Phase A

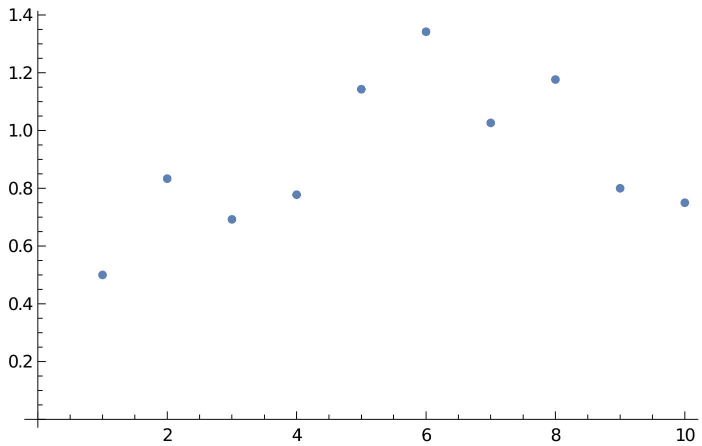
In[1]:=
```
astrologerSuccess = {3, 10, 9, 7, 16, 17, 13, 20, 4, 1};
expected = {6, 12, 13, 9, 14, 12.66666667, 12.66666667, 17, 5, 1.333333333};
```

In[3]:=
```
data = N[astrologerSuccess / expected]
```

Out[3]=
```
{0.5, 0.833333, 0.692308, 0.777778, 1.14286, 1.34211, 1.02632, 1.17647, 0.8, 0.75}
```

In[4]:=
```
ListPlot[data]
```

Out[4]=



In[8]:=
```
lmfA = LinearModelFit[data, x, x]
```

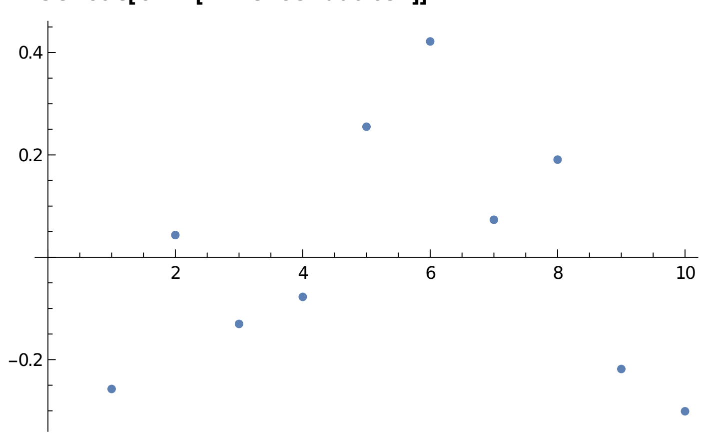
Out[8]=
```
FittedModel[ 0.724705 + 0.0326203 x ]
```

In[9]:=
```
Normal[lmfA]
```

Out[9]=
```
0.724705 + 0.0326203 x
```

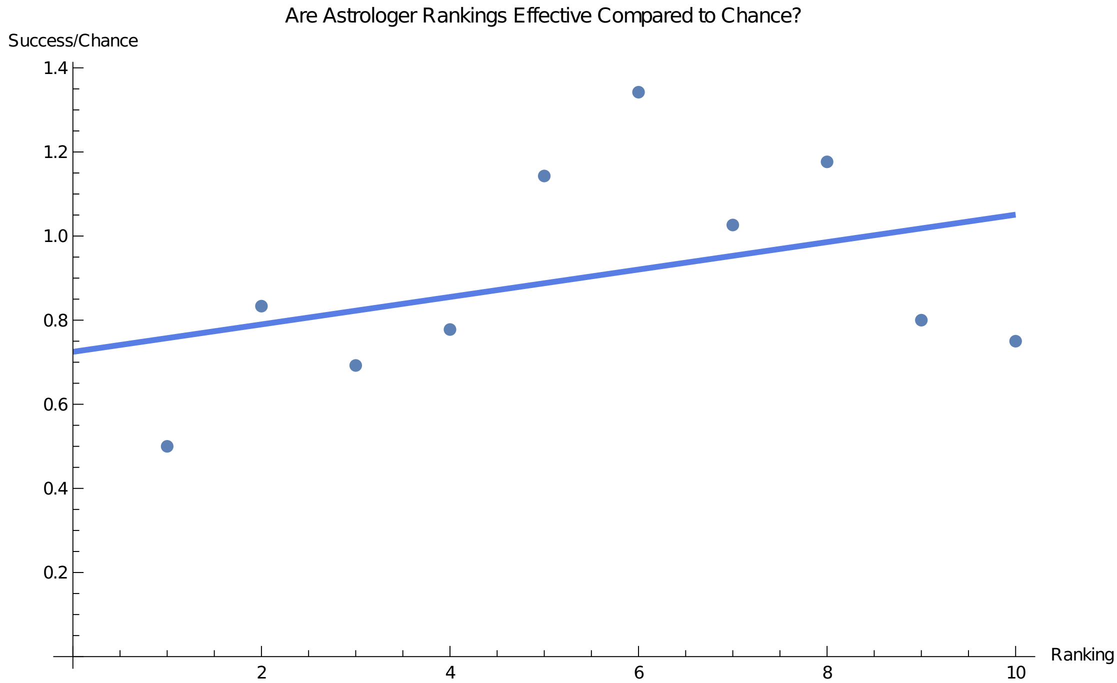
In[10]:=
```
ListPlot[lmfA["FitResiduals"]]
```

Out[10]=



In[12]:=
```
Show[ListPlot[data,
   PlotLabel → "Are Astrologer Rankings Effective Compared to Chance?",
   AxesLabel → {"Ranking", "Success/Chance"}],
  Plot[lmf[x], {x, 0, 10}, PlotTheme → "Business"], ImageSize → Large]
```

Out[12]=



In[13]:=
```
lmfA["ParameterTable"]
```

Out[13]=

	Estimate	Standard Error	t-Statistic	P-Value
1	0.724705	0.173139	4.18569	0.0030557
x	0.0326203	0.0279038	1.16903	0.276045

