Лекция 19. Сравнение на хипотези и алгоритми за обучение

19.1. Сравнение на точността на хипотези

Да предположим, че имаме две хипотези h_1 и h_2 за някоя целочислена целева функция. Хипотезата h_1 е тествана върху извадката S_1, съдържаща n_1 случайно избрани примера, а хипотезата h_2 е тествана върху извадката S_2, съдържаща n_2 случайно избрани примера от същото разпределение. Искаме да сравним точността на тези хипотези, т.е. да преценим разликата d между техните истински грешки:

$$d \equiv error_D(h_1) - error_D(h_2)$$

Ще използваме общата 4-стъпкова процедура, описана в предишната лекция, за да изведем доверителния интервал за оценката на d. След като идентифицирахме d като параметър, който искаме да оценим, трябва да определим оценителя. Един естествен избор за такъв оценител в дадения случай е разликата между извадковите грешки, които ще означим с $\hat{d}$:

$$\hat{d} \equiv error_{S_1}(h_1) - error_{S_2}(h_2)$$

Може да се покаже, че $\hat{d}$ дава обективната оценка за d, т.е. че $E[\hat{d}] = d$.

Какво е вероятностното разпределение, на което се подчинява случайната величина $\hat{d}$? От предишната лекция знаем, че при големи n_1 и n_2 (т.е. когато те са ≥ 30), $error_{S_1}(h_1)$ и $error_{S_2}(h_2)$ ще се подчиняват на разпределенията, които са приблизително нормални. Тъй като разликата между две нормални разпределения е пак нормалното разпределение, $\hat{d}$ също ще бъде приблизително нормално разпределено със средната стойност d. Може също да бъде показано, че дисперсията на това разпределение е сумата на дисперсиите на $error_{S_1}(h_1)$ и $error_{S_2}(h_2)$. Използвайки формулата за дисперсията на извадковата грешка, изведена в предишната лекция, получаваме:

$$\sigma_{\hat{d}}^2 \approx \frac{error_{S_1}(h_1)(1-error_{S_1}(h_1))}{n_1} + \frac{error_{S_2}(h_2)(1-error_{S_2}(h_2))}{n_2} \quad (19.1)$$

Сега, след като сме определили вероятностното разпределение, на което се подчинява оценителят $\hat{d}$, можем да изведем доверителните интервали, характеризиращи вероятната грешка при използването на $\hat{d}$ за оценяването на d. За случайната величина $\hat{d}$, следваща нормалното разпределение със средното d и

дисперсията σ^2, N% доверителен интервал за оценяване на d е $\hat{d} \pm z_N \sigma$. Използвайки приближената стойност на дисперсията от ф-ла (19.1), приблизителният N% доверителен интервал за оценката на d се определя от следната формула:

$$\hat{d} \pm z_N \sqrt{\frac{error_{S_1}(h_1)(1-error_{S_1}(h_1))}{n_1} + \frac{error_{S_2}(h_2)(1-error_{S_2}(h_2))}{n_2}} \quad (19.2)$$

Полученият израз определя общия двустранен доверителен интервал за оценяването на разликите между грешките на две хипотези. Едностранният интервал може да бъде получен чрез модифициране на този израз по начина, описан в предишната лекция.

Макар, че в преведения по-горе анализ хипотезите h_1 и h_2 са били тествани върху независими извадки от данни, допустимо е използване на ф-ла (19.2) и в случаи, когато и двете хипотези се тестват върху една и съща извадка от данни S (като S остава независимо от h_1 и h_2). В този случай, трябва предефинираме $\hat{d}$ като:

$$\hat{d} \equiv error_S(h_1) - error_S(h_2)$$

Дисперсията на това $\hat{d}$ обикновено е по-малка от дисперсията, задавана от формулата (19.1), в която S_1 и S_2 са заменени на S. Причината е в това, че използването на една единствена извадка S елиминира дисперсията, причинена от случайните разлики в композицията на S_1 и S_2. В този случай, доверителният интервал, определян от ф-ла (19.2) ще бъде прекалено консервативен, макар че, в общия случай, коректен.

