Fyrra R Verkefni

Thor Sanchez (ths281)

2024-02-04

```
library(tidyverse)
library(lubridate)
library(ggplot2)
library(knitr)
library(dplyr)
```

Hluti 1

#a)

```
#Lets get the dataframe
ths <- read_csv2("https://ahj.hi.is/kaupskra.csv",
                         locale = locale(encoding = "ISO8859-1"))
#To view the dataframe
glimpse(ths)
```

```
## Rows: 169,636
## Columns: 22
## $ faerslunumer         <dbl> 569113, 558760, 566833, 566833, 628860, 579617, 5…
## $ emnr                 <dbl> 437, 436, 411, 411, 441, 411, 411, 411, 411, 411,…
## $ skjalanumer          <chr> "S-003590/2012", "X-000916/2011", "S-003191/2012"…
## $ fastnum              <dbl> 2064400, 2074439, 2264635, 2264636, 2264633, 2023…
## $ heimlilisfang        <chr> "Nýbýlavegur 12", "Drangahraun 10", "Dugguvogur 9…
## $ postnr               <dbl> 200, 220, 104, 104, 104, 104, 104, 104, 104, 105,…
## $ heinum               <dbl> 1022800, 1118718, 1003941, 1003941, 1003941, 1003…
## $ svfn                 <chr> "1000", "1400", "0000", "0000", "0000", "0000", "…
## $ sveitarfelag         <chr> "Kópavogsbær", "Hafnarfjarðarkaupstaður", "Reykja…
## $ utgdag               <dttm> 2012-07-30, 2011-02-28, 2012-04-16, 2012-04-16, …
## $ thinglystdags        <dttm> 2012-08-01 08:27:51, 2011-03-02 09:12:33, 2012-0…
## $ kaupverd             <dbl> 87000, 36000, 31000, 31000, 23500, 33500, 31000, …
## $ fasteignamat         <dbl> 70850, 40790, 4679, 5516, 13200, 27100, 3975, 711…
## $ byggar               <dbl> 1985, 1983, 1962, 1962, 1962, 1962, 1962, 1962, 1…
## $ epilog               <chr> "010101", "010102", "010301", "010302", "010201",…
## $ einflm               <dbl> 780.4, 400.0, 310.2, 310.2, 71.4, 325.0, 310.2, 3…
## $ lod_flm              <dbl> 1105, 3000, 565, 565, 565, 565, 565, 565, 565, 57…
## $ lod_flmein           <chr> "m²", "m²", "m²", "m²", "m²", "m²", "m²", "m²", "…
## $ tegund               <chr> "Atvinnuhusnaedi", "Atvinnuhusnaedi", "Atvinnuhus…
## $ fullbuid             <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1…
## $ onothaefur_samningur <dbl> 0, 0, 1, 1, 0, 1, 1, 1, 1, 1, 1, 0, 0, 0, 0, 0, 0…
## $ ...22                <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
```

