Quantitative Risk Management
Fall Assignment

John Dodson

September 26, 2018

This assignment is not a regular homework. It is a group project worth half of the module grade for the fall term. Each team member will receive equal credit.

Please share your solution with me through Google Drive before the beginning of the classroom session on Wednesday, October 10. Please turn in your report directly to me. You are welcome to discuss the project with our TA, but she will not be grading it.

Your solution should include source code for programs and instructions for how to run them with the software available on Math Department computers. I recommend you use `tar` or `zip` if you will be submitting several files.

Introduction

We saw in the third assignment that the tail index for an index of equity simple returns seems to be negative, but also that the dataset seems to exhibit some clustering of loss events calling into question the assumption in the estimation technique that the sample is i.i.d.

We saw in the fourth assignment that the the α_1 parameter in GARCH(1,1) is materially positive, suggesting that there is conditional heteroskedasticity, which may contribute to clustering of extreme losses. The GARCH(1,1) fit can be used to attenuate this by converting the raw returns to standardized returns.

Problem Statement

Please produce a report or presentation explaining the problem and your approach to the solution, including intermediate and final results, and discussing potential interpretations and relevant implications.

- Fit GARCH(1,1) to simple daily total returns on the listed common equity of your assigned company using 1,000 business days up to September 26, 2018. **(20 points)**
- Fit Generalized Pareto to the lowest 2% of standardized residuals and report the standard error of the tail index. **(20 points)**

Grading Rubric

Ten out of fifty points will be based on the follow criteria:

- You follow all of the instructions. **(2 points)**
- I can reproduce your results with the code and documentation you provide. **(2 points)**
- Your write-up is clear and professional. **(6 points)**

Solution

In the case from September 26, I introduced the BHHH solver for maximum likelihood estimates, and illustrated how a fit to Generalized Pareto could be implemented. In this solution, I will continue with the BHHH solver in the context of the conditional quasi-MLE for GARCH(1,1),

$$\sigma_i^2 = \alpha_0 + \alpha_1 \varepsilon_{i-1}^2 + \beta_1 \sigma_{i-1}^2$$

for a timeseries of invariants $(X_i)_i$ where $\varepsilon_i = X_i - \mathrm{E}[X_i|\mathcal{F}_{i-1}]$ and $\sigma_i^2 = \mathrm{var}[X_i|\mathcal{F}_{i-1}]$.

The (quasi[1], conditional) negative log-likelihood for a sample $(x_i)_{i=1,\ldots,n}$ is

$$h(u) = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2}\left(\log\left(2\pi\sigma_i^2\right) + \frac{\epsilon_i^2}{\sigma_i^2}\right)$$

for parameters $u = (\alpha_0, \alpha_1, \beta_1)$, residuals $\varepsilon_i = x_i - \mu_i$, and unconditional variance $\sigma_1^2 = \alpha_0/(1-\beta_1-\alpha_1)$. We can assume that $\mu_i \equiv 0$ for daily returns.

BHHH is a quasi[2]-Newton method requiring an explicit gradient along with the objective function, from which the Hessian can be approximated from the Fisher information[3].

The partials are all of the form

$$\frac{\partial h}{\partial u_j} = \frac{1}{n}\sum_{i=1}^{n} \frac{1}{2\sigma_i^2}\left(1 - \frac{\varepsilon_i^2}{\sigma_i^2}\right)\frac{\partial \sigma_i^2}{\partial u_j}$$

and the partials of the conditional variance are themselves linear recursions.

$$\begin{aligned}
\frac{\partial \sigma_1^2}{\partial \alpha_0} &= \frac{1}{1-\beta_1-\alpha_1} \quad , \quad & \frac{\partial \sigma_i^2}{\partial \alpha_0} &= 1 + \beta_1 \frac{\partial \sigma_{i-1}^2}{\partial \alpha_0} \quad i = 2,\ldots,n \\
\frac{\partial \sigma_1^2}{\partial \alpha_1} &= \frac{1}{(1-\beta_1-\alpha_1)^2} \quad , \quad & \frac{\partial \sigma_i^2}{\partial \alpha_1} &= \varepsilon_{i-1}^2 + \beta_1 \frac{\partial \sigma_{i-1}^2}{\partial \alpha_1} \quad i = 2,\ldots,n \\
\frac{\partial \sigma_1^2}{\partial \beta_1} &= \frac{1}{(1-\beta_1-\alpha_1)^2} \quad , \quad & \frac{\partial \sigma_i^2}{\partial \beta_1} &= \sigma_{i-1}^2 + \beta_1 \frac{\partial \sigma_{i-1}^2}{\partial \beta_1} \quad i = 2,\ldots,n
\end{aligned}$$

