2018 MERL Tech DC workshop: Up your Data Game with R

This tutorial was developed for a workshop at MERL Tech DC in 2018 and presented with Jonathan Seiden. You can find the R code and slides on [Github] (https://github.com/cguedenet/MERL-Tech-workshop.git).

This is an R Markdown Notebook. When you execute code within the notebook, the results appear beneath the code.

R Studio is an "Integrated Development Environment", or IDE. This means it is a front-end for R that makes it much easier to work with. R Studio is also free, and available for Windows, Mac, and Linux platforms.

Cleaning your data

Analyzing survey data typically starts with cleaning, recoding, and restructuring, or even joining data sets. For example, you may want to know how many missing cases there are or how many people responded to each question. Or you may want to group certain continuous variables like ages or income into ranges. Lastly, you may want to find and deal with outliers.

Functions like distinct() cab quickly get rid of duplicate rows across all variables or for specific variables (e.g. a person's ID or email address)

Functions like mutate() can help create new variables. For example, creating a new variable for age ranges based on an existing variable for age.

```
# remove duplicated rows (if any). The distinct() function keeps only unique rows.
df <- distinct(df)

# You can also choose to remove duplicated rows for specific variables like ID.x and ID.y
df <- distinct(df, ID.x, ID.y, .keep_all = TRUE)

# Group responses and create a new variable using mutate function (dplyr package)
#this creates a new variable that recodes the Age variable into 5 age categories

df <- df %>% mutate(AgeCut = cut(Age, c(10,29,34,50,65,100)))
```

Overview of survey data and basic analysis

When you're analyzing survey data, one of the first things you need to do is get an overview of your data. For example, you may want to know basic stats for continuous variables or frequency tables for other types of data.

Get summary stats for Age and AgeCut variables

```
library(formattable)
summary(df$Age) %>% formattable()
```

Min. 1st Qu. Median Mean 3rd Qu. Max. NA's 10.00 23.00 27.00 29.18 33.00 86.00 2007

```
summary(df$AgeCut) %>% formattable()
```

(10,29] (29,34] (34,50] (50,65] (65,100] NA's 8240 2380 2564 400 27 2009

add new stats

```
df %>% summarise(
  mean = mean(Age, na.rm = TRUE),
  median = median(Age, na.rm = TRUE),
  IQR = IQR(Age, na.rm = TRUE),
  n = n()
  ) %>%
  kable()
```

mean	median	IQR	n
29.17542	27	10	15620

add a grouping variable

```
df %>%
  group_by(Gender) %>%
  summarise(
    mean = mean(Age, na.rm = TRUE),
    median = median(Age, na.rm = TRUE),
    IQR = IQR(Age, na.rm = TRUE), n = n()
    ) %>%
  kable()
```

Gender	mean	median	IQR	n
agender	25.27778	25.5	7.25	38
female	30.68599	28.0	11.00	2840
genderqueer	28.21538	27.0	9.00	66
male	28.81574	27.0	10.00	10766
trans	30.00000	27.0	10.25	36
NA	24.47500	23.5	7.00	1874

INSTRUCTIONS:

1. Get summary statistics for another continuous variable, like income, MonthsProgramming, ExpectedEarning, MoneyForLearning, etc. (reference the df object)
2. Add or change the summary stats you want to calculate (Other useful functions: mean, median, sd, IQR, min, max, quantile, first, last, nth, n, n_distinct)
3. Add a new grouping variable

val	frq	label	raw.prc	valid.prc	cum.prc
associate's degree	649	<none>	4.15	4.74	4.74
bachelor's degree	5644	<none>	36.13	41.26	46.00
high school diploma or equivalent (GED)	1356	<none>	8.68	9.91	55.92
master's degree (non-professional)	1445	<none>	9.25	10.56	66.48
no high school (secondary school)	258	<none>	1.65	1.89	68.37
Ph.D.	160	<none>	1.02	1.17	69.54
professional degree (MBA, MD, JD, etc.)	692	<none>	4.43	5.06	74.60
some college credit, no degree	2268	<none>	14.52	16.58	91.18
some high school	764	<none>	4.89	5.59	96.76
trade, technical, or vocational training	443	<none>	2.84	3.24	100.00
NA	1941	<none>	12.43	NA	NA

Here's another way to quickly generate summary statistics for select numeric variables using the psyche package

```
# Here's another way to quickly generate summary statistics for select numeric variables using the psyc

df %>% select(c("Age", "Income", "ExpectedEarning", "HoursLearning", "Gender"))%>%
  describeBy() %>% kable()
```

	vars	n	mean	sd	median	trimmed	mad	min	max
Age	1	13613	29.175421	9.017717e+00	27	28.174915	7.4130	10	86
Income	2	7329	44930.010506	3.558278e+04	37000	39292.514748	26983.3200	6000	200000
ExpectedEarning	3	6077	53717.586967	3.007851e+04	50000	51741.349784	26686.8000	6000	200000
HoursLearning	4	14942	15.323317	1.427487e+01	10	12.846077	8.8956	0	100
Gender*	5	13746	3.576313	8.241731e-01	4	3.720495	0.0000	1	5

