Visualising numerical and categorical data

K. Arnold, based on datasciencebox.org

Terminology

Number of variables involved

- **Univariate** data analysis - distribution of single variable
- **Bivariate** data analysis - relationship between two variables
- **Multivariate** data analysis - relationship between many variables at once
  - usually focusing on bivariate relationships while conditioning for others

Types of variables

- **Numerical variables** can be classified as **continuous** or **discrete** based on whether or not the variable can take on an infinite number of values or only non-negative whole numbers, respectively.
- If the variable is **categorical**, we can determine if it is **ordinal** based on whether or not the levels have a natural ordering.

Data

Data: Lending Club

- Thousands of loans made through the Lending Club, which is a platform that allows individuals to lend to other individuals
- Not all loans are created equal -- ease of getting a loan depends on (apparent) ability to pay back the loan
- Data includes loans *made*, these are not loan applications

Take a peek at data

```
library(openintro)
glimpse(loans_full_schema)
```

```
## Rows: 10,000
## Columns: 55
## $ emp_title                    <chr> "global config enginee…
## $ emp_length                   <dbl> 3, 10, 3, 1, 10, NA, 1…
## $ state                        <fct> NJ, HI, WI, PA, CA, KY…
## $ homeownership                <fct> MORTGAGE, RENT, RENT, …
## $ annual_income                <dbl> 90000, 40000, 40000, 3…
## $ verified_income              <fct> Verified, Not Verified…
## $ debt_to_income               <dbl> 18.01, 5.04, 21.15, 10…
## $ annual_income_joint          <dbl> NA, NA, NA, NA, 57000,…
## $ verification_income_joint    <fct> , , , , Verified, , No…
## $ debt_to_income_joint         <dbl> NA, NA, NA, NA, 37.66,…
## $ delinq_2y                    <int> 0, 0, 0, 0, 0, 1, 0, 1…
## $ months_since_last_delinq     <int> 38, NA, 28, NA, NA, 3,…
## $ earliest_credit_line         <dbl> 2001, 1996, 2006, 2007…
## $ inquiries_last_12m           <int> 6, 1, 4, 0, 7, 6, 1, 1…
## $ total_credit_lines           <int> 28, 30, 31, 4, 22, 32,…
## $ open_credit_lines            <int> 10, 14, 10, 4, 16, 12,…
## $ total_credit_limit           <int> 70795, 28800, 24193, 2…
```

Selected variables

```
loans <- loans_full_schema %>%
  select(loan_amount, interest_rate, term, grade,
         state, annual_income, homeownership, debt_to_income)
glimpse(loans)
```

```
## Rows: 10,000
## Columns: 8
## $ loan_amount    <int> 28000, 5000, 2000, 21600, 23000, 5000, 2…
## $ interest_rate  <dbl> 14.07, 12.61, 17.09, 6.72, 14.07, 6.72, …
## $ term           <dbl> 60, 36, 36, 36, 36, 36, 60, 60, 36, 36, …
## $ grade          <ord> C, C, D, A, C, A, C, B, C, A, C, B, C, B…
## $ state          <fct> NJ, HI, WI, PA, CA, KY, MI, AZ, NV, IL, …
## $ annual_income  <dbl> 90000, 40000, 40000, 30000, 35000, 34000…
## $ homeownership  <fct> MORTGAGE, RENT, RENT, RENT, RENT, OWN, M…
## $ debt_to_income <dbl> 18.01, 5.04, 21.15, 10.16, 57.96, 6.46, …
```

```
loans %>% head() %>% knitr::kable()
```

loan_amount	interest_rate	term	grade	state	annual_income	homeownership	debt_to_income
28000	14.07	60	C	NJ	90000	MORTGAGE	18.01
5000	12.61	36	C	HI	40000	RENT	5.04
2000	17.09	36	D	WI	40000	RENT	21.15
21600	6.72	36	A	PA	30000	RENT	10.16
23000	14.07	36	C	CA	35000	RENT	57.96
5000	6.72	36	A	KY	34000	OWN	6.46

Selected variables

variable	description
loan_amount	Amount of the loan received, in US dollars
interest_rate	Interest rate on the loan, in an annual percentage
term	The length of the loan, which is always set as a whole number of months
grade	Loan grade, which takes a values A through G and represents the quality of the loan and its likelihood of being repaid
state	US state where the borrower resides
annual_income	Borrower's annual income, including any second income, in US dollars
homeownership	Indicates whether the person owns, owns but has a mortgage, or rents
debt_to_income	Debt-to-income ratio

Variable types

variable	type
loan_amount	numerical, continuous
interest_rate	numerical, continuous
term	numerical, discrete
grade	categorical, ordinal
state	categorical, not ordinal
annual_income	numerical, continuous
homeownership	categorical, not ordinal
debt_to_income	numerical, continuous

Visualizing numerical data

Describing shapes of numerical distributions

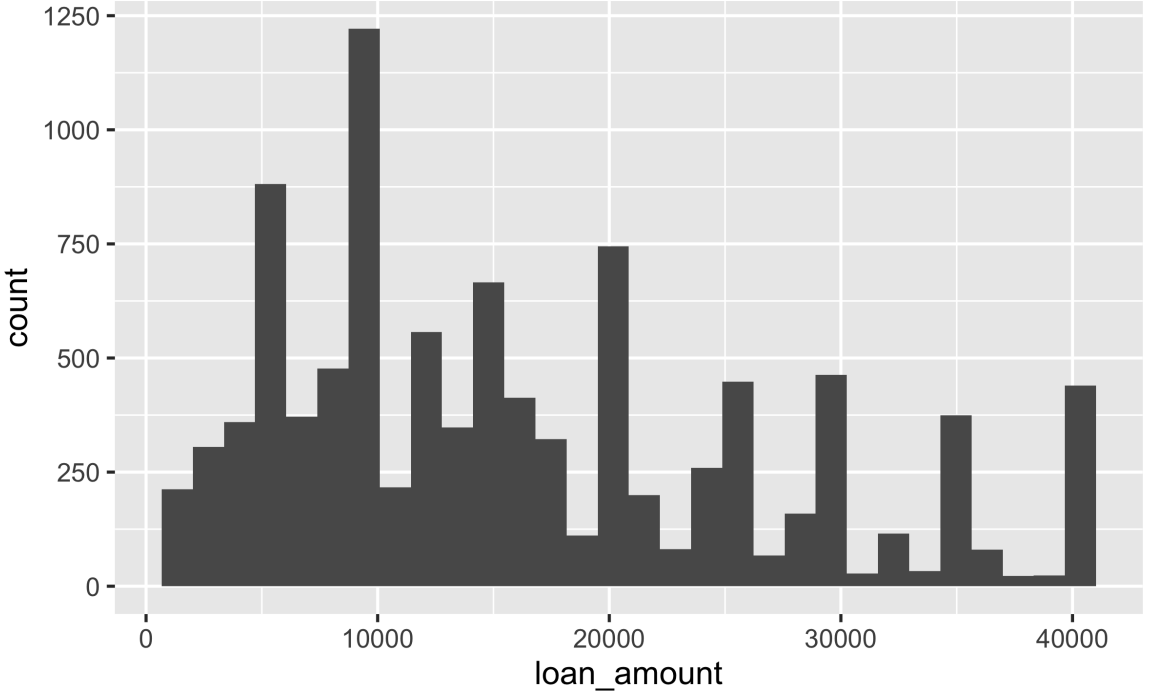
- center: mean (`mean`), median (`median`), mode (not always useful)
- spread: range (`range`), standard deviation (`sd`), inter-quartile range (`IQR`)
- shape:
  - skewness: right-skewed, left-skewed, symmetric (skew is to the side of the longer tail)
  - modality: unimodal, bimodal, multimodal, uniform
- unusual observations