#b)

```
# Filtering for completed properties that are "Fjolbyli" or "Serbyli" and have usab
le contracts
ths_filtered <- ths %>%
  filter(fullbuid == TRUE,
         tegund %in% c("Fjolbyli", "Serbyli"),
         onothaefur_samningur == TRUE)

# Overwriting the original dataframe with the filtered properties
ths <- ths_filtered

# To view the structure of the filtered dataframe
glimpse(ths)
```

```
## Rows: 27,683
## Columns: 22
## $ faerslunumer         <dbl> 559598, 616011, 607609, 536642, 512200, 551029, 5…
## $ emnr                 <dbl> 411, 441, 441, 411, 411, 434, 434, 441, 411, 437,…
## $ skjalanumer          <chr> "U-002796/2011", "E-000905/2017", "I-005204/2016"…
## $ fastnum              <dbl> 2019491, 2019515, 2019491, 2019516, 2019491, 2087…
## $ heimlilisfang        <chr> "Engjateigur 17-19", "Engjateigur 17-19", "Engjat…
## $ postnr               <dbl> 105, 105, 105, 105, 105, 230, 230, 107, 107, 201,…
## $ heinum               <dbl> 1004483, 1004483, 1004483, 1004483, 1004483, 1035…
## $ svfn                 <chr> "0000", "0000", "0000", "0000", "0000", "2000", "…
## $ sveitarfelag         <chr> "Reykjavíkurborg", "Reykjavíkurborg", "Reykjavíku…
## $ utgdag               <dttm> 2011-04-05, 2017-01-26, 2016-05-24, 2007-07-24, …
## $ thinglystdags        <dttm> 2011-04-15 14:39:06, 2017-02-06 12:05:24, 2016-0…
## $ kaupverd             <dbl> 37000, 20500, 84000, 30790, 42600, 24900, 25000, …
## $ fasteignamat         <dbl> 5390, 35750, 7035, 21550, 4544, 22880, 19550, 248…
## $ byggar               <dbl> 1992, 1992, 1992, 1992, 1992, 1960, 1960, 1969, 1…
## $ epilog               <chr> "010002", "010206", "010002", "010207", "010002",…
## $ einflm               <dbl> 307.2, 109.9, 307.2, 109.9, 214.0, 179.9, 179.9, …
## $ lod_flm              <dbl> 5702.0, 5702.0, 5702.0, 5702.0, 5702.0, 442.0, 44…
## $ lod_flmein           <chr> "m²", "m²", "m²", "m²", "m²", "m²", "m²", "m²", "…
## $ tegund               <chr> "Fjolbyli", "Fjolbyli", "Fjolbyli", "Fjolbyli", "…
## $ fullbuid             <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1…
## $ onothaefur_samningur <dbl> 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1…
## $ ...22                <lgl> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, N…
```

#c)

```
# Making new variable for price per square meter and formatting for display
#adding one decimal to fermetraverd
ths <- ths %>%
  mutate(fermetraverd = kaupverd / einflm) %>%
  mutate(fermetraverd_display = sprintf("%.1f", fermetraverd))

# Displaying the first few rows to verify the calculations of 'fermetraverd'
head(ths[, c("kaupverd", "einflm", "fermetraverd_display")])
```

```
## # A tibble: 6 × 3
##   kaupverd einflm fermetraverd_display
##      <dbl>  <dbl> <chr>
## 1    37000   307. 120.4
## 2    20500   110. 186.5
## 3    84000   307. 273.4
## 4    30790   110. 280.2
## 5    42600   214  199.1
## 6    24900   180. 138.4
```

#d)

```
# Making new variable for the year the property was sold
ths <- ths %>%
  mutate(ar = year(utgdag))
# To view thefirst few rows of the 'ar' column
head(select(ths, ar))
```

```
## # A tibble: 6 × 1
##      ar
##   <dbl>
## 1  2011
## 2  2017
## 3  2016
## 4  2007
## 5  2006
## 6  2009
```

#e)

```
# this is to find postal codes that have 200+ sérbýli
serbyli_df <- ths %>%
  filter(tegund == "Serbyli")

postal_code_counts <- serbyli_df %>%
  group_by(postnr) %>%
  summarise(count = n()) %>%
  filter(count >= 200)
postal_code_counts
```

```
## # A tibble: 12 × 2
##    postnr count
##     <dbl> <int>
##  1    112   226
##  2    200   272
##  3    210   357
##  4    220   270
##  5    221   209
##  6    230   223
##  7    260   280
##  8    270   290
##  9    300   248
## 10    600   258
## 11    800   429
## 12    810   212
```

```
# After finding three postal codes that meet the criteria, we intend to modify the
dataframe to only include properties from these postal codes (112, 200, and 210).

selected_postal_codes <- c(112, 200, 210)
ths_filtered <- ths %>%
  filter(postnr %in% selected_postal_codes)
# We need to overwrite ths (from part b) in order to use this dataframe in the rest
of project

ths <- ths_filtered
# Displaying the unique postal codes to confirm the code works
unique(ths$postnr)
```

```
## [1] 112 210 200
```

