УДК 004.852+004.42

БУСТІНГ І БЕГГІНГ ЯК МЕТОДИ ФОРМУВАННЯ АНСАМБЛЕЙ МОДЕЛЕЙ

Вербівський Дмитрій Сергійович
Карплюк Світлана Олександрівна
Фонарюк Олена Василівна
Сікора Ярослава Богданівна
к.п.н., доцент
Житомирський державний університет
імені Івана Франка
м.Житомир, Україна

Вступ. Дослідження методів інтелектуального аналізу даних (англ. *data mining*) свідчить, що жоден з них не є ідеальним (універсальним) – і саме тому їх з'явилося так багато; дослідники даного напрямку не один рік працюють над пошуком компромісу між точністю, простотою і інтерпретованістю кожної окремої моделі. Однак слід зазначити, що більшість експертів віддає перевагу точності, зазначаючи, що саме це якість і робить модель корисною.

Постає питання: як підвищити точність моделі, не змінюючи її суті? Одним із способів підвищення точності моделей є створення і навчання ансамблів моделей – тобто наборів моделей, що використовуються для вирішення однієї і тієї ж задачі. Під навчанням ансамблю розуміють навчання кінцевого набору базових класифікаторів з наступним об'єднанням результатів їх прогнозування в єдиний прогноз агрегованого класифікатора. Зрозуміло, що об'єднаний (агрегований) класифікатор дасть більш точний результат, особливо якщо:

– кожен з базових класифікаторів сам по собі має непогану точність;

– вони призводять до різних результатів (помиляються на різних множинах).

Експерти виділяють наступні причини доцільності об'єднання моделей (класифікаторів) в ансамбль:

Статистична. Як вже зазначалося, агрегований класифікатор «усереднює» помилку кожного з базових класифікаторів – відповідно, вплив випадковостей на усереднену гіпотезу істотно зменшується.

Обчислювальна. Аби не заглиблюватися в розлогі математичні пояснення, наведемо приклад з реального життя: припустимо, що на якійсь обмеженій території заритий скарб, який необхідно знайти, врахувавши наступні вимоги: рельєф і рослинність цієї місцевості, обмеженість в часі та погодні умови. У нашому випадку скарб – це глобальний оптимум, і ансамбль моделей має більший шанс знайти його, оскільки буде шукати його з різних точок вихідної множини гіпотез (з різних точок території, вказаної в прикладі).

Репрезентативна. Можливий випадок, коли агрегована гіпотеза буде знаходиться за межами множини гіпотез базових – в такому випадку при побудові комбінованої гіпотези ми розширимо множину можливих гіпотез.

Побудувати ансамбль моделей можна з використанням декількох методів, але головними є беггінг і бустінг.

Мета роботи. Розглянути ансамблі моделей та поліпшення прогнозів, що забезпечують необхідну різноманітність та ефективність моделей методами їх формування – бустінг і беггінг.

Матеріали і методи. Побудова та дослідження регресійних моделей різного типу реалізоване в програмному середовищі для статистичних обчислень, аналізу та зображення даних в графічному вигляді R, з використанням додаткових пакетів.

Результати та обговорення. В статистиці добре відомим є інтуїтивне міркування, згідно якого усереднення результатів спостережень може дати більш стійку надійну оцінку, оскільки послаблюється вплив випадкових флуктуацій в окремому вимірі. На аналогічній ідеї було засноване розвиток алгоритмів комбінування моделей, в результаті чого побудова їх ансамблів виявилося одним з найбільш потужних методів машинного навчання, нерідко перевершує за якістю прогнозів інші методи.

Одним з рішень, що забезпечують необхідну різноманітність моделей, є

їх повторне навчання на вибірках, випадково вибраних з генеральної сукупності, або інших підмножин даних, сконструйованих з уже наявних. Для отримання стійкого прогнозу часткові передбачення цих моделей тим чи іншим чином комбінують, наприклад, за допомогою простого усереднення або голосування (можливо, зваженого).

Розглянемо таке поняття як бутстреп – процедура генерації повторних випадкових вибірок з вихідного набору даних. Бутстреп-вибірки виробляються рівномірно і з поверненням, тому деякі вихідні приклади будуть відсутні, а інші – дублюватися: в середньому одна така вибірка містить близько 2/3 унікальних вихідних спостережень. Бутстреп при формуванні ансамблю моделей є особливо корисний у поєднанні з деревоподібними структурами, які дуже чутливі до невеликої зміни навчальних даних.

