за $\lambda = 0.2$ и $\lambda = 2$: на граници H_0 и иза ње учесталост остаје на нивоу $\alpha = 0.05$ или испод (тест је ваљан, унутар H_0 конзервативан), а унутар H_1 моћ расте до 0.77 за $\lambda = 0.2$, док за веће λ конзервативност сабија целу криву.

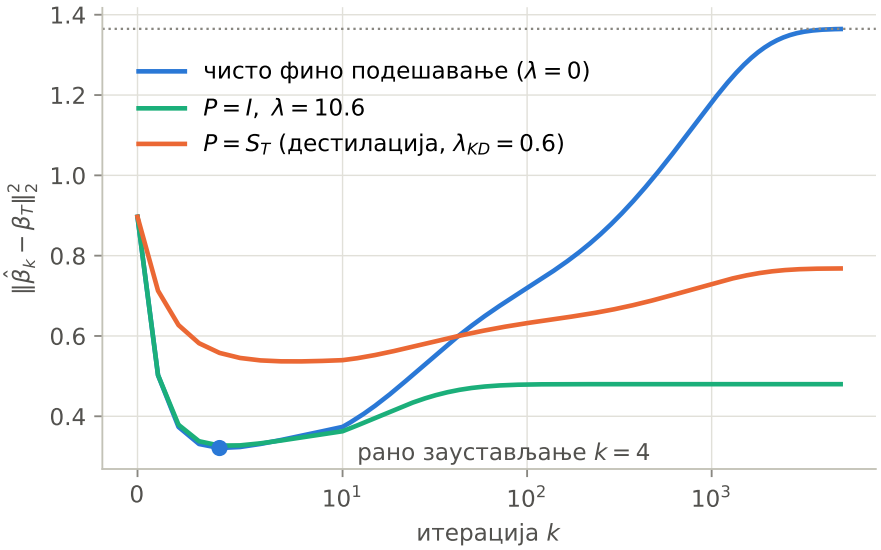


Слика 4.4: Учесталост одбацивања H_0 у зависности од удаљености извора дуж најслабијег правца (2000 понављања по тачки). Испрекидане вертикале су границе H_0; лево од њих пренос заиста помаже.

Фино подешавање и дестилација

Сва три члана фамилије (3.16) покрећемо из $\hat{\beta}_S$, са кораком $\alpha = 1/d_{\max}$, а слика 4.5 приказује грешку коефицијената дуж итерација. Код чистог финог подешавања ($\lambda = 0$) пренос се остварује искључиво кроз рано заустављање. Грешка најпре стрмо опада, достиже минимум 0.32 у $k = 4$, а затим расте ка грешци чисте циљне оцене (1.36), при чему је оптимални тренутак заустављања

оштар и зависи од података. Регуларизоване варијанте немају туману, јер за $P = I$ и $\lambda = \lambda^\star$ итерације конвергирају управо ка $\hat{\beta}_\lambda$ (грешка 0.480, у складу са последицом 3.1), а за $P = S_T$ са $\lambda_{\mathrm{KD}} = 0.6$ ка конвексној комбинацији $(1 - \lambda_{\mathrm{KD}})\hat{\beta}_T + \lambda_{\mathrm{KD}}\hat{\beta}_S$ (грешка 0.768, последица 3.2 и одељак 3.6). Целу трајекторију сваке методе описује формула (3.17).



Слика 4.5: Грешка коефицијената дуж итерација градијентног спуста за три члана фамилије (3.16), све из иницијализације $\hat{\beta}_S$. Тачкаста линија је грешка чисте циљне оцене.

