HW07 Spline Interpolation

Problem 1 (5 points)

(Analytical) Using four evenly spaced intervals, find the piecewise linear interpolant to to Runge's function,

$$f(x) = \frac{1}{1+x^2}$$

on the interval $[-5, 5]$.

Solution

Subintervals: $I_1 = [-5, -2.5], I_2 = [-2.5, 0], I_3 = [0, 2.5], I_4 = [2.5, 5]$

$$s_1 = \begin{cases} -\frac{2x+5}{130} + \frac{8x+40}{145} & x \in I_1 \\ 0 & x \notin I_1 \end{cases}$$

$$s_2 = \begin{cases} -\frac{8x}{145} + \frac{2x+5}{5} & x \in I_2 \\ 0 & x \notin I_2 \end{cases}$$

$$s_3 = \begin{cases} -\frac{2x-5}{5} + \frac{8x}{145} & x \in I_3 \\ 0 & x \notin I_3 \end{cases}$$

$$s_4 = \begin{cases} -\frac{8x-40}{145} + \frac{2x-5}{130} & x \in I_4 \\ 0 & x \notin I_4 \end{cases}$$

$s(x) = s_1(x) + s_2(x) + s_3(x) + s_4(x)$

Problem 2

(Julia) Use the `Interpolations.jl` package to find three piecewise cubic spline interpolants to Runge's function,

$$f(x) = \frac{1}{1+x^2}$$

on the interval $[-5, 5]$. Use 20, 100, and 500 evenly spaced intervals for your interpolants.

1. (3 points) Plot your results (y20, y100, y500) against the true solution using 1000 intervals

2. (2 points) Approximate the 2-norm error using these vectors and `norm` from the `LinearAlgebra` package

Solution

In [1]:
```
using Interpolations
xs20 = -5:0.5:5
xs100 = -5:0.1:5
xs500 = -5:0.02:5
```

Out[1]:
```
-5.0:0.02:5.0
```

In [2]:
```
function runges(x)
    return 1.0 / (1.0 + x^2)
end
```

Out[2]:
```
runges (generic function with 1 method)
```

In [3]:
```
ys20 = runges.(xs20)
ys100 = runges.(xs100)
ys500 = runges.(xs500)
```

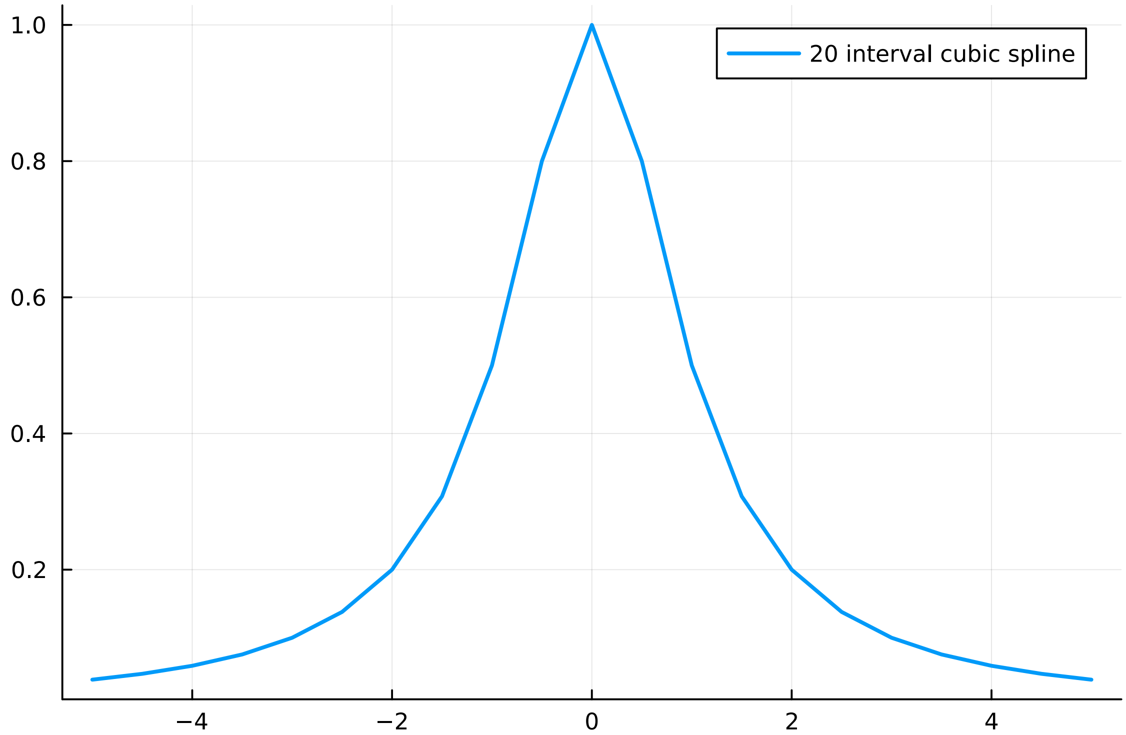
Out[3]:
```
501-element Vector{Float64}:
 0.038461538461538464
 0.03875908900637199
 0.039060058746328355
 0.0393644995197531
 0.03967246413609243
 0.03998400639744101
 0.04029918112063963
 0.04061804415993761
 0.04094065243023713
 0.04126706393093544
 0.0415973377703827
 0.04193153419097298
 0.042269714594887056
 ⋮
 0.04193153419097298
 0.0415973377703827
 0.04126706393093544
 0.04094065243023713
 0.04061804415993761
 0.04029918112063963
 0.03998400639744101
 0.03967246413609243
 0.0393644995197531
 0.039060058746328355
 0.03875908900637199
 0.038461538461538464
```