Working with Categorical data

summarize categorical data by creating frequency tables using sjmisc package

Frequency table for the variable: "SchoolDegree"

```
# summarize categorical data by creating frequency tables using sjmisc package
df %>% frq(SchoolDegree) %>% kable()
```

create cross-tabulations with two or more variables

```
df %>% flat_table(SchoolDegree, Gender) %>%
  data.frame() %>%
  spread(Gender,Freq) %>%
  kable()
```

SchoolDegree	agender	female	genderqueer	male	trans
associate's degree	1	111	2	534	1
bachelor's degree	10	1336	30	4229	10
high school diploma or equivalent (GED)	6	165	4	1172	3

SchoolDegree	agender	female	genderqueer	male	trans
master's degree (non-professional)	2	416	6	1010	2
no high school (secondary school)	3	17	3	231	0
Ph.D.	1	41	3	113	2
professional degree (MBA, MD, JD, etc.)	0	183	3	500	2
some college credit, no degree	9	392	9	1837	12
some high school	4	85	4	660	2
trade, technical, or vocational training	1	68	2	370	2

create marginal tables using "row", "col", or "cell"

```
df %>% flat_table(SchoolDegree, Gender, margin = "col") %>%
  data.frame() %>%
  spread(Gender,Freq) %>%
  kable()
```

SchoolDegree	agender	female	genderqueer	male	trans
associate's degree	2.70	3.94	3.03	5.01	2.78
bachelor's degree	27.03	47.48	45.45	39.69	27.78
high school diploma or equivalent (GED)	16.22	5.86	6.06	11.00	8.33
master's degree (non-professional)	5.41	14.78	9.09	9.48	5.56
no high school (secondary school)	8.11	0.60	4.55	2.17	0.00
Ph.D.	2.70	1.46	4.55	1.06	5.56
professional degree (MBA, MD, JD, etc.)	0.00	6.50	4.55	4.69	5.56
some college credit, no degree	24.32	13.93	13.64	17.24	33.33
some high school	10.81	3.02	6.06	6.19	5.56
trade, technical, or vocational training	2.70	2.42	3.03	3.47	5.56

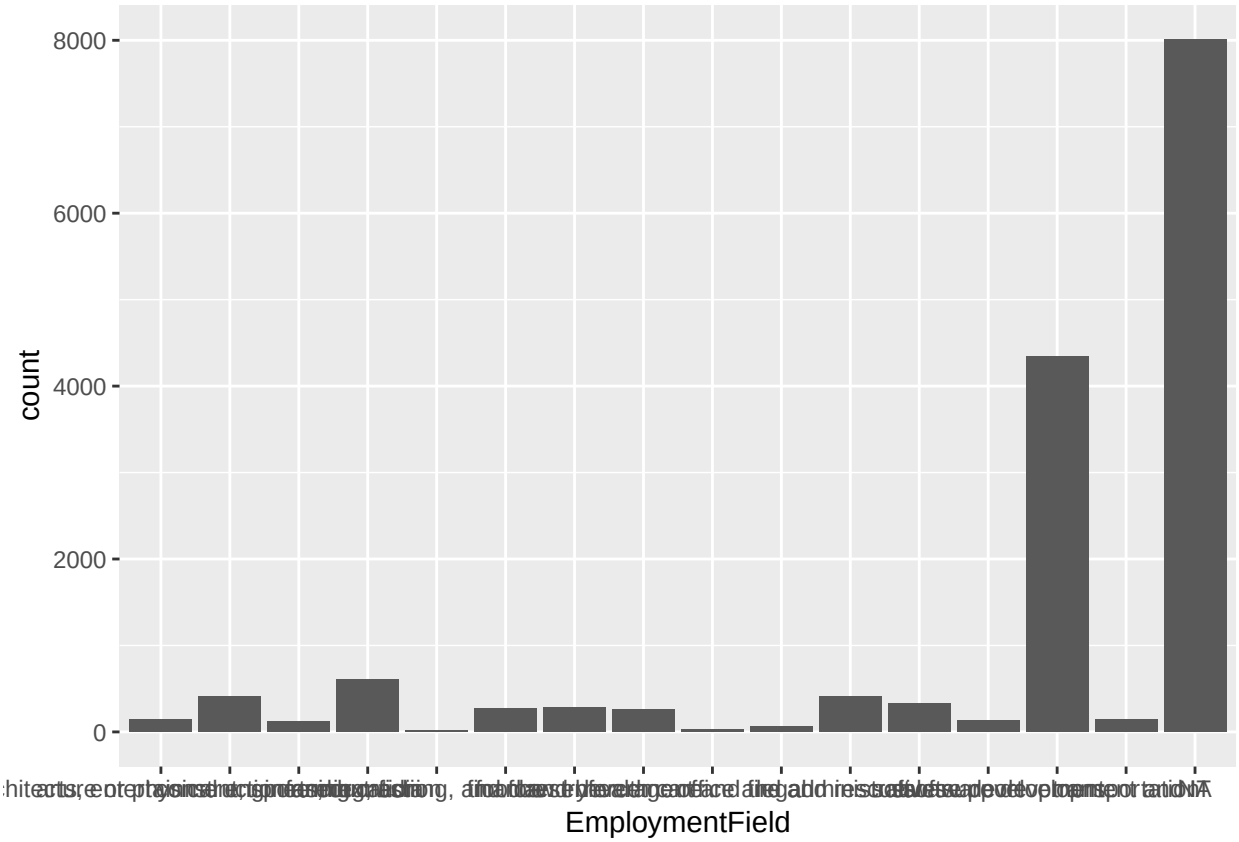
INSTRUCTIONS:

1. Create other cross-tabulations by changing the variables
2. try changing the margin argument to col, row, or cell

Creating charts in R using ggplot2 package

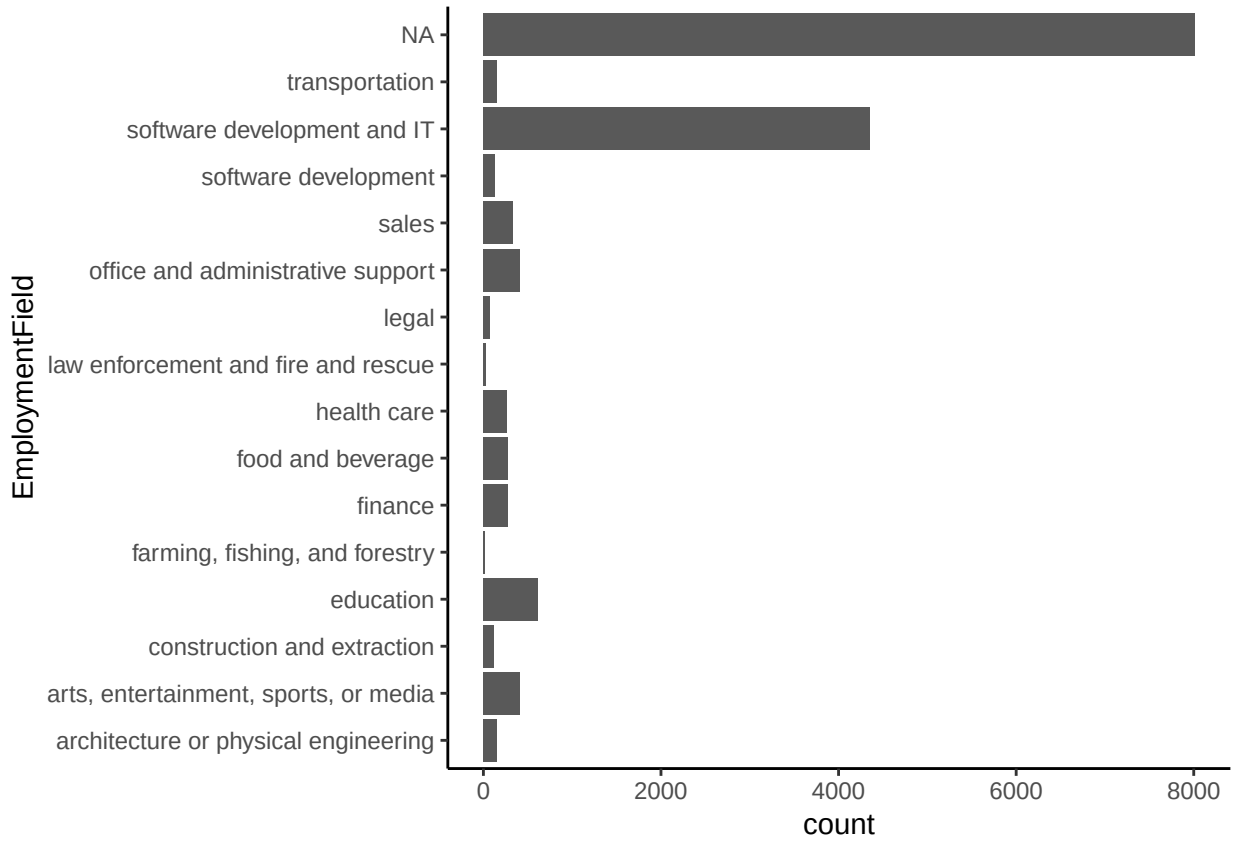
create a simple column chart

```
df %>% ggplot(aes(EmploymentField)) + geom_bar()
```



