Appendix: Code and Results

# Packages needed

library(forecast)  
library(zoo)  
library(plyr)  
library(urca)  
library(rugarch)  
library(pander)

# Load Data

SP500.raw <- read.csv("../data/S&P500.csv")  
VIX.raw <- read.csv("../data/Vix.csv")  
ReVol.raw <- read.csv("../data/RealizedVol.csv")

#Relized Volatility time series  
ReVol.df <- ReVol.raw[-(1:2),1:2]  
colnames(ReVol.df) <- c("Date", "Vol")  
ReVol.df$Date <- as.Date(ReVol.df$Date, format = "%Y%m%d")  
ReVol.df$Vol <- sqrt(as.numeric(as.character(ReVol.df$Vol)))  
ReVol.df<- ReVol.df[!is.na(ReVol.df$Vol),]  
ReVol.ts <- zoo(ReVol.df$Vol, ReVol.df$Date)  
  
#SP500 Close Price time series  
dates.sp <- as.Date(SP500.raw$Date,format="%Y-%m-%d")  
SP500.ts <- zoo(SP500.raw$Adj.Close,dates.sp)  
  
# SP500 Return time series  
SPret.ts <- diff(log(SP500.ts))  
  
# Vix Index time series  
dates.vix <- as.Date(VIX.raw$Date,format="%Y-%m-%d")  
VIX.ts <- zoo(VIX.raw$Adj.Close,dates.vix)  
  
# Align all Time Series   
startdate <- time(ReVol.ts)[1]  
SPret.ts <- window(SPret.ts, start = startdate)  
VIX.ts <- window(VIX.ts, start = startdate)

# Relized Volatility Forecasting

### Basic check for Relized Vol series

plot(ReVol.ts, ylim = c(0,0.06), cex.lab = 1.5, cex.main = 2,  
 main = "SP500 Daily Realized Volatility (5-min Sub-sampled)",  
 xlab = "Years",  
 ylab = "Realized Volatility (Standard Deviation)")

#Unit Root Test  
ReVol.ur <- ur.df(ReVol.ts, type = "none", selectlags="BIC")  
summary(ReVol.ur)

##   
## ###############################################   
## # Augmented Dickey-Fuller Test Unit Root Test #   
## ###############################################   
##   
## Test regression none   
##   
##   
## Call:  
## lm(formula = z.diff ~ z.lag.1 - 1 + z.diff.lag)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.026164 -0.001178 0.000104 0.001585 0.048236   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## z.lag.1 -0.039075 0.004912 -7.954 2.26e-15 \*\*\*  
## z.diff.lag -0.393531 0.013736 -28.649 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.003461 on 4477 degrees of freedom  
## Multiple R-squared: 0.1822, Adjusted R-squared: 0.1818   
## F-statistic: 498.6 on 2 and 4477 DF, p-value: < 2.2e-16  
##   
##   
## Value of test-statistic is: -7.9543   
##   
## Critical values for test statistics:   
## 1pct 5pct 10pct  
## tau1 -2.58 -1.95 -1.62

#ACF  
Acf(as.numeric(ReVol.ts), cex.lab = 1.5, cex.main = 2,   
 main = "Autocorrelation Plot of SP500 Realized Vol")  
#Pacf  
Pacf(as.numeric(ReVol.ts), cex.lab = 1.5, cex.main = 2,  
 main = "Partial Autocorrelation Plot of SP500 Realized Vol")

### Data Preprocessing

ma6.ts <- rollmean(ReVol.ts, k=6, align="right")  
Sign <- sign(SPret.ts) >0  
ret\_sign <- Sign \* SPret.ts  
  
lagvol <- cbind(ReVol.ts,  
 lag(ReVol.ts,-1,na.pad=TRUE),  
 lag(ReVol.ts,-2,na.pad=TRUE),   
 lag((ma6.ts),-2,na.pad=TRUE),  
 SPret.ts,  
 Sign,  
 ret\_sign)  
colnames(lagvol) <- c("vol", "volL1", "volL2", "ma6L2","Return", "Sign", "Return\*Sign")  
  
lagvol <- lagvol[complete.cases(lagvol),]  
  
# Data Partitioning  
ReVol.tra <- window(lagvol, end = "2015-01-02")  
ReVol.va <- window(lagvol, start = "2015-01-03")

### GRACH Model as Benchmark

# Specifications  
spec <- ugarchspec(variance.model=list(garchOrder=c(1,1)),  
 mean.model=list(armaOrder=c(0,0)))  
fittrain <- ugarchfit(spec = spec, data=ReVol.tra$Return)  
  
#forecast  
setfixed(spec) <- as.list(coef(fittrain))  
ugarchfilter <- ugarchfilter(spec=spec,data=ReVol.va$Return)  
  