In[14]:=
```
lmfA["ParameterConfidenceIntervals"]
```

Out[14]= {{0.325447, 1.12396}, {-0.031726, 0.0969666}}

In[15]:=
```
lmfA["RSquared"]
```

Out[15]= 0.145903

In[16]:=
```
lmfA["ANOVATable"]
```

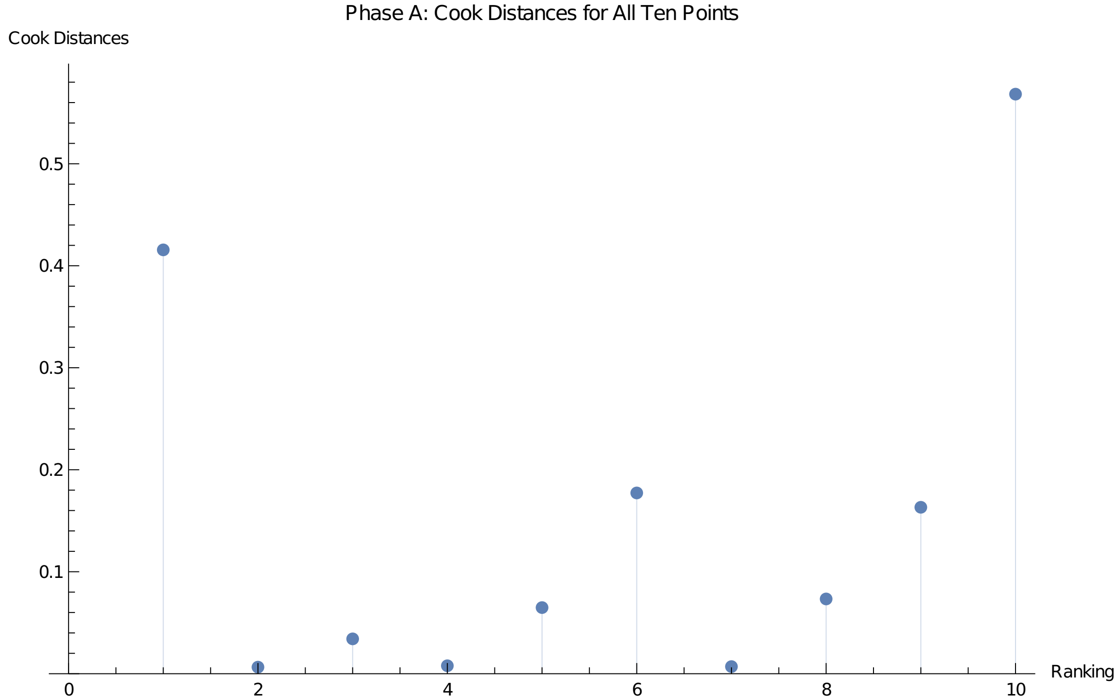
Out[16]=

	DF	SS	MS	F-Statistic	P-Value
x	1	0.0877867	0.0877867	1.36662	0.276045
Error	8	0.513891	0.0642364		
Total	9	0.601678			

Check Cook distances to identify highly influential points :

In[26]:=
```
ListPlot[cdA = lmfA["CookDistances"], PlotRange → {0, All},
 Filling → 0, AxesLabel → {"Ranking", "Cook Distances"},
 PlotLabel → "Phase A: Cook Distances for All Ten Points", ImageSize → "Large"]
```

Out[26]=

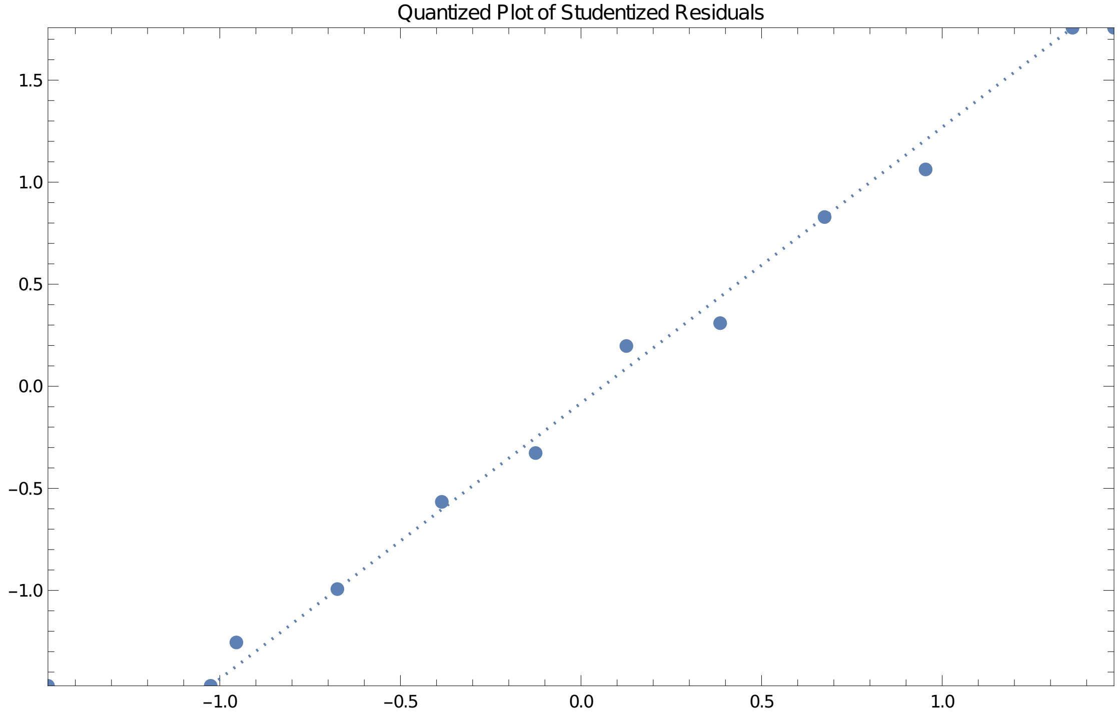


In[18]:=
```
Position[cdA, _?(# > .5 &)]
```

Out[18]= {{10}}

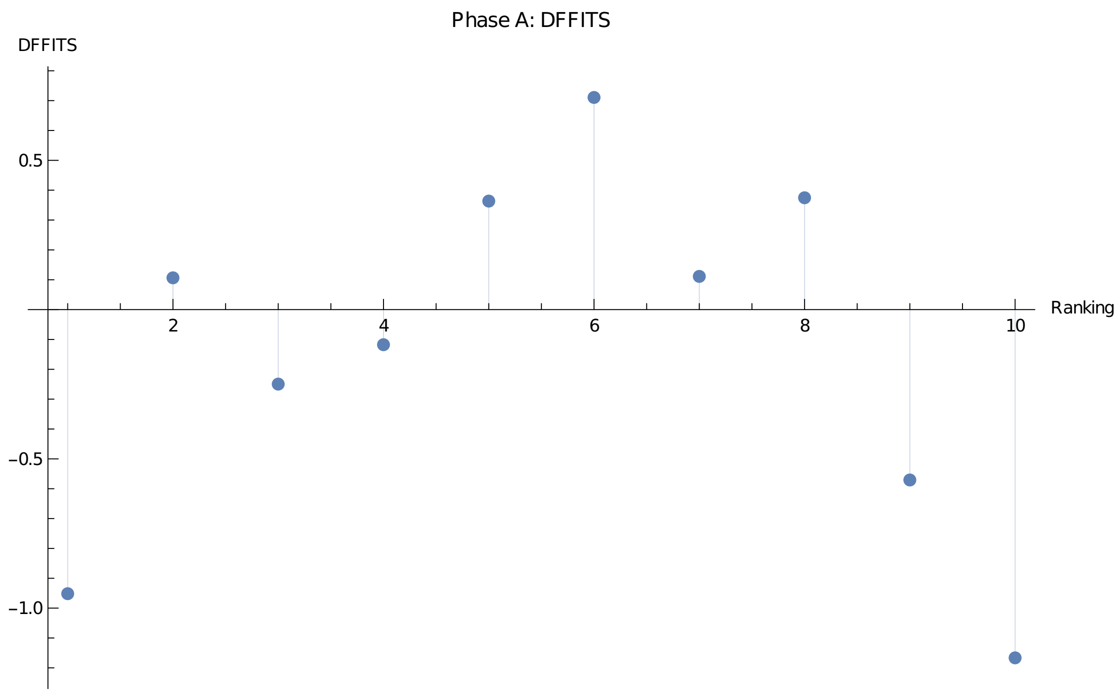
In[25]:=
```
QuantilePlot[lmfA["StandardizedResiduals"],
 Table[InverseCDF[NormalDistribution[], q], {q, 1/100, 99/100, 1/50}],
 PlotLabel → "Quantized Plot of Studentized Residuals", ImageSize → Large]
```

