Marek Gągolewski
Konstancja Bobecka-Wesołowska
Przemysław Grzegorzewski

Computer Statistics with R

6. Interval Estimation

Faculty of Mathematics and Information Science
Warsaw University of Technology
[Last update: December 9, 2012]

Contents

Info

These tutorials are likely to contain bugs and typos. In case you find any don't hesitate to *contact us*! Thanks in advance!

6.1. Preliminaries

Notation. Confidence level: $1-\alpha \in (0,1)$. Values of quantile functions of different distributions at α are denoted as follows. z_α — standardized normal distribution, $t_\alpha^{[n]}$ — Student's t distribution with n degrees of freedom, $\chi^2_{\alpha,n}$ — chi-square distribution with n degrees of freedom.

6.1.1. Confidence intervals for mean μ

6.1.1.1. Model I

$\mathbf{X} = (X_1, X_2, \dots, X_n)$ i.i.d. $\mathrm{N}(\mu,\sigma)$, known σ.

$$\left(\overline{\mathbf{X}} - z_{1-0.5\alpha}\frac{\sigma}{\sqrt{n}}, \quad \overline{\mathbf{X}} + z_{1-0.5\alpha}\frac{\sigma}{\sqrt{n}}\right). \tag{6.1}$$

is a confidence interval for μ with confidence level $1-\alpha$.

6.1.1.2. Model II

$\mathbf{X} = (X_1, X_2, \dots, X_n)$ i.i.d. $\mathrm{N}(\mu,\sigma)$, unknown σ.

$$\left(\overline{\mathbf{X}} - t^{[n-1]}_{1-0.5\alpha}\frac{s_{\mathbf{X}}}{\sqrt{n}}, \quad \overline{\mathbf{X}} + t^{[n-1]}_{1-0.5\alpha}\frac{s_{\mathbf{X}}}{\sqrt{n}}\right). \tag{6.2}$$

is a confidence interval for μ with confidence level $1-\alpha$.
See also: `t.test()`.

6.1.1.3. Model III

$\mathbf{X} = (X_1, X_2, \dots, X_n)$ i.i.d. ???, large n.

$$\left(\overline{\mathbf{X}} - z_{1-0.5\alpha}\frac{s_{\mathbf{X}}}{\sqrt{n}}, \quad \overline{\mathbf{X}} + z_{1-0.5\alpha}\frac{s_{\mathbf{X}}}{\sqrt{n}}\right). \tag{6.3}$$

is a confidence interval for μ with confidence level $1-\alpha$.
See also: `t.test()`.

6.1.2. Confidence intervals for variance σ^2

6.1.2.1. Model I

$\mathbf{X} = (X_1, X_2, \dots, X_n)$ i.i.d. $\mathrm{N}(\mu,\sigma)$, known μ.

$$\left(\frac{n\tilde{s}^2_{\mathbf{X}}}{\chi^2_{1-0.5\alpha,n}}, \quad \frac{n\tilde{s}^2_{\mathbf{X}}}{\chi^2_{0.5\alpha,n}}\right). \tag{6.4}$$

is a confidence interval for σ^2 with confidence level $1-\alpha$.

6.1.2.2. Model II

$\mathbf{X} = (X_1, X_2, \dots, X_n)$ i.i.d. $\mathrm{N}(\mu,\sigma)$, unknown μ.

$$\left(\frac{(n-1)s^2_{\mathbf{X}}}{\chi^2_{1-0.5\alpha,n-1}}, \quad \frac{(n-1)s^2_{\mathbf{X}}}{\chi^2_{0.5\alpha,n-1}}\right). \tag{6.5}$$

is a confidence interval for σ^2 with confidence level $1-\alpha$.

6.1.2.3. Model III

$\mathbf{X} = (X_1, X_2, \ldots, X_n)$ i.i.d. ???, large n.

$$\left(\frac{(2n-2)s_{\mathbf{X}}^2}{\left(\sqrt{2n-3} + z_{1-0.5\alpha}\right)^2}, \quad \frac{(2n-2)s_{\mathbf{X}}^2}{\left(\sqrt{2n-3} - z_{1-0.5\alpha}\right)^2} \right). \tag{6.6}$$

is a confidence interval for σ^2 with confidence level $1-\alpha$.

6.1.3. Confidence intervals for a proportion p

6.1.3.1. Model I

$\mathbf{X} = (X_1, X_2, \ldots, X_n)$ i.i.d. Bern(p), large n. Let $\widehat{p} = \frac{1}{n}\sum_{i=1}^{n} X_i$.

$$\left(\widehat{p} - z_{1-0.5\alpha}\sqrt{\frac{\widehat{p}(1-\widehat{p})}{n}}, \quad \widehat{p} + z_{1-0.5\alpha}\sqrt{\frac{\widehat{p}(1-\widehat{p})}{n}} \right). \tag{6.7}$$

is an approximate confidence interval for σ^2 with confidence level $1-\alpha$.

