Fletcher Monthly Average Temperatures

Andy Long
Spring, 2020

Data from Fletcher: Monthly Means of Mean, Min, and Max Temperatures, Thirty Year Averages, Period 1959 to 1988 (Means and Maxs, p. 81), and Period 1949-1978 (Mins, p. 82)

In[29]:=
```
DecimalDate = Table[i + 1/24, {i, 0, 1, 1/12}];
meanTempData =
  {24.1, 27.7, 39.6, 49.0, 60.7, 70.3, 73.5, 70.1, 65.0, 53.8, 41.6, 29.5, 24.1};
maxTempData = {30.5, 37.3, 46.6, 60.2, 72.7, 81.9,
   84.9, 80.1, 76.7, 64.5, 50.0, 36.5, 30.5};
minTempData = {15.5, 17.8, 27.5, 37.8, 48.6, 58.7, 61.8,
   59.9, 53.4, 40.2, 33.0, 21.9, 15.5};
```

Model Means

In[33]:=
```
data = Transpose[{DecimalDate, meanTempData}];
meanlm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
meanlm["ParameterTable"]
meanlm["RSquared"]
meanlm["AdjustedRSquared"]
meanlm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]) . (weight - Mean[weight])
residuals = meanlm["FitResiduals"];
```

Out[33]= FittedModel[50.2511 + 24.3348 Sin[2 π (−0.30373 + t)]]

Out[34]=

	Estimate	Standard Error	t-Statistic	P-Value
m	50.2511	0.488254	102.92	1.83746×10^{-16}
a	24.3348	0.665671	36.5568	5.57934×10^{-12}
e	0.30373	0.00467271	65.0008	1.80805×10^{-14}

Out[35]= 0.999113

Out[36]= 0.998847

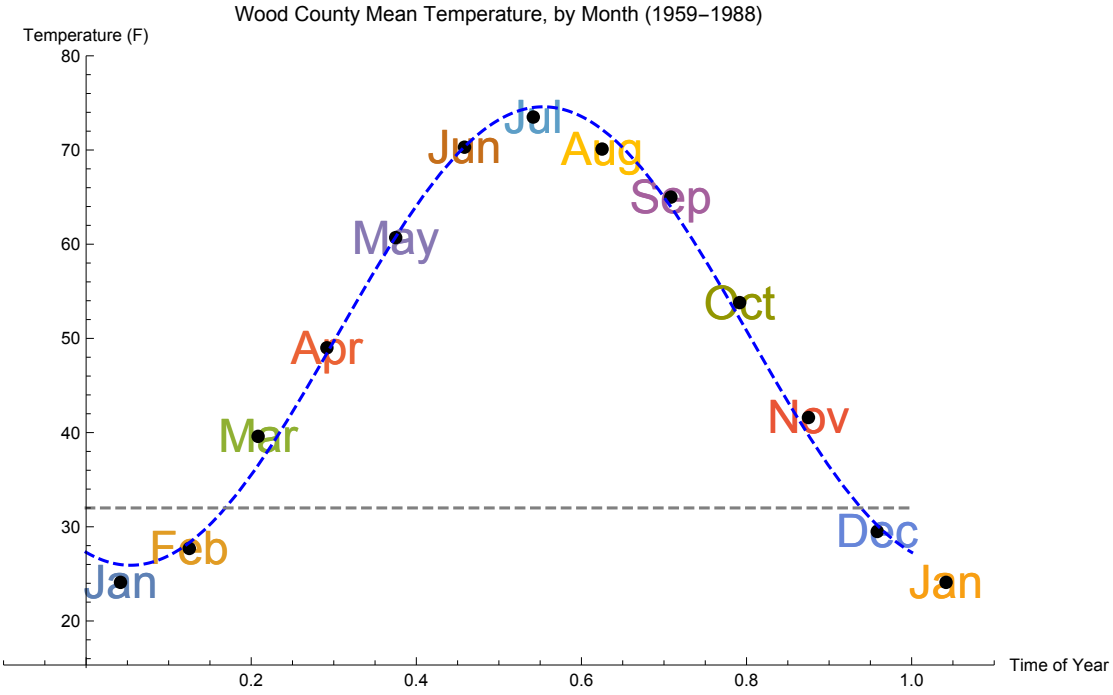
Out[37]= {{49.1633, 51.339}, {22.8516, 25.818}, {0.293318, 0.314141}}

Out[38]= (weight - Mean[weight]) . (weight - Mean[weight])

In[40]:=
```
months = {"Jan", "Feb", "Mar", "Apr", "May",
    "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec", "Jan"};
sizes = Table[Large, {i, 1, 13}];
markers = Transpose[{months, sizes}];

p0 = ListPlot[data, PlotStyle → {Black, PointSize[Large]}];
(*
Thanks
  https://mathematica.stackexchange.com/questions/190153/how-
  to-use-a-different-symbol-for-each-point-in-a-list-plot/190154
*)
p1 = ListPlot[List /@ data,
    PlotMarkers → markers,
    PlotLabel → "Wood County Mean Temperature, by Month (1959-1988)",
    PlotRange → {{-.1, 1.1}, {15, 80}},
    AxesLabel → {"Time of Year", "Temperature (F)"},
    ImageSize → Large];
p2 = Plot[meanlm[x], {x, 0, 1}, PlotStyle → {Blue, Dashed}];
p3 = Plot[32, {x, 0, 1}, PlotStyle → {Gray, Dashed}];
all1 = Show[p1, p0, p2, p3]
```

