Common

Data

In[12]:=
```
η = 0.001;
ηa = 0.05;
ηA = 0.001;
```

In[15]:=
```
error[w1_, w2_] := 3 * w1^2 + 10 * w2^2
```

In[16]:=
```
grad1 = D[error[w1, w2], {w1}]
```

Out[16]=
```
6 w1
```

In[17]:=
```
grad2 = D[error[w1, w2], {w2}]
```

Out[17]=
```
20 w2
```

In[18]:=
```
w1Start = -15;
w2Start = 20;
```

In[20]:=
```
iterations = 700;
```

In[21]:=
```
clear[] := Block[{},
  (* This removes all subscript values
   (https://mathematica.stackexchange.com/a/38650/41589) *)
  Clear["Subscript"];

  grad1[w1_, w2_] = grad1;
  grad2[w1_, w2_] = grad2;
 ]
```

In[22]:=
```
wSeries[alpha_] := Block[{},
  clear[];

  mi_[0] := 0;
  w1[0] := w1Start;
  w2[0] := w2Start;

  mi_[t_] := mi[t] = alpha * mi[t - 1] + gradi[w1[t - 1], w2[t - 1]];
  wi_[t_] := wi[t] = wi[t - 1] - η * mi[t];

  Table[{w1[t], w2[t], error[w1[t], w2[t]]}, {t, 0, iterations}]
 ]
```

In[23]:=
```
RMSProb[beta_] := Block[{},
  clear[];

  si_[0] := 0;
  w1[0] := w1Start;
  w2[0] := w2Start;

  si_[t_] := si[t] = beta * si[t - 1] + (1 - beta) * gradi[w1[t - 1], w2[t - 1]]^2;
  wi_[t_] := wi[t] = wi[t - 1] - ηa * gradi[w1[t - 1], w2[t - 1]] / Sqrt[si[t] + 10^-8];

  Table[{w1[t], w2[t], error[w1[t], w2[t]]}, {t, 0, iterations}]
 ]
```

In[24]:=
```
adam[beta1_, beta2_, eta_] := Block[{},
  clear[];

  s_i_[0] := 0;
  m_i_[0] := 0;
  w_1[0] := w1Start;
  w_2[0] := w2Start;

  m_i_[t_] := m_i[t] = beta1 * m_i[t - 1] + (1 - beta1) * grad_i[w_1[t - 1], w_2[t - 1]];
  s_i_[t_] := s_i[t] = beta2 * s_i[t - 1] + (1 - beta2) * grad_i[w_1[t - 1], w_2[t - 1]]^2;

  w_i_[t_] := w_i[t] = w_i[t - 1] - eta * (m_i[t]/(1-beta1^t)) / Sqrt[s_i[t]/(1-beta2^t) + 10^-8];

  Table[{w_1[t], w_2[t], error[w_1[t], w_2[t]]}, {t, 0, iterations}]
 ]
```

Momentum example

Classical gradient descent

In[25]:=
```
wClassic = wSeries[0][[1 ;; 3]];
Round[wClassic, 0.01]
```

Out[26]= {{-15., 20., 4675.}, {-14.91, 19.6, 4508.52}, {-14.82, 19.21, 4348.42}}

In[27]:=
```
{grad_1[w_1[0], w_2[0]], grad_2[w_1[0], w_2[0]]}
```

Out[27]= {-90, 400}

In[28]:=
```
Round[{grad_1[w_1[1], w_2[1]], grad_2[w_1[1], w_2[1]]}, 0.01]
```

Out[28]= {-89.46, 392.}

Momentum optimization

In[29]:=
```
wMomentum = wSeries[0.9][[1 ;; 3]];
Round[wMomentum, 0.01]
```

Out[30]= {{-15., 20., 4675.}, {-14.91, 19.6, 4508.52}, {-14.74, 18.85, 4204.23}}

In[31]:=
```
{grad_1[w_1[0], w_2[0]], grad_2[w_1[0], w_2[0]]}
```

Out[31]= {-90, 400}

In[32]:=
```
{m_1[1], m_2[1]}
```

Out[32]= {-90., 400.}

The gradients are the same in the first two iterations as only of the third iterations the weights differ.