See also: `binom.test()`, `prop.test()`.

6.1.4. Calculating confidence intervals in R

With R, it is easy to construct confidence intervals for the mean μ of a normal population $\mathrm{N}(\mu, \sigma)$ with unknown σ (or for any distribution if the sample size is large) and for the probability of success p of a sequence of Bernoulli trails. Other confidence intervals should be calculated manually, using the equations given above.

Example. Let $(X_1, \ldots, X_n)$ be a sample drawn from a normal distribution $\mathrm{N}(\mu, \sigma)$ with unknown μ and σ. To construct a confidence interval for the mean μ we can use the `t.test()` function with a data vector `x` containing sample values and a desired confidence level (0.95 by default) as arguments.

E.g. `t.test(x, conf.level=0.9)`.

Example. Let $(X_1, \ldots, X_n)$ be a sample drawn from a Bernoulli distribution Bern(p). To construct the confidence interval for p we can use the `binom.test()` function, which calculates an "exact" confidence interval or `prop.test()` which calculates an approximate confidence interval for p. The arguments of both functions are: the number of successes in a sample (k), sample size (n) and the desired confidence level (0.95 by default).

E.g. `binom.test(k, n, conf.level=0.9)`.

6.2. Statistical tables

6.2.1. Cumulative standardized normal distribution function

Cumulative Standardized Normal Distribution Function, $\Phi(z)$. *Note:* $z < 0 \Rightarrow \Phi(z) = 1 - \Phi(-z)$. $\Phi(z) = \texttt{pnorm}(z)$.