4.3 Реални подаци: хемијски састав и квалитет вина

Поставка

UCI Wine Quality скуп [2] садржи физичко-хемијска мерења португалских вина *vinho verde* и сензорну оцену квалитета (цео број од 3 до 8): 4898 белих и 1599 црвених вина, са једанаест предиктора (фиксна киселост, испарљива киселост, лимунска киселина, резидуални шећер, хлориди, слободни и укупни сумпор-диоксид, густина, pH, сулфати, алкохол). Бела вина играју улогу великог изворног домена, а из скупа црвених насумично извлачимо $N_T = 50$ за тренирање, а преосталих ≈ 1549 служи искључиво као тест скуп. Предикторе

стандардизујемо (рачунајући очекивање и стандардну девијацију узорка на црвеним винима, и трансформишемо оба скупа на исти начин - $z = (x - \mu)/\sigma$) и додајемо слободни члан ($D = 12$). Пошто је оцена квалитета цео број, претпоставка Гаусовог шума важи само приближно.

Предвиђање

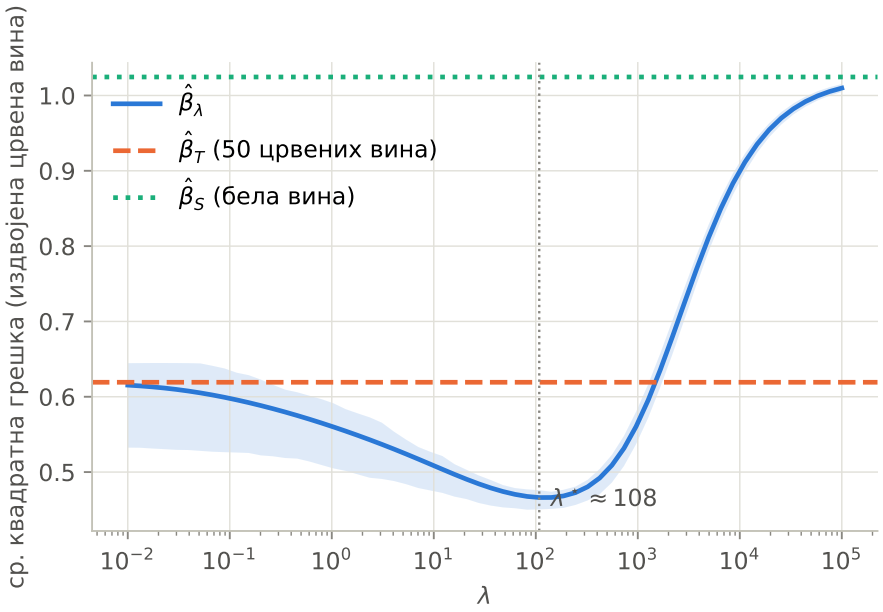
За сваку од 200 насумичних подела рачунамо $\hat{\beta}_T$, $\hat{\beta}_S$ и путању $\lambda \mapsto \hat{\beta}_\lambda$, и оцењујемо их средњом квадратном грешком на тест скупу. Слика 4.6 приказује просечну криву са интервалима поверења, а табела просеке преко свих подела:

оцена	средња квадратна грешка
$\hat{\beta}_T$ (само 50 црвених)	0.619
$\hat{\beta}_S$ (само бела)	1.025
$\hat{\beta}_\lambda$, λ унакрсном валидацијом	0.491
$\hat{\beta}_\lambda$, оптимално $\lambda^\star \approx 108$	0.466
МНК на свих 1599 црвених (репер)	0.424

Оптимална вредност $\lambda^\star$ је минимум тест грешке, у пракси недоступна и наведена само ради поређења, док је λ у претходном реду бирано петоструком унакрсном валидацијом искључиво на тренинг тачкама. Реперна вредност у добијена је десетоструком унакрсном валидацијом на свих 1599 црвених вина, тако да ниједна опсервација није коришћена истовремено за тренирање и оцену. Сам изворни модел је убедљиво најгори, али његово комбиновање са 50 циљних опсервација смањује грешку чистог циљног модела за око 21% (уз поштен избор λ валидацијом на самих 50 тренинг тачака), што је близу реперу који користи свих 1599 вина. За $\lambda \gtrsim 10^3$ наступа негативан пренос и оцена одлази ка $\hat{\beta}_S$.

