Mini Data-Analysis Deliverable 1

Shannon Edie

08/10/2021

```
# install.packages("devtools")
# devtools::install_github("UBC-MDS/datateachr")

library(datateachr)
library(tidyverse)
```

```
## Warning: package 'tidyverse' was built under R version 4.1.1
```

```
## -- Attaching packages --------------------------------------- tidyverse 1.3.1 --
```

```
## v ggplot2 3.3.5     v purrr   0.3.4
## v tibble  3.1.2     v dplyr   1.0.7
## v tidyr   1.1.3     v stringr 1.4.0
## v readr   1.4.0     v forcats 0.5.1
```

```
## -- Conflicts ------------------------------------------ tidyverse_conflicts() --
## x dplyr::filter() masks stats::filter()
## x dplyr::lag()    masks stats::lag()
```

```
library(knitr)
```

Task 1: Choose your favorite dataset (10 points)

The `datateachr` package by Hayley Boyce and Jordan Bourak currently composed of 7 semi-tidy datasets for educational purposes. Here is a brief description of each dataset:

- *apt_buildings*: Acquired courtesy of The City of Toronto's Open Data Portal. It currently has 3455 rows and 37 columns.
- *building_permits*: Acquired courtesy of The City of Vancouver's Open Data Portal. It currently has 20680 rows and 14 columns.
- *cancer_sample*: Acquired courtesy of UCI Machine Learning Repository. It currently has 569 rows and 32 columns.
- *flow_sample*: Acquired courtesy of The Government of Canada's Historical Hydrometric Database. It currently has 218 rows and 7 columns.
- *parking_meters*: Acquired courtesy of The City of Vancouver's Open Data Portal. It currently has 10032 rows and 22 columns.

- *steam_games*: Acquired courtesy of Kaggle. It currently has 40833 rows and 21 columns.
- *vancouver_trees*: Acquired courtesy of The City of Vancouver's Open Data Portal. It currently has 146611 rows and 20 columns.

Things to keep in mind

- We hope that this project will serve as practice for carrying our your own *independent* data analysis. Remember to comment your code, be explicit about what you are doing, and write notes in this markdown document when you feel that context is required. As you advance in the project, prompts and hints to do this will be diminished - it'll be up to you!
- Before choosing a dataset, you should always keep in mind **your goal**, or in other ways, *what you wish to achieve with this data*. This mini data-analysis project focuses on *data wrangling*, *tidying*, and *visualization*. In short, it's a way for you to get your feet wet with exploring data on your own.

And that is exactly the first thing that you will do!

1.1 Out of the 7 datasets available in the `datateachr` package, choose 4 that appeal to you based on their description. Write your choices below: 1: *apt_buildings*: Acquired courtesy of The City of Toronto's Open Data Portal. It currently has 3455 rows and 37 columns.

2: *building_permits*: Acquired courtesy of The City of Vancouver's Open Data Portal. It currently has 20680 rows and 14 columns.

3: *cancer_sample*: Acquired courtesy of UCI Machine Learning Repository. It currently has 569 rows and 32 columns.

4: *vancouver_trees*: Acquired courtesy of The City of Vancouver's Open Data Portal. It currently has 146611 rows and 20 columns.

1.2 One way to narrowing down your selection is to *explore* the datasets. Use your knowledge of dplyr to find out at least *3* attributes about each of these datasets (an attribute is something such as number of rows, variables, class type...). The goal here is to have an idea of *what the data looks like*. *Hint:* This is one of those times when you should think about the cleanliness of your analysis. I added a single code chunk for you, but do you want to use more than one? Would you like to write more comments outside of the code chunk?

