Homework #6

Siavash Mortezavi, Jason Liu, Holly Capell, Sangyu Shen, Kunal Kotian

October 5, 2017

Stock Questions

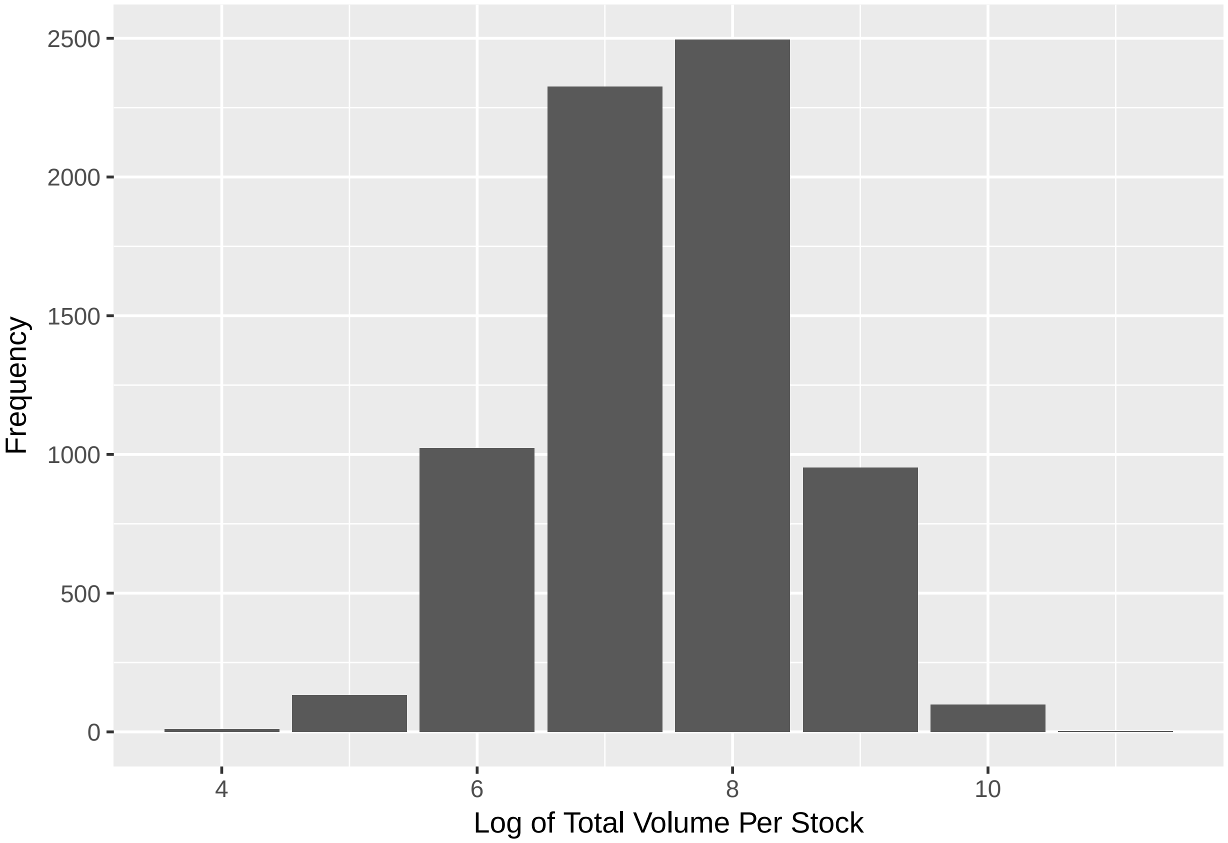
(1)

```
q_1 <- c("select avg(ret::numeric)
from
(select count(1) over () as total, ROW_NUMBER() over (order by ret), ret
from stocks2016.d2010
where
retdate = '2010-01-04' and ret != 'B' and ret!= 'C') as lhs
where ROW_NUMBER in ((total+1)/2,(total + 2)/2);")
dbGetQuery(first_database, q_1)
```

```
##         avg
## 1 0.0161615
```