# fitted volatility  
volfcast.grach<- zoo(sigma(ugarchfilter))  
  
#Validation set Forecasting errors of GARCH Model  
grach.acc <- accuracy(na.locf(as.ts(ReVol.va$vol)), na.locf(as.ts(volfcast.grach)))

plot(ReVol.va$vol, ylim = c(0,0.06), cex.lab = 1.5, cex.main = 2,  
 main = "SP500 Realized Volatility vs Forecasted Vol from GARCH",  
 xlab = "Years",  
 ylab = "Volatility (Standard Deviation)")  
lines(volfcast.grach, col="red")

### Benchmark Model For Relized Volatility

#ARMA model  
ReVol.arima <- auto.arima(zoo(ReVol.tra[,1], time(ReVol.tra)),d=0,ic="bic",seasonal=FALSE)  
#Best Arima model is AR2  
arima.ben <- lm(vol ~ volL1 + volL2,data=ReVol.tra)  
# MIDAS model  
midas.ben <- lm(vol ~ volL1 + ma6L2,data=ReVol.tra)  
  
  
arima.ben.pre <- predict(arima.ben, ReVol.va)  
arima.ben.res <- ReVol.va$vol- arima.ben.pre  
  
midas.ben.pre <- predict(midas.ben, ReVol.va)  
midas.ben.res <- ReVol.va$vol- midas.ben.pre

#Validation set Forecasting errors of AR2 Model  
arima.ben.acc <- accuracy(arima.ben.pre, ReVol.va$vol)  
arima.ben.acc

## ME RMSE MAE MPE MAPE  
## Test set -0.0004759644 0.002883321 0.001658777 -21.81033 33.52743

#Validation set Forecasting errors of MIDAS Model  
midas.ben.acc <- accuracy(midas.ben.pre, ReVol.va$vol)  
midas.ben.acc

## ME RMSE MAE MPE MAPE  
## Test set -0.00027188 0.002862302 0.0015601 -17.26183 30.55538

#Diebold/Mariano AR2 versus MIDAS  
dm.test(na.locf(as.ts(arima.ben.res)), na.locf(as.ts(midas.ben.res)))

##   
## Diebold-Mariano Test  
##   
## data: na.locf(as.ts(arima.ben.res))na.locf(as.ts(midas.ben.res))  
## DM = 0.56721, Forecast horizon = 1, Loss function power = 2,  
## p-value = 0.5707  
## alternative hypothesis: two.sided

### Leverage Effect as Dummy

midas.dummy <- lm(vol ~ volL1 + ma6L2 + Sign,data=ReVol.tra)  
midas.dummy.pre <- predict(midas.dummy, ReVol.va)  
midas.dummy.res <- ReVol.va$vol- midas.dummy.pre

#Validation set Forecasting errors of MIDAS Model with Dummy Leverage Effect  
midas.dummy.acc <- accuracy(midas.dummy.pre, ReVol.va$vol)  
midas.dummy.acc

## ME RMSE MAE MPE MAPE  
## Test set -0.0002959763 0.002811812 0.00152233 -16.75068 29.45156

#Diebold/Mariano MIDAS Model with Dummy Leverage Effect versus MIDAS Benchmark  
dm.test(na.locf(as.ts(midas.dummy.res)), na.locf(as.ts(midas.ben.res)))

##   
## Diebold-Mariano Test  
##   
## data: na.locf(as.ts(midas.dummy.res))na.locf(as.ts(midas.ben.res))  
## DM = -3.9266, Forecast horizon = 1, Loss function power = 2,  
## p-value = 9.169e-05  
## alternative hypothesis: two.sided

### Leaverage Effect as Return

midas.return <- lm(vol ~ volL1 + ma6L2 + Return, data=ReVol.tra)  
midas.return.pre <- predict(midas.return, ReVol.va)  
midas.return.res <- ReVol.va$vol- midas.return.pre

#Validation set Forecasting errors of MIDAS Model with Return  
midas.return.acc <- accuracy(midas.return.pre, ReVol.va$vol)  
midas.return.acc

## ME RMSE MAE MPE MAPE  
## Test set -0.0002618941 0.002753947 0.001496064 -16.51067 29.30235

#Diebold/Mariano MIDAS Model with Return versus MIDAS with dummy leverage effect  
dm.test(na.locf(as.ts(midas.return.res)), na.locf(as.ts(midas.dummy.res)))

##   
## Diebold-Mariano Test  
##   
## data: na.locf(as.ts(midas.return.res))na.locf(as.ts(midas.dummy.res))  
## DM = -2.4586, Forecast horizon = 1, Loss function power = 2,  
## p-value = 0.01411  
## alternative hypothesis: two.sided

### Best Forecasting Model: Leverage effect as Return and Cross-product term with sign dummy

midas.ret\_sign <- lm(vol ~ volL1 + ma6L2 + Return + `Return\*Sign`, data=ReVol.tra)  
midas.ret\_sign.pre <- predict(midas.ret\_sign, ReVol.va)  
midas.ret\_sign.res <- ReVol.va$vol- midas.ret\_sign.pre

