Homework2.rmd

Fabian Locher & Samuel Hänni

2024-12-09

Task 1

```
##
## 01 02 03 04 05 09 10 11 12
## 31 28 31 22 12  3 12 27 28
```

Explanation

Based on the results, the distribution of cold days per month aligns largely with my expectations. New York is known for its cold winters and warm summers, so the high number of cold days in winter months such as January (31 days), February (28 days), and December (28 days) seems reasonable. The data confirms that winter in New York is consistently cold, as expected.

What did surprise me, however, was the number of cold days in March (31 days). I had expected temperatures to begin warming in early spring, but this suggests that New York can experience lingering cold weather even into March. This could reflect colder-than-average weather patterns or specific cold fronts during that month.

In contrast, the results for the summer months show no cold days, reinforcing the idea that New York experiences reliably warm summers. This is consistent with my general understanding of its climate.

Overall, the results match expectations for New York's climate: cold, steady winters and predictably warm summers. The only notable exception is March, which appears unusually cold. These variations highlight how regional weather patterns can influence a city's climate dynamics, even within familiar seasonal trends.

Task 2

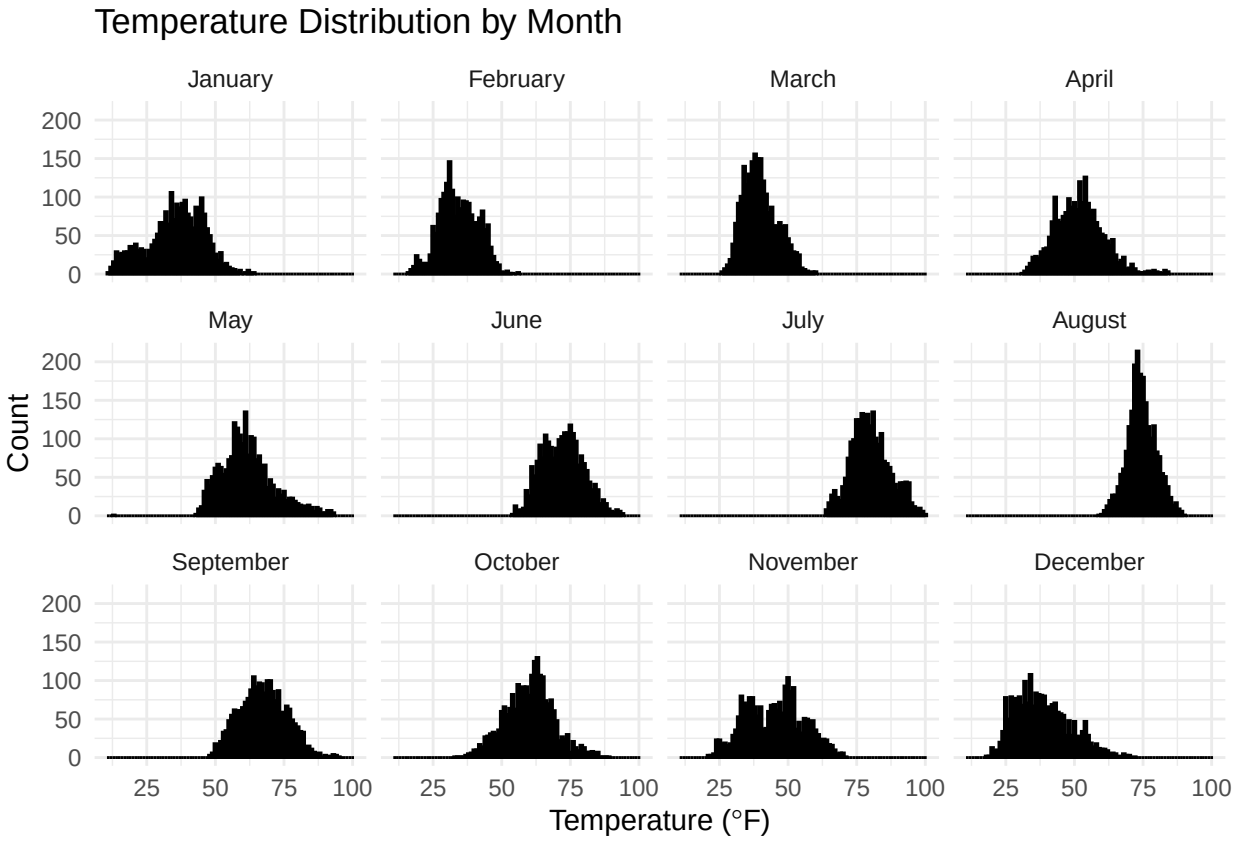
```
## # A tibble: 19 x 2
##    variable               missing_count
##    <chr>                          <int>
##  1 missing_year                       0
##  2 missing_month                      0
##  3 missing_day                        0
##  4 missing_dep_time                8255
##  5 missing_sched_dep_time             0
##  6 missing_dep_delay               8255
##  7 missing_arr_time                8255
##  8 missing_sched_arr_time             0
##  9 missing_arr_delay               8255
## 10 missing_carrier                    0
## 11 missing_flight                     0
## 12 missing_tailnum                 2512
## 13 missing_origin                     0
```

```
## 14 missing_dest                    0
## 15 missing_air_time             8255
## 16 missing_distance                0
## 17 missing_hour                    0
## 18 missing_minute                  0
## 19 missing_time_hour               0
```