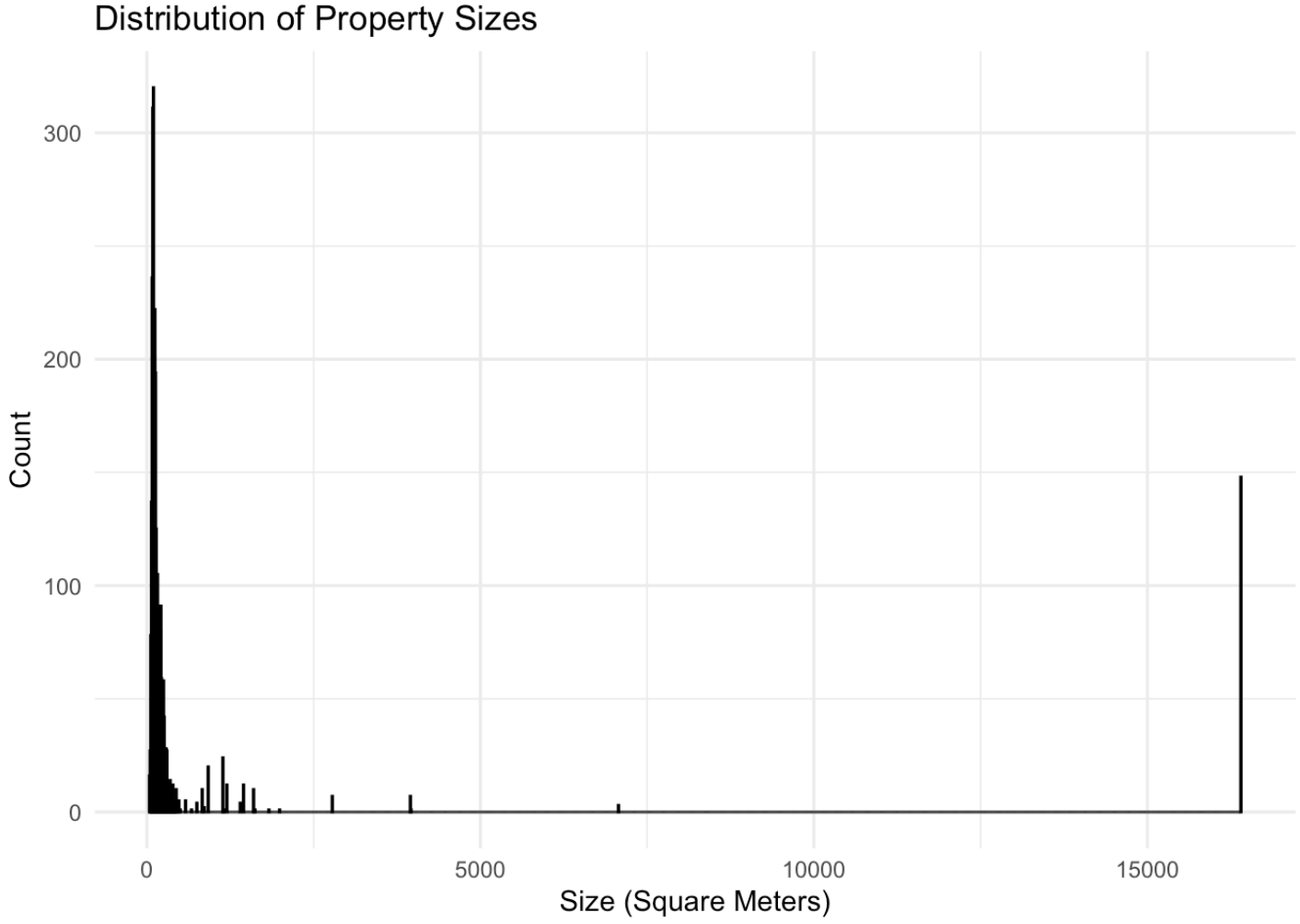
#f)

```
#Change postal codes to CHAR with length 3,
#because we know thats the criteria for a postal code in Iceland
ths$postnr <- as.character(ths$postnr)
ths$postnr <- sprintf("%03d", as.integer(ths$postnr))
```

