Question 1

(a)

Using C_t, C_{t-1}, r_t, and r_{t-1} as instruments, the GMM moment conditions are

$$\boldsymbol{h(\theta_0, Y_t)} = \begin{bmatrix} \left(\beta\dfrac{u'(C_{t+1})}{u'(C_t)}(1+r_{r+1})-1\right)C_t \\ \left(\beta\dfrac{u'(C_{t+1})}{u'(C_t)}(1+r_{r+1})-1\right)C_{t-1} \\ \left(\beta\dfrac{u'(C_{t+1})}{u'(C_t)}(1+r_{r+1})-1\right)r_t \\ \left(\beta\dfrac{u'(C_{t+1})}{u'(C_t)}(1+r_{r+1})-1\right)r_{t-1} \end{bmatrix} = \begin{bmatrix} \left(\beta\left(\dfrac{C_{t+1}}{C_t}\right)^{-\gamma}(1+r_{r+1})-1\right)C_t \\ \left(\beta\left(\dfrac{C_{t+1}}{C_t}\right)^{-\gamma}(1+r_{r+1})-1\right)C_{t-1} \\ \left(\beta\left(\dfrac{C_{t+1}}{C_t}\right)^{-\gamma}(1+r_{r+1})-1\right)r_t \\ \left(\beta\left(\dfrac{C_{t+1}}{C_t}\right)^{-\gamma}(1+r_{r+1})-1\right)r_{t-1} \end{bmatrix}$$

$$\implies \mathrm{E}[\boldsymbol{h(\theta_0, Y_t)}] = \mathbf{0}$$

(b)

For GMM estimation, we minimize the criterion function

$$\boldsymbol{Q}_T(\boldsymbol{\theta}|\boldsymbol{\mathcal{Y}}_T) = \boldsymbol{g}_T(\boldsymbol{\theta}|\boldsymbol{\mathcal{Y}}_T)'\ \boldsymbol{W}_T\ \boldsymbol{g}_T(\boldsymbol{\theta}|\boldsymbol{\mathcal{Y}}_T),$$

where

$$\boldsymbol{g}_T(\boldsymbol{\theta}|\boldsymbol{\mathcal{Y}}_T) \equiv \frac{1}{T}\sum_{t=1}^{T}\boldsymbol{h(\theta, y_t)}$$

and $\boldsymbol{W}_T$ is the weighting matrix.

We need to use numerical optimization to minimize $\boldsymbol{Q}_T(\boldsymbol{\theta}|\boldsymbol{\mathcal{Y}}_T)$. Using `optim` function in `R`, I got the GMM estimates of β and γ as

$$\hat{\beta} = 1.000 \quad \text{and} \quad \hat{\gamma} = 4.170.$$

I used the identity matrix as the weighting matrix and $(0.9, 3)$ as initial values for β and γ, respectively. I used the quasi Newton-Raphson method and set the boundary of the parameters as $(\beta, \gamma) \in [0,1] \times [0,10]$.

(c)

The first stage of GMM is to estimate $\hat{\theta}_{1st}$ with the identity matrix as the weighting matrix, which was done in part (b). So, $\hat{\theta}_{1st} = (1.000, 4.170)$. Then we use $\hat{\theta}_{1st}$ to estimate the the optimal weighting matrix

$$\hat{\boldsymbol{S}}_T(\hat{\boldsymbol{\theta}}_{1st}) = \frac{1}{T}\sum_{t=1}^{\infty} \boldsymbol{h}(\hat{\boldsymbol{\theta}}_{\boldsymbol{1st}}, \boldsymbol{y_t})\boldsymbol{h}(\hat{\boldsymbol{\theta}}_{\boldsymbol{1st}}, \boldsymbol{y_t})'$$

$$\implies \boldsymbol{W}_T^{opt} = [\hat{\boldsymbol{S}}_T(\hat{\boldsymbol{\theta}}_{1st})]^{-1}.$$

In R, I computed the estimate of the optimal weighting matrix as

$$\boldsymbol{W}_T^{opt} = \begin{bmatrix} 0.01826 & -14.04 & -0.01828 & -21.33 \\ -14.04000 & 47180.00 & 14.07000 & 10560.00 \\ -0.01828 & 14.07 & 0.01831 & 21.35 \\ -21.33000 & 10560.00 & 21.35000 & 57080.00 \end{bmatrix}.$$

(d)