z	0.00	0.01	0.02	0.03	0.04	0.05	0.06	0.07	0.08	0.09
0.0	0.5000000	0.5039894	0.5079783	0.5119665	0.5159534	0.5199388	0.5239222	0.5279032	0.5318814	0.5358564
0.1	0.5398278	0.5437953	0.5477584	0.5517168	0.5556700	0.5596177	0.5635595	0.5674949	0.5714237	0.5753454
0.2	0.5792597	0.5831662	0.5870644	0.5909541	0.5948349	0.5987063	0.6025681	0.6064199	0.6102612	0.6140919
0.3	0.6179114	0.6217195	0.6255158	0.6293000	0.6330717	0.6368307	0.6405764	0.6443088	0.6480273	0.6517317
0.4	0.6554217	0.6590970	0.6627573	0.6664022	0.6700314	0.6736448	0.6772419	0.6808225	0.6843863	0.6879331
0.5	0.6914625	0.6949743	0.6984682	0.7019440	0.7054015	0.7088403	0.7122603	0.7156612	0.7190427	0.7224047
0.6	0.7257469	0.7290691	0.7323711	0.7356527	0.7389137	0.7421539	0.7453731	0.7485711	0.7517478	0.7549029
0.7	0.7580363	0.7611479	0.7642375	0.7673049	0.7703500	0.7733726	0.7763727	0.7793501	0.7823046	0.7852361
0.8	0.7881446	0.7910299	0.7938919	0.7967306	0.7995458	0.8023375	0.8051055	0.8078498	0.8105703	0.8132671
0.9	0.8159399	0.8185887	0.8212136	0.8238145	0.8263912	0.8289439	0.8314724	0.8339768	0.8364569	0.8389129
1.0	0.8413447	0.8437524	0.8461358	0.8484950	0.8508300	0.8531409	0.8554277	0.8576903	0.8599289	0.8621434
1.1	0.8643339	0.8665005	0.8686431	0.8707619	0.8728568	0.8749281	0.8769756	0.8789995	0.8809999	0.8829768
1.2	0.8849303	0.8868606	0.8887676	0.8906514	0.8925123	0.8943502	0.8961653	0.8979577	0.8997274	0.9014747
1.3	0.9031995	0.9049021	0.9065825	0.9082409	0.9098773	0.9114920	0.9130850	0.9146565	0.9162067	0.9177356
1.4	0.9192433	0.9207302	0.9221962	0.9236415	0.9250663	0.9264707	0.9278550	0.9292191	0.9305634	0.9318879
1.5	0.9331928	0.9344783	0.9357445	0.9369916	0.9382198	0.9394292	0.9406201	0.9417924	0.9429466	0.9440826
1.6	0.9452007	0.9463011	0.9473839	0.9484493	0.9494974	0.9505285	0.9515428	0.9525403	0.9535213	0.9544860
1.7	0.9554345	0.9563671	0.9572838	0.9581849	0.9590705	0.9599408	0.9607961	0.9616364	0.9624620	0.9632730
1.8	0.9640697	0.9648521	0.9656205	0.9663750	0.9671159	0.9678432	0.9685572	0.9692581	0.9699460	0.9706210
1.9	0.9712834	0.9719334	0.9725711	0.9731966	0.9738102	0.9744119	0.9750021	0.9755808	0.9761482	0.9767045
2.0	0.9772499	0.9777844	0.9783083	0.9788217	0.9793248	0.9798178	0.9803007	0.9807738	0.9812372	0.9816911
2.1	0.9821356	0.9825708	0.9829970	0.9834142	0.9838226	0.9842224	0.9846137	0.9849966	0.9853713	0.9857379
2.2	0.9860966	0.9864474	0.9867906	0.9871263	0.9874545	0.9877755	0.9880894	0.9883962	0.9886962	0.9889893
2.3	0.9892759	0.9895559	0.9898296	0.9900969	0.9903581	0.9906133	0.9908625	0.9911060	0.9913437	0.9915758
2.4	0.9918025	0.9920237	0.9922397	0.9924506	0.9926564	0.9928572	0.9930531	0.9932443	0.9934309	0.9936128
2.5	0.9937903	0.9939634	0.9941323	0.9942969	0.9944574	0.9946139	0.9947664	0.9949151	0.9950600	0.9952012
2.6	0.9953388	0.9954729	0.9956035	0.9957308	0.9958547	0.9959754	0.9960930	0.9962074	0.9963189	0.9964274
2.7	0.9965330	0.9966358	0.9967359	0.9968333	0.9969280	0.9970202	0.9971099	0.9971972	0.9972821	0.9973646
2.8	0.9974449	0.9975229	0.9975988	0.9976726	0.9977443	0.9978140	0.9978818	0.9979476	0.9980116	0.9980738
2.9	0.9981342	0.9981929	0.9982498	0.9983052	0.9983589	0.9984111	0.9984618	0.9985110	0.9985588	0.9986051
3.0	0.9986501	0.9986938	0.9987361	0.9987772	0.9988171	0.9988558	0.9988933	0.9989297	0.9989650	0.9989992
3.1	0.9990324	0.9990646	0.9990957	0.9991260	0.9991553	0.9991836	0.9992112	0.9992378	0.9992636	0.9992886
3.2	0.9993129	0.9993363	0.9993590	0.9993810	0.9994024	0.9994230	0.9994429	0.9994623	0.9994810	0.9994991
3.3	0.9995166	0.9995335	0.9995499	0.9995658	0.9995811	0.9995959	0.9996103	0.9996242	0.9996376	0.9996505
3.4	0.9996631	0.9996752	0.9996869	0.9996982	0.9997091	0.9997197	0.9997299	0.9997398	0.9997493	0.9997585
3.5	0.9997674	0.9997759	0.9997842	0.9997922	0.9997999	0.9998074	0.9998146	0.9998215	0.9998282	0.9998347
3.6	0.9998409	0.9998469	0.9998527	0.9998583	0.9998637	0.9998689	0.9998739	0.9998787	0.9998834	0.9998879
3.7	0.9998922	0.9998964	0.9999004	0.9999043	0.9999080	0.9999116	0.9999150	0.9999184	0.9999216	0.9999247
3.8	0.9999277	0.9999305	0.9999333	0.9999359	0.9999385	0.9999409	0.9999433	0.9999456	0.9999478	0.9999499
3.9	0.9999519	0.9999539	0.9999557	0.9999575	0.9999593	0.9999609	0.9999625	0.9999641	0.9999655	0.9999670

6.2.2. Standardized normal quantiles

Standardized Normal Quantiles, z_α. *Note:* $0 < \alpha < 0.5 \Rightarrow z_\alpha = -z_{1-\alpha}$. $z_\alpha = \texttt{qnorm}(\alpha)$.

α	0.8	0.85	0.9	0.95	0.975	0.99	0.995	0.999	0.9995
z_α	0.8416212	1.0364334	1.2815516	1.6448536	1.9599640	2.3263479	2.5758293	3.0902323	3.2905267

6.2.3. Student's t quantiles

Student's t Quantiles, $t_\alpha^{[n]}$. *Note:* $0 < \alpha < 0.5 \Rightarrow t_\alpha^{[n]} = -t_{1-\alpha}^{[n]}$. $t_\alpha^{[n]} = \mathtt{qt}(\alpha, n)$.

