```r
library(tidyverse)
library(knitr)
```

Sampling Frame

Download the data

```r
# file path to csv with addresses
aru_file_path <-
  "https://opendata.arcgis.com/datasets/c3c0ae91dca54c5d9ce56962fa0dd645_68.csv"

ap_file_path <-
  "https://opendata.arcgis.com/datasets/aa514416aaf74fdc94748f1e56e7cc8a_0.csv"

# create a directory for downloading the data
if (!dir.exists("data/")) {
  dir.create("data")
}

# if the data doesn't already exist, download the data
if (!file.exists("data/aru.csv")) {
  download.file(aru_file_path, "data/aru.csv")
}

if (!file.exists("data/ap.csv")) {
  download.file(ap_file_path, "data/ap.csv")
}
```

Address Residential Units

The first dataset is Address Residential Units

The dataset does not contain a variable for quadrant, so we extract quadrant from the full address.

```r
aru <- read_csv("data/aru.csv") %>%
  rename_all(tolower) %>%
  select(unit_id, address_id, fulladdress, status, unitnum, unittype)

# extract quadrant
aru <- aru %>%
  mutate(quadrant = str_sub(fulladdress, start = -2, end = -1))
```

Address Residential Units contains residential units with status set to "RETIRED". We drop these cases as well.

```r
count(aru, status) %>%
  kable()
```

status	n
ACTIVE	244046
ASSIGNED	47
RETIRE	7087

```r
aru <- aru %>%
  filter(status != "RETIRE")
```

Adress Points

```r
# load the data and convert the variable names to lower case
ap <- read_csv("data/ap.csv", guess_max = 10000) %>%
  rename_all(tolower) %>%
  select(address_id, status, type_, entrancetype, quadrant, fulladdress,
         objectid_1, assessment_nbhd, cfsa_name, census_tract, vote_prcnct,
         ward, zipcode, anc, census_block, census_blockgroup, latitude,
         longitude, active_res_unit_count, res_type, active_res_occupancy_count)
```

Address Points contains residential units, non-residential units, and mixed-use units. Residential units and mixed-use units contain residences that belong to our sampling frame. We drop non-residential units.

```r
count(ap, res_type) %>%
  kable()
```

res_type	n
MIXED USE	473
NON RESIDENTIAL	15807
RESIDENTIAL	131370

```r
ap <- ap %>%
  filter(res_type != "NON RESIDENTIAL")
```

Address points contains residential units with status set to “RETIRED”. We drop these cases as well.

```r
count(ap, status) %>%
  kable()
```

status	n
ACTIVE	128490
ASSIGNED	668
RETIRE	2675
TEMPORARY	10

```
ap <- ap %>%
  filter(status != "RETIRE")
```

After the above filtering, there are 98 observations from Address Points and 3,706 observations in Address Residential Units that have missing addresses. We investigated joining the two datasets on `address_id` to fill in the address but all records missing an address in one dataset were missing an address in the other dataset.

We dropped the missing values which represented about 1.5 percent of observations in Address Residential Units and 0.07 percent of observations in Address Points.

```
ap <- ap %>%
  filter(!is.na(fulladdress))

aru <- aru %>%
  filter(!is.na(fulladdress))
```

Merge variables

Address Points has interesting variables not present in Address Residential Units. So we merge the Address Points dataset with the Address Residential Units dataset. The join works for all but 572 cases, most of which are in a new building at the Wharf.

```
aru_expanded <- aru %>%
  select(-status) %>%
  left_join(ap, by = c("fulladdress", "address_id")) %>%
  select(quadrant = quadrant.x, everything(), -quadrant.y)

anti_join(aru, ap, by = c("fulladdress", "address_id"))
```

```
## # A tibble: 572 x 7
##    unit_id address_id fulladdress         status unitnum unittype quadrant
##      <dbl>      <dbl> <chr>               <chr>  <chr>   <chr>    <chr>
##  1  223379     276680 600 WATER STREET SW ACTIVE 6-12    RENTAL   SW
##  2  223380     276680 600 WATER STREET SW ACTIVE 6-13    RENTAL   SW
##  3  223381     276680 600 WATER STREET SW ACTIVE 6-14    RENTAL   SW
##  4  223384     276680 600 WATER STREET SW ACTIVE 1-1     RENTAL   SW
##  5  223389     276680 600 WATER STREET SW ACTIVE 1-6     RENTAL   SW
##  6  223392     276680 600 WATER STREET SW ACTIVE 1-9     RENTAL   SW
##  7  223494     276680 600 WATER STREET SW ACTIVE 8-16    RENTAL   SW
##  8  223497     276680 600 WATER STREET SW ACTIVE 9-3     RENTAL   SW
##  9  223503     276680 600 WATER STREET SW ACTIVE 9-9     RENTAL   SW
## 10  223508     276680 600 WATER STREET SW ACTIVE 9-14    RENTAL   SW
## # ... with 562 more rows
```

```
rm(aru)
```

