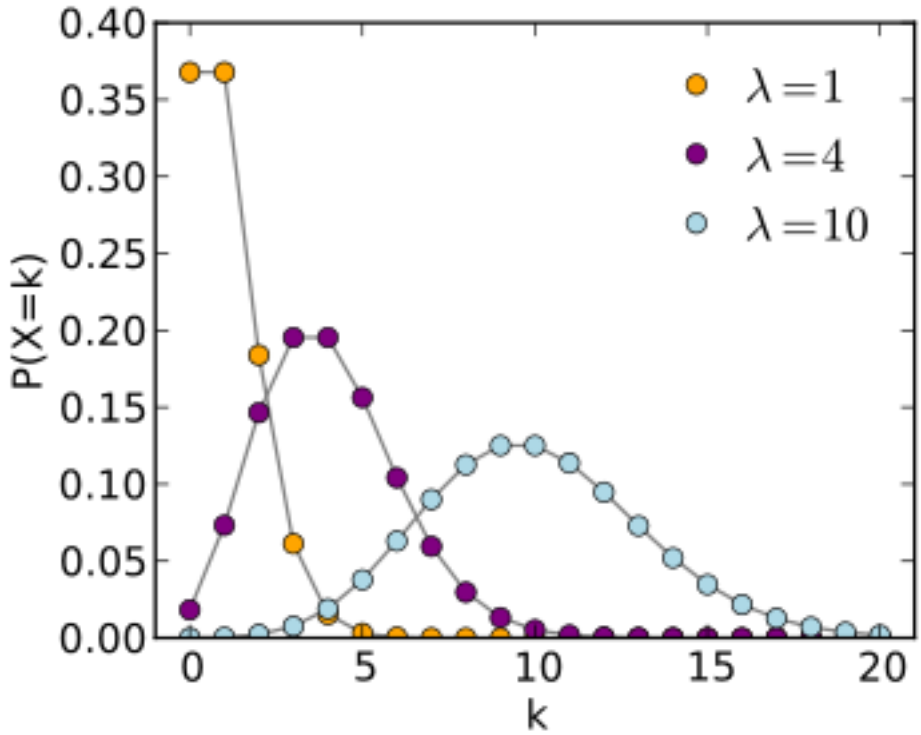
Distribution Manager
Lunar Game Assets

List of functions:

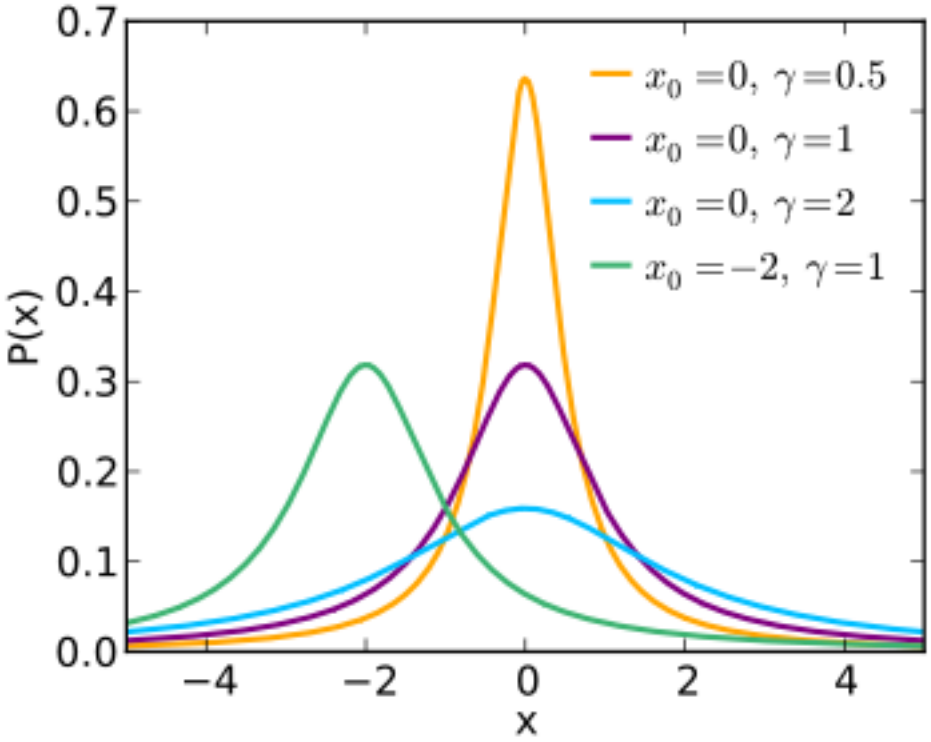
- int[] BinomialSample (float chance, int sampleLenght)
  chance - chance of success(value 1)
sampleLenght - length of sample array
Returns array with 1 or 0. Array length = sampleLenght

