2.28 c)

Given parameters (a, ξ, σ), we have data $Y_1, Y_2, \ldots, Y_n$ iid. with distribution

$$Y_i \overset{d}{=} \frac{\exp\{a(\xi + \sigma Z)\} - a}{a},$$

where Z is standard normal.

To derive the moment estimators of (a, ξ, σ), we will repeatedly make use of the MGF of a standard normal variable, which is given by

$$M(t) = E\left[e^{tZ}\right] = e^{\frac{1}{2}t^2}$$

To get started with the method of moments we need expressions for the theoretical moments of the distribution of Y_i. We start by computing the first moment:

$$\mu_1 := E[Y_i] = E\left[\frac{\exp\{a(\xi + \sigma Z)\} - 1}{a}\right] = \frac{1}{a}e^{a\xi}E\left[e^{a\sigma Z}\right] - \frac{1}{a} = \frac{1}{a}e^{a\xi}M(a\sigma) - \frac{1}{a},$$

I will continue to express the moments in terms of the MGF of the standard normal, as it makes the ensuing algebra a little bit easier to follow. To derive the second centralized moment we proceed in a similar fashion:

$$\begin{aligned}
\mu_2 :&= E(Y_i - E[Y_i])^2 \\
&= E\left(\frac{1}{a}\exp\{a(\xi + \sigma Z)\} - \frac{1}{a} - \left[\frac{1}{a}e^{a\xi}M(a\sigma) - \frac{1}{a}\right]\right)^2 \\
&= E\left(\frac{1}{a}e^{a\xi}[e^{a\sigma Z} - M(a\sigma)]\right)^2 \\
&= \frac{1}{a^2}e^{2a\xi}\left\{E(e^{2a\sigma Z}) - 2E(e^{a\sigma Z})M(a\sigma) + [M(a\sigma)]^2\right\} \\
&= \frac{1}{a^2}e^{2a\xi}\left\{E(e^{2a\sigma Z}) - 2[M(a\sigma)]^2 + [M(a\sigma)]^2\right\} \\
&= \frac{1}{a^2}e^{2a\xi}\left\{M(2a\sigma) - [M(a\sigma)]^2\right\}
\end{aligned}$$

Finally, we compute the third centralized moment:

$$\begin{aligned}
\mu_3 :&= E(Y_i - E[Y_i])^3 \\
&= E\left(\frac{1}{a}\exp\{a(\xi + \sigma Z)\} - \frac{1}{a} - \left[\frac{1}{a}e^{a\xi}M(a\sigma) - \frac{1}{a}\right]\right)^3 \\
&= E\left(\frac{1}{a}e^{a\xi}\left[e^{a\sigma Z} - M(a\sigma)\right]\right)^3 \\
&= \frac{1}{a^3}e^{3a\xi}\left\{E(e^{3a\sigma Z}) - 3E(e^{2a\sigma Z})M(a\sigma) + 3E\left[e^{a\sigma Z}\right][M(a\sigma)]^2 - [M(a\sigma)]^3\right\} \\
&= \frac{1}{a^3}e^{3a\xi}\left\{M(3a\sigma) - 3M(2a\sigma)M(a\sigma) + 3[M(a\sigma)]^3 - [M(a\sigma)]^3\right\} \\
&= \frac{1}{a^3}e^{3a\xi}\left\{M(3a\sigma) - 3M(2a\sigma)M(a\sigma) + 2[M(a\sigma)]^3\right\}
\end{aligned}$$

Having derived the three first empirical moments, we are now ready to compute the moment estimators numerically. Recall that the moment estimators are given as the solution in terms of (a, ξ, σ) to the equations

$$\begin{aligned}
\bar{Y} &= \mu_1(a, \xi, \sigma) \\
\frac{1}{n}\sum_{i=1}^{n}(Y_i - \bar{Y})^2 &= \mu_2(a, \xi, \sigma) \\
\frac{1}{n}\sum_{i=1}^{n}(Y_i - \bar{Y})^3 &= \mu_3(a, \xi, \sigma)
\end{aligned}$$

Solving these equations analytically is more of an exercise in tedium than anything else. Therefore, a numerical solution will have to suffice in this example. To solve these equations numerically, we employ the NLH trick of viewing the solution of the equation as an optimization problem.

In [ ]:
```
using Distributions, Optim
```

```julia
# Parameters
a = 0.33
ξ = 0.55
σ = 0.77

# Number of simulated data points
n = 4011

# Simulate data from the given distribution
y = zeros(n)
for i = 1:n
    y[i] = (exp(a*(ξ + σ * rand(Normal(0, 1)) )) - 1)/a
end

# Empirical moments
M1 = mean(y)
M2 = mean((y .- M1).^2)
M3 = mean((y .- M1).^3)
moments = [M1, M2, M3]

# MGF of standard normal
function M(t)
    return exp(0.5*t^2)
end

# Check that empirical moments are close to theoretical ones
println(moments)
println(
    [1.0/a * exp(a*ξ) * M(a*σ) - 1.0/a,
     1.0/a^2 * exp(2.0*a*ξ) * (M(2.0*a*σ) - (M(a*σ))^2),
     1.0/a^3 * exp(3.0*a*ξ) * (M(3.0*a*σ)- 3.0*M(2.0*a*σ)*M(a*σ) + 2.0 * (M(a*σ))^3)]
)


# Function to be optimized. A minimum of this function corresponds to a solution of the moment equations
function f(par, moments)
    # Unpack parameters, empirical moments
    a = par[1]
    ξ = par[2]
    σ = par[3]

    M1 = moments[1]
    M2 = moments[2]
    M3 = moments[3]

    # Compute theoretical moments for given (a, ξ, σ)
    μ1 = 1.0/a * exp(a*ξ) * M(a*σ) - 1.0/a
    μ2 = 1.0/a^2 * exp(2.0*a*ξ) * (M(2.0*a*σ) - (M(a*σ))^2)
    μ3 = 1.0/a^3 * exp(3.0*a*ξ) * (M(3.0*a*σ)- 3.0*M(2.0*a*σ)*M(a*σ) + 2.0 * (M(a*σ))^3)

    # Return squared norm for optimization purposes
    return (μ1 - M1)^2 + (μ2 - M2)^2 + (μ3 - M3)^2
end

target = par -> f(par, moments)

result = optimize(target, [1.0, 1.0, 1.0], NelderMead())
println(result) # Verify that our optimization routine has converged
res = Optim.minimizer(result)
println("Estimated")
println("a = $(round(res[1], digits = 3)), ξ = $(round(res[2], digits = 3)), σ = $(round(res[3], digits = 3))")
println("True values")
println("a = $a, ξ = $ξ, σ = $σ")
```

```
[0.7269802606574826, 0.9318369005043657, 0.6958918756245466]
[0.7222860738528367, 0.939219924066778, 0.7208990055952195]
 * Status: success

 * Candidate solution
    Final objective value:     2.282019e-09

 * Found with
    Algorithm:     Nelder-Mead

 * Convergence measures
    √(Σ(yᵢ-ȳ)²)/n ≤ 1.0e-08

 * Work counters
    Seconds run:   0  (vs limit Inf)
    Iterations:    81
    f(x) calls:    149

Estimated
a = 0.323, ξ = 0.557, σ = 0.77
True values
a = 0.33, ξ = 0.55, σ = 0.77
```