Тест на реалним подацима

На реалним подацима права разлика међу доменима није позната, па не говоримо о моћи, већ о емпиријској учесталости одбацивања. За фиксирано $\lambda = 0.5$ тест одбацује H_0 (потврђује користан пренос) у 34% подела при $N_T = 50$, у 62% при $N_T = 100$ и у 86% при $N_T = 200$, и на фиксираној подели коришћеној у наставку одбацује са $p = 0.039$. За велике, предикционо оптималне вредности λ практично не одбацује ($\leq$ 3% подела при $\lambda = 20$), у складу са



Слика 4.6: Средња квадратна грешка на издвојеним црвеним винима у зависности од λ: просек и интерквартилни распон преко 200 насумичних подела ($N_T = 50$).

конзервативном калибрацијом. Иако је оцена квалитета целобројна (Гаусова претпоставка важи приближно) и однос највеће и најмање сопствене вредности S_T-а реда 10^3, препорука из симулације преноси се директно, тест са малим λ као потврда, а право λ се добија унакрсном валидацијом.

Интерпретација коефицијената

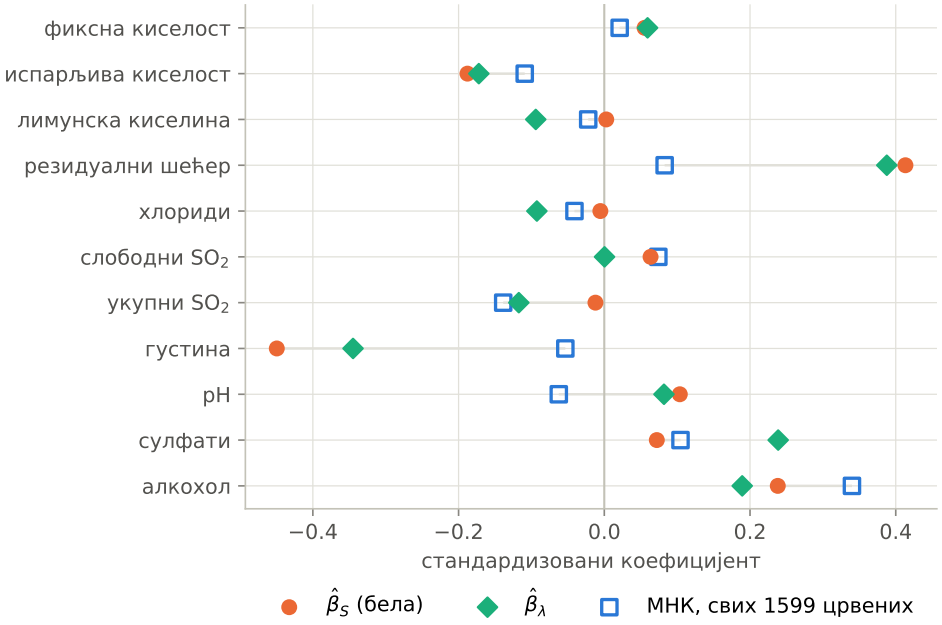
Табела 4.1 и слика 4.7 пореде, за једну фиксирану поделу, оцене $\hat{\beta}_S$, $\hat{\beta}_T$ и $\hat{\beta}_\lambda$ (за $\lambda = 20$, у равном делу криве предвиђања), уз оцену на свих 1599 црвених као спољашњу референцу.

Неколико образаца илуструје механизам из одељка 3.3:

- Коефицијент уз *укупни сумпор-диоксид* у изворном моделу је близу нуле (-0.012), а на свим црвенима јасно негативан (-0.139). Оцена ту промену хвата већ из 50 опсервација (-0.117), јер је правац довољно заступљен у циљном узорку. Коефицијент уз *сулфате* у циљном узорку је већи него у изворном (0.231 наспрам 0.072) и $\hat{\beta}_\lambda$ ту прати циљ (0.239), нешто изнад

Табела 4.1: Коефицијенти уз стандардизоване предикторе за једну фиксирану поделу ($N_T = 50$, $\lambda = 20$). Последња колона служи само као референца.