- **float[] PolinomialSample (float[] chances,float[] values,int sampleLenght)**
  Note: chances length should be same as values length
chances - array with chances for each value
values - array with values
Returns array filled with values numbers
Example: chances = {0.1,0.25,0.65} , values = {7,10,13}
You have shotgun that shoot 10 bullets at time, each bullet can deal
7 damage with 10% chance
10 damage with 25% chance
13 damage with 65 chance
float[] bulletsDamage = DistributionManager.PolinomialSample(chances,values,10)
Now you can change bullets damage.

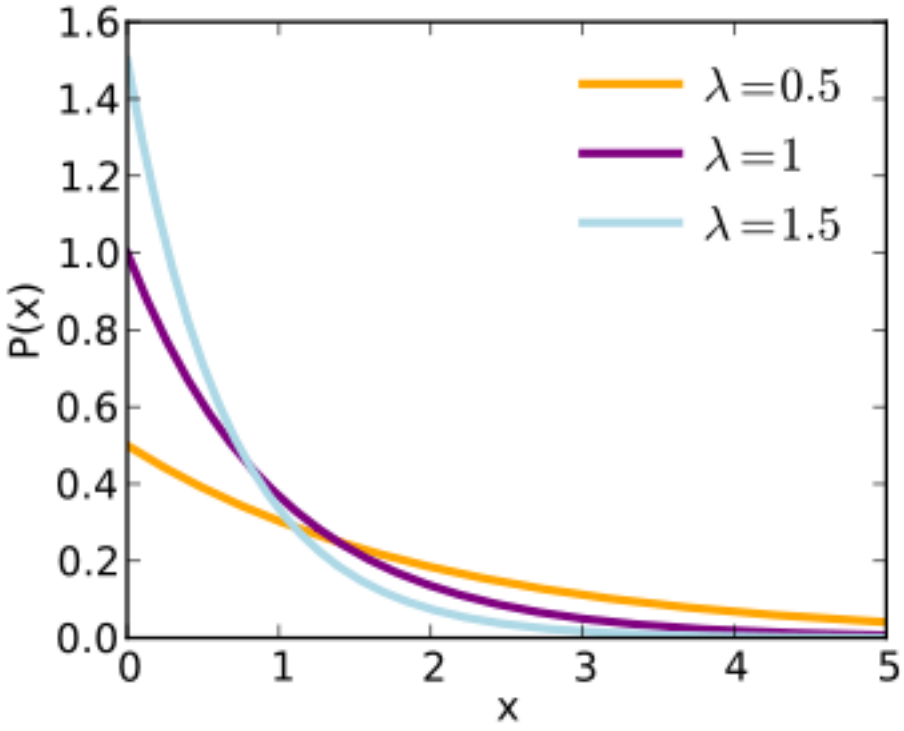
- **int[] PoissSample (float lambda, int sampleLenght)**
  Here is density graphic. P(x = k) - chance of x to be equal to k



- **float[] CauchySample (float location, float scale, int sampleLenght)**
  Density graphic, x_0 - location(“Head position”), gamma - scale

