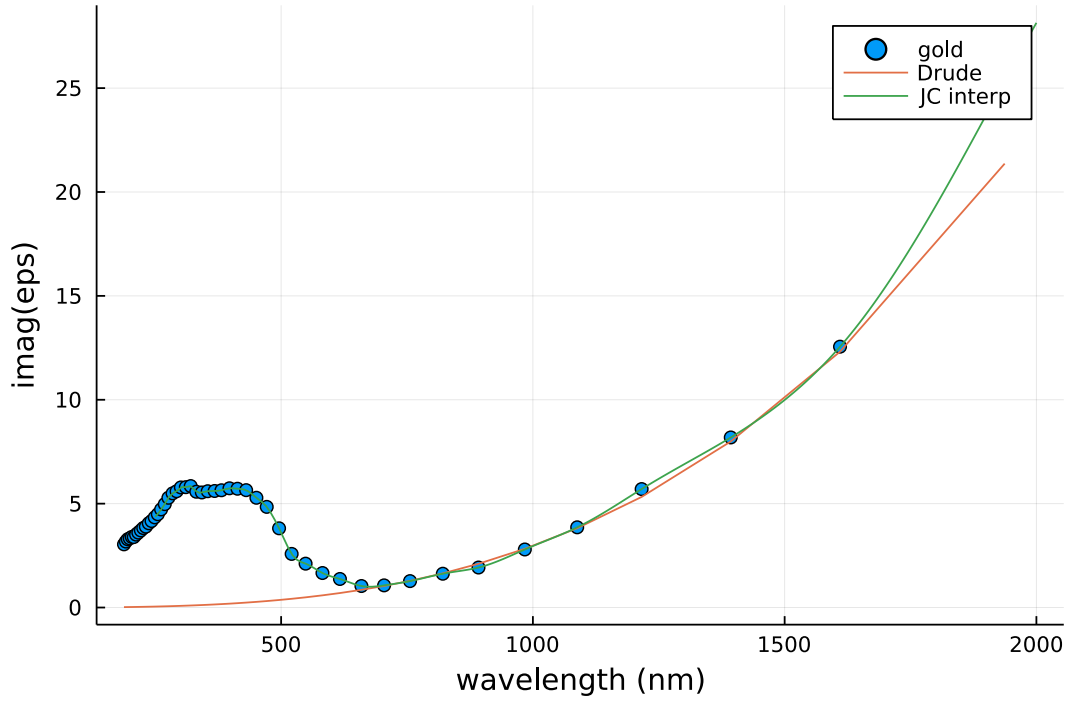
```
[0.227056+3.04128im, 0.295191+3.17592im, 0.292524+3.28568im, 0.203899+3.32766im, 0.138171
```

```
# get the dielectric function of gold by Johnson & Christy from the web
begin
    url = "https://refractiveindex.info/data_csv.php?datafile=data/main/Au/Johnson.yml"
    data = Downloads.download(url)
    data_n = CSV.File(data; footerskip=50)   # first come the n-values
    data_k = CSV.File(data; skipto=53)      # then below the k-values for same λ

    λ = data_n.wl * 1u"μm" ;
    ior_gold = data_n.n + 1im * data_k.n
    eps_gold = ior_gold .^2

end
```

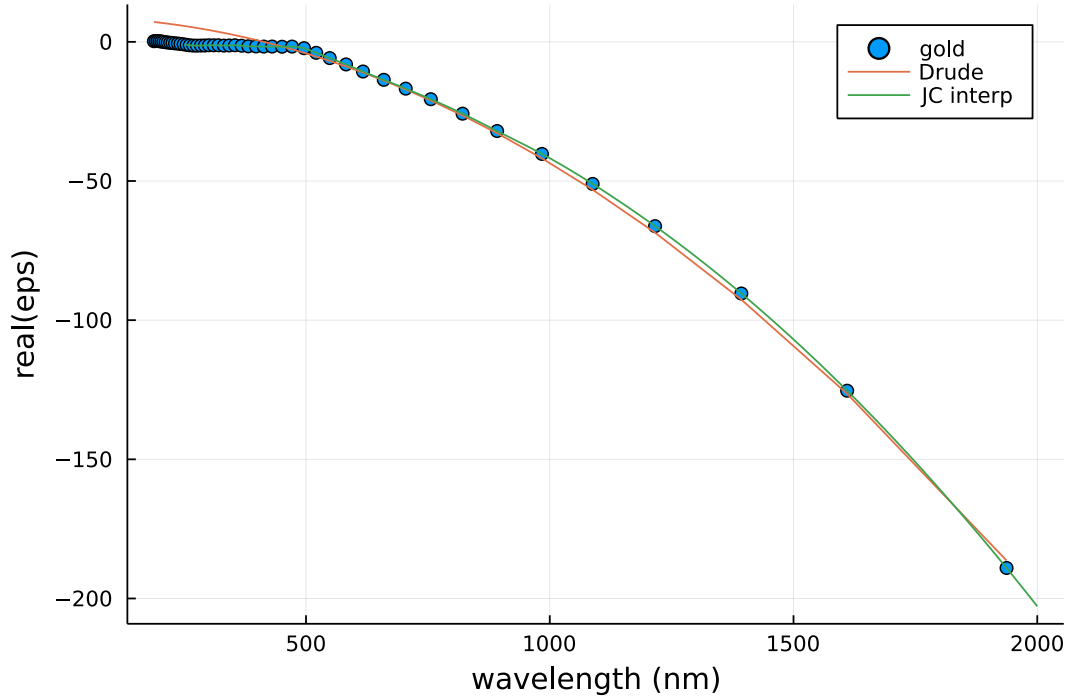
```
eps_JC (generic function with 1 method)
```

```
# interpolate experimental data
begin
    # NB: order of arguments is (Y,X)
    ip_n =CubicSpline(data_n.n , data_n.wl ) # create interpolation function
    ip_k =CubicSpline(data_k.n , data_k.wl ) # create interpolation function

    function eps_JC(wl)
        wl_um = ustrip(u"μm",wl);
        return ( ip_n(wl_um) + 1im * ip_k(wl_um) )^2
    end
end
```

```
wlrange = (250.0:8.793969849246231:2000.0) nm
```




```
begin
    plot(ustrip.(u"nm",λ), imag(eps_gold), label="gold", xlabel = "wavelength (nm)", ylabel="imag(eps)", seriestype = :scatter)
    plot!(ustrip.(u"nm",λ), @. imag(ϵ_in( 2 * pi * c_0 / λ)); label="Drude")

    plot!(ustrip.(u"nm",wlrange), @. imag(eps_JC(wlrange)); label="JC interp")
end
```




```
begin
    plot(ustrip.(u"nm",λ), real(eps_gold), label="gold", xlabel = "wavelength (nm)", ylabel="real(eps)", seriestype = :scatter)
    plot!(ustrip.(u"nm",λ), @. real(ϵ_in( 2 * pi * c_0 / λ)); label="Drude")
    plot!(ustrip.(u"nm",wlrange), @. real(eps_JC(wlrange)); label="JC interp")

end
```

```
9
```

```
# some constants
begin
    ω_p = 9u"eV" / ħ;  # frequency units are eV
    γ = 0.07u"eV" / ħ;
    ϵ_∞ = 9;
end
```

```
ϵ_in (generic function with 1 method)
```

```
# Drude model for the metal, all constants are defined above
function ϵ_in(ω)
    return ϵ_∞ - ω_p^2 / (ω * (ω + 1im * γ))
end
```