KidIQ: kidiq.csv

Widen the notebook.

```
html"""
<style>
    main {
        margin: 0 auto;
        max-width: 2000px;
        padding-left: max(160px, 10%);
        padding-right: max(160px, 10%);
    }
</style>
"""

```

```
using Pkg ✓ , DrWatson ✓
```

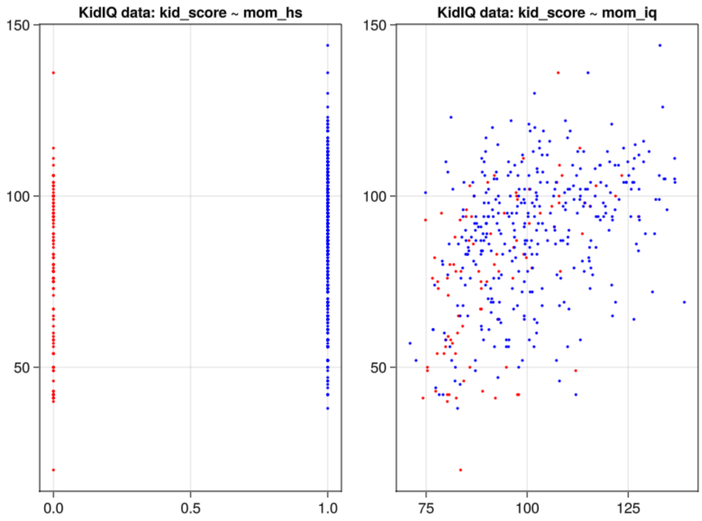
```
begin
    # Specific to this notebook
    using GLM ✓

    # Specific to ROSTuringPluto
    using Optim ✓
    using Logging ✓
    using Turing ✓

    # Graphics related
    using GLMakie ✓

    # Common data files and functions
    using RegressionAndOtherStories ✓
    import RegressionAndOtherStories: link

    Logging.disable_logging(Logging.Warn)
end;
```

```
Replacing docs for `RegressionAndOtherStories.tr
DataFrame, AbstractString}` in module `Regressio
```

```
kidiq =
```

	kid_score	mom_hs	mom_iq	mom_work
1	65	1	121.118	4
2	98	1	89.3619	4
3	85	1	115.443	4
4	83	1	99.4496	3
5	115	1	92.7457	4
6	98	0	107.902	1
7	69	1	138.893	4
8	106	1	125.145	3
9	102	1	81.6195	1
10	95	1	95.0731	1
⋮ more				
434	70	1	91.2533	2

```julia
kidiq = CSV.read(ros_datadir("KidIQ", "kidiq.csv"), DataFrame)
```



```
let
    f = Figure()
    ax = Axis(f[1, 1]; title="KidIQ data:
    kid_score ~ mom_hs")
    scatter!(kidiq[kidiq.mom_hs .== 0,
    :mom_hs], kidiq[kidiq.mom_hs .== 0,
    :kid_score]; color=:red, markersize = 3)
    scatter!(kidiq[kidiq.mom_hs .== 1,
    :mom_hs], kidiq[kidiq.mom_hs .== 1,
    :kid_score]; color=:blue, markersize =
    3)
    ax = Axis(f[1, 2]; title="KidIQ data:
    kid_score ~ mom_iq")
    scatter!(kidiq[kidiq.mom_hs .== 0,
    :mom_iq], kidiq[kidiq.mom_hs .== 0,
    :kid_score]; color=:red, markersize = 3)
    scatter!(kidiq[kidiq.mom_hs .== 1,
    :mom_iq], kidiq[kidiq.mom_hs .== 1,
    :kid_score]; color=:blue, markersize =
    3)
    current_figure()
end
```

```
stan10_1 = "
data {
    int N;
    vector[N] mom_hs;
    vector[N] kid_score;
}
parameters {
    real a;
    real b;
    real sigma;
}
model {
    vector[N] mu;
    a ~ normal(100, 10);
    b ~ normal(5, 10);
    mu = a + b * mom_hs;
    kid_score ~ normal(mu, sigma);
}
";
```

```
ppl10_1 (generic function with 2 methods)
```

