

# STAT430\_\_HW01

*Taiga Hasegawa(taigah2)*

*2019/1/25*

```
amazon = read.csv("AMZN_2012-06-21_34200000_57600000_message_10.csv", header = F)
names(amazon)=c("Time" , "Type", "OrderID", "Size", "Price", "Direction")
amazon$Size <- as.numeric(amazon$Size)
amazon$Price <- as.numeric(amazon$Price)

library("lubridate")

## Warning: package 'lubridate' was built under R version 3.4.4
##
## Attaching package: 'lubridate'
## The following object is masked from 'package:base':
##
##      date
demodate="2016-06-21"
options(digits.secs=3)
amazon$tstamp=as_datetime(demodate,tz="US/Eastern")+amazon$Time

## Warning in as.POSIXlt.POSIXct(x, tz): unknown timezone 'zone/tz/2018g.1.0/'
## zoneinfo/America/Chicago'

amazon_subset=subset(amazon, Type %in% c(4,5))
head(amazon_subset,n=5)

## Warning in as.POSIXlt.POSIXct(x, tz): unknown timezone 'zone/tz/2018g.1.0/'
## zoneinfo/America/Chicago'

##      Time Type  OrderID Size  Price Direction      tstamp
## 1  34200.02   5         0   1 2238200        -1 2016-06-21 09:30:00.017
## 33 34200.19   4 11885113  21 2238100         1 2016-06-21 09:30:00.190
## 34 34200.19   4 11534792  26 2237500         1 2016-06-21 09:30:00.190
## 38 34200.37   5         0 100 2238400        -1 2016-06-21 09:30:00.372
## 39 34200.38   5         0 100 2238400        -1 2016-06-21 09:30:00.375

imbalance_tick <- function(dat)
{
  n <- length(dat$Price)
  imbalance <- rep(0, n)
  price_diff <- diff(dat$Price)
  for(i in 2:n)
  {
    imbalance[i] <- sign(price_diff[i-1])*(price_diff[i-1]!=0) + imbalance[i-1]*(price_diff[i-1]==0)
  }
  imbalance
}

library(pracma)

## Warning: package 'pracma' was built under R version 3.4.4
```

```

Tstar_tib=function(data,w0=10,bkw_T=5,bkw_b=5){
  b_t=imbalance_tick(data)
  w0=max(min(which(cumsum(b_t)!=0)),w0)
  Tstar=w0
  E0t=Tstar
  repeat{
    Tlast=sum(Tstar)
    nbt=min(bkw_T,Tlast-1)
    P=pracma::movavg(b_t[1:Tlast],n=nbt,type="e")
    P<- tail(P,1)
    b_t_expected=E0t*abs(P)
    b_t_cumsum=abs(cumsum(b_t[-(1:Tlast)]))
    if(max(b_t_cumsum)<b_t_expected){break}else{
      Tnew=min(which(b_t_cumsum>=b_t_expected))
    }
    Tlast=Tlast+Tnew
    if(Tlast>dim(data)){break}else{
      Tstar=c(Tstar,Tnew)
      if(length(Tstar)<=2){
        E0t=mean(Tstar)
      }else{
        nt=min(bkw_T,length(Tstar)-1)
        E0t=pracma::movavg(Tstar[1:length(Tstar)],n=nt,type="e")
        E0t <- tail(E0t,1)
      }
    }
  }
  return(Tstar)
}

##Tick imbalance bar
tib=Tstar_tib(amazon_subset)

## Warning in if (Tlast > dim(data)) {: the condition has length > 1 and only
## the first element will be used

## Warning in if (Tlast > dim(data)) {: the condition has length > 1 and only
## the first element will be used

## Warning in if (Tlast > dim(data)) {: the condition has length > 1 and only
## the first element will be used

## Warning in if (Tlast > dim(data)) {: the condition has length > 1 and only
## the first element will be used

## Warning in if (Tlast > dim(data)) {: the condition has length > 1 and only
## the first element will be used

imbalance_volume <- function(dat)
{
  n <- length(dat$Price)
  imbalance <- rep(0, n)

