

NOM, NOTATION	LOI DISCRÈTE	ESPÉRANCE, VARIANCE	MODÈLE, PROPRIÉTÉS
Loi uniforme finie $X \hookrightarrow \mathcal{U}(\llbracket 1, n \rrbracket)$, $n \in \mathbf{N}^*$	$X(\Omega) = \llbracket 1, n \rrbracket$ $\forall k \in \llbracket 1, n \rrbracket, \mathbf{P}(X = k) = \frac{1}{n}$	$\mathbf{E}(X) = \frac{n+1}{2}$ $\mathbf{V}(X)$ hors-programme	piocher dans une urne contenant n boules numérotées de 1 à n ou dans un paquet de n cartes, lancer un dé à n faces. PYTHON : <code>rd.randint(1,n)</code>
Loi de Bernoulli $X \hookrightarrow \mathcal{B}(p)$, $p \in]0, 1[$	$X(\Omega) = \{0, 1\}$ $\mathbf{P}(X = 1) = p$ $\mathbf{P}(X = 0) = 1 - p = q$	$\mathbf{E}(X) = p$ $\mathbf{V}(X) = pq$	expérience à deux issues : succès (1) ou échec (0) PYTHON : <code>rd.random() < p</code>
Loi binomiale $X \hookrightarrow \mathcal{B}(n, p)$ $n \in \mathbf{N}^*, p \in]0, 1[$	$X(\Omega) = \llbracket 0, n \rrbracket$ $\forall k \in \llbracket 0, n \rrbracket,$ $\mathbf{P}(X = k) = \binom{n}{k} p^k q^{n-k}$	$\mathbf{E}(X) = np$ $\mathbf{V}(X) = npq$	nombre de succès dans la répétition de n épreuves de Bernoulli, indépendantes , et de même paramètre p . Loi des tirages avec remise. PYTHON : <code>sum[rd.random() < p for _ in range(n)]</code> Stabilité : $X_i \hookrightarrow \mathcal{B}(n_i, p)$ indépendantes $\implies \sum X_i \hookrightarrow \mathcal{B}(\sum n_i, p)$
Loi géométrique $X \hookrightarrow \mathcal{G}(p)$ $p \in]0, 1[$	$X(\Omega) = \mathbf{N}^*$ $\forall k \in \mathbf{N}^*,$ $\mathbf{P}(X = k) = p \times q^{k-1}$	$\mathbf{E}(X) = \frac{1}{p}$ $\mathbf{V}(X) = \frac{q}{p^2}$	rang d'apparition du premier succès dans la répétition d'une infinité d'épreuves de Bernoulli, indépendantes , et de même paramètre p . PYTHON : <code>rang = 1</code> <code>while rd.random() > p :</code> <code> rang += 1</code> <code>return rang</code> Loi sans mémoire : $\forall s, t \in \mathbf{N}, \mathbf{P}_{[X>s]}(X > s+t) = \mathbf{P}(X > t)$
Loi de Poisson $X \hookrightarrow \mathcal{P}(\lambda)$ $\lambda \in \mathbf{R}_+^*$	$X(\Omega) = \mathbf{N}$ $\forall k \in \mathbf{N},$ $\mathbf{P}(X = k) = \frac{\lambda^k}{k!} e^{-\lambda}$	$\mathbf{E}(X) = \lambda$ $\mathbf{V}(X) = \lambda$	loi des événements rares Stabilité : $X_i \hookrightarrow \mathcal{P}(\lambda_i)$ indépendantes $\implies \sum X_i \hookrightarrow \mathcal{P}(\sum \lambda_i)$ $n \geq 30, p \leq 0,1 : \mathcal{B}(n, p) \approx \mathcal{P}(np)$

NOM, NOTATION	LOI À DENSITÉ	ESPÉRANCE, VARIANCE	MODÈLE, PROPRIÉTÉS
Loi uniforme sur $[a, b]$ $X \hookrightarrow \mathcal{U}([a, b])$ $a, b \in \mathbf{R}, a < b$	$X(\Omega) = [a, b]$ densité : $f_X(t) = \begin{cases} \frac{1}{b-a} & \text{si } t \in [a, b] \\ 0 & \text{sinon} \end{cases}$ répartition : $F_X(x) = \begin{cases} 0 & \text{si } x < a \\ \frac{x-a}{b-a} & \text{si } x \in [a, b] \\ 1 & \text{si } x > b \end{cases}$	$\mathbf{E}(X) = \frac{a+b}{2}$ $\mathbf{V}(X) = \frac{(b-a)^2}{12}$	Choix équiprobable dans un intervalle réel $[a, b]$ PYTHON : <code>rd.random()*(b-a)+a</code>
Loi exponentielle $X \hookrightarrow \mathcal{E}(\lambda)$ $\lambda \in \mathbf{R}_+^*$	$X(\Omega) = \mathbf{R}_+$ densité : $f_X(t) = \begin{cases} \lambda e^{-\lambda t} & \text{si } t \geq 0 \\ 0 & \text{sinon} \end{cases}$ répartition : $F_X(x) = \begin{cases} 0 & \text{si } x < 0 \\ 1 - e^{-\lambda x} & \text{si } x \geq 0 \end{cases}$	$\mathbf{E}(X) = \frac{1}{\lambda}$ $\mathbf{V}(X) = \frac{1}{\lambda^2}$	Durée de vie d'une particule radioactive. Invariance temporelle : $\forall s, t \geq 0, \mathbf{P}_{[X \geq s]}(X \geq s+t) = \mathbf{P}(X \geq t)$ PYTHON : <code>-log(rd.random())/lamb</code>
Loi normale centrée réduite $X \hookrightarrow \mathcal{N}(0, 1)$	$X(\Omega) = \mathbf{R}$ densité : $\forall t \in \mathbf{R}, \varphi(t) = \frac{1}{\sqrt{2\pi}} e^{-\frac{t^2}{2}}$ répartition : $\forall x \in \mathbf{R}, \Phi(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{t^2}{2}} dt$	$\mathbf{E}(X) = 0$ $\mathbf{V}(X) = 1$	Mouvement brownien... Utilisée dans le TCL PYTHON : <code>rd.gauss(0,1)</code>
Loi normale $X \hookrightarrow \mathcal{N}(\mu, \sigma^2)$ $\mu \in \mathbf{R}, \sigma \in \mathbf{R}_+^*$	$X(\Omega) = \mathbf{R}$ densité : $\forall t \in \mathbf{R}, f_X(t) = \frac{1}{\sigma} \times \varphi\left(\frac{t-\mu}{\sigma}\right)$ répartition : $\forall x \in \mathbf{R}, F_X(x) = \Phi\left(\frac{x-\mu}{\sigma}\right)$	$\mathbf{E}(X) = \mu$ $\mathbf{V}(X) = \sigma$	Stabilité : si $X_i \hookrightarrow \mathcal{N}(\mu_i, \sigma_i^2)$ sont indépendantes, alors : $\sum X_i \hookrightarrow \mathcal{N}\left(\sum \mu_i, \sum \sigma_i^2\right)$ PYTHON : <code>sigma*rd.gauss(0,1) + mu</code> $n \geq 30, np \geq 5, nq \geq 5 : \mathcal{B}(n, p) \approx \mathcal{N}(np, npq)$