I used the `glimpse` function to look at each of the datasets. This function shows me three attributes for each dataset (# of rows, # of columns, class type for each value).

```
# Check breakdown of class types for each column

apt_buildings %>%
  glimpse()
```

```
## Rows: 3,455
## Columns: 37
## $ id                                <dbl> 10359, 10360, 10361, 10362, 10363, 10~
## $ air_conditioning                  <chr> "NONE", "NONE", "NONE", "NONE", "NONE~
## $ amenities                         <chr> "Outdoor rec facilities", "Outdoor po~
## $ balconies                         <chr> "YES", "YES", "YES", "YES", "NO", "NO~
## $ barrier_free_accessibilty_entr    <chr> "YES", "NO", "NO", "YES", "NO", "NO",~
## $ bike_parking                      <chr> "0 indoor parking spots and 10 outdoo~
```

```
## $ exterior_fire_escape             <chr> "NO", "NO", "NO", "YES", "NO", NA, "N~
## $ fire_alarm                       <chr> "YES", "YES", "YES", "YES", "YES", "Y~
## $ garbage_chutes                   <chr> "YES", "YES", "NO", "NO", "NO", "NO",~
## $ heating_type                     <chr> "HOT WATER", "HOT WATER", "HOT WATER"~
## $ intercom                         <chr> "YES", "YES", "YES", "YES", "YES", "Y~
## $ laundry_room                     <chr> "YES", "YES", "YES", "YES", "YES", "Y~
## $ locker_or_storage_room           <chr> "NO", "YES", "YES", "YES", "NO", "YES~
## $ no_of_elevators                  <dbl> 3, 3, 0, 1, 0, 0, 0, 2, 4, 2, 0, 2, 2~
## $ parking_type                     <chr> "Underground Garage , Garage accessib~
## $ pets_allowed                     <chr> "YES", "YES", "YES", "YES", "YES", "Y~
## $ prop_management_company_name     <chr> NA, "SCHICKEDANZ BROS. PROPERTIES", N~
## $ property_type                    <chr> "PRIVATE", "PRIVATE", "PRIVATE", "PRI~
## $ rsn                              <dbl> 4154812, 4154815, 4155295, 4155309, 4~
## $ separate_gas_meters              <chr> "NO", "NO", "NO", "NO", "NO", "NO", "~
## $ separate_hydro_meters            <chr> "YES", "YES", "YES", "YES", "YES", "Y~
## $ separate_water_meters            <chr> "NO", "NO", "NO", "NO", "NO", "NO", "~
## $ site_address                     <chr> "65  FOREST MANOR RD", "70  CLIPPER R~
## $ sprinkler_system                 <chr> "YES", "YES", "NO", "YES", "NO", "NO"~
## $ visitor_parking                  <chr> "PAID", "FREE", "UNAVAILABLE", "UNAVA~
## $ ward                             <chr> "17", "17", "03", "03", "02", "02", "~
## $ window_type                      <chr> "DOUBLE PANE", "DOUBLE PANE", "DOUBLE~
## $ year_built                       <dbl> 1967, 1970, 1927, 1959, 1943, 1952, 1~
## $ year_registered                  <dbl> 2017, 2017, 2017, 2017, 2017, NA, 201~
## $ no_of_storeys                    <dbl> 17, 14, 4, 5, 4, 4, 4, 7, 32, 4, 4, 7~
## $ emergency_power                  <chr> "NO", "YES", "NO", "NO", "NO", "NO", ~
## $ `non-smoking_building`           <chr> "YES", "NO", "YES", "YES", "YES", "NO~
## $ no_of_units                      <dbl> 218, 206, 34, 42, 25, 34, 14, 105, 57~
## $ no_of_accessible_parking_spaces  <dbl> 8, 10, 20, 42, 12, 0, 5, 1, 1, 6, 12,~
## $ facilities_available             <chr> "Recycling bins", "Green Bin / Organi~
## $ cooling_room                     <chr> "NO", "NO", "NO", "NO", "NO", "NO", "~
## $ no_barrier_free_accessible_units <dbl> 2, 0, 0, 42, 0, NA, 14, 0, 0, 1, 25, ~
```

```
# Check breakdown of class types for each column
building_permits %>%
  glimpse()
```

```
## Rows: 20,680
## Columns: 14
## $ permit_number               <chr> "BP-2016-02248", "BU468090", "DB-2016-0445~
## $ issue_date                  <date> 2017-02-01, 2017-02-01, 2017-02-01, 2017-~
## $ project_value               <dbl> 0, 0, 35000, 15000, 181178, 0, 15000, 0, 6~
## $ type_of_work                <chr> "Salvage and Abatement", "New Building", "~
## $ address                     <chr> "4378 W 9TH AVENUE, Vancouver, BC V6R 2C7"~
## $ project_description         <chr> NA, NA, NA, NA, NA, NA, NA, NA, NA, NA, NA~
## $ building_contractor         <chr> NA, NA, NA, "Mercury Contracting Ltd", "08~
## $ building_contractor_address <chr> NA, NA, NA, "88 W PENDER ST  \r\nUnit 2069~
## $ applicant                   <chr> "Raffaele & Associates DBA: Raffaele and A~
## $ applicant_address           <chr> "2642 East Hastings\r\nVancouver, BC  V5K ~
## $ property_use                <chr> "Dwelling Uses", "Dwelling Uses", "Dwellin~
## $ specific_use_category       <chr> "One-Family Dwelling", "Multiple Dwelling"~
## $ year                        <dbl> 2017, 2017, 2017, 2017, 2017, 2017, 2017, ~
## $ bi_id                       <dbl> 524, 535, 539, 541, 543, 546, 547, 548, 54~
```