Explanation

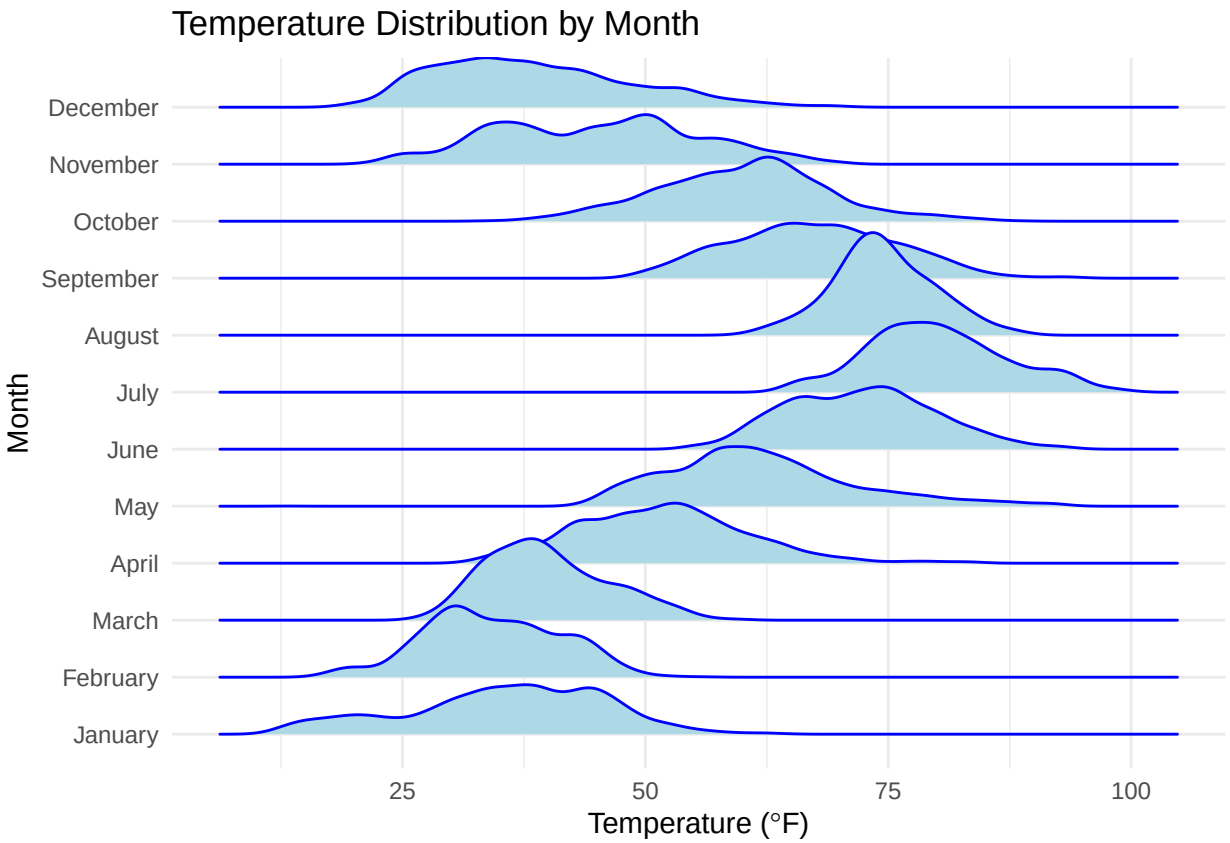
Rows with missing dep_time likely represent flights that were canceled. These rows have missing values for other related columns, such as arr_time, air_time, and sched_dep_time, because a canceled flight does not have actual departure or arrival times recorded. Additionally, the absence of sched_dep_time might indicate flights removed from the schedule altogether due to external factors such as weather conditions, operational issues, or low demand.