Out[47]=



Model Maxs

In[48]:=
```
data = Transpose[{DecimalDate, maxTempData}];
maxlm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
maxlm["ParameterTable"]
maxlm["RSquared"]
maxlm["AdjustedRSquared"]
maxlm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]) . (weight - Mean[weight])
residuals = maxlm["FitResiduals"];
SSE = residuals.residuals
```

Out[49]= FittedModel[59.9276 + 26.7048 Sin[2 π (-0.301018 + t)]]

Out[50]=

	Estimate	Standard Error	t-Statistic	P-Value
m	59.9276	0.672843	89.0662	7.78866×10^{-16}
a	26.7048	0.917174	29.1164	5.32574×10^{-11}
e	0.301018	0.00586869	51.2921	1.91886×10^{-13}

Out[51]= 0.998801

Out[52]= 0.998441

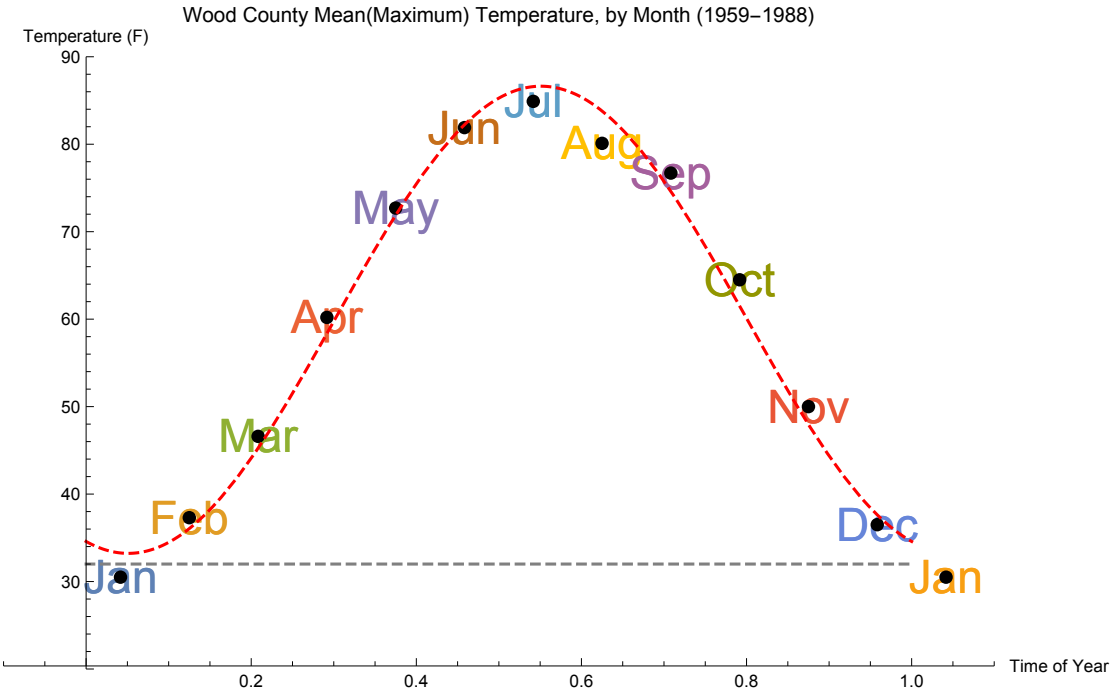
Out[53]= {{58.4284, 61.4268}, {24.6612, 28.7484}, {0.287941, 0.314094}}

Out[54]= (weight - Mean[weight]) . (weight - Mean[weight])

Out[56]= 58.2066

In[57]:=
```
p0 = ListPlot[data, PlotStyle → {Black, PointSize[Large]}];
(*
Thanks
  https://mathematica.stackexchange.com/questions/190153/how-
  to-use-a-different-symbol-for-each-point-in-a-list-plot/190154
*)
p1 = ListPlot[List /@ data,
    PlotMarkers → markers,
    PlotLabel → "Wood County Mean(Maximum) Temperature, by Month (1959-1988)",
    PlotRange → {{-.1, 1.1}, {20, 90}},
    AxesLabel → {"Time of Year", "Temperature (F)"},
    ImageSize → Large];
p2 = Plot[maxlm[x], {x, 0, 1}, PlotStyle → {Red, Dashed}];
p3 = Plot[32, {x, 0, 1}, PlotStyle → {Gray, Dashed}];
all2 = Show[p1, p0, p2, p3]
```

Out[61]=



Model Mins

In[62]:=
```
data = Transpose[{DecimalDate, minTempData}];
minlm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
minlm["ParameterTable"]
minlm["RSquared"]
minlm["AdjustedRSquared"]
minlm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]) . (weight - Mean[weight])
residuals = minlm["FitResiduals"];
SSE = residuals.residuals
```

Out[62]= FittedModel[39.5675 + 22.8968 Sin[2 π (−0.307879 + t)]]

Out[63]=

	Estimate	Standard Error	t-Statistic	P-Value
m	39.5675	0.38418	102.992	1.82467×10^{-16}
a	22.8968	0.523966	43.6991	9.46195×10^{-13}
e	0.307879	0.0039064	78.8139	2.64161×10^{-15}

Out[64]= 0.999146

Out[65]= 0.99889

Out[66]= {{38.7115, 40.4236}, {21.7294, 24.0643}, {0.299175, 0.316583}}

Out[67]= (weight - Mean[weight]) . (weight - Mean[weight])