```
# building_permits %>%
#   summarise_if(is.character, function(x){length(unique(x))})
```

```
# Check breakdown of class types for each column
cancer_sample %>%
  glimpse()
```

```
## Rows: 569
## Columns: 32
## $ ID                      <dbl> 842302, 842517, 84300903, 84348301, 84358402, ~
## $ diagnosis               <chr> "M", "M", "M", "M", "M", "M", "M", "M", "M", "~
## $ radius_mean             <dbl> 17.990, 20.570, 19.690, 11.420, 20.290, 12.450~
## $ texture_mean            <dbl> 10.38, 17.77, 21.25, 20.38, 14.34, 15.70, 19.9~
## $ perimeter_mean          <dbl> 122.80, 132.90, 130.00, 77.58, 135.10, 82.57, ~
## $ area_mean               <dbl> 1001.0, 1326.0, 1203.0, 386.1, 1297.0, 477.1, ~
## $ smoothness_mean         <dbl> 0.11840, 0.08474, 0.10960, 0.14250, 0.10030, 0~
## $ compactness_mean        <dbl> 0.27760, 0.07864, 0.15990, 0.28390, 0.13280, 0~
## $ concavity_mean          <dbl> 0.30010, 0.08690, 0.19740, 0.24140, 0.19800, 0~
## $ concave_points_mean     <dbl> 0.14710, 0.07017, 0.12790, 0.10520, 0.10430, 0~
## $ symmetry_mean           <dbl> 0.2419, 0.1812, 0.2069, 0.2597, 0.1809, 0.2087~
## $ fractal_dimension_mean  <dbl> 0.07871, 0.05667, 0.05999, 0.09744, 0.05883, 0~
## $ radius_se               <dbl> 1.0950, 0.5435, 0.7456, 0.4956, 0.7572, 0.3345~
## $ texture_se              <dbl> 0.9053, 0.7339, 0.7869, 1.1560, 0.7813, 0.8902~
## $ perimeter_se            <dbl> 8.589, 3.398, 4.585, 3.445, 5.438, 2.217, 3.18~
## $ area_se                 <dbl> 153.40, 74.08, 94.03, 27.23, 94.44, 27.19, 53.~
## $ smoothness_se           <dbl> 0.006399, 0.005225, 0.006150, 0.009110, 0.0114~
## $ compactness_se          <dbl> 0.049040, 0.013080, 0.040060, 0.074580, 0.0246~
## $ concavity_se            <dbl> 0.05373, 0.01860, 0.03832, 0.05661, 0.05688, 0~
## $ concave_points_se       <dbl> 0.015870, 0.013400, 0.020580, 0.018670, 0.0188~
## $ symmetry_se             <dbl> 0.03003, 0.01389, 0.02250, 0.05963, 0.01756, 0~
## $ fractal_dimension_se    <dbl> 0.006193, 0.003532, 0.004571, 0.009208, 0.0051~
## $ radius_worst            <dbl> 25.38, 24.99, 23.57, 14.91, 22.54, 15.47, 22.8~
## $ texture_worst           <dbl> 17.33, 23.41, 25.53, 26.50, 16.67, 23.75, 27.6~
## $ perimeter_worst         <dbl> 184.60, 158.80, 152.50, 98.87, 152.20, 103.40,~
## $ area_worst              <dbl> 2019.0, 1956.0, 1709.0, 567.7, 1575.0, 741.6, ~
## $ smoothness_worst        <dbl> 0.1622, 0.1238, 0.1444, 0.2098, 0.1374, 0.1791~
## $ compactness_worst       <dbl> 0.6656, 0.1866, 0.4245, 0.8663, 0.2050, 0.5249~
## $ concavity_worst         <dbl> 0.71190, 0.24160, 0.45040, 0.68690, 0.40000, 0~
## $ concave_points_worst    <dbl> 0.26540, 0.18600, 0.24300, 0.25750, 0.16250, 0~
## $ symmetry_worst          <dbl> 0.4601, 0.2750, 0.3613, 0.6638, 0.2364, 0.3985~
## $ fractal_dimension_worst <dbl> 0.11890, 0.08902, 0.08758, 0.17300, 0.07678, 0~
```

```
# Check breakdown of class types for each column
vancouver_trees %>%
  glimpse()
```

```
## Rows: 146,611
## Columns: 20
## $ tree_id            <dbl> 149556, 149563, 149579, 149590, 149604, 149616, 149~
## $ civic_number       <dbl> 494, 450, 4994, 858, 5032, 585, 4909, 4925, 4969, 7~
## $ std_street         <chr> "W 58TH AV", "W 58TH AV", "WINDSOR ST", "E 39TH AV"~
## $ genus_name         <chr> "ULMUS", "ZELKOVA", "STYRAX", "FRAXINUS", "ACER", "~
```

```
## $ species_name       <chr> "AMERICANA", "SERRATA", "JAPONICA", "AMERICANA", "C~
## $ cultivar_name      <chr> "BRANDON", NA, NA, "AUTUMN APPLAUSE", NA, "CHANTICL~
## $ common_name        <chr> "BRANDON ELM", "JAPANESE ZELKOVA", "JAPANESE SNOWBE~
## $ assigned           <chr> "N", "N", "N", "Y", "N", "N", "N", "N", "N", "N", "~
## $ root_barrier       <chr> "N", "N", "N", "N", "N", "N", "N", "N", "N", "N", "~
## $ plant_area         <chr> "N", "N", "4", "4", "4", "B", "6", "6", "3", "3", "~
## $ on_street_block    <dbl> 400, 400, 4900, 800, 5000, 500, 4900, 4900, 4900, 7~
## $ on_street          <chr> "W 58TH AV", "W 58TH AV", "WINDSOR ST", "E 39TH AV"~
## $ neighbourhood_name <chr> "MARPOLE", "MARPOLE", "KENSINGTON-CEDAR COTTAGE", "~
## $ street_side_name   <chr> "EVEN", "EVEN", "EVEN", "EVEN", "EVEN", "ODD", "ODD~
## $ height_range_id    <dbl> 2, 4, 3, 4, 2, 2, 3, 3, 2, 2, 2, 5, 3, 2, 2, 2, 2, ~
## $ diameter           <dbl> 10.00, 10.00, 4.00, 18.00, 9.00, 5.00, 15.00, 14.00~
## $ curb               <chr> "N", "N", "Y", "Y", "Y", "Y", "Y", "Y", "Y", "Y", "~
## $ date_planted       <date> 1999-01-13, 1996-05-31, 1993-11-22, 1996-04-29, 19~
## $ longitude          <dbl> -123.1161, -123.1147, -123.0846, -123.0870, -123.08~
## $ latitude           <dbl> 49.21776, 49.21776, 49.23938, 49.23469, 49.23894, 4~
```