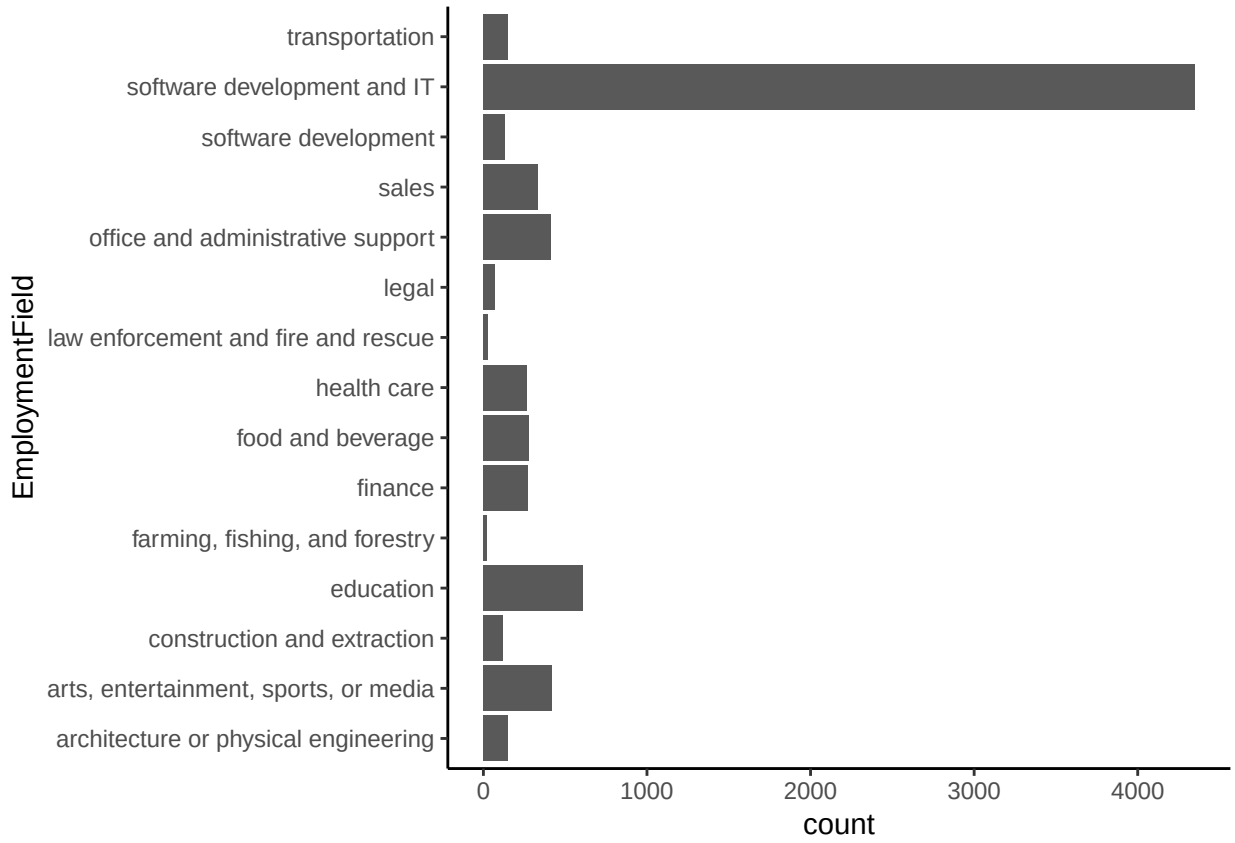
flip it so that it becomes a bar chart and the labels are easier to read. Plus, add a theme

```
df %>% ggplot(aes(EmploymentField)) +
  geom_bar() +
  coord_flip() +
  theme_classic()
```



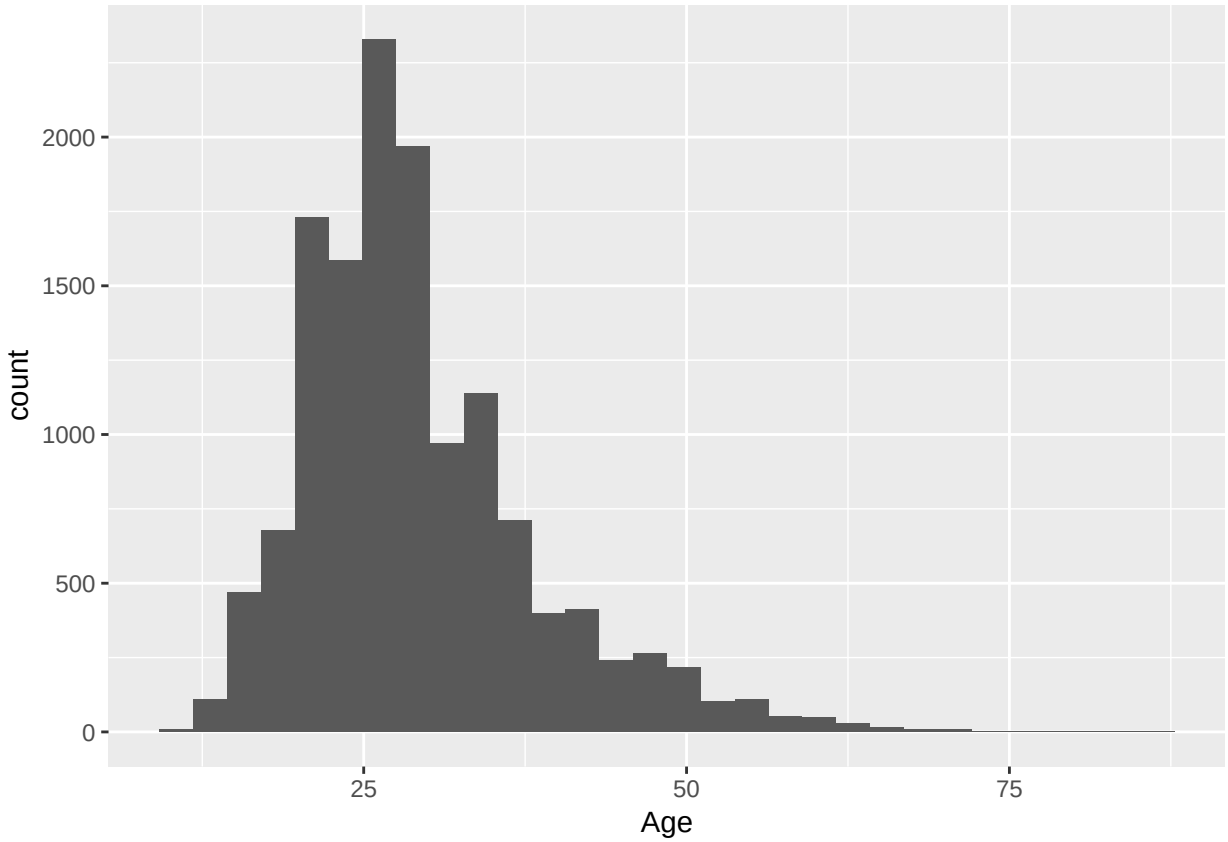
Get rid of NA

```
df %>% filter(EmploymentField!="") %>%
  ggplot(aes(EmploymentField)) +
  geom_bar() +
  coord_flip() +
  theme_classic()
```