Histogram

Histogram

```
ggplot(loans, aes(x = loan_amount)) +
  geom_histogram()
```

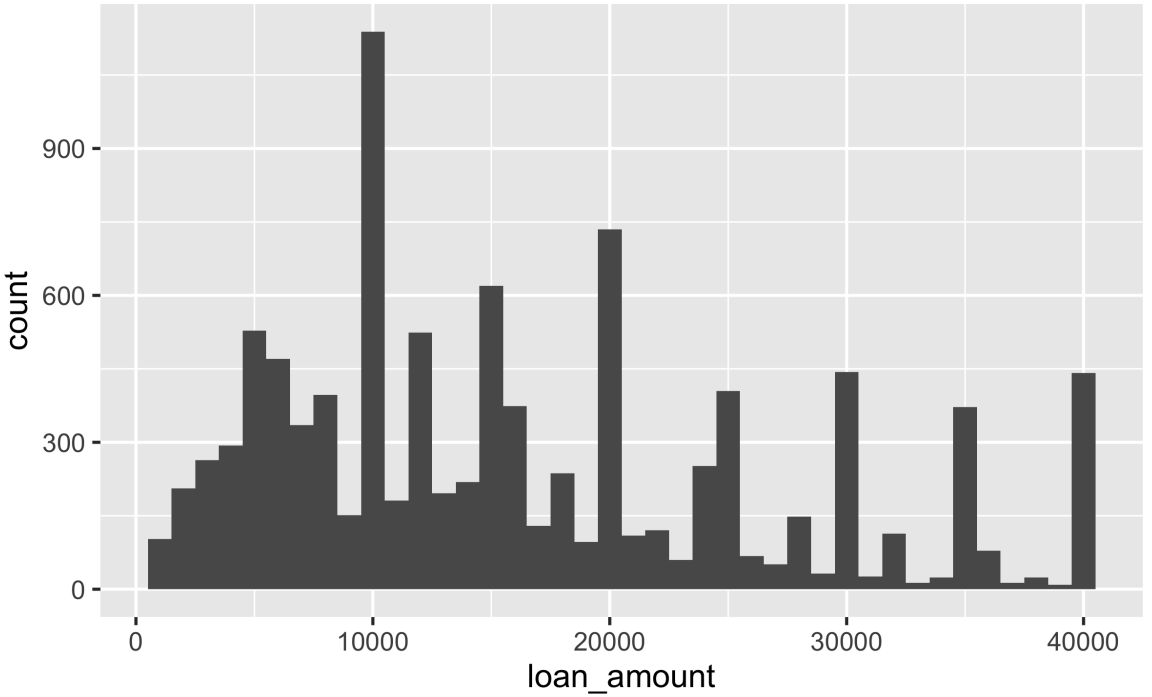
```
## `stat_bin()` using `bins = 30`. Pick better value with
## `binwidth`.
```



Histograms and binwidth

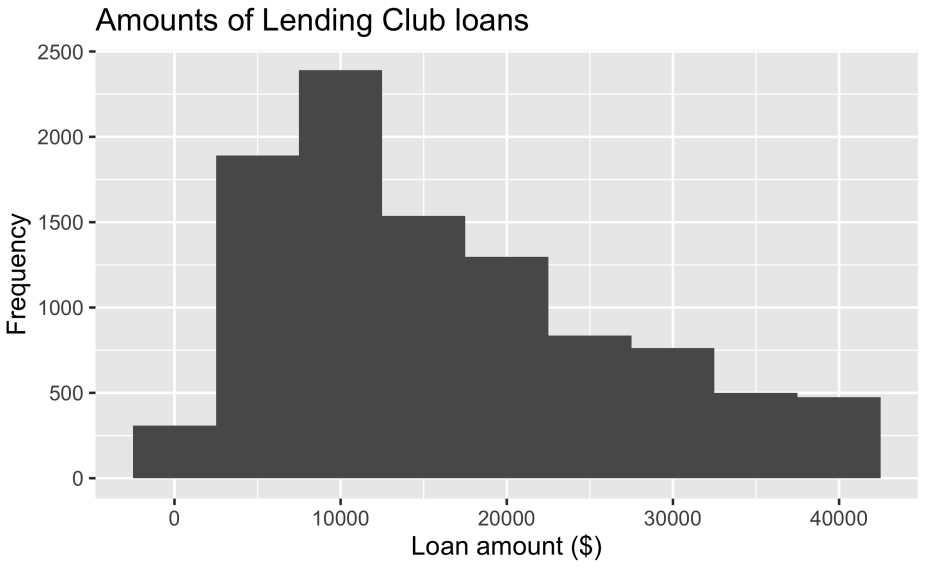
binwidth = 1000 | binwidth = 5000 | binwidth = 20000

```
ggplot(loans, aes(x = loan_amount)) +
  geom_histogram(binwidth = 1000)
```