```

```

b_t=rep(0,n)
price_diff <- diff(dat$Price)
v_t=dat$Size
for(i in 2:n)
{
  b_t[i] <- sign(price_diff[i-1])*(price_diff[i-1]!=0) + b_t[i-1]*(price_diff[i-1]==0)
  imbalance[i]=b_t[i]*v_t[i]
}
imbalance
}

```

```

Vstar_vib=function(data,w0=10,bkw_T=5,bkw_b=5){
  bv_t=imbalance_volume(data)
  w0=max(min(which(cumsum(bv_t)!=0)),w0)
  Tstar=w0
  E0t=Tstar
  repeat{
    Tlast=sum(Tstar)
    nbt=min(bkw_b,Tlast-1)
    P=pracma::movavg(bv_t[1:Tlast],n=nbt,type="e")
    P=tail(P,1)
    bv_t_expected=E0t*abs(P)
    bv_t_cumsum=abs(cumsum(bv_t[-(1:Tlast)]))
    if(max(bv_t_cumsum)<bv_t_expected){break}else{
      Tnew=min(which(bv_t_cumsum>=bv_t_expected))
    }
    Tlast=Tlast+Tnew
    if(Tlast>dim(data)[1]){break}else{
      Tstar=c(Tstar,Tnew)
      if(length(Tstar)<=2){
        E0t=mean(Tstar)
      }else{
        nt=min(bkw_T,length(Tstar)-1)
        E0t=pracma::movavg(Tstar[1:length(Tstar)],n=nt,type="e")
        E0t=tail(E0t,1)
      }
    }
  }
  return(Tstar)
}

```

```

##Volume imbalance bar
vib=Vstar_vib(amazon_subset)
cumsum(vib)

```

```
## [1] 10 42 55 245 724
```

```

Tstar_trb <- function(dat, w0=10, bkw_T=5, bkw_Pb1=5)
{
  b_t <- imbalance_tick(dat)
  nb <- length(b_t)
  nx <- dim(dat)[1]
  th_T <- sapply(1:nb, function(i){
    b_t_tmp <- b_t[1:i]
    if(sum(b_t_tmp %in% c(-1,1))==0){out <- 0}else

```

```

{
  out <- max(cumsum(b_t_tmp[b_t_tmp==1]), -cumsum(b_t_tmp[b_t_tmp==1]))
}
out
})
w0 <- max(min(which(th_T != 0)), w0)
w0 <- max(min(which(b_t==1)), w0)
Tvec <- w0
EOT <- T_last <- Tvec
Pb1 <- sum(b_t[1:w0]==1) / w0
Pb1vec <- Pb1
th_T_Expected <- EOT*max(Pb1, 1-Pb1)
while(T_last<nx)
{
  T_last <- sum(Tvec)
  for(j in 1:(nb-T_last-1))
  {
    b_t_tmp <- b_t[(T_last+1):(T_last+j)]
    if(sum(b_t_tmp %in% c(-1,1))==0){th_T_tmp <- 0}else
    {
      th_T_tmp <- max(cumsum(b_t_tmp[b_t_tmp==1]), -cumsum(b_t_tmp[b_t_tmp==1]))
    }
    if(th_T_tmp >= th_T_Expected)
    {
      new_flag <- TRUE
      T_new <- j
      Tvec <- c(Tvec, T_new)
      T_last <- T_last + T_new

      Pb1_new <- sum(b_t_tmp==1) / j
      Pb1vec <- c(Pb1vec, Pb1_new)
      break
    }
  }
}

if(new_flag==TRUE)
{
  new_flag <- FALSE
  nTvec <- length(Tvec)
  if(nTvec <= 2)
  {
    EOT <- mean(Tvec)
    Pb1 <- mean(Pb1vec)
  }else
  {
    nT <- min(bkw_T, length(Tvec)-1)
    EOT <- pracma::movavg(Tvec[(nTvec-nT):nTvec], n=nT, type = "e")
    EOT <- EOT[length(EOT)]
    nPb1 <- min(bkw_Pb1, length(Tvec)-1)
    Pb1 <- pracma::movavg(Pb1vec[(nTvec-nPb1):nTvec], n=nPb1, type = "e")
    Pb1 <- Pb1[length(Pb1)]
  }
  th_T_Expected <- EOT*max(Pb1, 1-Pb1)
}

