R Codes For Final Project

William Kubin

Data import & manipulations

```
setwd(“/Users/paa.willie/Fall2019/Time Series/Project”)
mydata = read.csv(“GDP_swiss.csv”, header = TRUE)
library(astsa)
swiss.gdp = ts(mydata$GDP, start = c(1980, 1), end = c(2019, 2), frequency = 4)
tsplot(swiss.gdp, xlab=“YEARS”, ylab=“Millions of Swiss Francs”, main = “GDP of Switzerland”)
tsplot(diff(log(swiss.gdp)), xlab=“YEARS”, ylab=“Millions of Swiss Francs”, main = “Growth Rate of Switzerland”)
acf2(diff(log(swiss.gdp)))
```

Test of stationarity of growth rate of GDP

```
library(“tseries”)
adf.test(diff(log(swiss.gdp))) # ADF test
```

ARIMA model fit

```
sarima(log(swiss.gdp), p = 1, d = 1, q = 0) # ARMA(1,1,0) on growth rate
mean.data = mean(diff(log(swiss.gdp)))
```

Forecasting growth rate of GDP and Confidence Intervals

```
fCast = sarima.for(diff(log(swiss.gdp)), n.ahead = 24, p = 1, d = 0, q = 0)
fCast
text(2017, 0.021, “PAST”); text(2022, 0.021, “FUTURE”)
abline(h = mean.data, col = 4) # estimated mean
abline(v = 2019.25, lty = 2, col = 4) # vertical line showing second quarter of 2019
predictions = fCastprederrors = fCastse
lowerCI = c()
upperCI = c()
for (i in 1:24){
lowerCI[i] = predictions[i] - 1.96*errors[i]
upperCI[i] = predictions[i] + 1.96*errors[i]
}
```

```
lowerCI
upperCI
```

Spectral Estimation

parametric spectral estimation - AR estimation

```
AR.spec = spec.ar(diff(log(swiss.gdp)), col = 4)
abline(v = 0, col = 6, lty = “dashed”) # ?? year cycle
N = length(swiss.gdp)
c() -> AIC -> BIC
for (k in 1:30){
sigma2 = ar(diff(log(swiss.gdp)), order = k, aic = FALSE)$var.pred
BIC[k] = log(sigma2) + k*log(N)/N
AIC[k] = log(sigma2) + (N + 2*k)/N
}
IC = cbind(AIC, BIC + 1)
ts.plot(IC, type = “o”, xlab = “order (p)”, ylab = “AIC/BIC”, xlim = c(0, 10))
Grid()
```

non-parametric spectral estimation - Periodogram

raw periodogram

```
per = mvspec(diff(log(swiss.gdp)), col = rgb(.05, .6, .75), lwd = 2)
abline(v = 1/10, lty = “dashed”) # 10 year cycle
abline(v = 0.175, lty = “dashed”) # close to the El Nino Effect
abline(v = 3/10, lty = “dashed”)
abline(v = 5/10, lty = “dashed”)
per$details[1:50,]
par(mfrow = c(3,1))
x.t0 = mvspec(diff(log(swiss.gdp)), spans = c(9,9), col = rgb(.05, .6, .75), lwd = 2)
abline(v = 0.175, lty = “dashed”, col = 2) # close to the El Nino Effect
```

20% tapering

```
x.t1 = mvspec(diff(log(swiss.gdp)), spans = c(9,9), taper = 0.2, col = rgb(.05, .6, .75), lwd = 2)
abline(v = 0.175, lty = “dashed”, col = 2)
```

50% full tapering

```
x.t2 = mvspec(diff(log(swiss.gdp)), spans = c(9,9), taper = 0.5, col = rgb(.05, .6, .75), lwd = 2)
abline(v = 0.175, lty = “dashed”, col = 2)
```