Out[69]= 18.9764

In[70]:=

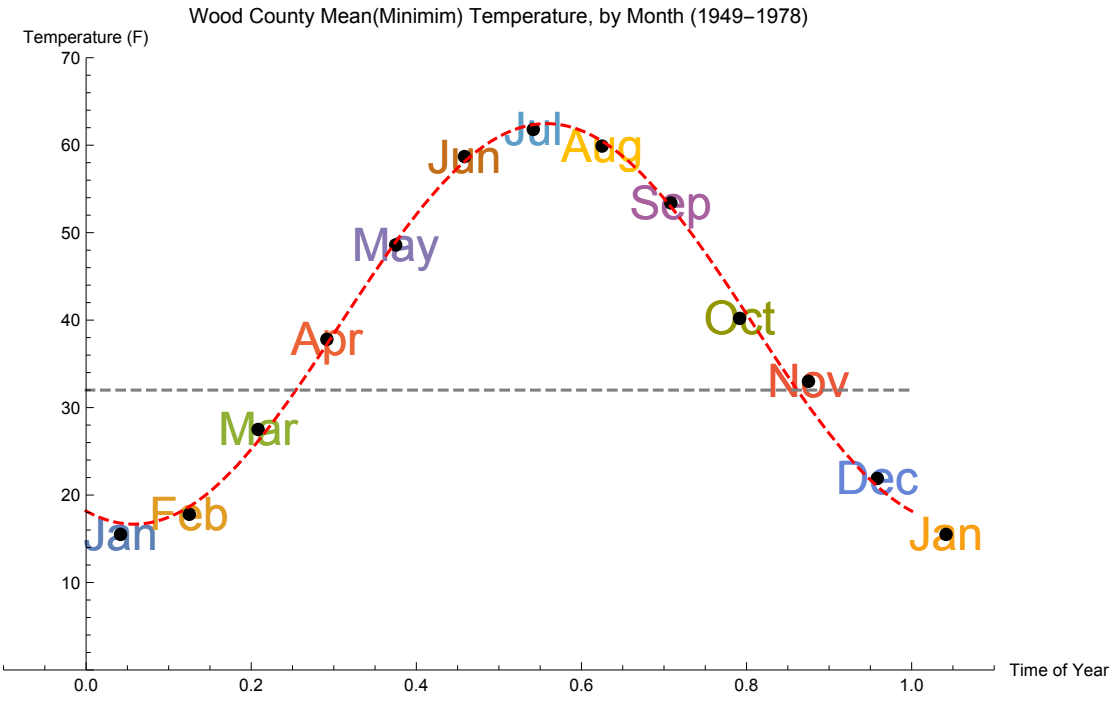
	Estimate	Standard Error	t-Statistic	P-Value
m	39.5675	0.38418	102.992	1.82467×10^{-16}
a	22.8968	0.523966	43.6991	9.46195×10^{-13}
e		□	78.8139	2.64161×10^{-15}

Out[70]=

	Estimate	Standard Error	t-Statistic	P-Value
m	39.5675	0.38418	102.992	1.82467×10^{-16}
a	22.8968	0.523966	43.6991	9.46195×10^{-13}
e		□	78.8139	2.64161×10^{-15}

In[71]:=
```
p0 = ListPlot[data, PlotStyle → {Black, PointSize[Large]}];
p1 = ListPlot[List /@ data,
    PlotMarkers → markers,
    PlotLabel → "Wood County Mean(Minimim) Temperature, by Month (1949-1978)",
    PlotRange → {{-.1, 1.1}, {0, 70}},
    AxesLabel → {"Time of Year", "Temperature (F)"},
    ImageSize → Large];
p2 = Plot[minlm[x], {x, 0, 1}, PlotStyle → {Red, Dashed}];
p3 = Plot[32, {x, 0, 1}, PlotStyle → {Gray, Dashed}];
all3 = Show[p1, p0, p2, p3]
```

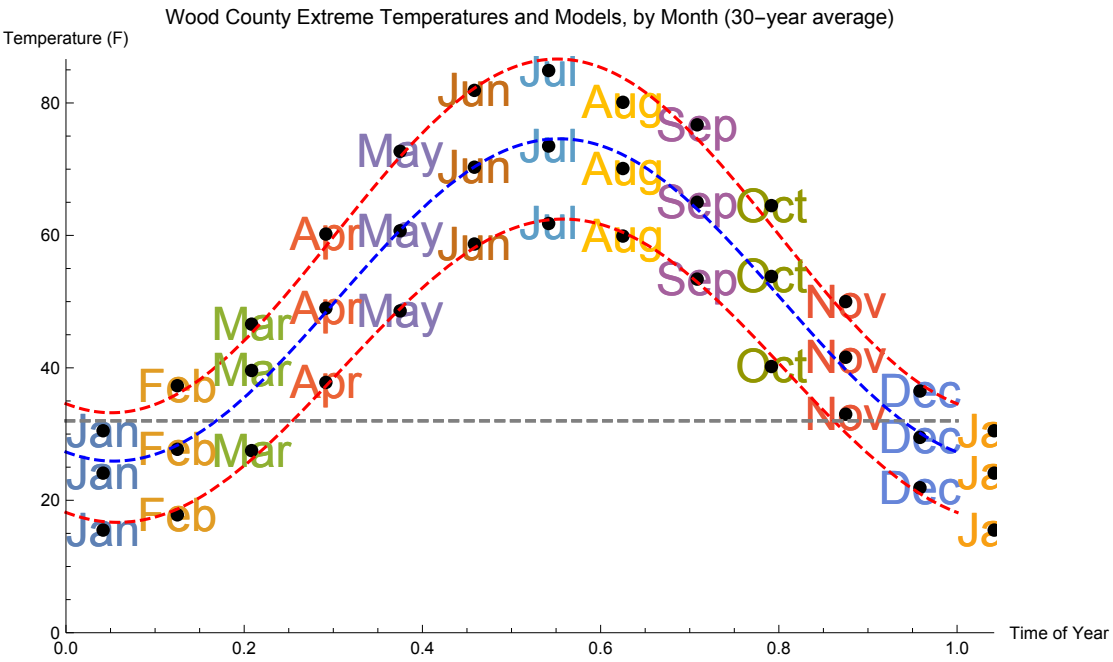
Out[75]=



All Together Now!

