Random Walk in 1D

Preliminaries

Clear all symbols from previous evaluations to avoid conflicts

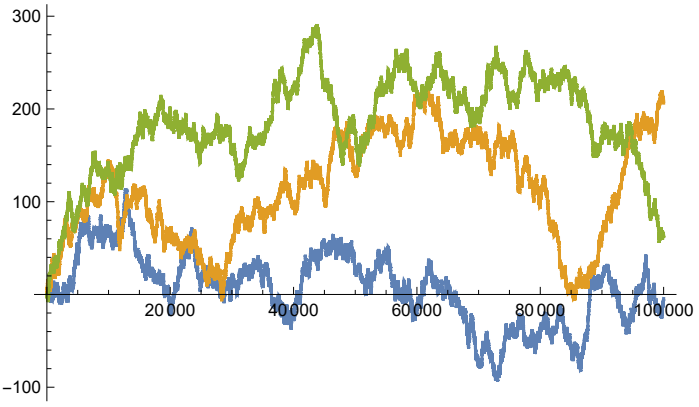
In[ ]:=
```
Clear["Global`*"]
```

Simple walk in 1D - arbitrary step of maximal length *d*

Start at origin and make 10000 random steps:

In[ ]:=
```
p = 3; (* number of walks *)
n = 100 000;(* number of steps *)
d = 1.0;(* maximal length of each step *)
X = ConstantArray[0.0, {p, n + 1}];
(* SeedRandom[2021] *)
Do[
  Do[
   X[[ip, in + 1]] = X[[ip, in]] + d * RandomReal[{-1.0, 1.0}],
   {in, 1, n}
  ],
  {ip, 1, p}
 ];
ListPlot[X, Joined → {True}]
```

Out[ ]=



Take p random walks for each number of steps n (up to nmax) and calculate the mean distance from the origin

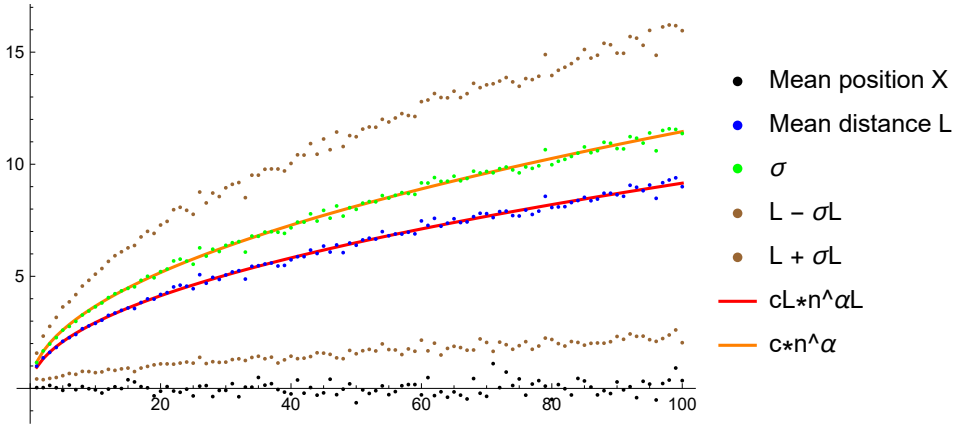
In[ ]:=
```
(* SeedRandom[2021] *)
nmax = 100;(* maximal number of steps *)
p = 1000;(* number of walks to average *)
d = 2.0;(* maximal length of each step *)
(* array for averaging the final position for each number of steps *)
X = ConstantArray[0.0, nmax];
(* array for averaging distances from origin for each number of steps *)
L = ConstantArray[0.0, nmax];
(* array for calculating standard deviation
 of final position for each number of steps *)
σ = ConstantArray[0.0, nmax];
(* array for calculating standard deviation
 of distances from origin for each number of steps *)
σL = ConstantArray[0.0, nmax];
Do[
  Do[
   x = 0.0;
   Do[
    x += d * RandomReal[{-1.0, 1.0}],
    {in, 1, n}
   ];
   X[[n]] += x;
   L[[n]] += Abs[x];
   σ[[n]] += x^2;
   σL[[n]] += Abs[x]^2,
   {ip, 1, p}
  ];
  X[[n]] = X[[n]] / p;
  L[[n]] = L[[n]] / p;
  σ[[n]] = Sqrt[σ[[n]] / p - X[[n]]^2];
  σL[[n]] = Sqrt[σL[[n]] / p - L[[n]]^2],
  {n, 1, nmax}
 ];
{c1, α1} = {a, b} /. FindFit[L, a * y^b, {a, b}, y];
{c2, α2} = {a, b} /. FindFit[σ, a * y^b, {a, b}, y];
Print["Parameters of fitting cL*n^αL to mean distance: cL = ", c1, ", αL = ", α1]
Print["Parameters of fitting c*n^α to σ: c = ", c2, ", α = ", α2]
Print["For σ we should get: c = ", N[d / Sqrt[3]], ", α = ", N[1 / 2]]
fn1 = Array[c1 * #^α1 &, nmax];
fn2 = Array[c2 * #^α2 &, nmax];
ListPlot[{X, L, σ, L - σL, L + σL, fn1, fn2},
 Joined → {False, False, False, False, False, True, True},
 PlotStyle → {Black, Blue, Green, Brown, Brown, Red, Orange},
 PlotLegends →
  {"Mean position X", "Mean distance L", "σ", "L - σL", "L + σL", "cL*n^αL", "c*n^α"}]
```

```
Parameters of fitting cL*n^αL to mean distance: cL = 0.940783, αL = 0.494103

Parameters of fitting c*n^α to σ: c = 1.17527, α = 0.494429

For σ we should get: c = 1.1547, α = 0.5
```