In[33]:=
```
{grad_1[w_1[1], w_2[1]], grad_2[w_1[1], w_2[1]]}
```

Out[33]= {-89.46, 392.}

In[34]:=
```
Round[{m_1[2], m_2[2]}, 0.01]
```

Out[34]= {-170.46, 752.}

Visualization

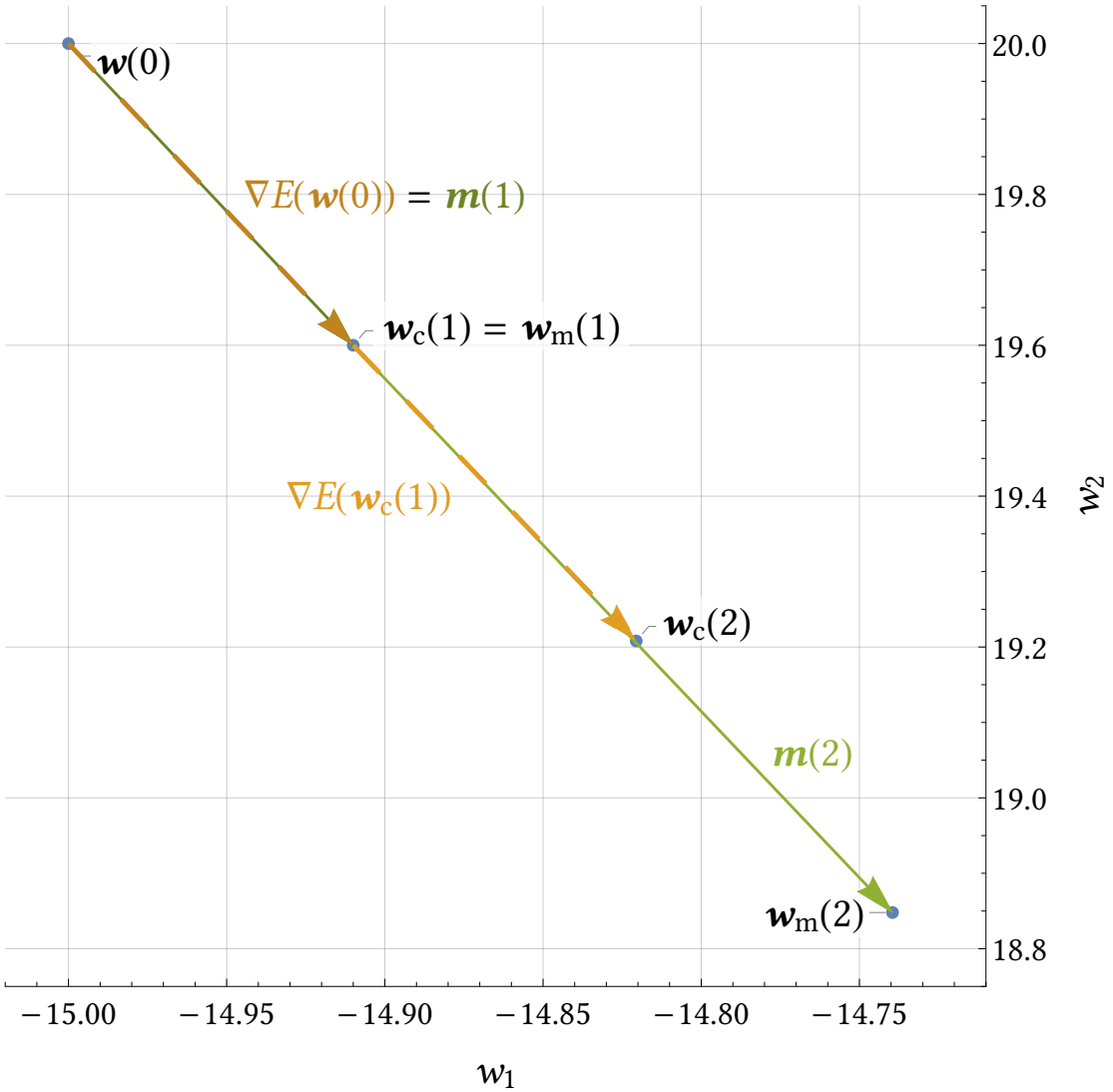
In[35]:=
```
wClassic = wClassic[[ ;; , 1 ;; 2]]
```
Out[35]=
```
{{-15, 20}, {-14.91, 19.6}, {-14.8205, 19.208}}
```

In[36]:=
```
wMomentum = wMomentum[[ ;; , 1 ;; 2]]
```
Out[36]=
```
{{-15, 20}, {-14.91, 19.6}, {-14.7395, 18.848}}
```

In[37]:=
```
colorVec1B = ;
colorVec1A = Darker[colorVec1B, 0.15];
colorVec2B = ;
colorVec2A = Darker[colorVec2B, 0.25];
```

In[41]:=
```
Show[
 ListPlot[{
   Callout[wClassic[[1]], Row[{bi["w"], "(0)"}]],
   Callout[wClassic[[2]],
    Row[{Subscript[bi["w"], "c"], "(1) = ", Subscript[bi["w"], "m"], "(1)"}]],
   Callout[wClassic[[3]], Row[{Subscript[bi["w"], "c"], "(2)"}]],
   Callout[wMomentum[[3]], Row[{Subscript[bi["w"], "m"], "(2)"}], Before]
  },
  PlotTheme → "myTheme",
  GridLines → Automatic,
  PlotRange → {{-15.02, -14.71}, {18.75, 20.05}},
  PlotRangePadding → None,
  AspectRatio → 1,
  FrameLabel → {"w1", None, None, "w2"},
  Frame → {True, False, False, True},
  FrameTicks → {{None, All}, {All, None}}
 ],
 Graphics[{
   {
    AbsoluteThickness[1.5],
    colorVec2A,
    Arrow[{wMomentum[[1]], wMomentum[[2]]}],
    colorVec2B,
    Arrow[{wMomentum[[2]], wMomentum[[3]]}],
    Style[Text[Row[{bi["m"], "(2)"}], {-14.765, 19.06}],
     FontFamily → "Libertinus Serif", FontSize → 22]
   },
   Style[
    Text[
     Row[{
       Style[Row[{"∇", it["E"], "(", bi["w"], "(0))"}], colorVec1A],
       " = ",
       Style[Row[{bi["m"], "(1)"}], colorVec2A]
      }],
     {-14.9, 19.8}
    ],
    FontFamily → "Libertinus Serif",
    FontSize → 22
   ],
   {
    AbsoluteThickness[2.5],
    AbsoluteDashing[20],
    colorVec1A,
    Arrow[{wClassic[[1]], wClassic[[2]]}],
    colorVec1B,
    Arrow[{wClassic[[2]], wClassic[[3]]}],
    Style[Text[Row[{"∇", it["E"], "(", Subscript[bi["w"], "c"], "(1))"}], {-14.905, 19.4}],
     FontFamily → "Libertinus Serif", FontSize → 22]
   }
  }]
]
```

Out[41]=



RMSProb example

Adaptive learning rate

In[42]:=
```
rms = RMSProb[0.9][[1 ;; 3, 1 ;; 2]];
rmsRound = Round[rms, 0.01]
```

Out[43]= {{-15., 20.}, {-14.84, 19.84}, {-14.73, 19.73}}

In[44]:=
```
{grad1[w1[0], w2[0]], grad2[w1[0], w2[0]]}
```

Out[44]= {-90, 400}

In[45]:=
```
{s1[1], s2[1]}
```

Out[45]= {810., 16 000.}

In[46]:=
```
rmsGrad1 = Round[
  {grad1[rmsRound[[2, 1]], rmsRound[[2, 2]]], grad2[rmsRound[[2, 1]], rmsRound[[2, 2]]]}, 0.01]
```

Out[46]= {-89.04, 396.8}

In[47]:=
```
rmsScale2 = Round[0.9 * {s1[1], s2[1]} + 0.1 * rmsGrad1^2, 0.01] // FullForm
```

Out[47]//FullForm= List[1521.81`, 30145.02`]

In[48]:=
```
Round[{s1[2], s2[2]}, 0.01]
```

Out[48]= {1522.01, 30 148.}

In[49]:=
```
Round[Round[rms, 0.01][[2]] - 0.05 * rmsGrad1/Sqrt[rmsScale2 + 10^-8], 0.01]
```