Out[25]=



Use DFFITS values to assess the influence of each point on the fitted values :

In[24]:=
```
ListPlot[lmfA["FitDifferences"], PlotRange → All,
 Filling → 0, "PlotLabel" → "Phase A: DFFITS",
 AxesLabel → {"Ranking", "DFFITS"}, ImageSize → Large]
```

Out[24]=


Use DFBETAS values to assess the influence of each point on each estimated parameter :

In[∘]:=
```
N[2 / Sqrt[10]]
```

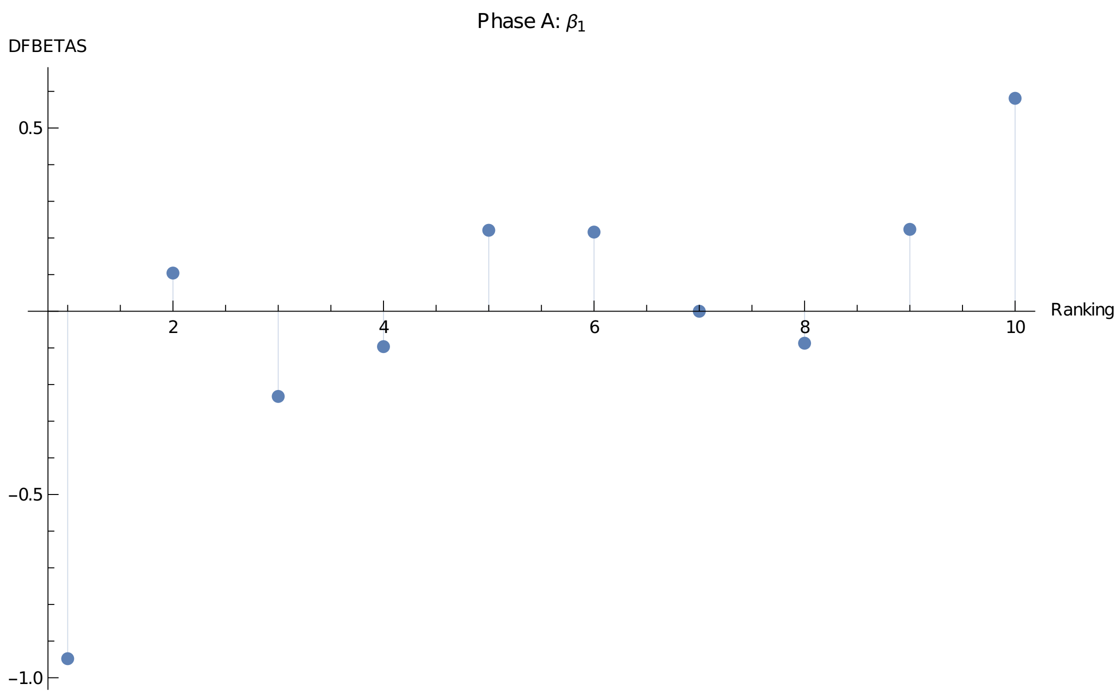
Out[∘]= 0.632456

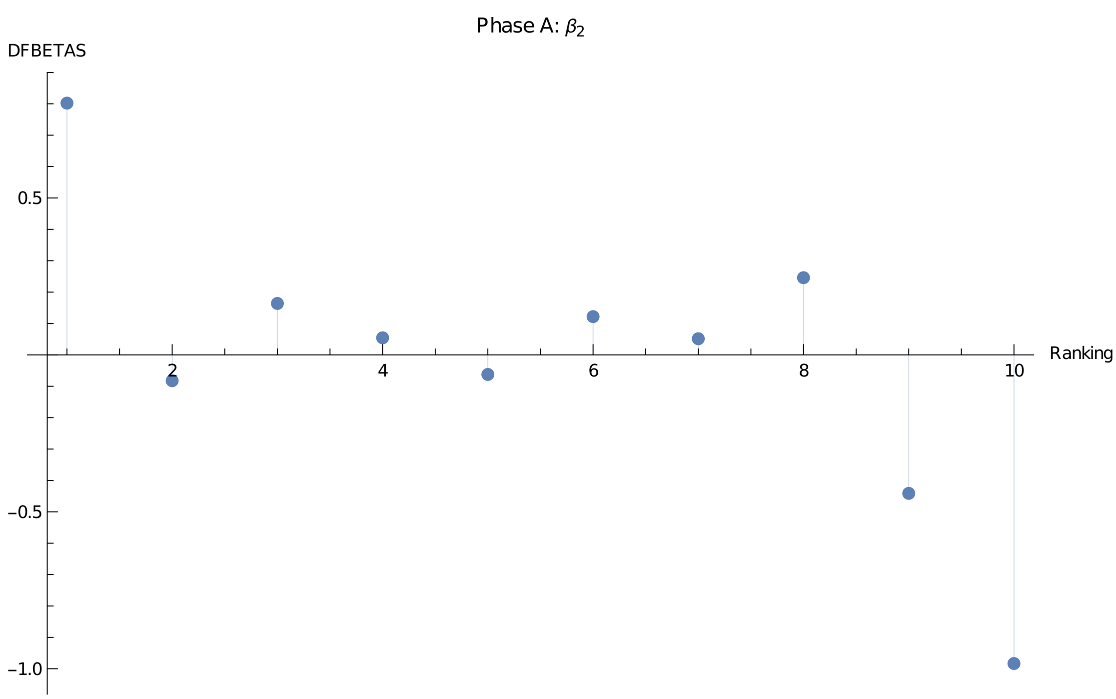
In[28]:=
```
dfbetasA = Transpose[lmfA["BetaDifferences"]]
```

Out[28]= {{-0.94802, 0.104226, -0.232384, -0.0964876, 0.22102, 0.216049, -4.00511×10^{-18}, -0.0872096, 0.223478, 0.581074}, {0.802133, -0.0823081, 0.163853, 0.0544264, -0.0623362, 0.121868, 0.0514793, 0.245964, -0.441205, -0.98331}}

In[33]:=
```
MapThread[ListPlot[#1, PlotRange → All, Filling → 0, PlotLabel → "Phase A: " <> #2,
    ImageSize → Large, AxesLabel → {"Ranking", "DFBETAS"}] &,
  {Transpose[lmf["BetaDifferences"]], {"β1", "β2"(*,"β3"*)}}] // Row
```

Out[33]=





Phase B

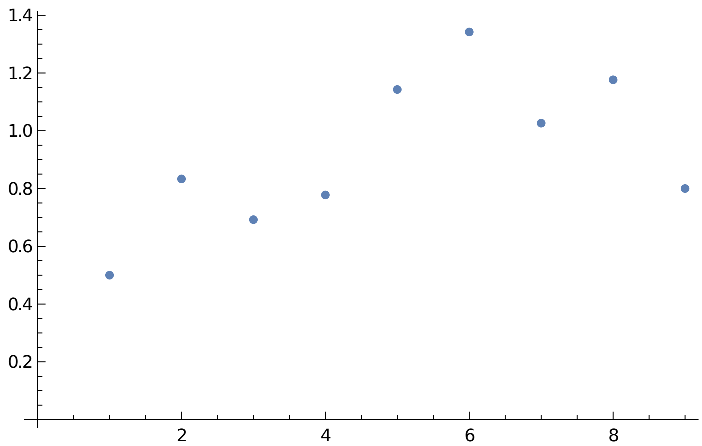
In[80]:=
```
astrologerSuccess = {3, 10, 9, 7, 16, 17, 13, 20, 4(*,1*)};
expected = {6, 12, 13, 9, 14, 12.66666667, 12.66666667, 17, 5(*,1.333333333*)};
```

