Global Tactical Asset Allocation, Replication

matReturns.com

```
    require(quantmod)
    require(PerformanceAnalytics)
     # Define the GTAA Function
GTAA_returns <- function(startDate = '2000-01-01',
                         endDate = '2023-12-31',
                         SMA.n=10) {
  # 5 major Global Asset Classes represented by ETF tickers
  symbols <- c("SPY", # SPDR S&P 500
               "VGK", # Vanguard FTSE Europe Index
               "IEF", # iShares 7-10 Year Treasury Bond
               "VNQ", # Vanguard  Real Estate Index
               "GSG") # iShares S&P GSCI Commodity-Indexed Trust
  # Get adjusted prices from the tickers and combine prices
  prices <- list()
  for(i in 1:length(symbols)) {
    etfPrices <- Ad(get(getSymbols(symbols[i],
                                   from = startDate,
                                   to = endDate)))
    colnames(etfPrices) <- symbols[i]
    prices[[i]] <- etfPrices
  }
  prices <- na.omit(do.call(cbind, prices))
  # Convert to monthly prices and calculate returns
  monthlyPrices <- to.monthly(prices, OHLC=FALSE)
  colnames(monthlyPrices) <- symbols
  retsGlobalFive <- na.omit(Return.calculate(monthlyPrices))

  # Create SMA data of five assets
  SMAglobal <- list()
  for(i in 1:length(symbols)) {
    smaPrices <- SMA(monthlyPrices[,i], SMA.n)
    colnames(smaPrices) <- paste0("SMA_" , symbols[i])
    SMAglobal[[i]] <- smaPrices
  }
  SMAglobal <- na.omit(do.call(cbind, SMAglobal))

  # Fit monthly price data to the same length as SMA data
  ratioPrices <- monthlyPrices[
    ((nrow(monthlyPrices)-nrow(SMAglobal))+1):nrow(monthlyPrices), ]

  # Create Ratio data of five assets, make a list,
  # calculate ratio of Price/SMA, cbind list
  ratio <- list()
  for(i in 1:length(symbols)) {
```

```
    priceTOsma <- (ratioPrices[,i]/SMAglobal[,i]) - 1
    colnames(priceTOsma) <- paste0("P2SMA_" , symbols[i])
    ratio[[i]] <- priceTOsma
  }
  ratio <- na.omit(do.call(cbind, ratio))

  # Create signals if ratio > 0, return TRUE else FALSE
  sigs <- list()
  for(i in 1:length(symbols)) {
    signals <- ratio[,i] > 0
    colnames(signals) <- paste0("Sig_" , symbols[i])
    sigs[[i]] <- signals
  }
  sigs <- na.omit(do.call(cbind, sigs))
  # Lag sig dataframe to even up rows with rets in the same period (months)
  # Make global 5 assets return data the same length as sig data
  lagSigs <- na.omit(lag(sigs))
  sigRets <- retsGlobalFive[
    ((nrow(retsGlobalFive)-nrow(sigs))+2) : nrow(retsGlobalFive), ]

  # Multiply returns to sig list, bind the table
  GTAA5rets <- list()
  for(i in 1:length(symbols)) {
    returns <- lagSigs[,i] * sigRets[,i]
    colnames(returns) <- paste0("GTAA_rets_" , symbols[i])
    GTAA5rets[[i]] <- returns
  }
  GTAA5rets <- na.omit(do.call(cbind, GTAA5rets))

  # Sum rows to get cumulative sum
  GTAA5 <- list()
  for(i in 1:nrow(GTAA5rets)) {
    sums <- sum(GTAA5rets[i, 1:length(symbols)])
    GTAA5[[i]] <- sums
  }
  GTAA5 <- na.omit(do.call(rbind, GTAA5))
  colnames(GTAA5) <- "GTAA5"
  # Divide by 5 to get portfolio returns with equal weight
  GTAA5df <- cbind(GTAA5rets, (GTAA5/length(symbols)))
  out <- GTAA5df$GTAA5
  return(out)
}
startDate = '2000-01-01'
endDate = '2023-12-31'
SMA.n=10
gtaaReturns <- GTAA_returns(startDate, endDate, SMA.n)
```

Thanks for reading, please visit matReturns.com

Creative Commons License

This work is licensed under a Creative Commons Attribution 4.0 International License.

Note: The code and data provided in this analysis are for illustrative purposes only and do not constitute financial advice. Investors should conduct thorough due diligence before making any investment decisions.