In[76]:=
```
p1 = ListPlot[List /@ data,
    PlotMarkers → markers,
    PlotLabel →
     "Wood County Extreme Temperatures and Models, by Month (30-year average)",
    PlotRange → {{-.1, 1.2}, {0, 70}},
    AxesLabel → {"Time of Year", "Temperature (F)"},
    ImageSize → Large];
all3 = Show[p1, p0, p2, p3];
Show[all3, all2, all1, PlotRange -> All]
```

Out[78]=



In[79]:=
```
iris = Import["/Users/longa/Desktop/NKU/classes/000
      2020Spring/mat375/Fletcher/analysis/alldata.csv", "CSV"];
header = iris[[1]];
readData = iris[[2 ;;]];
splitData = Transpose[readData];
data = Transpose[{splitData[[2]], splitData[[3]]}];
```

In[84]:=
```
alllm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
alllm["ParameterTable"]
alllm["RSquared"]
alllm["AdjustedRSquared"]
alllm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]).(weight - Mean[weight])
residuals = alllm["FitResiduals"];
SSE = residuals.residuals
Histogram[residuals]
```

Out[84]= FittedModel[50.0533 + 24.0969 Sin[2 π (−0.305964 + t)]]

... General: $0.209606^{730.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $\frac{1.38936 \times 10^{-305}}{730.5}$ is too small to represent as a normalized machine number; precision may be lost.

Out[85]=

	Estimate	Standard Error	t-Statistic	P-Value
m	50.0533	0.674352	74.2242	0.
a	24.0969	0.953678	25.2674	3.61196×10^{-117}
e	0.305964	0.00629883	48.5747	2.89972×10^{-307}

Out[86]= 0.807982

Out[87]= 0.807588

... General: $0.00262678^{730.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.00262678^{730.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.00262678^{730.5}$ is too small to represent as a normalized machine number; precision may be lost.

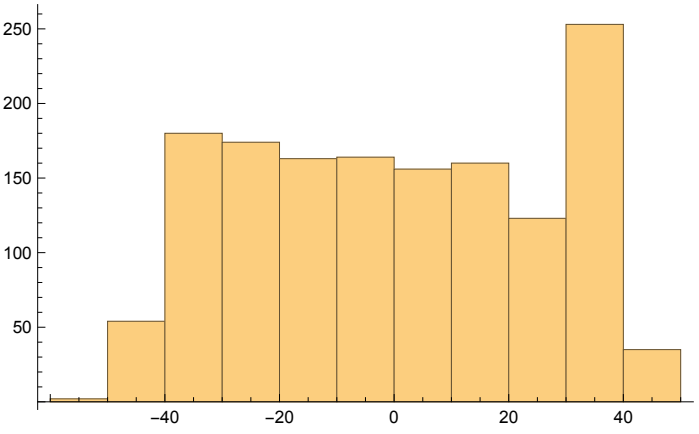
... General: Further output of General::munfl will be suppressed during this calculation.

Out[88]= {{48.7305, 51.3761}, {22.2262, 25.9677}, {0.293608, 0.318319}}

Out[89]= (weight - Mean[weight]).(weight - Mean[weight])

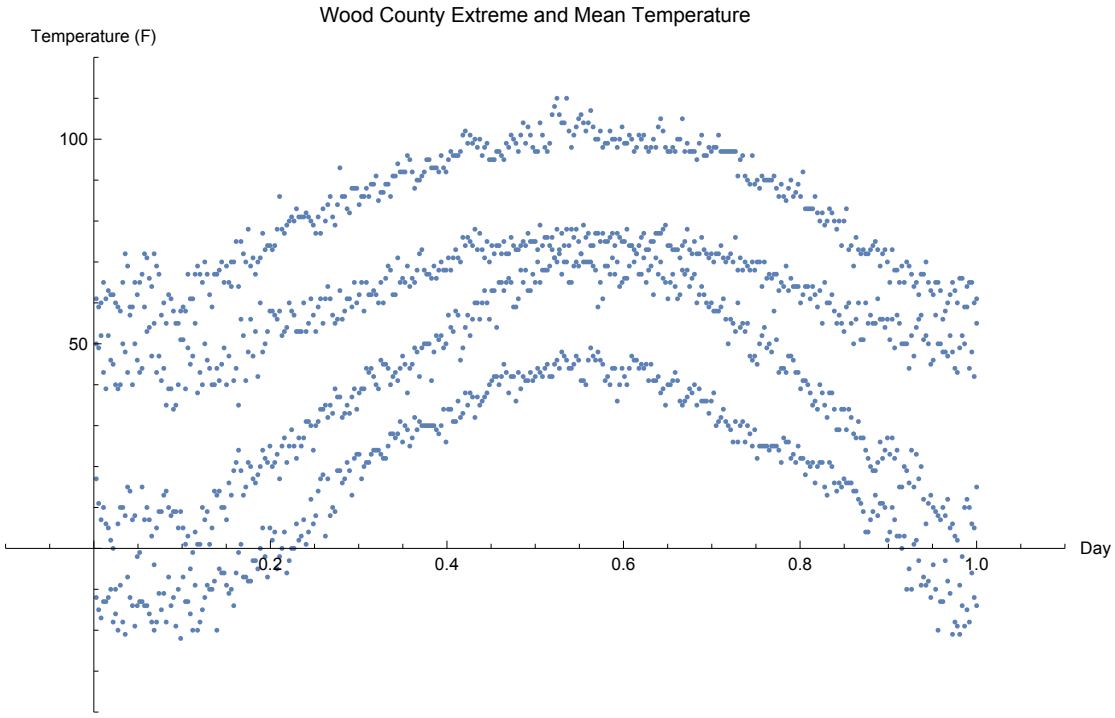
Out[91]= 972 669.