Out[49]=
```
{Round[-14.84 + 4.452/Sqrt[1/100000000 + List[1521.81`, 30145.02`]], 0.01],
 Round[19.84 - 19.84/Sqrt[1/100000000 + List[1521.81`, 30145.02`]], 0.01]}
```

Visualization

In[50]:=
```
wClassic
```

Out[50]=
```
{{-15, 20}, {-14.91, 19.6}, {-14.8205, 19.208}}
```

In[51]:=
```
rms
```

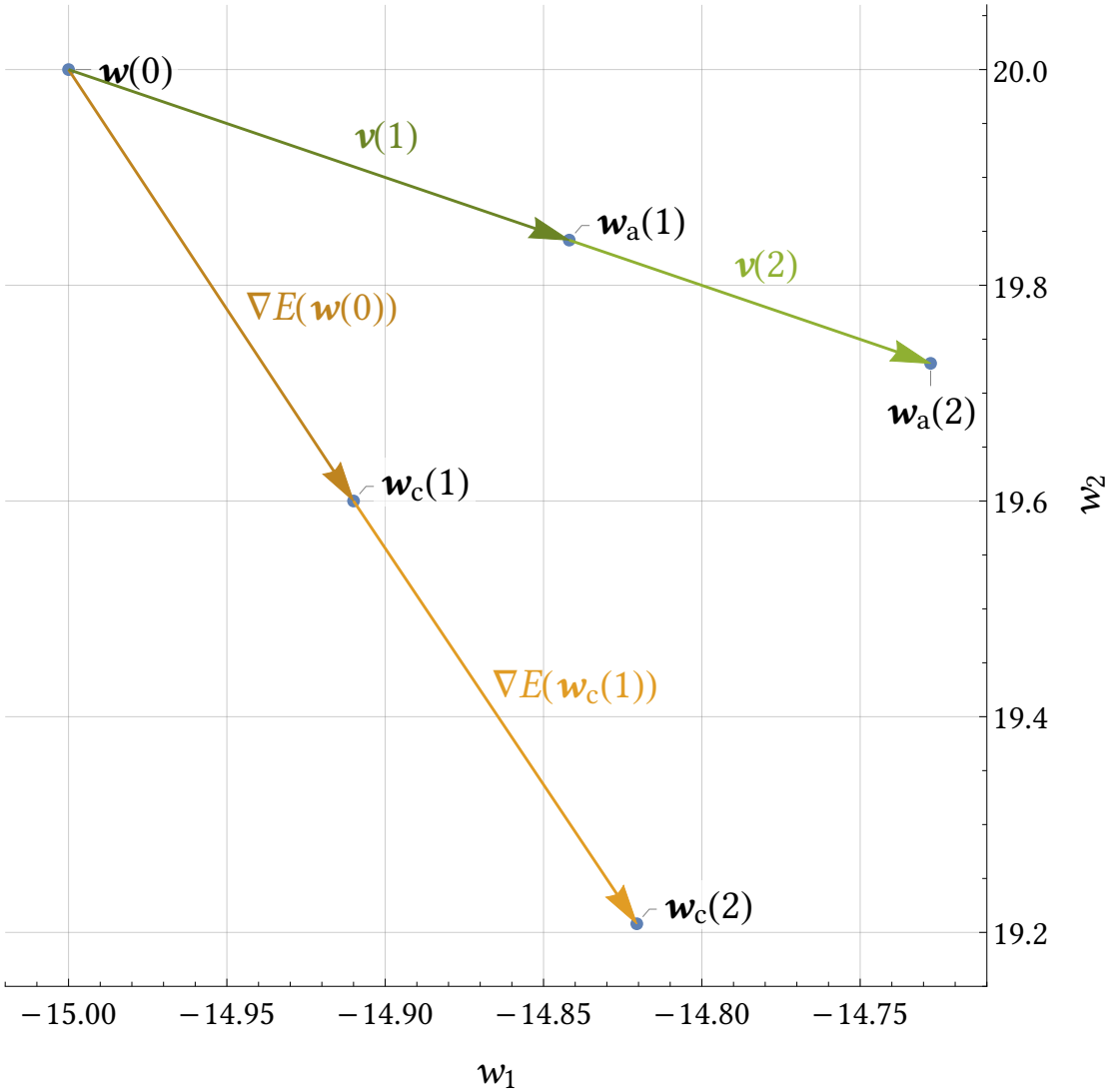
Out[51]=
```
{{-15, 20}, {-14.8419, 19.8419}, {-14.7278, 19.7276}}
```

In[52]:=
```
Show[
 ListPlot[{
   Callout[wClassic[[1]], Row[{bi["w"], "(0)"}], After],
   Callout[wClassic[[2]], Row[{Subscript[bi["w"], "c"], "(1)"}]],
   Callout[wClassic[[3]], Row[{Subscript[bi["w"], "c"], "(2)"}]],
   Callout[rms[[2]], Row[{Subscript[bi["w"], "a"], "(1)"}]],
   Callout[rms[[3]], Row[{Subscript[bi["w"], "a"], "(2)"}], Below]
  },
  PlotTheme → "myTheme",
  GridLines → Automatic,
  PlotRange → {{-15.02, -14.71}, {19.15, 20.06}},
  PlotRangePadding → None,
  AspectRatio → 1,
  FrameLabel → {"w1", None, None, "w2"},
  Frame → {True, False, False, True},
  FrameTicks → {{None, All}, {All, None}}
 ],
 Graphics[{
   AbsoluteThickness[1.5],

   colorVec1A,
   Arrow[{wClassic[[1]], wClassic[[2]]}],
   Style[Text[Row[{"∇", it["E"], "(", bi["w"], "(0))"}], {-14.92, 19.78}],
    FontFamily → "Libertinus Serif", FontSize → 22],
   colorVec1B,
   Arrow[{wClassic[[2]], wClassic[[3]]}],
   Style[Text[Row[{"∇", it["E"], "(", Subscript[bi["w"], "c"], "(1))"}], {-14.84, 19.43}],
    FontFamily → "Libertinus Serif", FontSize → 22],

   colorVec2A,
   Arrow[{rms[[1]], rms[[2]]}],
   Style[Text[Row[{bi["v"], "(1)"}], {-14.9, 19.94}],
    FontFamily → "Libertinus Serif", FontSize → 22],
   colorVec2B,
   Arrow[{rms[[2]], rms[[3]]}],
   Style[Text[Row[{bi["v"], "(2)"}], {-14.78, 19.82}],
    FontFamily → "Libertinus Serif", FontSize → 22]
  }]
]
```