	$\hat{\beta}_S$ (бела)	$\hat{\beta}_T$ (50 црв.)	$\hat{\beta}_\lambda$	оцена за сва црвена
слободни члан	5.878	6.128	5.948	5.755
фиксна киселост	0.055	−0.109	0.059	0.021
испарљива киселост	−0.188	−0.265	−0.172	−0.109
лимунска киселина	0.003	−0.150	−0.094	−0.022
резидуални шећер	0.413	0.309	0.388	0.083
хлориди	−0.005	−0.122	−0.093	−0.041
слободни SO_2	0.063	−0.143	0.000	0.074
укупни SO_2	−0.012	−0.037	−0.117	−0.139
густина	−0.449	0.217	−0.345	−0.053
pH	0.104	−0.097	0.082	−0.062
сулфати	0.072	0.231	0.239	0.105
алкохол	0.238	0.350	0.189	0.340



Слика 4.7: Поређење коефицијената на реалним подацима: изворни модел, оцена $\hat{\beta}_\lambda$ и референтна оцена на свим црвеним винима.

вредности на свим црвенима (0.105), што је цена ослањања на мали узорак у добро заступљеном правцу.

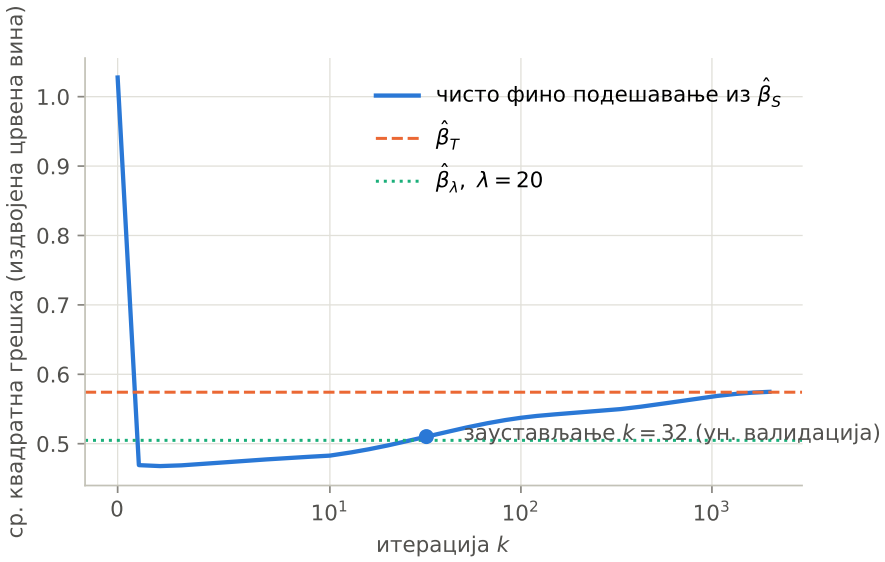
- Коефицијенти уз *лимунску киселину* и *pH* на црвеним винима се разликују од оних на белим, али су мали, па се $\hat{\beta}_\lambda$ помера само делимично. Са

50 опсервација нема основа да се одбрани пуна промена знака.

- *Резидуални шећер* и *густина* заиста се разликују међу доменима, али су међусобно јако корелисани и међу црвеним винима слабо варирају (мале сопствене вредности у S_T), па $\hat{\beta}_\lambda$ остаје близу изворних вредности, јер, оно што циљни узорак не види, пренос не може да поправи.
- $[\hat{\beta}_\lambda]_j$ не мора лежати између $[\hat{\beta}_S]_j$ и $[\hat{\beta}_T]_j$ (нпр. укупни сумпор-диоксид): комбинација (3.2) је матрична, па се код корелисаних предиктора мешање дешава у сопственој бази, а не координата по координата.

Фино подешавање на реалним подацима

Слика 4.8 приказује чисто фино подешавање из $\hat{\beta}_S$ на фиксираној подели. Тренутак заустављања биран је петоструком унакрсном валидацијом на 50 тренинг вина. Добијено $k = 32$ даје тест грешку 0.510, бољу од $\hat{\beta}_T$ (0.574) и упоредиву са $\hat{\beta}_\lambda$ (0.505). Минимум тест криве је већ у $k = 2$ са грешком 0.468, али је она оптимална вредност, у пракси недоступна, и управо та оштрина оптимума показује зашто је регуларизована оцена $\hat{\beta}_\lambda$, која исту околину достиже асимптотски, лакша за контролу од раног заустављања.