14.1.1. Проверка на хипотези

В някои случаи сме заинтересувани не в оценяване на доверителните интервали за определени параметри, а в преценката за вероятността, че определено утвърждение е вярно. Да предположим, например, че искаме да изясним каква е вероятността, че $error_D(h_1) > error_D(h_2)$. Съгласно схемата, описана в предишния раздел, да предположим, че ние измерихме извадковите грешки за h_1 и h_2, използвайки две независими извадки S_1 и S_2 с размер 100, и изчислихме, че $error_{S_1}(h_1) = 0.30$, а $error_{S_2}(h_2) = 0.20$, така че наблюдаваната разлика $\hat{d} = 0.10$. Ясно е, че поради случайната вариация в извадката от данни, ние можем да наблюдаваме тази разлика в извадковата грешка дори когато $error_D(h_1) \leq error_D(h_2)$. Каква е вероятността, че $error_D(h_1) > error_D(h_2)$, при условие, че извадковата грешка $\hat{d} = 0.10$? Т.е. каква е вероятността, че $d > 0$ при наблюдаваната $\hat{d} = 0.10$?

Обърнете внимание, че *Pr(d > 0)* е равна на вероятността, че $\hat{d}$ не надценява d с повече от 0.10. С други думи, това е вероятността, че $\hat{d}$ попада в едностранен интервал $\hat{d} < d + 0.10$. Тъй като d е средната стойност на разпределението, на което се подчинява $\hat{d}$, този едностранен интервал може да бъде преписан в еквивалентна форма като $\hat{d} < \mu_{\hat{d}} + 0.10$.

Да преизчислим по-горния интервал в термините на стандартното отклонение, т.е. $\hat{d} < \mu_{\hat{d}} + z_N \sigma_{\hat{d}}$ (да представим 0.10 във вид: $0.10 = z_N \sigma_{\hat{d}}$). За изчисляването на $\sigma_{\hat{d}}$ ще използваме ф-ла (19.1); получаваме $\sigma_{\hat{d}} \approx 0.061$. Следователно, $z_N = \frac{0.10}{\sigma_{\hat{d}}} \approx 1.64$.

Какво равнище на достоверност (N%) отговаря на получения едностранен интервал за нормалното разпределение? Съгласно Таблица 18-1 от предишната лекция, на константата 1.64 отговаря двустранен интервал с нивото на достоверност 90%, следователно едностранен интервал (виж раздел 18.4.1. от предишната лекция) ще има асоциирано равнище на достоверност от 95%.

И така, при наблюдавана стойност $\hat{d} = 0.10$, вероятността, че $error_D(h_1) > error_D(h_2)$ е приблизително 95%. Съгласно терминологията, приета в статистическата литература, можем да кажем, че *„приемаме хипотезата, че $error_D(h_1) > error_D(h_2)$"* с 95% увереност (достоверност). Аналогично, можем да кажем, че ние *отхвърляме обратната хипотеза (че $error_D(h_1) = error_D(h_2)$)*, т.е. че наблюдаваните различия се дължат на случайни флуктуации в извадковите данни – така наречена *нулевата хипотеза*) с равнището на значимост (1 – 0.95) = 0.05. (С други думи, вероятността, че отхвърлената нулева хипотеза е правилна (а ние сме сгрешили), е 5%).

19.2. Сравнение на алгоритми за обучение

В машинното самообучение ние сме често заинтересувани не от сравняване на хипотези, а от сравняване на различни *алгоритми за самообучение*. Какъв тест е подходящ за сравнение на такива алгоритми и как можем да определим, дали наблюдаваните разлики в поведението на алгоритми за самообучение са статистически значими? Този въпрос все още не е намерил еднозначното решение сред изследователи, работещи в тази област. В този раздел ще разгледаме един от най-общо приетите подходи към този въпрос.

19.2.1. t-тест по двойки

Както винаги започваме с определяне на параметъра, който искаме да оценяваме. Да предположим, че искаме да определим, кой от двата алгоритъма за обучение L_A или L_B е по-добър метод (в средно) за научаване на определена целева функция f. Един разумен начин да дефинираме понятие "в средно" е да разгледаме

относителното поведение на тези два алгоритъма осреднено по всички възможни обучаващи множества с размер *n*, които могат да бъдат съставени по случаен начин от примери, взети от разпределението $\mathcal{D}$. С други думи, искаме да оценим очакваната (средна) стойност на разликата между грешките на тези алгоритми:

$$\underset{S \subset D}{E}[error_D(L_A(S)) - error_D(L_B(S))] \quad (19.3)$$

където *L(S)* означава хипотеза, извеждана от метод за самообучение *L*, когато е зададена извадка *S* от обучаващите данни, а осредняване по $S \subset D$ означава, че очакваната стойност се взима по всички извадки *S*, съставени по случаен начин съгласно вероятностното разпределение $\mathcal{D}$.