	$\alpha = 0.8$	0.85	0.9	0.95	0.975	0.99	0.995	0.999	0.9995
$n = 1$	1.3763819	1.9626105	3.0776835	6.3137515	12.7062047	31.8205160	63.6567412	318.3088390	636.6192488
2	1.0606602	1.3862066	1.8856181	2.9199856	4.3026527	6.9645567	9.9248432	22.3271248	31.5990546
3	0.9784723	1.2497781	1.6377444	2.3533634	3.1824463	4.5407029	5.8409093	10.2145319	12.9239786
4	0.9409646	1.1895669	1.5332063	2.1318468	2.7764451	3.7469474	4.6040949	7.1731822	8.6103016
5	0.9195438	1.1557673	1.4758840	2.0150484	2.5705818	3.3649300	4.0321430	5.8934295	6.8688266
6	0.9057033	1.1341569	1.4397557	1.9431803	2.4469119	3.1426684	3.7074280	5.2076262	5.9588162
7	0.8960296	1.1191591	1.4149239	1.8945786	2.3646243	2.9979516	3.4994833	4.7852896	5.4078825
8	0.8888895	1.1081454	1.3968153	1.8595480	2.3060041	2.8964594	3.3553873	4.5007909	5.0413054
9	0.8834039	1.0997162	1.3830287	1.8331129	2.2621572	2.8214379	3.2498355	4.2968057	4.7809126
10	0.8790578	1.0930581	1.3721836	1.8124611	2.2281389	2.7637695	3.1692727	4.1437005	4.5868939
11	0.8755300	1.0876664	1.3634303	1.7958848	2.2009852	2.7180792	3.1058065	4.0247010	4.4369793
12	0.8726093	1.0832114	1.3562173	1.7822876	2.1788128	2.6809980	3.0545396	3.9296333	4.3177913
13	0.8701515	1.0794687	1.3501713	1.7709334	2.1603687	2.6503088	3.0122758	3.8519824	4.2208317
14	0.8680548	1.0762802	1.3450304	1.7613101	2.1447867	2.6244941	2.9768427	3.7873902	4.1404541
15	0.8662450	1.0735314	1.3406056	1.7530504	2.1314495	2.6024803	2.9467129	3.7328344	4.0727652
16	0.8646670	1.0711372	1.3367572	1.7458837	2.1199053	2.5834872	2.9207816	3.6861548	4.0149963
17	0.8632790	1.0690331	1.3333794	1.7396067	2.1098156	2.5669340	2.8982305	3.6457674	3.9651263
18	0.8620487	1.0671695	1.3303909	1.7340636	2.1009220	2.5523796	2.8784405	3.6104849	3.9216458
19	0.8609506	1.0655074	1.3277282	1.7291328	2.0930241	2.5394832	2.8609346	3.5794001	3.8834059
20	0.8599644	1.0640158	1.3253407	1.7247182	2.0859634	2.5279770	2.8453397	3.5518083	3.8495163
21	0.8590740	1.0626697	1.3231879	1.7207429	2.0796138	2.5176480	2.8313596	3.5271537	3.8192772
22	0.8582661	1.0614488	1.3212367	1.7171444	2.0738731	2.5083246	2.8187561	3.5049920	3.7921307
23	0.8575296	1.0603365	1.3194602	1.7138715	2.0686576	2.4998667	2.8073357	3.4849644	3.7676268
24	0.8568555	1.0593189	1.3178359	1.7108821	2.0638986	2.4921595	2.7969395	3.4667773	3.7453986
25	0.8562362	1.0583844	1.3163451	1.7081408	2.0595386	2.4851072	2.7874358	3.4501887	3.7251439
26	0.8556652	1.0575232	1.3149719	1.7056179	2.0555294	2.4786298	2.7787145	3.4349972	3.7066117
27	0.8551372	1.0567270	1.3137029	1.7032884	2.0518305	2.4726599	2.7706830	3.4210336	3.6895917
28	0.8546475	1.0559887	1.3125268	1.7011309	2.0484071	2.4671401	2.7632625	3.4081552	3.6739064
29	0.8541920	1.0553022	1.3114336	1.6991270	2.0452296	2.4620214	2.7563859	3.3962403	3.6594050
30	0.8537673	1.0546623	1.3104150	1.6972609	2.0422725	2.4572615	2.7499957	3.3851849	3.6459586
35	0.8520119	1.0520194	1.3062118	1.6895725	2.0301079	2.4377225	2.7238056	3.3400452	3.5911468
40	0.8506998	1.0500458	1.3030771	1.6838510	2.0210754	2.4232568	2.7044593	3.3068777	3.5509658
45	0.8496819	1.0485158	1.3006493	1.6794274	2.0141034	2.4121159	2.6895850	3.2814798	3.5202515
50	0.8488692	1.0472949	1.2987137	1.6759050	2.0085591	2.4032719	2.6777933	3.2614091	3.4960129
75	0.8464401	1.0436493	1.2929415	1.6654254	1.9921022	2.3771018	2.6429831	3.2024888	3.4250309
100	0.8452304	1.0418359	1.2900748	1.6602343	1.9839715	2.3642174	2.6258905	3.1737395	3.3904913
∞	0.8416212	1.0364334	1.2815516	1.6448536	1.9599640	2.3263479	2.5758293	3.0902323	3.2905267

6.2.4. χ^2 quantiles

χ^2 Quantiles, $\chi^2_{\alpha,n}$. $\chi^2_{\alpha,n} = \texttt{qchisq}(\alpha, n)$.