Слика 4.8: Чисто фино подешавање на подацима о винима: средња квадратна грешка на тест скупу дуж итерација, из иницијализације $\hat{\beta}_S$. Означен је тренутак заустављања изабран унакрсном валидацијом.

Укратко, и на вештачким и на реалним подацима оцена добијена преносом понашала се онако како теорија прописује. Смањила је грешку предвиђања малог модела за око петину на реалним подацима (и за ред величине у повољном симулационом режиму), разлика $\hat{\beta}_\lambda - \hat{\beta}_S$ указала је на смислене разлике између црвеног и белог вина, а начини отказивања (негативан пренос за превелико λ, слепило у правцима које циљни узорак не покрива) управо су они које теорија предвиђа и од којих штите тест и унакрсна валидација.

Глава 5

Закључак

У овом раду разматрали смо пренос наученог знања у линеарној регресији кроз оцену добијену генерализацијом гребене регресије [4, 1]. Показали смо да ова оцена има два природна тумачења: као решење регуларизационог оптимизационог проблема (3.3), и као очекивање апостериорне расподеле при бајесовском тумачењу [6], где регуларизациони параметар λ добија природно значење као однос дисперзије шума и априорне дисперзије.

Централни аналитички алат био нам је свођење свих величина на сопствену базу Грамове матрице. У тој бази, оцена се читаво време понаша координата по координата: у правцима у којима имамо довољно циљних података (d_j веће од λ) практично одбацује извор и користи циљна оцена, а у правцима у којима немамо довољно циљних података (d_j мање од λ) практично одбацује циљ и користи извор. Разлика $\hat{\beta}_\lambda - \hat{\beta}_S$ директно указује на правце у којима је циљни домен различит од изворног, што смо на реалном скупу података о хемијском саставу и квалитету вина илустровали кроз ефекте *сумпор-диоксида*, *киселина* и pH вредности.

Извели смо и експлицитну декомпозицију средње квадратне грешке (3.13), из које следе тврдње: (i) за било које довољно мало $\lambda > 0$, пренос строго смањује грешку у односу на циљну оцену (израз (3.14)); (ii) за велико λ, ако је извор далеко од циља, наслеђујемо његову пристрасност и грешка може постати већа него код циљне оцене, односно добијамо негативан пренос. Ове две тврдње у пракси значе да λ морамо пажљиво да подесимо (валидацијски или преко тест статистике), а не фиксирамо унапред, за шта смо извели тест статистику засновану на Фишеровој расподели.

Поред затвореног облика, размотрили смо и фино подешавање помоћу гра-

дијентног спуста (одељак 3.5) [7]. Ова фамилија оцена у себи обједињује три практична приступа: фино подешавање са изворном иницијализацијом ($\lambda = 0$), гребену регресију и дестилацију знања. Заједничка тачка јесте итерација (3.17): у првим итерацијама оцена је близу извора, а са растом k конвергира ка асимптоти која зависи од избора референце и јачине регуларизације. За чисто фино подешавање, асимптота је циљна оцена, што значи да се пренос остварује искључиво кроз рано заустављање; за два регуларизована случаја, асимптота сачува удео извора без обзира на број итерација.

5.1 Ограничења

Оцена (3.4) претпоставља да су изворни и циљни модели у истом простору коефицијената, тј. да имају исте предикторе. Ово је једноставна и најчешћа претпоставка, али и она која се не мора увек испунити у пракси, на пример, циљни домен може имати додатне предикторе које извор није користио. У том случају би било могуће, али нетривијално, проширити оцену тако да користи *пројекцију* изворног модела на заједнички простор предиктора.

Такође, теорија је развијена под претпоставком Гаусовског шума. Тест статистика из одељка 3.7 се ослања на ову претпоставку, а за не-Гаусовски шум, тест је асимптотски исправан, али за мање узорке морамо да будемо опрезнији.

