Modern R with tidyverse [Solutions]

Jonathan de Bruin & Barbara Vreede

Contents

This document is part of the workshop ***Introduction to R & Data*** *by Utrecht University RDM Support.*

This document is work in progress.

```
library(tidyverse)
```

```
## -- Attaching packages --------------------------------- tidyverse 1.2.1 --
```

```
## ✓ ggplot2 2.2.1     ✓ purrr   0.2.4
## ✓ tibble  1.4.2     ✓ dplyr   0.7.4
## ✓ tidyr   0.8.0     ✓ stringr 1.3.1
## ✓ readr   1.1.1     ✓ forcats 0.3.0
```

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

1. Read and save data

Basic exercise I - Read data files

a) Read the dataset `menus.csv`

```
filepath <- file.path('data', 'menus.csv')
data_menus <- read_csv(filepath)
```

```
## Parsed with column specification:
## cols(
##   id = col_character(),
##   menus.amountMax = col_double(),
##   menus.amountMin = col_double(),
##   menus.currency = col_character(),
##   menus.dateSeen = col_character(),
##   menus.description = col_character(),
##   menus.name = col_character()
## )
```

```
head(data_menus)
```

```
## # A tibble: 6 x 7
##   id         menus.amountMax menus.amountMin menus.currency menus.dateSeen
##   <chr>                <dbl>           <dbl> <chr>          <chr>
## 1 AVwc_6KEI~            22.5            15.5 USD            2016,31+Mar
## 2 AVwc_6KEI~            19.0            19.0 USD            2016,31+Mar
```

```
## 3 AVwc_6qRB~            12.0            12.0 USD            2015,23+Oct
## 4 AVwc_6qRB~            13.0            13.0 USD            2015,23+Oct
## 5 AVwc_6qRB~            13.0            13.0 USD            2015,23+Oct
## 6 AVwc_6qRB~            15.0            15.0 USD            2015,23+Oct
## # ... with 2 more variables: menus.description <chr>, menus.name <chr>
```

b) Load `readxl`

```
library("readxl")
```

c) Read the restaurants dataset

```
filepath <- file.path('data', 'restaurants.xlsx')
data_restaurants <- read_excel(filepath)

head(data_restaurants)
```

```
## # A tibble: 6 x 14
##   id     address categories city  country latitude longitude name  zipcode
##   <chr>  <chr>   <chr>      <chr> <chr>      <dbl>     <dbl> <chr> <chr>
## 1 AVwc_~ Cascad~ Pizza Pla~ Bend  US          44.1    -121.  Litt~ 97701
## 2 AVwc_~ 148 S ~ American ~ Los ~ US          34.1    -118.  The ~ 90049
## 3 AVwc_~ 5142 H~ Pizza Pla~ Los ~ US          34.1    -118.  Brav~ 90027
## 4 AVwc_~ 801 Sa~ Bar,Beer ~ Hous~ US          29.8     -95.4 Luck~ 77003
## 5 AVwc_~ 478 So~ American ~ Hyan~ US          41.6     -70.3 Road~ 02601
## 6 AVwc_~ 1 N Un~ Universit~ Provo US          40.3    -112.  Brig~ 84602
## # ... with 5 more variables: priceRangeCurrency <chr>,
## #   priceRangeMin <dbl>, priceRangeMax <dbl>, menuPageURL <chr>,
## #   state <chr>
```

Basic exercise II - Dataset properties

```
glimpse(data_restaurants)
```

```
## Observations: 989
## Variables: 14
## $ id                 <chr> "AVwc_6KEIN2L1WUfrKAH", "AVwc_6qRByjofQCxkc...
## $ address            <chr> "Cascade Village Mall Across From Target", ...
## $ categories         <chr> "Pizza Place", "American Restaurant,Bar,Bak...
## $ city               <chr> "Bend", "Los Angeles", "Los Angeles", "Hous...
## $ country            <chr> "US", "US", "US", "US", "US", "US", "US", "...
## $ latitude           <dbl> 44.10266, 34.06456, 34.10174, 29.75248, 41....
## $ longitude          <dbl> -121.30080, -118.46902, -118.30197, -95.354...
## $ name               <chr> "Little Pizza Paradise", "The Brentwood", "...
## $ zipcode            <chr> "97701", "90049", "90027", "77003", "02601"...
## $ priceRangeCurrency <chr> NA, "USD", NA, "USD", NA, NA, "USD", "USD",...
## $ priceRangeMin      <dbl> NA, 50, NA, 25, NA, NA, 0, 25, NA, 0, 25, N...
## $ priceRangeMax      <dbl> NA, 55, NA, 40, NA, NA, 25, 40, NA, 30, 40,...
## $ menuPageURL        <chr> NA, NA, NA, NA, NA, NA, NA, NA, "http://www...
## $ state              <chr> "OR", "CA", "CA", "TX", "MA", "UT", "TX", "...
```

```
glimpse(data_menus)
```

```
## Observations: 3,510
```

```
## Variables: 7
## $ id                <chr> "AVwc_6KEIN2L1WUfrKAH", "AVwc_6KEIN2L1WUfrKA...
## $ menus.amountMax   <dbl> 22.50, 18.95, 12.00, 13.00, 13.00, 15.00, 15...
## $ menus.amountMin   <dbl> 15.50, 18.95, 12.00, 13.00, 13.00, 15.00, 15...
## $ menus.currency    <chr> "USD", "USD", "USD", "USD", "USD", "USD", "U...
## $ menus.dateSeen    <chr> "2016,31+Mar", "2016,31+Mar", "2015,23+Oct",...
## $ menus.description <chr> NA, NA, NA, NA, "Olives, onions, capers, tom...
## $ menus.name        <chr> "Bianca Pizza", "Cheese Pizza", "Pizza, Marg...
```