Combination

Next, we need to drop addresses in the Address Points dataset that exist in the Address Residential Units dataset so we don't overcount addresses in multi-dwelling units.

```
ap <- ap %>%
  filter(!address_id %in% unique(aru_expanded$address_id))
```

Finally, we can combine the two datasets to create a sampling frame that contains approximately every residential address in Washington D.C.

```
sampling_frame <- bind_rows(ap, aru_expanded)

rm(ap, aru_expanded)

#summarize_all(addresses, list(~sum(is.na(.))))

write_csv(sampling_frame, "sampling_frame.csv")
```

Pilot survey

```
set.seed(20190714)

pilot_sample <- sampling_frame %>%
  group_by(quadrant) %>%
  sample_n(25)

write_csv(pilot_sample, "data/pilot_sample.csv")

rm(pilot_sample)
```

Picking stratum sizes

Sample mean

We begin with a derivation of Exact Optimal Sample Allocation for $\bar{y}$.

Decomposition of $V(\bar{y}_h)$:

By Wright (12.4), $V(\bar{y}_{str}) = \sum_{h=1}^{H}(\frac{N_h}{N})^2 V(\bar{y}_h) = \sum_{h=1}^{H}(\frac{N_h}{N})^2 \frac{N_h - n_h}{N_h} \frac{S_h^2}{n_h}$

$V(\bar{y}_h) = (\frac{N_h}{N})^2 \frac{N_h - n_h}{N_h} \frac{S_h^2}{n_h}$

$V(\bar{y}_h) = (\frac{N_h^2}{N^2})(1 - \frac{n_h}{N_h})\frac{S_h^2}{n_h}$

$V(\bar{y}_h) = (\frac{N_h^2 S_h^2}{N^2})(\frac{1}{n_h}) - \frac{N_h^2 n_h S_h^2}{N^2 N_h n_h}$

$V(\bar{y}_h) = (\frac{N_h^2 S_h^2}{N^2})(\frac{1}{n_h}) - \frac{N_h S_h^2}{N^2}$

$V(\bar{y}_h) = (\frac{N_h^2 S_h^2}{N^2})(1 - \frac{1}{1 \cdot 2} - \frac{1}{2 \cdot 3} - ... - \frac{1}{n_h(n_h - 1)}) - \frac{N_h S_h^2}{N^2}$

$V(\bar{y}_h) = \frac{N_h(N_h - 1)S_h^2}{N^2} - \frac{N_h^2 S_h^2}{N^2 \cdot 1 \cdot 2} - \frac{N_h^2 S_h^2}{N^2 \cdot 2 \cdot 3} - ... - -\frac{N_h^2 S_h^2}{N^2 n_h(n_h - 1)}$

Decomposition of $V(\bar{y}_{str})$

$V(\bar{y}_{str}) = \sum_{h=1}^{H} \frac{N_h(N_h - 1)S_h^2}{N^2}$

$-\frac{N_1^2 S_1^2}{N^2 \cdot 1 \cdot 2} - \frac{N_1^2 S_1^2}{N^2 \cdot 2 \cdot 3} - ... - -\frac{N_1^2 S_1^2}{N^2 n_1(n_1 - 1)}$

...

$-\frac{N_h^2 S_h^2}{N^2 \cdot 1 \cdot 2} - \frac{N_h^2 S_h^2}{N^2 \cdot 2 \cdot 3} - ... - -\frac{N_h^2 S_h^2}{N^2 n_h(n_h - 1)}$

...

$-\frac{N_H^2 S_H^2}{N^2 \cdot 1 \cdot 2} - \frac{N_H^2 S_H^2}{N^2 \cdot 2 \cdot 3} - ... - -\frac{N_H^2 S_H^2}{N^2 n_H(n_H - 1)}$

For a desired bound V_0 on the sampling variance $V(\bar{y}_{str})$, we may find an optimal allocation using the followng algorithm:

1) Assign, for each stratum, 1 unit to be selected for the sample.