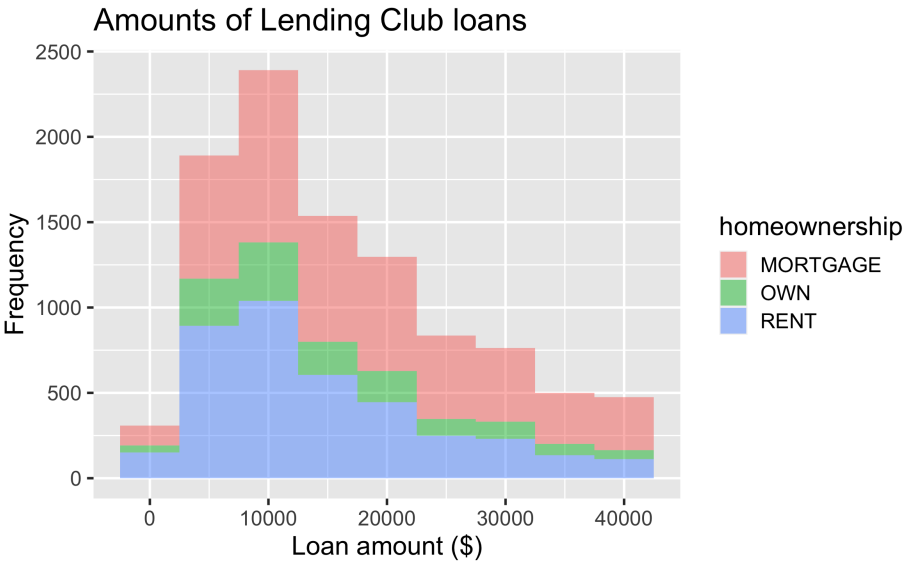
Customizing histograms

Plot Code



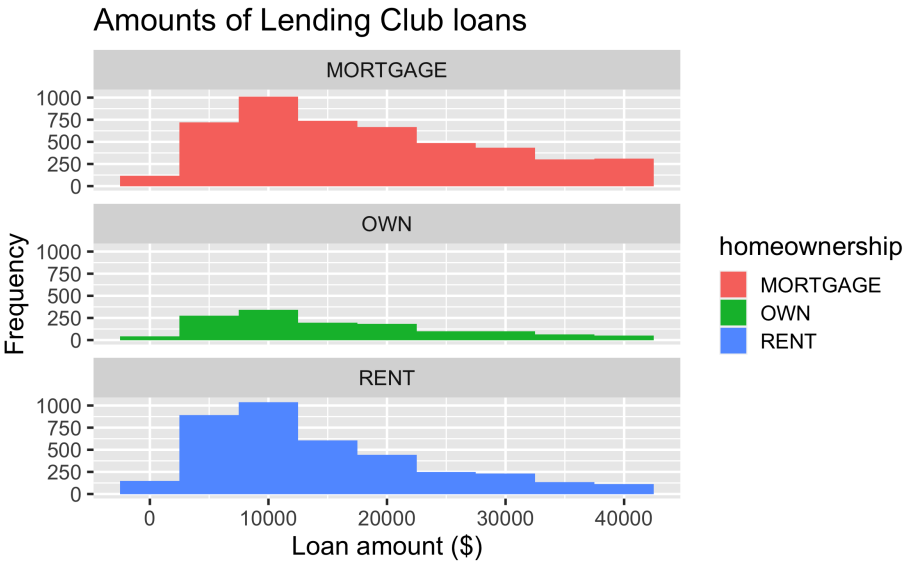
Fill with a categorical variable

Plot Code



Facet with a categorical variable

Plot Code



Density plot

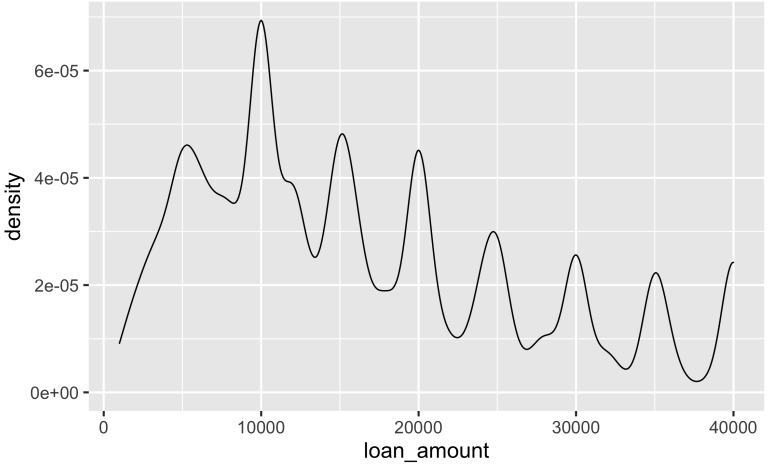
Density plot

```
ggplot(loans, aes(x = loan_amount)) +
  geom_density()
```

Density plots and adjusting bandwidth

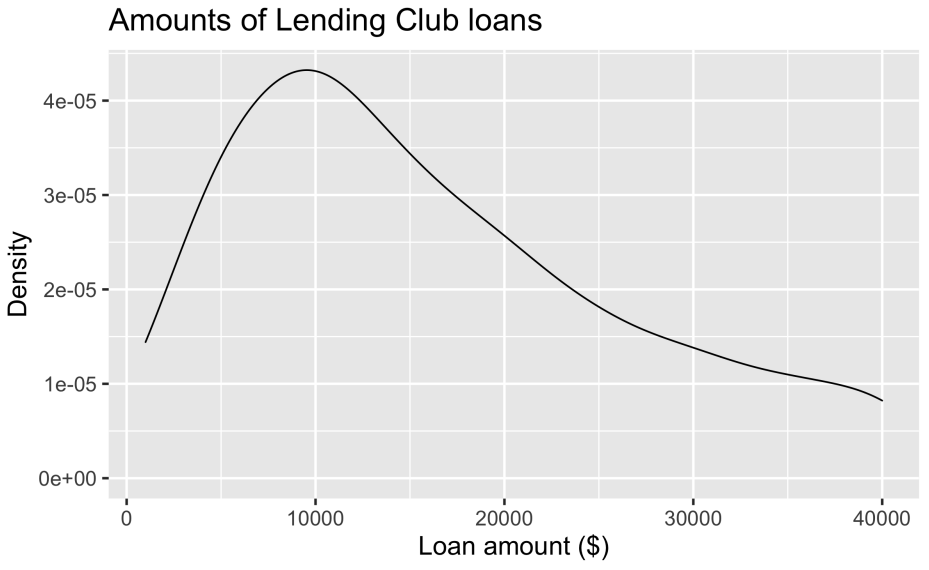
adjust = 0.5 | adjust = 1 | adjust = 2

```
ggplot(loans, aes(x = loan_amount)) +
  geom_density(adjust = 0.5)
```