Task 3a

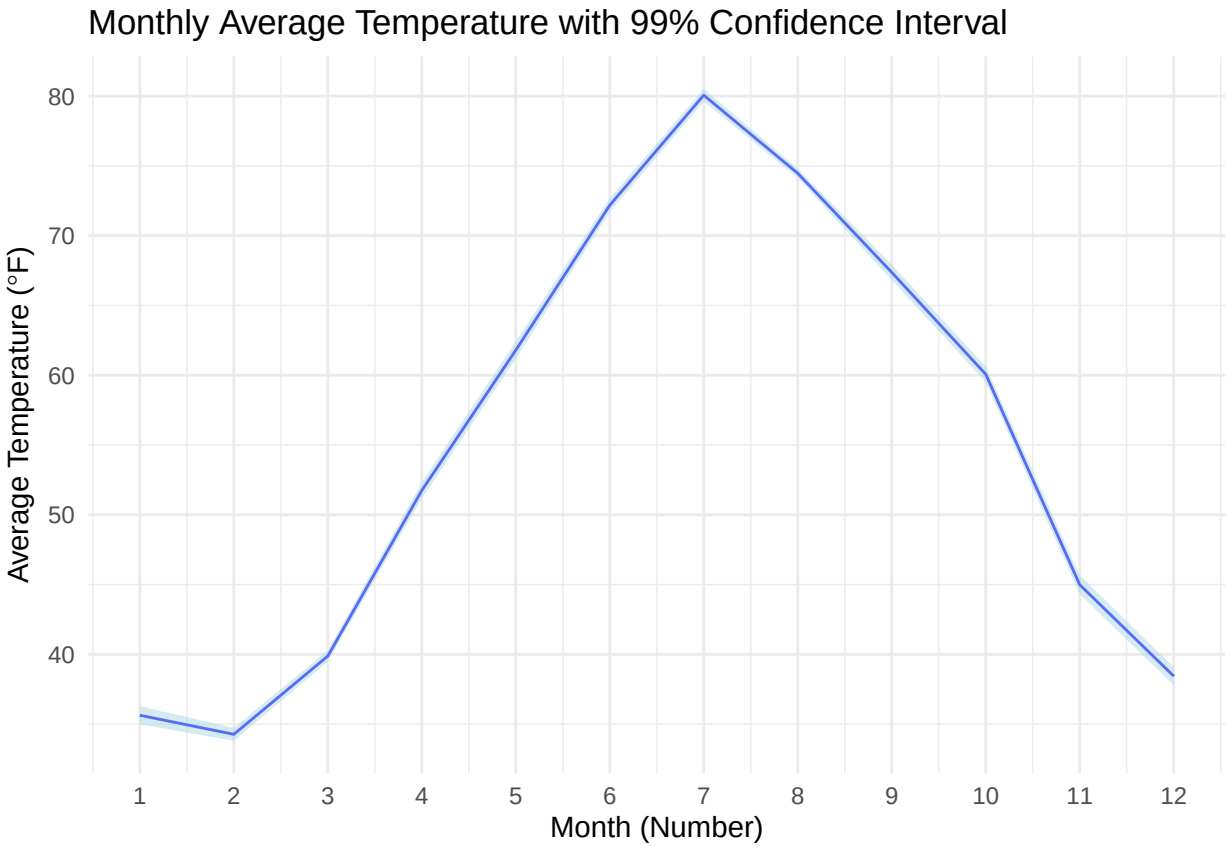


Task 3b

```
## Picking joint bandwidth of 1.58
```


Temperature Distribution by Month
December
November
October
September
August
July
June
May
April
March
February
January
Month
25
50
75
100
Temperature (°F)

Task 3c



Task 4

date of birth

Problem: The 'date of birth' column contains dates in different formats. Some are in "YYYY-MM-DD" format, while others use "DD.MM.YYYY". This inconsistency needs to be addressed to ensure uniformity in the dataset.

Solution: Convert all date formats to a standard "YYYY-MM-DD" format using lubridate.

Height

Problem: The 'height' column has inconsistent units. Some values are in 'cm', while others are in 'm' (example., '1,82m'). This inconsistency makes it difficult to analyze the data.

Solution: Convert all heights to centimeters. For values in meters ('m'), multiply by 100 after converting them to numeric.

Feet

Problem: The 'foot' column contains unusually large values, likely due to a mix-up with shoe sizes. Foot lengths above 40 cm are in our opinion unrealistic and should be treated as invalid.

Solution: Use mutate() to replace values greater than 40 with NA, indicating invalid entries.

Hair

Problem: The 'hair' column contains the value "Glatze" This is inconsistent with numerical hair lengths. It should be replaced with 0.

Solution: Use mutate() to replace "Glatze" with 0 and ensure all values are numeric.

eye colour

Problem: The 'eye colour' column contains inconsistent values due to: - Mixed languages (German and English) - Inconsistent capitalization - Multiple colors separated by spaces or slashes

Solution: Use mutate() to standardize all values to lowercase English and unify multiple colors with a hyphen.

Cash

Problem: The 'cash (CHF)' column has missing values (interpreted as 0) and unrealistically high values like 4000. These high values likely refer to salary and should be removed or capped.

Solution: - Replace missing values with 0. - Cap all values greater than 400 to 400 (reasonable maximum for cash on hand).