1.3 Now that you've explored the 4 datasets that you were initially most interested in, let's narrow it down to 2. What lead you to choose these 2? Briefly explain your choices below, and feel free to include any code in your explanation. I ruled out the *cancer_sample* dataset because I found that the attributes were less interpretable. I also ruled out the *building_permits* dataset because a lot of the dataset was categorical data with a large number of categories. This type of data can be challenging to work with. I am left with:

1. *apt_buildings*: Acquired courtesy of The City of Toronto's Open Data Portal. It currently has 3455 rows and 37 columns.
2. *vancouver_trees*: Acquired courtesy of The City of Vancouver's Open Data Portal. It currently has 146611 rows and 20 columns.

1.4 Time for the final decision! Going back to the beginning, it's important to have an *end goal* in mind. For example, if I had chosen the `titanic` dataset for my project, I might've wanted to explore the relationship between survival and other variables. Try to think of 1 research question that you would want to answer with each dataset. Note them down below, and make your final choice based on what seems more interesting to you!

1. Do apartment buildings differ in accessibility-related attributes depending on what ward the building was constructed?
2. Does the location and surface boundaries (e.g. proximity to roads) influence tree growth?

I chose the first dataset, *apt_buildings*, because I felt that the research question was more compelling.

Task 2: Exploring your dataset (15 points)

Introduction

The dataset I selected is the *apt_buildings* dataset, which is a dataset of apartment buildings registered in Toronto with the Apartment Building Standard (ABS) program. The data includes attributes such as whether the building has air conditioning or not, whether there is visitor parking, and how many units the apartment has. There are 37 such attributes in total, recorded for 3,455 apartment buildings.

2.1 Complete *4 out of the following 8 exercises* to dive deeper into your data. All datasets are different and therefore, not all of these tasks may make sense for your data - which is why you should only answer *4*. Use *dplyr* and *ggplot*.

2.2 For each of the 4 exercises that you complete, provide a *brief explanation* of why you chose that exercise in relation to your data (in other words, why does it make sense to do that?), and sufficient comments for a reader to understand your reasoning and code.

3. Investigate how many missing values there are per variable. Can you find a way to plot this? In any data analytics pipeline, it is valuable to first explore the missingness in the dataset. I first looked at the number of missing data points for each variable (Figure 1). The most missing values occurred for the"amenities" and the "property management company name" variables.

When looking at the definition of the "amenities" attribute though (Are there amenities available in the building? If so, what is available? Note: Amenities include outdoor or indoor pool(s), indoor rec. room, child play area, etc.), it is not clear whether these are truly "missing" values or if they simply indicate that the building has no amenities.

For the property management company name attribute, I thought the missingness might have been correlated with the type of property (e.g. perhaps privately-owned companies were less likely to have a property manager). However, there did not appear to be such a trend (Table 1).

The rest of the variables had <160 missing data points, and most had <100 missing data points. This corresponds to <5% of the data. While it may be valuable to check if this missingess is correlated with the other covariates in the dataset, it doesn't seem like excluding missing data will pose a large problem for downstream analyses, depending on the model.

```
apt_buildings %>%
  summarise_all(function(x){sum(is.na(x))}) %>%
  t() %>% as.data.frame() %>%
  transmute(missingness=V1,
            variable=rownames(.)) %>%
ggplot(aes(x=variable, y=missingness)) +
  geom_bar(stat='identity') +
  theme_bw() +
  theme(axis.text.x = element_text(angle = 45, hjust=1))
```

```
kable(table(is.na(apt_buildings$prop_management_company_name), apt_buildings$property_type),
      caption="\\label{missingness.propertymanager} Missingness status of property management company na
```

Table 1: Missingness status of property management company name attribute versus the type of property (privately owned, social housing, or TCHC).

	PRIVATE	SOCIAL HOUSING	TCHC
FALSE	1789	127	176
TRUE	1099	113	151

4. Explore the relationship between 2 variables in a plot. My initial exploratory question was the relationship between accessibility and wards. So, I started by plotting the relationship between the percentage of apartment units that were barrier-free over time (according to year built) for each ward. This