Естествено, на практика при сравняване на методи за самообучение ние разполагаме само с определена ограничена извадка от данни D_0. В такива случаи един разумен начин за оценяване на описаната по-горе стойност е да разбием D_0 на обучаващото множество S_0 и непресичащото с него тестово множество T_0. Обучаващото множество ще се използва за обучение и на двата алгоритъма L_A и L_B, а тестовото множество ще се използва за сравнение на точността на научените две хипотези. С други думи, ние ще измерваме:

$$error_{T_0}(L_A(S_0)) - error_{T_0}(L_B(S_0)) \quad (19.4)$$

Обърнете внимание на две ключови разлики между този оценител и величината от уравнението (19.3). Първо, ние използваме $error_{T_0}(h)$, за да апроксимираме $error_D(h)$. Второ, ние измерваме разликата в грешки само за едно единствено обучаващо множество S_0, вместо да търсим средната стойност на тази разлика върху всички извадки от *S*, които могат да бъдат направени съгласно вероятностното разпределение $\mathcal{D}$.

Един начин за подобряване на оценителя, зададен от ф-ла (19.4) е да повторим няколко пъти разбиването на наличните данни D_0 на непресичащи се обучаващи и тестващи множества и да изчислим средната стойност от грешките върху всички тези тестови множества. Това ни води до метода за оценяването на разликата между грешки на два алгоритъма за обучение върху фиксирано множество от обучаващи данни D_0 чрез така наречена процедура за *крос-валидацията* (cross-validation), схематично описана в Таблица 19-1. При тази процедура отначало данните се разделят на *k* непресичащи се подмножества с еднакъв размер, който трябва да бъде не е по-малък от 30. След това сравняваните алгоритми за обучение се тестват *k* пъти, използвайки по ред всеки от тези *k* подмножества като тестовото множество, а останалите данни (*k–1* подмножества) – като обучаващото множество. По този начин алгоритмите за самообучение се тестват върху *k* независими тестови множества и средната стойност на разликите между направените грешки $\bar{\delta}$ се връща като оценител на разликата между истинските грешки на сравняваните алгоритми.

1. Разбий наличните данни D_0 на k непресичащи се подмножества $T_1, T_2, \ldots, T_k$ с еднакъв размер, който е не по-малък от 30.
2. От i до k направи:
   Използвай T_i като тестовото множество, а останалите данни – като обучаващото множество S_i

   - $S_i \leftarrow \{D_0 - T_i\}$
   - $h_A \leftarrow L_A(S_i)$
   - $h_B \leftarrow L_B(S_i)$
   - $\delta_i \leftarrow error_{T_i}(h_A) - error_{T_i}(h_B)$

3. Върни стойността $\overline{\delta}$, където
   $$\overline{\delta} \equiv \frac{1}{k}\sum_{i=1}^{k}\delta_i$$

Таблица 19-1. Процедурата за k-крос-валидация

Стойността на $\overline{\delta}$ може да се използва като оценителят на желаната величина от уравнение (19.3), по-точно ние може да разглеждаме $\overline{\delta}$ като оценителя на величината:

$$\underset{S\subset D_0}{E}[error_D(L_A(S)) - error_D(L_B(S))] \quad (19.5)$$

където S представлява случайната извадка с размер $\frac{k-1}{k}|D_0|$, избрана по един унифициран начин от D_0. Единствената разлика между този израз и оригиналния израз от ф-ла (19.3) е, че в новият израз средната стойност се изчислява върху подмножествата на наличното множество от данни D_0, а не върху подмножествата, избрани от цялото възможно разпределение на примери $\mathcal{D}$.