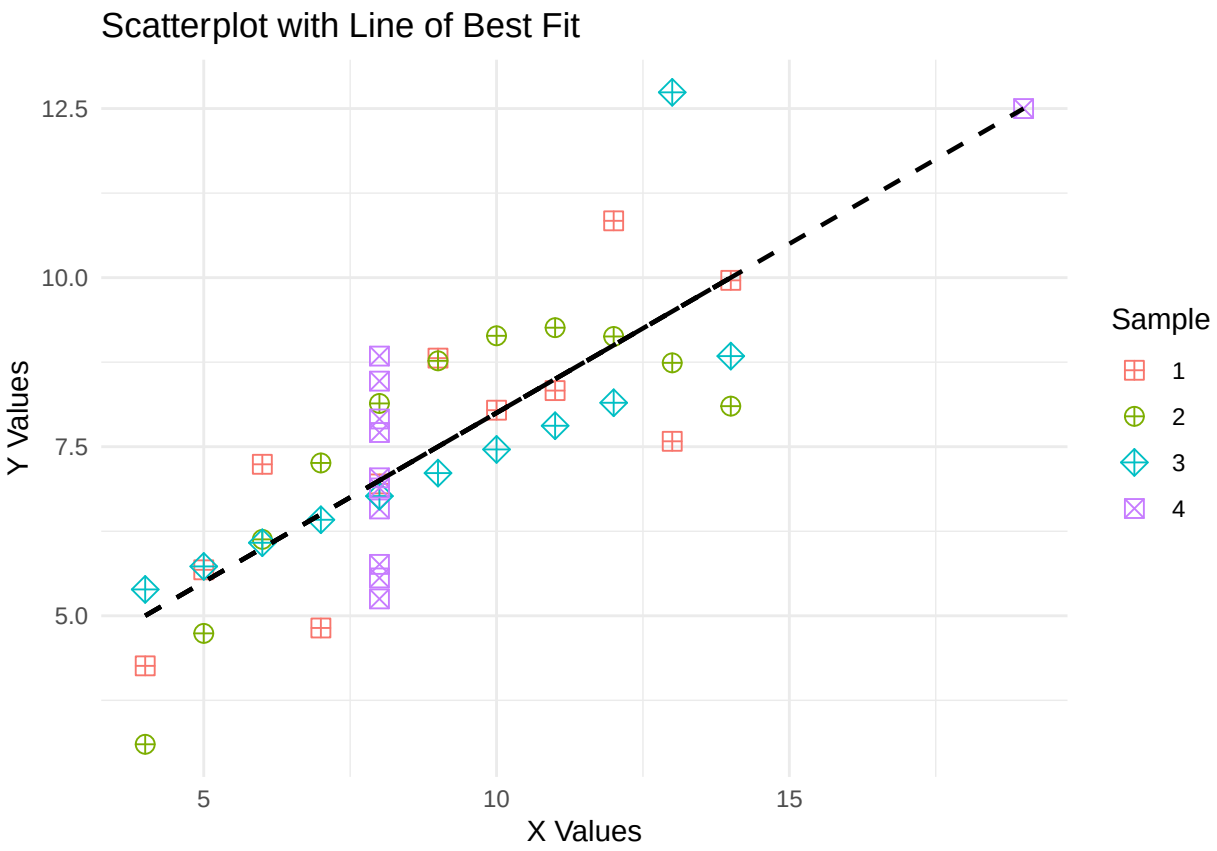
Task 5a

The dataset is not tidy because: Variables are stored in column names: Each sample (1, 2, 3, 4) has its x and y values in separate columns like x1, y1, x2, etc. Instead, each sample should #be identified as a separate observation. Data is not normalized: Instead of having one column for x, one for y, and another for the sample, the dataset has redundant columns for each sample. Structure is inefficient: This wide format makes it harder to analyze the data as a whole since the relationships between x and y for all samples are split #across multiple columns. Why tidying is important: To analyze the data properly, we need to convert it into a tidy format where:

Each variable has its own column (x, y, sample). Each observation is in its own row.

