Discrete distributions	**Random variable (z)**	**Parameters**	**Moments**	**R functions**	**JAGS functions for likelihood of data (y)**	Conjugate relationship

Poisson
 $[z\|\lambda] =$
 $\frac{\lambda^z e^{-\lambda}}{z!}$
	Counts of things that occur randomly over time or space, e.g., the number of birds in a forest stand, the number of fish in a kilometer of river, the number of prey captured per minute.	λ, the mean number of occurrences per time or space
		 $\lambda = \mu$
			$\mu = \lambda$
			 $\sigma^2 = \lambda$
				`dpois(x, lambda, log = FALSE)`
				 `ppois(q, lambda)`
				 `qpois(p, lambda),`
				 `rpois(n, lambda)`
					`y[i] ~ dpois(lambda)`	$P(\lambda\|\mathbf{y}) =$
						 gamma $\left(\alpha + \sum_{i=1}^{n} y_i, \beta + n\right)$

Binomial
 $[z \mid \eta, \phi] =$
 $\binom{\eta}{z}\phi^z (1-\phi)^{\eta - z}$
 $\binom{\eta}{z} = \frac{\eta!}{z!(\eta-z)!}$
 $[z \mid \eta, \phi] \propto$
 $\phi^z (1-\phi)^{\eta-z}$
	Number of "successes" on a given number of trials, e.g., number of survivors in a sample of individuals, number of plots containing an exotic species from a sample, number of terrestrial pixels that are vegetated in an image.	η, the number of trials
		 ϕ, the probability of a success
		 $\phi = 1 - \sigma^2/\mu$
		 $\eta = \mu^2/(\mu - \sigma^2)$
			$\mu = \eta\phi$
			 $\sigma^2 = \eta\phi(1-\phi)$
				`dbinom(x, size, prob, log = FALSE)`
				 `pbinom(q, size, prob)`
				 `qbinom(p, size, prob)`
				 `rbinom(n, size, prob)`
					`y[i] ~ dbin(p,n)`	$P(p\|\mathbf{y}) =$
						 beta $(\alpha + y, \beta + n - y)$

Bernoulli
 $[z\|\phi] = \phi^z (1-\phi)^{1-z}$
	A special case of the binomial where the number of trials = 1 and the random variable can take on values 0 or 1. Widely used in survival analysis, occupancy models.	ϕ, the probability that the random variable = 1
		 $\phi = \mu$
		 $\phi = 1/2 + 1/2\sqrt{1 - 4\sigma^2}$
			$\mu = \phi$
			 $\sigma^2 = \phi(1-\phi)$
				`dbinom(x, size=1, prob, log = FALSE)`
				 `pbinom(q, size=1, prob)`
				 `qbinom(p, size=1, prob)`
				 `rbinom(n, size=1, prob)`
				 Note that size *must* = 1.
					`y[i]~dbern(p)`	

Negative binomial
 $[z\|\lambda, \kappa] =$
 $\frac{\Gamma(z+\kappa)}{\Gamma(\kappa)z!}\left(\frac{\kappa}{\kappa+\lambda}\right)^{\kappa}$
 $\times\left(\frac{\lambda}{\kappa+\lambda}\right)^{z}$
 (Read R help about alternative parameterization.)
	Counts of things occurring randomly over time or space, as with the Poisson. Includes dispersion parameter κ allowing the variance to exceed the mean.	λ, the mean number of occurrences per time or space.
		 κ, the dispersion parameter.
		 $\lambda = \mu$
		 $\kappa = \frac{\mu^2}{\sigma^2 - \mu}$
			$\mu = \lambda$
			 $\sigma^2 = \lambda + \frac{\lambda^2}{\kappa}$
				`dnbinom(x, size, mu )`
				 `pnbinom(q, size, mu)`
				 `qnbinom(p, mu)`
				 `rnbinom(n, size, mu)`
				 Size is the dispersion parameter, κ
					`y[i] ~ dnegbin(k / (k + lambda) , k)`
					 Uses alternative parameterization. The variable k is the dispersion parameter.	