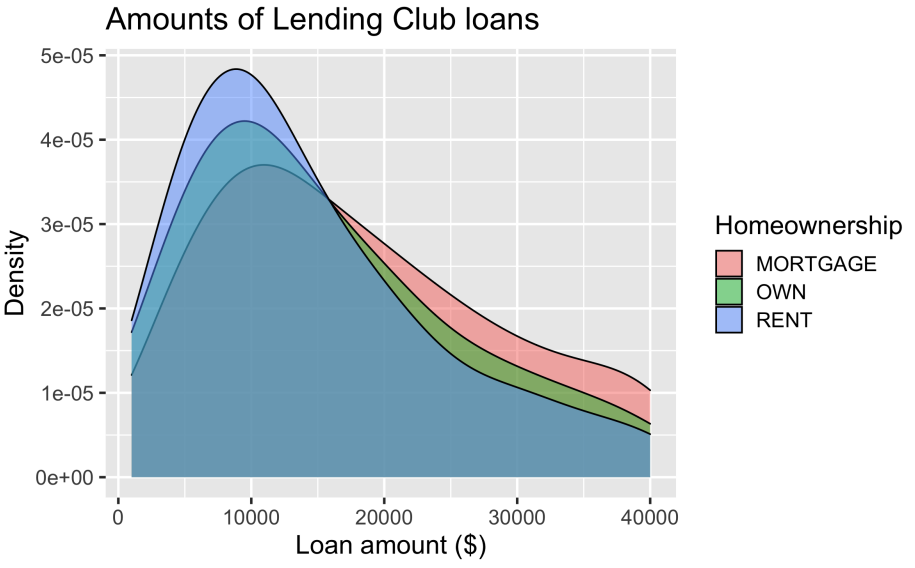
Customizing density plots

Plot Code



Adding a categorical variable

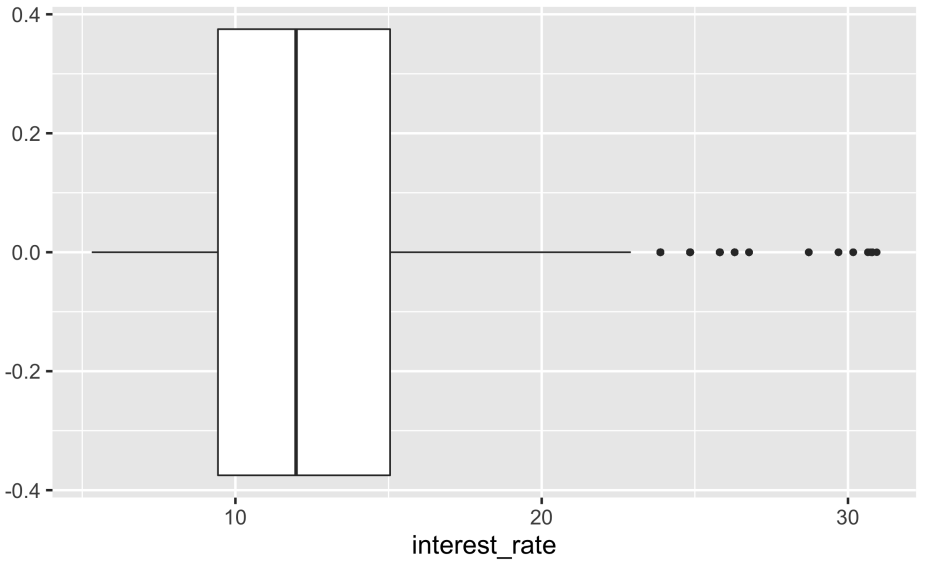
Plot Code



Box plot

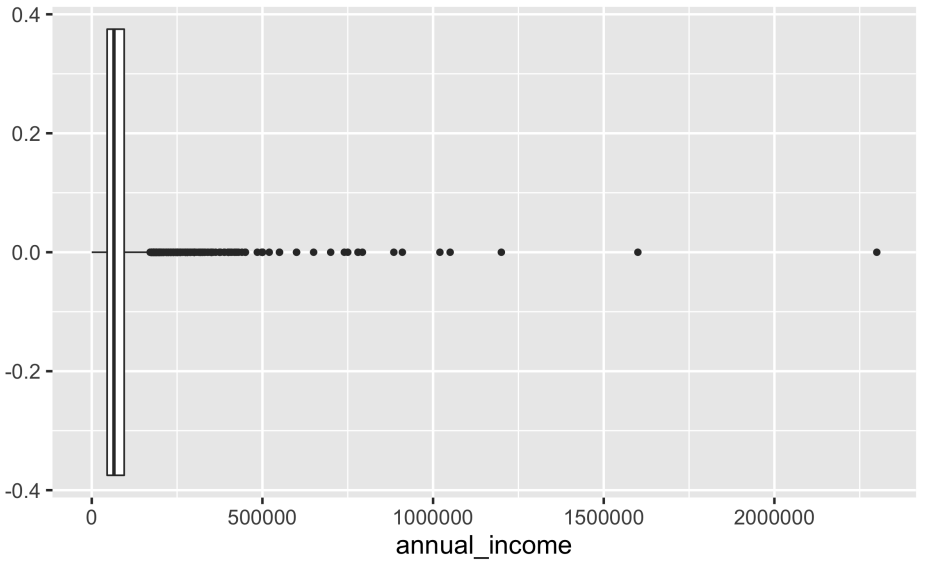
Box plot

```
ggplot(loans, aes(x = interest_rate)) +
  geom_boxplot()
```



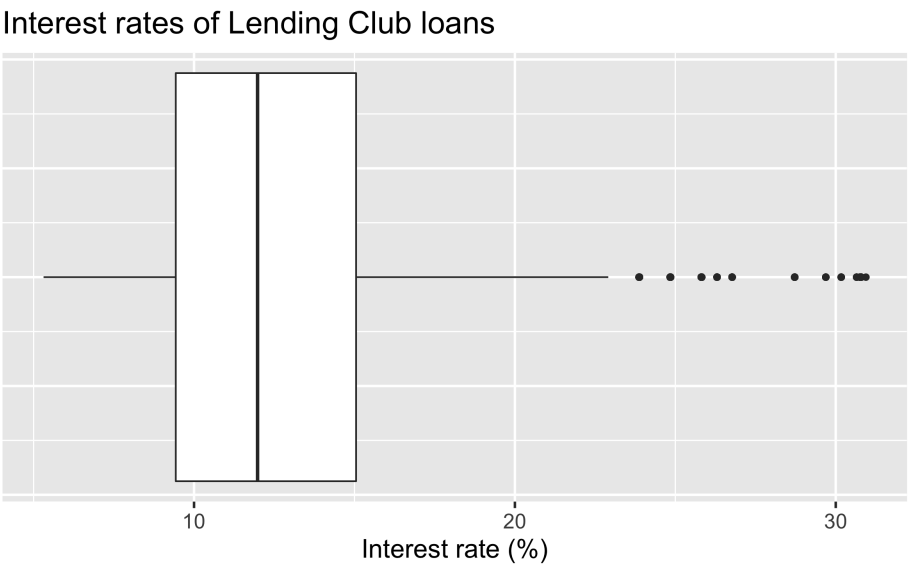
Box plot and outliers

```
ggplot(loans, aes(x = annual_income)) +
  geom_boxplot()
```