2) Fill in the following table and number these values starting from 1, in decreasing order.

$\frac{N_1^2 S_1^2}{N^2 \cdot 1 \cdot 2}$	$\frac{N_1^2 S_1^2}{n^2 \cdot 2 \cdot 3}$	$\frac{N_1^2 S_1^2}{N^2 \cdot 3 \cdot 4}$	...
$\frac{N_2^2 S_2^2}{N^2 \cdot 1 \cdot 2}$	$\frac{N_2^2 S_2^2}{N^2 \cdot 2 \cdot 3}$	$\frac{N_2^2 S_2^2}{N^2 \cdot 3 \cdot 4}$	...
.	.	.	...
.	.	.	...
.	.	.	...
$\frac{N_H^2 S_H^2}{N^2 \cdot 1 \cdot 2}$	$\frac{N_H^2 S_H^2}{N^2 \cdot 2 \cdot 3}$	$\frac{N_H^2 S_H^2}{N^2 \cdot 3 \cdot 4}$	...

3) Since the initial allocation is $(n_{11}, n_{21}, ..., n_{H1}) = (1, 1, ..., 1)$, compute $V(\bar{y}_{str}|n_{11} = 1, n_{21} = 1, ..., n_{H1} = 1) = \sum_{h=1}^{H} \frac{N_h(N_h - 1)S_h^2}{N^2}$

4) Pick value (1) from the table and increase the associated stratum's sample size by 1, o that the updated allocation is $(n_{12}, n_{22}, ..., n_{H2})$, where exactly one of the n_{h2}'s is equal to 2 and the rest are equal to 1. Then, compute $V(\bar{y}_{str}|n_{12}, ..., n_{H2} = V(\bar{y}_{str}|n_{11}, ..., n_{H1}) - \frac{1}{N^2}$ where "(1)" represents the largest value

from the table. If $V(\bar{y}_{str}|N_{12}, ..., n_{H2} \leq V_0$, then stop with $n_1 = n_{12}, ..., N_H = N_{H2}$. Otherwise, go to step 5.

5) Pick value (2) from the table and increase the associated stratum's sample size by 1, so that the updated allocation is $(n_{13}, ..., n_{H3})$. Then compute $V(\bar{y}_{str}|n_{13}, ..., n_{H3}) = V(\bar{y}_{str}|n_{12}, ..., n_{H2} - \frac{(2)}{N^2}$, where "(2)" represents the second value from the table. If $V(\bar{y}_{str}|n_{13}, ..., N_H = n_{H3}$. Otherwise, continue until step j, where $V(\bar{y}_s tr|n_{1j}, ..., n_{Hj}) \leq V_0$. The final allocation is $n_{1j}, ..., n_{Hj}$ and $n = n_{1j} + \cdots + n_{Hj}$.

```r
# load the completed pilot survey and clean the values
pilot_sample <- read_csv("data/pilot_sample_completed.csv") %>%
  mutate(land_value = ifelse(!is.na(rf_land_value),
                             rf_land_value,
                             land_value),
         improvement_value = ifelse(!is.na(rf_improvement_value),
                                    rf_improvement_value,
                                    improvement_value)) %>%
  mutate(property_value = land_value + improvement_value) %>%
  mutate(property_value = ifelse(unittype == "RENTAL" &
                                 active_res_occupancy_count > 4 &
                                 property_value > 2000000,
                               property_value / active_res_occupancy_count,
                               property_value
                               ))
```

```r
# find Nh and s2 for each strata
# (1) and (2)
s_squared_h <- pilot_sample %>%
  group_by(stratum = quadrant) %>%
  summarize(s_squared_h = var(property_value, na.rm = TRUE),
            missing_prop = mean(is.na(property_value)))

Nh <- sampling_frame %>%
  count(stratum = quadrant) %>%
  rename(Nh = n)

strata <- left_join(s_squared_h, Nh, by = "stratum") %>%
  # adjust N because of missingness
  mutate(Nh = Nh * (1 - missing_prop)) %>%
  mutate(N = sum(Nh))

rm(s_squared_h, Nh)

kable(strata)
```

stratum	s_squared_h	missing_prop	Nh	N
NE	55231295979	0.08	68953.08	297153.2
NW	728182282168	0.12	166931.60	297153.2
SE	136823871969	0.28	49423.68	297153.2
SW	25025018879	0.20	11844.80	297153.2

