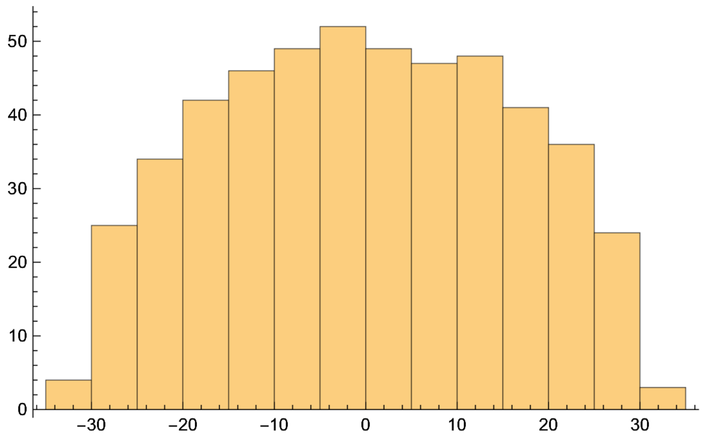
Directory

GOE Eigenvalues

Create GOE matrix

In[28]:=
```
(* matrix from a GOE: matGOE *)
(* dimension of the matrix: dim *)
(* Variance_(ii)=1, Variance_(ij)=1/2 *)

Clear[dim, rm, matGOE, Egoe];
dim = 500;

var = 1.;
rm =
  Table[Table[RandomReal[NormalDistribution[0, var]], {j, 1, dim}], {k, 1, dim}];
matGOE = ( rm + Transpose[rm]) / 2;
Egoe = Eigenvalues[matGOE];


Histogram[Egoe]

Print[];
Print["The lowest eigenvalue is ", Min[Egoe]]
Print["and the highest eigenvalue is ", Max[Egoe]]
Print["They are indeed close to √(2 Dim) = ", Sqrt[2. dim]]

Print[];
Print["The variance is ", Variance[Egoe]]
Print["which is indeed close to dim/2 = ", dim / 2.]
```

Out[34]=




```
The lowest eigenvalue is -31.8339
and the highest eigenvalue is 30.9359
They are indeed close to √2 Dim = 31.6228

The variance is 249.971
which is indeed close to dim/2 = 250.
```

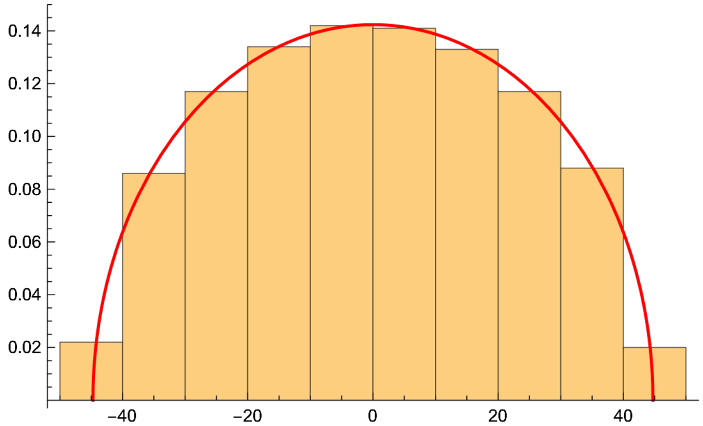
Histogram of the EIGENVALUES

In[ ]:=

```
te = Import["Eig_GOE_D1000AveR0001.dat", "Table"];
ta = Table[te[[k, 1]], {k, 1, Length[te]}];
dim = Length[ta];

Print[]
Print[]
Print["NORMALIZED Histogram with the analytical expression"]
aa = Histogram[ta, Automatic, "Probability"];
binsize = 10;
bb = Plot[(binsize / (Pi dim)) Sqrt[2 dim - x^2],
   {x, -Sqrt[2. dim], Sqrt[2. dim]}, PlotRange → All, PlotStyle → Red];
Show[{aa, bb}]

Print[];
Print["The lowest eigenvalue is ", Min[ta]]
Print["and the highest eigenvalue is ", Max[ta]]
Print["They are indeed close to √2 Dim = ", Sqrt[2. dim]]

Print[];
Print["The variance is ", Variance[ta]]
Print["which is indeed close to dim/2 = ", dim / 2.]
```

NORMALIZED Histogram with the analytical expression

Out[ ]=



The lowest eigenvalue is −44.3228

and the highest eigenvalue is 44.6234

They are indeed close to $\sqrt{2\,\text{Dim}}$ = 44.7214

The variance is 502.454

which is indeed close to dim/2 = 500.