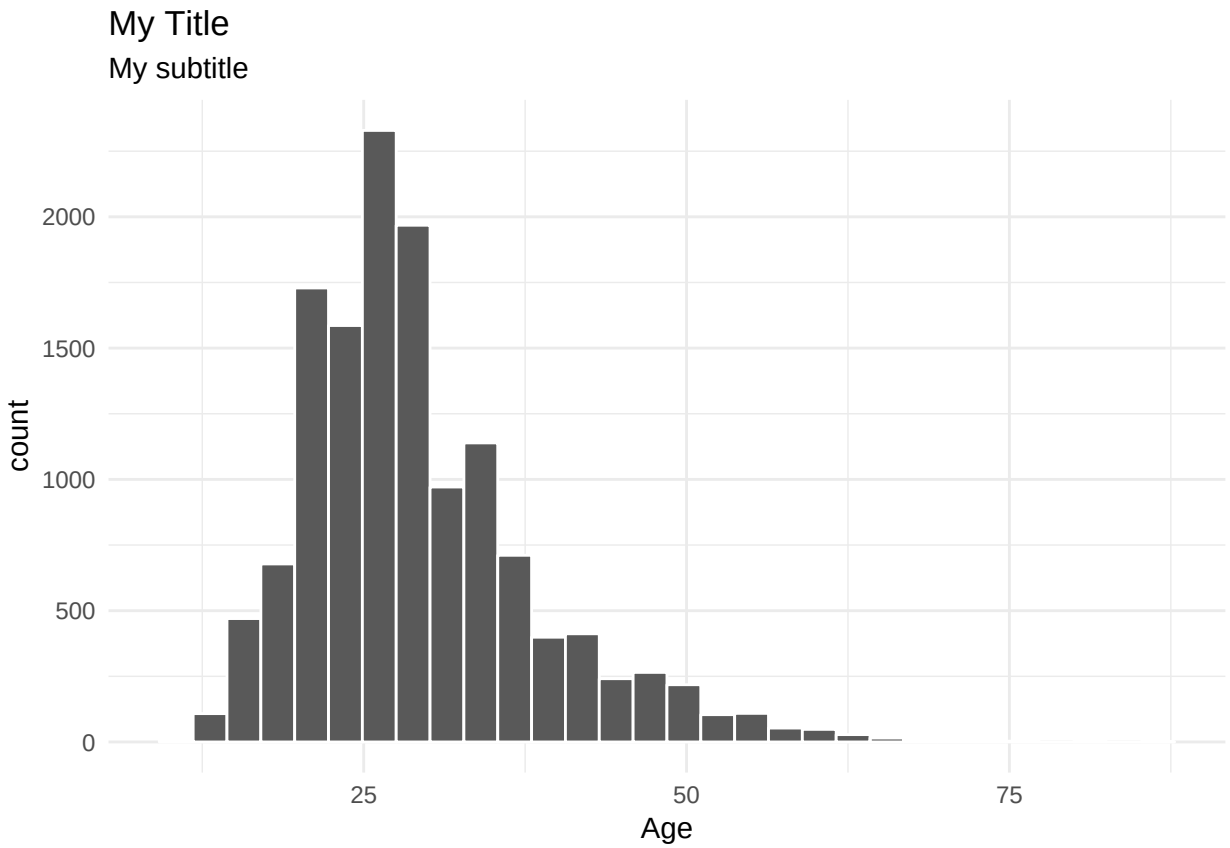
Create a simple histogram

```
df %>% ggplot(aes(x=Age)) +
  geom_histogram()
```