Step 3: $\hat{V}(\bar{y}|1,1,1,1) = \sum_{h=1}^{H}(\frac{N_h}{N})^2\frac{N_h-n_h}{N_H}\frac{s_h^2}{n_h} = \sum_{h=1}^{H}(\frac{N_h}{N})^2\frac{N_h-1}{N_H}\frac{s_h^2}{1}$

(Wright 12.5)

```
# Let the initial allocation be (n_11, n_21, n_31, n_41) = (1, 1, 1, 1)
# (3) and (6)
starting_variance <- strata %>%
  mutate(strata_variance = Nh * (Nh - 1) * s_squared_h / N^2)

kable(starting_variance)
```

stratum	s_squared_h	missing_prop	Nh	N	strata_variance
NE	55231295979	0.08	68953.08	297153.2	2973894581
NW	728182282168	0.12	166931.60	297153.2	229802057101
SE	136823871969	0.28	49423.68	297153.2	3784970868
SW	25025018879	0.20	11844.80	297153.2	39758730

```
starting_variance <- starting_variance %>%
  summarize(V = sum(strata_variance)) %>%
  pull()

starting_variance
```

```
## [1] 236600681280
```

Step 3:

Prioirty value $= \frac{N_1^2 \cdot s_1^2}{N_1^2 \cdot n_h(n_h-1)}$

```
# create a table of priority values
# (4) and (5)
n_strata <-
  tibble(n = c(1:500, 1:500, 1:500, 1:500),
       stratum = c(rep("NE", 500), rep("NW", 500), rep("SE", 500), rep("SW", 500))) %>%
  left_join(strata, by = "stratum")

# step 2
priority_values <- n_strata %>%
  group_by(stratum) %>%
  mutate(priority_value = (Nh ^ 2 * s_squared_h) / (n * lag(n) * N ^ 2)) %>%
  ungroup() %>%
  arrange(desc(priority_value))

kable(head(select(priority_values, -missing_prop), n = 10))
```

n	stratum	s_squared_h	Nh	N	priority_value
2	NW	728182282168	166931.6	297153.2	114901716867
3	NW	728182282168	166931.6	297153.2	38300572289

n	stratum	s_squared_h	Nh	N	priority_value
4	NW	728182282168	166931.6	297153.2	19150286144
5	NW	728182282168	166931.6	297153.2	11490171687
6	NW	728182282168	166931.6	297153.2	7660114458
7	NW	728182282168	166931.6	297153.2	5471510327
8	NW	728182282168	166931.6	297153.2	4103632745
9	NW	728182282168	166931.6	297153.2	3191714357
10	NW	728182282168	166931.6	297153.2	2553371486
11	NW	728182282168	166931.6	297153.2	2089122125

Step 4:

```
# (7)
priority_values <- priority_values %>%
  mutate(agg_priority_value = cumsum(priority_value)) %>%
  mutate(marginal_variance = starting_variance - agg_priority_value) %>%
  mutate(marginal_sd = sqrt(marginal_variance))

kable(head(select(priority_values, -missing_prop, -N), n = 50), digits = 0)
```

n	stratum	s_squared_h	Nh	priority_value	agg_priority_value	marginal_variance	marginal_sd
2	NW	728182282168	166932	114901716867	114901716867	121698964413	348854
3	NW	728182282168	166932	38300572289	153202289155	83398392124	288788
4	NW	728182282168	166932	19150286144	172352575300	64248105980	253472
5	NW	728182282168	166932	11490171687	183842746987	52757934293	229691
6	NW	728182282168	166932	7660114458	191502861444	45097819835	212362
7	NW	728182282168	166932	5471510327	196974371771	39626309508	199064
8	NW	728182282168	166932	4103632745	201078004517	35522676763	188475
9	NW	728182282168	166932	3191714357	204269718874	32330962406	179808
10	NW	728182282168	166932	2553371486	206823090360	29777590920	172562
11	NW	728182282168	166932	2089122125	208912212485	27688468795	166399
2	SE	136823871969	49424	1892523726	210804736211	25795945069	160611
12	NW	728182282168	166932	1740935104	212545671315	24055009965	155097
2	NE	55231295979	68953	1486968855	214032640170	22568041110	150227
13	NW	728182282168	166932	1473098934	215505739104	21094942176	145241
14	NW	728182282168	166932	1262656229	216768395333	19832285946	140827
15	NW	728182282168	166932	1094302065	217862697399	18737983881	136887
16	NW	728182282168	166932	957514307	218820211706	17780469574	133343
17	NW	728182282168	166932	844865565	219665077271	16935604008	130137
18	NW	728182282168	166932	750991614	220416068885	16184612395	127219
19	NW	728182282168	166932	671939865	221088008749	15512672530	124550
3	SE	136823871969	49424	630841242	221718849991	14881831288	121991
20	NW	728182282168	166932	604745878	222323595870	14277085410	119487
21	NW	728182282168	166932	547151033	222870746902	13729934377	117175
22	NW	728182282168	166932	497410030	223368156932	13232524348	115033
3	NE	55231295979	68953	495656285	223863813217	12736868063	112858
23	NW	728182282168	166932	454156984	224317970201	12282711079	110827
24	NW	728182282168	166932	416310568	224734280769	11866400511	108933
25	NW	728182282168	166932	383005723	225117286492	11483394788	107161
26	NW	728182282168	166932	353543744	225470830236	11129851043	105498