Finally, recall from the solution to the third assignment that the Cramér-Rao lower bound on the variance of the MLE for the tail index (assuming it is unbiased) is

$$\mathrm{se}(\hat{\xi}) = \frac{\xi + 1}{\sqrt{n/50}}$$

The data consist of adjusted daily log-returns from the close of the session on October 8, 2014, through the close of the session on September 26, 2018 for six common equities listed in Table 1. Cumulative returns are plotted in the Figure 1.

Results

Results for the fits are in Table 2. With the possible exception of Comcast, the constant (α_0) and ARCH (α_1) terms are materially positive.

Recall that the three-parameter GARCH(1,1) degenerates into two different single-parameter models at the parameter boundary.

[1] in the sense of maximum entropy residuals
[2] in the sense of approximate Hessian
[3] Note that in this case the Cramér-Rao lower bound result does not apply because the sample is not i.i.d..

AMGN	Amgen Inc.
ADBE	Adobe Systems Inc.
CMCSA	Comcast Corp.
CSCO	Cisco Systems Inc.
ISRG	Intuitive Surgical Inc.
PEP	Pepsico Inc.

Table 1: Common equity securities analyzed

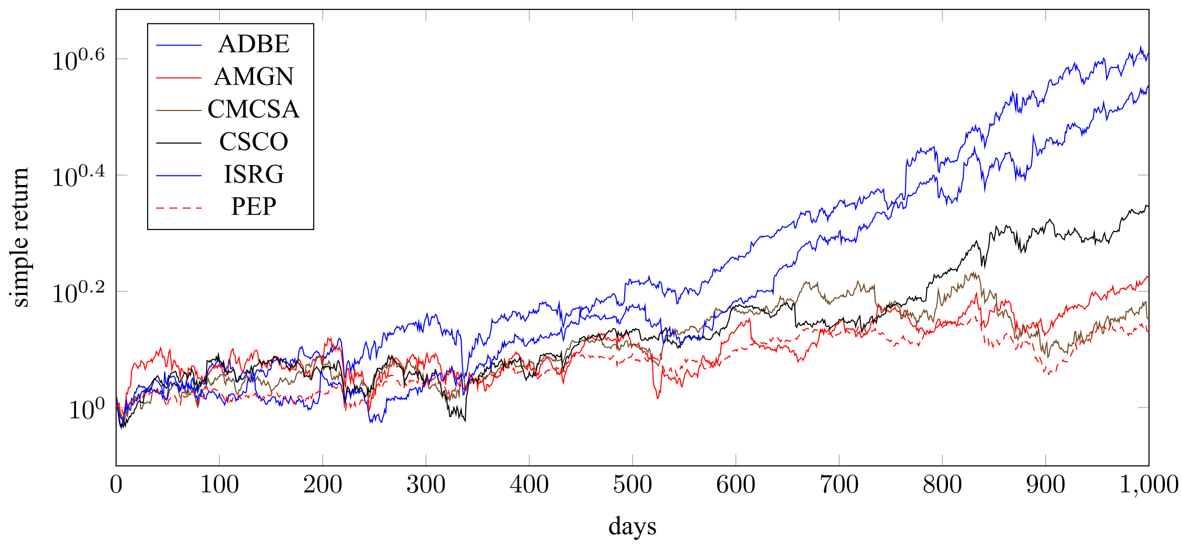

Figure 1: Cumulative simple return from October 8, 2014 through September 26, 2018.

	AMGN	ADBE	CMCSA	CSCO	ISRG	PEP
$\hat{\alpha}_0$	27.2×10-6	44.7×10-6	1.3×10-6	18.7×10-6	84.7×10-6	19.9×10-6
$\hat{\alpha}_1$	0.111	0.270	0.043	0.159	0.305	0.234
$\hat{\beta}_1$	0.773	0.580	0.952	0.748	0.319	0.523
σ^2_{n+1}	146.3×10-6	191.4×10-6	306.2×10-6	104.0×10-6	168.4×10-6	85.4×10-6
σ^2_{∞}	235.1×10-6	298.2×10-6	253.5×10-6	201.6×10-6	225.7×10-6	82.0×10-6
η	-2.273	-2.151	-2.255	-1.987	-2.060	-2.291
$\hat{\beta}$	0.498	0.638	0.944	1.173	0.874	0.559
$\hat{\xi}$	+0.341	+0.068	-0.035	+0.025	-0.029	-0.027
se($\hat{\xi}$)	0.300	0.239	0.216	0.229	0.217	0.218