Hluti 2

#g)

```
#Many properties are registered with an incredibly high square meter number, hence significant outliers.
ggplot(ths, aes(x = einflm)) +
  geom_histogram(binwidth = 10, fill = "blue", color = "black") +
  theme_minimal() +
  labs(title = "Distribution of Property Sizes",
       x = "Size (Square Meters)",
       y = "Count")
```

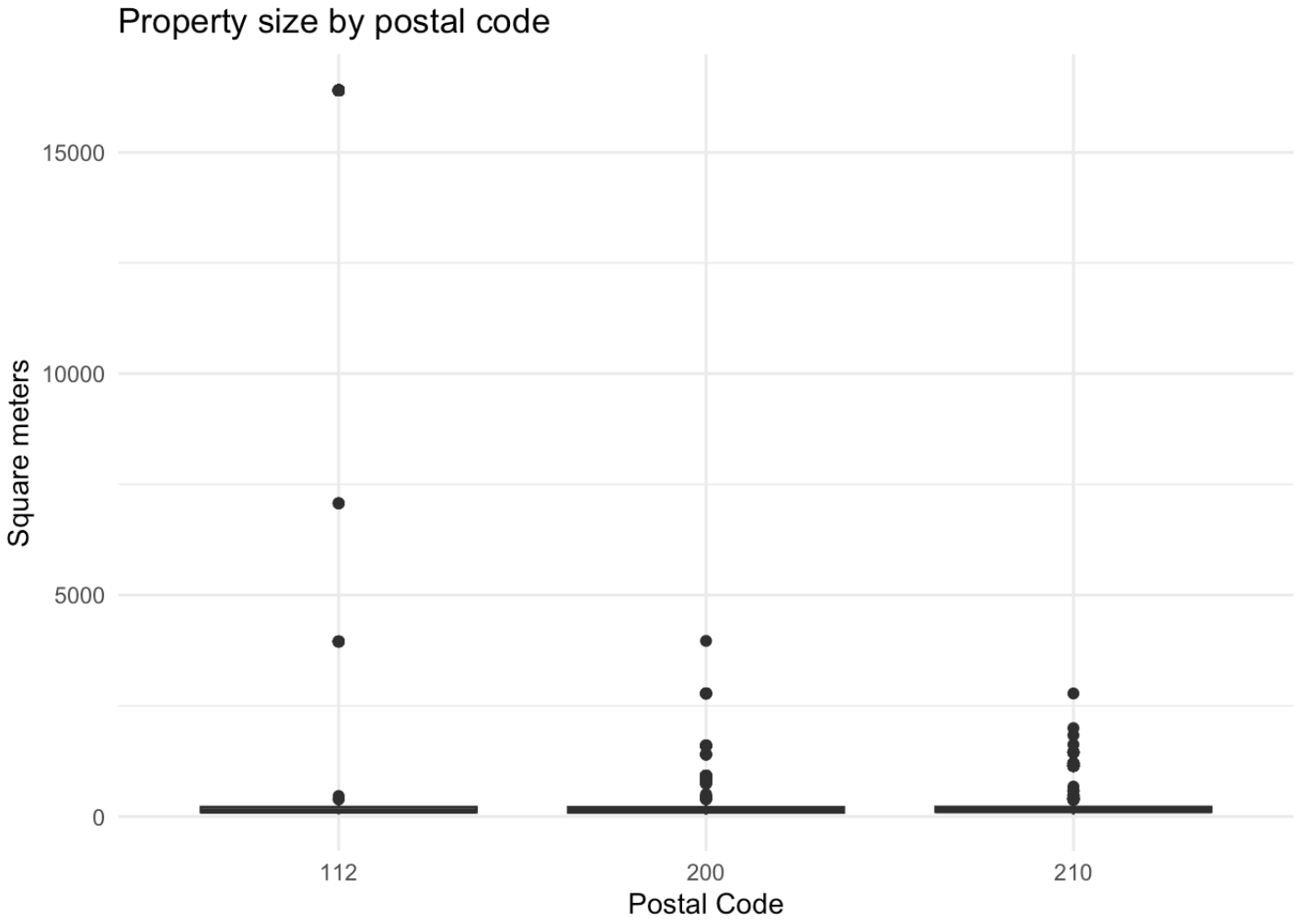


#h)

```
# Bar chart showing the number of properties by postal codes and types of propertie
s (sérbýli eða fjölbýli).
ggplot(ths, aes(x = postnr, fill = tegund)) +
  geom_bar(position = "dodge") +