n	stratum	s_squared_h	Nh	priority_value	agg_priority_value	marginal_variance	marginal_sd
27	NW	728182282168	166932	327355319	225798185555	10802495725	103935
4	SE	136823871969	49424	315420621	226113606176	10487075104	102406
28	NW	728182282168	166932	303972796	226417578972	10183102308	100911
29	NW	728182282168	166932	283009155	226700588127	9900093153	99499
30	NW	728182282168	166932	264141878	226964730005	9635951275	98163
4	NE	55231295979	68953	247828143	227212558147	9388123133	96892
31	NW	728182282168	166932	247100466	227459658613	9141022666	95609
32	NW	728182282168	166932	231656687	227691315301	8909365979	94389
33	NW	728182282168	166932	217616888	227908932189	8691749091	93230
34	NW	728182282168	166932	204815895	228113748083	8486933196	92125
35	NW	728182282168	166932	193112129	228306860212	8293821067	91070
5	SE	136823871969	49424	189252373	228496112585	8104568695	90025
36	NW	728182282168	166932	182383678	228678496263	7922185017	89007
37	NW	728182282168	166932	172525100	228851021363	7749659917	88032
38	NW	728182282168	166932	163444832	229014466195	7586215085	87099
39	NW	728182282168	166932	155063046	229169529241	7431152039	86204
5	NE	55231295979	68953	148696886	229318226126	7282455153	85337
40	NW	728182282168	166932	147309893	229465536020	7135145260	84470
41	NW	728182282168	166932	140124045	229605660065	6995021215	83636
42	NW	728182282168	166932	133451471	229739111536	6861569744	82835
43	NW	728182282168	166932	127244426	229866355962	6734325317	82063

```r
rm(strata, n_strata)
```

Proportion

We begin with a derivation of Exact Optimal Sample Allocation for $\hat{p}$.

Decomposition of $V(\hat{p}_{str})$

By Wright (12.14), $V(\hat{p}_{str}) = \sum_{h=1}^{H}(\frac{N_h}{N})^2 V(\hat{p}_h) = \sum_{h=1}^{H}(\frac{N_h}{N})^2 \frac{N_h-n_h}{N_h-1}\frac{\hat{p}(1-\hat{p})}{n_h}$

$V(\hat{p}_h) = (\frac{N_h}{N})^2 \frac{N_h-n_h}{N_h-1}\frac{\hat{p}(1-\hat{p})}{n_h}$

$V(\hat{p}_h) = \frac{N_h^2}{N^2}\frac{N_h}{N_h-1}\frac{\hat{p}(1-\hat{p})}{n_h} - \frac{N_h^2}{N^2}\frac{n_h}{N_h-1}\frac{\hat{p}(1-\hat{p})}{n_h}$

$V(\hat{p}_h) = \frac{N_h^2}{N^2}\frac{N_h}{N_h-1}\frac{\hat{p}(1-\hat{p})}{n_h} - \frac{N_h^2}{N^2}\frac{n_h}{N_h-1}\frac{\hat{p}(1-\hat{p})}{n_h}$

$V(\hat{p}_h) = \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)}\frac{1}{n_h} - \frac{N_h^2\hat{p}(1-\hat{p})}{N^2(N_h-1)}$