Table 2: Parameter fits for 1,000 daily log-returns through September 26, 2018

worst loss	AMGN	ADBE	CMCSA	CSCO	ISRG	PEP
per year	-5.01%	-5.45%	-5.78%	-5.41%	-5.04%	-2.84%
per decade	-9.32%	-8.30%	-8.69%	-9.22%	-7.66%	-3.88%

Table 3: Extrapolated extreme daily (simple-interest) losses based on Generalized Pareto fits

- If $\alpha_0 \approx 0$, the exponentially weighted moving average (EWMA) model might be adequate; although with an undefined unconditional variance, it is not realistic for financial invariants over longer horizons.
- If $\alpha_1 \approx 0$, a fixed variance might be adequate.

In the case of Comcast, the fixed variance model might be a reasonable alternative to GARCH.

Considering the left tail, it is probably difficult to reject the hypothesis that extreme negative log-returns are in the Gumbel class ($\xi = 0$). The Gumbel is the most reasonable extreme value class for log-returns since it does not impose an arbitrary bound; and since finite expected values for prices requires all moments in the exponent.

While the Gumbel class does includes the normal, that is only an example. Considering the empirical 2% thresholds, all but Cisco Systems (and essentially Intuitive Surgical) have $\eta < \Phi^{-1}(0.02) \approx -2.05$ suggesting some skewness or excess kurtosis.

We can use the Generalized Pareto model to extrapolate to extreme but plausible idiosyncratic loss scenarios for equity holders. For example, assuming that there are on average 252.75 trading days per year, the worst day in a year corresponds to a quantile level of approximately $1/252.75 \approx 0.0040$. Using the inverse of the distribution function, we can determine the corresponding standardized residual, which we can scale with the square-root of the unconditional variance and exponentiate to get the simple-return loss. A sample of these is in Table 3.

For context, there was a systematic decline in US equity markets on October 10, 2018, just a few days after the end of this calibration period. The narrative at the time was that the market was reacting to very strong jobs report and hence renewed fear of Fed tightening; and technology stocks seems to be the most affected. Adobe Systems in particular lost 6.39% (simple interest) during the session, which exceeds the forecast worst lost per year. Our other stocks also suffered declines, but comfortably less negative than the predictions in the table.

Julia[4] implementation (`fallassign.jl`)

```
module Fallassign

using Statistics
using LinearAlgebra

"GARCH(1,1) conditional variance for θ = [α0;α1;β1]"
function garch(ϵ,θ)
	(α0,α1,β1) = θ
	σ² = fill(NaN,length(ϵ))
	if α0>0 && α1>0 && β1≥0 && α1<1-β1
		σ²[1] = α0/(1-β1-α1)
		for i = 2:length(ϵ)
			σ²[i] = α0+α1*ϵ[i-1]^2+β1*σ²[i-1]
		end
	end
	return σ²
end

"GARCH(1,1) conditional variance (α0,α1,β1) partials"
```