Out[92]=

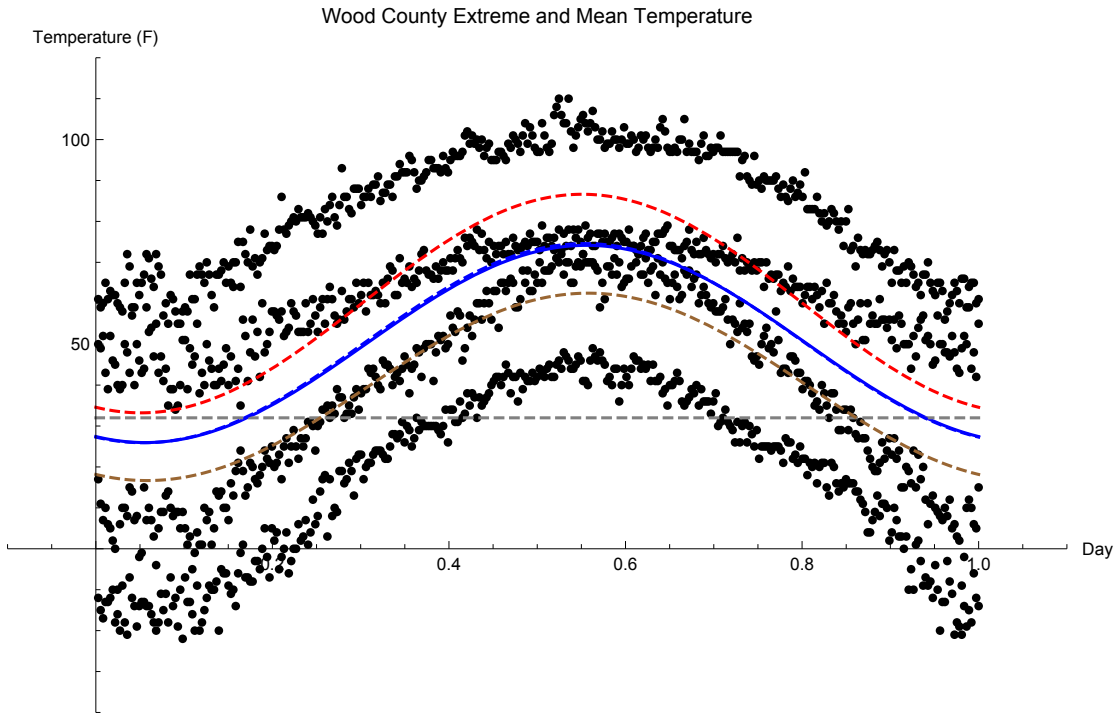


In[93]:=
```
p0 = ListPlot[data, PlotStyle → {Black, PointSize[Medium]}];
p1 = ListPlot[data,
  PlotLabel → "Wood County Extreme and Mean Temperature",
  PlotRange → {{-.1, 1.1}, {-40, 120}},
  AxesLabel → {"Day", "Temperature (F)"},
  ImageSize → Large]
p2 = Plot[alllm[x], {x, 0, 1}, PlotStyle → {Blue}];
p3 = Plot[32, {x, 0, 1}, PlotStyle → {Gray, Dashed}];
p4mu = Plot[meanlm[x], {x, 0, 1}, PlotStyle → {Blue, Dashed}];
p4min = Plot[minlm[x], {x, 0, 1}, PlotStyle → {Brown, Dashed}];
p4max = Plot[maxlm[x], {x, 0, 1}, PlotStyle → {Red, Dashed}];
all = Show[p1, p0, p2, p3, p4mu, p4min, p4max, PlotRange → {{-.1, 1.1}, {-40, 120}}]
all = Show[p1, p0, p2, p3, PlotRange → {{-.1, 1.1}, {-40, 120}}]
```