	$\alpha = 0.005$	0.01	0.025	0.05	0.95	0.975	0.99	0.995
n =1	0.0000393	0.0001571	0.0009821	0.0039321	3.8414588	5.0238862	6.6348966	7.8794386
2	0.0100251	0.0201007	0.0506356	0.1025866	5.9914645	7.3777589	9.2103404	10.5966347
3	0.0717218	0.1148318	0.2157953	0.3518463	7.8147279	9.3484036	11.3448667	12.8381565
4	0.2069891	0.2971095	0.4844186	0.7107230	9.4877290	11.1432868	13.2767041	14.8602590
5	0.4117419	0.5542981	0.8312116	1.1454762	11.0704977	12.8325020	15.0862725	16.7496023
6	0.6757268	0.8720903	1.2373442	1.6353829	12.5915872	14.4493753	16.8118938	18.5475842
7	0.9892557	1.2390423	1.6898692	2.1673499	14.0671404	16.0127643	18.4753069	20.2777399
8	1.3444131	1.6464974	2.1797307	2.7326368	15.5073131	17.5345461	20.0902350	21.9549550
9	1.7349329	2.0879007	2.7003895	3.3251128	16.9189776	19.0227678	21.6659943	23.5893508
10	2.1558565	2.5582122	3.2469728	3.9402991	18.3070381	20.4831774	23.2092512	25.1881796
11	2.6032219	3.0534841	3.8157483	4.5748131	19.6751376	21.9200493	24.7249703	26.7568489
12	3.0738236	3.5705690	4.4037885	5.2260295	21.0260698	23.3366642	26.2169673	28.2995188
13	3.5650346	4.1069155	5.0087505	5.8918643	22.3620325	24.7356049	27.6882496	29.8194712
14	4.0746750	4.6604251	5.6287261	6.5706314	23.6847913	26.1189480	29.1412377	31.3193496
15	4.6009156	5.2293489	6.2621378	7.2609439	24.9957901	27.4883929	30.5779142	32.8013206
16	5.1422054	5.8122125	6.9076644	7.9616456	26.2962276	28.8453507	31.9999269	34.2671865
17	5.6972171	6.4077598	7.5641864	8.6717602	27.5871116	30.1910091	33.4086636	35.7184657
18	6.2648047	7.0149109	8.2307462	9.3904551	28.8692994	31.5263784	34.8053057	37.1564515
19	6.8439714	7.6327296	8.9065165	10.1170131	30.1435272	32.8523269	36.1908691	38.5822566
20	7.4338443	8.2603983	9.5907774	10.8508114	31.4104328	34.1696069	37.5662348	39.9968463
21	8.0336534	8.8971979	10.2828978	11.5913052	32.6705733	35.4788759	38.9321727	41.4010648
22	8.6427164	9.5424923	10.9823207	12.3380146	33.9244385	36.7807121	40.2893604	42.7956550
23	9.2604248	10.1957156	11.6885519	13.0905142	35.1724616	38.0756273	41.6383981	44.1812752
24	9.8862335	10.8563615	12.4011502	13.8484250	36.4150285	39.3640770	42.9798201	45.5585119
25	10.5196521	11.5239754	13.1197200	14.6114076	37.6524841	40.6464691	44.3141049	46.9278902
26	11.1602374	12.1981469	13.8439050	15.3791566	38.8851387	41.9231701	45.6416827	48.2898823
27	11.8075874	12.8785044	14.5733827	16.1513958	40.1132721	43.1945110	46.9629421	49.6449153
28	12.4613359	13.5647098	15.3078606	16.9278750	41.3371382	44.4607918	48.2782358	50.9933763
29	13.1211489	14.2564546	16.0470717	17.7083662	42.5569678	45.7222858	49.5878845	52.3356178
30	13.7867199	14.9534565	16.7907723	18.4926610	43.7729718	46.9792422	50.8921813	53.6719619
35	17.1918203	18.5089262	20.5693766	22.4650152	49.8018496	53.2033485	57.3420734	60.2747709
40	20.7065353	22.1642613	24.4330392	26.5093032	55.7584793	59.3417071	63.6907398	66.7659618
45	24.3110142	25.9012692	28.3661523	30.6122591	61.6562334	65.4101590	69.9568321	73.1660608
50	27.9907489	29.7066827	32.3573637	34.7642517	67.5048065	71.4201952	76.1538912	79.4899785
55	31.7347575	33.5704753	36.3981111	38.9580265	73.3114930	77.3804656	82.2921168	85.7489516
60	35.5344911	37.4848515	40.4817480	43.1879585	79.0819445	83.2976749	88.3794189	91.9516982
65	39.3831408	41.4436091	44.6029925	47.4495811	84.8206455	89.1771450	94.4220790	98.1051438
70	43.2751795	45.4417173	48.7575648	51.7392780	90.5312254	95.0231842	100.4251842	104.2148988
75	47.2060477	49.4750289	52.9419398	56.0540723	96.2166708	100.8393384	106.3929229	110.2855834
80	51.1719319	53.5400773	57.1531729	60.3914784	101.8794740	106.6285677	112.3287925	116.3210565
85	55.1696043	57.6339298	61.3887746	64.7493958	107.5217410	112.3933736	118.2357493	122.3245807
90	59.1963042	61.7540790	65.6466176	69.1260304	113.1452701	118.1358926	124.1163187	128.2989436
95	63.2496485	65.8983615	69.9248671	73.5198352	118.7516118	123.8579666	129.9726787	134.2465495
100	67.3275633	70.0648949	74.2219275	77.9294652	124.3421134	129.5611972	135.8067232	140.1694894