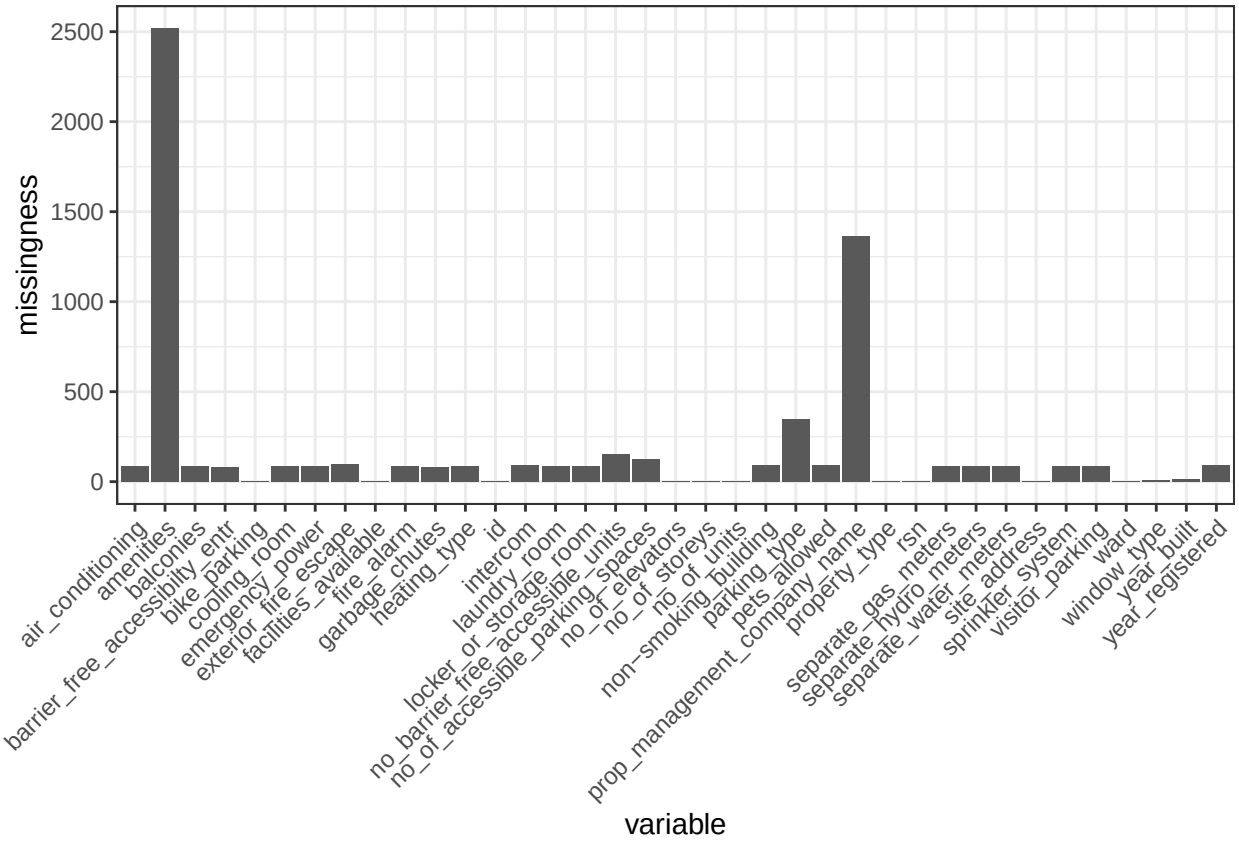

Figure 1: The number of records missing data for each of the attributes in the dataset.

could give me an idea of whether there has been a noticeable increase in accessibility over time, which I expect there to be. By color-coding by ward, I could see if the wards appeared to line up for development of accessible units or not.

By looking at Figure 2, iI could see that there was a general increase in percent of accessible units developed in the late 20th century, followed by a flattening out, where the rate of overall unit development was equal to the rate of accessible unit development. Surprisingly, this plateau of percentage of barrier-free units was quite different depending on the ward; in some wards, over 20% of the units were accessible by 2000; in other wards, virtually no units were accessible.

```
apt_buildings %>%
  # Group by years built
  group_by(year_built, ward) %>%
  # Sum the number of units in that year
  summarise(no_barrier_free_accessible_units=sum(no_barrier_free_accessible_units, na.rm=T),
            no_of_units = sum(no_of_units, na.rm=T)) %>%
  # Ungroup by year, now let's only group by ward, and arrange by year
  ungroup() %>% group_by(ward) %>% arrange(year_built) %>%
  # Calculate the cumulative sum of units and barrier-free units,
  # as well as the percent of units that are barrier free
  mutate(csum.barrierfree=cumsum(no_barrier_free_accessible_units),
         csum.units = cumsum(no_of_units),
         percent.csum.barrierfree = csum.barrierfree / csum.units) %>%
ggplot(aes(x=year_built, y=percent.csum.barrierfree*100, col=ward)) +
  geom_line() +
  theme_bw() +
  xlim(c(1900, 2020)) +
  xlab("Year built") + ylab("Percent of new units that are barrier-free accessible")
```

```
## `summarise()` has grouped output by 'year_built'. You can override using the `.groups` argument.
```

```
## Warning: Removed 24 row(s) containing missing values (geom_path).
```

6. Use a boxplot to look at the frequency of different observations within a single variable. You can do this for more than one variable if you wish! I wanted to explore aspects of accessibility for apartment buildings and how consistent "accessible" buildings are. I chose to compare the number of barrier-free accessible units reported for buildings that reported having a barrier-free accessibility entrance versus those that don't.

Unsurprisingly, buildings with accessible entrances appeared to have, on average, a higher number of accessible units (Figure 3). (Note that this relationship does not take into account the total number of units in the building, however). It was surprising to me that there were so many buildings with reportedly 'barrier-free accessible units', but the buildings didn't have accessible entrances. This led me to question the legitimacy of the "barrier-free accessible units".

I wondered if different wards had differing amounts of this "false accessibility", where buildings reportedly had barrier-free units but no barrier-free doors. I followed up by calculating the percentage of buildings in each ward that reported having barrier-free units which did not have a barrier-free entrance, and I found that this percentage differed pretty substantially across wards (Figure 4)

```
# Plot barrier-free accessible units according to whether or not the building had a barrier-free access
apt_buildings %>%
  filter(!is.na(barrier_free_accessibilty_entr)) %>%
```