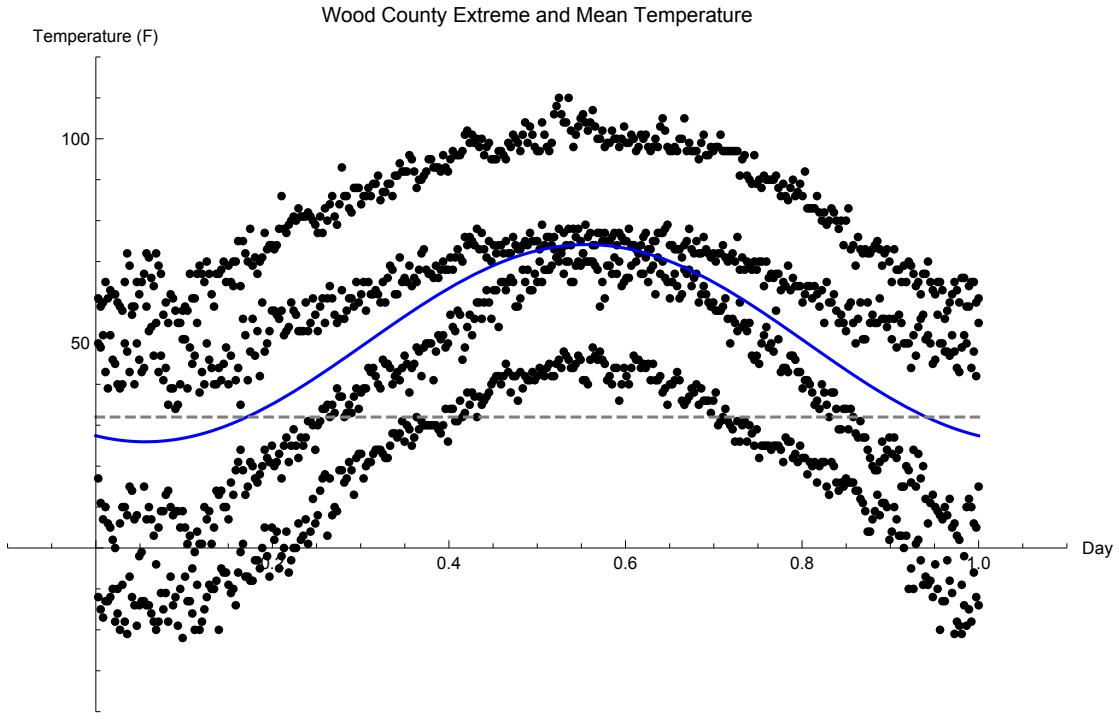
Out[94]=



Out[100]=



Out[101]=



Now let's pull all the separate data sets in, so that we can colorize that graph above:

In[102]:=
```
iris = Import["/Users/longa/Desktop/NKU/classes/000
      2020Spring/mat375/Fletcher/analysis/minmindata.csv", "CSV"];
header = iris[[1]];
minminData = iris[[2 ;;]];
iris = Import["/Users/longa/Desktop/NKU/classes/000
      2020Spring/mat375/Fletcher/analysis/maxmindata.csv", "CSV"];
header = iris[[1]];
maxminData = iris[[2 ;;]];
iris = Import["/Users/longa/Desktop/NKU/classes/000
      2020Spring/mat375/Fletcher/analysis/minmaxdata.csv", "CSV"];
header = iris[[1]];
minmaxData = iris[[2 ;;]];
iris = Import["/Users/longa/Desktop/NKU/classes/000
      2020Spring/mat375/Fletcher/analysis/maxmaxdata.csv", "CSV"];
header = iris[[1]];
maxmaxData = iris[[2 ;;]];
```

In[114]:=
```
data = minminData;
minminlm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
minminlm["ParameterTable"]
minminlm["RSquared"]
minminlm["AdjustedRSquared"]
minminlm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]) . (weight - Mean[weight])
residuals = minminlm["FitResiduals"];
Histogram[residuals]
```

Out[115]= FittedModel[16.8251 + 29.4705 Sin[2 π (−0.307752 + t)]]

... General: $0.0114573^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

Out[116]=

	Estimate	Standard Error	t-Statistic	P-Value
m	16.8251	0.22769	73.895	7.55125×10^{-221}
a	29.4705	0.322002	91.5227	7.49778×10^{-253}
e	0.307752	0.00173897	176.974	0.

Out[117]= 0.974436

Out[118]= 0.974225

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

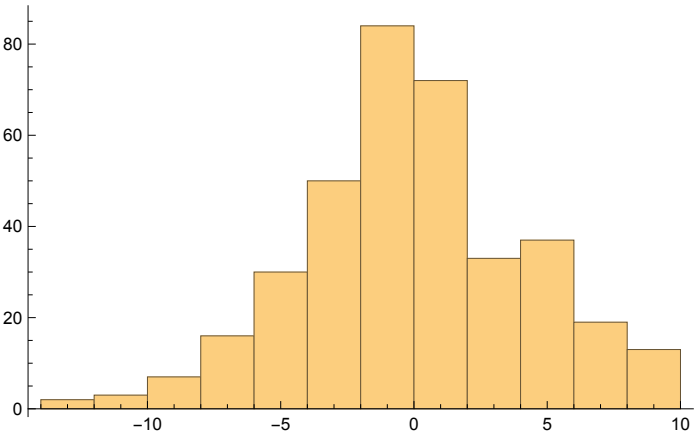
... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: Further output of General::munfl will be suppressed during this calculation.