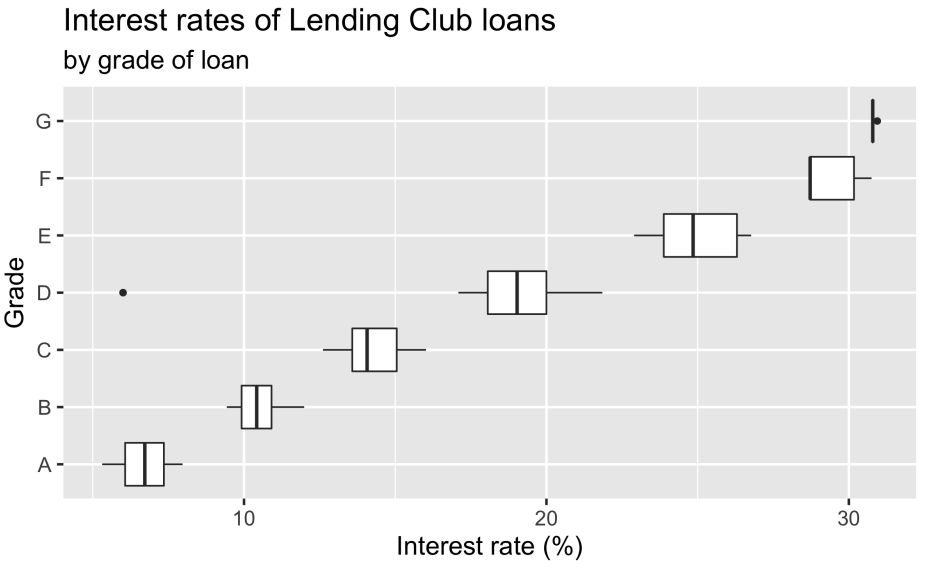
Customizing box plots

Plot Code



Adding a categorical variable

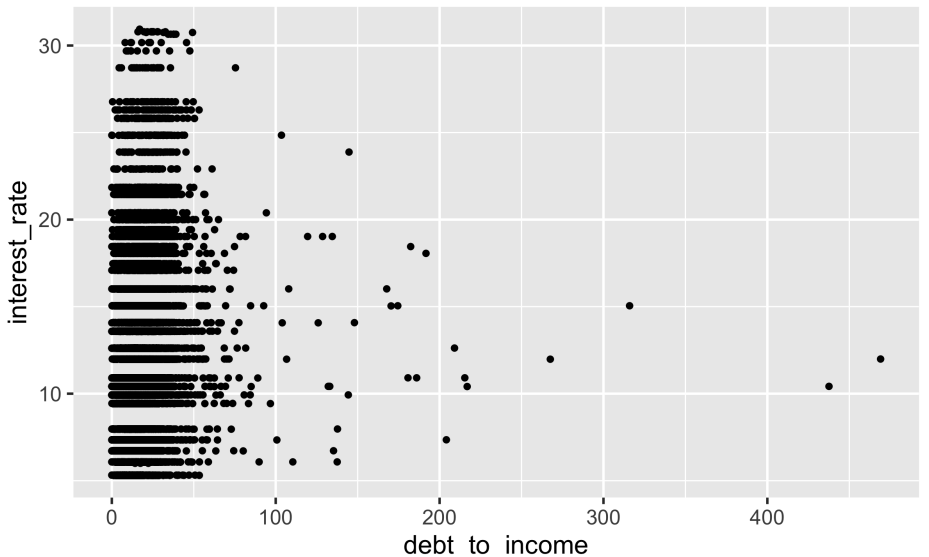
Plot Code



Relationships between numerical variables

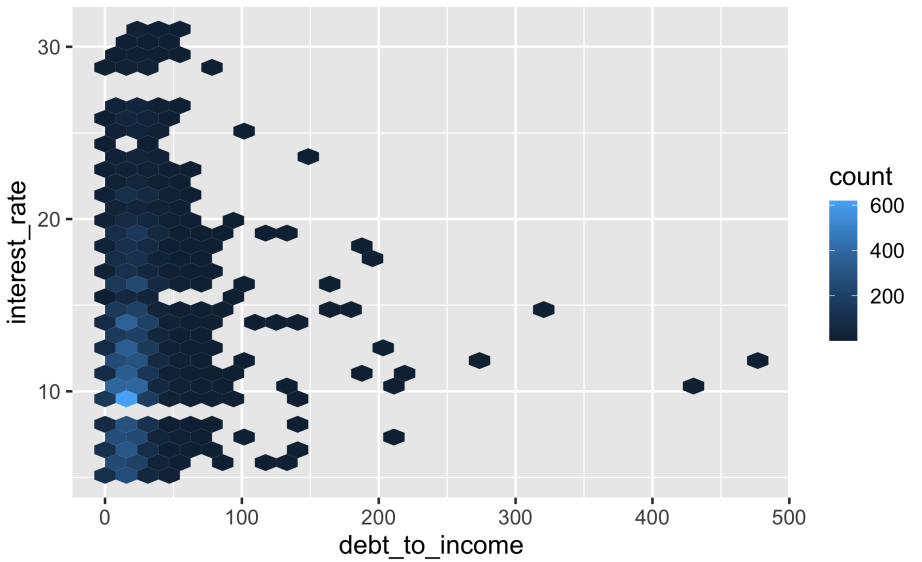
Scatterplot

```
ggplot(loans, aes(x = debt_to_income, y = interest_rate)) +
  geom_point()
```



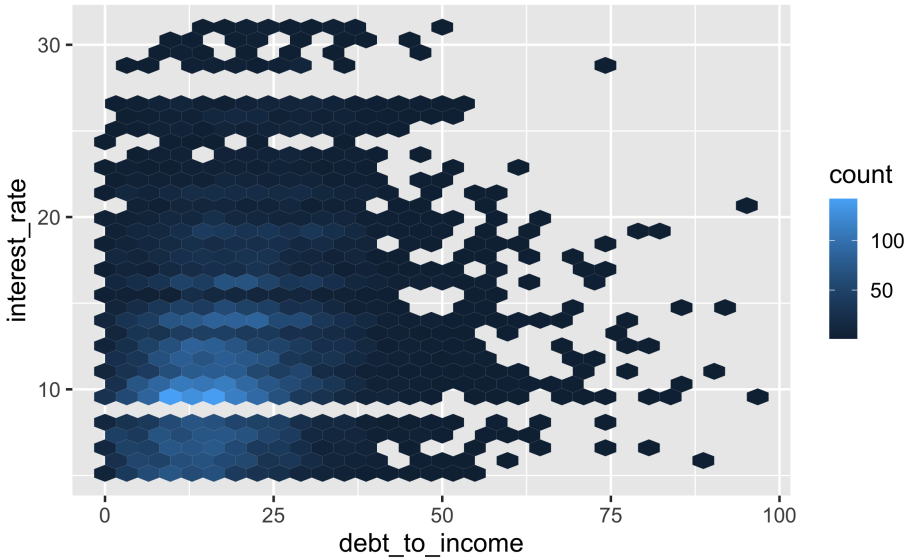
Hex plot

```
ggplot(loans, aes(x = debt_to_income, y = interest_rate)) +
  geom_hex()
```



Hex plot

```
ggplot(loans %>% filter(debt_to_income < 100),
       aes(x = debt_to_income, y = interest_rate)) +
  geom_hex()
```



Categorical Data

Which variables are *categorical*?