```

```

    }else{break}
  }
  return(Tvec)
}

##Tick runs bar
trb=Tstar_trb(amazon_subset)

Vstar_vrb <- function(dat, w0=10, bkw_T=5, bkw_Pb1=5, bkw_v=5)
{
  b_t <- imbalance_tick(dat)
  nb <- length(b_t)
  nx <- dim(dat)[1]
  volume=amazon_subset$Size
  th_T <- sapply(1:nb, function(i){
    b_t_tmp <- b_t[1:i]
    volume_tmp=volume[1:i]
    if(sum(b_t_tmp %in% c(-1,1))==0){out <- 0}else
    {
      index1=which(b_t_tmp==1)
      index2=which(b_t_tmp==-1)
      index_for_1=c(index1)
      index_for_2=c(index2)
      out <- max(cumsum(b_t_tmp[index1]*volume_tmp[index1]), -cumsum(b_t_tmp[index2]*volume_tmp[index2])
    }
    out
  })

  w0 <- max(min(which(th_T != 0)), w0)
  w0 <- max(min(which(b_t==1)), w0)
  Tvec <- w0
  EOT <- T_last <- Tvec
  Pb1 <- sum(b_t[1:w0]==1) / w0
  Pb1vec <- Pb1
  bool1=b_t[1:w0]==1
  nw1=min(bkw_v,sum(bool1))
  volume_average_1=pracma::movavg(volume[bool1],n=nw1,type = "e")
  volume_average_1=volume_average_1[length(volume_average_1)]
  bool2=b_t[1:w0]==-1
  nw2=min(bkw_v,sum(bool2))
  volume_average_2=pracma::movavg(volume[bool2],n=nw2,type = "e")
  th_T_Expected <- EOT*max(Pb1*volume_average_1, (1-Pb1)*volume_average_2)
  volume_average_2=volume_average_2[length(volume_average_2)]

  while(T_last<nx)
  {
    T_last <- sum(Tvec)
    for(j in 1:(nb-T_last-1))
    {
      b_t_tmp <- b_t[(T_last+1):(T_last+j)]
      volume_tmp=volume[(T_last+1):(T_last+j)]
      if(sum(b_t_tmp %in% c(-1,1))==0){th_T_tmp <- 0}else
      {
        index1=which(b_t_tmp==1)

