Project 3: Transit Cost Analysis

Lindsay Chu, LDC2368

This is the dataset used in this project:

```
transit_cost <- readr::read_csv('https://raw.githubusercontent.com/rfordatascience/tidytuesday/master/da
```

Link to the dataset: *https://github.com/rfordatascience/tidytuesday/blob/master/data/2021/2021-01-05/readme.md*

Data Cleaning Section

```
transit_cost <- transit_cost %>%
  # omit all rows without ids
  filter(!is.na(e)) %>%
  rename(
    id = e,
    railroad = rr,
    proposed_length = length,
    actual_length = tunnel,
    cost_usd = real_cost,
    cost_per_km_usd = cost_km_millions
    ) %>%
  mutate(
    # round other numeric variables to 2 decimal places
    cost_usd = round(as.numeric(cost_usd), 2),
    start_year = as.numeric(start_year),
    end_year = as.numeric(end_year),
    proposed_length = round(proposed_length, 2),
    actual_length = round(actual_length, 2),
    # edit percentage of tunnel completed to a numeric value instead of a %
    tunnel_per = as.numeric(sub("%", "", tunnel_per)),
    ppp_rate = round(ppp_rate, 2),
    cost_per_km_usd = round(cost_per_km_usd, 2),
    # create a new variable that calculates project duration
    project_duration = end_year - start_year
    ) %>%
  mutate(
    # create a new variable called "continent" to sort the countries
    continent = countrycode(sourcevar = country,
                            origin = "iso2c",
                            destination = "continent"),
    # convert all country codes to full names and update "country" variable
    country = countrycode(sourcevar = country,
                          origin = "iso2c",
                          destination = "country.name")
    ) %>%
  # replace NA values
  mutate(
```

```
    continent = ifelse(is.na(continent), "Europe", continent),
    country = ifelse(is.na(country), "United Kingdom", country)
    ) %>%
  mutate_all(~replace(., is.na(.), 0)) %>%
  # select all variables except: id, source1, cost, currency, year (midpoint), ppp_rate, source2, and r
  # reorder
  select(continent, country, city, line, stations, railroad, start_year, end_year, project_duration,
         proposed_length, actual_length, tunnel_per, cost_usd, cost_per_km_usd)
```

Note: Prior to analysis, I had to clean the data to address a few issues.
1. Some entries did not have information on **cost_per_km**, etc., a variable crucial to Part 1's analysis. Therefore, such entries were omitted from the analysis using `filter()`.
2. The variables **real_cost** , **start_year** , **end_year** and **tunnel_per** were saved as *character* variables rather than doubles, so I converted them with the `as.numeric()` function.
3. The dataset used country codes (i.e. abbreviations), which could be unfamiliar or difficult to interpret. Hence, I imported the `countrycode` library to convert all codes to full names, and updated the **country** variable.
4. The dataset also had NA country values, so I also used `countrycode` to replace them with the intended "United Kingdom" value and sort them into the continent "Europe".

Part 1

Question: *Which are the top 10 countries for number of projects, and what are their distributions in terms of cost (in millions) per kilometer?*

Introduction: This project focuses on analyzing the `transit_cost` dataset, originally compiled by the Transit Cost project and posted on GitHub repository, `tidytuesday` on Jan 5, 2021. This dataset compiles records of 427 railroad and non-railroad projects since the 1980s, and each record contains the following 14 characteristics: **continent**, **country**, **city**, **line** (i.e. project name), the number of **stations**, whether or not the project is a **railroad** (1==railroad, 0==non-railroad), the project's **start_year**, **end_year**, the **project_duration** (difference between the two years), **proposed_length**, **actual_length**, the percentage how much was or has been constructed (**tunnel_per**), the total cost in million USD (**cost_usd**), and cost per kilometer in million USD (**cost_per_km_usd**).

Part 1 asks, "Which are the top 10 countries for number of projects, and what are their distributions in terms of cost (in millions) per kilometer?" To answer this question, we will work with the following variables: the **country** names (specifically the top 10 countries in terms of the number of projects), the **cost_per_km_usd**, the **tunnel_per** to calculate the proportion of completed projects per country, and the **continent** names (to colorcode the top 10 countries in the visualizations).