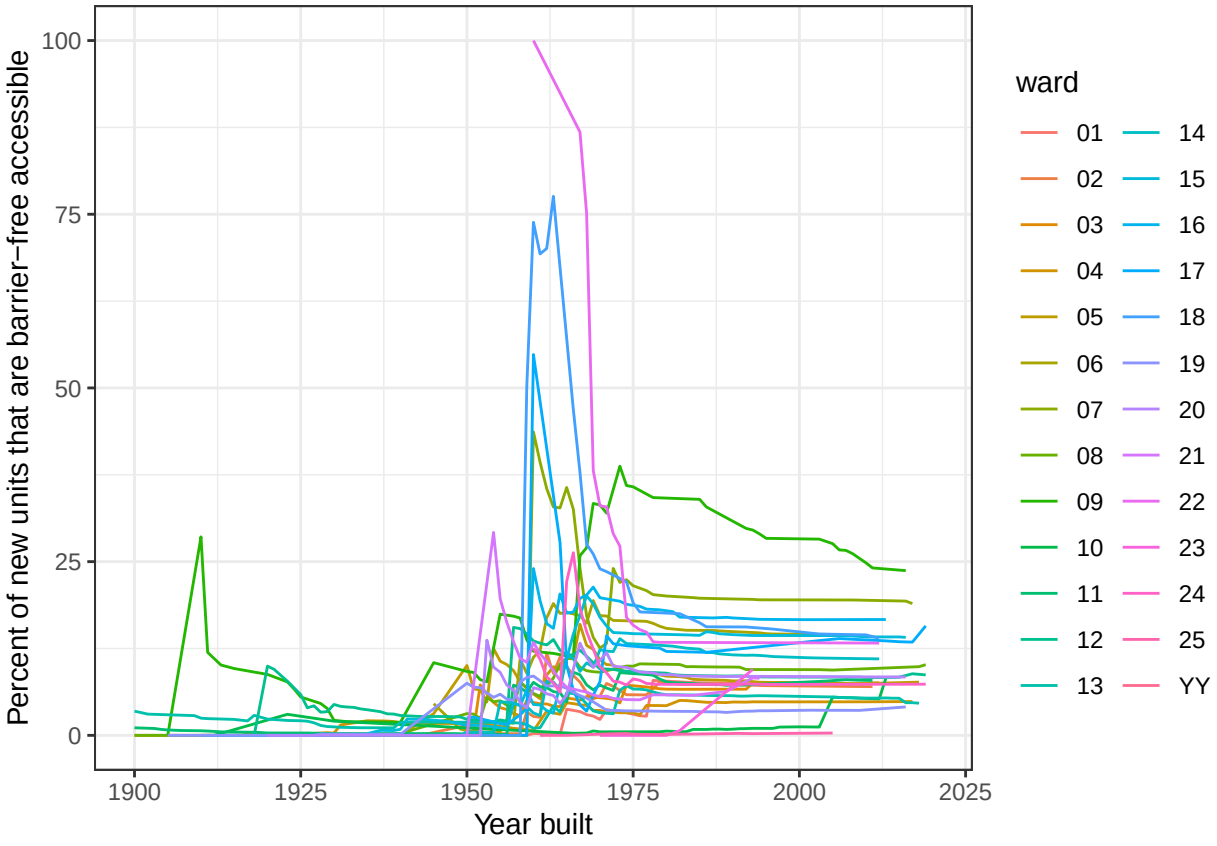

Figure 2: The percentage of barrier-free apartment units available in each ward through time.

```
ggplot(aes(x=barrier_free_accessibilty_entr,
           y=no_barrier_free_accessible_units)) +
  geom_boxplot(na.rm=T) +
  # nice display
  theme_bw() +
  # square-root scale the y-axis
  coord_trans(y="sqrt") +
  xlab("Barrier-free accessibility entrance") +
  ylab("Number of barrier-free accessible units")
```