```

```

        index2=which(b_t_tmp==1)
        th_T_tmp <- max(cumsum(b_t_tmp[index1]*volume_tmp[index1]), -cumsum(b_t_tmp[index2]*volume_tmp[
    ]
    }
    if(th_T_tmp >= th_T_Expected)
    {
        new_flag <- TRUE
        T_new <- j
        Tvec <- c(Tvec, T_new)
        T_last <- T_last + T_new

        Pb1_new <- sum(b_t_tmp==1) / j
        Pb1vec <- c(Pb1vec, Pb1_new)
        break
    }
}

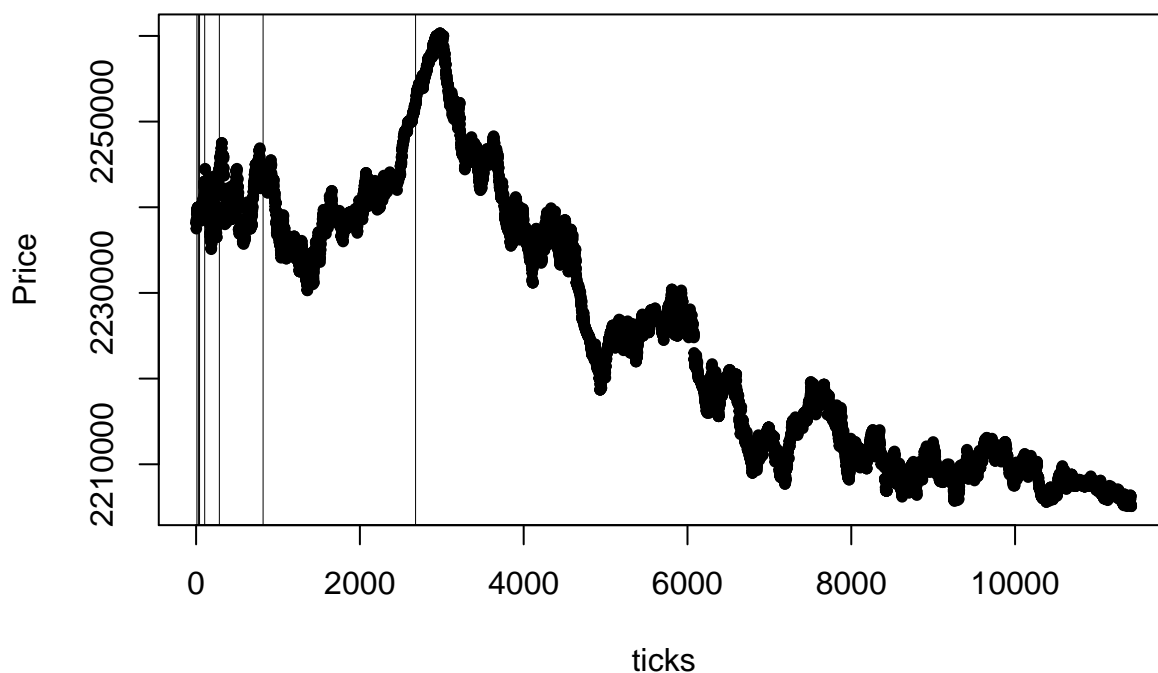
if(new_flag==TRUE)
{
    new_flag <- FALSE
    nTvec <- length(Tvec)
    bool1=b_t[1:T_last]==1
    nw1=min(bkw_v,sum(bool1))
    volume_average_1=pracma::movavg(volume[bool1],n=nw1,type = "e")
    volume_average_1=volume_average_1[length(volume_average_1)]
    bool2=b_t[1:T_last]==-1
    nw2=min(bkw_v,sum(bool2))
    volume_average_2=pracma::movavg(volume[bool2],n=nw2,type = "e")
    volume_average_2=volume_average_2[length(volume_average_2)]
    if(nTvec <= 2)
    {
        EOT <- mean(Tvec)
        Pb1 <- mean(Pb1vec)
    }else
    {
        nT <- min(bkw_T, length(Tvec)-1)
        EOT <- pracma::movavg(Tvec[(nTvec-nT):nTvec], n=nT, type = "e")
        EOT <- EOT[length(EOT)]
        nPb1 <- min(bkw_Pb1, length(Tvec)-1)
        Pb1 <- pracma::movavg(Pb1vec[(nTvec-nPb1):nTvec], n=nPb1, type = "e")
        Pb1 <- Pb1[length(Pb1)]
    }
    th_T_Expected <- EOT*max(Pb1*volume_average_1, (1-Pb1)*volume_average_2)
}else{break}
}
return(Tvec)
}

vrb=Vstar_vrb(amazon_subset)

plot(amazon_subset$Price, pch=20, xlab="ticks", ylab="Price", main="Where to sample tick imbalance bars")
abline(v=cumsum(tib), lwd=0.2)

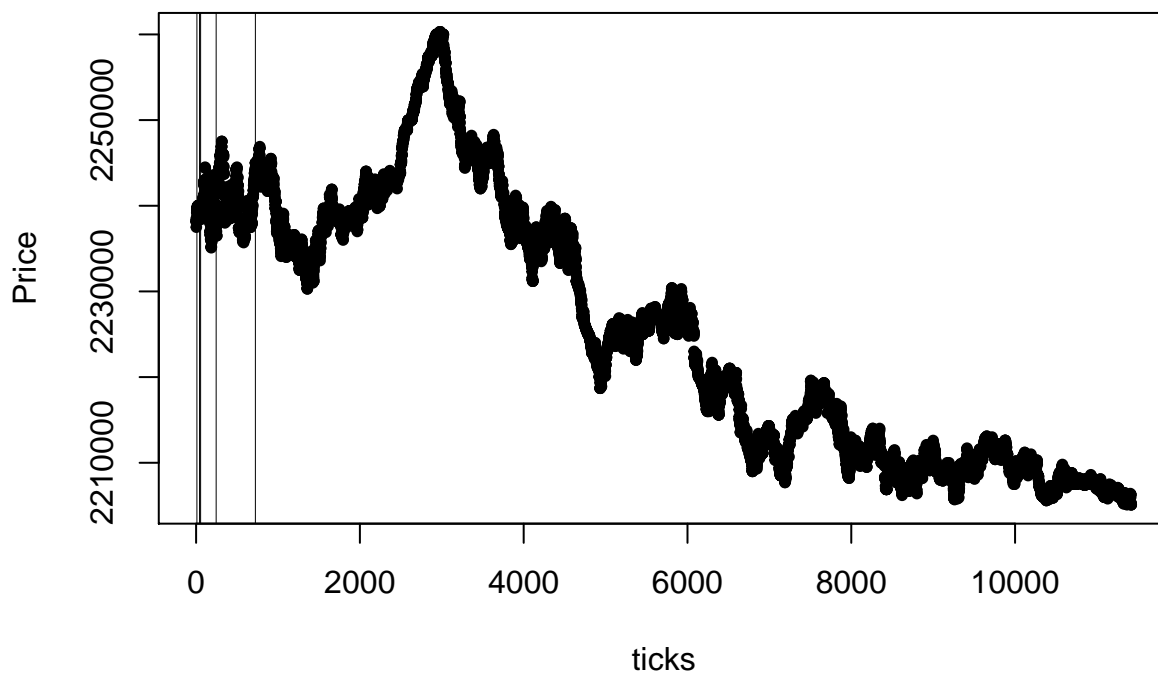
```