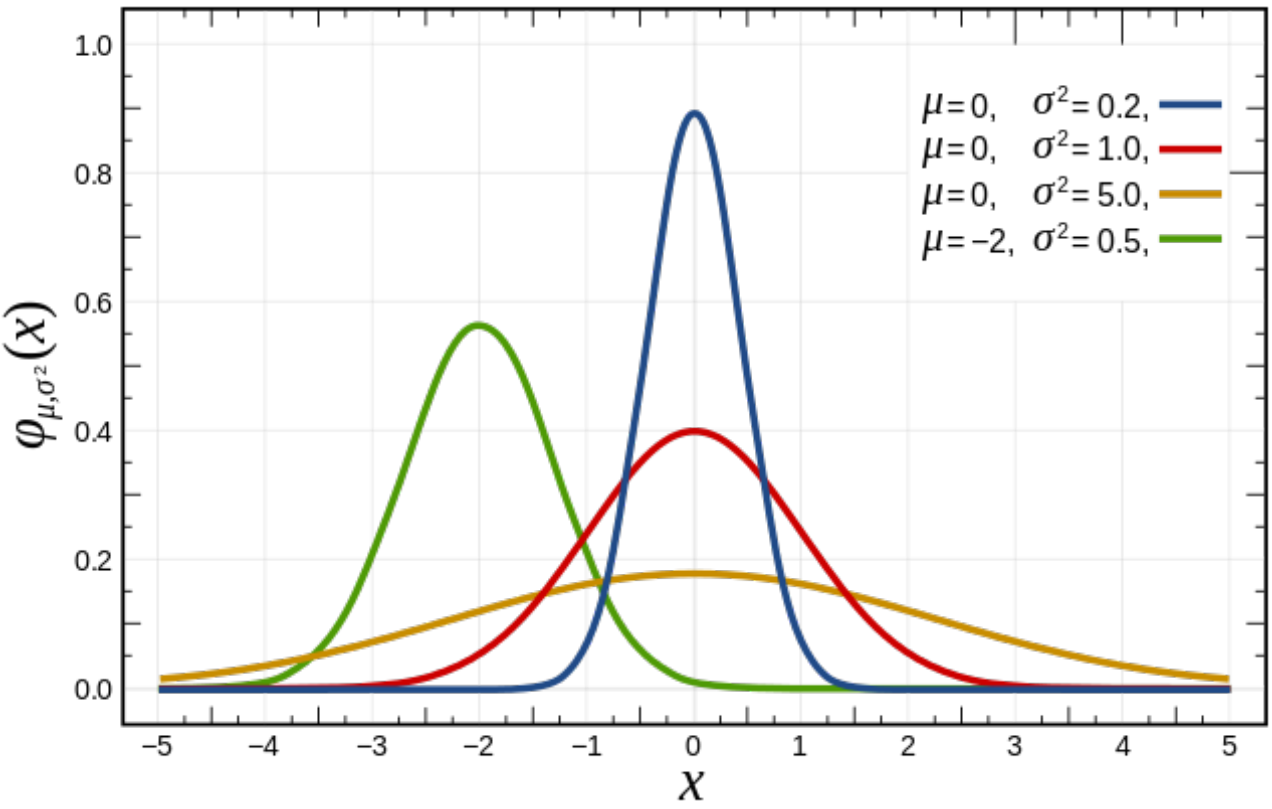


- **float[] ExponentialSample (float rate, int sampleLenght)**
  Density graphic, rate - lambda