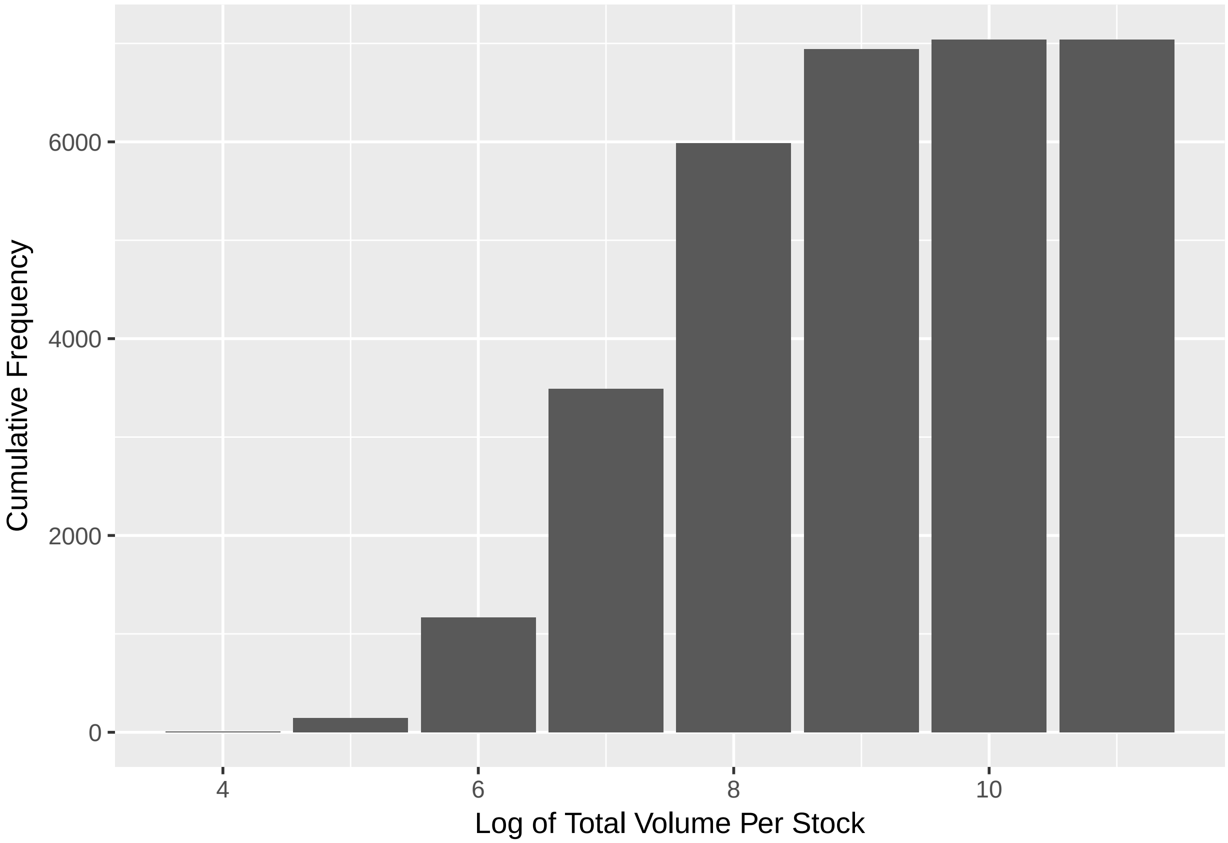
(2)

```
q_2 <- c("select count(permno), round(sumOfVol) as sv
FROM
    (select permno, log(sum(vol)) as sumOfVol
    FROM stocks2016.d2010
    where date_part('year', retdate) = 2010 and vol!=0
    group by permno) as LHS
group by 2;")
frq2 = dbGetQuery(first_database,q_2)
frq2 %>% ggplot() +
  geom_bar(aes(x = frq2$sv, y = frq2$count), stat = "identity") +
  xlab('Log of Total Volume Per Stock') + ylab('Frequency')
```