In[81]:=
```
data = N[astrologerSuccess / expected]
```

Out[81]=
```
{0.5, 0.833333, 0.692308, 0.777778, 1.14286, 1.34211, 1.02632, 1.17647, 0.8}
```

In[82]:=
```
ListPlot[data]
```

Out[82]=



In[83]:=
```
lmfB = LinearModelFit[data, x, x]
```

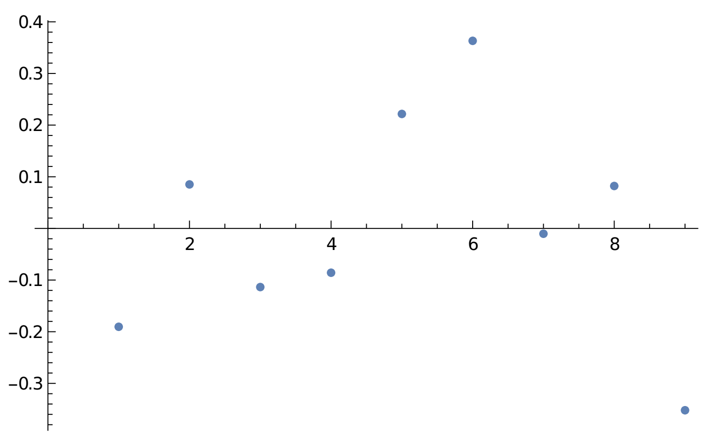
Out[83]=
```
FittedModel[ 0.632761 + 0.0576959 x ]
```

In[84]:=
```
Normal[lmfB]
```

Out[84]=
```
0.632761 + 0.0576959 x
```

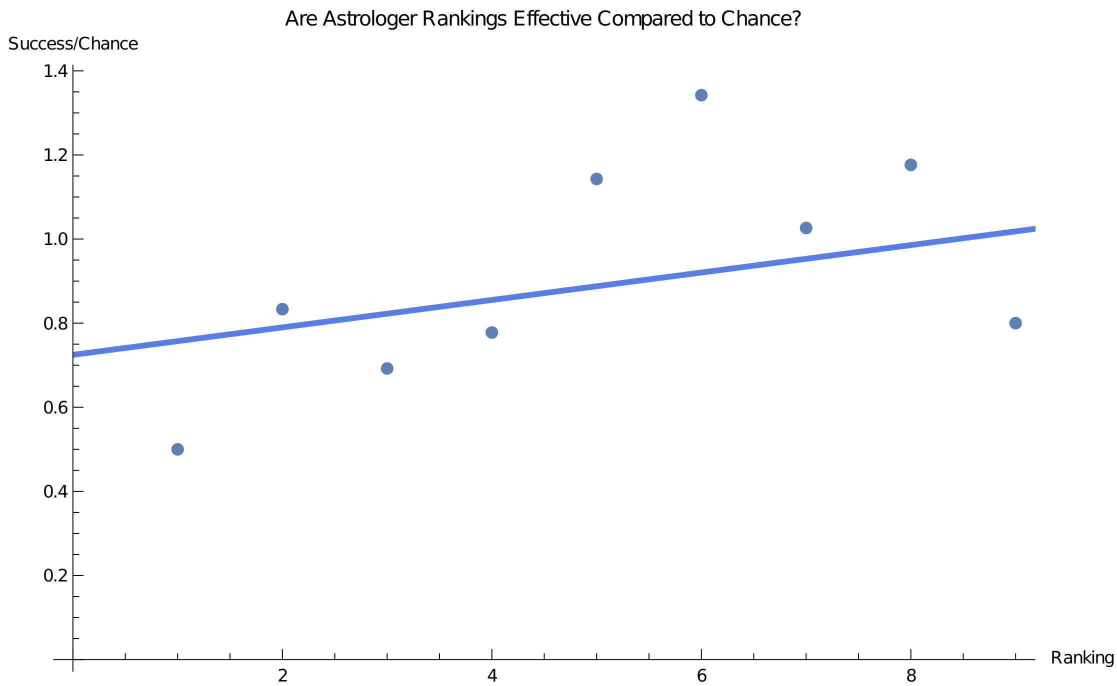
In[85]:=
```
ListPlot[lmfB["FitResiduals"]]
```

Out[85]=



In[86]:=
```
Show[ListPlot[data,
   PlotLabel → "Are Astrologer Rankings Effective Compared to Chance?",
   AxesLabel → {"Ranking", "Success/Chance"}],
  Plot[lmf[x], {x, 0, 10}, PlotTheme → "Business"], ImageSize → Large]
```

Out[86]=



In[87]:=
```
lmfB["ParameterTable"]
```

Out[87]=

	Estimate	Standard Error	t-Statistic	P-Value
1	0.632761	0.168273	3.76032	0.00707189
x	0.0576959	0.0299029	1.92944	0.0950004

In[88]:=
```
lmfB["ParameterConfidenceIntervals"]
```

Out[88]= {{0.234859, 1.03066}, {-0.0130132, 0.128405}}

In[89]:=
```
lmfB["RSquared"]
```

Out[89]= 0.347182

In[90]:=
```
lmfB["ANOVATable"]
```

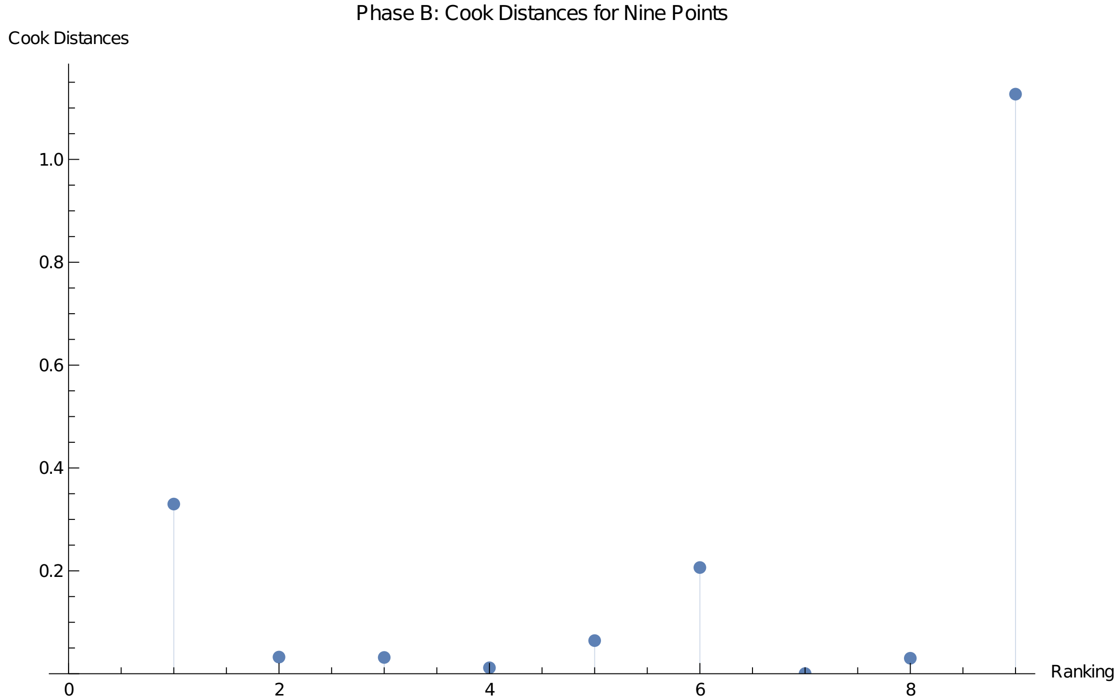
Out[90]=

	DF	SS	MS	F-Statistic	P-Value
x	1	0.199729	0.199729	3.72274	0.0950004
Error	7	0.375557	0.0536511		
Total	8	0.575287			