## Where to sample tick imbalance bars?



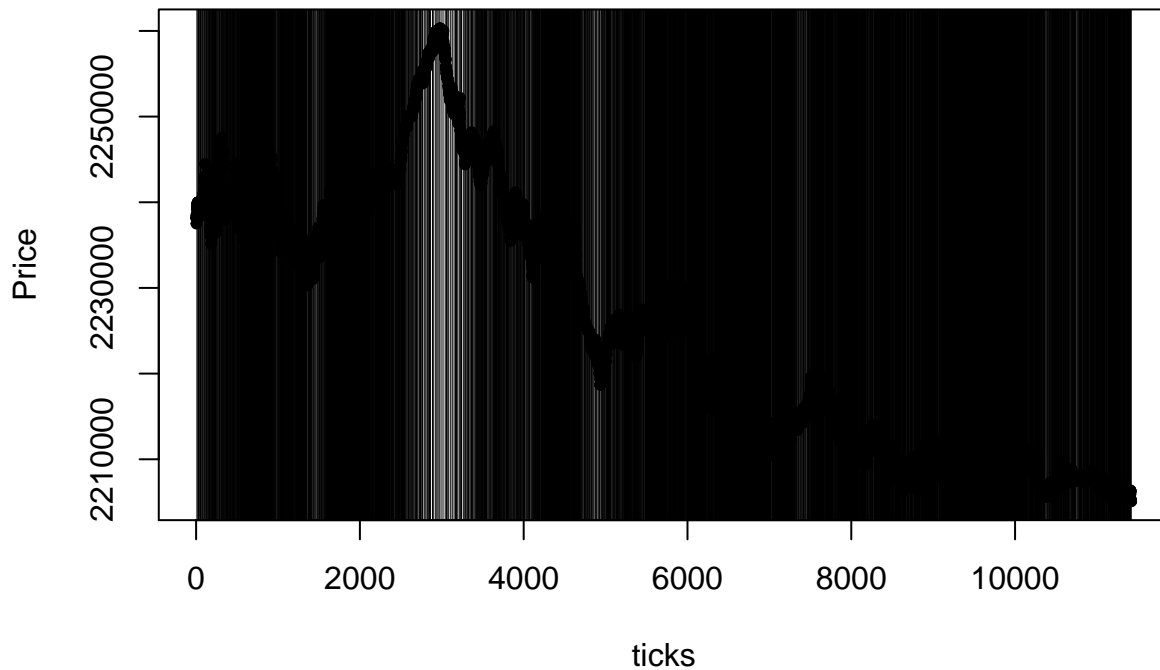
```
plot(amazon_subset$Price, pch=20, xlab="ticks", ylab="Price", main="Where to sample volume imbalance bars")
abline(v=cumsum(vib), lwd=0.2)
```