Приблизителният N% доверителен интервал за оценяване на величината от уравнение (19.5) чрез $\overline{\delta}$, се задава от формулата:

$$\overline{\delta} \pm t_{N,k-1}s_{\overline{\delta}} \quad (19.6)$$

където $t_{N,k-1}$ е константа, която играе роля, аналогична на z_N при предишните записи на доверителните интервали, а $s_{\overline{\delta}}$ е оценката за стандартното отклонение на разпределение, на което се подчинява $\overline{\delta}$. В частност, $s_{\overline{\delta}}$ се определя като:

$$s_{\bar{\delta}} \equiv \sqrt{\frac{1}{k(k-1)}\sum_{i=1}^{k}(\delta_i - \bar{\delta})^2} \quad (19.7)$$

Константата $t_{N,k-1}$ характеризира областта под кривата на вероятностното разпределение, известно като t-разпределение на Стюдент, по същия начин като z_N характеризира областта под кривата на нормалното разпределение. t-разпределение също има камбановидната форма както и нормалното, но е по-широко и по-късо, отразявайки по-голямата дисперсия, причинена от използването на $s_{\bar{\delta}}$ за апроксимацията на истинското стандартно отклонение $\sigma_{\bar{\delta}}$. $t_{N,k-1}$ има два индекса – първият определя желаното равнище на достоверност (често вместо N се използва еквивалентната стойност на $\alpha = 1 - N$ - равнище на значимост), както става и при z_N. Вторият параметър, който се нарича *степени на свобода*, който обикновено се означава с v, е свързан с броя на независими случайни събития, които участват в определяне на стойността на случайната величина $\bar{\delta}$. В нашия случай $v = k - 1$. Някои избрани стойности на $t_{N,k-1}$ са показани в Таблица 19-2. При $k \to \infty$ стойността на $t_{N,k-1}$ се приближава към тази на z_N.

	Равнище на достоверност N			
	90%	95%	98%	99%
$\nu = 2$	2.92	4.30	6.96	9.92
$\nu = 5$	2.02	2.57	3.36	4.03
$\nu = 10$	1.81	2.23	2.76	3.17
$\nu = 20$	1.72	2.09	2.53	2.84
$\nu = 30$	1.70	2.04	2.46	2.75
$\nu = 120$	1.66	1.98	2.36	2.62
$\nu = \infty$	1.64	1.96	2.33	2.58

Таблица 19-2. t-разпределение на Стюдент за двустранни доверителни интервали

Описаната по-горе процедура за сравнение на два метода за обучение се базира върху тестването на научените хипотези върху едни и същи тестови множества. Това контрастира с метода за сравняване на точността на хипотези, разгледан в началото на тази лекция, където хипотези са били оценявани върху различни тестови множества. Тестове, където хипотезите се оценяват върху еднакви извадки, се наричат *тестове по двойки (paired tests)*. Тестовете по двойки обикновено произвеждат по-тесни доверителни интервали, тъй като всички разлики в наблюдаваните грешки при един тест по двойки са причинени само от разликите между сравняваните хипотези. При тестването на хипотези върху различни извадки от данни, наблюдаваните разлики могат да се дължат и на разликите във конструирането на самите извадки.

Пример 1.

За илюстрация да сравним работата на два реални алгоритъма за обучение – наивен Бейсов класификатор - NB и алгоритъмът за най-близкия съсед – 1-NN, класифициращ нов пример с класа на най-близкия му съсед сред обучаващите примери[1]. Точността на алгоритмите ще оценяваме върху базата от реални данни, съдържаща 258 примера, събрани в Медицинския университет в Любляна. Всеки пример се описва с 9 атрибута и отразява определени параметри на пациентка, на която преди време беше направена операцията срещу рака на гърдите. Целевият атрибут показва, дали пациентката отново е болна от рак на гърдата. Целта на алгоритъма за обучение е да използва тези данни за да построи хипотеза, позволяваща да предсказва повтарянето на заболяването.