5.2 Правци даљих истраживања

Природних наставака рада има више. Најпре, уопштење на случај у ком располажемо са више извора, свакога уз своју сличност са циљем, води ка тежинској суми $\sum_k \lambda_k \|\beta - \hat{\beta}_S^{(k)}\|_2^2$, али оставља и питање како одабрати вектор λ уместо скалара. Наравно, свуда где смо користили неку функцију губитка коју смо оптимизовали можемо поставити и питање како је генерализовати. Коначно, могуће је и дискутовати о преласку на компликованије моделе, где би прелазак на генерализоване линеарне моделе био изводљив уз мање измена [10], али би прелаз на егзотичније моделе захтевао много више пажње.

Овим правцима, као и практичним применама у ситуацијама у којима интерпретабилност матичног модела има реалну вредност (медицина, енергетика, финансије), рад отвара више питања него што их затвара, што је, надам се, најбоља ствар коју један мастер рад може да оствари.

Додатак А

Сатервајтова апроксимација

Нека је

$$Y = \sum_j w_j (Z_j + m_j)^2, \qquad Z_j \overset{\text{изв.}}{\sim} \mathcal{N}(0,1), \quad w_j > 0, \quad m_j \in \mathbb{R}.$$

Циљ нам је изразити Y као $c \cdot \chi^2(\nu)$ избором c, ν тако да се прва два момента поклапају.

За $X = (Z+m)^2$, $Z \sim \mathcal{N}(0,1)$:

$$\mathbb{E}[X] = \mathbb{E}[Z^2] + 2m\mathbb{E}[Z] + m^2 = 1 + m^2.$$

Развојем $X^2 = (Z+m)^4 = Z^4 + 4mZ^3 + 6m^2Z^2 + 4m^3Z + m^4$ и коришћењем $\mathbb{E}[Z] = \mathbb{E}[Z^3] = 0$, $\mathbb{E}[Z^2] = 1$, $\mathbb{E}[Z^4] = 3$ следи

$$\mathbb{E}[X^2] = 3 + 6m^2 + m^4,$$

$$D(X) = \mathbb{E}[X^2] - (\mathbb{E}[X])^2 = 3 + 6m^2 + m^4 - (1+m^2)^2 = 2 + 4m^2 = 2(1+2m^2).$$

Из независности Z_j:

$$\mathbb{E}[Y] = \sum_j w_j (1 + m_j^2),$$

$$D(Y) = \sum_j w_j^2 \cdot 2(1+2m_j^2) = 2\sum_j w_j^2 (1+2m_j^2).$$

За $W \sim c \cdot \chi^2(\nu)$ важи $\mathbb{E}[W] = c\nu$, $D(W) = 2c^2\nu$. Дакле, потребно нам је

$$c\nu = \mathbb{E}[Y], \qquad 2c^2\nu = D(Y),$$

односно, дељењем једначина

$$c = \frac{D(Y)}{2\mathbb{E}[Y]}, \qquad \nu = \frac{2(\mathbb{E}[Y])^2}{D(Y)}.$$

У нашем тесту, $w_j = (\lambda/(d_j+\lambda))^2$, $m_j = \sqrt{d_j}\,\tilde{\delta}_j/\sigma_T$. Како посматрамо само најмање могуће b_λ, Из везе $\sum_j w_j\tilde{\delta}_j^2 = \sigma_T^2 c_\lambda$ следи

$$w_{j_{\min}}\tilde{\delta}^{\star 2}_{j_{\min}} = \sigma_T^2 c_\lambda \implies \tilde{\delta}^{\star 2}_{j_{\min}} = \tfrac{\sigma_T^2 c_\lambda (d_{\min}+\lambda)^2}{\lambda^2},$$