Reading exercise - `readr` versus base R

Optional exercise (+) - Save data to a CSV file.

```
write_delim(data_menus, "menus_in_csv_format.csv", delim = ";")
```

Optional exercise (++) - Read SPSS, SAS and Excel data files

```
library("haven") # to read and write SPSS, STATA and SAS files
library("readxl") # to read Excel files
```

a) Write data frame to SPSS, SAS, STATA data files.

```
# create a directory
if (!dir.exists('tmp')){
  dir.create("tmp")
}

# read and write files
write_sav(data_restaurants, file.path("tmp", "restaurants_spss.sav"))
data_restaurants_spss <- read_sav(file.path("tmp", "restaurants_spss.sav"))

write_sas(data_restaurants_spss, file.path("tmp", "restaurants_sas.sas7bdat"))
data_restaurants_sas <- read_sas(file.path("tmp", "restaurants_sas.sas7bdat"))

write_dta(data_restaurants_sas, file.path("tmp", "restaurants_stata.dta"))
data_restaurants_stata <- read_dta(file.path("tmp", "restaurants_stata.dta"))

head(data_restaurants_stata)
```

```
## # A tibble: 6 x 14
##   id     address categories city  country latitude longitude name  zipcode
##   <chr>  <chr>   <chr>      <chr> <chr>      <dbl>     <dbl> <chr> <chr>
## 1 AVwc_~ Cascad~ Pizza Pla~ Bend  US          44.1    -121.  Litt~ 97701
## 2 AVwc_~ 148 S ~ American ~ Los ~ US          34.1    -118.  The ~ 90049
## 3 AVwc_~ 5142 H~ Pizza Pla~ Los ~ US          34.1    -118.  Brav~ 90027
## 4 AVwc_~ 801 Sa~ Bar,Beer ~ Hous~ US          29.8     -95.4 Luck~ 77003
## 5 AVwc_~ 478 So~ American ~ Hyan~ US          41.6     -70.3 Road~ 02601
## 6 AVwc_~ 1 N Un~ Universit~ Provo US          40.3    -112.  Brig~ 84602
## # ... with 5 more variables: priceRangeCurrency <chr>,
## #   priceRangeMin <dbl>, priceRangeMax <dbl>, menuPageURL <chr>,
## #   state <chr>
```

b) Write data frame to Excel.

This is not possible with `tidyverse` at the moment. `readxl` only support Excel file reading. This is not a problem for a researcher, because we don't use Excel, isn't it?

Optional exercise (+++) - Parse datetime columns

No solutions available at the moment.