Task 5b

```
## `geom_smooth()` using formula = 'y ~ x'
```

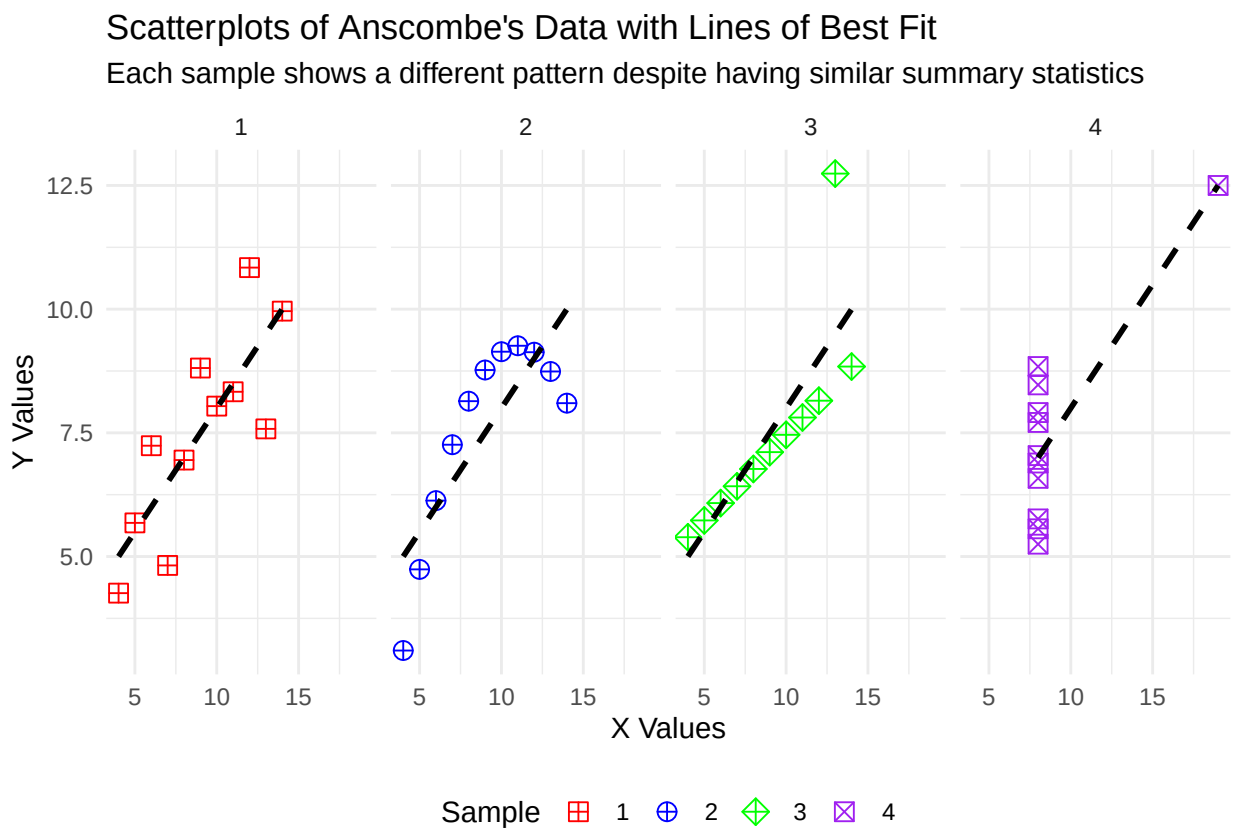


Task 5c

Sample	mean_x	mean_y	sd_X	sd_Y	corr_xy
1	9	7.501	3.317	2.032	0.816
2	9	7.501	3.317	2.032	0.816
3	9	7.500	3.317	2.030	0.816
4	9	7.501	3.317	2.031	0.817

Task 5d

```
## `geom_smooth()` using formula = 'y ~ x'
```



Workload Samuel

We sat down at the beginning and divided up the tasks. The first thing we did was set up a Git repository so that we could easily collaborate and continuously track each other's progress. We organized the tasks so that Samuel took on tasks 1, 2 and 3, while Fabian handled tasks 4 and 5. This clear division helped us focus and avoid overlapping efforts. After completing our initial assignments, we held a short meeting to discuss the status and share updates on our progress. This check-in proved beneficial, as it highlighted some areas that needed further refinement. Although a few items were still incomplete, we felt that we were moving in the right direction and understood what remained to be done. To ensure the quality of each other's work, we decided that Fabian would review and make corrections to Samuel's tasks, and vice versa. This mutual review process not only helped catch errors but also facilitated a better understanding of each other's approach. By the end of the session, we felt more confident about our progress and looked forward to wrapping up the remaining tasks.

Workload Fabian

I began by taking on tasks 4 and 5, focusing on completing them with care. After finishing my assignments, I did a final review of everything to ensure that everything was working as expected. I saw that we had a mistake in number 1, i think we had 35 cold days in january, which is of course rather difficult in a month that only has 31 days. But I think I was able to fix it now, but that also shows how important it is to check the whole thing again . Once I was satisfied with the quality of the work, I submitted everything. By the end, I felt confident that the tasks were well-executed and met the necessary standards.