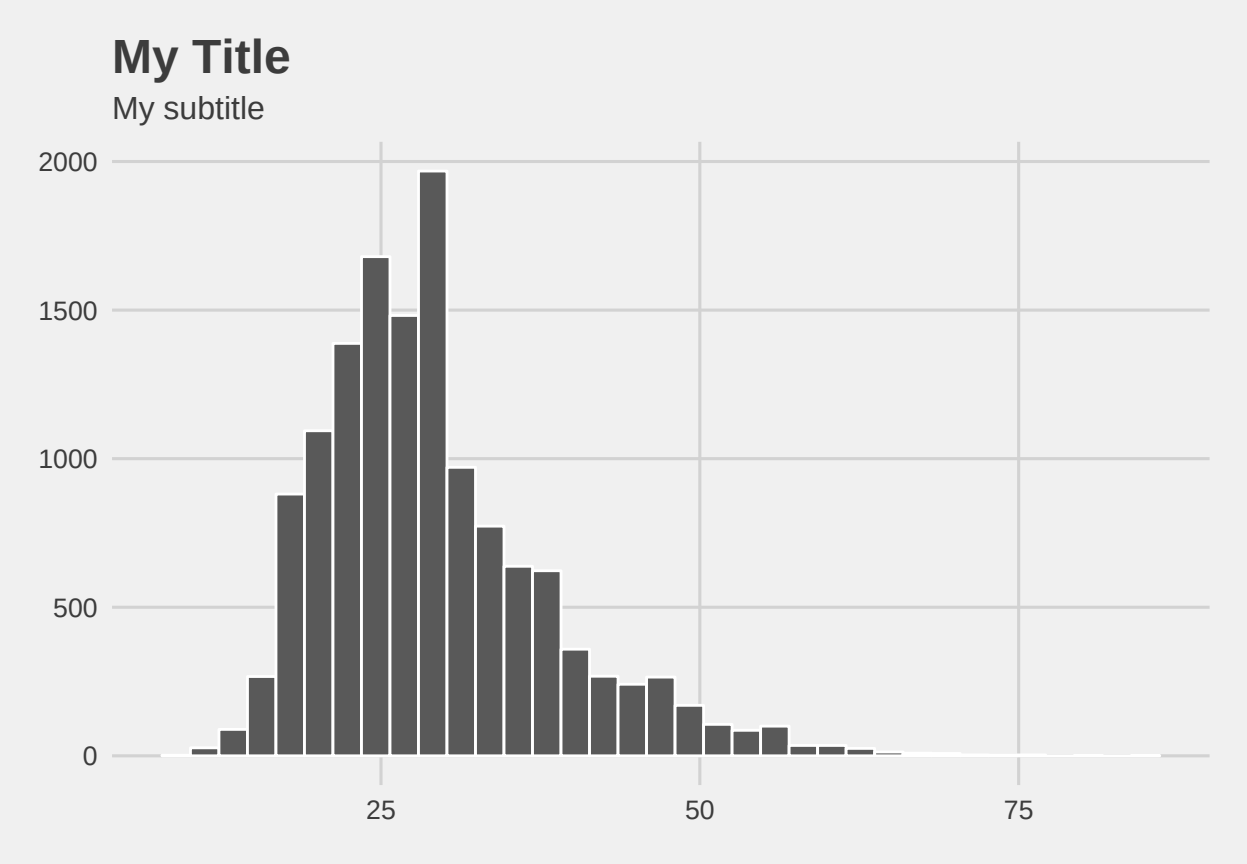
make it look pretty

```
df %>% ggplot(aes(x=Age)) +
  geom_histogram(color = "white") +
  theme_minimal() +
  labs(title = "My Title", subtitle = "My subtitle")
```



try another theme

```
df %>% ggplot(aes(x=Age)) +
  geom_histogram(color = "white", bins = 35) +
  theme_fivethirtyeight() +
  labs(title = "My Title", subtitle = "My subtitle")
```