Approach: To identify the top 10 countries and examine how their distributions compare in terms of cost/km (in million USD), we will use two methods.

We will create a **summary table** that will `count` the frequency of countries (i.e. number of projects for that country), retain the top 10 causes, and lump the remaining causes into a category called **"Other"** using the functions `fct_lump_n()` and `fct_infreq()`. The table will also calculate each country's minimum, average, and maximum costs per kilometer, and the proportion of completed , for comparative purposes.

Subsequently, we will create **boxplots** with `geom_boxplot()` to visualize the distributions of cost/km for the top 10 countries. The boxplots help reinforce the summary table's calculations for minimum, mean, and maximum values. They are also arranged in terms of the greatest spread of values to better understand how much each country tends to finance its projects.

Analysis:

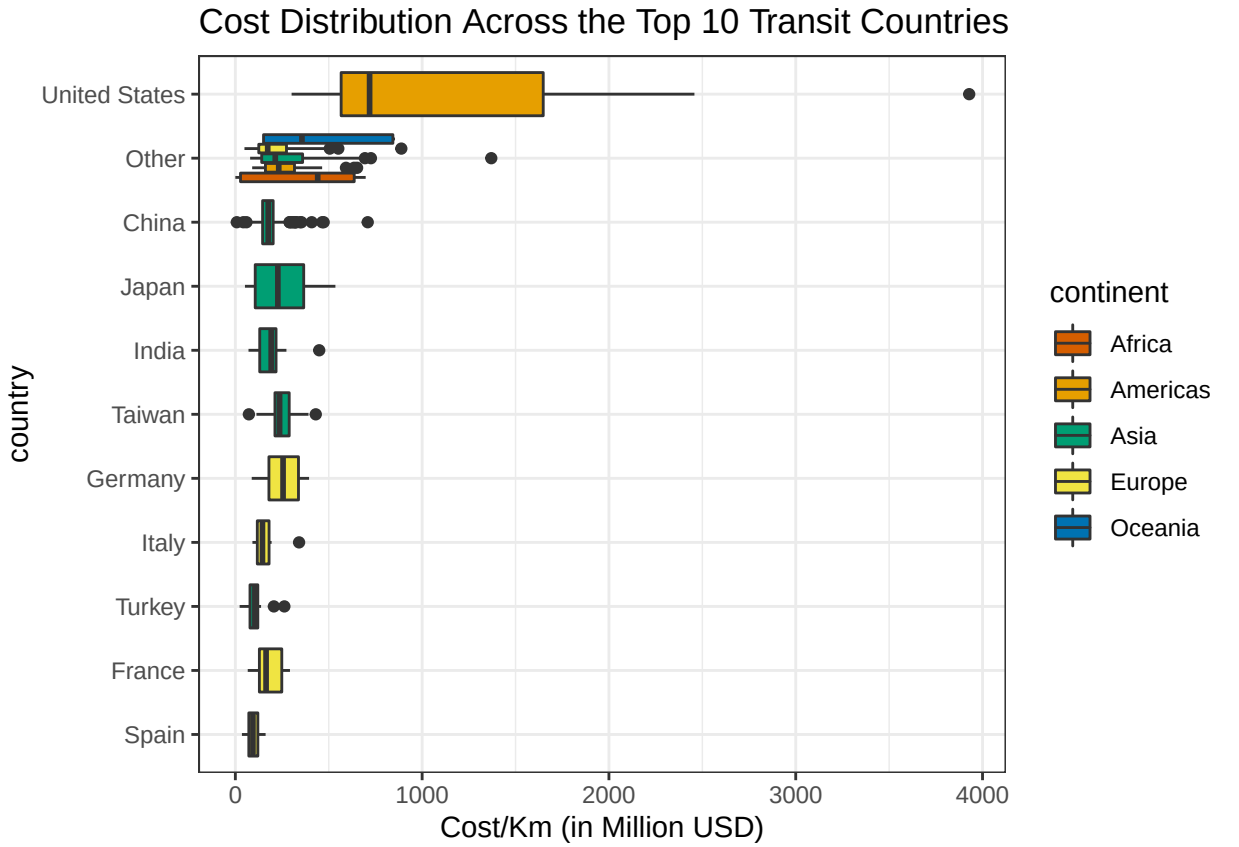
country	total_projects	percent_completed	min_cost_per_km	avg_cost_per_km	max_cost_per_km
China	253	58.89%	7.79	184.39	708.65
Other	141	46.81%	0.00	275.62	1370.12
India	29	3.45%	70.50	186.57	448.96
Turkey	20	75%	22.88	107.96	262.42
Spain	15	60%	35.25	97.16	161.93
France	15	73.33%	66.25	183.38	292.50
Japan	15	40%	51.43	241.26	535.71
Germany	13	92.31%	87.75	251.65	394.69
United States	13	84.62%	301.18	1211.47	3928.57
Taiwan	12	25%	72.57	246.94	430.61