От начало ще приложим процедурата за *k*-крос-валидацията, описана в Таблица 19-1, като изберем $k = 6$. Това означава, че ще разбием базата на 6 (почти) равни части и ще тестваме, последователно, и двата алгоритъма 6 пъти, използвайки всяка част за тестване, а останали данни – за обучение. Резултати от тези експерименти (класификационни грешки) са показани в Таблица 19-3:

№	$error_i(NB)$	$error_i(1\text{-}NN)$	$\delta_i = error_i(1\text{-}NN) - error_i(NB)$
1	0.28	0.21	-0.07
2	0.27	0.31	0.04
3	0.36	0.40	0.04
4	0.33	0.29	- 0.04
5	0.21	0.45	0.24
6	0.27	0.33	0.06
Средно	*0.29*	*0.33*	$\overline{\delta} = 0.04$

Таблица 19-3. Класификационни грешки на NB и 1-NN

Експеримент показа, че наблюдаваната средна класификационната грешка на наивен Бейсов класификатор е по-малка от тази на алгоритъма за най-близкия съсед. Дали можем да твърдим, че тази разлика е *статистически значима*, т.е. че истинската грешка на първият алгоритъм ще бъде по-малка от втория?

За да отговорим на този въпрос ще построим нулевата (H_0) и алтернативната (H_A) хипотези е:

- H_0 – алгоритмите имат една и съща истинската грешка,
- H_A – наивният Бейсов класификатор е по-точен от алгоритъма за най-близкия съсед върху неограничен брой данни от този тип (т.е. $error_D(1\text{-}NN) > error_D(NB)$)

[1] Алгоритмите както и схема за тяхното сравнение са реализирани от автора на тези лекции и неговите студенти от СУ в експерименталната система DaMiS.

След това ще изчислим статистиката $t = \frac{\bar{\delta}}{s_{\bar{\delta}}}$, където $s_{\bar{\delta}}$ се изчислява по ф-ла (19.7)

След изчисленията получаваме: $t = 1.14$. За да отхвърлим нулевата хипотеза с равнище на значимост 0.05 (т.е.със равнище на достоверност 95%), трябва да бъде изпълнено условие: $t \geq t_{N,k-1} = t_{95,5}$. Използвайки Таблицата 19-2 и припомняйки, че константа, съответстваща на 95% едностранен доверителен интервал отговаря на 90% двустранен интервал, намираме, че $t_{95,5} = 2.02$. Следователно, ние не можем да отхвърлим нулевата хипотеза, че предсказващата точност за повтаряне на рака и на двата алгоритъма е еднаква с исканата степен на увереност – по тази причина алтернативната хипотеза не се приема. С други думи, от статистическата гледна точка наблюдаваната разлика в точността на двата алгоритъма върху избраната база данни е резултат от случайния начин на нареждането на примери, използвани за обучение и тестване, а не от различното класификационно поведение на алгоритмите.

19.2.2. Тест на Уилкоксон за знаково ранжиране по двойки

Както вече стана ясно от предишните лекции, всеки алгоритъм за индуктивното самообучение притежава явно или неявно индуктивно пристрастие, позволяващо му да класифицира неизвестни примери. Когато характеристиките на проблемната област съвпадат с индуктивното пристрастие на алгоритъма, неговата класификационна точност върху примери от тази област е голяма. Обаче, основната цел ма машинното самообучение е не да конструират алгоритми, насочени към строго определена предметна област, а да бъдат създадени сравнително универсални алгоритми, които притежават достатъчно добра класификационна точност върху широк клас от проблемните области. Обаче, съгласно така наречена "теорема за липса на безплатен обед" или "консервативен закон" (Schaffter 1994) е доказано, че ако един алгоритъм за обучение е по-добър от другия в някои предметни области, то задължително ще се намерят някои други области, в които това съотношение е обратно! Повече от това - средната точност на един алгоритъм върху всички възможни проблемни области е константа, независеща от вида на алгоритъма.

Дали тези теоретични резултати означават, че няма смисъл да търсим по-универсални алгоритми за самообучение? Не, необходимо е да се прави разликата между всички *математически възможни* проблемни области, които са просто резултат от използвания език за представяне на примери и хипотези, и тези проблемни области, които се срещат в реалния свят и по тази причина са от особената важност за машинното самообучение. Ясно е, че определени алгоритми ще продължат да "пасват" по-добре на определени класове от реални проблемни области, постигайки по-добра класификационна точност от колко другите алгоритми, както и че това може да бъде обратно за другите класове на проблемните области. Обаче това не значи, че не е възможно да бъде създаден

алгоритъм, който ще бъде по-добър от най-добрите за всеки клас от проблемните области.