In [4]:
```
itp20 = CubicSplineInterpolation(xs20, ys20)
itp100 = CubicSplineInterpolation(xs100, ys100)
itp500 = CubicSplineInterpolation(xs500, ys500)
```

Out[4]:
```
501-element extrapolate(scale(interpolate(OffsetArray(::Vector{Float64}, 0:502), B
Spline(Cubic(Line(OnGrid())))), (-5.0:0.02:5.0,)), Throw()) with element type Floa
t64:
 0.038461538461538464
 0.03875908900637198
 0.039060058746328355
 0.03936449951975309
 0.03967246413609243
 0.039984006397441006
 0.040299181120639634
 0.040618044159937604
 0.04094065243023714
 0.04126706393093543
 0.0415973377703827
 0.04193153419097298
 0.04226971459488706
 ⋮
 0.04193153419097296
 0.0415973377703827
 0.04126706393093544
 0.04094065243023713
 0.040618044159937625
 0.04029918112063964
 0.03998400639744101
 0.03967246413609243
 0.03936449951975308
 0.03906005874632834
 0.03875908900637199
 0.03846153846153847
```

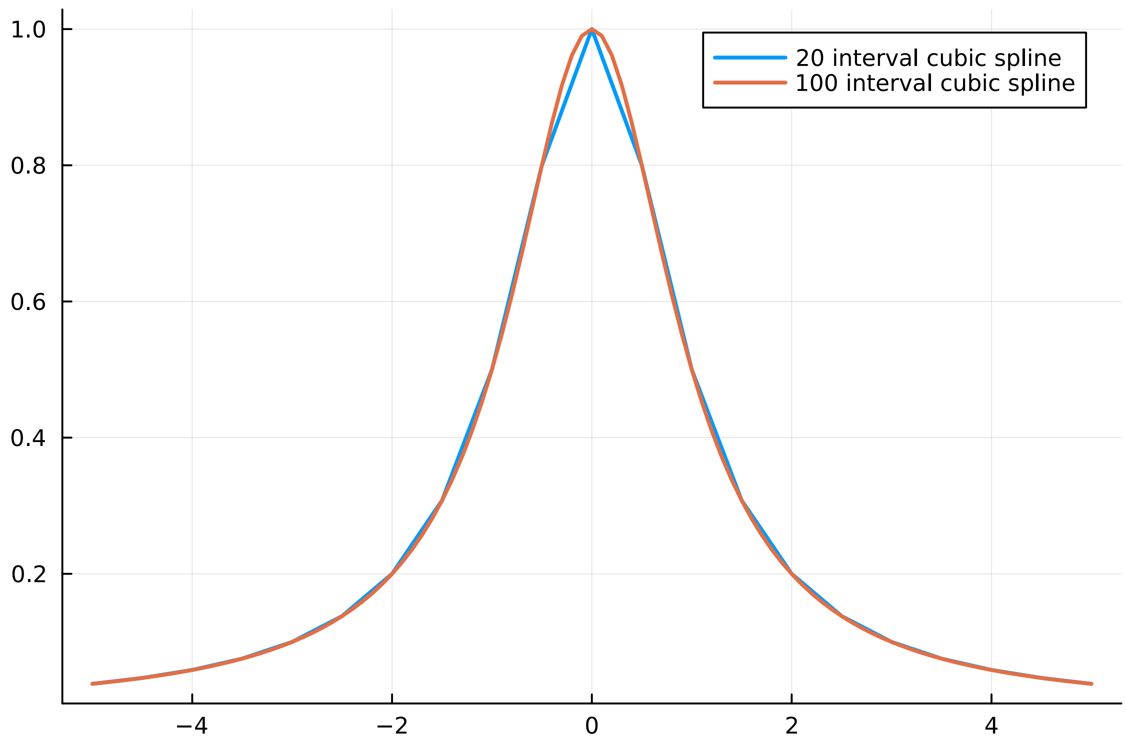
In [5]:
```
using Plots
xs1000 = -5:0.01:5
plot(xs20, itp20(xs20), lw=2, label="20 interval cubic spline")
```

Out[5]:



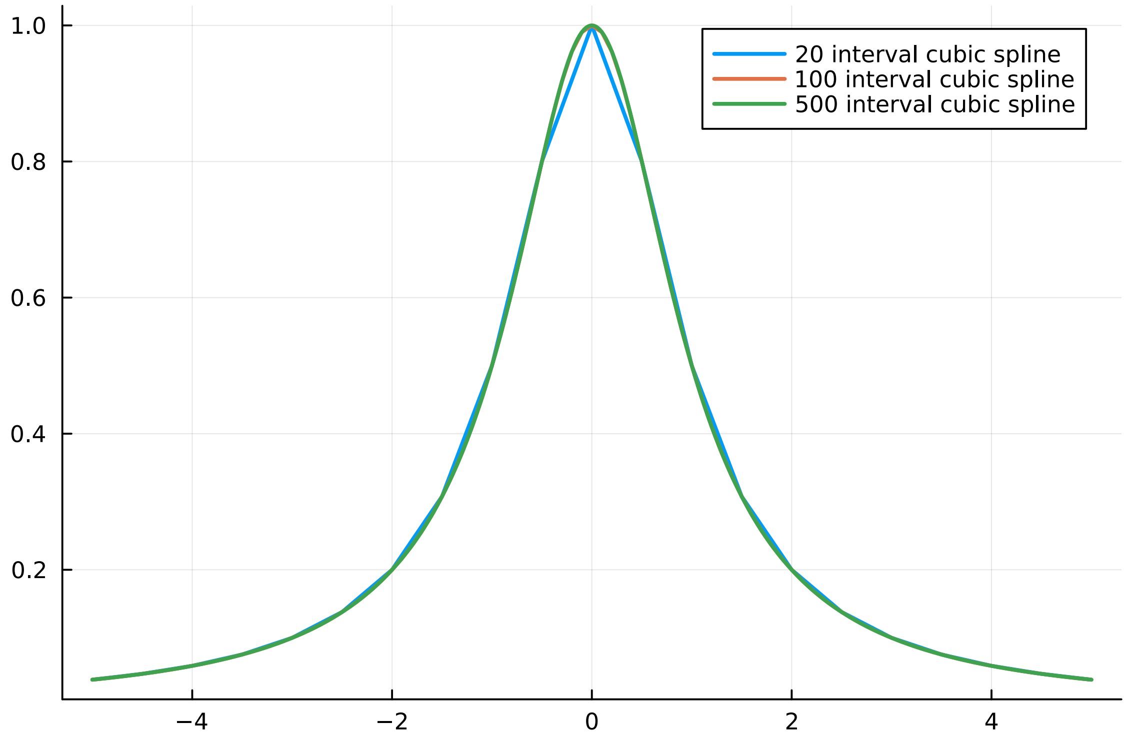
In [6]:
```
plot!(xs100, itp100(xs100), lw=2, label="100 interval cubic spline")
```