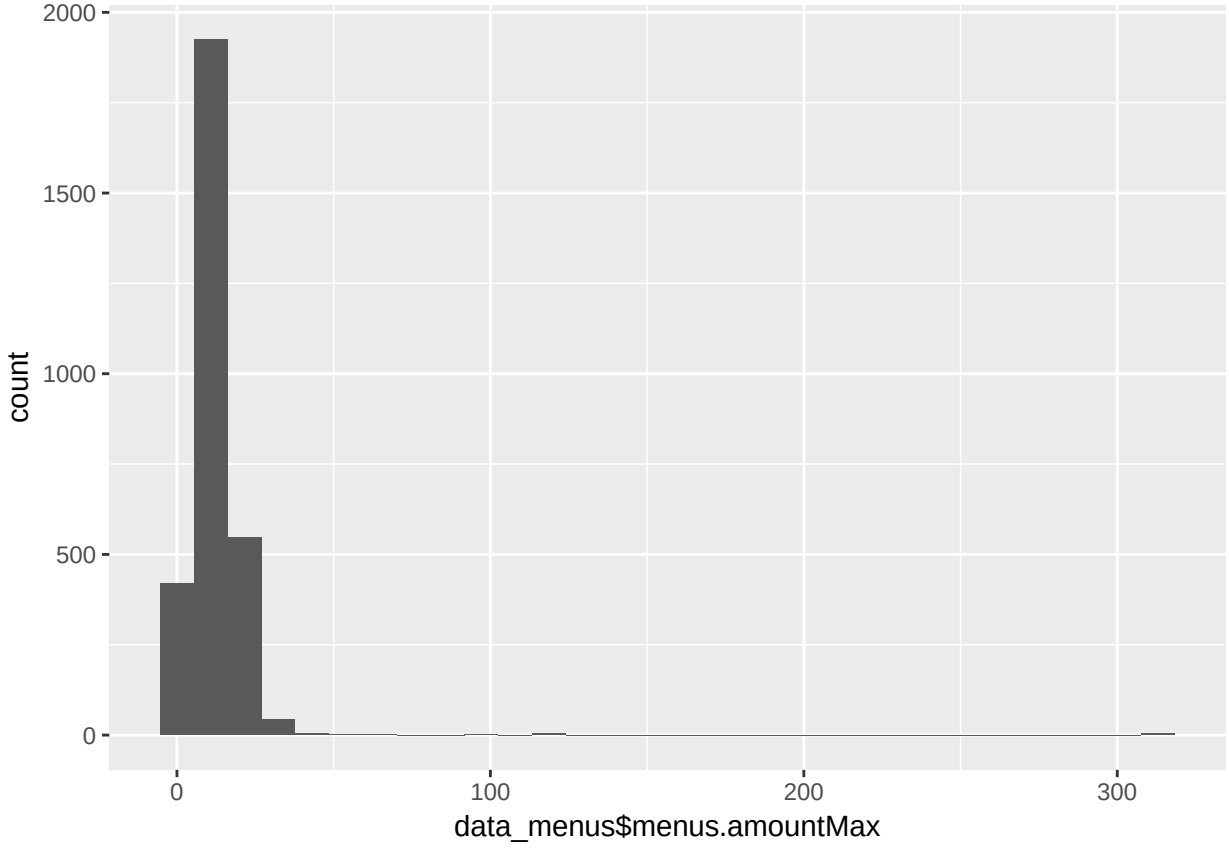
2. Data visualisation

Basic exercise I - Quick plots of the menus

a) Single column plots

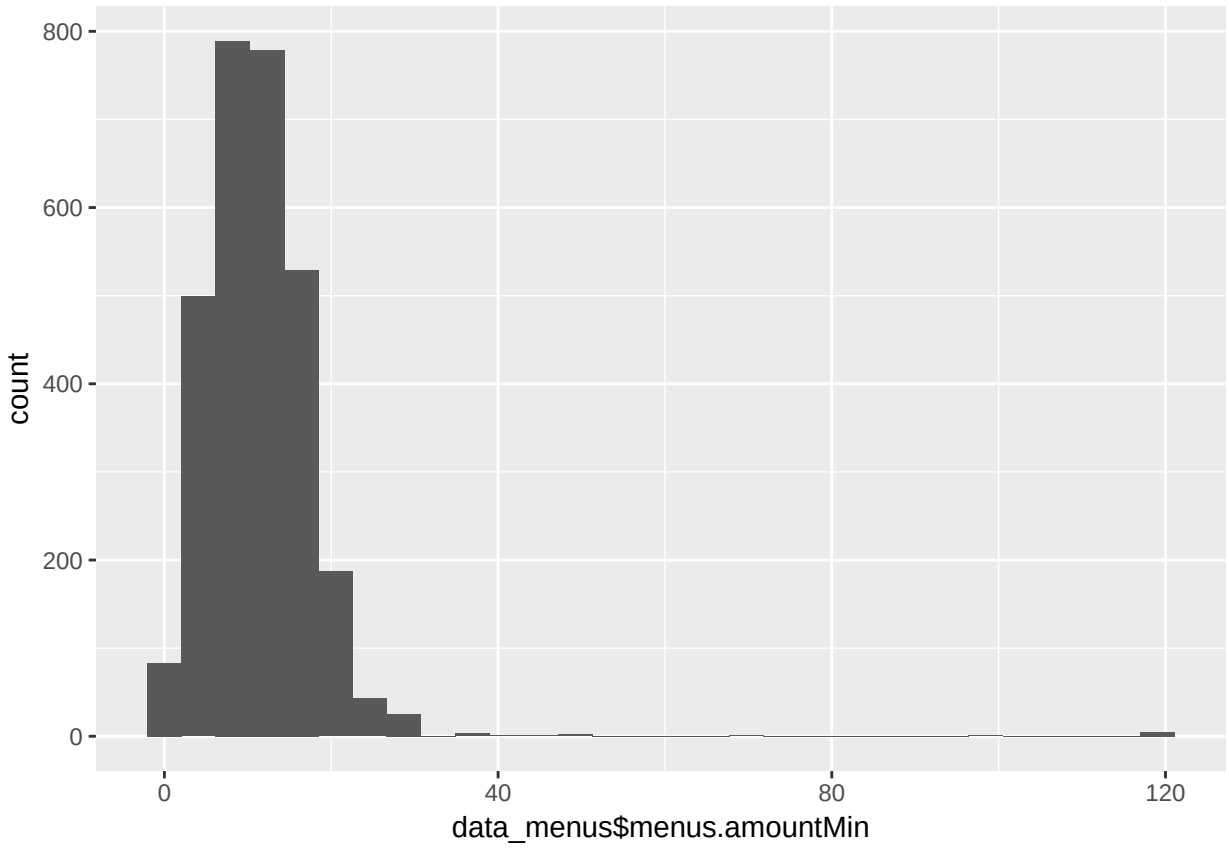
```
qplot(data_menus$menus.amountMax)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 562 rows containing non-finite values (stat_bin).
```