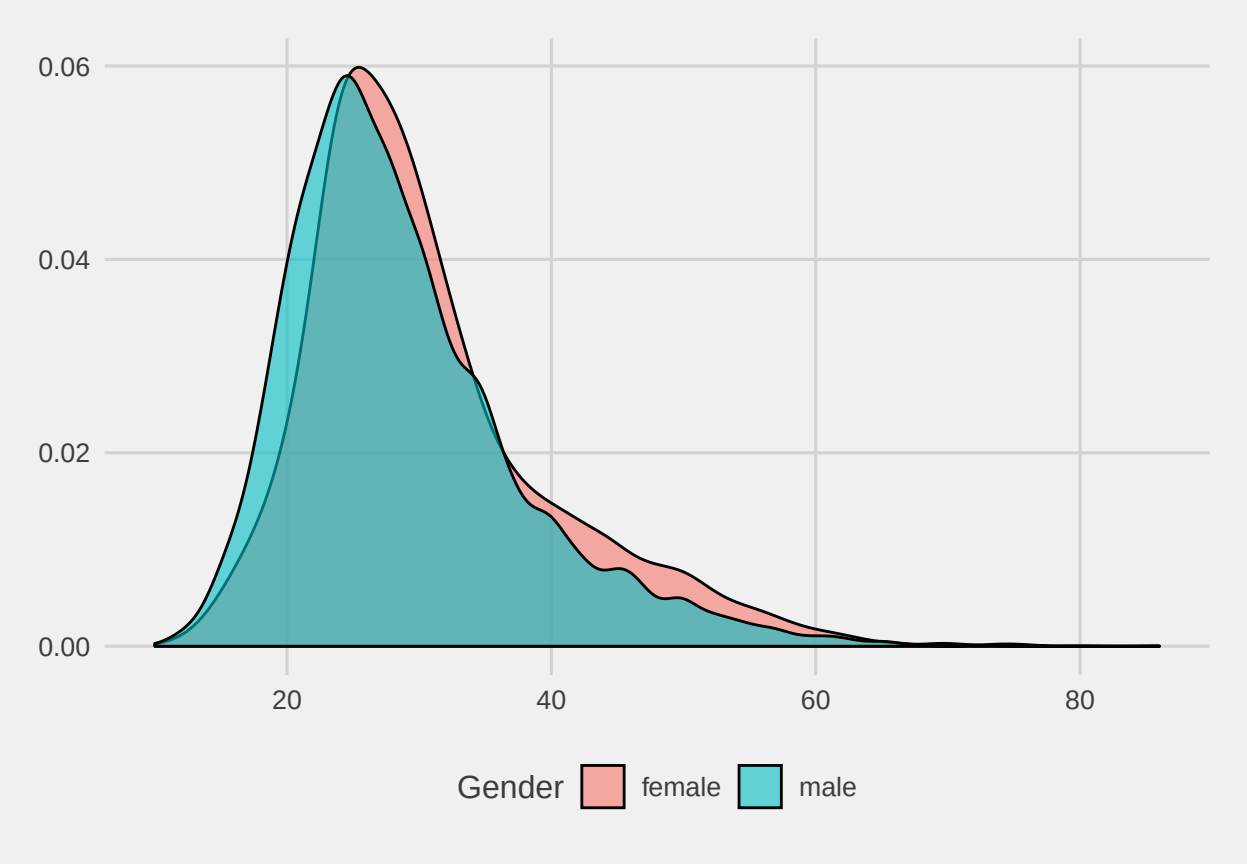


INSTRUCTIONS

1 Change the x variable with another numeric variable
2 Change the theme & titles
3 Change the number of bins

Create a density plot instead of a histogram

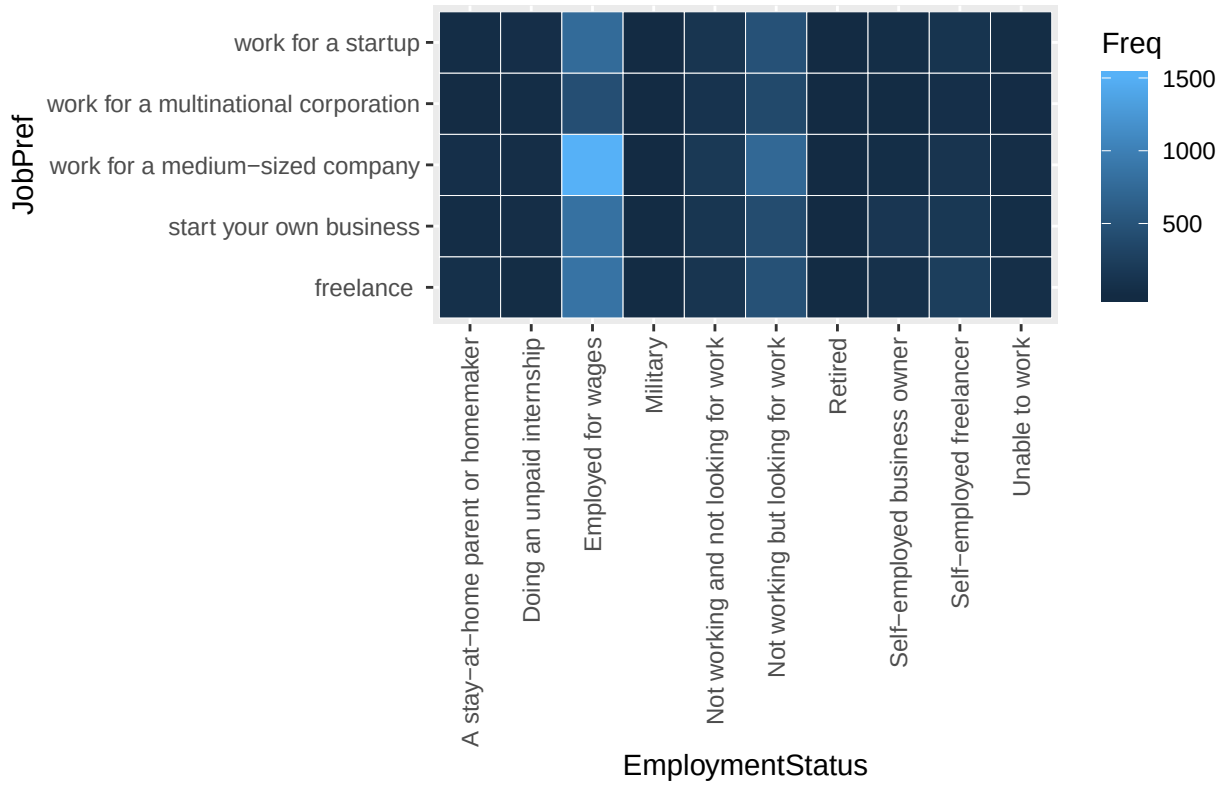
```
# Compare only males and females comparison
df %>% filter(Gender == "female" | Gender == "male") %>%
  ggplot(aes(x=Age, fill = Gender)) +
  geom_density(alpha = .6) +
  theme_fivethirtyeight()
```