а померање у правцу $j_{\min}$ добијамо са

$$m^{\star 2} := m^{\star 2}_{j_{\min}} = \tfrac{d_{\min}\tilde{\delta}^{\star 2}_{j_{\min}}}{\sigma_T^2} = \tfrac{d_{\min} c_\lambda (d_{\min}+\lambda)^2}{\lambda^2}, \qquad m_j^\star = 0 \;\text{ за }\; j \neq j_{\min}.$$

$$\mathbb{E}[Y]\big|_\star = a_\lambda + w_{j_{\min}} m^{\star 2},$$

где је $a_\lambda = \sum_j w_j$. Даље директно добијамо

$$w_{j_{\min}} m^{\star 2} = \tfrac{\lambda^2}{(d_{\min}+\lambda)^2} \cdot \tfrac{d_{\min} c_\lambda (d_{\min}+\lambda)^2}{\lambda^2} = d_{\min} c_\lambda,$$

па је $\mathbb{E}[Y]\big|_\star = a_\lambda + d_{\min} c_\lambda = s_\lambda$, што смо и очекивали.

Коначно, следи:

$$\mathrm{D}(Y)\big|_\star = 2\sum_j w_j^2 + 4 w^2_{j_{\min}} m^{\star 2}.$$

$$k = \tfrac{s_\lambda^2}{\sum_j w_j^2 + 2 w^2_{j_{\min}} m^{\star 2}}, \qquad w_j = \left(\tfrac{\lambda}{d_j+\lambda}\right)^2, \qquad m^{\star 2} = \tfrac{d_{\min} c_\lambda (d_{\min}+\lambda)^2}{\lambda^2}.$$

Напоменимо да је ова апроксимација боља што је више сабирака у оригиналној суми, али да такође није добра за мали број, и тада је није паметно користити.

Додатак Б

Псеудокод алгоритама

У овом прилогу дајемо псеудокод три поступка коришћена у глави 4: оцењивача преноса у затвореном облику, статистичког теста корисности преноса и финог подешавања градијентним спустом.

Алгоритам 1 Оцењивач преноса $\hat{\beta}_\lambda$

Улаз: $X_T \in \mathbb{R}^{N_T \times D}$, $\mathbf{Y}_T \in \mathbb{R}^{N_T}$, $\hat{\beta}_S \in \mathbb{R}^D$, $\lambda \geq 0$
Излаз: $\hat{\beta}_\lambda$

$S_T \leftarrow X_T^\top X_T$
реши систем $(S_T + \lambda I)\,\beta = X_T^\top \mathbf{Y}_T + \lambda \hat{\beta}_S$ по β ▷ једначина (3.4)
return β

Алгоритам 2 Фино подешавање градијентним спустом

Улаз: X_T, $\mathbf{Y}_T$, иницијализација $\hat{\beta}_S$, параметри $\lambda \geq 0$, $P \succeq 0$, $\beta_0 \in \mathbb{R}^D$, корак α, број итерација K
Излаз: путања $\beta^{(0)}, \ldots, \beta^{(K)}$

$M \leftarrow S_T + \lambda P$; $\quad g \leftarrow X_T^\top \mathbf{Y}_T + \lambda P \beta_0$
$\beta^{(0)} \leftarrow \hat{\beta}_S$
for $k \leftarrow 1, \ldots, K$ **do**
    $\beta^{(k)} \leftarrow \beta^{(k-1)} - \alpha \left(M\beta^{(k-1)} - g \right)$ ▷ тврђење 3.2
end for
return $\beta^{(0)}, \ldots, \beta^{(K)}$

Алгоритам 3 Тест корисности преноса

Улаз: X_T, $\mathbf{Y}_T$, $\hat{\beta}_S$, $\lambda > 0$, ниво значајности α
Излаз: одлука: одбацити H_0 (пренос је користан) или не