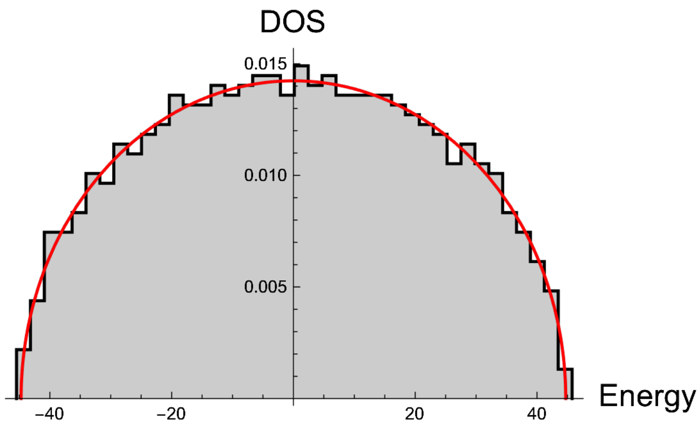
DOS

In[ ]:=

```
te = Import["DOS_GOE_D1000AveR0001.dat", "Table"];
dim = 1000.;
aa = ListPlot[te, Joined → True, PlotStyle → Black,
   Filling → Axis, AxesLabel → {"Energy", "DOS"}];
bb = Plot[(1. / (Pi dim)) Sqrt[2 dim - x^2], {x, -Sqrt[2. dim], Sqrt[2. dim]},
   PlotRange → All, PlotStyle → Red];
Print[]
Print[]
Print["NORMALIZED Histogram with the analytical expression"]
Show[{aa, bb}]
```

NORMALIZED Histogram with the analytical expression

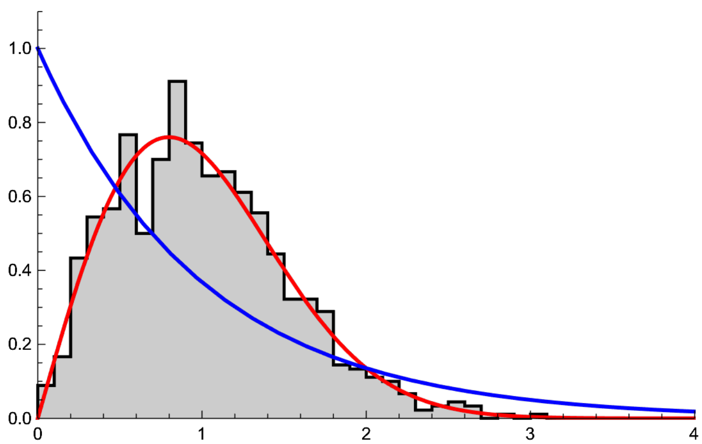
Out[ ]=



Level spacing distribution: P(s)

In[∘]:=
```
(* Theoretical curves *)
Clear[WignerDyson, Poisson];
WignerDyson = Plot[Pi s / 2. Exp[-Pi s^2 / 4.],
   {s, 0, 8}, PlotRange → {0, 1}, PlotStyle → {Red, Thick},
   LabelStyle → Directive[Black, Bold, Medium], AxesLabel → {"s", "P"}];
Poisson = Plot[Exp[-s], {s, 0, 8}, PlotRange → {0, 1}, PlotStyle → {Blue, Thick},
   LabelStyle → Directive[Black, Bold, Medium], AxesLabel → {"s", "P"}];

(* Numerical results *)
te = Import["Ps_GOE_D1000AveR0001.dat", "Table"];
dim = 1000.;
aa = ListPlot[te, Joined → True,
   PlotStyle → Black, Filling → Axis, PlotRange → {{0, 4}, {0, 1.1}}];

Show[{aa, WignerDyson, Poisson}]
```

Out[∘]=



Average of the ratio of consecutive levels: $\tilde{r}$

In[∘]:=
```
te = Import["rTilde_GOE_D1000AveR0001.dat", "Table"];
Print["The numerical result r̃=", te⟦1, 1⟧, " is close to theoretical r̃=0.54"]
```