Out[52]=



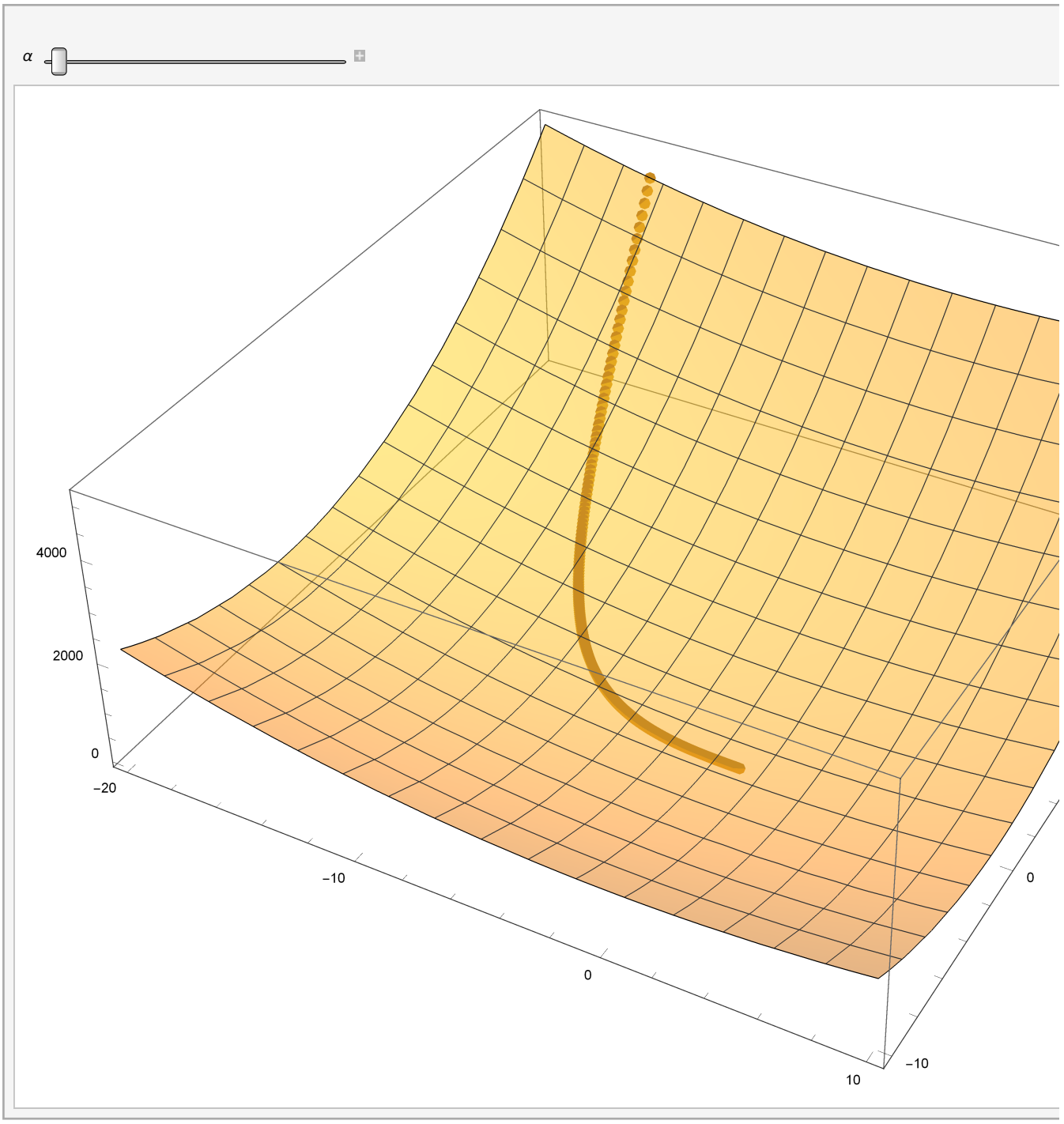
Testing

Plots

Trajectories

In[62]:=
```
Manipulate[
 Show[
  Plot3D[error[w1, w2], {w1, -20, 10}, {w2, -10, 20}, ImageSize → 800, PlotStyle → Opacity[0.5]],
  ListPointPlot3D[{wSeries[0], wSeries[α]}, PlotStyle → PointSize[0.01]]
 ]
 , {α, 0, 1}, TrackedSymbols :→ {α}]
```

Out[62]=



In[63]:=
```
min = FindMinimum[error[w1, w2], {w1, w2}][[2, ;; , 2]]
```

Out[63]= {0., 0.}

In[64]:=
```
alphas = {0, 0.6, 0.9};
```

In[65]:=
```
color1 = Darker[■, 0.25];
color2 = Darker[■, 0.25];
color3 = Darker[■, 0.25];
lineColors = {color1, color2, color3}
```

Out[68]= {■, ■, ■}

In[69]:=
```
colorMin = ■;
```

In[70]:=
```
plotBackground = DensityPlot[error[w1, w2], {w1, -21, 21}, {w2, -21, 21},
   PlotTheme → "myTheme",
   PlotLegends → Placed[
     BarLegend[Automatic,
      LabelStyle → fontTicks,
      LegendMargins → {{50, 0}, {0, 0}},
      LegendMarkerSize → 520,
      LegendLabel → Style[Row[{it["E"], "(", bi["w"], ")"}], fontLabels]
     ],
     Above
    ],
   FrameLabel → {"w1", "w2"},
   PlotPoints → 100,
   MeshFunctions → {#3 &, #3 &},
   Mesh → Automatic,
   PerformanceGoal → "Quality"
  ];
plotMin = ListPlot[{min},
   PlotTheme → "myTheme",
   PlotStyle → Directive[colorMin, PointSize[0.02]]
  ];
```