6.3. Examples

Ex. 6.1. Generate $m = 50$ random samples of size $n = 10$ from the normal distribution $\mathrm{N}(1, 2)$. Construct a confidence interval for the mean for each sample with the 95% confidence level and display these confidence intervals on a single graph. How many of them should cover the true mean $\mu = 1$?

Solution.
For a sample $\mathbf{X} = (X_1, \ldots, X_n)$ from a normal distribution $N(\mu, \sigma)$, with an unknown parameter μ, and unknown σ, the $(1-\alpha)100\%$ confidence interval for μ is defined by (6.2).

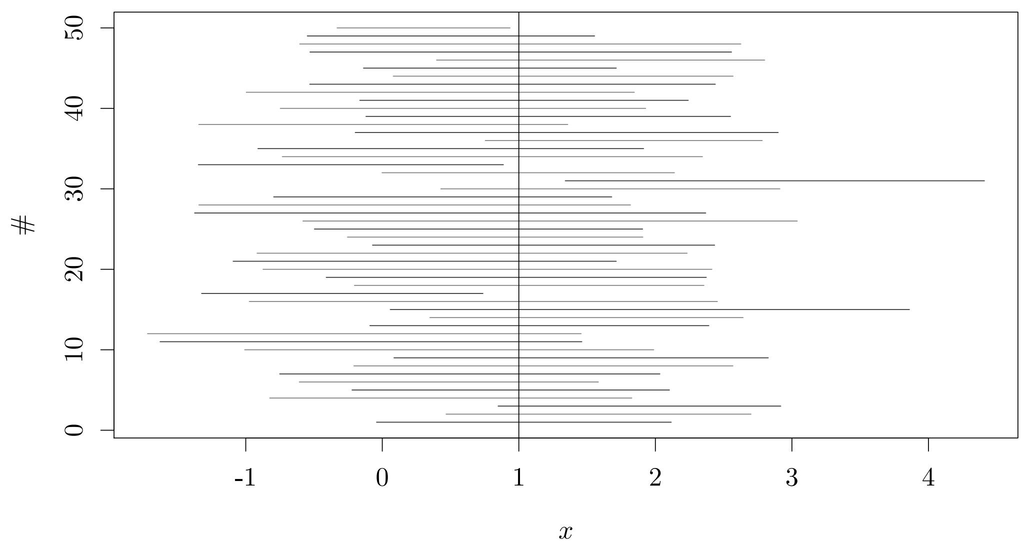
Let us generate m samples of size n from the normal distribution $\mathrm{N}(1, 2)$ and create 2 vectors: `mn` containing the means, $\overline{\mathbf{X}}$, and `d` containing the values of $s_{\mathbf{X}}/\sqrt{n}$ for each sample:

```
m <- 50
n <- 10
mu <- 1
sigma <- 2

results <- replicate(m, {
    X <- rnorm(n, mu, sigma)
    c(mean(X), sd(X)/sqrt(n))
})
mn <- results[1, ]
d <- results[2, ]
```

Now let us create the confidence intervals and plot them in one figure:

```
alpha <- 0.05; # one minus confidence level
q <- qt(1-alpha/2,n-1); # Student's t quantile
sum((mu >= mn-q*d) & (mu <= mn+q*d)) / m; # obsrvd frac. of intervals covering mu

## [1] 0.92
```

```
matplot(rbind(mn - q * d, mn + q * d), rbind(1:m, 1:m), type = "l", lty = 1,
    col = c("gray20", "gray50"), xlab = "$x$", ylab = "\\#")
abline(v = mu)
```



⊡

Ex. 6.2. Generate $m = 10000$ random samples of size $n = 10$ from the standard normal distribution. For each sample, construct a 95% confidence interval for the mean. Compare the fraction of intervals covering the true mean with the assumed confidence level.