Out[119]= {{16.3774, 17.2729}, {28.8372, 30.1037}, {0.304333, 0.311172}}

Out[120]= (weight - Mean[weight]) . (weight - Mean[weight])

Out[122]=



In[123]:=
```
data = maxminData;
maxminlm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
maxminlm["ParameterTable"]
maxminlm["RSquared"]
maxminlm["AdjustedRSquared"]
maxminlm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]) . (weight - Mean[weight])
residuals = maxminlm["FitResiduals"];
Histogram[residuals]
```

Out[124]= FittedModel[61.5738 + 15.5851 Sin[2 π (−0.310813 + t)]]

... General: $0.00383623^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

Out[125]=

	Estimate	Standard Error	t-Statistic	P-Value
m	61.5738	0.200553	307.02	0.
a	15.5851	0.283625	54.9497	5.16897×10^{-178}
e	0.310813	0.00289638	107.311	4.8005×10^{-277}

Out[126]= 0.996282

Out[127]= 0.996252

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

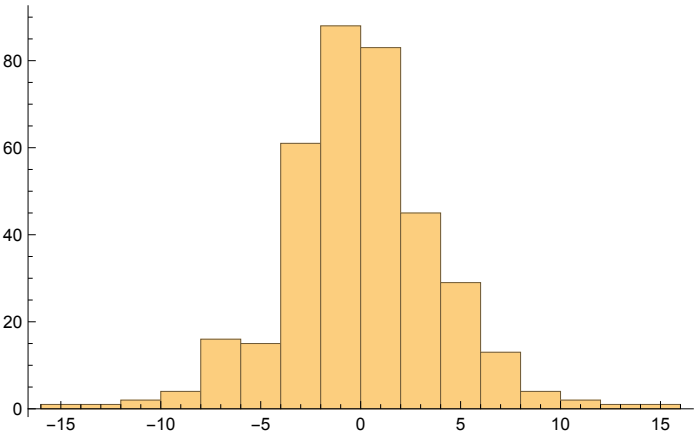
... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: Further output of General::munfl will be suppressed during this calculation.

Out[128]= {{61.1794, 61.9682}, {15.0273, 16.1428}, {0.305117, 0.316509}}

Out[129]= (weight - Mean[weight]) . (weight - Mean[weight])

Out[131]=



In[132]:=
```
data = minmaxData;
minmaxlm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
minmaxlm["ParameterTable"]
minmaxlm["RSquared"]
minmaxlm["AdjustedRSquared"]
minmaxlm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]) . (weight - Mean[weight])
residuals = minmaxlm["FitResiduals"];
Histogram[residuals]
```

Out[133]= FittedModel[38.1831 + 31.3381 Sin[2 π (-0.3103 + t)]]

... General: $0.0125876^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.00985921^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

Out[134]=

	Estimate	Standard Error	t-Statistic	P-Value
m	38.1831	0.226276	168.745	0.
a	31.3381	0.320003	97.9307	4.17672×10^{-263}
e	0.3103	0.00162518	190.933	0.

Out[135]= 0.990554

Out[136]= 0.990476

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

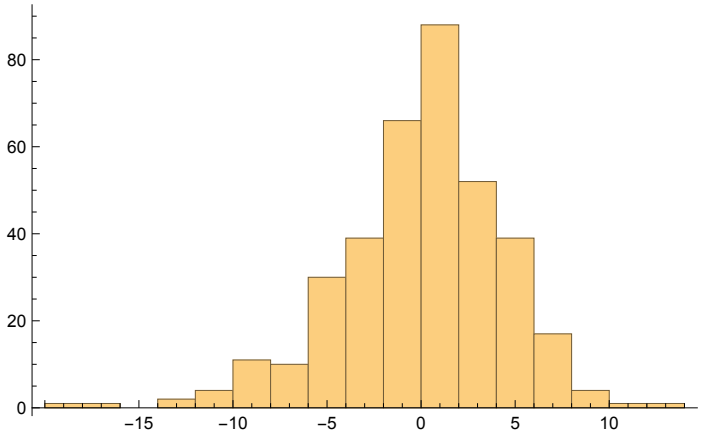
... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: Further output of General::munfl will be suppressed during this calculation.

Out[137]= {{37.7381, 38.628}, {30.7088, 31.9674}, {0.307104, 0.313496}}

Out[138]= (weight - Mean[weight]) . (weight - Mean[weight])

Out[140]=



In[141]:=
```
data = maxmaxData;
maxmaxlm = NonlinearModelFit[data, m + a * Sin[2 π * (t - e)],
  {{m, 50}, {a, 25}, {e, .26}}, t, MaxIterations → 100 000]
maxmaxlm["ParameterTable"]
maxmaxlm["RSquared"]
maxmaxlm["AdjustedRSquared"]
maxmaxlm["ParameterConfidenceIntervals"]
SST = (weight - Mean[weight]) . (weight - Mean[weight])
residuals = maxmaxlm["FitResiduals"];
Histogram[residuals]
```

Out[142]= FittedModel[83.6311 + 20.0835 Sin[2 π (-0.292796 + t)]]

... General: $0.00189284^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.0190623^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

Out[143]=

	Estimate	Standard Error	t-Statistic	P-Value
m	83.6311	0.191154	437.507	0.
a	20.0835	0.270332	74.2919	1.2186×10^{-221}
e	0.292796	0.00214229	136.674	3.00503×10^{-314}

Out[144]= 0.99816

Out[145]= 0.998145

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

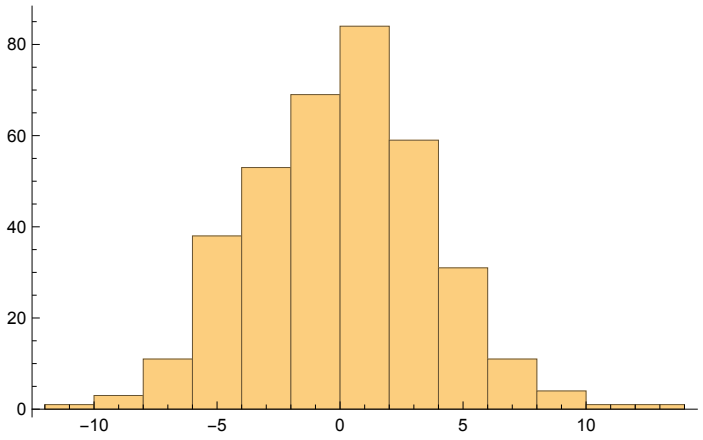
... General: $0.0105412^{181.5}$ is too small to represent as a normalized machine number; precision may be lost.