```
# top 10 countries for projects and average cost (in million USD)
top10 <- mutate(transit_cost, country = fct_lump(country, 10)) %>%
                arrange(cost_per_km_usd) %>%
                mutate(country = fct_reorder(country, cost_per_km_usd))

table <-
  group_by(top10, country) %>%
  summarize(
    total_projects = n(),
    percent_completed = paste0(round(100 * sum(tunnel_per==100.00) / total_projects, 2), "%"),
    min_cost_per_km = round(min(cost_per_km_usd), 2),
    avg_cost_per_km = round(mean(cost_per_km_usd), 2),
    max_cost_per_km = round(max(cost_per_km_usd), 2)
  ) %>%
  arrange(-total_projects) %>%
  head(10) %>%
  kbl() %>%
  kable_material(c("striped", "hover"))
```

```
## `summarise()` ungrouping output (override with `.groups` argument)
```

```
table
```

```
# boxplots
top10 %>%
  mutate(country = fct_reorder(country, cost_per_km_usd, function(x) { max(x) - min(x) })) %>%
  ggplot(aes(cost_per_km_usd, country, fill = continent)) +
  ggtitle("Cost Distribution Across the Top 10 Transit Countries") +
  geom_boxplot() +
  labs(x="Cost/Km (in Million USD)") +
  theme_bw() +
  scale_fill_manual(values =
                      c(Americas = ("#E69F00"),
                        Africa = ("#D55E00"),
                        Asia =  ("#009E73") ,
                        Europe = ("#F0E442"),
                        Oceania = ("#0072B2")
                        )
                    )
```