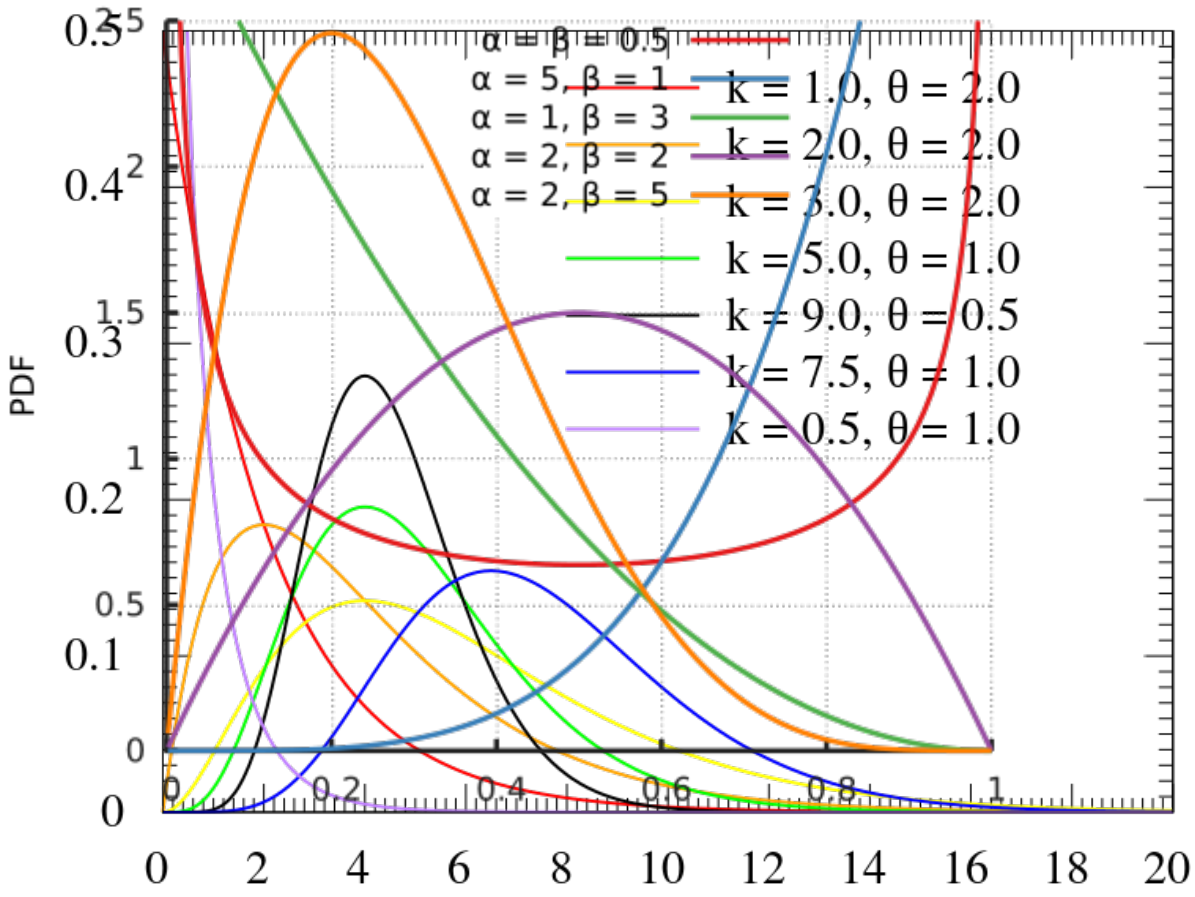


- **float[] NormalDistSample (float mean, float standartDevitation, int sampleLenght)**
  Sample from Normal(Gauss) distribution

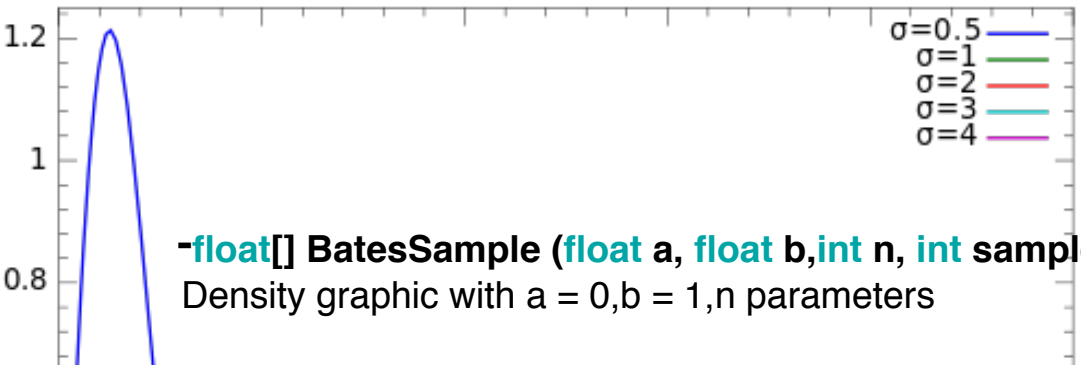
Density graphic mean - μ, standartDevitation - Sigma squared