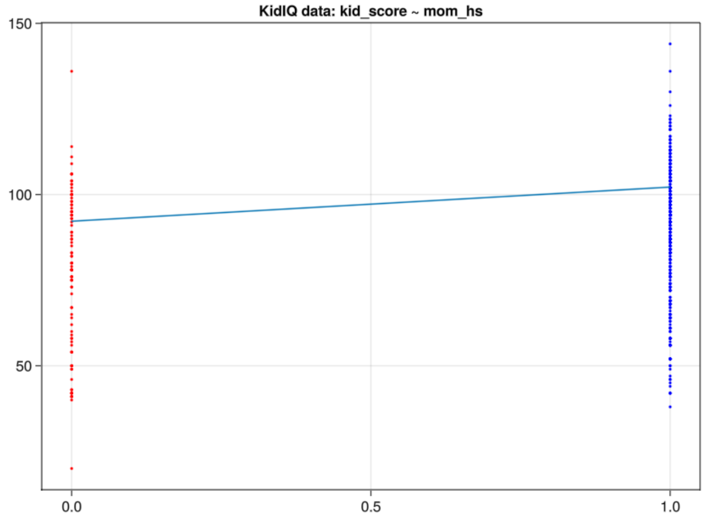
```
@model function ppl10_1(x, y)
    a ~ Normal(100, 10)
    b ~ Normal(5, 10)
    σ ~ Exponential(1)
    μ = a .+ b .* x
    for i in eachindex(y)
        y[i] ~ Normal(μ[i], σ)
    end
end
```

	parameters	mean	std	naive_se
1	:a	92.188	1.45294	0.0229729
2	:b	9.98464	1.64319	0.0259811
3	:σ	14.2282	0.486428	0.0076911

```
begin
    m10_1t = ppl10_1(kidiq.mom_hs, kidiq.mom_iq)
    chns10_1t = sample(m10_1t, NUTS(), MCMCThreads(), 1000, 4)
    describe(chns10_1t)
end
```

	parameters	median	mad_sd	mean	st
1	"a"	92.209	1.427	92.188	1.45
2	"b"	9.975	1.628	9.985	1.64
3	"σ"	14.225	0.492	14.228	0.48

```
begin
    post10_1t = DataFrame(chns10_1t)[:, 3:5]
    ms10_1t = model_summary(post10_1t, [:a,
    :b, :σ])
end
```



```
let
    f = Figure()
    ax = Axis(f[1, 1]; title="KidIQ data:
    kid_score ~ mom_hs")
    scatter!(kidiq[kidiq.mom_hs .== 0,
    :mom_hs], kidiq[kidiq.mom_hs .== 0,
    :kid_score]; color=:red, markersize = 3)
    scatter!(kidiq[kidiq.mom_hs .== 1,
    :mom_hs], kidiq[kidiq.mom_hs .== 1,
    :kid_score]; color=:blue, markersize =
    3)
    lines!([0.0, 1.0], [ms10_1t[:a,
    :median], ms10_1t[:a, :median] +
    ms10_1t[:b, :median]])
    current_figure()
end
```

```
ppl10_2 (generic function with 2 methods)
```

```julia
@model function ppl10_2(x, y)
    a ~ Normal(25, 3)
    b ~ Normal(1, 2)
    σ ~ Exponential(1)
    μ = a .+ b .* x
    for i in eachindex(y)
        y[i] ~ Normal(μ[i], σ)
    end
end
```

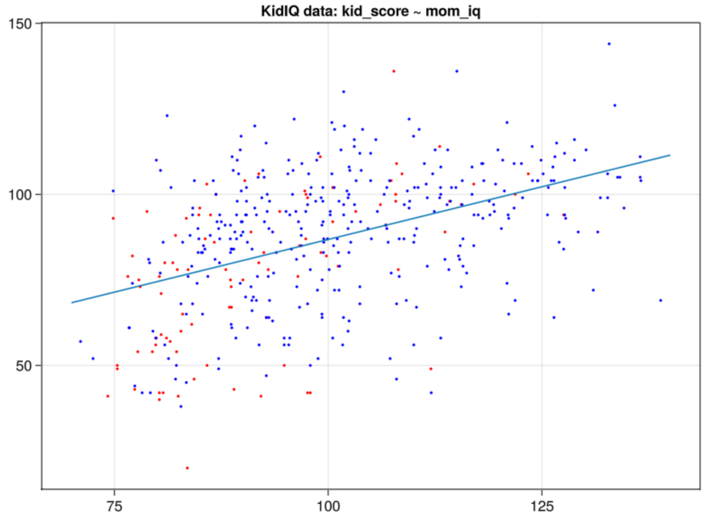
▶ [

	parameters	mean	std	naive_
1	:a	25.2691	2.76227	0.043675
2	:b	0.615021	0.0283445	0.000448
3	:σ	17.9314	0.587894	0.009295

```julia
begin
    m10_2t = ppl10_2(kidiq.mom_iq,
    kidiq.kid_score)
    chns10_2t = sample(m10_2t, NUTS(),
    MCMCThreads(), 1000, 4)
    describe(chns10_2t)
end
```