(3)

```
qn_pt1_3 <- c("select sum(count) over (ORDER BY sv) as cumul_sum, sv
FROM
    (select count(permno), round(log(sumOfVol)) as sv
    FROM
        (select permno, sum(vol) as sumOfVol
        FROM stocks2016.d2010
        where date_part('year', retdate) = 2010 and vol!=0 group by permno) as LHS
    group by sv) as allof;")
frq3 = dbGetQuery(first_database,qn_pt1_3)
frq3 %>% ggplot() +
  geom_bar(aes(x = frq3$sv, y = frq3$cumul_sum), stat = "identity") +
  xlab('Log of Total Volume Per Stock') +
  ylab('Cumulative Frequency')
```

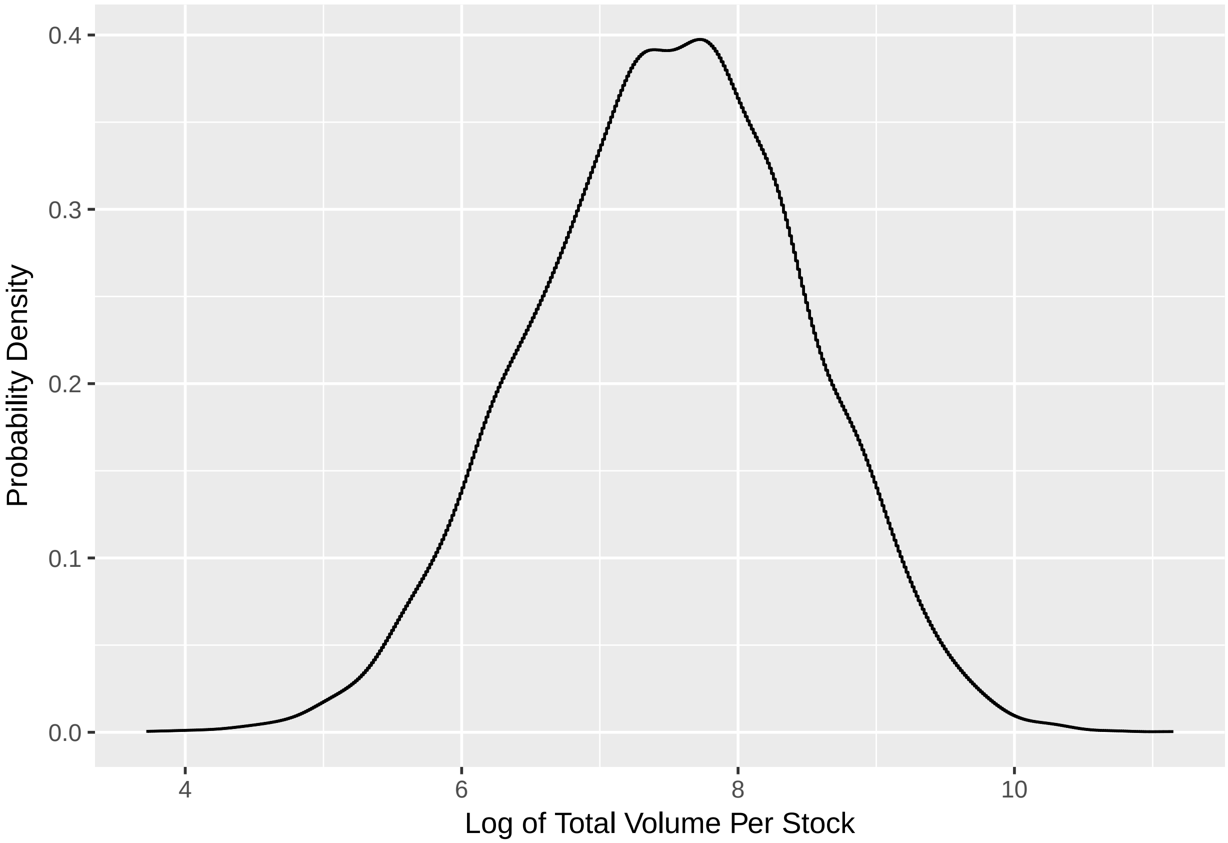


(4)