Multinomial $[\mathbf{z} \mid \eta, \boldsymbol{\phi}] =$ $\eta! \prod_{i=1}^{k} \frac{\phi_i^{z_i}}{z_i}$	Counts that fall into $k > 2$ categories, e.g., number of individuals in age classes, number of pixels in different landscape categories, number of species in trophic categories in a sample from a food web.	$\mathbf{z}$ a vector giving the number of counts in each category, ϕ a vector of the probabilities of occurrence in each category, $\sum_{i=1}^{k} \phi_i = 1$, $\sum_{i=1}^{k} z_i = \eta$	$\mu_i = \eta\phi_i$ $\sigma_i^2 =$ $\eta\phi_i(1-\phi_i)$	`rmultinom(n, size, prob)` `dmultinom(x, size, prob, log = FALSE)`	`y[i,]~dmulti(p[],n)`	

Continuous Distributions	**Random variable (z)**	**Parameters**	**Moments**	**R functions**	**JAGS function**	**Conjugate prior for**	**Vague Prior**

Normal
$[z\|\mu,\sigma^2] =$
$\frac{1}{\sigma\sqrt{2\pi}}e^{-\frac{(z-\mu)^2}{2\sigma^2}}$
	Continuously distributed quantities that can take on positive or negative values. Sums of things are normally distributed.	μ, σ^2	μ, σ^2	`dnorm(x, mean, sd, log = FALSE)`
				`pnorm(q, mean, sd)`
				`qnorm(p, mean, sd)`
				`rnorm(n, mean, sd)`
					`# tau = 1/sigma^2#`
					`#likelihood`
					`y[i]~dnorm(mu,tau)`
					`#prior`
					`theta ~ dnorm(mu,tau)`
						normal mean (with known variance)	`dnorm(0,1E-6)`
							#This is scale dependent.

Lognormal
$[z \mid \alpha, \beta]$
$\frac{1}{z\sqrt{2\pi\beta^2}}\,e^{-\frac{(\ln z-\alpha)^2}{2\beta^2}}$
	Continuously distributed quantities with non-negative values. Random variables that have the property that their logs are normally distributed. Thus if z is normally distributed then $\exp(z)$ is lognormally distributed. Products of things are lognormally distributed.	α, the mean of z on the log scale
		β, the standard deviation of z on the log scale
		$\alpha = \log(\text{median}(z))$
		$\alpha = \ln(\mu) - 1/2\,\ln\left(\frac{\sigma^2+\mu^2}{\mu^2}\right)$
		$\beta = \sqrt{\ln\left(\frac{\sigma^2+\mu^2}{\mu^2}\right)}$
			$\mu = e^{\alpha+\frac{\beta^2}{2}}$
			$\text{median}(z) = e^{\alpha}$
			$\sigma^2 = \left(e^{\beta^2}-1\right)e^{2\alpha+\beta^2}$
				`dlnorm(x, meanlog, sdlog)`
				`plnorm(q, meanlog, sdlog)`
				`qlnorm(p, meanlog, sdlog)`
				`rlnorm(n, meanlog, sdlog)`
					`#likelihood`
					`y[i]~dlnorm(alpha,tau)`
					`#prior`
					`theta~ dlnorm(alpha,tau)`		