Out[6]:



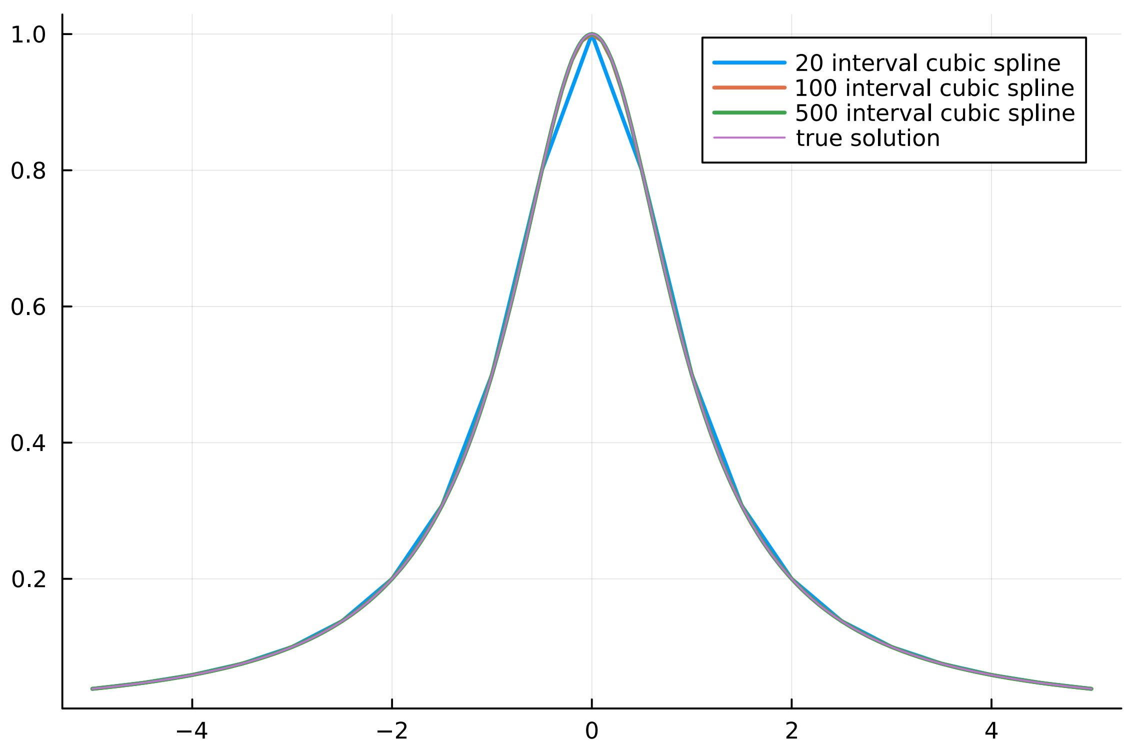
In [7]:
```
plot!(xs500, itp500(xs500), lw=2, label="500 interval cubic spline")
```

Out[7]:



In [8]:
```
plot!(xs1000, runges, lw=1, label="true solution")
```

Out[8]:



In [9]:
```
using LinearAlgebra
```

In [10]:
```
norm(runges.(xs20) - itp20(xs20))
```

Out[10]:
```
1.6478959984131287e-16
```

In [11]:
```
norm(runges.(xs100) - itp100(xs100))
```

Out[11]: 1.2181662539039428e-15

In [12]:
```
norm(runges.(xs500) - itp500(xs500))
```

Out[12]: 1.8998762468966097e-15