Discussion: From the summary table, we observe that China has the highest number of transit projects, but only around half of them have completed construction. The US and Germany have much fewer lines than most other countries, but significantly higher percentages of completed tunnels. Based on the table and the boxplots, it is evident that the United States has a significantly higher **cost_per_km** than that of any other country, even other developed countries like France and Germany. The United States also has the highest range, or spread, of values for **cost_per_km**, with a maximum expenditure of almost *$4000 million*, and the highest average at *$1211.47 million*. The rest of the countries appear to have much lower spread of values that fall *below $1000 million.*; this is most likely due to very low number of projects, less money and or resources to execute these projects, etc.

Part 2

Question: *Are there any significant relationships among the numeric variables, and can the dataset be reduced to smaller dimensions to better understand such relationships?*

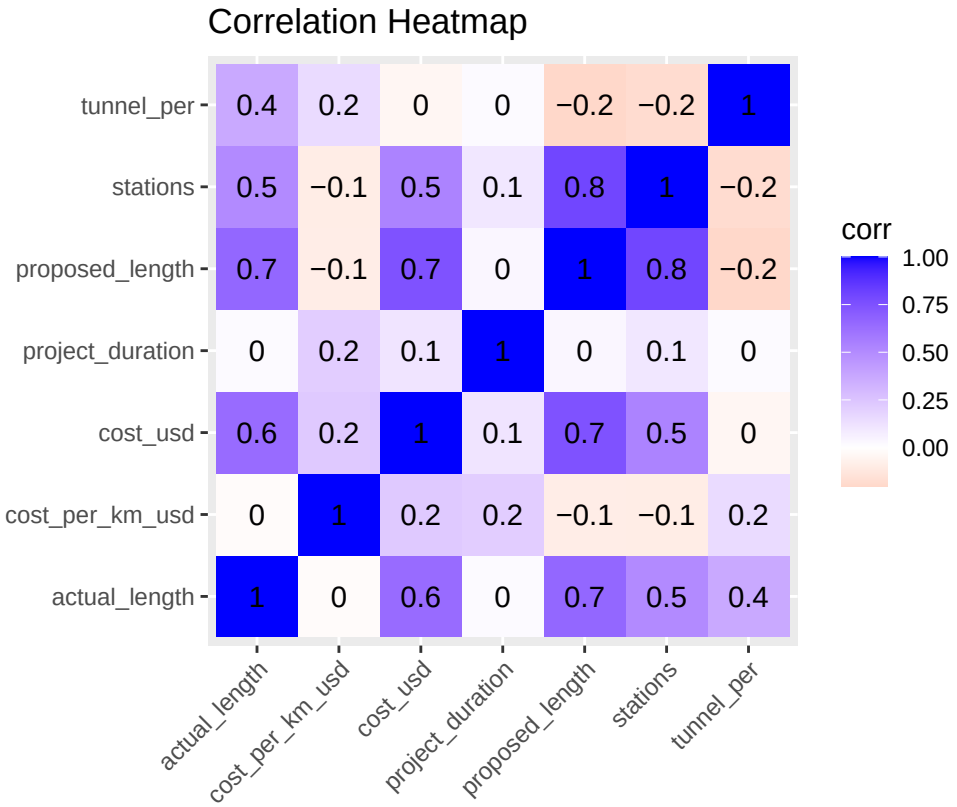
Introduction: To answer these questions, we will work with the following numeric variables: the **tunnel_per**, the number of **stations**, the **proposed_length**, the **project_duration**, the **cost_usd**, the **cost_per_km_usd**, and the **actual_length**; the variables **start_year** and **end_year** have been excluded due to the existence of the **project_duration** variable (i.e. the difference between the aforementioned variables).

Approach: The first part of our analysis asks, "Are there any significant relationships among the numeric variables?" To answer this, we will create a **correlation heatmap** by calculating all correlation values among all numeric values, and then displaying them using the `geom_tile()` function of ggplot graphics. This heatmap will help us determine which variables, if any, have particularly strong associations with one another based on values from -1 to 1, inclusive, and based on a color gradient.

The second part examines whether or not the dataset can be reduced to smaller dimensions to better understand such relationships among variables. To address this, we will perform Principal Components Analysis (PCA), a type of dimension reduction technique. This method is ideal because the `transit_cost` dataset contains various characteristics of different train lines, and it would be useful to understand if the dataset could be broken down into smaller factors without losing any critical information about trains. First, we will `scale()` the data and use `prcomp()` to break down the dataset into different principal components (PCs). Each PC has loading values for the variables listed in the Introduction section; these are similar to correlations in that they help explain which of the attributes are most associated with each PC! Afterwards, we will use `ggplot()` to plot a few pairs of PCs and see if the variables gravitate towards any PC in particular; this will be denoted by whether or not an arrow for a given variable points towards a PC's axis. These graphs are known as rotation matrices, and they help show relationships between variables. We will also use create a simple bar chart with `geom_col()` to show the variance explained by each component. The higher the % variance explained, the higher influence that PC has on the entire dataset.

Analysis:

```
part2data <-
  transit_cost %>%
  select(where(is.numeric), -start_year, -end_year, -railroad)

# correlation heatmap
corrmap <-
  part2data %>%
  cor %>%
  as.data.frame %>%
  rownames_to_column %>%
  pivot_longer(-1) %>%
  ggplot(aes(rowname, name, fill=value)) +
  ggtitle("Correlation Heatmap") +
  geom_tile() +
  geom_text(aes(label=round(value,1))) +
  labs(x = "", y = "", fill="corr") +
  coord_fixed() +
  theme(axis.text.x = element_text(angle=45, vjust=1, hjust=1)) +
  scale_fill_gradient2(low="red", mid="white", high="blue")

corrmap
```