  theme_minimal() +
  labs(title = "Number of Properties by Postal Codes and Types of Properties",
       x = "Postal Code",
       y = "Number of Properties",
       fill = "Type of Property")
```

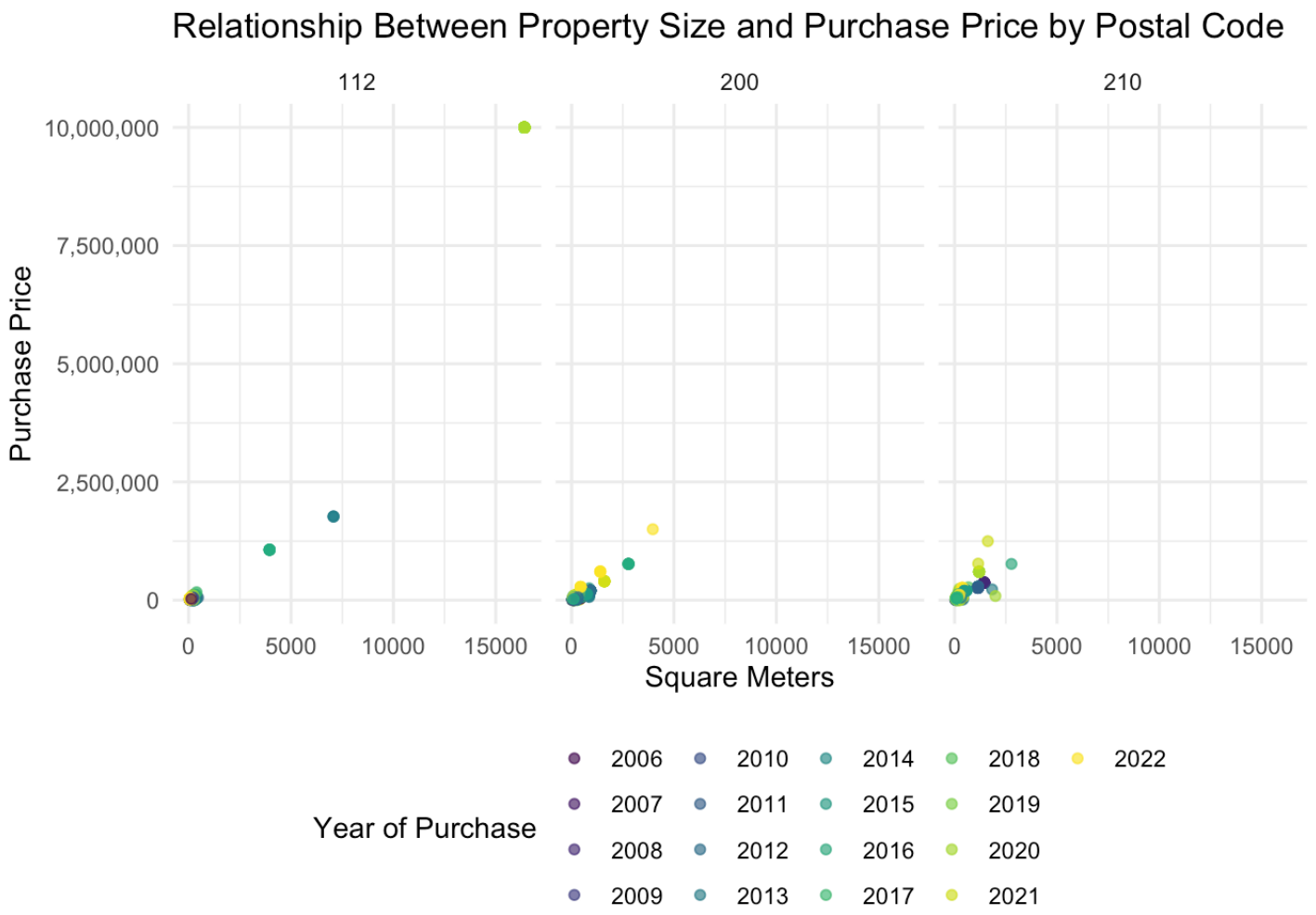
#i) Again, there are outliers, and since these outliers meet all the criteria for this project, I have decided not to remove them from the data.

```
ggplot(ths, aes(x = postnr, y = einflm)) +
  geom_boxplot() +
  theme_minimal() +
  labs(title = "Property size by postal code",
       x = "Postal Code",
       y = "Square meters")
```