В предишния раздел ние се научихме как да сравняваме алгоритми за самообучение върху една единствена проблемна област. В този раздел ще бъде показано, как можем да сравним два алгоритъма върху няколко различни проблемни области. Възниква естествения въпрос – не можем ли да използваме за тази цел вече разгледаните по-рано в тази лекция методи? Отговорът е "не". Проблемът е в това, че в основа на тези методи лежат определени *базови* предположения, такива като че всички извадки от данни се правят от *една и съща генерална съвкупност* от данни, както и че случайните величини, използвани за моделиране на процеса на сравнение, се подчиняват на определено вероятностно разпределение – биномно, нормално или на Стюдент, с известни параметри (средно, дисперсия). По тази причина тези методи се наричат "параметрични".

В случая на използване на *няколко* проблемни области, вече не можем да говорим за една и съща генерална съвкупност, подчиняваща се на определено вероятностно разпределение. Обаче, ние можем да разглеждаме разликата в точността на двата алгоритъма върху всяка предметна област като наблюдаваните стойности на някоя случайна величина, вероятностното разпределение на която не ни е известно, и да опитаме статистически да оценим *качествена* хипотезата, дали тази разлика, като средно, е положителна върху цялото избрано множество от проблемните области, върху които се сравняват два алгоритъма. За отговор на този въпрос могат да бъдат използвани така наречени *непараметрични* статистически методи за сравняване на хипотези (които не правят никакви предположения за вида на разпределението), един от които е тестът на Уилкоксон за знаково ранжиране по двойки (Wilcoxon signed-ranked test – matched pairs).

Да предположим, че искаме да определим, кой от двата алгоритъма за обучение – *A* или *B,* има по-добра класификационна точност върху множество бази от обучаващи данни от различни предметни области $Dom_1, \ldots, Dom_n$. Съгласно общия метод за оценяване на хипотези, започваме с определяне на величина, която искаме да оценим. Нека разликата в класификационната грешка на двата алгоритъма върху всички избрани проблемни области е:

$$\delta \equiv error(L_A(D)) - error(L_B(D)), \text{където D} \equiv \bigcup_{i=1}^{n} Dom_i$$

За удобство, да означим $\Delta \equiv 1 - \delta$ - разликата в класификационната точност на алгоритмите – това ще е величина, подлежаща на оценяване.

Като оценител, ще използваме разликите в наблюдаемите класификационни точности на двата алгоритъма върху проблемните области Dom_i. $\Delta_i \equiv 1 - \delta_i$, където $\delta_i = error(L_A(Dom_i)) - error(L_B(Dom_i))$.

Тест на Уилкоксон за знаково ранжиране по двойки се състои в следното:

Едностранен тест:

H_0 (Нулева хипотеза): Класификационната точност на двата алгоритъма като цяло върху всички бази $Dom_1, ..., Dom_n$. е еднаква (т.е. $\Delta = 0$).

H_A (Алтернативна хипотеза): Класификационната точност на алгоритъма A като цяло върху всички бази $Dom_1, ..., Dom_n$. е по-голяма (т.е. $\Delta > 0$).

(или H_A): Класификационната точност на алгоритъма B като цяло върху всички бази $Dom_1, ..., Dom_n$. е по-голяма (т.е. $\Delta < 0$).

- Изчисляваме разликите в класификационните точности на двата алгоритъма $\Delta_i \equiv 1 - \delta_i$ за всички n проблемни области. За получаването на тази оценка можем да използваме вече познат ни алгоритъм за k-крос-валидацията (виж Таблица 19-1), като за да намалим влияние на начина за конструиране на всяка тестова извадка желателно и двата алгоритъма да се тестват върху едни и същи извадки.
- Ранжираме абсолютните стойности на тези n разлики от най-малката (ранг 1), до най-голямата (ранг n) и изчисляваме статистиката - сумата на рангове T^- от отрицателните разлики (или сумата на рангове T^+ от положителните разлики).
- Нулевата хипотеза се отхвърля, ако

  $$T^- \le T_0 \ (\textit{или}\ T^+ \le T_0)$$

  където T_0 се взима от таблицата за критични стойности на теста на Уилкоксон за знаково ранжиране по двойки (виж, например, Таблица 19-4).