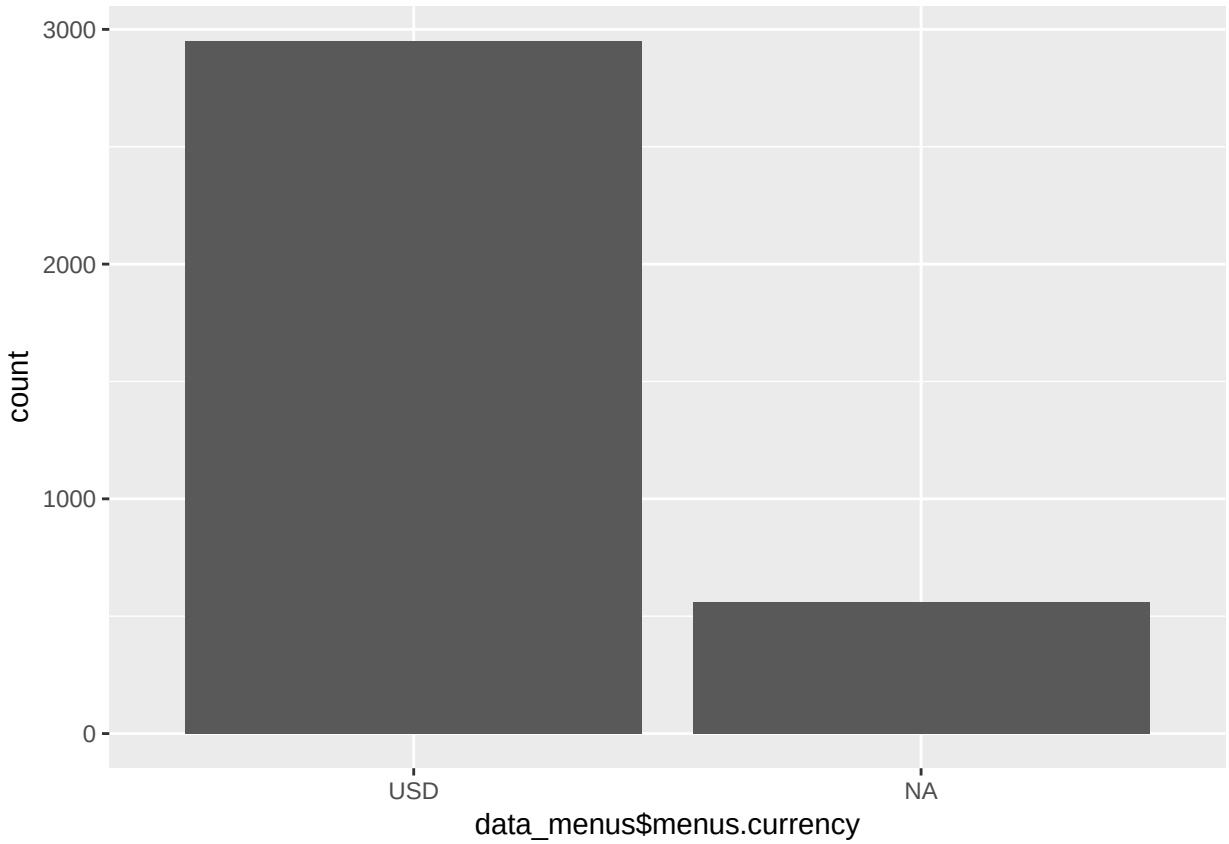



```
qplot(data_menus$menus.amountMin)
```

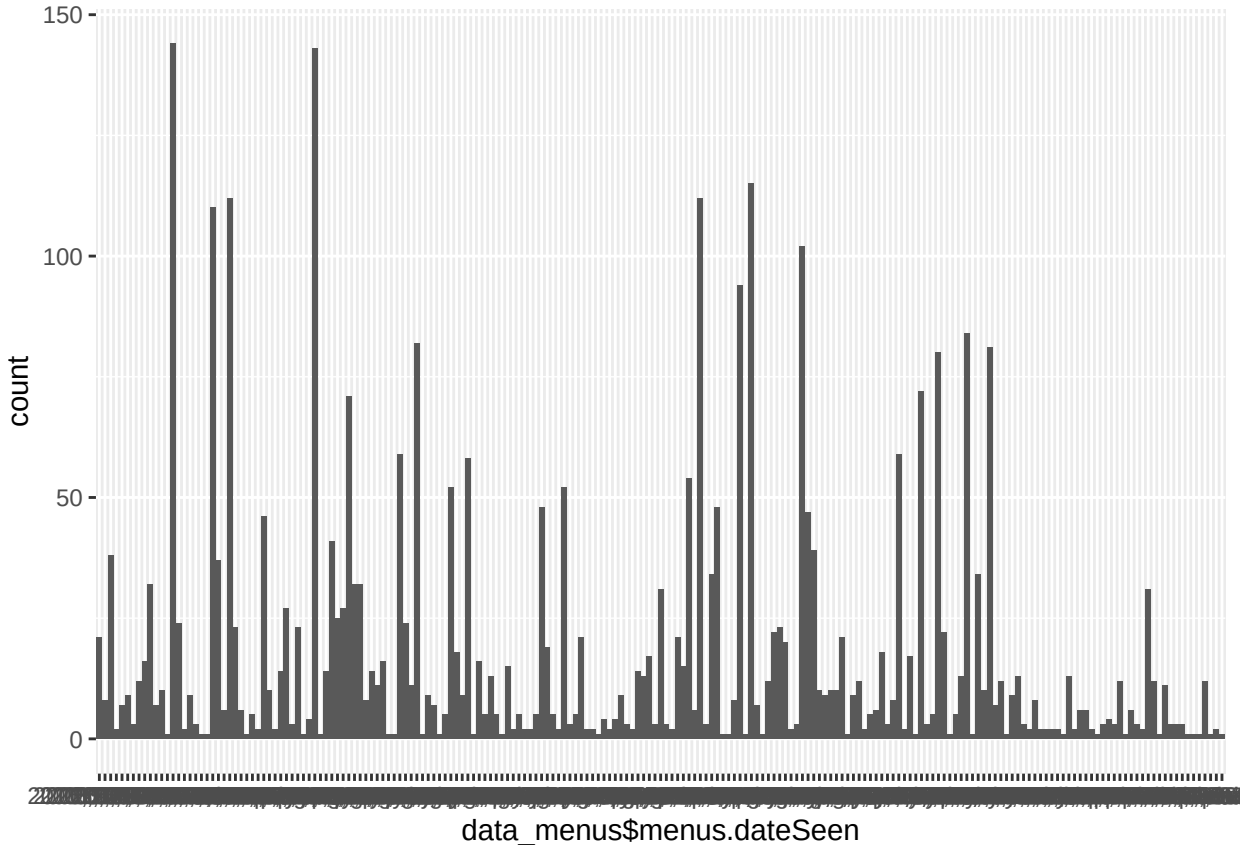
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 562 rows containing non-finite values (stat_bin).
```