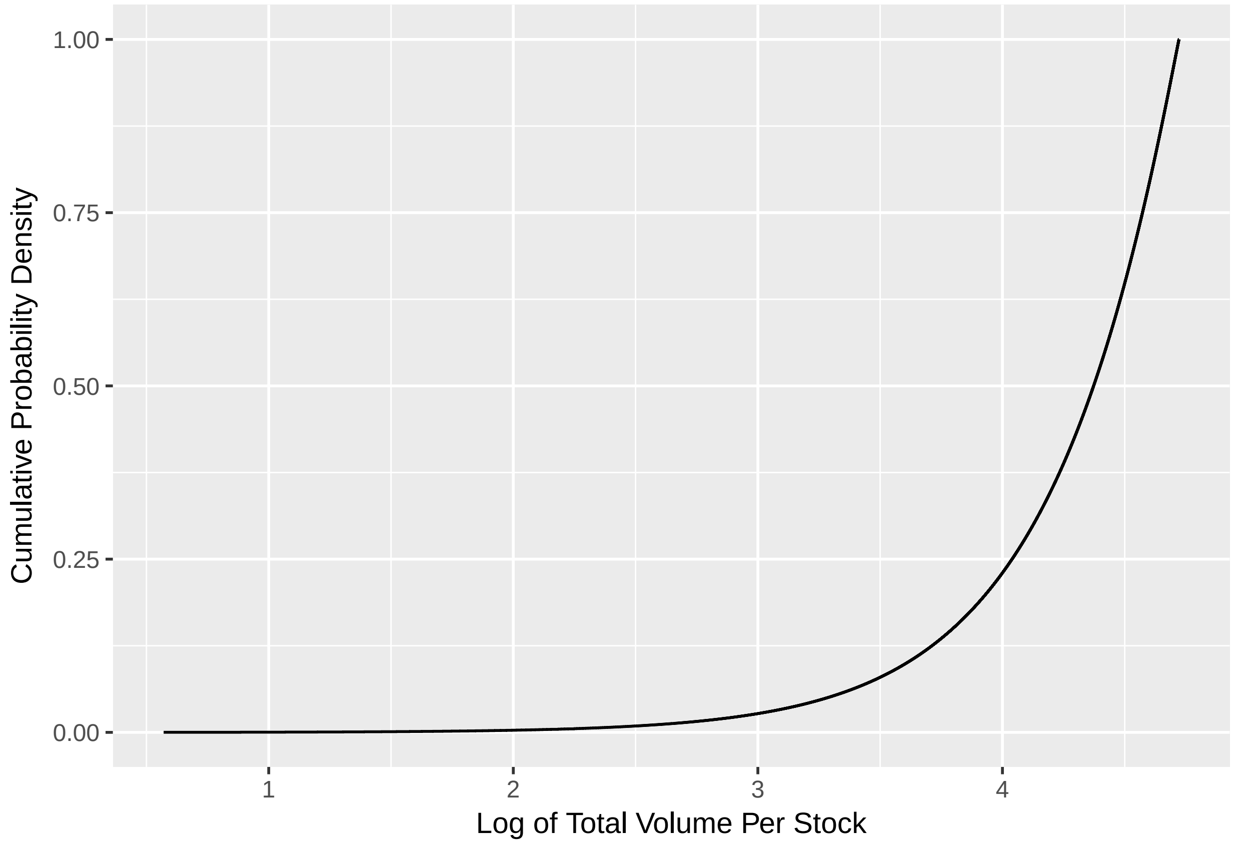
```
# Extract data for plotting
q_4 <- c("select log(sumOfVol) as logSumofVol,
log(sum(log(sumOfVol)) over(order by log(sumOfVol))) as cumlogSumofVol
from
    (select sum(vol) as sumOfVol
    from stocks2016.d2010
    where vol <> 0
    group by permno) as InnerQ;")
(frq4 = dbGetQuery(first_database, q_4))
```

```
##     logsumofvol cumlogsumofvol
## 1      3.718253      0.5703389
## 2      3.949390      0.8846619
## 3      4.041393      1.0685211
## 4      4.161368      1.2005880
## 5      4.269513      1.3040577
##  [ reached getOption("max.print") -- omitted 7036 rows ]
```

```
# Generate plots:
# PDF of the distribution of the log of total volume per stock
frq4 %>%
  ggplot(aes(x = frq4$logsumofvol)) + stat_density(geom = "step") +
  xlab('Log of Total Volume Per Stock') +
  ylab('Probability Density')
```



```
# CDF of the distribution of the cumulative log of total volume per stock
frq4 %>% ggplot(aes(x = frq4$cumlogsumofvol)) +
  stat_ecdf(geom = "step", pad = "FALSE") +
  xlab('Log of Total Volume Per Stock') +
  ylab('Cumulative Probability Density')
```