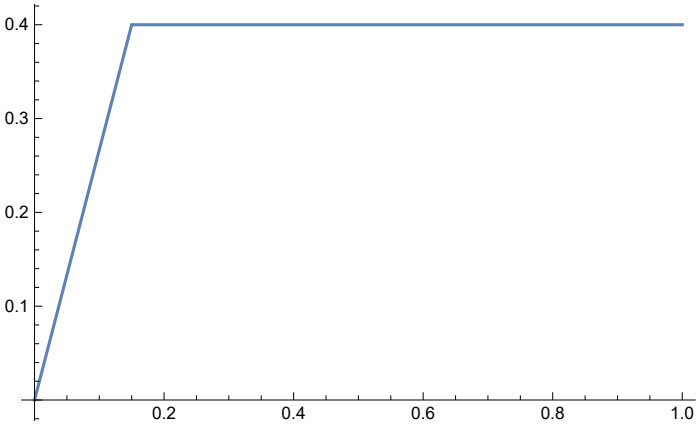
In[72]:=
```
momentSeries = Table[wSeries[α][[ ;; , 1 ;; 2]], {α, alphas}];
momentSeries // Dimensions
```

Out[73]= {3, 701, 2}

In[74]:=
```
lightMax = 0.4;
colorMap[v_, tThresh_] := { v/tThresh * lightMax   v ≤ tThresh
                            lightMax               True
```

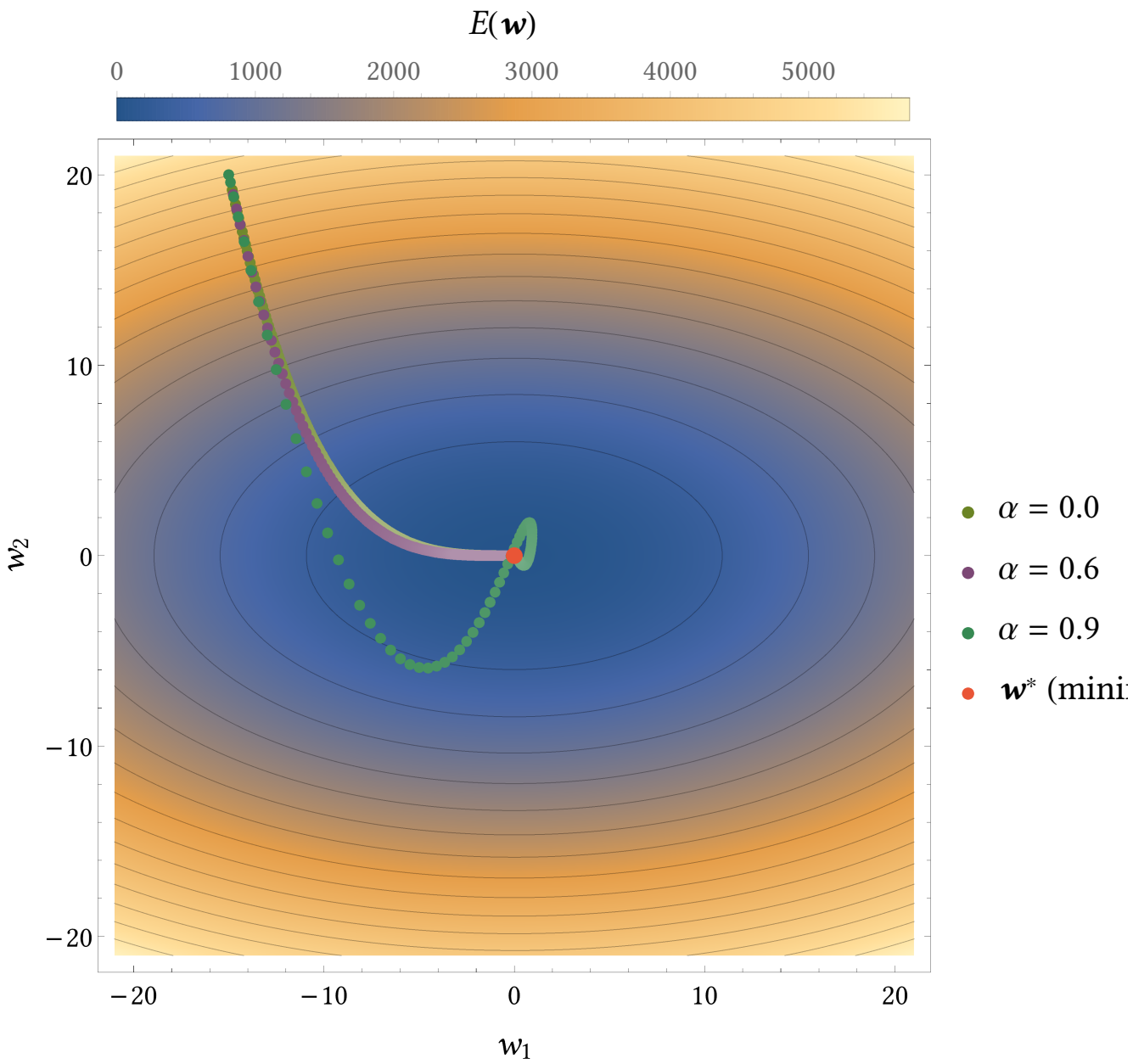
In[76]:=
```
Plot[colorMap[v, 0.15], {v, 0, 1}, PlotRange → All]
```

Out[76]=



In[77]:=
```
plotTrajectoryMomentum = Show[
  plotBackground,
  ListPlot[Table[
    Style[
     momentSeries[[s, t, ;;]],
     Lighter[lineColors[[s]], colorMap[t/iterations, 0.15]]
    ],
    {s, 1, Length[momentSeries]},
    {t, 1, iterations}
   ],
   PlotTheme → "myTheme",
   PlotRange → All,
   PlotLegends → PointLegend[
     Join[lineColors, {colorMin}],
     Join[Table["α = " <> ToString[equalNumberForm[α, 1, 1]], {α, alphas}],
      {Row[{Superscript[bi["w"], "*"], " (minimum)"}]}],
     LegendMarkerSize → 15
    ],
   BaseStyle → {FontOpacity → 0.999}
  ],
  plotMin
]
```