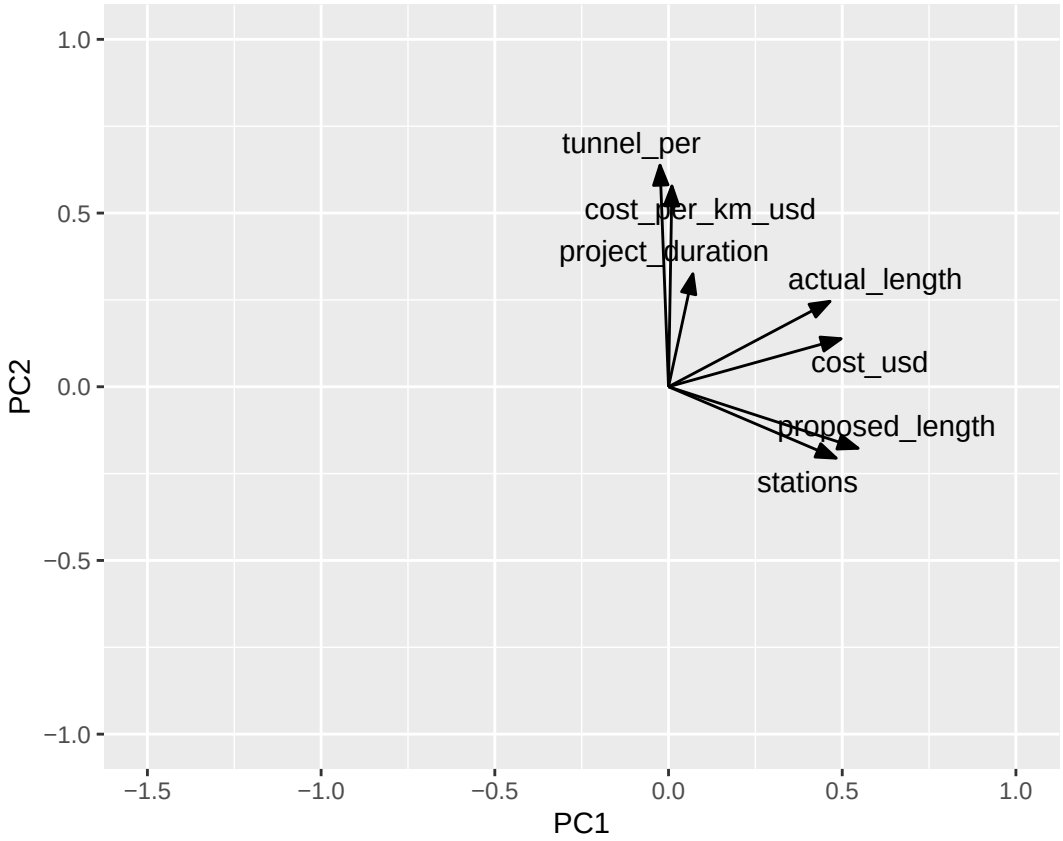



```
# dimension reduction
pca_fit <-
  part2data %>%
  scale() %>%
  prcomp()

pca_fit
```