	parameters	median	mad_sd	mean	st
1	"a"	25.216	2.723	25.269	2.76
2	"b"	0.616	0.029	0.615	0.02

```julia
begin
    post10_2t = DataFrame(chns10_2t)[:, 3:5]
    ms10_2t = model_summary(post10_2t, [:a,
    :b, :sigma])
end
```



```julia
let
    f = Figure()
    ax = Axis(f[1, 1]; title="KidIQ data: kid_score ~ mom_iq")
    scatter!(kidiq[kidiq.mom_hs .== 0, :mom_iq], kidiq[kidiq.mom_hs .== 0, :kid_score]; color=:red, markersize = 3)
    scatter!(kidiq[kidiq.mom_hs .== 1, :mom_iq], kidiq[kidiq.mom_hs .== 1, :kid_score]; color=:blue, markersize = 3)
    x = LinRange(70.0, 140.0, 100)
    lines!(x, ms10_2t[:a, :median] .+ ms10_2t[:b, :median] .* x)
    current_figure()
end
```

```
ppl10_3 (generic function with 2 methods)
```

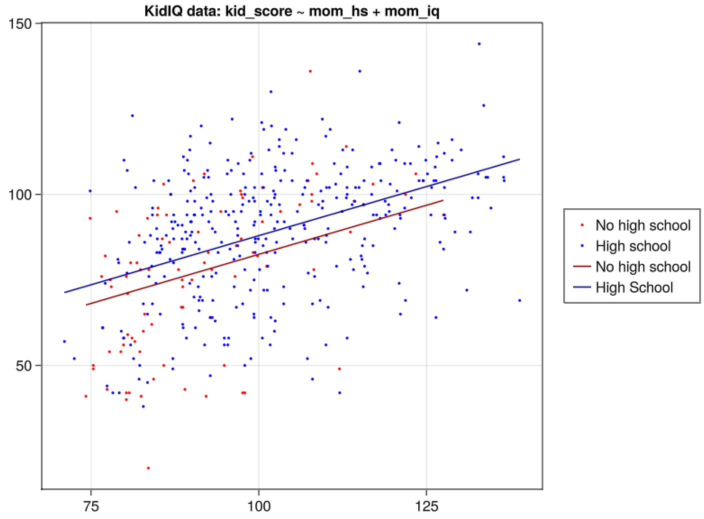
```julia
@model function ppl10_3(x, y, z)
    a ~ Normal(25, 2)
    b ~ Normal(5, 2)
    c ~ Normal(1, 2)
    σ ~ Exponential(1)
    μ = a .+ b .* x .+ c .* y
    for i in eachindex(y)
        z[i] ~ Normal(μ[i], σ)
    end
end
```

	parameters	mean	std	naive_s
1	:a	25.0745	1.92529	0.0304414
2	:b	5.43611	1.43634	0.0227106
3	:c	0.57433	0.0233893	0.0003698
4	:σ	17.7739	0.586972	0.0092808

```
begin
    m10_3t = ppl10_3(kidiq.mom_hs,
    kidiq.mom_iq, kidiq.kid_score)
    chns10_3t = sample(m10_3t, NUTS(),
    MCMCThreads(), 1000, 4)
    describe(chns10_3t)
end
```

	parameters	median	mad_sd	mean	st
1	"a"	25.069	1.899	25.075	1.92
2	"b"	5.459	1.472	5.436	1.43
3	"c"	0.574	0.023	0.574	0.02
4	"σ"	17.747	0.603	17.774	0.58

```
begin
    post10_3t = DataFrame(chns10_3t)[:, 3:6]
    ms10_3t = model_summary(post10_3t, [:a,
    :b, :c, :σ])
end
```



```
let
    momnohs(x) = x == 0
    nohs = findall(momnohs, kidiq.mom_hs)

    momhs(x) = x == 1
    hs = findall(momhs, kidiq.mom_hs)

    f = Figure()
    ax = Axis(f[1, 1]; title="KidIQ data:
    kid_score ~ mom_hs + mom_iq")
    sca1 = scatter!(kidiq[kidiq.mom_hs .==
    0, :mom_iq], kidiq[kidiq.mom_hs .== 0,
    :kid_score]; color=:red, markersize = 3)
    sca2 = scatter!(kidiq[kidiq.mom_hs .==
    1, :mom_iq], kidiq[kidiq.mom_hs .== 1,
    :kid_score]; color=:blue, markersize =
    3)
    x = sort(kidiq.mom_iq[nohs])
    lin1 =lines!(x, ms10_3t[:a, :median] .+
    ms10_3t[:b, :median] .*
    kidiq.mom_hs[nohs] .+ ms10_3t[:c,
    :median] .* x;
        color=:darkred)
    x = sort(kidiq.mom_iq[hs])
    lin2 =lines!(x, ms10_3t[:a, :median] .+
    ms10_3t[:b, :median] .*
    kidiq.mom_hs[hs] .+ ms10_3t[:c,
    :median] .* x;
        color=:darkblue)
    Legend(f[1, 2],
        [sca1, sca2, lin1, lin2],
        ["No high school", "High school",
"No high school", "High School"])
    current_figure()
end
```