$V(\hat{p}_h) = \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)}(1 - \frac{1}{1\cdot 2} - \frac{1}{2\cdot 3} - \cdots - \frac{1}{n_h(n_h-1)}) - \frac{N_h^2\hat{p}(1-\hat{p})}{N^2(N_h-1)}$

$V(\hat{p}_h) = \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)} - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 1\cdot 2} - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 2\cdot 3} - \cdots - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot n_h(n_h-1)} - \frac{N_h^2\hat{p}(1-\hat{p})}{N^2(N_h-1)}$

$V(\hat{p}_h) = \frac{(N_h^3-N_h^2)\hat{p}(1-\hat{p})}{N^2(N_h-1)} - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 1\cdot 2} - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 2\cdot 3} - \cdots - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot n_h(n_h-1)}$

$V(\hat{p}_h) = \frac{N_h^2(N_h-1)\hat{p}(1-\hat{p})}{N^2(N_h-1)} - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 1\cdot 2} - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 2\cdot 3} - \cdots - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot n_h(n_h-1)}$

Decomposition of $V(\hat{p}_{str})$

$V(\hat{p}_{str}) = \sum_{h=1}^{H} \frac{N_h^2(N_h-1)\hat{p}(1-\hat{p})}{N^2(N_h-1)}$

$-\frac{N_1^3\hat{p}(1-\hat{p})}{N^2(N_1-1)\cdot 1\cdot 2} - \frac{N_1^3\hat{p}(1-\hat{p})}{N^2(N_1-1)\cdot 2\cdot 3} - \ldots - \frac{N_1^3\hat{p}(1-\hat{p})}{N^2(N_1-1)n_h(n_h-1)}$

$\ldots$

$-\frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 1\cdot 2} - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)\cdot 2\cdot 3} - \ldots - \frac{N_h^3\hat{p}(1-\hat{p})}{N^2(N_h-1)n_h(n_h-1)}$

$\ldots$

$-\frac{N_H^3\hat{p}(1-\hat{p})}{N^2(N_H-1)\cdot 1\cdot 2} - \frac{N_H^3\hat{p}(1-\hat{p})}{N^2(N_H-1)\cdot 3\cdot 3} - \ldots - \frac{N_H^3\hat{p}(1-\hat{p})}{N^2(N_H-1)n_h(n_h-1)}$

For a desired bound on V_0 on the sampling variance $V(\hat{p}_{str})$, we may find an optimal allocation using the following algorithm:

1) Assign, for each stratum, 1 unit to be selected for the sample.

2) Fill in the following table and number these values starting from 1, in decreasing order. We assume $p_h = 0.5$ because that is where the variance reaches its global maximum.

$\frac{\frac{1}{4}N_1^3}{N^2(N_1-1)\cdot 1\cdot 2}$	$\frac{\frac{1}{4}N_1^3}{N^2(N_1-1)\cdot 2\cdot 3}$	$\frac{\frac{1}{4}N_1^3}{N^2(N_1-1)\cdot 3\cdot 4}$	$\ldots$
$\frac{\frac{1}{4}N_2^3}{N^2(N_2-1)\cdot 1\cdot 2}$	$\frac{\frac{1}{4}N_2^3}{N^2(N_2-1)\cdot 2\cdot 3}$	$\frac{\frac{1}{4}N_2^3}{N^2(N_2-1)\cdot 3\cdot 4}$	$\ldots$
.	.	.	$\ldots$
.	.	.	$\ldots$
.	.	.	$\ldots$
$\frac{\frac{1}{4}N_H^3}{N^2(N_H-1)\cdot 1\cdot 2}$	$\frac{\frac{1}{4}N_H^3}{N^2(N_H-1)\cdot 2\cdot 3}$	$\frac{\frac{1}{4}N_H^3}{N^2(N_H-1)\cdot 3\cdot 4}$	$\ldots$

3) Since the initial allocation is $(n_{11}, n_{21}, ..., n_{H1}) = (1, 1, ..., 1)$, compute $V(\hat{p}_{str}|n_{11} = 1, n_{21} = 1, ..., n_{H1} = 1) = \frac{1}{N^2}\sum_{h=1}^{H}((N_h^2 - N_h)S_h^2)$

4) Pick value (1) from the table and increase the associated stratum's sample size by 1, o that the updated allocation is $(n_{12}, n_{22}, ..., n_{H2})$, where exactly one of the n_{h2}'s is equal to 2 and the rest are equal to 1. Then, compute $V(\hat{p}_{str}|n_{12}, ..., n_{H2} = V(\hat{p}_{str}|n_{11}, ..., n_{H1}) - \frac{1}{N^2}$ where "(1)" represents the largest value from the table. If $V(\hat{p}_{str}|N_{12}, ..., n_{H2} \leq V_0$, then stop with $n_1 = n_{12}, ..., N_H = N_{H2}$. Otherwise, go to step 5.