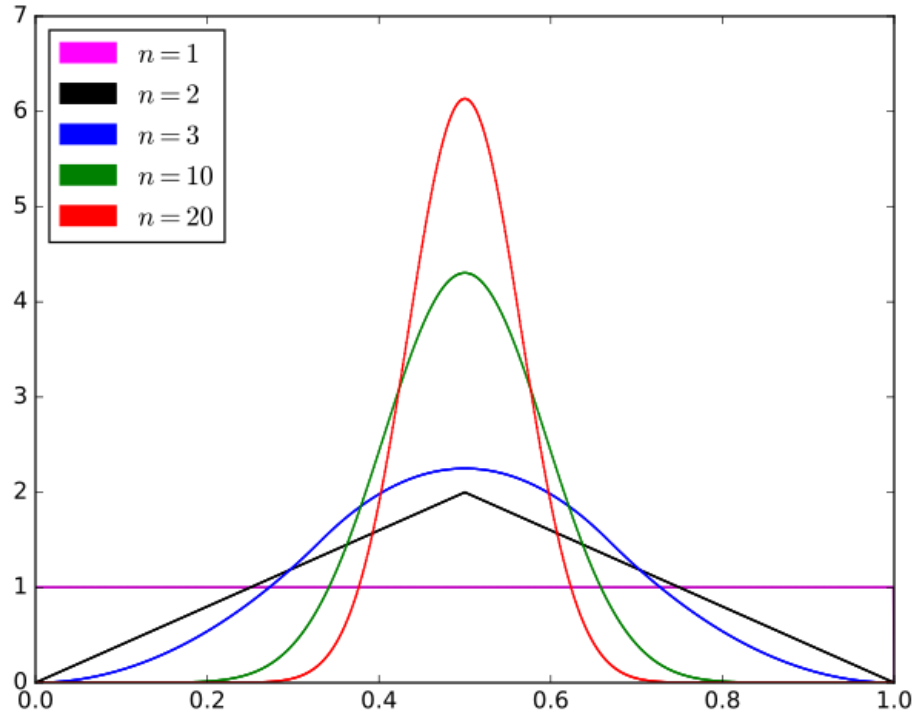
- **float[] GammaSample (float shape, float scale, int sampleLenght)**
  Density graphic shape - k, scale - θ
- **float[] BetaSample (float alpha, float beta, int sampleLenght)**
  Density graphic with (alpha,beta) parameters