Since we have four moment conditions ($r = 4$) and two parameters ($k = 2$), we want to test the validity of the $r - k$ overidentifying restrictions. The test statistic is

$$J_T(\hat{\boldsymbol{\theta}}_{2nd}) = T \cdot \boldsymbol{g}_T(\hat{\boldsymbol{\theta}}_{2nd})' \ \boldsymbol{W}_T^{opt} \ \boldsymbol{g}_T(\hat{\boldsymbol{\theta}}_{2nd}),$$

which converges to $\chi^2(r-k)$ by distribution.
With the optimal weighting matrix obtained in part (d), we run the second stage estimation to compute $\hat{\boldsymbol{\theta}}_{2nd}$. Then the test statistic is computed as

$$J_T(\hat{\boldsymbol{\theta}}_{2nd}) = 1.199,$$

of which p-value is 0.2745 for 2 degrees of freedom. Therefore, at a reasonable confidence level (say, 95 %), we fail to reject the null that the model is correctly specified. That is, we cannot reject the model.

Appendix: R Code

```
setwd('/Users/DSP/Dropbox/UCSC/Coursework/1stYear_3Q/211C/Exams/Final')
finalDat = read.csv("finalDat.csv")

##################################### Question 1 (b) #####################################
# Criterion function
critFn = function(theta,cVec,rVec,wMat){
  # Parameters
  beta = theta[1]
  gamma = theta[2]

  # Lagging/Leading the data
  TT  = length(cVec)
  C_tm1 = cVec[1:(TT-2)]
  C_t   = cVec[2:(TT-1)]
  C_tp1 = cVec[3:TT]
  r_tm1 = rVec[1:(TT-2)]
  r_t   = rVec[2:(TT-1)]
  r_tp1 = rVec[3:TT]

  # Moment conditions evaluated from data
  TTT = length(C_tm1)
  hMat = matrix(0,4,TTT)
  for(j in 1:TTT){
    hMat[,j] = (beta*(C_tp1[j]/C_t[j])^(-gamma)*(1+r_tp1[j])-1) * c(C_t[j],C_tm1[j],r_t[j],
        r_tm1[j])
  }

  # G vector
  gVec = apply(hMat,1,mean)

  # Criterion function
  Q = t(gVec) %*% wMat %*% gVec

  return(Q)
}

# GMM estimation of parameters (idendity matrix as weighting matrix)
gmmEst = optim(theta<-c(0.9,3), critFn, cVec=finalDat$Cons, rVec=finalDat$SPY, wMat=diag(4),
    method="L-BFGS-B", lower = c(0,0), upper = c(1,10))
thetaEst = gmmEst$par


##################################### Question 1 (c) #####################################
# A function to compute optimal weighting matrix
optWmat = function(theta,cVec,rVec){
  # Parameters
  beta = theta[1]
  gamma = theta[2]

  # Lagging/Leading the data
  TT  = length(cVec)
  C_tm1 = cVec[1:(TT-2)]
```

```
  C_t   = cVec[2:(TT-1)]
  C_tp1 = cVec[3:TT]
  r_tm1 = rVec[1:(TT-2)]
  r_t   = rVec[2:(TT-1)]
  r_tp1 = rVec[3:TT]

  # Moment conditions evaluated from data
  TTT = length(C_tm1)
  hMat = matrix(0,4,TTT)
  for(j in 1:TTT){
    hMat[,j] = (beta*(C_tp1[j]/C_t[j])^(-gamma)*(1+r_tp1[j])-1) * c(C_t[j],C_tm1[j],r_t[j],
        r_tm1[j])
  }

  # S matrix (i.e. inverse of optimal weighting matrix)
  sMat = (1/TTT) * hMat %*% t(hMat)

  # Optimal weighting matrix
  wMat = solve(sMat)

  return(wMat)
}

# Two-stage estimation of optimal weighting matrix
## First stage: using identity matrix for weighting matrix
gmmEst_1 = optim(theta<-c(0.9,3), critFn, cVec=finalDat$Cons, rVec=finalDat$SPY, wMat=diag(4),
    method="L-BFGS-B", lower=c(0,0), upper=c(1,10))
thetaEst_1 = gmmEst_1$par

## Second stage: using the optimal weighting matrix
wMatOpt = optWmat(theta=thetaEst_1, cVec=finalDat$Cons, rVec=finalDat$SPY)
gmmEst_2 = optim(theta<-c(0.9,3), critFn, cVec=finalDat$Cons, rVec=finalDat$SPY, wMat=wMatOpt,
    method="L-BFGS-B", lower=c(0,0), upper=c(1,10))
thetaEst_2 = gmmEst_2$par

##################################### Question 1 (d) #####################################
# Compute J statistic
jStat = TTT * critFn(thetaEst_2, finalDat$Cons, finalDat$SPY, wMatOpt)

# Compute p-value
dchisq(jStat, df=2)
```