#j)

```
ggplot(ths, aes(x = einflm, y = kaupverd, color = as.factor(ar))) +
  # Using alpha for better visualization if points overlap
  geom_point(alpha = 0.7) +
  facet_grid(~ postnr) +
  scale_y_continuous(labels = scales::comma) +
  scale_color_viridis_d() +
  theme_minimal() +
  # Moves the legend to the bottom
  theme(legend.position = "bottom") +
  labs(title = "Relationship Between Property Size and Purchase Price by Postal Cod
e",
       x = "Square Meters",
       y = "Purchase Price",
       color = "Year of Purchase")
```



Hluti 3

#k)

```
# Create table
property_type_table <- table(ths$postnr, ths$tegund)

#Create table with kable()
kable(property_type_table, caption = "Number of properties in the three postal code
s by type of property")
```

Number of properties in the three postal codes by type of property

	Fjolbyli	Serbyli
112	857	226
200	799	272
210	751	357

#l)

```
# Make a table to count properties by postal code and type
property_type_table <- table(ths$postnr, ths$tegund)

# Make percentages for each postal code
property_type_proportions <- prop.table(property_type_table, margin = 1)

# Make the percentages easier to read by rounding them
property_type_proportions_percentage <- round(property_type_proportions * 100, 2)

# Create table with kable()
kable(property_type_proportions_percentage, caption = "Proportion of Property Types
by Postal Code (%)")
```

Proportion of Property Types by Postal Code (%)

	Fjolbyli	Serbyli
112	79.13	20.87
200	74.60	25.40
210	67.78	32.22

#m)

```
# To remove rows with missing data to clean up the dataset
stats_table <- ths %>%
  filter(!is.na(ar), !is.na(postnr), !is.na(tegund), !is.na(fermetraverd)) %>%
  # Group data by year, postal code, and type of property
  group_by(ar, postnr, tegund) %>%
  # Calculate average, standard deviation, median, and count for square meter price
  summarise(
    meðaltal = mean(fermetraverd, na.rm = TRUE),
    staðalfrávik = sd(fermetraverd, na.rm = TRUE),
    miðgildi = median(fermetraverd, na.rm = TRUE),
    fjöldi = n()
  ) %>%
  #To stop grouping
  ungroup() %>%
  #Create table with kable
  kable(caption = "Statistics of Price per Square Meter by Year, Postal Code, and P
roperty Type")

# Display it
stats_table
```