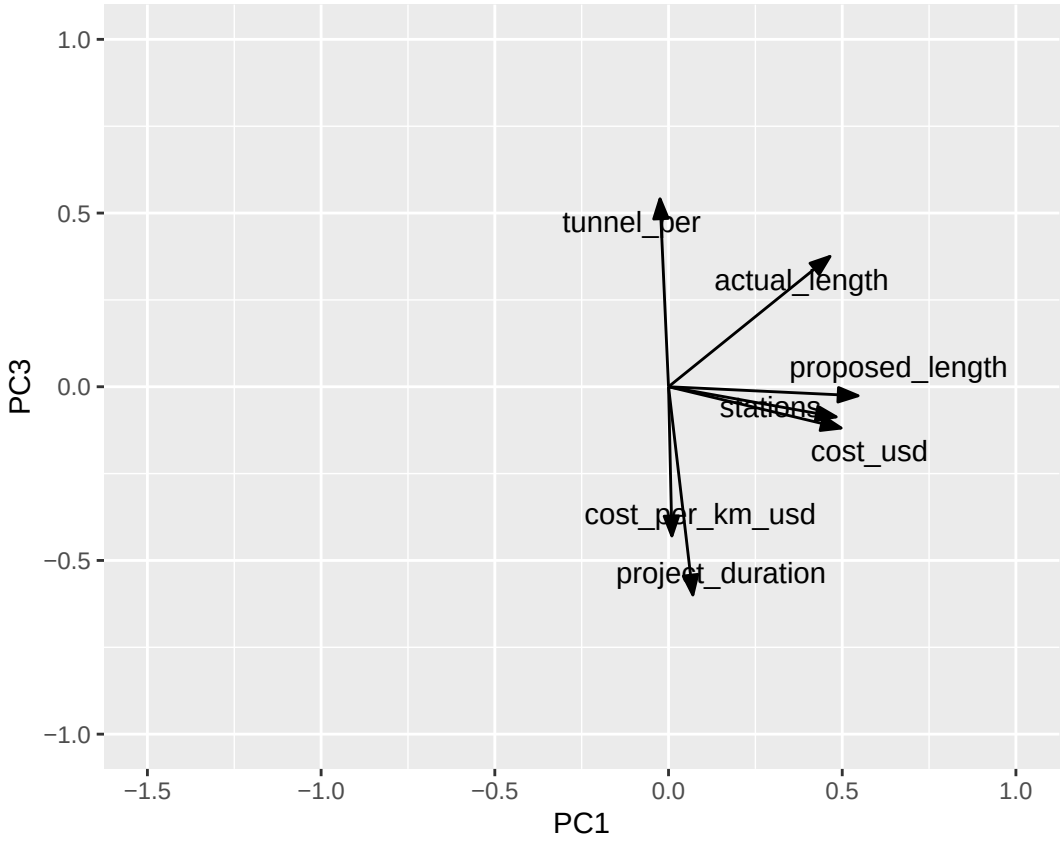
```
## Standard deviations (1, .., p=7):
## [1] 1.7212136 1.1823793 1.0764869 0.8996269 0.6215065 0.4214955 0.3275996
##
## Rotation (n x k) = (7 x 7):
##                              PC1        PC2         PC3        PC4          PC5
## stations           0.482805218 -0.2055846 -0.08676846  0.1242290 -0.736985078
## project_duration   0.070118803  0.3253645 -0.59953767  0.7156236  0.126497522
## proposed_length    0.545838029 -0.1771114 -0.02581804 -0.0254432 -0.003607179
## actual_length      0.465056457  0.2459249  0.37532352  0.1039532  0.189193491
## tunnel_per        -0.024439900  0.6377908  0.54111323  0.2161256 -0.225622157
## cost_usd           0.497066224  0.1382942 -0.11897942 -0.2689208  0.519505835
## cost_per_km_usd    0.009678538  0.5780337 -0.42957172 -0.5847801 -0.290262834
##                             PC6         PC7
## stations           0.18098654 -0.35463975
## project_duration -0.03220532  0.02489984
## proposed_length  -0.06953469  0.81518733
## actual_length    -0.66811885 -0.29898571
## tunnel_per        0.39674430  0.21166093
```

```
## cost_usd          0.55408618 -0.26538198
## cost_per_km_usd  -0.22488061  0.06678226
```