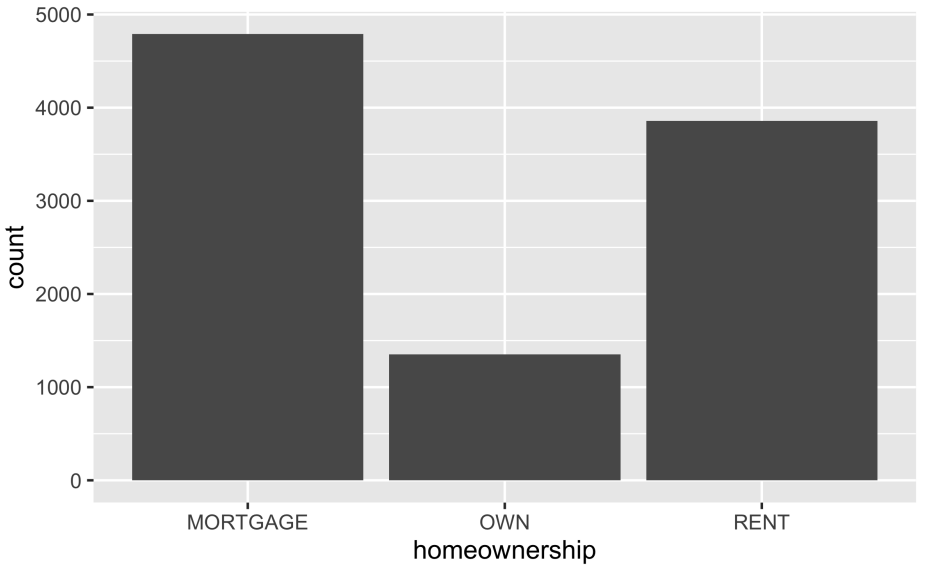
```
glimpse(loans)
```

```
## Rows: 10,000
## Columns: 8
## $ loan_amount    <int> 28000, 5000, 2000, 21600, 23000, 5000, 2…
## $ interest_rate  <dbl> 14.07, 12.61, 17.09, 6.72, 14.07, 6.72, …
## $ term           <dbl> 60, 36, 36, 36, 36, 36, 60, 60, 36, 36, …
## $ grade          <ord> C, C, D, A, C, A, C, B, C, A, C, B, C, B…
## $ state          <fct> NJ, HI, WI, PA, CA, KY, MI, AZ, NV, IL, …
## $ annual_income  <dbl> 90000, 40000, 40000, 30000, 35000, 34000…
## $ homeownership  <fct> MORTGAGE, RENT, RENT, RENT, RENT, OWN, M…
## $ debt_to_income <dbl> 18.01, 5.04, 21.15, 10.16, 57.96, 6.46, …
```

Bar plot

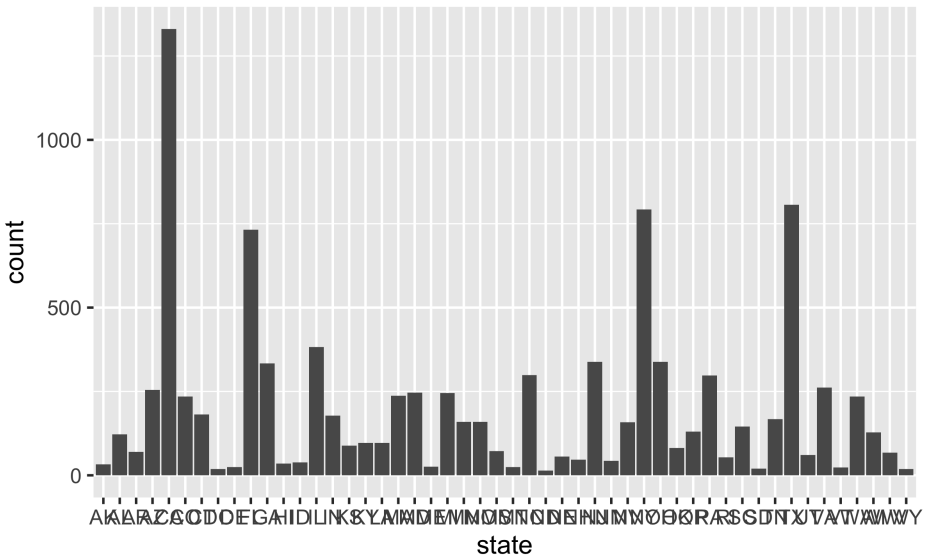
Bar plot

```
ggplot(loans, aes(x = homeownership)) +
  geom_bar()
```



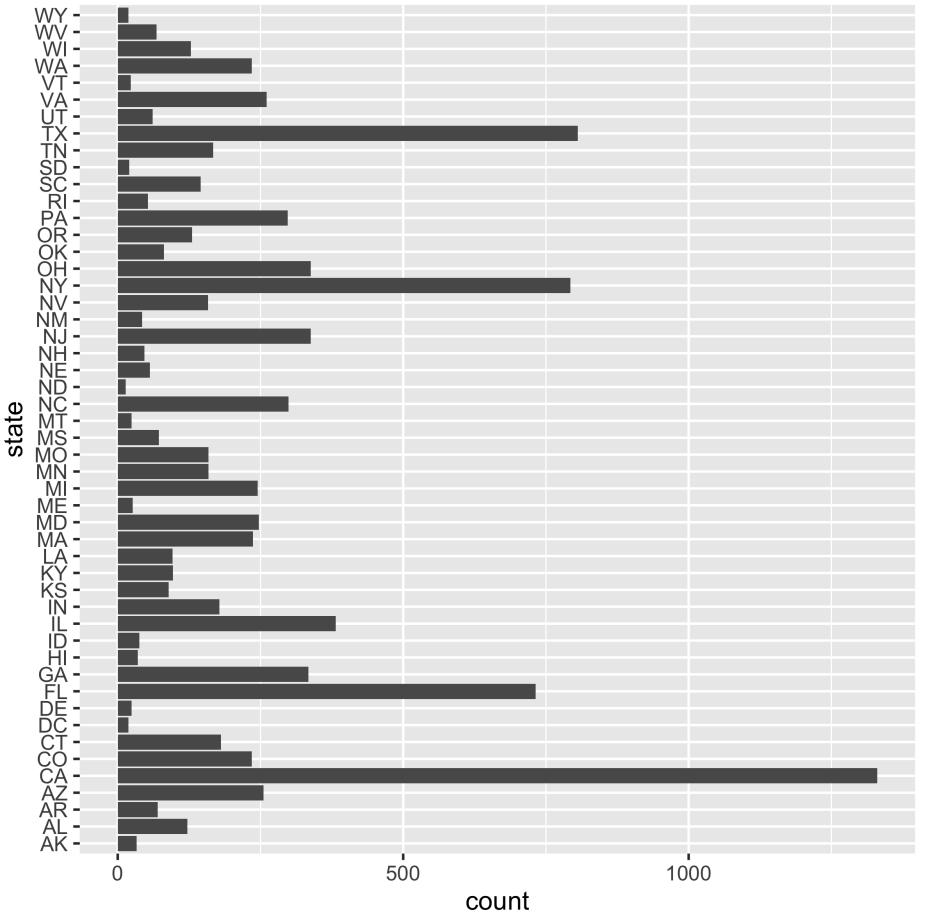
Bar plot with lots of categories

```
ggplot(loans, aes(x = state)) +
  geom_bar()
```