Забележка 1: Разликите, равни на 0, се елиминират и броят на разликите n се намалява съответно. Еднакви по величина абсолютни разлики получават еднакви рангове, равни на средната стойност от рангове, които те би получиха, ако нямаше съвпадение. *(Например, имаме (подредени и абсолютни) разлики 0; 0; 0.1; 1.2; 2.1; 2.1; 2.1; 3.5; 4. На тях се присвояват следните рангове: 1; 2; (3+4+5)/3; (3+4+5)/3; (3+4+5)/3; 6; 7, а n = 9 – 2 = 7)*

Забележка 2: При *двустранен тест* алтернативната хипотеза е: $\Delta \neq 0$, а в качеството на статистиката се избира $T = min\{T^-, T^+\}$.

Едностранен	Двустранен	n = 5	n = 6	n = 7	n = 8	n = 9	n = 10
$\alpha = .05$	$\alpha = .10$	1	2	4	6	8	11
$\alpha = .025$	$\alpha = .05$		1	2	4	6	8
$\alpha = .01$	$\alpha = .02$			0	2	3	5
$\alpha = .005$	$\alpha = .01$				0	2	3
		n = 11	n = 12	n = 13	n = 14	n = 15	n = 16
$\alpha = .05$	$\alpha = .10$	14	17	21	26	30	36
$\alpha = .025$	$\alpha = .05$	11	14	17	21	25	30
$\alpha = .01$	$\alpha = .02$	7	10	13	16	20	24
$\alpha = .005$	$\alpha = .01$	5	7	10	13	16	19

Таблица 14.4. Част от таблицата за критични стойности на теста на Уилкоксон за знаково ранжиране по двойки

Пример 2.

Ще продължим със сравнение на класификационната точност на два реални алгоритъма за обучение – наивен Бейсов класификатор – NB и алгоритъмът за най-близкия съсед – 1-NN, за които ставаше дума в примера 1. Сега ще ги сравним върху 15 еталонни бази с реални данни, взети от хранилище на еталонни бази за машинно самообучение http://www.ics.uci.edu/~mlearn/MLRepository.html. Някои основни характеристики на тези бази са приведени в Таблица19-5.

База	***Описание***	***Примери***	***Класове***	***Брой Атрибути***
Breast Cancer (BC)	Съдържа база свързана с рака от института по онкология в Любляна, Югославия	286	2	9
Breast Cancer Wisconsin (BCW)	Още една база, свързана с рака от университета в Уисконсин	699	2	10
Heart Diseses (HD)	Една медицинска база, свързана с болести на сърцето от Др. Детрано	303	2	13
Credit (CR)	База, съдържаща информация за кредитни карти (всички имена и символни стойности са заменени с безсмислени за да се запази тайната на информацията.	690	2	15
Diabetes (DI)	Съдържа информация за разпространението на диабет между индианците Пима	768	2	8
Glass (GL)	База, съдържаща информация за разпознаване на вида на стъклото	214	6	9
Glass2 (GL2)	Както по-горе, но са махнати част от класовете и съответно е с по-малко примери	163	2	9
Horse (HC) Colic	База, свързана с разпространението на колики при конете	368	2	22
Iris (IR)	База за класификация на растенията Iris (перуника)	150	3	4
Hepatitis (HE)	База, свързана с заболяване на хепатит, предоставена от университета в Любляна	5155	2	20
Liver Disease (LD)	База със случаи на заболявания на черен дроб и данните от кръвни проби	345	2	7
Vehicle (VE)	База за класифициране силуета на кола	846	4	18
Vote (VO)	База съдържаща към коя партия принадлежи един конгресмен, в зависимост от това как е гласувал mf 16 ключови гласувания	435	2	16
Wine (WI)	Съдържа химическа информация за вино от три различни места в Италия	178	3	13
Zoo (ZO)	Проста база съдържаща само булеви атрибути и определяща класификация на животни	101	7	16

Таблица 19-5. Някои параметри на бази данни, използвани при експерименти

Всяка база от данни се разбива на 6 (приблизително) равни части и всеки от сравнявани алгоритми се тества върху тях с използване на процедурата на 6-крос-

валидация. Тъй като някои бази съдържат непрекъснати атрибути, то преди да бъде пуснат наивният Бейсов класификатор всяко обучаващо множество се подлага на процедурата за дискретизация. При използването на алгоритъма за най-близък съсед непрекъснати атрибути бяха подлагани само на процедурата за нормализация.