Out[77]=



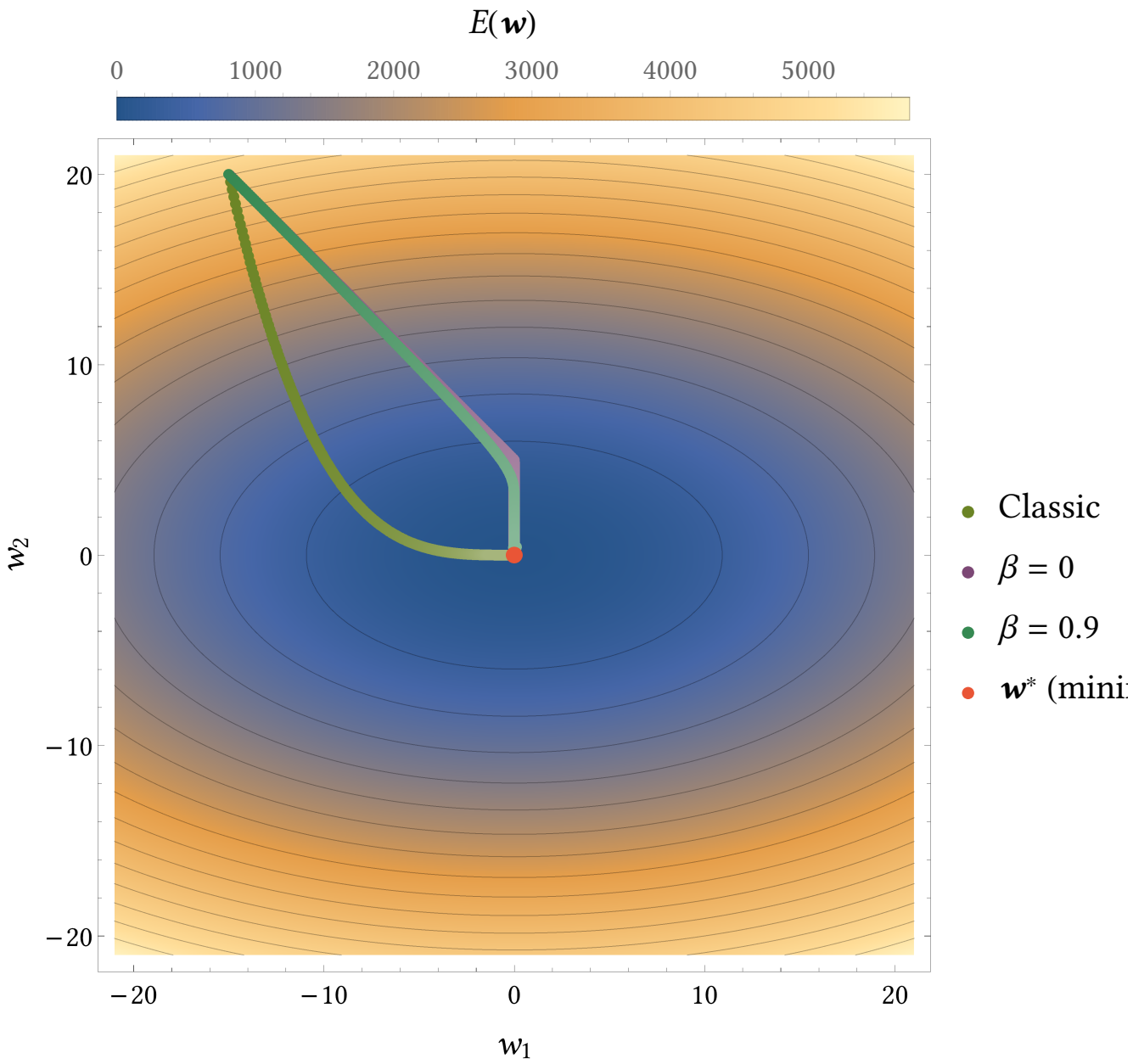
In[78]:=
```
(*plotTrajectoryMomentum[[1]]=rasterizeBackground[plotTrajectoryMomentum[[1]]];
Export[
 FileNameJoin[{NotebookDirectory[],"Optimizers_MomentumTrajectory.pdf"}],
 plotTrajectoryMomentum
];*)
```

In[79]:=
```
adaptiveSeries =
  {wSeries[0][[ ;; , 1 ;; 2]], RMSProb[0][[ ;; , 1 ;; 2]], RMSProb[0.9][[ ;; , 1 ;; 2]]};
adaptiveSeries // Dimensions
```

Out[80]=
```
{3, 701, 2}
```

In[81]:=
```
plotTrajectoryAdaptive = Show[
  plotBackground,
  ListPlot[Table[
    Style[
     adaptiveSeries[[s, t, ;;]],
     Lighter[lineColors[[s]], colorMap[t / iterations, 400 / 700]]
    ],
    {s, 1, Length[adaptiveSeries]},
    {t, 1, iterations}
   ],
   PlotTheme → "myTheme",
   PlotRange → All,
   PlotLegends → PointLegend[
     Join[lineColors, {colorMin}],
     Join[{"Classic", "β = 0", "β = 0.9"}, {Row[{Superscript[bi["w"], "*"], " (minimum)"}]}],
     LegendMarkerSize → 15
    ],
   BaseStyle → {FontOpacity → 0.999}
  ],
  plotMin
 ]
```

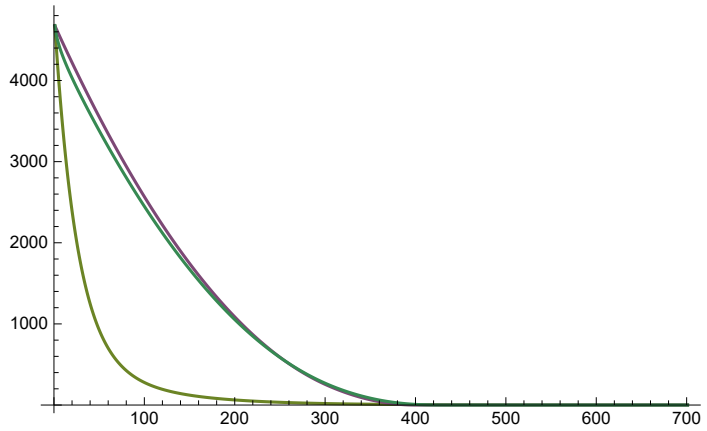
Out[81]=



In[82]:=
```
(*plotTrajectoryAdaptive[[1]]=rasterizeBackground[plotTrajectoryAdaptive[[1]]];
Export[
 FileNameJoin[{NotebookDirectory[],"Optimizers_AdaptiveTrajectory.pdf"}],
 plotTrajectoryAdaptive
];*)
```