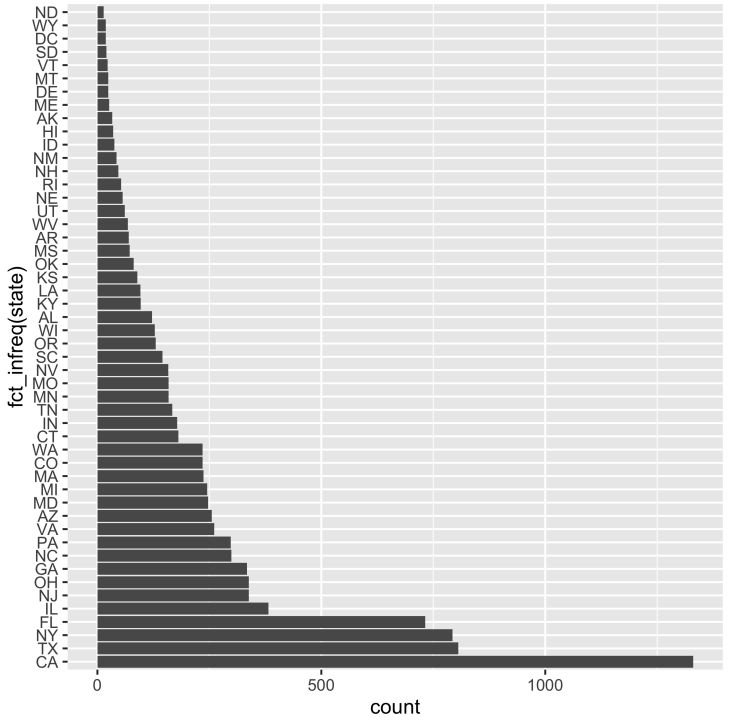
Flip!

```
ggplot(loans, aes(y = state)) +
  geom_bar()
```

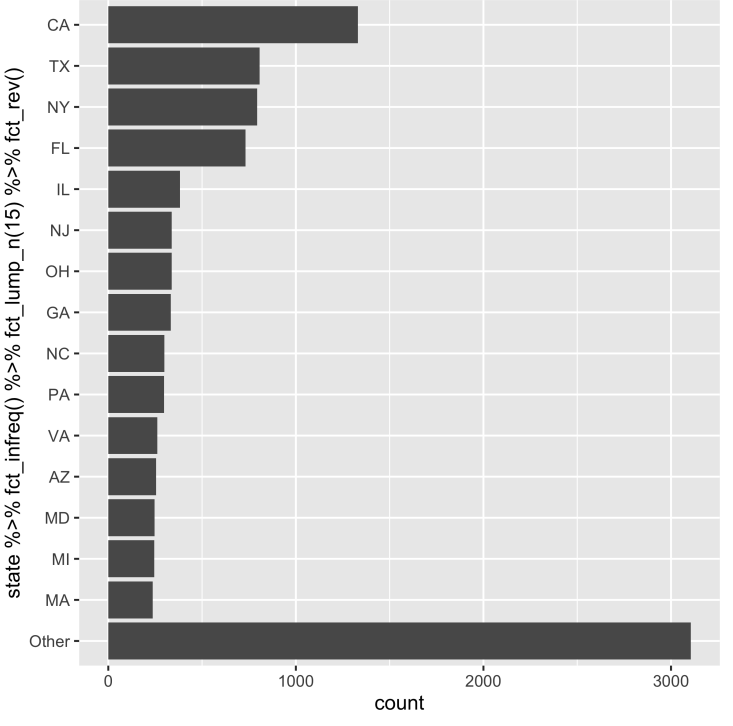


Use a meaningful order!

```
ggplot(loans, aes(y = fct_infr
  geom_bar()
```

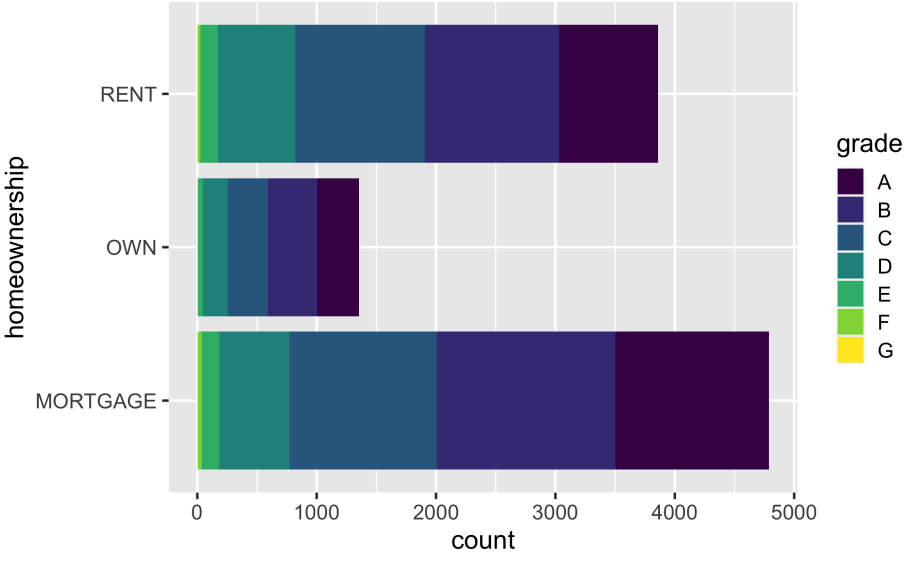



```
# bonus!
ggplot(loans, aes(y = state %>
  geom_bar()
```



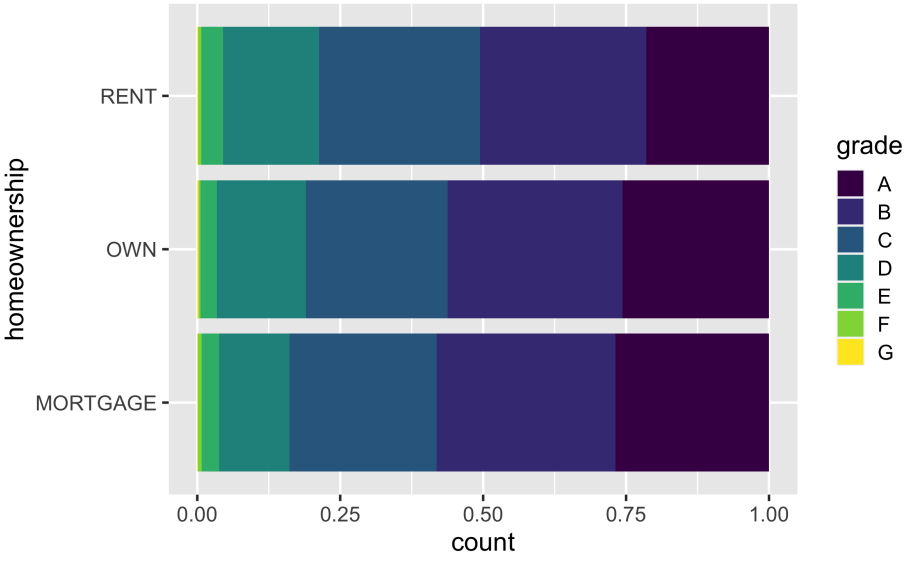
Segmented bar plot

```
ggplot(loans, aes(y = homeownership,
                  fill = grade)) +
  geom_bar()
```