Other examples of charts with the ggplot package

Relationship between employment status and job preference

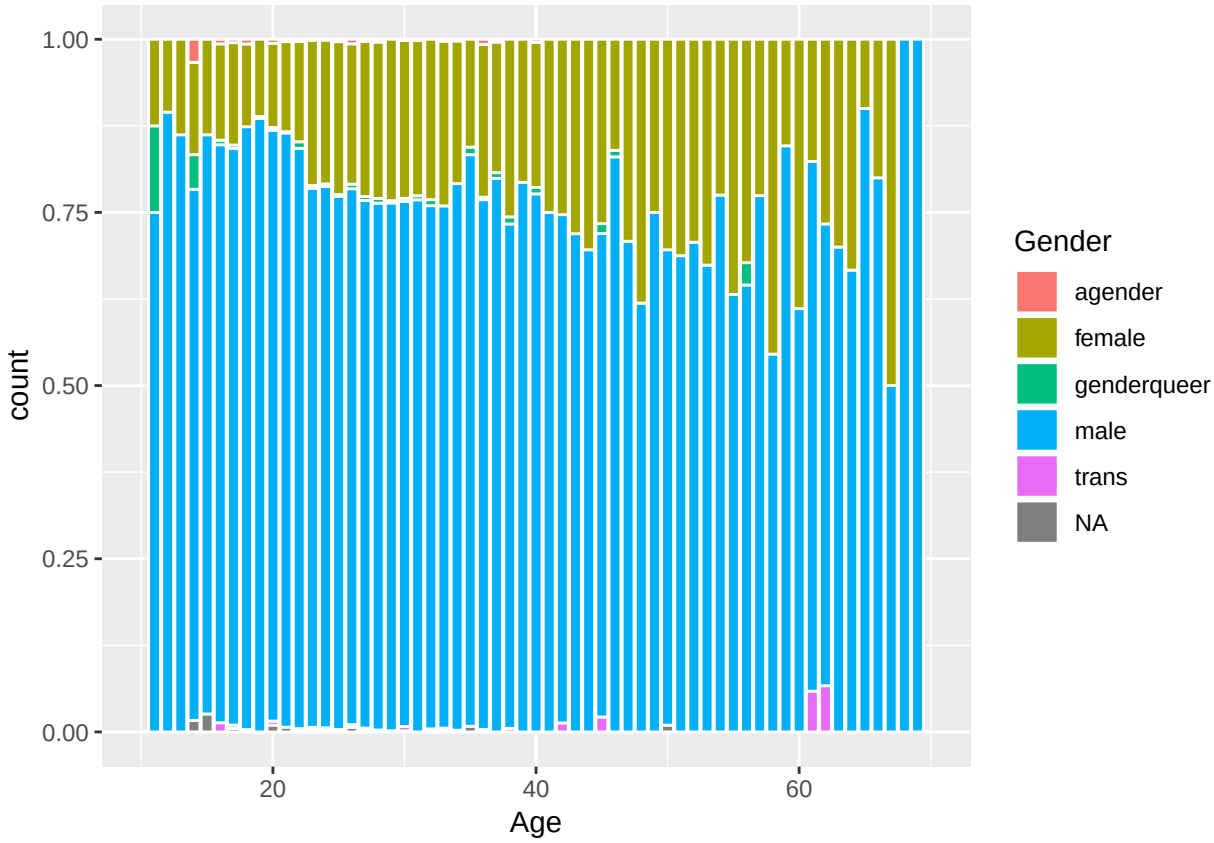
```
df %>% flat_table(EmploymentStatus, JobPref) %>% data.frame() %>%
  ggplot(aes(EmploymentStatus, JobPref)) +
  geom_tile(aes(fill = Freq), colour = "white") +
  scale_fill_continuous() +
  coord_fixed(ratio = 1) +
  theme(axis.text.x=element_text(angle=90,hjust=1,vjust=0.5))
```



Stacked bar chart

Comparing gender breakdown across ages

```
df %>% select(Age, Gender) %>%
  group_by(Age, Gender) %>%
  summarise(count=n()) %>%
  ggplot(aes(Age,count,fill=Gender)) +
  geom_bar(stat='identity', position='fill', color='white') +
  xlim(c(10,70))
```


1.00
0.75
0.50
0.25
0.00
count
20
40
60
Age
Gender
agender
female
genderqueer
male
trans
NA