In[83]:=
```
ListLinePlot[{wSeries[0][[ ;; , 3]], RMSProb[0][[ ;; , 3]], RMSProb[0.9][[ ;; , 3]]},
 PlotStyle → lineColors]
```

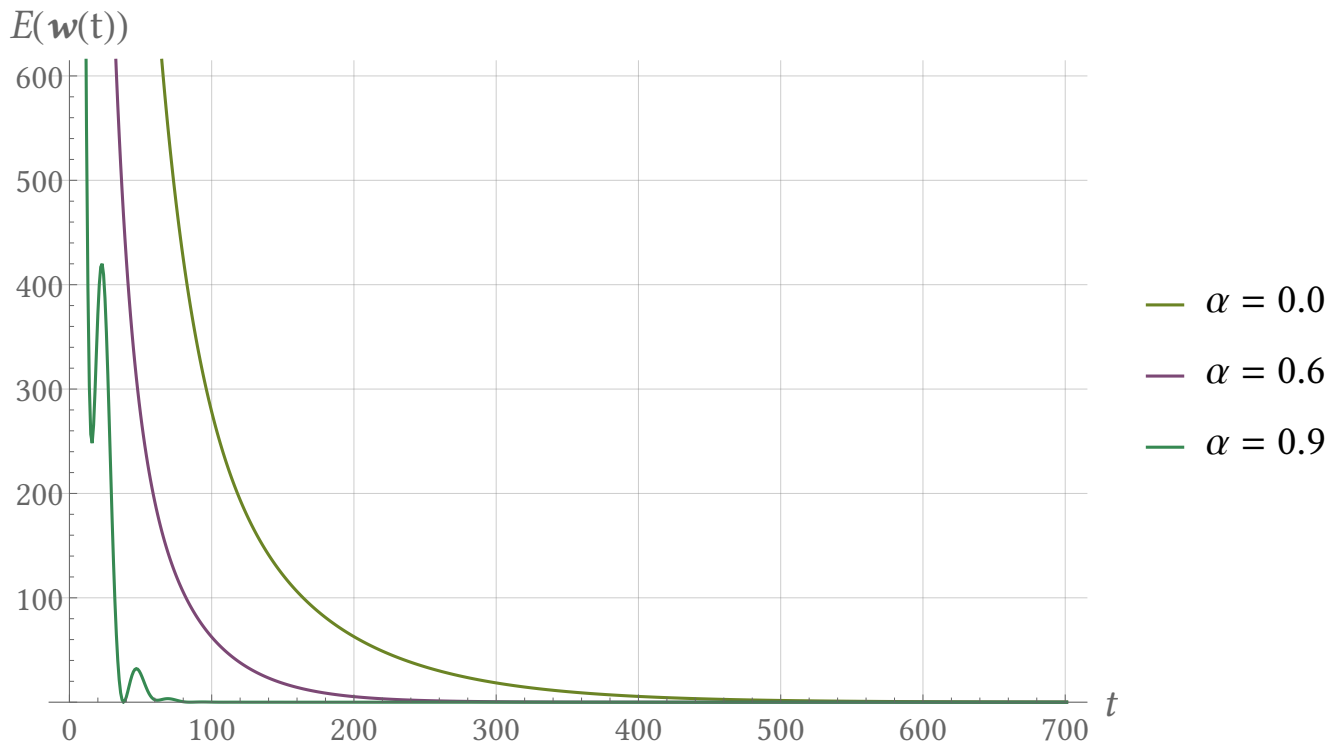
Out[83]=



Speed improvements

In[84]:=
```
plotErrorComparison = ListLinePlot[Table[wSeries[α][[ ;; , 3]], {α, {0, 0.5, 0.9}}],
  PlotTheme → "myTheme",
  PlotRange → {Automatic, {0, 615}},
  PlotStyle → lineColors,
  AxesLabel → {it["t"], Row[{it["E"], "(", bi["w"], "(t))"}]},
  GridLines → Automatic,
  PlotLegends → Table["α = " <> ToString[equalNumberForm[α, 1, 1]], {α, alphas}]
 ]
```

Out[84]=



In[85]:=
```
(*Export[FileNameJoin[{NotebookDirectory[],"MomentumOptimizer_SpeedImprovements.svg"}],
  plotErrorComparison];*)
```