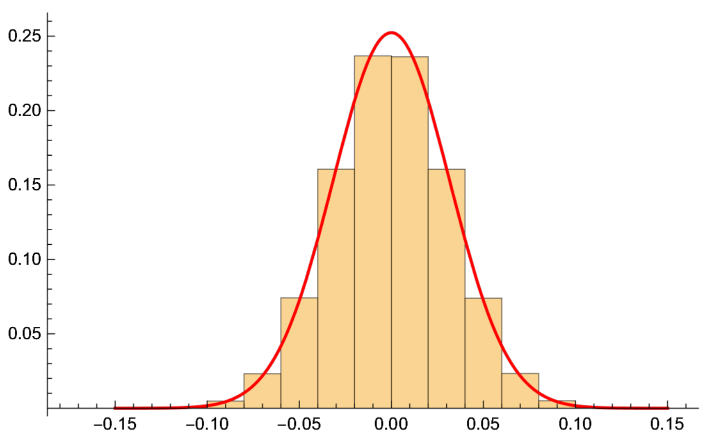
```
The numerical result r̃=0.534359 is close to theoretical r̃=0.54
```

GOE Eigenstates

Gaussian distribution of the coefficients

In[∘]:=
```
(* matrix from a GOE: matGOE *)
(* dimension of the matrix: dim *)
(* Variance_(ii)=1, Variance_(ij)=1/2 *)

Clear[dim, rm, matGOE, Egoe];
dim = 1000;

var = 1.;
rm =
  Table[Table[RandomReal[NormalDistribution[0, var]], {j, 1, dim}], {k, 1, dim}];
matGOE = ( rm + Transpose[rm]) / 2;
Egoe = Eigenvalues[matGOE];
Vgoe = Eigenvectors[matGOE];

vv = Flatten[Vgoe];
aa = Histogram[{vv}, Automatic, "Probability"];
bb = Plot[0.02 Sqrt[dim / (2 Pi )] Exp[- dim x^2 / 2],
   {x, - 0.15, 0.15}, PlotStyle → Red];
Show[{aa, bb}]
```

Out[∘]=



Porter-Thomas distribution

2nd moment

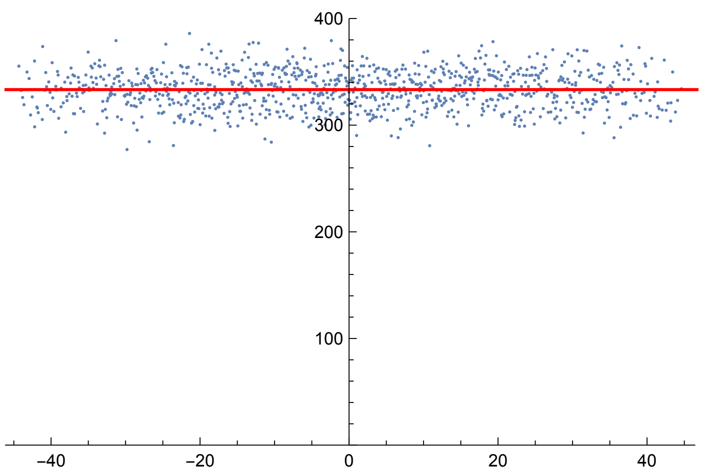
4th moment

PR and Shannon

In[∘]:=
```
te = Import["PRSh_GOE_D1000AveR0001.dat", "Table"];
pr = Table[{te⟦k, 1⟧, te⟦k, 2⟧}, {k, 1, Length[te]}];
sh = Table[{te⟦k, 1⟧, te⟦k, 3⟧}, {k, 1, Length[te]}];

prplot = ListPlot[pr, PlotRange → {0, dim / 2.5}];
shplot = ListPlot[sh, PlotRange → {0, Log[2 dim]}];
dim = Length[te];
prgoe = Plot[dim / 3., {x, -100, 100}, PlotStyle → Red];
shgoe = Plot[Log[0.48 dim], {x, -100, 100}, PlotStyle → Red];

Print[];
Print["Participation Ratio"];
Show[{prplot, prgoe}]
Print[];
Print["Shannon Entropy"];
Show[{shplot, shgoe}]
```