Подібно до того, як усереднення декількох спостережень знижує оцінку дисперсії даних, так і розумним способом зниження дисперсії прогнозує отримання великої кількості порцій даних з генеральної сукупності, побудова передбачуваної моделі по кожній навчальній вибірці і усереднення отриманих прогнозів. Якщо замість окремих навчальних вибірок (яких нам, як правило, завжди не вистачає) виконати бутстреп і на основі згенерованих псевдо вибірок побудувати B дерев регресії, то середній колективний прогноз

$f_{bag}(x) = \{f^1(x) + f^2(x) + \ldots + f^B(x)\}/B$ буде мати більш низьку дисперсією. Ця процедура і є беггінгом. Беггінг можна проводити не тільки по відношенню до дерев регресії, а й інших моделей: опорних векторів, нейронних мереж, лінійних дискримінантів, байєсовских ймовірностей тощо.

Метод випадкового лісу (*Random Forest*) являє собою подальше поліпшення беггінгу дерев рішень, яке полягає в усуненні кореляції між деревами. Як і у випадку з беггінгом, ми будуємо кілька сотень дерев рішень по навчальних бутстреп-вибірках. Однак на кожній ітерації побудови дерева випадковим чином вибирається m з p предикторів, що підлягають розгляду і розбиття дозволяється виконувати тільки по одному з цих m змінних.

Сенс цієї процедури, яка є досить ефективною для підвищення якості

отримуваних рішень, полягає в тому, що з імовірністю (*p* - *m)/p* блокується який-небудь потенційно домінуючий предиктор, який прагне увійти в кожне дерево. Якщо домінування таких предикторів дозволити, то всі дерева в результаті будуть дуже схожі один на одного, а одержувані на їх основі передбачення будуть сильно корелювати і зниження дисперсії буде не настільки очевидним. Завдяки блокуванню домінантів, інші предиктори отримають свій шанс, і варіація дерев зростає.

Вибір малого значення *m* при побудові випадкового лісу зазвичай буде корисним при наявності великого числа корелюючих предикторів. Природньо, якщо випадковий ліс будується з використанням *m = p,* то вся процедура зводиться до простого беггінгу.

Беггінг (*bootstrap aggregating*) використовує паралельне навчання базових класифікаторів (якщо говорити мовою математичної логіки, то беггінг – поліпшує об'єднання). В ході беггінга відбувається наступне:

– з множини вихідних даних випадковим чином відбирається кілька підмножин, що містять кількість прикладів, що відповідає кількості прикладів вихідної множини;

– оскільки відбір здійснюється випадковим чином, то набір прикладів завжди буде різним: деякі приклади потраплять в кілька підмножин, а деякі не потраплять ні в одне;

– на основі кожної вибірки будується класифікатор;

– висновки класифікаторів агрегуються (шляхом голосування або усереднення).

Очікується, що результат прогнозу агрегованого класифікатора буде набагато точніший ніж результат прогнозу одиночної моделі на тому ж наборі даних.

Перш ніж говорити про бустінг, слід розглянути два поняття data mining – сильної і слабкої моделей. Сильної моделлю називається та модель, яка допускає мінімальну кількість помилок класифікації. Слабка ж модель, навпаки, допускає безліч помилок – тобто не є точною (або втрачає в

надійності).

При бустінгу відбувається послідовне навчання класифікаторів. Таким чином, навчальний набір даних на кожному наступному кроці залежить від точності прогнозування попереднього базового класифікатора. Перший алгоритм Boost1, наприклад, застосовував три базових класифікатора. При цьому перший класифікатор навчався на всьому наборі даних, другий на вибірці прикладів, а третій – на наборі тих даних, де результати прогнозування перших двох класифікаторів розійшлися. Сучасна модифікація першого алгоритму має на увазі використання необмеженої кількості класифікаторів, кожен з яких навчається на одному наборі прикладів, по черзі застосовуючи їх на різних етапах.

Бустінгом (від англ. *boosting* – посилення) називається метод, спрямований на перетворення слабких моделей в сильні шляхом побудови ансамблю класифікаторів. Ідея бустінгу полягає в ітеративному процесі послідовної побудови приватних моделей. Кожна нова модель навчається з використанням інформації про помилки, зроблених на попередньому етапі, а результуюча функція являє собою лінійну комбінацію всього ансамблю моделей з урахуванням мінімізації будь-якої штрафної функції. Бустінг є загальним підходом, який можна застосовувати до багатьох статистичних методів регресії і класифікації. Тут ми обмежимося обговоренням градієнтного бустінга в контексті дерев регресії.