(5)

```
q_5 <- c("select tic, count(tic) over(order by tic) -1 as count from stocks2016.fnd
where fyear = 2010 and tic is not null and tic >= 'A'
order by tic;")
dbGetQuery(first_database, q_5)
```

```
##            tic count
## 1            A     0
## 2           AA     1
## 3        AABVF     2
## 4         AACC     3
## 5         AACS     4
##  [ reached getOption("max.print") -- omitted 10837 rows ]
```

(6)

```
# Note:
# The month of dec is ignored in the calculation of the number of successes
# as well as possibilities - because we are asked to use 2010 data and
# 'next month' does not exist in 2010 for Dec.
q_6 <- c("select num_successes::float / (select (count(distinct permno) * 11)
from stocks2016.d2010) as probability
from
    (select count(1) as num_successes
    from(
        select permno, month, prc_diff_current,
lead(prc_diff_current, 1) over() as prc_diff_next
        from
```

```
            (select permno, month, (close_val - open_val) as prc_diff_current
            from
                (select permno, date_part('month', retdate) as month,
                last_value(abs(prc)) over(partition by permno,
date_part('month', retdate)) as close_val,
                first_value(abs(prc)) over(partition by permno,
date_part('month', retdate)) as open_val
                from stocks2016.d2010
                where prc is not null and retdate is not null) as InnerQ1
            group by 1, 2, 3
            order by 1, 2) as InnerQ2) as InnerQ3
    where prc_diff_current > 0 and prc_diff_next > 0
and month <> 12) as InnerQ4;")
dbGetQuery(first_database, q_6)
```

```
##   probability
## 1   0.2393973
```

(7)

```
q_7 <- c("select prc, permno, permco, retdate,
prc - lag(prc) over () as nominal_diff
from stocks2016.d2010
where prc is not null;")
dbGetQuery(first_database, q_7)
```

```
##                   prc permno permco    retdate  nominal_diff
## 1             3.01000  83399  14498 2010-02-19            NA
## 2            -3.11000  83399  14498 2010-02-22      -6.12000
##  [ reached getOption("max.print") -- omitted 1658749 rows ]
```

(8)

```
q_8 <- c("select RHS2.prc, LHS.permno, LHS.permco, RHS2.retdate,
RHS2.prc - lag(RHS2.prc) over (partition by LHS.permno,LHS.permco) as nominal_diff
from
(select distinct permno, permco from stocks2016.d2010) as LHS
cross join
(select distinct retdate from stocks2016.d2010) as RHS
left join stocks2016.d2010 as RHS2
on LHS.permno = RHS2.permno and LHS.permco = RHS2.permco and RHS.retdate = RHS2.retdate;")
dbGetQuery(first_database, q_8)
```

```
##                   prc permno permco    retdate nominal_diff
## 1            11.10000  10001   7953 2010-04-26           NA
## 2            11.80000  10001   7953 2010-08-16      0.70000
##  [ reached getOption("max.print") -- omitted 1791214 rows ]
```

(9)

```
q_9 <- c("select permno,permco,  sum(case1)
from
(
select permno, permco, case when prc >= max_9 then 1 else 0 end as case1
from
(
select permno,permco, prc, .9*max(prc) over(partition by permno)as max_9
from stocks2016.d2010) as lhs) as more
group by permno, permco;")
dbGetQuery(first_database, q_9)
```

```
##      permno permco sum
## 1     10001   7953  90
## 2     10002   7954   8
## 3     10025   7975  14
##  [ reached getOption("max.print") -- omitted 7105 rows ]
```