```
qplot(data_menus$menus.currency)
```




```
qplot(data_menus$menus.dateSeen)
```

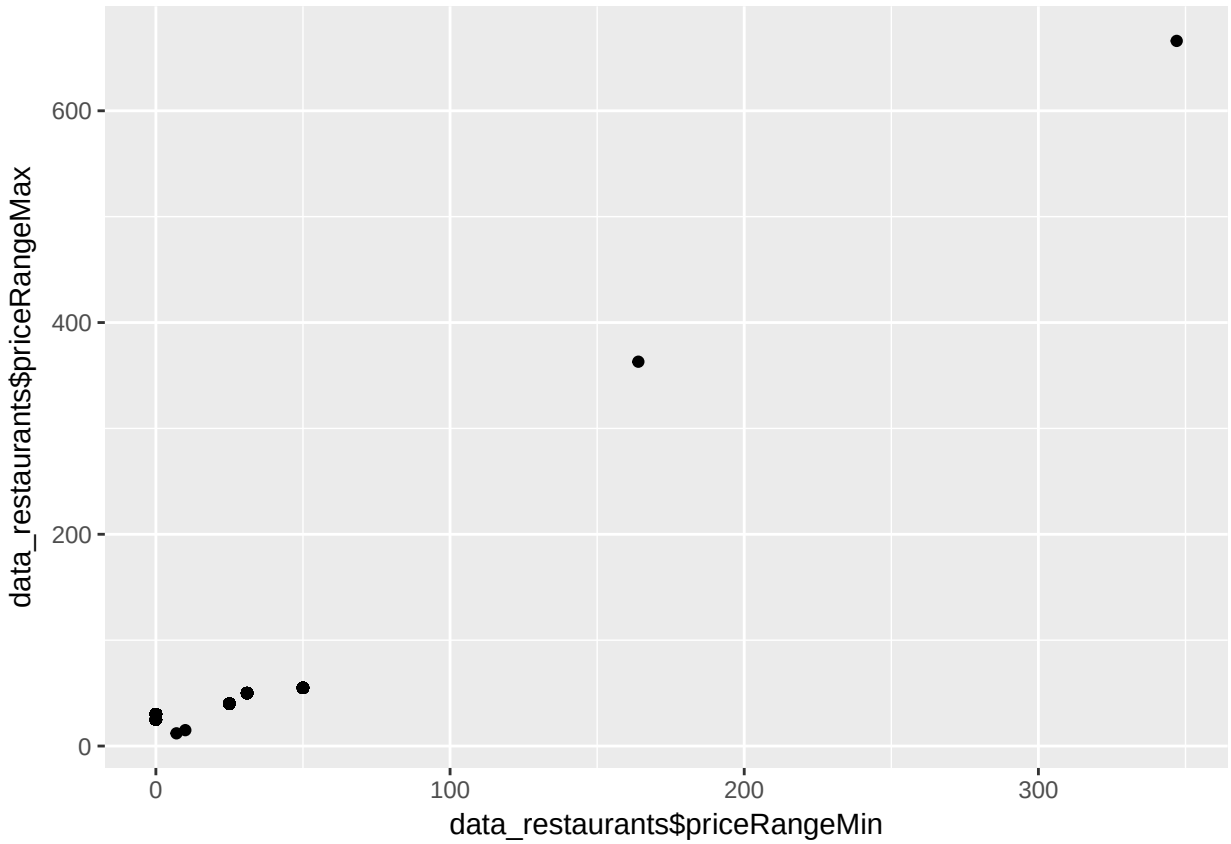



```
qplot(data_menus$menus.description)
```

b) Two column in plots

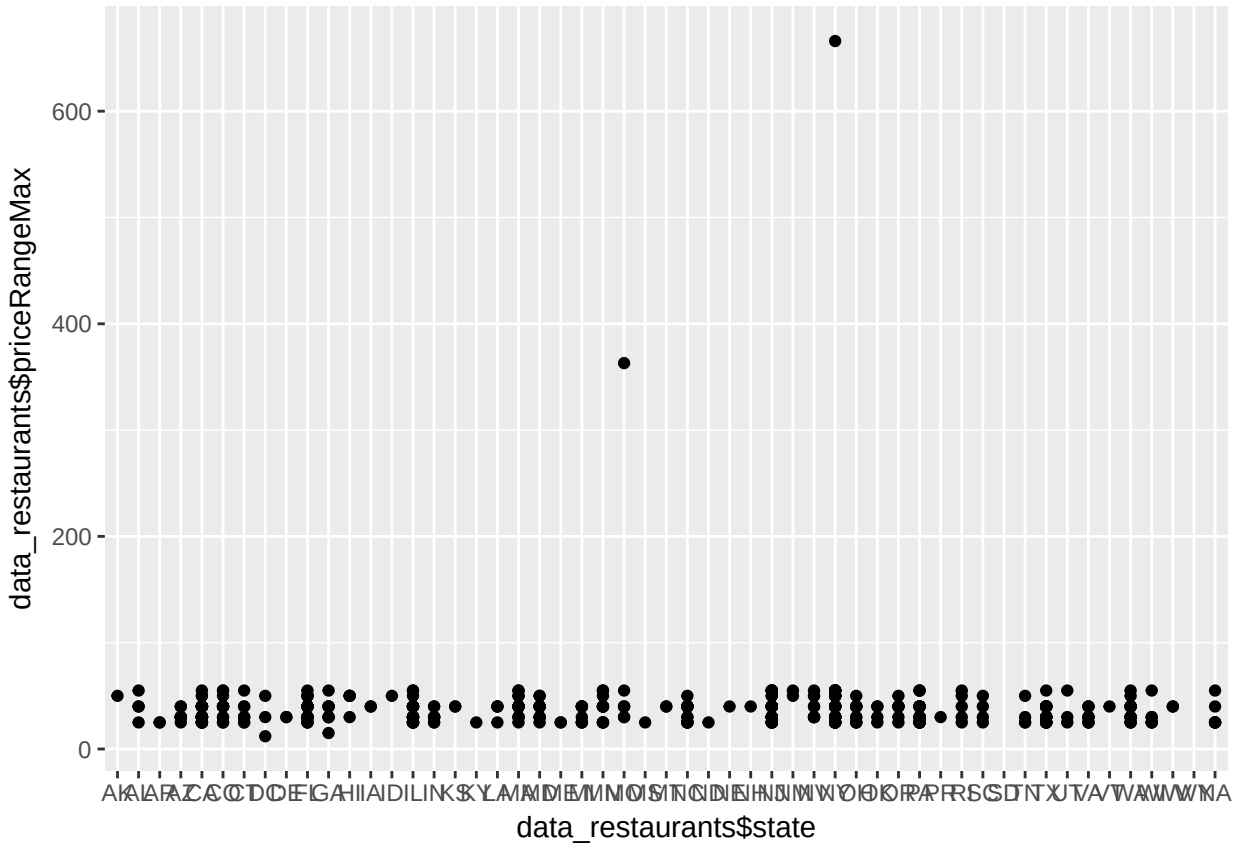
```
qplot(x = data_restaurants$priceRangeMin, y = data_restaurants$priceRangeMax)
```

