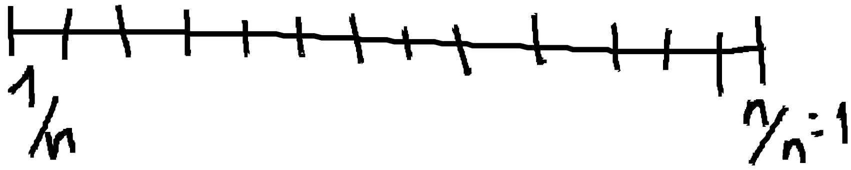
np.arange(1,n+1)/n :=



Stochastic Simulation
Professor: Prof. Fabio Nobile
Assistant: Eliott Van Dieren

Week of 2025-09-08 to 2025-09-12

Lab 1 of Thursday 11th September 2025

Exercice 1.

Consider `Scipy`'s default **uniform** random number generator (RNG) `uniform` within the Statistics module `scipy.stats` and use it to generate a sequence of numbers $U_1, U_2, \dots, U_n$. See this for the module documentation and see this for the full list of available continuous distributions under `scipy.stats`. Consider different values of n, for example $n = 25, 100, 10^3, 10^5$, and address the following points:

1) Plot the cumulative distribution function (CDF) of the theorized $\mathcal{U}(0,1)$ distribution together with the empirical CDF of the data. Furthermore, produce a Q-Q plot of the data. Use both plots to assess the quality of the sequence with respect to the theorized $\mathcal{U}(0,1)$ distribution. Describe your observations.

2) Implement the Kolmogorov–Smirnov test to ascertain whether the empirical CDF of $U_1, U_2, \dots, U_n$ matches the theoretical CDF of the $\mathcal{U}(0,1)$ distribution at level $\alpha = 0.1$, i.e., we reject the null hypothesis H_0 at level $\alpha > 0$ that the sample $U_1, \dots, U_n \overset{\text{iid}}{\sim} \mathcal{U}(0,1)$ if $\sqrt{n} D_n > K_{\alpha,n}$, where $D_n = \sup_{x\in\mathbb{R}} |\hat{F}(x) - F(x)|$, and $K_{\alpha,n}$ is such that $\mathbb{P}(\sqrt{n} D_n > K_{\alpha,n}) < \alpha$.

   It is known that the appropriately scaled test statistic $D_n = \sup_{x\in\mathbb{R}} |\hat{F}_n(x) - F(x)|$ converges in distribution to a Kolmogorov random variable K independently of F, where $\mathbb{P}(K \leqslant x) = 1 + 2\sum_{j=1}^{\infty} (-1)^j e^{-2j^2x^2}$, $x > 0$. This asymptotic result can then be used to compute the required $1-\alpha$ quantiles, $K_{\alpha,n} \simeq K_{\alpha,\infty}$. It is however also possible to characterize the distribution of D_n directly, which is useful for small values of n. Table 1 presents some of these pre-asymptotic $1-\alpha$ quantiles $K_{\alpha,n}$.

3) Implement the χ^2 goodness of fit test to ascertain whether the sequence $U_1, U_2, \dots, U_n$ is equidistributed. A description of such method can be found on section 8.7.4 of *Handbook of Monte Carlo Methods* (See also page 10 of the lecture notes).

   Hint. you can use the `ppf` function of the `scipy.stats.chi2` class to compute quantiles of a χ^2 distribution.

4) Repeat the tests in points 2. and 3. for different values of α. What do you observe? Explain your findings.

Exercice 2.

Implement the linear congruential generator (LCG)

$$X_k = (aX_{k-1} + b) \bmod m\ , \quad U_k := \frac{X_k}{m}\ ,$$

with $a = 3$, $b = 0$, and $m = 31$.

	α			
n	0.20	0.10	0.05	0.01
1	0.90	0.95	0.98	0.99
2	0.96	1.10	1.19	1.32
3	0.97	1.11	1.23	1.44
4	0.98	1.12	1.24	1.46
5	1.01	1.14	1.25	1.50
6	1.00	1.15	1.27	1.52
7	1.01	1.16	1.30	1.53
8	1.02	1.16	1.30	1.53
9	1.02	1.17	1.29	1.53
10	1.01	1.17	1.30	1.55
11	1.03	1.16	1.29	1.56
12	1.04	1.18	1.32	1.56
15	1.05	1.16	1.32	1.55
20	1.03	1.16	1.30	1.57
30	1.04	1.20	1.31	1.59
35	1.06	1.24	1.36	1.60
40	1.08	1.20	1.33	1.58
45	1.07	1.21	1.34	1.61
$n > 45$	1.07	1.22	1.36	1.63

Table 1: Critical values for the Kolmogorov–Smirnov test statistic $D_n = \sup_{x\in\mathbb{R}}\left|\widehat{F}_n(x) - F(x)\right|$. The tabulated values $K_{n,\alpha}$ are such that $\mathbb{P}(\sqrt{n}D_n \geqslant K_{n,\alpha}) = \alpha$.

1) Use your LCG procedure to generate a sequence $U_1, U_2, \dots, U_n$ and repeat Exercise 1. Discuss your results.

2) Explain why one would expect that the Serial test (with $d = 2$, say) is an appropriate test to scrutinize the LCG. Support your explanation by applying the Serial test at level $\alpha = 0.1$ to sequences (for various values of n) from both the LCG and from the default `scipy.stats` RNG `uniform`.

3) Implement the Gap test. Apply the test to both a sequence obtained from the default `scipy.stats` RNG `uniform` and to a sequence generated by the LCG. What do you observe?

Note on built-in functions

Many of the functions that need to be implemented in this Lab already exist as `Python`built-in functions. For example, the empirical CDF can be conveniently plotted using the `ECDF` function that is available in the `statsmodels` package. The Kolmogorov-Smirnov test and χ^2 test are also available. There is, of course, very little reason to reinvent the wheel and we strongly encourage you to use these built-in functions in future Labs, if not stated otherwise. However, before naively relying on built-in functions, it is important to understand the underlying mathematical procedure.