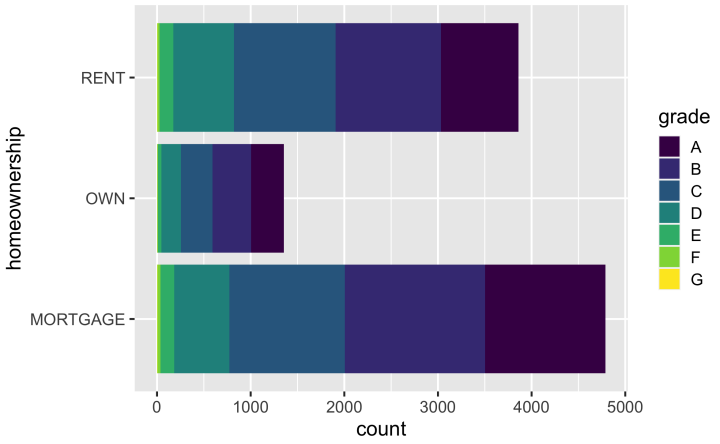


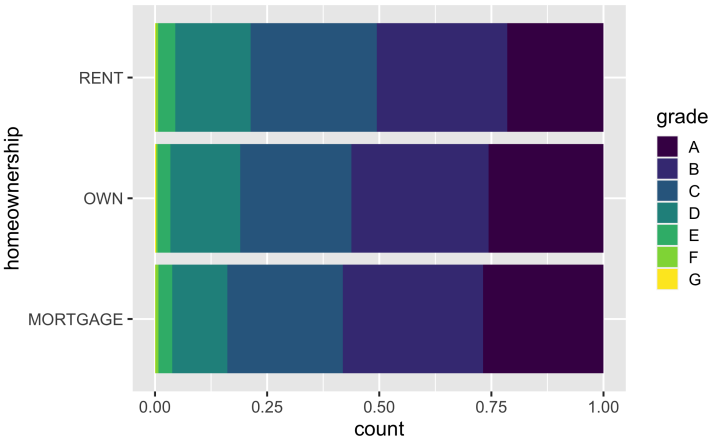
Segmented bar plot

```
ggplot(loans, aes(y = homeownership, fill = grade)) +
  geom_bar(position = "fill")
```



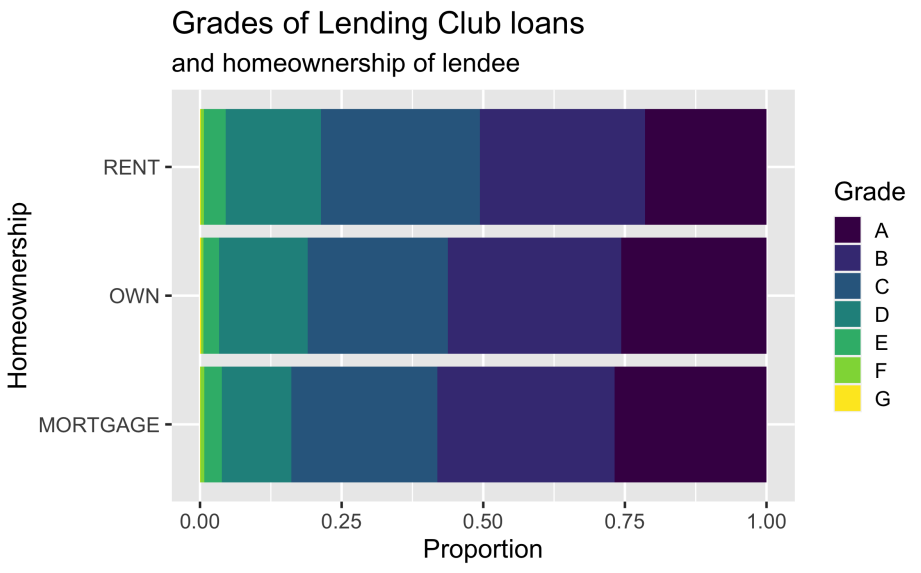
Which bar plot is a more useful representation for visualizing the relationship between homeownership and grade?





Customizing bar plots

Plot Code



Gotcha: geom_bar summarizes the data for you!

Counting | Plotting

```
loan_proportions <- loans %>%
  group_by(homeownership, grade) %>%
  summarize(count = n()) %>%
  group_by(homeownership) %>%
  mutate(prop = count / sum(count))
loan_proportions
```

```
## # A tibble: 21 × 4
## # Groups:   homeownership [3]
##   homeownership grade count    prop
##   <fct>         <ord> <int>   <dbl>
## 1 MORTGAGE      A      1285 0.268
## 2 MORTGAGE      B      1499 0.313
## 3 MORTGAGE      C      1234 0.258
## 4 MORTGAGE      D       587 0.123
## 5 MORTGAGE      E       148 0.0309
```

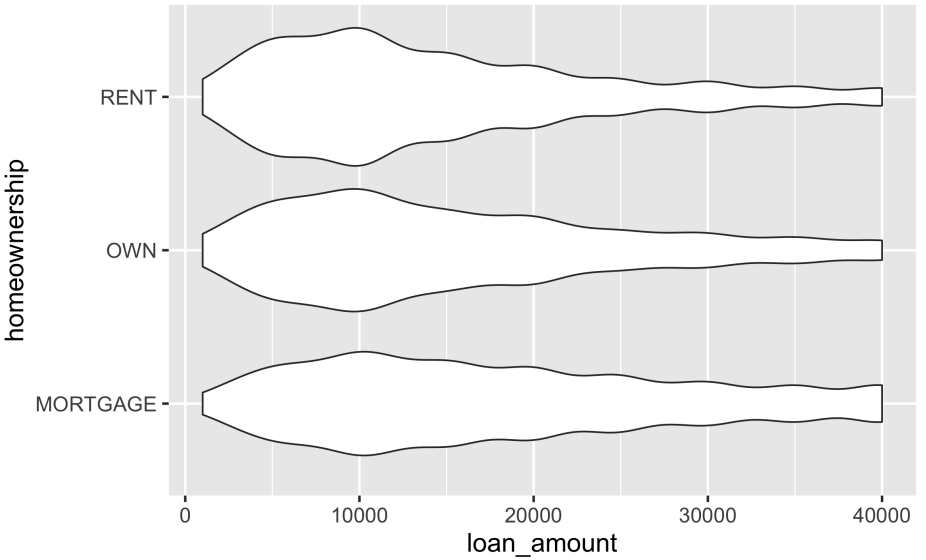
Relationships between numerical and categorical variables

Already talked about...

- Colouring and faceting histograms and density plots
- Side-by-side box plots

Violin plots

```
ggplot(loans, aes(y = homeownership, x = loan_amount)) +
  geom_violin()
```



Ridge plots

```
library(ggridges)
ggplot(loans, aes(x = loan_amount, y = grade, fill = grade, colo
  geom_density_ridges(alpha = 0.5)
```