```
## Warning: Removed 452 rows containing missing values (geom_point).
```




```
qplot(x = data_restaurants$state, y = data_restaurants$priceRangeMax)
```

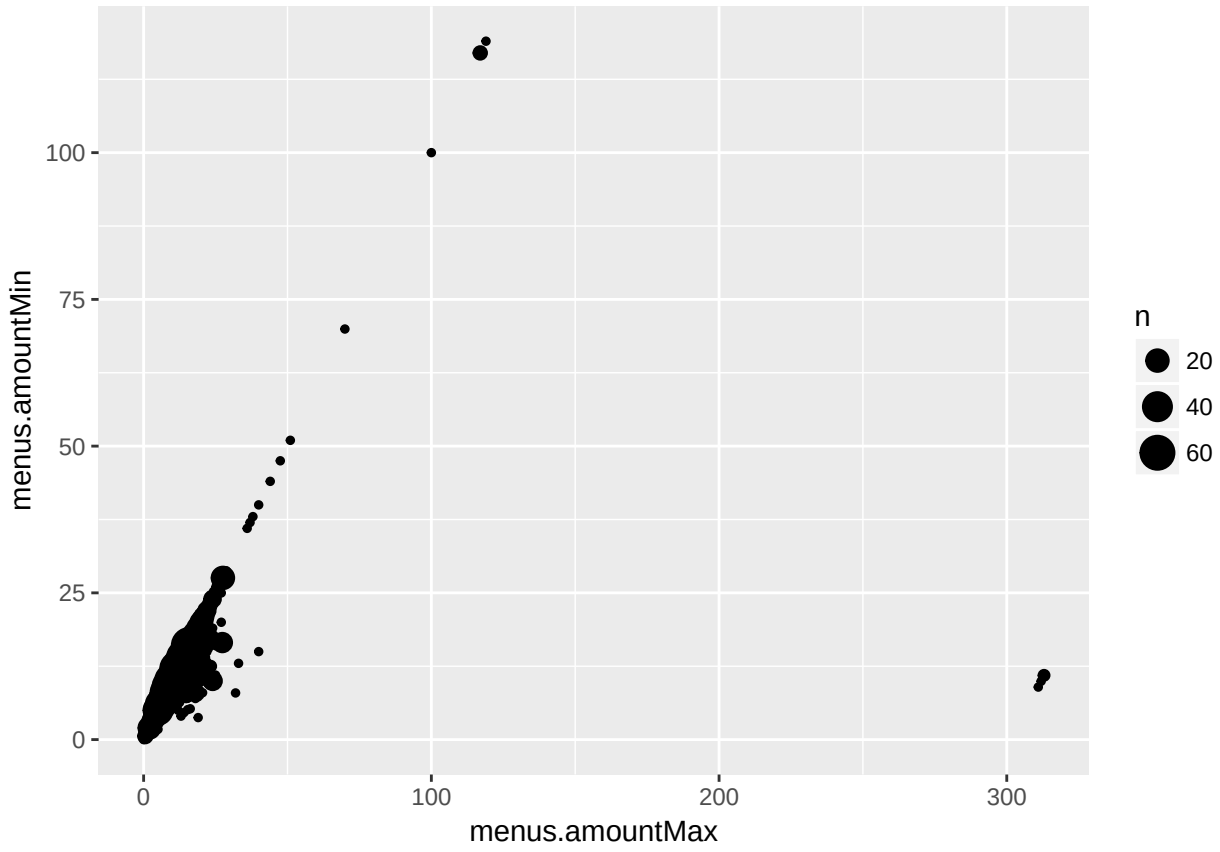
```
## Warning: Removed 452 rows containing missing values (geom_point).
```



Basic exercise II - Using ggplot for graphs

```
ggplot(data_menus, aes(menus.amountMax, menus.amountMin)) +
  geom_count()
```