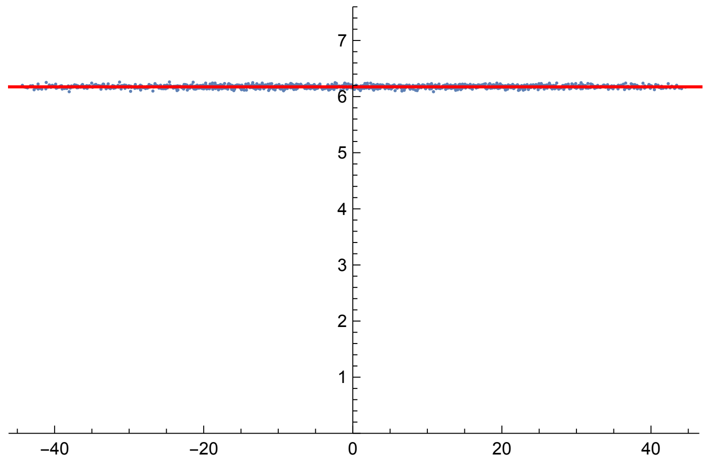
Participation Ratio

Out[∘]=



Shannon Entropy

Out[•]=



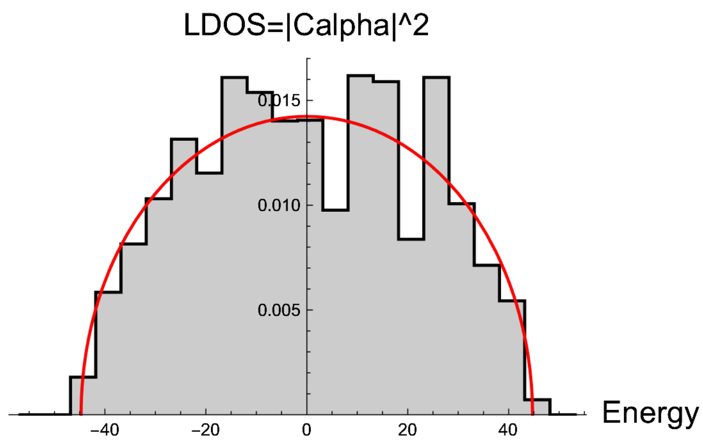
GOE Dynamics

LDOS

In[ ]:=
```
ena = Import["Calpha_GOE_D1000AveR0001AveI0100.dat"];
dim = Length[ena];
Energies = Table[ena[[k, 1]], {k, 1, dim}];
Calpha = Table[ena[[k, 2]], {k, 1, dim}];


Clear[lbin, tot, minE, maxE, Eint, Em];
lbin = 5.;
Min[Energies];
Max[Energies];
minE = Min[Energies] - 2 lbin;
maxE = Max[Energies] + 2 lbin;
tot = (maxE - minE) / lbin + 1;
Eint = Table[(minE - lbin / 2.) + lbin (k - 1), {k, 1, tot + 1}];
Em = Table[ minE + lbin (k - 1), {k, 1, tot}];

Clear[ldos];
Do[
  ldos[k] = 0.;
  , {k, 1, tot + 10}];

Do[
  Do[
    If[Eint[[k]] ≤ Energies[[j]] < Eint[[k + 1]], ldos[k] = ldos[k] + Calpha[[j]] ^2 ];
    , {k, 1, tot}];
  , {j, 1, dim}];

ldosLista = Flatten[Table[
    {{Eint[[k]], ldos[k] / (lbin )}, {Eint[[k + 1]], ldos[k] / (lbin )}}, {k, 1, tot}], 1];

ldosPlot = ListPlot[ldosLista, Joined → True, PlotStyle → Black, Filling → Axis,
   AxesLabel → {"Energy", "LDOS=|Calpha|^2"}, PlotRange → {All, {0, All}}];
bb = Plot[(1. / (Pi dim)) Sqrt[2 dim - x^2], {x, -Sqrt[2. dim], Sqrt[2. dim]},
   PlotRange → All, PlotStyle → Red];
Print[]
Print[]
Print["The LDOS is naturally normalized"]
Show[{ldosPlot, bb}]
```

