Ultimate local-realistic CHSH simulation of two level entangled state event by event.
By Fred Diether with 2D vectors, also with Joy's hidden variable and modified epr-simple HV by Michel Fodje. Some parts by John Reed and Bill Nelson.

In[1592]:=
```
<< Quaternions`;
β0 = Quaternion[1, 0, 0, 0];
β1 = Quaternion[0, 1, 0, 0];
β2 = Quaternion[0, 0, 1, 0];
β3 = Quaternion[0, 0, 0, 1];
Qcoordinates = {β1, β2, β3};
m = 5000;
SeedRandom[6789];
λ2 = ConstantArray[0, m];
Ls1 = ConstantArray[0, m];
Ls2 = ConstantArray[0, m];
list1A = Table[{0, 0, 0}, m];
list1B = Table[{0, 0, 0}, m];
list2A = Table[{0, 0, 0}, m];
list2B = Table[{0, 0, 0}, m];
list3A = Table[{0, 0, 0}, m];
list3B = Table[{0, 0, 0}, m];
```

In[1609]:=
```
Do[vectorS = Flatten[{RandomPoint[Circle[]], 0}]; (*Singlet spin vector*)
 s1 = ToSphericalCoordinates[vectorS][[3]];
 λ2[[i]] = 1/4 Cos[s1/2]^2; (*hidden variable*)
 λ = RandomChoice[{-1, 1}]; (*hidden variable*)
 Ls1[[i]] = λ * vectorS.Qcoordinates; (*singlet spin quaternion with HV*)
 Ls2[[i]] = -Ls1[[i]], {i, m}]
```

In[1610]:=
```
Do[vectorA = RandomChoice[{{1, 0, 0}, {0, 1, 0}}]; (*0 and +90 degrees x-y*)
 Da = vectorA.Qcoordinates; (*Convert to quaternion coordinates*)
 If[Abs[Re[Da ** Ls1[[i]]]] > λ2[[i]], A = Sign[Re[Da ** Ls1[[i]]]], A = Null];
 If[Abs[ArcTan[vectorA[[1]], vectorA[[2]]]] ≤ π/4, Aa = 1, Aa = Null];
 If[Abs[ArcTan[vectorA[[1]], vectorA[[2]]]] ≤ π/4, AA = 1, AA = Null];
 list1A[[i]] = {ArcTan[vectorA[[1]], vectorA[[2]]] * 180/π, A, i};
 list2A[[i]] = {ArcTan[vectorA[[1]], vectorA[[2]]] * 180/π, Aa, m + i};
 list3A[[i]] = {ArcTan[vectorA[[1]], vectorA[[2]]] * 180/π, AA, 2 m + i}, {i, m}]
list1A1 = DeleteCases[list1A, {_, Null, _}];
list2A1 = DeleteCases[list2A, {_, Null, _}];
list3A1 = DeleteCases[list3A, {_, Null, _}];
outA = Catenate[{list1A1, list2A1, list3A1}];
```

In[1615]:=
```
Do[vectorB = RandomChoice[{{1/Sqrt[2], 1/Sqrt[2], 0}, {-1/Sqrt[2], 1/Sqrt[2], 0}}]; (*  +45, +135*)
 Db = vectorB.Qcoordinates;
 If[Abs[Re[Ls2[[i]] ** Db]] > λ2[[i]], B = Sign[Re[Db ** Ls2[[i]]]], B = Null ];
 If[13 π/16 ≤ Abs[ArcTan[vectorB[[1]], vectorB[[2]]]] ≤ π, Bb = 1, Bb = Null];
 If[Abs[ArcTan[vectorB[[1]], vectorB[[2]]]] ≤ π/4, BB = -1, BB = Null];
 list1B[[i]] = {ArcTan[vectorB[[1]], vectorB[[2]]] * 180/π, B, i};
 list2B[[i]] = {ArcTan[vectorB[[1]], vectorB[[2]]] * 180/π, Bb, m + i};
 list3B[[i]] = {ArcTan[vectorB[[1]], vectorB[[2]]] * 180/π, BB, 2 m + i}, {i, m}]
list1B1 = DeleteCases[list1B, {_, Null, _}];
list2B1 = DeleteCases[list2B, {_, Null, _}];
list3B1 = DeleteCases[list3B, {_, Null, _}];
outB = Catenate[{list1B1, list2B1, list3B1}];
```

CHSH Analysis of Particle Data

In[1620]:=
```
listA3 = outA[[All, 3]]; (*match trial numbers*)
listB3 = outB[[All, 3]];
listAd = Take[Select[outA, Intersection[{#[[3]]}, listB3] == {#[[3]]} &], m];
listBd = Take[Select[outB, Intersection[{#[[3]]}, listA3] == {#[[3]]} &], m];
a = listAd[[All, 1]];
b = listBd[[All, 1]];
A1 = listAd[[All, 2]];
B1 = listBd[[All, 2]];
```

In[1628]:=
```
nP1 = 0; nN1 = 0; nP2 = 0; nN2 = 0; nP3 = 0; nN3 = 0; nP4 = 0; nN4 = 0;
Do[a1 = a[[j]];
 b1 = b[[j]];
 aliceD = A1[[j]]; bobD = B1[[j]];
 If[(b1 == 45) && (a1 - b1 == -45) && aliceD * bobD == 1, nP1++];
 If[(b1 == 45) && (a1 - b1 == -45) && aliceD * bobD == -1, nN1++];
 If[(a1 - b1) == -135 && aliceD * bobD == 1, nP2++];
 If[(a1 - b1) == -135 && aliceD * bobD == -1, nN2++];
 If[(a1 - b1) == 45 && aliceD * bobD == 1, nP3++];
 If[(a1 - b1) == 45 && aliceD * bobD == -1, nN3++];
 If[a1 == 90 && (a1 - b1) == -45 && aliceD * bobD == 1, nP4++];
 If[a1 == 90 && (a1 - b1) == -45 && aliceD * bobD == -1, nN4++], {j, m}]
E1 = N[(nP1 - nN1) / (nP1 + nN1)];
E2 = N[(nP2 - nN2) / (nP2 + nN2)];
E3 = N[(nP3 - nN3) / (nP3 + nN3)];
E4 = N[(nP4 - nN4) / (nP4 + nN4)];
CHSH = Abs[E1 - E2 + E3 + E4];
Print["CHSH = ", CHSH]
```

```
CHSH = 2.53455
```