5) Pick value (2) from the table and increase the associated stratum's sample size by 1, so that the updated allocation is $(n_{13}, ..., n_{H3})$. Then compute $V(\hat{p}_{str}|n_{13}, ..., n_{H3}) = V(\hat{p}_{str}|n_{12}, ..., n_{H2} - \frac{(2)}{N^2}$, where "(2)" represents the second value from the table. If $V(\hat{p}_{str}|n_{13}, ..., N_H = n_{H3}$. Otherwise, continue until step j, where $V(\hat{p}_{str}|n_{1j}, ..., n_{Hj}) \leq V_0$. The final allocation is $n_{1j}, ..., n_{Hj})$ and $n = n_{1j} + \cdots + n_{Hj}$.

```r
#
strata <- sampling_frame %>%
  count(stratum = quadrant) %>%
  rename(Nh = n) %>%
  mutate(N =sum(Nh),
         s_squared_h = 0.5 * (1 - 0.5))

kable(strata)
```

stratum	Nh	N	s_squared_h
NE	74949	348094	0.25
NW	189695	348094	0.25
SE	68644	348094	0.25
SW	14806	348094	0.25

```r
# Let the initial allocation be (n_11, n_21, n_31, n_41) = (1, 1, 1, 1)
# (3) and (6)
starting_variance <- strata %>%
  mutate(strata_variance = (Nh / N) ^ 2 * ((Nh - 1) / Nh) * (s_squared_h / 1))

kable(starting_variance)
```

stratum	Nh	N	s_squared_h	strata_variance
NE	74949	348094	0.25	0.0115897
NW	189695	348094	0.25	0.0742432
SE	68644	348094	0.25	0.0097218
SW	14806	348094	0.25	0.0004523

```r
starting_variance <- starting_variance %>%
  summarize(V = sum(strata_variance)) %>%
  pull()

starting_variance
```

```
## [1] 0.09600692
```

```r
# create a table of priority values
# (4) and (5)

n_strata <-tibble(n = c(1:500, 1:500, 1:500, 1:500),
                  stratum = c(rep("NE", 500), rep("NW", 500), rep("SE", 500), rep("SW", 500))) %>%
  left_join(strata, by = "stratum")

# step 2
priority_values <- n_strata %>%
  group_by(stratum) %>%
  mutate(priority_value = (0.25 * Nh * (Nh - 1)) / (N ^ 2 * n * lag(n))) %>%
```

```r
  ungroup() %>%
  arrange(desc(priority_value))

kable(head(priority_values, n = 10))
```

n	stratum	Nh	N	s_squared_h	priority_value
2	NW	189695	348094	0.25	0.0371216
3	NW	189695	348094	0.25	0.0123739
4	NW	189695	348094	0.25	0.0061869
2	NE	74949	348094	0.25	0.0057949
2	SE	68644	348094	0.25	0.0048609
5	NW	189695	348094	0.25	0.0037122
6	NW	189695	348094	0.25	0.0024748
3	NE	74949	348094	0.25	0.0019316
7	NW	189695	348094	0.25	0.0017677
3	SE	68644	348094	0.25	0.0016203

```r
# (7)
priority_values <- priority_values %>%
  mutate(agg_priority_value = cumsum(priority_value)) %>%
  mutate(marginal_variance = starting_variance - agg_priority_value) %>%
  mutate(marginal_sd = sqrt(marginal_variance))

kable(head(select(priority_values, -N), n = 50), align = "l")
```