Solution.

```
m <- 10000;      # the number of experiments
n <- 10;         # sample size
alpha <- 0.05;   # one minus confidence level
q <- qt(0.975,9) # Student's t quantile
```

We generate the samples and "check" the confidence level:

```
Cover <- replicate(m,
{
    X  <- rnorm(n, 0, 1); # N(0,1)
    mn <- mean(X);
    s  <- sd(X);
    (mn-q*s/sqrt(n) < 0) && (mn+q*s/sqrt(n) > 0) # result:
                 #  TRUE means that $\mu=0$ falls into the confidence level
});
Cover[1:10] # first 10 results
## [1]  TRUE FALSE  TRUE FALSE  TRUE  TRUE  TRUE  TRUE  TRUE  TRUE
```

The observed fraction of confidence intervals covering $\mu = 0$:

```
sum(Cover)/m
## [1] 0.9516
```

⊡

Ex. 6.3. Generate $m = 1000$ random samples of size $n = 50$ from the Bernoulli distribution with probability of success $p = 0.2$. Calculate and compare the average width and the approximate coverage of known 95% confidence intervals for p.

Solution.

```
m <- 1000;      # number of experiments
n <- 50;         # sample size
alpha <- 0.05;   # one minus confidence level
p <- 0.2;        # probability of success
```

We generate the samples and "check" the confidence level:

```
Results <- replicate(m,
{
   # generate a random number of successes in n Bernoulli trials:
    k  <- rbinom(1, n, p);

    phat <- k/n;

    ci1 <- c(phat-qnorm(1-alpha*0.5)*sqrt(phat*(1-phat)/n),
             phat+qnorm(1-alpha*0.5)*sqrt(phat*(1-phat)/n));
    ci2 <- binom.test(k, n, conf.level=1-alpha)$conf.int;
    ci3 <- prop.test (k, n, conf.level=1-alpha)$conf.int

    c(
       ci1[2]-ci1[1], # width 1
       ci2[2]-ci2[1], # width 2
```

```
        ci3[2]-ci3[1], # width 3
        ci1[1] < p && p < ci1[2],
        ci2[1] < p && p < ci2[2],
        ci3[1] < p && p < ci3[2]
      );
});

Results[,1:5] # preview some results
##         [,1]   [,2]   [,3]   [,4]   [,5]
## [1,] 0.2130 0.2489 0.2032 0.2130 0.2130
## [2,] 0.2286 0.2626 0.2194 0.2286 0.2286
## [3,] 0.2287 0.2604 0.2202 0.2287 0.2287
## [4,] 1.0000 1.0000 1.0000 1.0000 1.0000
## [5,] 1.0000 1.0000 1.0000 1.0000 1.0000
## [6,] 1.0000 1.0000 1.0000 1.0000 1.0000
```

Average widths:

```
mean(Results[1, ])
## [1] 0.2171
mean(Results[2, ])
## [1] 0.2326
mean(Results[3, ])
## [1] 0.2325
```

The observed fractions of confidence intervals covering p:

```
mean(Results[4, ])
## [1] 0.937
mean(Results[5, ])
## [1] 0.967
mean(Results[6, ])
## [1] 0.976
```

⊡

Ex. 6.4. A random sample of 50 college textbooks gave a mean price of \$28.40. It is known that the standard deviation of the prices of all college textbooks is \$4.75.

1. Construct a 99% confidence interval for the mean price of all college textbooks.
2. How many (at least) additional book prices should be considered to obtain a 99% confidence interval of width less than \$2?

Assume a normal population.

Solution.
For a sample $\mathbf{X} = (X_1, \ldots, X_n)$ from a normal distribution $\mathrm{N}(\mu, \sigma)$, with known σ, the $(1-\alpha)$-confidence interval for μ is given by (6.1).

Here we have: $\sigma = 4.75$, $n = 50$, $\bar{\mathbf{X}} = 28.4$, $1-\alpha = 0.99$. Hence the endpoints of the confidence interval for μ are the following:

```
28.4 - qnorm(0.995) * 4.75/sqrt(50)  # lower bound
## [1] 26.67
28.4 + qnorm(0.995) * 4.75/sqrt(50)  # upper bound
## [1] 30.13
```

Now we require that the 99% confidence interval's width be <$2:

$$2z_{0.995}\frac{\sigma}{\sqrt{n}} < 2.$$

Solving the above inequality for n gives:

$$n > (z_{0.995}\sigma)^2$$

```
(nmin <- (qnorm(0.995) * 4.75)^2)
## [1] 149.7
ceiling(nmin)
## [1] 150
```

Thus, the minimal sample size should be $n_{\min} = 150$ and hence 100 additional textbooks should be considered.