... General: Further output of General::munfl will be suppressed during this calculation.

Out[146]= {{83.2552, 84.0071}, {19.5519, 20.6151}, {0.288583, 0.297009}}

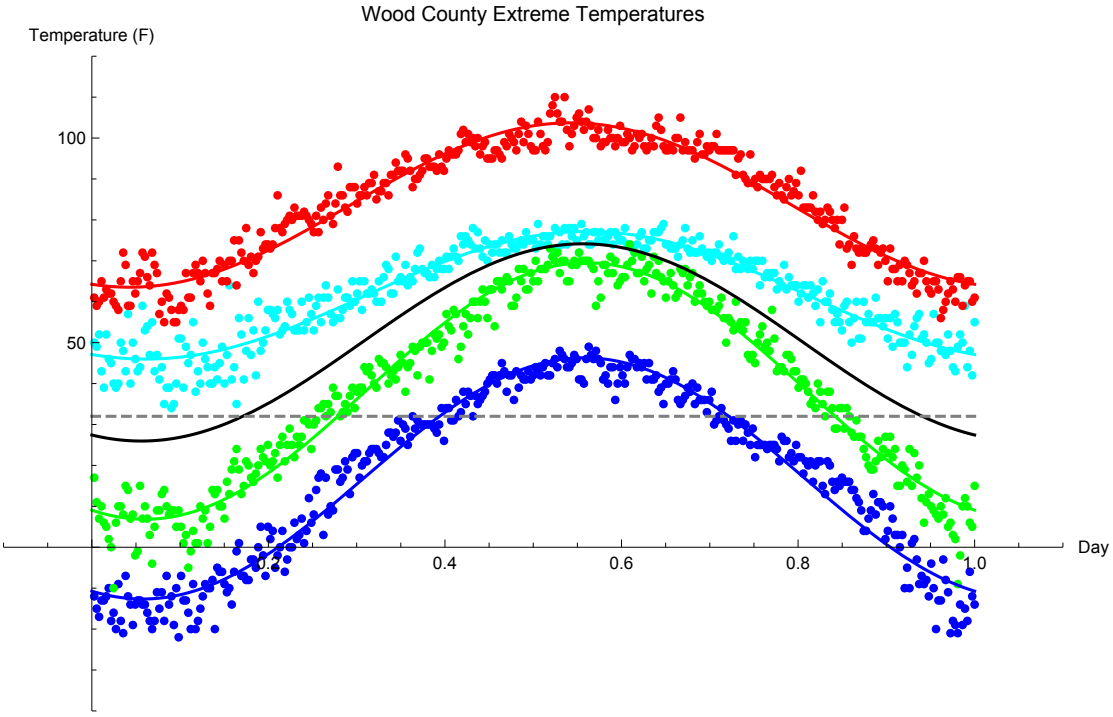
Out[147]= (weight - Mean[weight]) . (weight - Mean[weight])

Out[149]=



In[150]:=
```
p0a = ListPlot[minminData, PlotStyle → {Blue, PointSize[Medium]}];
p0b = ListPlot[maxminData, PlotStyle → {Cyan, PointSize[Medium]}];
p0c = ListPlot[minmaxData, PlotStyle → {Green, PointSize[Medium]}];
p0d = ListPlot[maxmaxData, PlotStyle → {Red, PointSize[Medium]}];
p1 = ListPlot[minminData,
    PlotLabel → "Wood County Extreme Temperatures",
    PlotRange → {{-.1, 1.1}, {-40, 120}},
    AxesLabel → {"Day", "Temperature (F)"},
    ImageSize → Large];
p2 = Plot[alllm[x], {x, 0, 1}, PlotStyle → {Black}];
p3 = Plot[32, {x, 0, 1}, PlotStyle → {Gray, Dashed}];
p4minmin = Plot[minminlm[x], {x, 0, 1}, PlotStyle → {Blue}];
p4maxmin = Plot[maxminlm[x], {x, 0, 1}, PlotStyle → {Cyan}];
p4minmax = Plot[minmaxlm[x], {x, 0, 1}, PlotStyle → {Green}];
p4maxmax = Plot[maxmaxlm[x], {x, 0, 1}, PlotStyle → {Red}];
allDataTogether = Show[p1, p0a, p0b, p0c, p0d, p2, p3, p4maxmin,
  p4maxmax, p4minmin, p4minmax, PlotRange → {{-.1, 1.1}, {-40, 120}}]
```

Out[161]=



In[162]:=
```
Manipulate[
 Show[allDataTogether,
  ListPlot[{{day, alllm[day]}}], Plot[alllm[day] +
    (maxmaxlm[day] - maxminlm[day]) * PDF[NormalDistribution[day, sigma], x] / PDF[
      NormalDistribution[day, sigma], day], {x, day - .3, day + .3}, PlotRange → All]
 ],
 {{sigma, 0.037}, 0.01, .3},
 {{day, .3}, 0, 1}
]
```

Out[162]=