$S_T \leftarrow X_T^\top X_T$; израчунај сопствене вредности $d_1, \ldots, d_D$ матрице S_T
$\hat{\beta}_T \leftarrow S_T^{-1} X_T^\top \mathbf{Y}_T$; $\quad \hat{\beta}_\lambda \leftarrow$ алгоритам 1$(X_T, \mathbf{Y}_T, \hat{\beta}_S, \lambda)$
$\mathrm{RSS}_T \leftarrow \|\mathbf{Y}_T - X_T\hat{\beta}_T\|_2^2$; $\quad \mathrm{RSS}_\lambda \leftarrow \|\mathbf{Y}_T - X_T\hat{\beta}_\lambda\|_2^2$
$\hat{\sigma}_T^2 \leftarrow \mathrm{RSS}_T/(N_T - D)$
for $j \leftarrow 1, \ldots, D$ **do**
    $w_j \leftarrow \left(\dfrac{\lambda}{d_j + \lambda}\right)^2$
end for
$a_\lambda \leftarrow \displaystyle\sum_{j=1}^{D} w_j$; $\quad c_\lambda \leftarrow \sum_{j=1}^{D} \dfrac{\lambda(2d_j + \lambda)}{d_j(d_j + \lambda)^2}$
$j_{\min} \leftarrow \arg\min_j d_j$; $\quad d_{\min} \leftarrow d_{j_{\min}}$
$s_\lambda \leftarrow a_\lambda + d_{\min} c_\lambda$ ▷ инфимум очекивања под H_0
$m^{\star 2} \leftarrow \dfrac{d_{\min} c_\lambda (d_{\min} + \lambda)^2}{\lambda^2}$ ▷ најнеповољнија тачка, поглавље A
$k \leftarrow \dfrac{s_\lambda^2}{\sum_j w_j^2 + 2w_{j_{\min}}^2 m^{\star 2}}$
$T \leftarrow \dfrac{\mathrm{RSS}_\lambda - \mathrm{RSS}_T}{\hat{\sigma}_T^2 \, s_\lambda}$
if $T < F_\alpha(k, N_T - D)$ **then**
    return одбаци H_0
else
    return не одбацуј H_0
end if

Библиографија

[1] Aiyou Chen, Art B. Owen, and Minghui Shi. Data enriched linear regression. *Electronic Journal of Statistics*, 9(1):1078–1112, 2015.

[2] Paulo Cortez, António Cerdeira, Fernando Almeida, Telmo Matos, and José Reis. Modeling wine preferences by data mining from physicochemical properties. *Decision Support Systems*, 47(4):547–553, 2009.

[3] Geoffrey Hinton, Oriol Vinyals, and Jeff Dean. Distilling the knowledge in a neural network. *arXiv preprint arXiv:1503.02531*, 2015.

[4] Arthur E. Hoerl and Robert W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12(1):55–67, 1970.

[5] Roger A. Horn and Charles R. Johnson. *Matrix Analysis*. Cambridge University Press, Cambridge, 2 edition, 2013.

[6] Ilja Kuzborskij and Francesco Orabona. Stability and hypothesis transfer learning. In *Proceedings of the 30th International Conference on Machine Learning*, volume 28 of *Proceedings of Machine Learning Research*, pages 942–950. PMLR, 2013.

[7] David Obst, Bernard Ghattas, Sandra Claudel, Jairo Cugliari, Yannig Goude, and Gilles Oppenheim. Improved linear regression prediction by transfer learning. *Computational Statistics & Data Analysis*, 174:107499, 2022.

[8] Sinno Jialin Pan and Qiang Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22(10):1345–1359, 2010.

[9] Franklin E. Satterthwaite. Synthesis of variance. *Psychometrika*, 6:309–316, 1941.

[10] Ye Tian and Yang Feng. Transfer learning under high-dimensional generalized linear models. *Journal of the American Statistical Association*, 118(544):2684–2697, 2023.

Биографија аутора

Урош Миленковић је рођен у Лесковцу 17. јула 1999. године. Након завршене основне школе у родном граду, сели се у Београд, где похађа менторско одељење Математичке гимназије. Потом уписује Математички факултет, који завршава 2022. године са просеком 9.92. Одмах по дипломирању, оснива (заједно са пријатељима) стартап SignAvatar, у коме остаје све до његовог (не)славног завршетка при крају 2025. године. Тренутно је запослен у компанији Perplexity AI, где ради као инжењер машинског учења у области претраживања интернета.