#Validation set Forecasting errors of MIDAS Model with Return and Dummy  
midas.ret\_sign.acc <- accuracy(midas.ret\_sign.pre, ReVol.va$vol)  
midas.ret\_sign.acc

## ME RMSE MAE MPE MAPE  
## Test set -0.0003111873 0.002511126 0.001404977 -16.45516 27.87236

#Diebold/Mariano MIDAS Model with Return and Dummy versus MIDAS with Return  
dm.test(na.locf(as.ts(midas.ret\_sign.res)), na.locf(as.ts(midas.return.res)))

##   
## Diebold-Mariano Test  
##   
## data: na.locf(as.ts(midas.ret\_sign.res))na.locf(as.ts(midas.return.res))  
## DM = -3.2588, Forecast horizon = 1, Loss function power = 2,  
## p-value = 0.001154  
## alternative hypothesis: two.sided

RMSE <- rep(NA,6)  
RMSE[1] <- grach.acc[2]  
RMSE[2] <- arima.ben.acc[2]  
RMSE[3] <- midas.ben.acc[2]  
RMSE[4] <- midas.dummy.acc[2]  
RMSE[5] <- midas.return.acc[2]  
RMSE[6] <- midas.ret\_sign.acc[2]  
  
pander(data.frame(RMSE = RMSE,   
 row.names = c("GARCH Model",  
 "AR2 Benchmark",  
 "MIDAS Benchmark",  
 "Leverage Effect as Dummy Model",  
 "Leverage Effect as Return Model",  
 "Leverage Effect with Return and Dummy Model")))

summary(midas.ret\_sign)

##   
## Call:  
## lm(formula = vol ~ volL1 + ma6L2 + Return + `Return\*Sign`, data = ReVol.tra)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -0.022113 -0.001516 -0.000351 0.001093 0.055451   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 0.0007296 0.0001022 7.141 1.11e-12 \*\*\*  
## volL1 0.4038146 0.0148035 27.278 < 2e-16 \*\*\*  
## ma6L2 0.3836716 0.0165200 23.225 < 2e-16 \*\*\*  
## Return -0.1834257 0.0075254 -24.374 < 2e-16 \*\*\*  
## `Return\*Sign` 0.2942447 0.0132994 22.125 < 2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 0.003191 on 3733 degrees of freedom  
## Multiple R-squared: 0.7425, Adjusted R-squared: 0.7422   
## F-statistic: 2691 on 4 and 3733 DF, p-value: < 2.2e-16

# Final Conclusion

vix.plot <- window(VIX.ts, start = "2015-01-05")/sqrt(252)/100  
  
plot(ReVol.va$vol, ylim = c(0,0.06), cex.lab = 1.5, cex.main = 2,  
 main = "SP500 Realized Volatility and Forecastings vs Vix Index ",  
 xlab = "Years",  
 ylab = "Volatility (Standard Deviation)")  
lines(volfcast.grach, col="red")  
lines(zoo(midas.ret\_sign.pre, time(ReVol.va$vol)), col="blue")  
lines(vix.plot, col = "green")  
legend("topright",   
 legend=c("Realized Vol", "GARCH Forecast",   
 "MIDAS with leverage effect forecast","Vix" ),  
 col=c("black", "red", "blue", "green"),lty=1:1, cex=1.5)

# Potential Financial Applications

### Control Volatility

vol.fcast <- midas.ret\_sign.pre  
# set for 10 percent annual standard deviation  
target <- 0.05/sqrt(252)  
# now portfolio weight vector  
weight <- target/vol.fcast  
  
# dynamic portfolio (assuming 3 percent/year interest)  
pret <- ReVol.va[,5]\*weight + 0.03/252 \*(1-weight)  
pretEquity <- ReVol.va[,5]\* 1  
  
# Volatility for portfolios (daily to annual)  
pstd <- sqrt(252)\*ReVol.va[,1]\*weight  
pstdEquity <- sqrt(252)\*ReVol.va[,1]\*1  
  
sharp <- rep(NA,2)  
ret.mean <- rep(NA,2)  
ret.sd <- rep(NA,2)  
sd.sd <- rep(NA,2)  
  
ret.mean[1] <- mean(pret)\*252  
ret.mean[2] <- mean(pretEquity)\*252  
ret.sd[1] <- sd(pret)\*sqrt(252)  
ret.sd[2] <- sd(pretEquity)\*sqrt(252)  
sd.sd[1] <- sd(pstd)  
sd.sd[2] <-sd(pstdEquity)  
sharp <- (ret.mean-0.03)/ret.sd  
  
pander(data.frame(SharpRatio = sharp,  
 MeanReturn = ret.mean,  
 SDReturn = ret.sd,  
 SDofSD = sd.sd,  
 row.names = c("Vol Control Method", "Direct Investment")))

plot(pstd,ylim=c(0,0.8), cex.lab = 1.5, cex.main = 2,  
 main = "SP500 Realized Volatility vs Vol after Control ",  
 xlab = "Years",  
 ylab = "Annualized Std")  
lines(pstdEquity,col="red")  
grid()