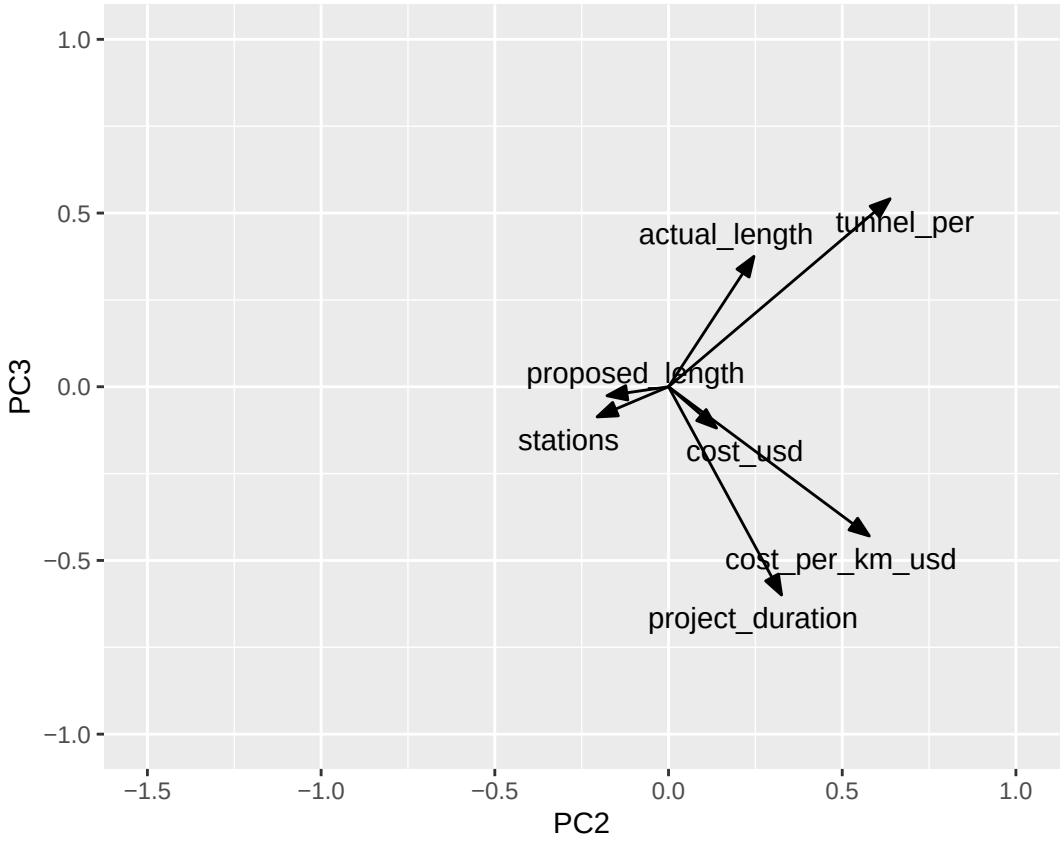
```r
# rotation matrix plots
arrow_style <- arrow(
  angle = 20, length = grid::unit(8, "pt"),
  ends = "first", type = "closed"
  )

p12 <- pca_fit %>%
  tidy(matrix = "rotation") %>%
  pivot_wider(
  names_from = "PC", values_from = "value",
  names_prefix = "PC"
  ) %>%
  ggplot(aes(PC1, PC2)) +
  geom_segment(
    xend = 0, yend = 0,
    arrow = arrow_style
  ) +
  geom_text_repel(
    aes(label = column),
    max.overlaps = Inf
  ) +
  xlim(-1.5, 1) + ylim(-1, 1) +
  coord_fixed()

p12
```