Check Cook distances to identify highly influential points :

In[91]:=
```
ListPlot[cdB = lmfB["CookDistances"], PlotRange → {0, All},
 Filling → 0, AxesLabel → {"Ranking", "Cook Distances"},
 PlotLabel → "Phase B: Cook Distances for Nine Points", ImageSize → "Large"]
```

Out[91]=

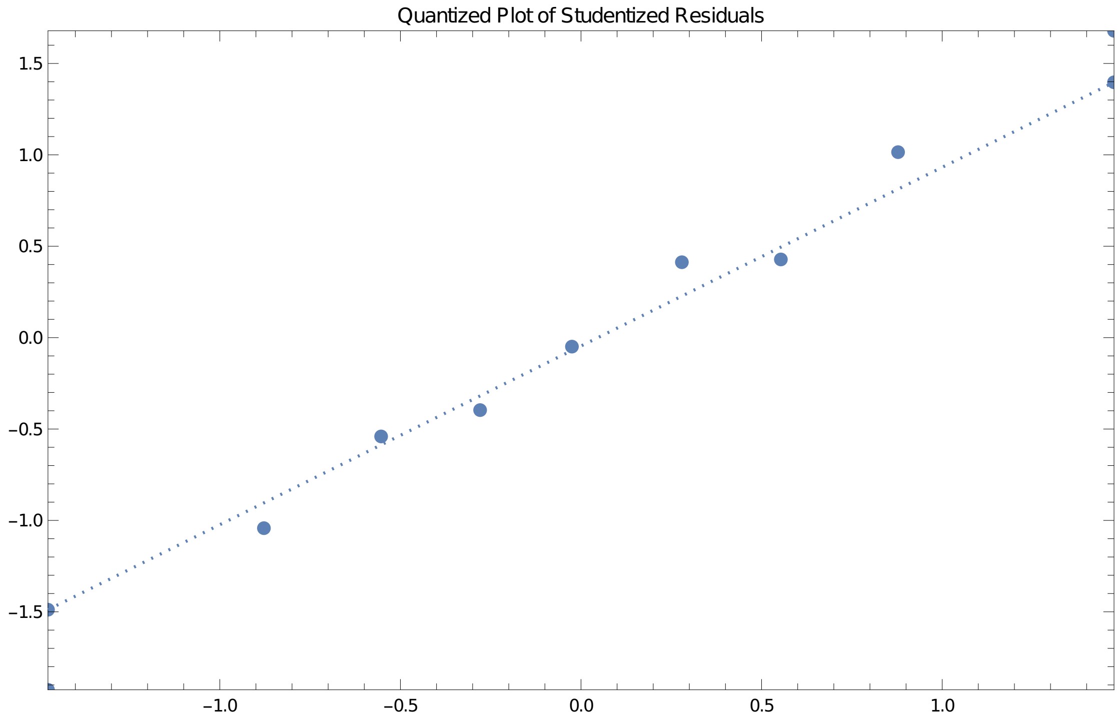


In[92]:=
```
Position[cdB, _?(# > .5 &)]
```

Out[92]= {{9}}

In[93]:=
```
QuantilePlot[lmfB["StandardizedResiduals"],
 Table[InverseCDF[NormalDistribution[], q], {q, 1/100, 99/100, 1/50}],
 PlotLabel → "Quantized Plot of Studentized Residuals", ImageSize → Large]
```

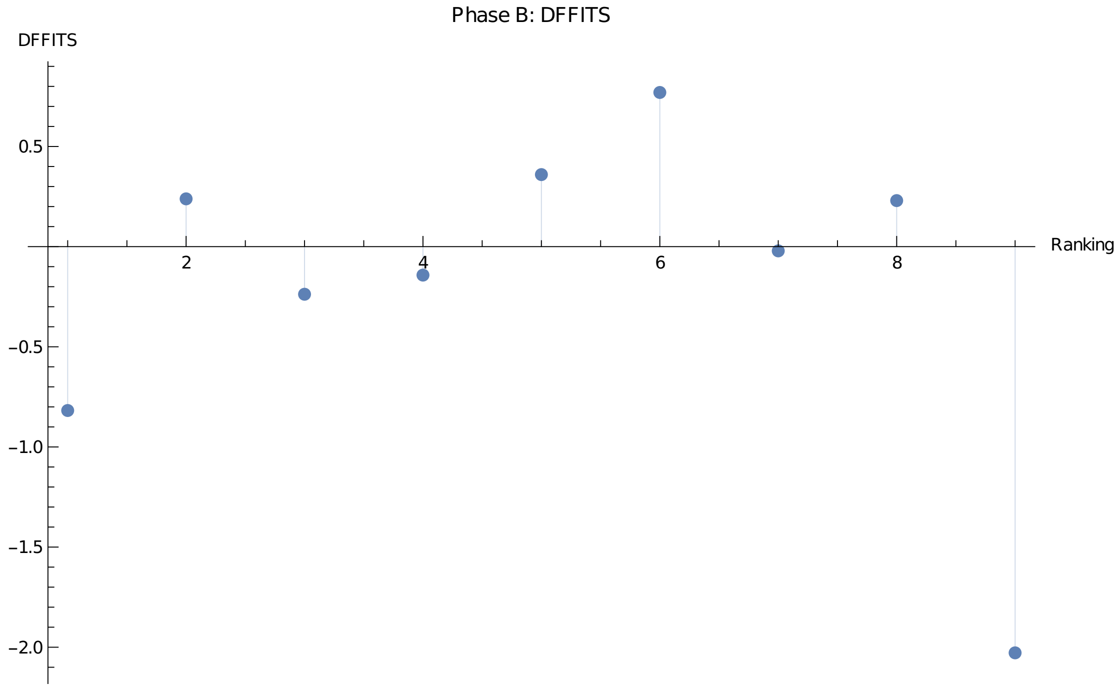
Out[93]=



Use DFFITS values to assess the influence of each point on the fitted values :

In[94]:=
```
ListPlot[lmfB["FitDifferences"], PlotRange → All,
 Filling → 0, "PlotLabel" → "Phase B: DFFITS",
 AxesLabel → {"Ranking", "DFFITS"}, ImageSize → Large]
```

Out[94]=



Use DFBETAS values to assess the influence of each point on each estimated parameter :