Statistics of Price per Square Meter by Year, Postal Code, and Property Type

ar	postnr	tegund	meðaltal	staðalfrávik	miðgildi	fjöldi

2006	112	Fjolbyli	198.7213	41.05096	211.9796	36
2006	112	Serbyli	228.5211	29.95751	222.7171	11
2006	200	Fjolbyli	186.3903	55.38564	203.1961	38
2006	200	Serbyli	206.9934	34.13388	188.4236	3
2006	210	Fjolbyli	250.2646	57.39220	266.3916	20
2006	210	Serbyli	268.5285	45.11014	253.8945	12
2007	112	Fjolbyli	237.4549	37.03878	242.2611	75
2007	112	Serbyli	249.4249	43.04924	251.5484	11
2007	200	Fjolbyli	202.0475	89.63742	207.8978	37
2007	200	Serbyli	167.1104	54.60250	175.1388	15
2007	210	Fjolbyli	297.7378	49.46205	300.0000	109
2007	210	Serbyli	265.0202	99.72331	255.4347	12
2008	112	Fjolbyli	250.4172	68.97803	254.2182	60
2008	112	Serbyli	247.8259	64.51969	264.7711	12
2008	200	Fjolbyli	232.9778	61.12364	239.6804	27
2008	200	Serbyli	231.5709	81.17203	223.7342	16
2008	210	Fjolbyli	289.4582	45.22889	286.7384	57
2008	210	Serbyli	281.3082	41.79387	278.5205	21
2009	112	Fjolbyli	228.7167	53.50292	236.3636	61
2009	112	Serbyli	248.6749	35.27622	242.9476	19
2009	200	Fjolbyli	220.5430	46.42727	216.1828	53
2009	200	Serbyli	223.4536	66.46024	212.8631	20
2009	210	Fjolbyli	261.4470	42.81427	252.6030	36
2009	210	Serbyli	269.7821	90.79421	262.9969	35
2010	112	Fjolbyli	215.8609	59.52756	219.4357	79
2010	112	Serbyli	216.1243	41.76213	220.4206	33
2010	200	Fjolbyli	217.7880	45.44253	210.2297	55
2010	200	Serbyli	201.3879	69.64169	205.4583	25
2010	210	Fjolbyli	247.3200	34.78022	245.5845	25

2010	210	Serbyli	227.6318	71.31984	232.9451	27
2011	112	Fjolbyli	203.3754	49.53651	210.0227	56
2011	112	Serbyli	232.4645	42.15203	230.5203	18
2011	200	Fjolbyli	200.7176	46.52047	208.7804	50
2011	200	Serbyli	182.9730	49.02260	172.7019	23
2011	210	Fjolbyli	247.2728	43.37410	243.1762	51
2011	210	Serbyli	227.2763	75.57196	210.4164	28
2012	112	Fjolbyli	223.8468	30.85860	227.7626	50
2012	112	Serbyli	217.6548	56.68364	227.1620	17
2012	200	Fjolbyli	227.5815	58.49346	217.8412	75
2012	200	Serbyli	200.7804	65.47617	186.4914	26
2012	210	Fjolbyli	243.1212	41.90047	247.3947	44
2012	210	Serbyli	223.2854	58.74734	229.6459	26
2013	112	Fjolbyli	238.0129	49.94886	250.2172	37
2013	112	Serbyli	224.7982	70.05496	243.2297	11
2013	200	Fjolbyli	227.5435	87.73870	218.3947	74
2013	200	Serbyli	226.6020	75.23720	208.9985	18
2013	210	Fjolbyli	274.3048	59.05439	275.0911	14
2013	210	Serbyli	263.0151	72.50618	279.9133	24
2014	112	Fjolbyli	256.4921	55.13545	259.5506	43
2014	112	Serbyli	247.4165	74.25226	257.6750	20
2014	200	Fjolbyli	232.7951	66.89366	237.7415	51
2014	200	Serbyli	194.0921	64.76733	190.3614	31
2014	210	Fjolbyli	292.6544	70.24265	302.3772	30
2014	210	Serbyli	254.4090	77.55526	260.4933	26
2015	112	Fjolbyli	256.7502	70.79601	259.8985	36
2015	112	Serbyli	243.3563	79.66747	267.0188	13
2015	200	Fjolbyli	290.1868	110.73879	273.9365	56
2015	200	Serbyli	200.2462	73.72029	191.8841	19