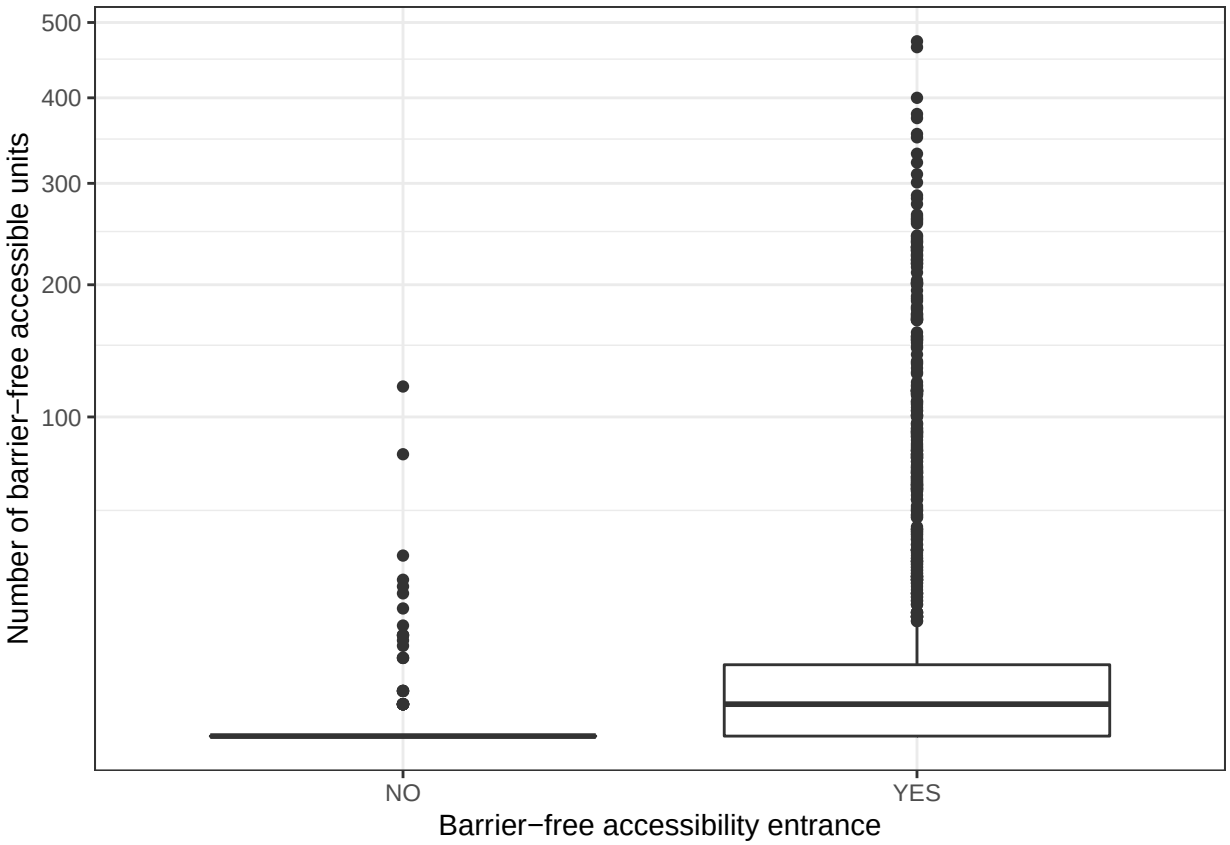

Figure 3: The number of barrier-free units in apartment buildings that reported having a barrier-free accessible entrance versus apartment buildings that reported not having a barrier-free accessible entrance.

```
# Percent of buildings with barrier-free accessible units that do not have an accessible entrance
apt_buildings %>%
  # Filter to only look at units with "barrier-free accessible units"
  filter(no_barrier_free_accessible_units > 0) %>%
  # Group by ward
  group_by(ward) %>%
  # For each ward, calculate the % of buildings that don't have an accessible entrance
  summarise(percent.without.accessible.entrance = sum(barrier_free_accessibilty_entr=="NO")/n()) %>%
  # Plot as a barchart
ggplot(aes(x=ward, y=percent.without.accessible.entrance*100)) +
  geom_bar(stat='identity') +
```

```
  xlab("Ward") + ylab("Percent of buildings with accessible units\nthat don't have accessible entrances
  theme_bw()
```

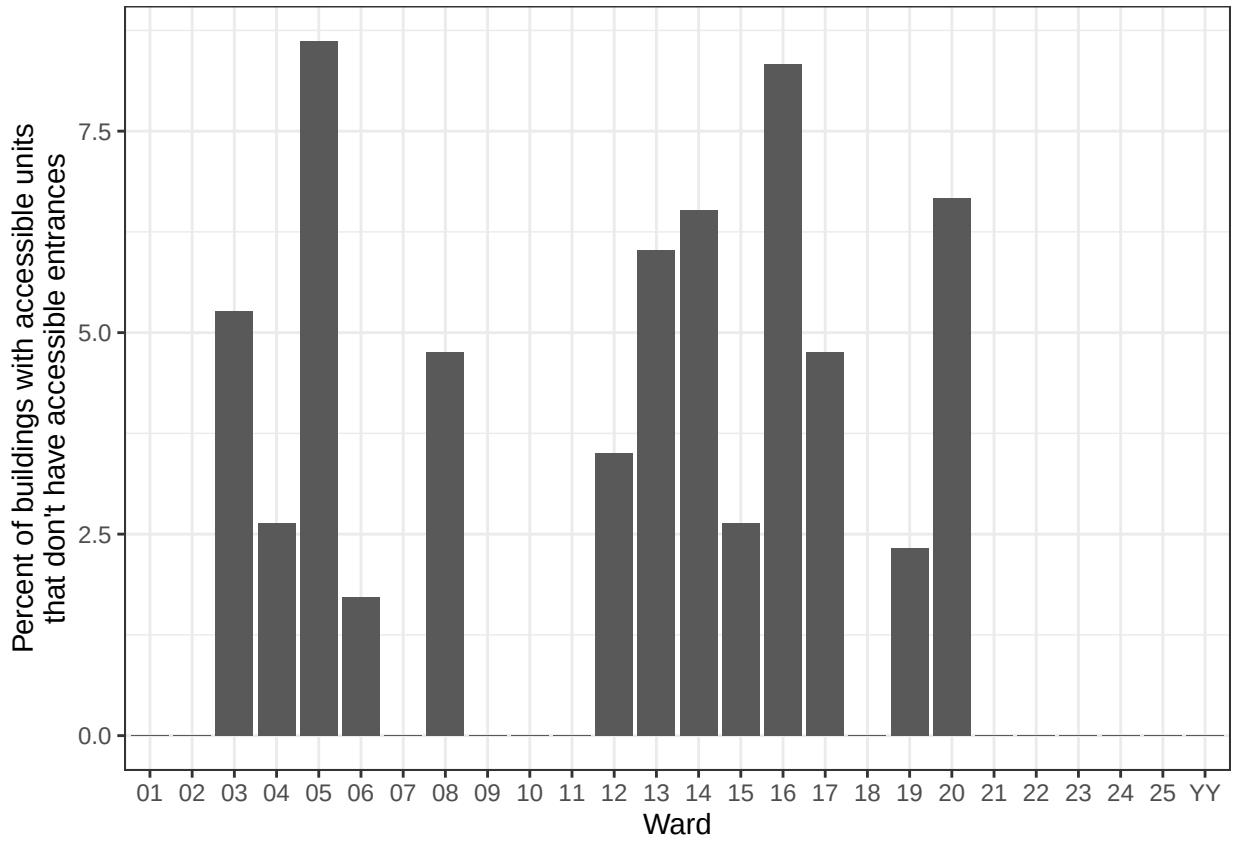

Figure 4: The percent of buildings with barrier-free accessible units that do not have an accessible entrance in each ward.