In[51]:=
```
N[2 / Sqrt[9]]
```

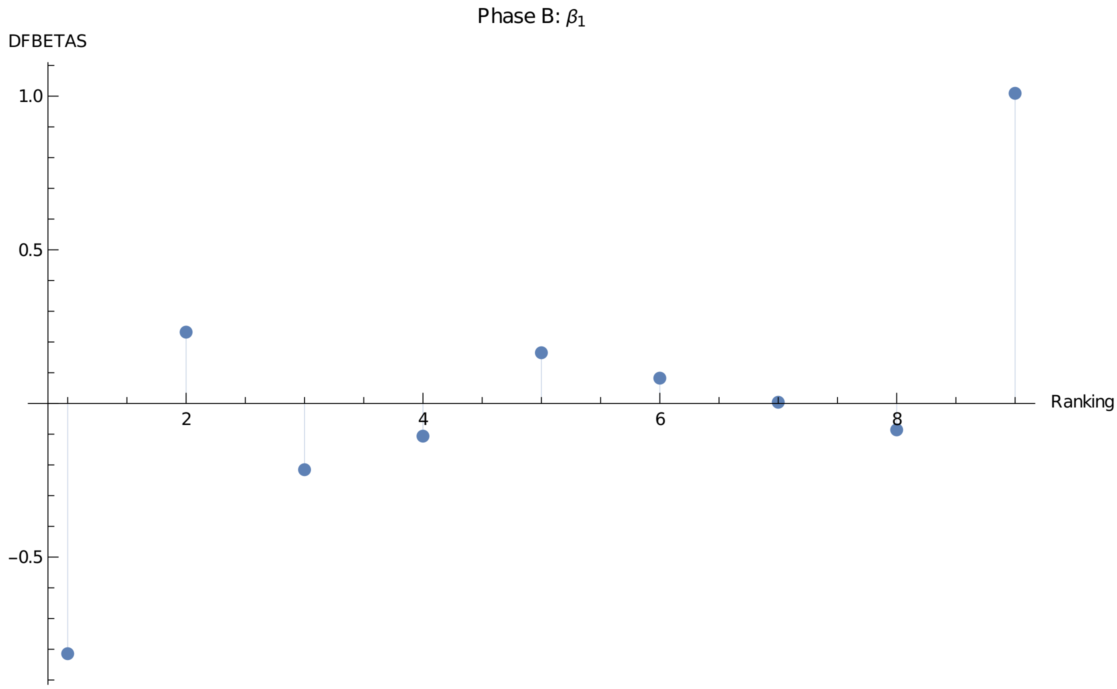
Out[51]= 0.666667

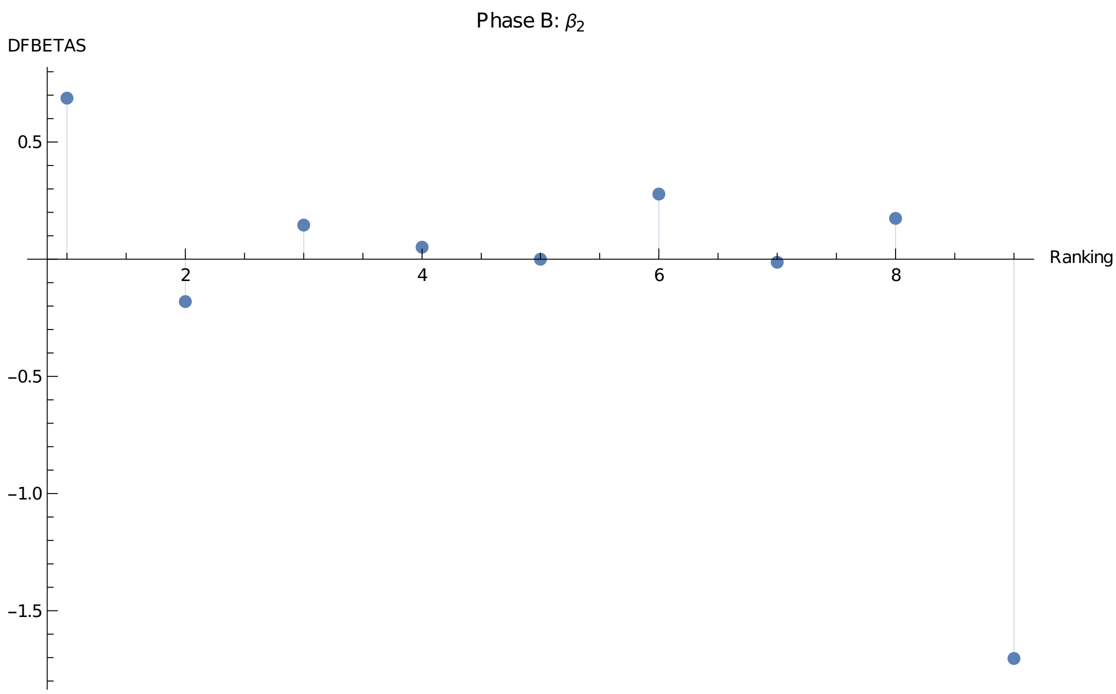
In[52]:=
```
dfbetasB = Transpose[lmfB["BetaDifferences"]]
```

Out[52]= {{-0.814351, 0.232095, -0.215593, -0.106398, 0.165037, 0.0823324, 0.00383596, -0.0860024, 1.00929}, {0.687391, -0.180841, 0.145585, 0.0513201, -1.37421×10^{-17}, 0.277986, -0.0129517, 0.174227, -1.70389}}

In[54]:=
```
MapThread[ListPlot[#1, PlotRange → All, Filling → 0, PlotLabel → "Phase B: " <> #2,
    ImageSize → Large, AxesLabel → {"Ranking", "DFBETAS"}] &,
  {Transpose[lmfB["BetaDifferences"]], {"β1", "β2"(*,"β3"*)}}] // Row
```

Out[54]=





Phase C

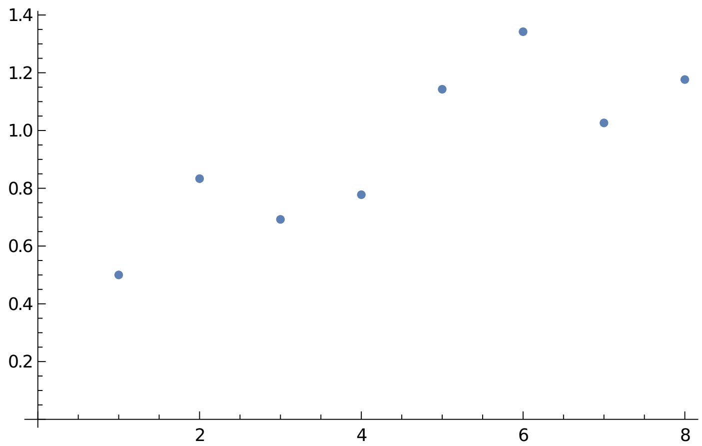
In[55]:=
```
astrologerSuccess = {3, 10, 9, 7, 16, 17, 13, 20(*,4*)(*,1*)};
expected = {6, 12, 13, 9, 14, 12.66666667, 12.66666667, 17(*,5*)(*,1.333333333*)};
```

In[56]:=
```
data = N[astrologerSuccess / expected]
```

Out[56]=
```
{0.5, 0.833333, 0.692308, 0.777778, 1.14286, 1.34211, 1.02632, 1.17647}
```