⊡

Ex. 6.5. The following data resulted from 18 independent measurements of the melting point of lead (in °C):

$$\begin{aligned}&330.0, 322.0, 345.0, 328.6, 331.0, 342.0,\\&342.4, 340.4, 329.7, 334.0, 326.5, 325.8,\\&337.5, 327.3, 322.6, 341.0, 340.0, 333.0.\end{aligned}$$

We may assume that the measurements can be regarded as constituting an i.i.d. normal sample which expected value is the true melting point of lead. Determine 95% two-sided confidence intervals for the mean and for standard deviation of the melting point of lead.

Also, calculate a 99% one-sided confidence interval for the mean.

Solution.
$(X_1, \ldots, X_n)$ is a sample of random variables from a normal distribution $\mathrm{N}(\mu, \sigma)$ with unknown parameters μ and σ. Therefore, to construct a confidence interval for the expected value μ, we can use the `t.test()` function.

```
x <- c(330, 322, 345, 328.6, 331, 342, 342.4, 340.4, 329.7, 334, 326.5, 325.8,
    337.5, 327.3, 322.6, 341, 340, 333)
mean(x)
## [1] 333.3
t.test(x, conf.level = 0.95)$conf.int
## [1] 329.6 336.9
## attr(,"conf.level")
## [1] 0.95
```

Compare this result with the interval calculated using (6.2):

```
mean(x) - qt(0.975, length(x) - 1) * sd(x)/sqrt(length(x))
## [1] 329.6
mean(x) + qt(0.975, length(x) - 1) * sd(x)/sqrt(length(x))
## [1] 336.9
```

The $(1-\alpha)100\%$ confidence interval for σ^2 has the form (6.5). Hence, the lower confidence limit (LCL) for a confidence interval for σ is equal to:

```
sqrt((length(x) - 1) * var(x)/qchisq(0.975, (length(x) - 1)))
## [1] 5.46
```

and the upper limit (UCL) is equal to:

```
sqrt((length(x) - 1) * var(x)/qchisq(0.025, (length(x) - 1)))
## [1] 10.91
```

99% one-sided confidence intervals for the mean can be obtained by:

```
t.test(x, conf.level = 0.99, alternative = "less")$conf.int  # right-sided ci
## [1]  -Inf 337.7
## attr(,"conf.level")
## [1] 0.99
t.test(x, conf.level = 0.99, alternative = "greater")$conf.int  #  left-sided ci
## [1] 328.9   Inf
## attr(,"conf.level")
## [1] 0.99
```

or (note that we do not put 0.995 here!):

```
mean(x)+qt(0.99, length(x)-1)*sd(x)/sqrt(length(x)); # UCL for the right-sided ci
## [1] 337.7
mean(x)-qt(0.99, length(x)-1)*sd(x)/sqrt(length(x)); # LCL for the left-sided ci
## [1] 328.9
```

Therefore, for example, the right-sided 99% confidence interval for μ is: $(-\infty, 337.6691]$.
⊡

Ex. 6.6. In a *Time* magazine poll of 1014 adults, 578 said it is bad for children that women do work out of home (Time, June 22, 1987). Construct a 95% confidence interval for the proportion of all adults who hold this view.

Solution.
For a (large) sample $(X_1, \ldots, X_n)$ from a Bern(p) distribution, the approximate $(1-\alpha)$ confidence interval for p is given by (6.7).

We have:

```
p <- 578/1014
n <- 1014
p + c(-1, 1) * qnorm(0.975) * sqrt(p * (1 - p)/n)
## [1] 0.5395 0.6005
```

To construct the approximate confidence interval for p we can also use the `prop.test` function:

```
prop.test(578, 1014, conf.level = 0.95)$conf.int
## [1] 0.5388 0.6007
## attr(,"conf.level")
## [1] 0.95
```

Compare these results with the "exact" confidence interval:

```
binom.test(578, 1014, conf.level = 0.95)$conf.int
## [1] 0.5389 0.6007
## attr(,"conf.level")
## [1] 0.95
```

⊡

Ex. 6.7. Mary rolled her (possibly unfair) dice 12 times and found that "⚂" appeared on the top face 3 times. Construct a 95% confidence interval for the probability of obtaining the number 3 on a single roll.

Solution.
We have a small sample $(X_1, \ldots, X_n)$ from Bern (p) distribution with an unknown parameter p. To construct a confidence interval for p we use the function `binom.test`:

```
binom.test(3, 12, conf.level = 0.95)$conf.int

## [1] 0.05486 0.57186
## attr(,"conf.level")
## [1] 0.95
```

$\boxdot$