Gamma
$[z\|\alpha,\beta] =$
$\frac{\beta^\alpha}{\Gamma(\alpha)}z^{\alpha-1}e^{-\beta z}$
$\Gamma(\alpha) =$
$\int_0^\infty t^{\alpha-1}e^{-t}\,dt\,.$
	The time required for a specified number of events to occur in a Poisson process. Any continuous quantity that is non-negative.	$\alpha =$ shape
		$\beta =$ rate
		$\alpha = \frac{\mu^2}{\sigma^2}$
		$\beta = \frac{\mu}{\sigma^2}$
		Note–be very careful about rate, defined as above, and scale $= \frac{1}{\beta}$.
			$\mu = \frac{\alpha}{\beta}$
			$\sigma^2 = \frac{\alpha}{\beta^2}$
				`dgamma(x, shape, rate, log = FALSE)`
				`pgamma(q, shape, rate)`
				`qgamma(p, shape, rate)`
				`rgamma(n, shape, rate)`
					`#likelihood`
					`y[i]~ dgamma(r,n)`
					`#prior`
					`theta~dgamma(r,n)`
						1) Poisson mean
						2) normal precision (1/variance)
						3) n parameter (rate) in the gamma distribution
							`dgamma(.001,.001)`

Beta
$[z\|\alpha,\beta] =$
$B\,z^{\alpha-1}(1-z)^{\beta-1}$
$B = \frac{\Gamma(\alpha+\beta)}{\Gamma(\alpha)\Gamma(\beta)}$
	Continuous random variables that can take on values between 0 and 1–any random variable that can be expressed as a proportion; e.g,survival, proportion of landscape invaded by exotic.	$\alpha = \frac{(\mu^2-\mu^3-\mu\sigma^2)}{\sigma^2}$
		$\beta = \frac{\mu-2\mu^2+\mu^3-\sigma^2+\mu\sigma^2}{\sigma^2}.$
			$\mu = \frac{\alpha}{\alpha+\beta}$
			$\sigma^2 = \frac{\alpha\beta}{(\alpha+\beta)^2(\alpha+\beta+1)}$
				`dbeta(x, shape1, shape2)`
				`pbeta(q, shape1, shape2 )`
				`qbeta(p, shape1, shape2 )`
				`rbeta(n, shape1, shape2)`
					`#likelihood`
					`y[i] ~ dbeta(alpha, beta)`
					`#prior`
					`theta ~ dbeta(alpha, beta)`
						p in binomial distribution	`dbeta(1,1)`

Dirichlet
$[\mathbf{z}\|\boldsymbol{\alpha}] = \Gamma\left(\sum_{i=1}^{k}\alpha_i\right)\times\frac{\prod_{j=1}^{k} z_j^{\alpha_j-1}}{\Gamma(\alpha_j)}$
	Vectors of >2 elements of continuous random variables that can take on values between 0 and 1 and that sum to one.	$\alpha_i = \mu_i\alpha_0$
		$\alpha_0 = \sum_{i=1}^{k}\alpha_i$
			$\mu_i = \frac{\alpha_i}{\sum_{i=1}^{k}\alpha_i}$
			$\sigma_i^2 = \frac{\alpha_i(\alpha_0-\alpha_i)}{\alpha_0^2(\alpha_0+1)},$
				`library(gtools)`
				`rdirichlet(n, alpha)`
				`ddirichlet(x, alpha)`
					#likelihood
					`y[]~ddrich(alpha[] )`
					`p[] ~ ddrich(alpha[] )`
					`y, alpha, p are vectors`
						vector **p** in multinomial distribution	`ddrich(1,1,1...1)`

Uniform
$[z\|\alpha,\beta] = \begin{cases}\frac{1}{\beta-\alpha} & \text{for } \alpha \le z \le \beta,\\ 0 & \text{for } z < \alpha \text{ or } z > \beta\end{cases}$
	Any real number.	$\alpha =$ lower limit
		$\beta =$ upper limit
		$\alpha = \mu - \sigma\sqrt{3}$
		$\beta = \mu + \sigma\sqrt{3}$
			$\mu = \frac{\alpha+\beta}{2}$
			$\sigma^2 = \frac{(\beta-\alpha)^2}{12}$
				`dunif(x, min, max,log = FALSE)`
				`punif(q, min, max)`
				`qunif(p, min max)`
				`runif(n, min, max)`
					`#prior`
					`theta~dunif(a,b)`
							a and b such that posterior is "more than entirely" between a and b