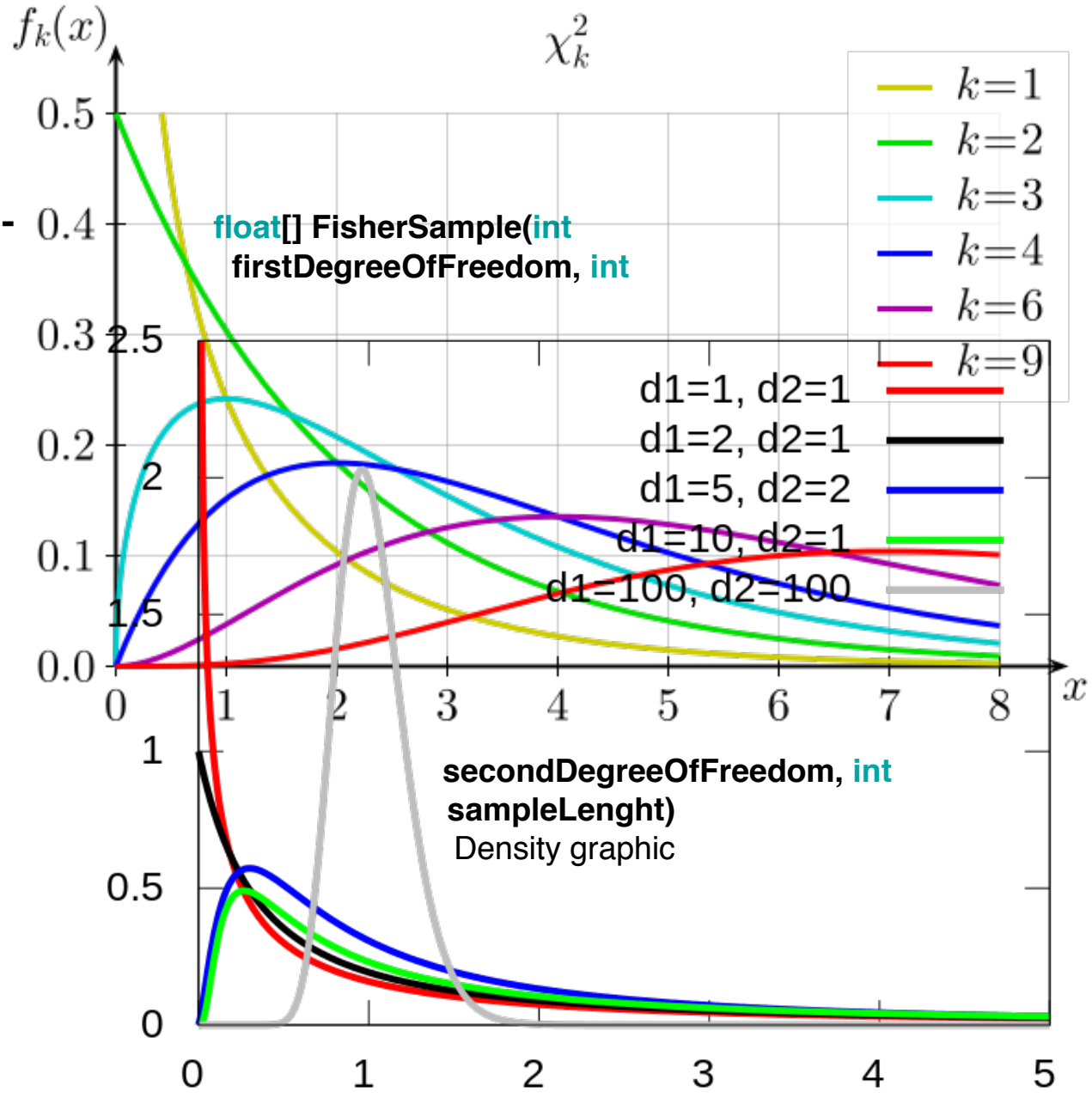
- **float[] RayleighSample (float scale, int sampleLenght)**
  Density graphic



- **float[] BatesSample (float a, float b,int n, int sampleLenght)**
  Density graphic with a = 0,b = 1,n parameters



- **float[] ChiSquareSample (int freedomDegree, int sampleLenght)**
  Density graphic with k = freedomDegree



- **float[] FisherSample(int firstDegreeOfFreedom, int secondDegreeOfFreedom, int sampleLenght)**
  Density graphic

– **float SampleMean (float[] sample)**
Returns sample mean (average value)

– **float SampleStandartDevitation (float[] sample)**
Returns sample Standard Devitation, formula below

$$\sigma = \sqrt{\sum_{i=1}^{N} p_i (x_i - \mu)^2}, \quad \text{where } \mu = \sum_{i=1}^{N} p_i x_i .$$

– **float SampleMedian (float[] sample)**
Returns sample median

– **float SampleModa (float[] sample)**
Returns sample moda