Бутстреп-вибірки в ході реалізації бустінга не створюються, але замість цього кожне дерево будується по набору даних $\{X, r\}$, який на кожному кроці модифікується певним чином. На першій ітерації за значеннями вихідних предикторів будується дерево $f^1(x)$ і знаходиться вектор залишків r_1. На наступному етапі нове регресійне дерево $f^2(x)$ будується вже не за навчальним даними X, а по залишкам r_1 попередньої моделі. Лінійна комбінація прогнозу по побудованим деревам дає нам нові залишки $r \leftarrow r + \lambda\, f2(x)$, і цей ітераційний процес повторюється B разів. Завдяки побудові неглибоких дерев по залишкам, прогноз відгуку повільно поліпшується в областях, де одиночне дерево працює

не дуже добре. Такі дерева можуть бути досить невеликими, лише з кількома кінцевими вузлами, параметр стиснення регулює λ швидкість цього процесу, дозволяючи створювати комбінації дерев більш складної форми для "атаки" залишків. Підсумкова модель бустінгу: $f(x) = \sum_{b=1}^{B} \alpha f^b(x)$.

В середовищі R для побудови бустінг-моделей на основі дерев рішень можна використовувати функцію gbm() з пакета gbm (Generalized Boosted Models). Процес моделювання проходить під управлінням трьох гіперпараметрів:

1. Число дерев B (формальний параметр n.tree). На відміну від беггінгу, бустінг може, хоча і повільно, приводити до перенавчання при надмірно великому B.

2. Параметр стиснення λ (shrinkage), який коригує величину вкладу кожного додаткового дерева і контролює швидкість, з якою відбувається навчання моделі при реалізації бустінга. Типові значення варіюють від 0.01 до 0.001, і їх оптимальний вибір залежить від розв'язуваної проблеми. Для досягнення хорошої якості передбачень дуже низькі значення λ вимагають достатньо великого значення B.

3. Число внутрішніх вузлів d (interaction.depth) в кожному дереві, яке контролює складність одержуваного в результаті бустінгу ансамблю моделей. За своєю суттю, параметр d відображає глибину взаємодій між предикторами в підсумковій моделі. Якщо ці взаємодії не дуже виражені, то добре працює d=1, і тоді додаткові дерева являють собою просто "пеньки" (stump), тобто містять тільки один внутрішній вузол. В такому випадку одержуваний в результаті бустінгу ансамбль стає адитивною моделлю, оскільки кожен її член представлений тільки однією змінною.

Висновки. Точність моделі є найважливішим з її властивостей. Підвищити точність моделі можна за допомогою побудови і навчання ансамблів моделей – тобто набору моделей, що працюють над вирішенням однієї і тієї ж задачі. Два основних способи побудови ансамблів моделей,

бустінг і беггінг, дають значно більш точний результат, ніж застосування одиночної моделі на шуканому наборі даних. В свою чергу бутстреп дає хорошу можливість провести спеціальну процедуру перехресної перевірки, яка називається "тестом за спостереженнями, які не потрапили в сумку" (out-of-bag observations). Оскільки ключова ідея беггінга заключається в багаторазовій побудові моделей за спостереженнями з бутстреп-вибірок, то кожне конкретне дерево будується на основі приблизно двох третин усіх спостережень. Інша третина спостережень не використовується в навчанні, але цілком може бути використана для незалежного тестування: помилка на таких залишкових даних (out-of-bag error) є складовою оцінкою помилки на контрольній вибірці.

СПИСОК ЛІТЕРАТУРИ:

1. Borcard D., Gillet F., Legendre P. Numerical Ecology with R. N.Y.: Springer, 2011. 306 p.

2. Box G., Cox D.R. An analysis of Transformation // Journal of Royal Statistical Society B. 1964. V. 26. P. 211-243.

3. Cameron A.C, Trivedi P.K. Regression Analysis of Count Data. Cambridge University Press, Cambridge. 2013.

4. Chiu YW. Machine learning with R. Cookbook. Packt Publishing, 2015.

5. Maindonald J., Braun W.J. Data Analysis and Graphics Using R. Cambridge: University Press, 2010. 525 p.

6. Mount J., Zumel N. Exploring Data Science. Manning Publications Co., 2016. 184 p.