## Where to sample volume imbalance bars?



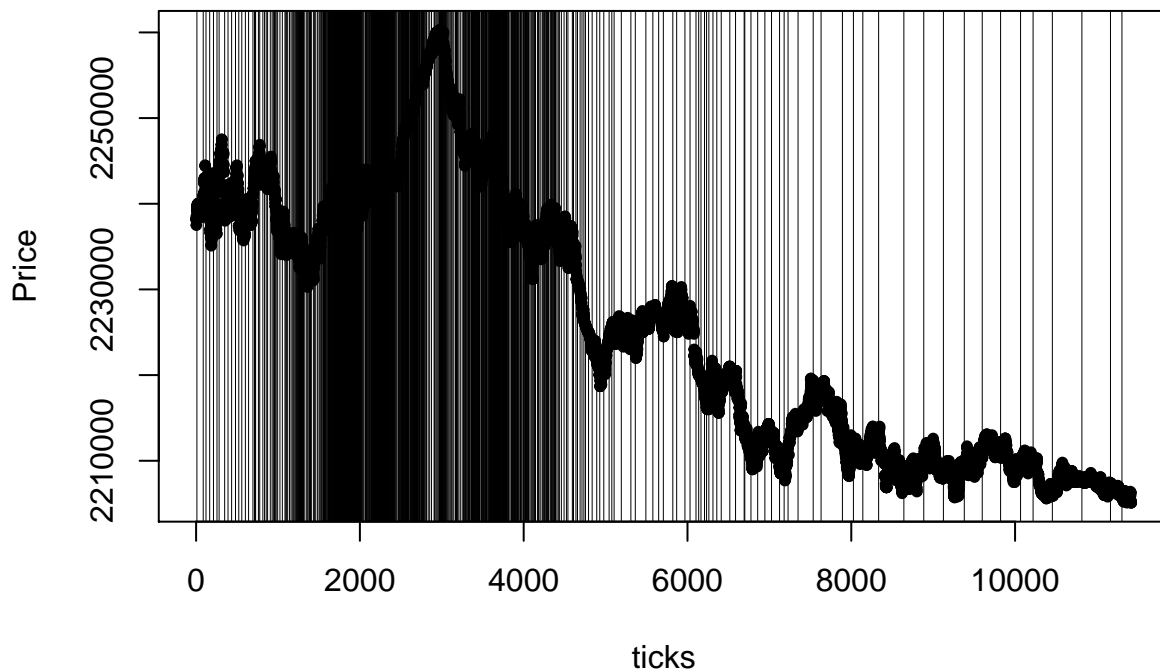
```
plot(amazon_subset$Price, pch=20, xlab="ticks", ylab="Price", main="Where to sample tick runs bars?")
abline(v=cumsum(trb), lwd=0.2)
```

## Where to sample tick runs bars?



```
plot(amazon_subset$Price, pch=20, xlab="ticks", ylab="Price", main="Where to sample volume runs bars?")
abline(v=cumsum(vrb), lwd=0.2)
```

## Where to sample volume runs bars?



Tick imbalance bars show up at 10, 28, 42, 105, 284, 818 and 2680 ticks. Volume imbalance bars show up at 10, 42, 55, 245, and 724 ticks. They have the same values but volume imbalance bars have fewer values. Tick runs bars and volume runs bars have much more bars than imbalance bars. The number of bars are



increasing when the price goes up and down in both runs bars. Tick runs bars have more bars than volume runs bars. They are all made for detecting the unusual deviation from the normal price or volume. Runs bars are more useful than imbalance bars because the run bars shows the bars more obviously when the price goes up and down.