(10)

```
q_10 <- c("select lhs.retdate,lhs.permno, lhs.permco, lhs.prc, lhs.row_num as numberofdays
from
(select retdate,permno, permco, prc,
ROW_NUMBER() over(partition by permno, permco order by retdate) as row_num
,max(prc) over(partition by permno, permco) as maxp
from stocks2016.d2010 order by permno, permco, retdate) as lhs
where lhs.prc = lhs.maxp;
")
dbGetQuery(first_database, q_10)
```

```
##         retdate permno permco         prc numberofdays
## 1    2010-08-03  10001   7953     12.3500          147
## 2    2010-04-29  10002   7954      6.3000           81
##  [ reached getOption("max.print") -- omitted 7751 rows ]
```

(11)

```
q_11 <- c("select permno, permco, numdays
from(
select permno, permco, retdate as date1,
sum(different_days) over (partition by permno, permco order
by retdate asc) as numdays, prc, maximum
from(
select permno, permco, retdate,
case when retdate - lag(retdate) over () > 0
then retdate - lag(retdate) over ()
else Null end as different_days,
prc, max(prc) over(partition by permco, permno) as maximum
from stocks2016.d2010
order by permno, permco, retdate
limit 10000) as table1) as table2
where maximum = prc;")
dbGetQuery(first_database, q_11)
```

```
##    permno permco numdays
## 1   10001   7953     211
## 2   10002   7954     115
## 3   10025   7975      10
##  [ reached getOption("max.print") -- omitted 45 rows ]
```

LTV Questions

(1)

```r
q_LTV1 <- c("select sum(case when dt = min_date and transtype = 'Unit'
and subs > 0 then 1 else 0 end)::float/count(distinct userid) as percent
from
(select userid, transtype, dt,
min(dt) over (partition by userid) as min_date,
sum(case when transtype = 'Subscription' then 1 else 0 end)
over (partition by userid) as subs from
cls.ltv) q1;")
```

(2)

```r
q_LTV2 <- c("select distinct(userid) from
(select userid from cls.ltv where transtype = 'Unit') as LHS
inner join
(select userid from cls.ltv where transtype = 'Subscription') as RHS
using(userid);")
```

(3)

```r
q_LTV3 <- c("select avg(time_diff) from
(select dt - lag(dt) over (partition by userid order by dt) as time_diff from cls.ltv
where transtype = 'Unit') q1
where time_diff is not null;")
```

(4)

```r
q_LTV4 <- c("select avg(time_diff) from
(select count(1) over () as total, row_number() over (order by time_diff), time_diff from
(select dt - lag(dt) over (partition by userid order by dt) as time_diff from cls.ltv
where transtype = 'Unit') q1
where time_diff is not null)q2
where row_number in ((total+1)/2,(total + 2)/2);")
```

(5)

```r
q_LTV5 <- c("select distinct userid, max(amt) over (partition by userid, index), index
from
(select * from
(select userid, amt, dt, NTILE(4) over (partition by userid order by amt) as index,
min(dt) over (partition by userid) as min_date
where dt > current_date - '6 month'::interval
from Trans) as tot
where min_date < current_date - '6 month'::interval) as selected
where index < 4;")
```

(6)

```r
q_LTV6 <- c("select row_number as months, avg(total_rev) as month_mult
from
(select cohort,
row_number() over (partition by cohort, month,
rev/lag(rev) over (partition by cohort order by month)) as total_rev
from
(select cohort, date_trunc('month', dt)::date as month, sum(amt) as rev
from
(select userid, amt, date_trunc('month',
first_value(dt) over (partition by userid order by dt))::date as cohort, dt
from cls.ltv) q1
where cohort < date_trunc('month', now())::date
group by 1,2)q2)q3
group by 1;")
```

In order to find the LTV of a user up until a certain month, you can multiply the initial (first month) revenue by each of the month multipliers preceding that month and sum these amounts. This calculation gives the revenue that a customer will generate after x months.