8. Use a density plot to explore any of your variables (that are suitable for this type of plot). Finally, I looked at the density of storeys in each ward (Figure 5). I would expect this to differ substantially for each ward– wards that are more suburban I would expect to see smaller-storeyed buildings. Understanding this distribution for each ward is important for interpreting factors such as bike parking versus car parking– more dense areas (wards with more big-storey apartments) would likely have more bike parking and less car parking.

```
apt_buildings %>%
ggplot(aes(x=no_of_storeys, y=ward)) +
  ggridges::geom_density_ridges() +
  xlab("Number of storeys") + ylab("Density") +
  theme_bw()
```

```
## Picking joint bandwidth of 1.56
```

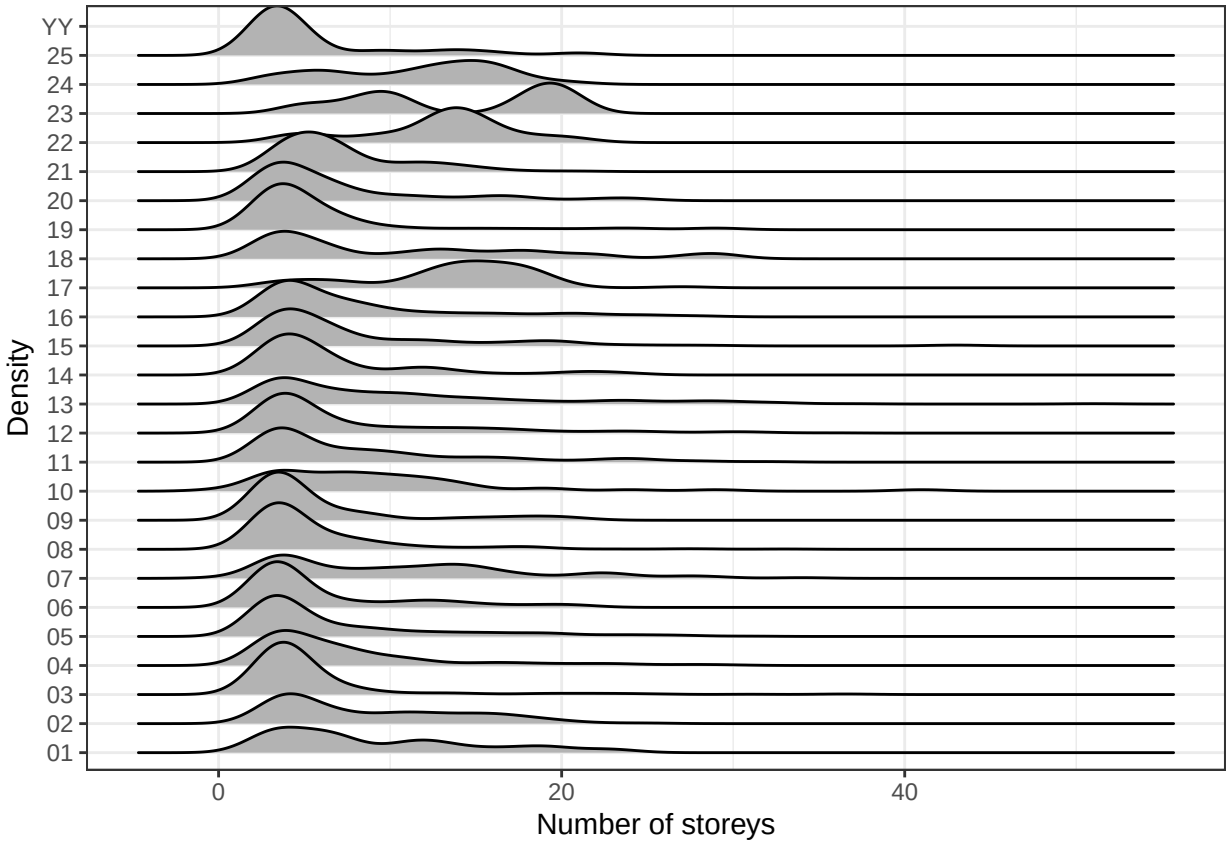

Figure 5: Density plot of the number of storeys per apartment building for each ward in the dataset.

Task 3: Write your research questions (5 points)

So far, you have chosen a dataset and gotten familiar with it through exploring the data. Now it's time to figure out 4 research questions that you would like to answer with your data! Write the 4 questions and any additional comments at the end of this deliverable. These questions are not necessarily set in stone - TAs will review them and give you feedback; therefore, you may choose to pursue them as they are for the rest of the project, or make modifications!

There are four overarching attributes that many of the variables in this dataset fit into:

1. Accessibility (e.g. barrier-free entrance, number of elevators, intercom, number of accessible parking spaces, garbage chutes, laundry room, number of elevators, number of barrier-free accessible units)
2. Sustainability (e.g. window type, air conditioning, bike parking, heating type)
3. Safety (exterior fire escape, fire alarm, sprinkler system, emergency power)
4. Quality of life (amenities, balconies, bike parking, visitor parking, locker and/or storage room, pets allowed)

For each of these attributes, I want to assess:

1. Whether the attribute is associated with the ward in which the building exists.
2. Whether the attribute has "increased" (i.e. more safety standards, higher 'quality of life', more sustainable & accessible buildings) over time.

Attribution

Thanks to Icíar Fernández Boyano for mostly putting this together, and Vincenzo Coia for launching.

Powered by the Academic theme for Hugo.