In[57]:=
```
ListPlot[data]
```

Out[57]=



In[58]:=
```
lmfC = LinearModelFit[data, x, x]
```

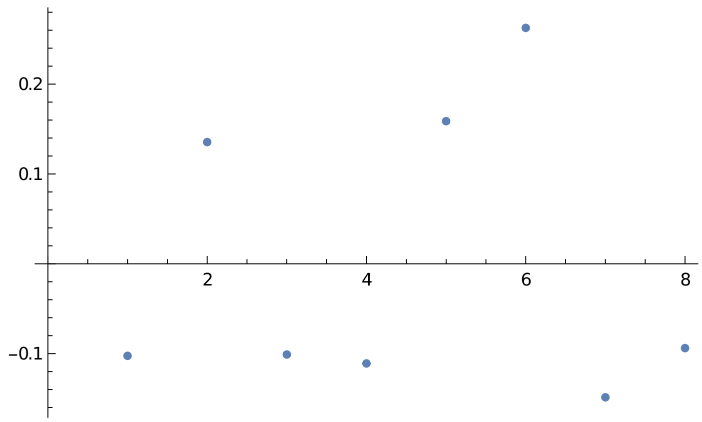
Out[58]=
```
FittedModel[ 0.507038 + 0.0954128 x ]
```

In[59]:=
```
Normal[lmfC]
```

Out[59]=
```
0.507038 + 0.0954128 x
```

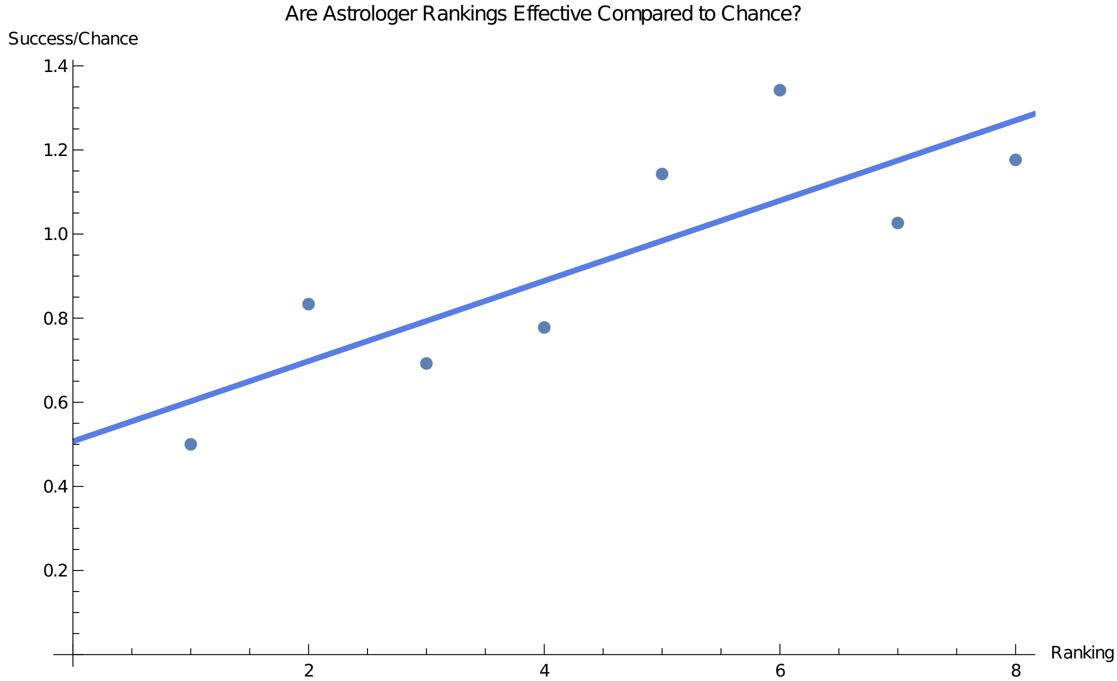
In[60]:=
```
ListPlot[lmfC["FitResiduals"]]
```

Out[60]=



In[61]:=
```
Show[ListPlot[data,
   PlotLabel → "Are Astrologer Rankings Effective Compared to Chance?",
   AxesLabel → {"Ranking", "Success/Chance"}],
 Plot[lmfC[x], {x, 0, 10}, PlotTheme → "Business"], ImageSize → Large]
```

Out[61]=



In[62]:=
```
lmfC["ParameterTable"]
```

Out[62]=

	Estimate	Standard Error	t-Statistic	P-Value
1	0.507038	0.133603	3.7951	0.00901931
x	0.0954128	0.0264574	3.60628	0.0112811

In[63]:=
```
lmfC["ParameterConfidenceIntervals"]
```

Out[63]= {{0.180123, 0.833953}, {0.030674, 0.160152}}

In[64]:=
```
lmfC["RSquared"]
```

Out[64]= 0.684298

In[65]:=
```
lmfC["ANOVATable"]
```

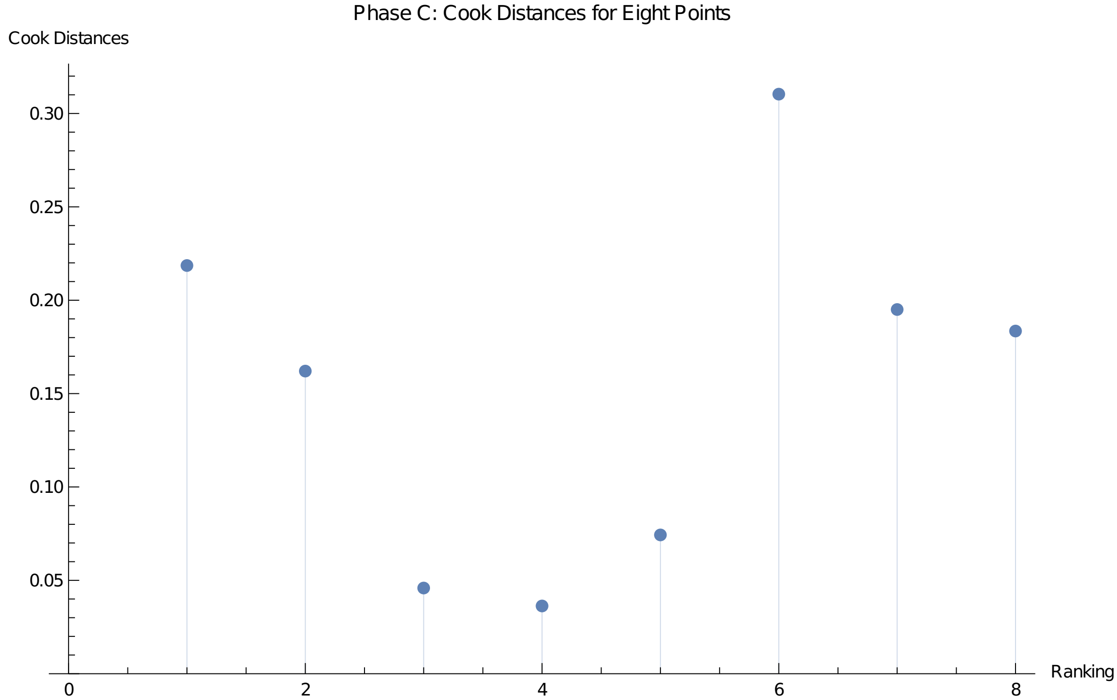
Out[65]=

	DF	SS	MS	F-Statistic	P-Value
x	1	0.382352	0.382352	13.0053	0.0112811
Error	6	0.176398	0.0293997		
Total	7	0.55875			