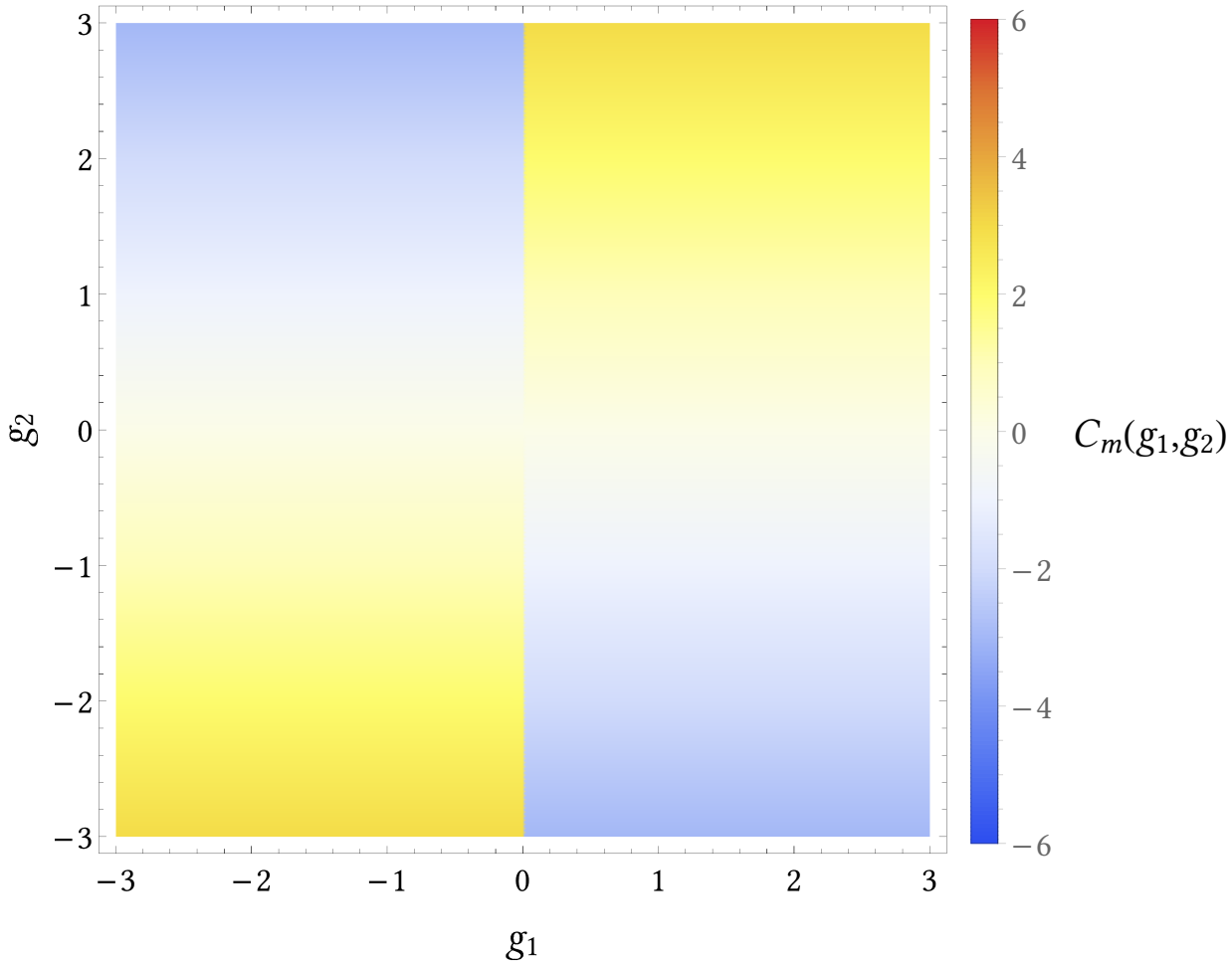
Concept of acceleration/deceleration

In[86]:=
```
Cmα_[g1_, g2_] := { (α * g1 + g2) - g1   g1 ≥ 0
                  { g1 - (α * g1 + g2)   g1 < 0
```

In[87]:=
```
sampleData = Table[Cmα[g1, g2], {α, 0, 1}, {g1, -3, 3, 0.1}, {g2, -3, 3, 0.1}];
maxAbsValue = Max[Abs[Min@sampleData], Abs[Max@sampleData]];
lower = -maxAbsValue;
upper = maxAbsValue;
plotAccDec[α_] := DensityPlot[Cmα[g1, g2], {g1, -3, 3}, {g2, -3, 3},
  PlotLegends → BarLegend[{Automatic, {lower, upper}},
    LegendMarkerSize → 462,
    LegendMargins → {{0, 0}, {42, 0}},
    LabelStyle → fontTicks,
    LegendLabel → Placed[
      Style[Row[{Subscript[it["C"], it["m"]], "(g1,g2)"}], fontLabels],
      After
     ]
   ],
  PlotTheme → "myTheme",
  ImageSize → 500,
  ColorFunctionScaling → False,
  ColorFunction → (ColorData["TemperatureMap"][(# - lower)/(upper - lower)] &),
  Exclusions → None,
  PlotPoints → 100,
  FrameLabel → {"g1", "g2"}
 ]
```

In[92]:=
```
plotCases = plotAccDec[1]
```

Out[92]=



In[93]:=
```
(*Export[
  FileNameJoin[{NotebookDirectory[],"Optimizers_MomentumCases.pdf"}],
  rasterizeBackground[plotCases]
 ];*)
```

Parameter β in the adaptive learning scheme

In[94]:=
```
plotComp[beta_] := Module[{comp, β, minValue, maxValue, scale, fontTicks, fontLabels},
  comp[g1_, g2_] := Abs[g2 / Sqrt[β * ((1 - β) * g1^2) + (1 - β) * g2^2 + 10^-8]] - Abs[g2];

  minValue = Minimize[{comp[g1, g2] /. {β → 0}, -6 ≤ g1 ≤ 6, -6 ≤ g2 ≤ 6}, {g1, g2}][[1]] // N;
  maxValue = Maximize[{comp[g1, g2] /. {β → 0.9}, -6 ≤ g1 ≤ 6, -6 ≤ g2 ≤ 6}, {g1, g2}][[1]] // N;
  scale[x_] := { x/maxValue * 0.5 + 0.5   x ≥ 0
                 (x-minValue)/(0-minValue) * 0.5   x < 0 };

  fontTicks = Directive[FontFamily → "Libertinus Serif", FontSize → 24];
  fontLabels = Directive[FontFamily → "Libertinus Serif", FontSize → 28];

  DensityPlot[comp[g1, g2] /. {β → beta}, {g1, -6, 6}, {g2, -6, 6},
   PlotTheme → "myTheme",
   LabelStyle → fontLabels,
   FrameTicksStyle → fontTicks,
   ColorFunctionScaling → False,
   ColorFunction → Function[{x}, ColorData["TemperatureMap"][scale[x]]],
   Exclusions → None,
   PlotPoints → 500,
   FrameLabel → {"g1", "g2"},
   PlotLegends → Placed[
     BarLegend[Automatic,
      LegendMarkerSize → 525,
      LegendMargins → {{50, 0}, {0, 0}},
      LegendLabel → Placed[
        Style[Row[{Subscript[it["C"], it["a"]], "(g1,g2)"}], fontLabels],
        Above
       ],
      LabelStyle → fontTicks
     ]
     , Above]
   ]
  ]
```

In[95]:=
```
(*Do[
 Export[
  FileNameJoin[
   {NotebookDirectory[],"Optimizers_AdaptiveDensity"<>ToString[Round[β*10]]<>".pdf"}],
  rasterizeBackground[plotComp[β]]
 ]
 ,{β,{0,0.6,0.9}}]*)
```