Получените средни стойности на точността, разликите, техни абсолютни стойности и назначени на тях рангове са приведени в Таблица 19-6.

База	***Точност % (NB)***	***Точност % (1-NN)***	Δ_i	$\lvert\Delta_i\rvert$	***Ранг***
BC	71.7	66.8	4.9	4.9	12
BCW	97.6	95.9	1.7	1.7	(3+4)/2=3.5
CR	85.5	79.8	5.7	5.7	13
DI	74.3	70.8	3.5	3.5	10
GL	72.0	69.2	2.8	2.8	(7+8)/2=7.5
GL2	81.6	79.7	1.9	1.9	5
HD	83.3	80.5	2.8	2.8	(7+8)/2=7.5
HE	79.6	80.0	**-0.4**	0.4	**1**
HC	79.6	78.5	1.1	1.1	2
IR	93.3	95.3	**-2.0**	2.0	**6**
LD	63.2	64.9	**-1.7**	1.7	(3+4)/2=**3.5**
VE	63.0	70.1	**-7.1**	7.1	**14**
VO	90.1	94.9	**-4.8**	4.8	**11**
WI	98.3	94.9	3.4	3.4	9
ZO	97.0	97.0	0	0	-

Таблица 19-6. Данни за теста на Уилкоксон за знаково ранжиране по двойки

И така, $n = 15 - 1 = 14$; $T^- = 1 + 6 + 3.5 + 11 + 14 = 35.5$; $T_0(n = 14, \alpha = 0.05) = 26$ (виж Таблица 19.4). Следователно, не можем да отхвърлим нулевата хипотеза с достоверност от 95%. По тази причина смятаме, че статистически и двата алгоритъма имат една и съща класификационна точност върху тези 15 бази от данни.

Резюме

Основните дефиниции и факти от математическата статистика са резюмирани в следната таблица.

- *Случайна величина* може да се разглежда като име на един експеримент с вероятностен резултат. Нейната стойност е резултатът от този експеримент.

- *Вероятностно разпределение* на някоя случайна величина Y определя вероятността $Pr(Y = y_i)$, че Y ще приеме стойност y_i за всяко възможно значение на y_i.
- *Математическо очакване* или *средно (средна стойност)* на някоя случайна величина Y е $E[Y] = \sum_i y_i \Pr(Y = y_i)$. Често за представяне на *E[Y]* се използва буквата μ.
- *Дисперсия* на някоя случайна величина Y е $Var(Y) = E[(Y - \mu_Y)^2]$. Дисперсията характеризира ширината или разсейването на разпределението относно средната стойност.
- *Стандартно отклонение* на Y е $\sqrt{Var(Y)}$. За означаване на стандартното отклонение често се използва символът σ_Y.
- *Биномното разпределение* задава вероятността за наблюдаване на r гербове в серията от n независими хвърления на една монета, ако вероятността да падне герб при едно хвърляне е p.
- *Нормалното разпределение* е камбановидното вероятностно разпределение, което моделира много естествени явления.
- *Централната пределна теорема* е теоремата, утвърждаваща, че сумата от голямо количество независими, еднакво разпределени случайни величини приблизително следва нормалното разпределение.
- *Оценителят* е случайна величина Y, която се ползва за оценяване на определен параметър p на генералната съвкупност от данни
- *Оценъчното предпочитание (субективната оценка)* на Y като оценителят на p е величината *(E[Y] – p)*. Обективният (не субективен) оценител е този, за който субективната оценка е нула.
- *N% доверителният интервал* за оценяване на параметър p е такъв интервал, който включва p с вероятността *N%*.
- *t-test по двойки* позволява сравнение на класификационната точност на два алгоритъма за обучение върху избраната база от данни
- *Тест на Уилкоксон за знаково ранжиране по двойки* позволява сравнение на класификационната точност на два алгоритъма за обучение върху множество от различни бази данни.