[4] `https://julialang.org/`

```
function garch_grad(ϵ,θ)
	(α0,α1,β1) = θ
	σ² = garch(ϵ,θ)
	grad = fill([NaN;NaN;NaN],length(ϵ))
	if α0>0 && α1>0 && β1≥0 && α1<1-β1
		grad[1] = [
			1/(1-β1-α1);
			α0/(1-β1-α1)^2;
			α0/(1-β1-α1)^2 ]
		for i = 2:length(ϵ)
			grad[i] = [
				1+β1*grad[i-1][1];
				ϵ[i-1]^2+β1*grad[i-1][2];
				σ²[i-1]+β1*grad[i-1][3] ]
		end
	end
	return grad
end

"negative quasi log-likelihood for GARCH"
function qmle_obj(ϵ,θ)
	σ² = garch(ϵ,θ)
	return (log.(2π*σ²)+ϵ.^2 ./σ²)/2
end

"negative quasi log-likelihood for GARCH (α0,α1,β1) partials"
function qmle_grad(ϵ,θ)
	σ² = garch(ϵ,θ)
	return (1 .-ϵ.^2 ./σ²)./(2*σ²).*garch_grad(ϵ,θ)
end

"indicator for domain for GP parameters"
function valid(z,θ)
	(β,ξ) = θ
	if β≤0 || ξ<-1 || maximum(z)>0
		return false
	end
	if ξ<0 && minimum(z)≤β/ξ
		return false
	end
	return true
end

"negative log-likelihood for GP for θ = [β;ξ]"
function mle_obj(z,θ)
	if !valid(z,θ)
		return fill(NaN,length(z))
	end
	(β,ξ) = θ
	if abs(ξ)<eps()
		return log(β).-z/β
```

```
    end
    return log(β).+(1+1/ξ)log.(1 .-ξ*z/β)
end

"negative log-liklihood for GP (β,ξ) partials"
function mle_grad(z,θ)
    if !valid(z,θ)
        return fill([NaN;NaN],length(z))
    end
    (β,ξ) = θ
    if abs(ξ)<eps()
        return [[
                   (1+z/β)/β;
                -z/β*(1+z/2β) ] for z in z]
    end
    return [[
        (1-(1+1/ξ)*(1-1/(1-ξ*z/β)))/β;
        (1+1/ξ)*(1-1/(1-ξ*z/β))/ξ-
                log(1-ξ*z/β)/ξ^2 ] for z in z]
end

"Newton's method minimizer"
function newtMin(h_obj::Function,h_grad::Function
                ,h_hess::Function,u0::Vector
                    ;maxiter=100,tol=1.e-8,δ=1.e-4)
    u1 = u0
    h1 = h_obj(u1)
    if isnan(h1)
        throw(DomainError(u0,"invalid initial value"))
    end
    while maxiter>0
        u0 = u1
        h0 = h1
        k = 0
        while maxiter>0 && (k==0 || isnan(h1)
            || h1-h0>δ*dot(u1-u0,h_grad(u0)))
                u1 = u0-2.0^k*h_hess(u0)\h_grad(u0)
                h1 = h_obj(u1)
                k -= 1
                maxiter -= 1
        end
        if abs(h1-h0)<tol
            return u1
        end
    end
    return u0
end

"BHHH solver for maximum likelihood estimates"
function bhhh(x::Vector,obj::Function,grad::Function,θ₀::Vector)
    h_obj = θ->mean(obj(x,θ))
```

```
        h_grad = θ->mean(grad(x,θ))
        h_hess = θ->cov(grad(x,θ))
        return newtMin(h_obj,h_grad,h_hess,θ₀)
end

export garch, qmle_obj, qmle_grad, mle_obj, mle_grad, bhhh

end # Fallassign

#        ------------------
#        ----- SCRIPT -----
#        ------------------

using .Fallassign
using CSV
using Statistics

"dataset"
df = CSV.read("fallassign.csv")
dates = df[:Date] # assume oldest first
tickers = [symb for symb in names(df) if symb != :Date]

"GARCH parameters"
parms_garch = Dict{Symbol,Vector{Float64}}()
"GP parameters"
parms_gp = Dict{Symbol,Vector{Float64}}()

for ticker in tickers
# prepare invariants
        x = diff(log.(df[ticker])) # daily log-returns
        μ = zeros(length(x)) # assume zero conditional means
        ϵ = x-μ # residuals
# fit GARCH
        (α1₀,β1₀) = (.2,.7) # guess initial GARCH coeffs
        α0₀ = var(ϵ)*(1-β1₀-α1₀) # match moments
        θ_garch = bhhh(ϵ,qmle_obj,qmle_grad,[α0₀;α1₀;β1₀])
        σ² = garch([ϵ;NaN],θ_garch)
        σ²₁ = pop!(σ²) # forecast
# lower tail of standardized returns
        z = collect(partialsort!(ϵ./sqrt.(σ²)
                ,1:div(length(ϵ),50))) # fast sort
        η = z[end] # threshold, empirical 2% quantile
# fit GP
        ξ₀ = (1-mean(z.-η)^2/var(z))/2 # match moments
        β₀ = -mean(z.-η)*(1-ξ₀) # match moments
        θ_gp = bhhh(z,mle_obj,mle_grad,[β₀;ξ₀])
# memo results
        parms_garch[ticker] = [θ_garch;σ²₁]
        parms_gp[ticker] = [η;θ_gp]
end
```