2015	210	Fjolbyli	346.2046	59.97571	351.4543	47
2015	210	Serbyli	260.9321	81.67332	274.8599	24
2016	112	Fjolbyli	271.5360	110.29636	299.2278	53
2016	112	Serbyli	272.3814	50.26129	275.7560	15
2016	200	Fjolbyli	345.2988	103.81444	363.7821	60
2016	200	Serbyli	196.4257	75.52249	192.2621	14
2016	210	Fjolbyli	383.2041	80.33558	391.4405	45
2016	210	Serbyli	278.0918	74.05150	269.7369	30
2017	112	Fjolbyli	365.3000	111.07456	393.7359	30
2017	112	Serbyli	324.7816	110.05238	366.8708	8
2017	200	Fjolbyli	372.3612	171.71574	360.3510	46
2017	200	Serbyli	279.0051	144.98121	289.8898	14
2017	210	Fjolbyli	430.0508	136.60157	433.0709	45
2017	210	Serbyli	317.5342	96.30977	296.2534	21
2018	112	Fjolbyli	318.7801	127.08964	359.0734	17
2018	112	Serbyli	354.1981	94.42876	364.6646	15
2018	200	Fjolbyli	348.9537	147.68728	385.3408	32
2018	200	Serbyli	262.8782	108.05034	268.0052	18
2018	210	Fjolbyli	470.7774	79.93595	477.3270	49
2018	210	Serbyli	365.6372	108.02840	398.9761	14
2019	112	Fjolbyli	334.0661	135.71280	398.1576	21
2019	112	Serbyli	291.6703	109.11815	239.5470	5
2019	200	Fjolbyli	435.7950	133.99947	464.8239	52
2019	200	Serbyli	340.9529	99.96357	344.8437	7
2019	210	Fjolbyli	513.0802	111.89248	524.2390	61
2019	210	Serbyli	424.0773	133.24931	419.4347	14
2020	112	Fjolbyli	584.2775	87.28519	609.6966	165
2020	112	Serbyli	354.4451	126.63627	359.5464	12
2020	200	Fjolbyli	443.2431	165.42967	450.0000	34

2020	200	Serbyli	326.3086	209.65770	379.6740	11
2020	210	Fjolbyli	508.3886	144.90278	508.6207	67
2020	210	Serbyli	423.5357	242.04395	387.5453	22
2021	112	Fjolbyli	320.7080	139.65309	277.7778	11
2021	112	Serbyli	383.3451	31.35947	380.8605	4
2021	200	Fjolbyli	397.9937	193.79505	409.1228	34
2021	200	Serbyli	284.9949	156.80314	298.3625	6
2021	210	Fjolbyli	400.6294	218.32809	344.2091	32
2021	210	Serbyli	428.3882	212.25426	470.2784	15
2022	112	Fjolbyli	526.9018	174.56901	638.5070	27
2022	112	Serbyli	312.5092	350.72541	312.5092	2
2022	200	Fjolbyli	494.3135	137.41135	526.9187	25
2022	200	Serbyli	336.5107	138.11639	345.9500	6
2022	210	Fjolbyli	558.0121	232.62941	563.5148	19
2022	210	Serbyli	579.1528	180.74001	693.1638	6

#n)

```
prob_exactly_one <- dbinom(1, size = 3, prob = 0.20)
```

```
prob_more_than_one <- dbinom(2, size = 3, prob = 0.20) + dbinom(3, size = 3, prob = 0.20)
```

The probability of exactly one detached house is 0.384 and the probability of more than one detached house is 0.104.

#o)

```
# i)
# mean size of houses
mean_size <- 170
# standard deviation of house sizes
sd_size <- 20

# Probability of a house being larger than 180 square meters
prob_larger_than_180 <- 1 - pnorm(180, mean = mean_size, sd = sd_size)
```

```
# ii)
size_for_top_5_percent <- qnorm(0.95, mean = mean_size, sd = sd_size)
```

In the neighborhood we're considering, where detached houses follow a normal distribution with a mean size of 170 square meters and a standard deviation of 20 square meters:

i. The probability that a randomly selected detached house is larger than 180 square meters is 30.85%.

ii. To be among the top 5% largest detached houses in the neighborhood, Jón would need to build a house that is at least 202.9 square meters.