Check Cook distances to identify highly influential points :

In[66]:=
```
ListPlot[cdC = lmfC["CookDistances"], PlotRange → {0, All},
 Filling → 0, AxesLabel → {"Ranking", "Cook Distances"},
 PlotLabel → "Phase C: Cook Distances for Eight Points", ImageSize → "Large"]
```

Out[66]=

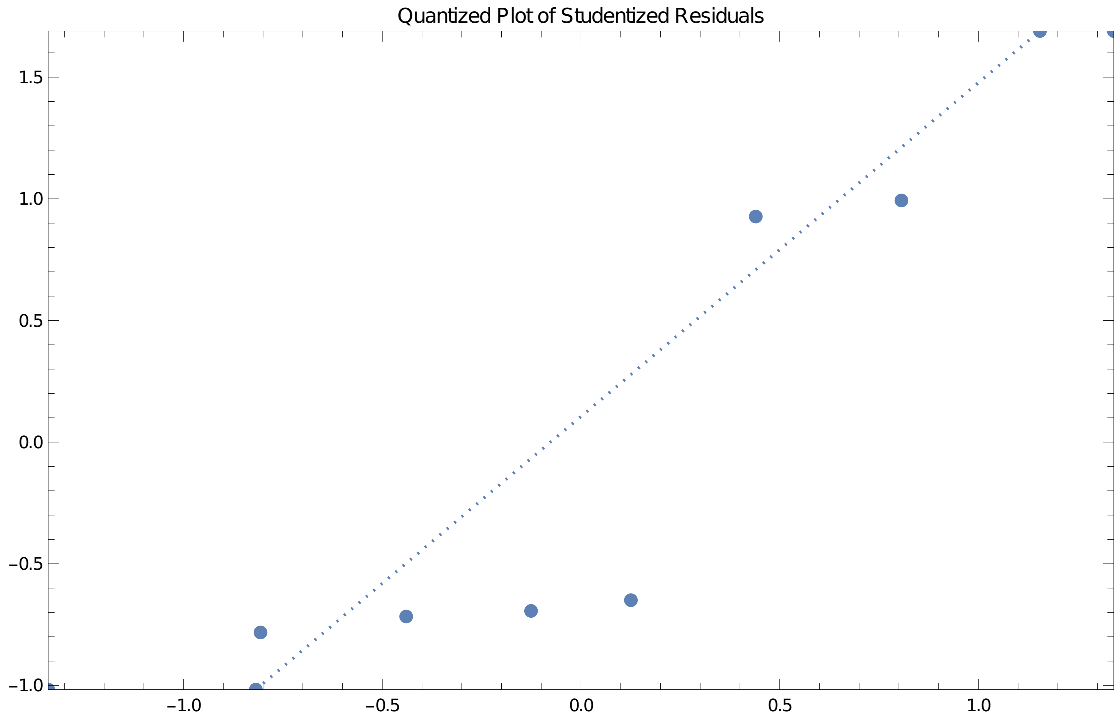


In[67]:=
```
Position[cdC, _?(# > .5 &)]
```

Out[67]= {}

In[68]:=
```
QuantilePlot[lmfC["StandardizedResiduals"],
 Table[InverseCDF[NormalDistribution[], q], {q, 1/100, 99/100, 1/50}],
 PlotLabel → "Quantized Plot of Studentized Residuals", ImageSize → Large]
```

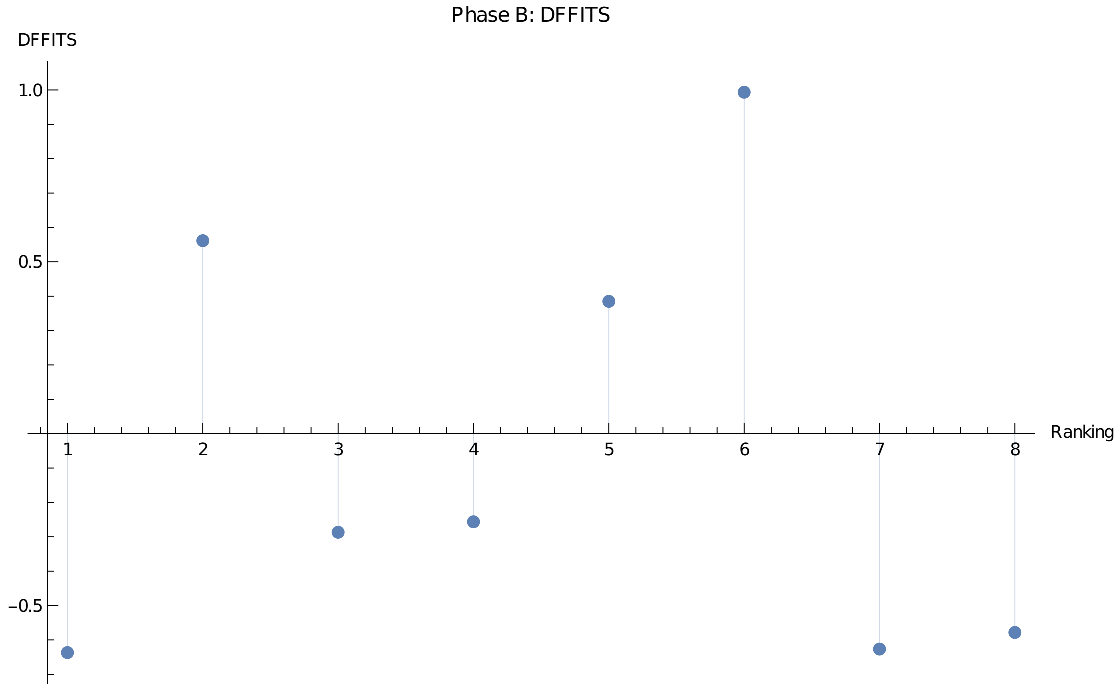
Out[68]=



Use DFFITS values to assess the influence of each point on the fitted values :

In[69]:=
```
ListPlot[lmfC["FitDifferences"], PlotRange → All,
 Filling → 0, "PlotLabel" → "Phase B: DFFITS",
 AxesLabel → {"Ranking", "DFFITS"}, ImageSize → Large]
```

Out[69]=



Use DFBETAS values to assess the influence of each point on each estimated parameter :

In[70]:=
```
N[2 / Sqrt[8]]
```

Out[70]= 0.707107

In[72]:=
```
dfbetasC = Transpose[lmfC["BetaDifferences"]]
```

Out[72]=
```
{{-0.633177, 0.540997, -0.248877, -0.162369,
  0.0975326, -0.107752, 0.219579, 0.287464}, {0.532898, -0.413924,
  0.157096, 0.0546615, 0.0820859, 0.544121, -0.462009, -0.483874}}
```

In[98]:=

```
MapThread[ListPlot[#1, PlotRange → All, Filling → 0, PlotLabel → "Phase C: " <> #2,
    ImageSize → Large, AxesLabel → {"Ranking", "DFBETAS"}] &,
  {Transpose[lmfC["BetaDifferences"]], {"β1", "β2"(*,"β3"*)}}] // Row
```

Out[98]=