```
## Warning: Removed 562 rows containing non-finite values (stat_sum).
```

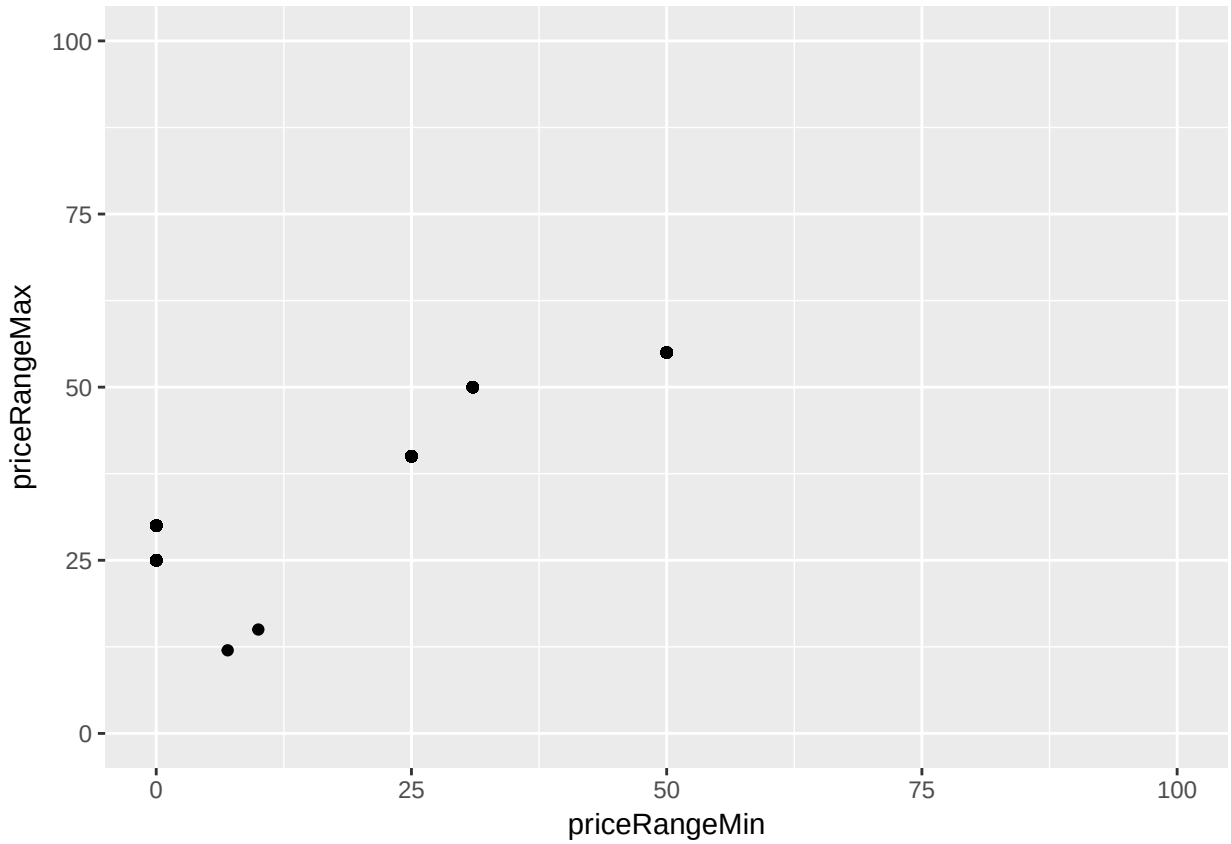


Reading exercise - Statistical layers for graphs.

Optional excercise (+) - Scale axis

```
ggplot(data_restaurants, aes(priceRangeMin, priceRangeMax)) +
  geom_point() +
  scale_x_continuous(limits = c(0, 100)) +
  scale_y_continuous(limits = c(0, 100))
```

```
## Warning: Removed 454 rows containing missing values (geom_point).
```



Optional excercise (++) - Plot the restaurants on a map

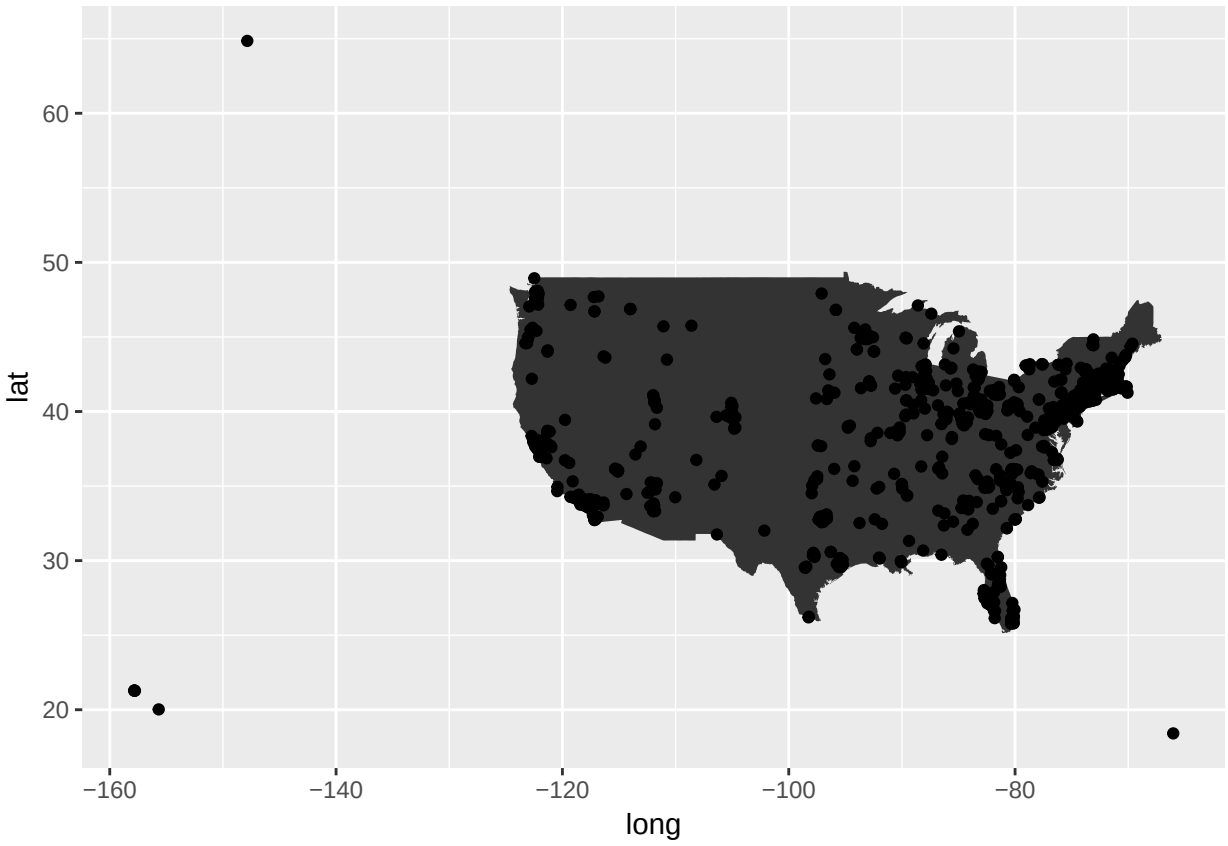
a) Install package the `maps`