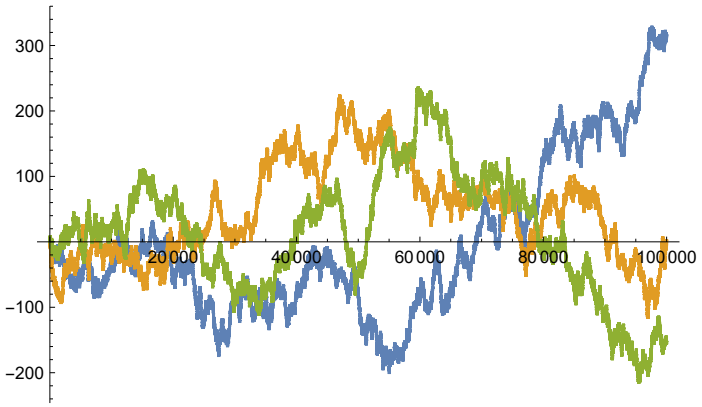
Out[ ]=



Simple walk in 1D - discrete steps ($d = 1$) of random direction

Start at origin and make 10000 random steps:

```
p = 3; (* number of walks *)
n = 100 000;(* number of steps *)
d = 1.0;(* maximal length of each step *)
X = ConstantArray[0.0, {p, n + 1}];
(* SeedRandom[2021] *)
Do[
  Do[
   X[[ip, in + 1]] = X[[ip, in]] + d * RandomChoice[{-1, 1}],
   {in, 1, n}
  ],
  {ip, 1, p}
 ];
ListPlot[X, Joined → {True}]
```



Take p random walks for each number of steps n (up to nmax) and calculate the mean distance from the origin

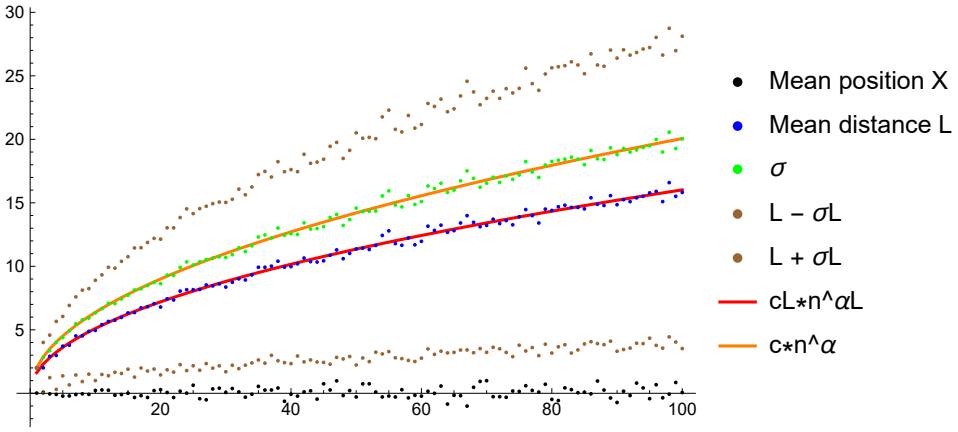
In[ ]:=
```
nmax = 100;(* maximal number of steps *)
p = 1000;(* number of walks to average *)
d = 2.0;(* maximal length of each step *)
(* array for averaging the final position for each number of steps *)
X = ConstantArray[0.0, nmax];
(* array for averaging distances from origin for each number of steps *)
L = ConstantArray[0.0, nmax];
(* array for calculating standard deviation
 of final position for each number of steps *)
σ = ConstantArray[0.0, nmax];
(* array for calculating standard deviation
 of distances from origin for each number of steps *)
σL = ConstantArray[0.0, nmax];
Do[
  Do[
   x = 0.0;
   Do[
    x += d * RandomChoice[{-1, 1}],
    {in, 1, n}
   ];
   X[[n]] += x;
   L[[n]] += Abs[x];
   σ[[n]] += x^2;
   σL[[n]] += Abs[x]^2,
   {ip, 1, p}
  ];
  X[[n]] = X[[n]] / p;
  L[[n]] = L[[n]] / p;
  σ[[n]] = Sqrt[σ[[n]] / p - X[[n]]^2];
  σL[[n]] = Sqrt[σL[[n]] / p - L[[n]]^2],
  {n, 1, nmax}
 ];
{c1, α1} = {a, b} /. FindFit[L, a * y^b, {a, b}, y];
{c2, α2} = {a, b} /. FindFit[σ, a * y^b, {a, b}, y];
Print["Parameters of fitting cL*n^αL to mean distance: cL = ", c1, ", αL = ", α1]
Print["Parameters of fitting c*n^α to σ: c = ", c2, ", α = ", α2]
Print["For σ we should get: c = ", N[d], ", α = ", N[1 / 2]]
fn1 = Array[c1 * #^α1 &, nmax];
fn2 = Array[c2 * #^α2 &, nmax];
ListPlot[{X, L, σ, L - σL, L + σL, fn1, fn2},
 Joined → {False, False, False, False, False, True, True},
 PlotStyle → {Black, Blue, Green, Brown, Brown, Red, Orange},
 PlotLegends →
  {"Mean position X", "Mean distance L", "σ", "L - σL", "L + σL", "cL*n^αL", "c*n^α"}]
```

```
Parameters of fitting cL*n^αL to mean distance: cL = 1.62563, αL = 0.496821

Parameters of fitting c*n^α to σ: c = 2.02468, α = 0.498024

For σ we should get: c = 2., α = 0.5
```

Out[ ]=


30
25
20
15
10
5
20
40
60
80
100
Mean position X
Mean distance L
σ
L − σL
L + σL
cL*n^αL
c*n^α