The LDOS is naturally normalized

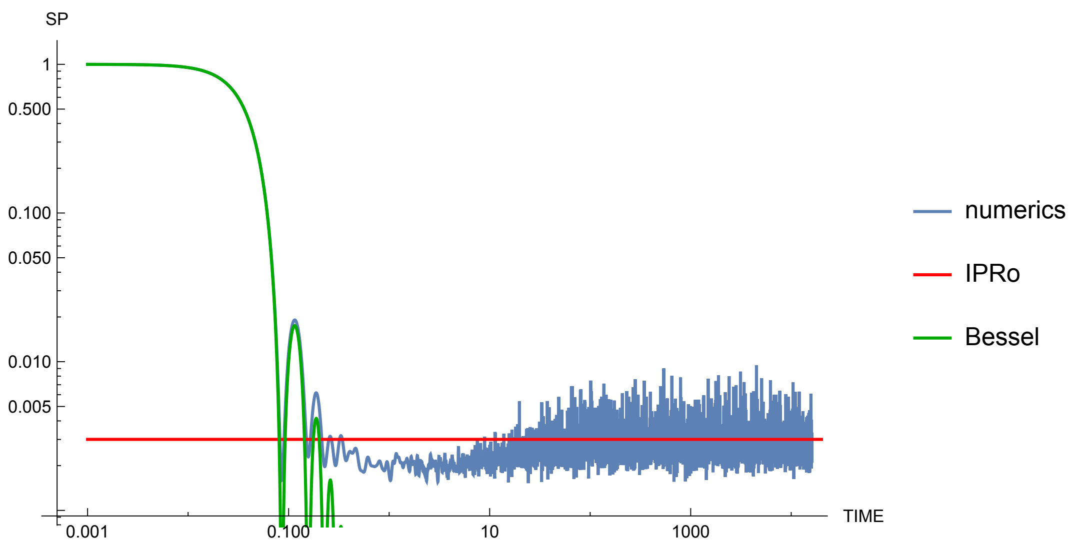
Out[ ]=



GOE SP

In[∘]:=
```
totIni = 100;
ta = Import["SP_GOE_D1000AveR0001AveI0100.dat", "Table"];
goeplot = ListLogLogPlot[ta, Joined → True, PlotRange → All,
   AxesLabel → {"TIME", "SP"}, PlotLegends → {"numerics"}];
te = Import["InfoIni_GOE_D1000AveR0001AveI0100.dat", "Table"];

ipr0 = Sum[te⟦11 + 15 (k - 1), 1⟧, {k, 1, totIni}] / totIni;
ipr0plot =
  LogLogPlot[ipr0, {x, 0.001, 20 000}, PlotStyle → Red, PlotLegends → {"IPRo"}];

gamSq = Sum[te⟦8 + 15 (k - 1), 1⟧, {k, 1, totIni}] / totIni;
besselplot = LogLogPlot[BesselJ[1, 2 Sqrt[gamSq] x] ^ 2 / (gamSq x ^ 2),
   {x, 0.001, 20}, PlotStyle → Darker[Green], PlotLegends → {"Bessel"}];

Show[{goeplot, ipr0plot, besselplot}]
```

Out[∘]=



IPR(t) averaged over 100 initial states

In[214]:=
```
totIni = 100;
ta = Import["IPRt_GOE_D1000AveR0001AveI0100.dat", "Table"];
splot = ListLogLogPlot[ta, Joined → True, PlotRange → All,
   AxesLabel → {"TIME", "IPR"}, PlotLegends → {"numerics"}];
te = Import["InfoIni_GOE_D1000AveR0001AveI0100.dat", "Table"];

ipr = Sum[te[[14 + 15 (k - 1), 1]], {k, 1, totIni}] / totIni;
iprplot = LogLogPlot[ipr, {x, 0.001, 200 000},
   PlotStyle → Red, PlotLegends → {"IPR_sat"}];

gamSq = Sum[te[[8 + 15 (k - 1), 1]], {k, 1, totIni}] / totIni;
besselplot = LogLogPlot[(BesselJ[1, 2 Sqrt[gamSq] x] ^2 / (gamSq x ^2) ) ^2,
   {x, 0.001, 20}, PlotStyle → Darker[Green], PlotLegends → {"Bessel^2"}];

Show[{splot, iprplot, besselplot}]
```

Out[222]=