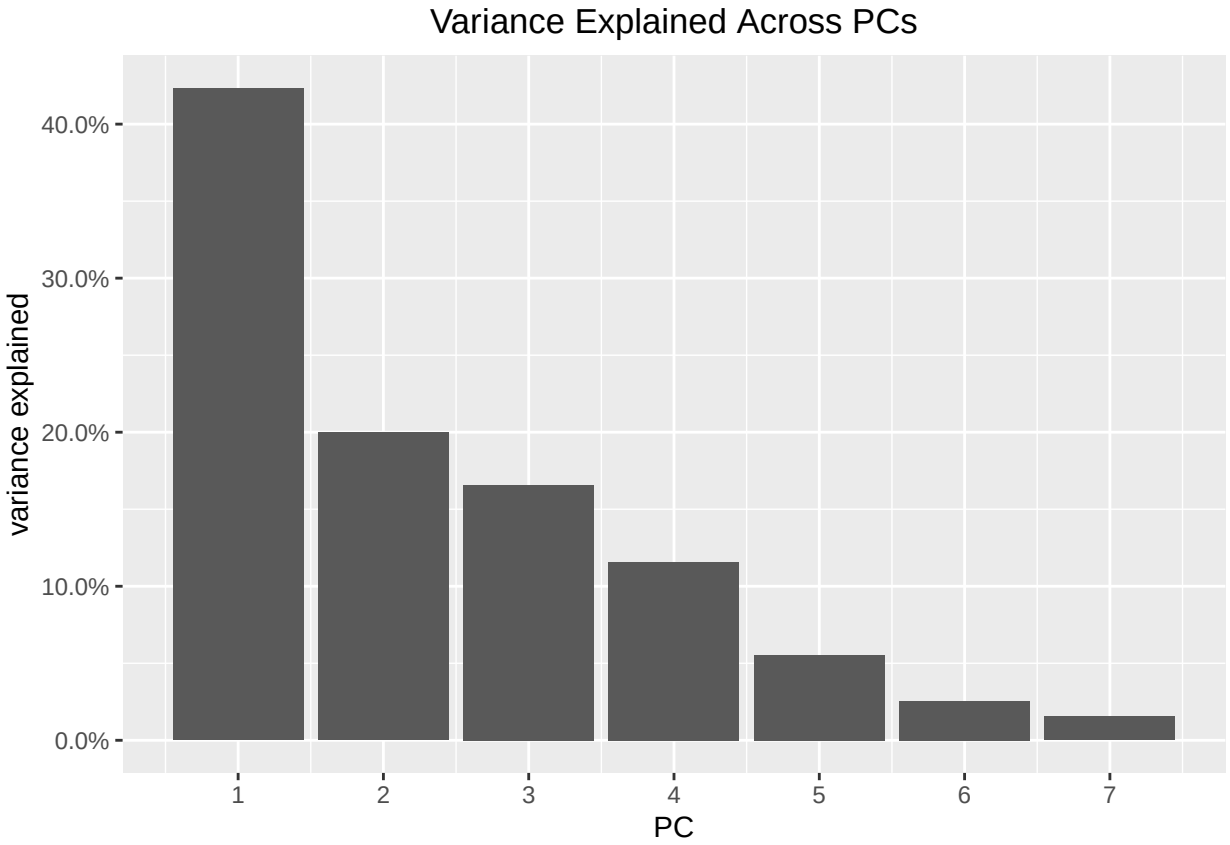

```
p13 <- pca_fit %>%
  tidy(matrix = "rotation") %>%
  pivot_wider(
  names_from = "PC", values_from = "value",
  names_prefix = "PC"
  ) %>%
  ggplot(aes(PC1, PC3)) +
  geom_segment(
    xend = 0, yend = 0,
    arrow = arrow_style
  ) +
  geom_text_repel(
    aes(label = column),
    max.overlaps = Inf
  ) +
  xlim(-1.5, 1) + ylim(-1, 1) +
  coord_fixed()

p13
```




```
p23 <- pca_fit %>%
  tidy(matrix = "rotation") %>%
  pivot_wider(
  names_from = "PC", values_from = "value",
  names_prefix = "PC"
  ) %>%
  ggplot(aes(PC2, PC3)) +
  geom_segment(
    xend = 0, yend = 0,
    arrow = arrow_style
  ) +
  geom_text_repel(
    aes(label = column),
    max.overlaps = Inf
  ) +
  xlim(-1.5, 1) + ylim(-1, 1) +
  coord_fixed()

p23
```




```
# plot the variance explained
pca_fit %>%
  tidy(matrix = "eigenvalues") %>%
  ggplot(aes(PC, percent)) +
  geom_col() +
  ggtitle("Variance Explained Across PCs") +
  theme(plot.title = element_text(hjust = 0.5)) +
  scale_x_continuous(
    breaks = 1:8
  ) +
  scale_y_continuous(
    name = "variance explained",
    label = scales::label_percent(accuracy = 0.5)
  )
```



Discussion: Based on the correlation heatmap, there are strong positive associations between **proposed_length** and **cost_usd**, and **proposed_length** and **stations**; the correlation values for these two pairs are 0.7 and 0.8, respectively. This is understandable because a larger number for a proposed length would require more stations for the line and cost more money to construct. There appear to be no strong inverse relationships between variables, since none of the correlation values have very negative values. Interestingly, **project_duration** does not seem to have a notable relationship with any other variable; this may be due to the fact that there are various combinations of project statuses that could potentially counter/cancel out effects. For example, some shorter lines may be incomplete and longer lines may be completed, some countries have fewer projects but spend a high amount per kilometer used and others may have more projects but be more cost-efficient, etc.

With the second part of the analysis, the PCA method extracted 7 PCs. Based on the variance explained graph, PC1 explains a little more than 40% of the total variance and PC2 about 20%. The first rotation matrix graph, with PC1 and PC2, has all arrows facing away from the two axes, not really indicating any strong relationships. The second graph, between PC1 and PC3, has the arrows for **project-duration** and **cost_per_km_usd** aimed towards the "x-axis" for PC1. The last rotation matrix for PC2 and PC3 has the two cost variables and **project_duration** aimed towards PC2, while **stations** and **proposed_length** are slightly more inclined towards PC3.