n	stratum	Nh	s_squared_h	priority_value	agg_priority_value	marginal_variance	marginal_sd
2	NW	189695	0.25	0.0371216	0.0371216	0.0588853	0.2426630
3	NW	189695	0.25	0.0123739	0.0494954	0.0465115	0.2156652
4	NW	189695	0.25	0.0061869	0.0556824	0.0403245	0.2008097
2	NE	74949	0.25	0.0057949	0.0614772	0.0345297	0.1858217
2	SE	68644	0.25	0.0048609	0.0663381	0.0296688	0.1722463
5	NW	189695	0.25	0.0037122	0.0700503	0.0259566	0.1611107
6	NW	189695	0.25	0.0024748	0.0725250	0.0234819	0.1532380
3	NE	74949	0.25	0.0019316	0.0744567	0.0215503	0.1468000
7	NW	189695	0.25	0.0017677	0.0762244	0.0197826	0.1406505
3	SE	68644	0.25	0.0016203	0.0778447	0.0181623	0.1347674
8	NW	189695	0.25	0.0013258	0.0791704	0.0168365	0.1297555
9	NW	189695	0.25	0.0010312	0.0802016	0.0158053	0.1257193
4	NE	74949	0.25	0.0009658	0.0811674	0.0148395	0.1218176
10	NW	189695	0.25	0.0008249	0.0819923	0.0140146	0.1183833
4	SE	68644	0.25	0.0008101	0.0828025	0.0132045	0.1149106
11	NW	189695	0.25	0.0006749	0.0834774	0.0125295	0.1119353
5	NE	74949	0.25	0.0005795	0.0840569	0.0119500	0.1093162
12	NW	189695	0.25	0.0005624	0.0846193	0.0113876	0.1067126
5	SE	68644	0.25	0.0004861	0.0851054	0.0109015	0.1044102
13	NW	189695	0.25	0.0004759	0.0855813	0.0104256	0.1021057
14	NW	189695	0.25	0.0004079	0.0859893	0.0100176	0.1000882
6	NE	74949	0.25	0.0003863	0.0863756	0.0096313	0.0981393
15	NW	189695	0.25	0.0003535	0.0867291	0.0092778	0.0963213

n	stratum	Nh	s_squared_h	priority_value	agg_priority_value	marginal_variance	marginal_sd
6	SE	68644	0.25	0.0003241	0.0870532	0.0089537	0.0946241
16	NW	189695	0.25	0.0003093	0.0873625	0.0086444	0.0929751
7	NE	74949	0.25	0.0002759	0.0876385	0.0083684	0.0914791
17	NW	189695	0.25	0.0002730	0.0879114	0.0080955	0.0899749
18	NW	189695	0.25	0.0002426	0.0881541	0.0078529	0.0886163
7	SE	68644	0.25	0.0002315	0.0883855	0.0076214	0.0873005
2	SW	14806	0.25	0.0002261	0.0886117	0.0073953	0.0859956
19	NW	189695	0.25	0.0002171	0.0888287	0.0071782	0.0847241
8	NE	74949	0.25	0.0002070	0.0890357	0.0069712	0.0834938
20	NW	189695	0.25	0.0001954	0.0892311	0.0067758	0.0823154
21	NW	189695	0.25	0.0001768	0.0894079	0.0065991	0.0812346
8	SE	68644	0.25	0.0001736	0.0895815	0.0064255	0.0801590
9	NE	74949	0.25	0.0001610	0.0897424	0.0062645	0.0791485
22	NW	189695	0.25	0.0001607	0.0899031	0.0061038	0.0781268
23	NW	189695	0.25	0.0001467	0.0900499	0.0059571	0.0771820
9	SE	68644	0.25	0.0001350	0.0901849	0.0058220	0.0763023
24	NW	189695	0.25	0.0001345	0.0903194	0.0056875	0.0754158
10	NE	74949	0.25	0.0001288	0.0904481	0.0055588	0.0745571
25	NW	189695	0.25	0.0001237	0.0905719	0.0054350	0.0737226
26	NW	189695	0.25	0.0001142	0.0906861	0.0053208	0.0729439
10	SE	68644	0.25	0.0001080	0.0907941	0.0052128	0.0721996
27	NW	189695	0.25	0.0001058	0.0908999	0.0051070	0.0714635
11	NE	74949	0.25	0.0001054	0.0910052	0.0050017	0.0707225
28	NW	189695	0.25	0.0000982	0.0911035	0.0049035	0.0700247
29	NW	189695	0.25	0.0000914	0.0911949	0.0048120	0.0693688
11	SE	68644	0.25	0.0000884	0.0912833	0.0047237	0.0687288
12	NE	74949	0.25	0.0000878	0.0913711	0.0046358	0.0680871

```
rm(pilot_sample, strata, n_strata)
```