```
# install.packages('maps')
library(maps)
```

```
##
## Attaching package: 'maps'
## The following object is masked from 'package:purrr':
##
##     map
```

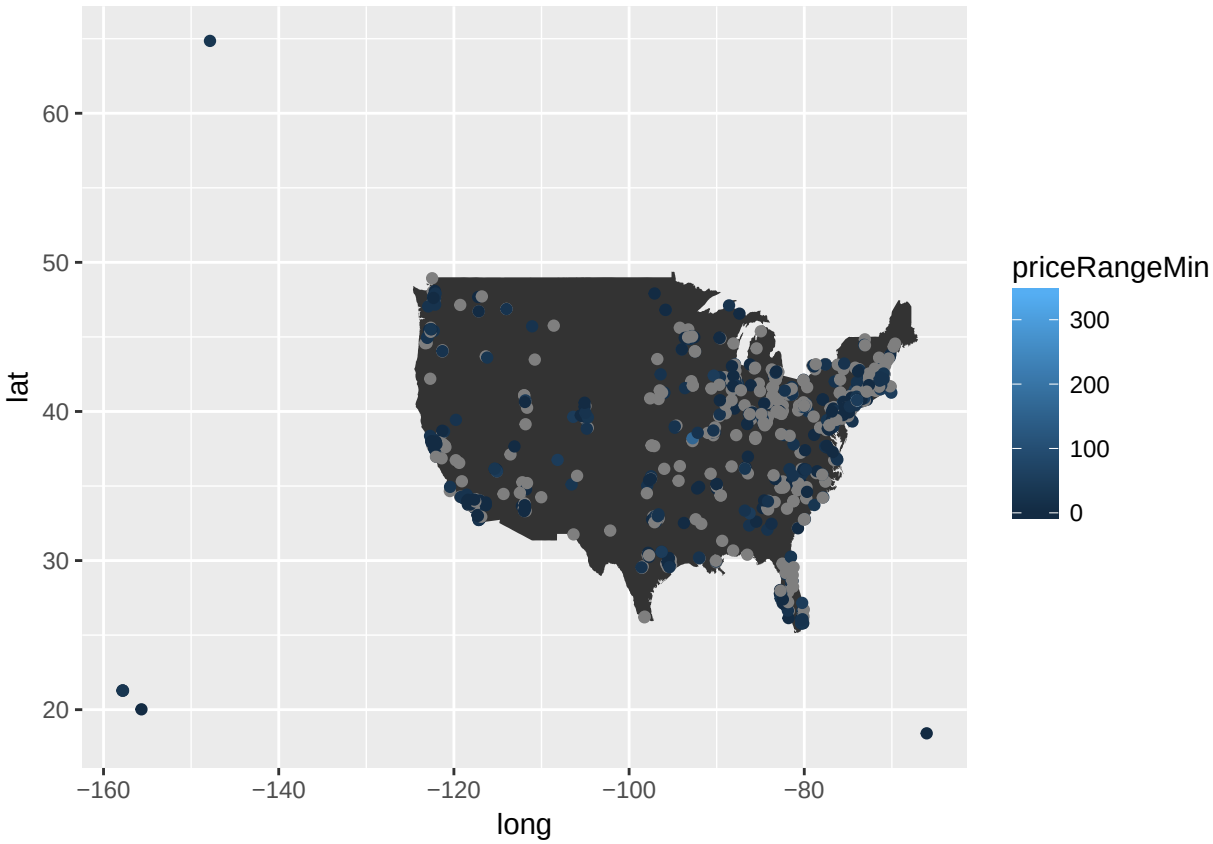
b) Plot the restaurants on the map of the USA.

```
usa <- map_data("usa")

ggplot(data_restaurants) +
  geom_polygon(data = usa, aes(x=long, y = lat)) +
  geom_point(aes(x=longitude, y=latitude))
```



c) Use the maximum price to colour the restaurants.

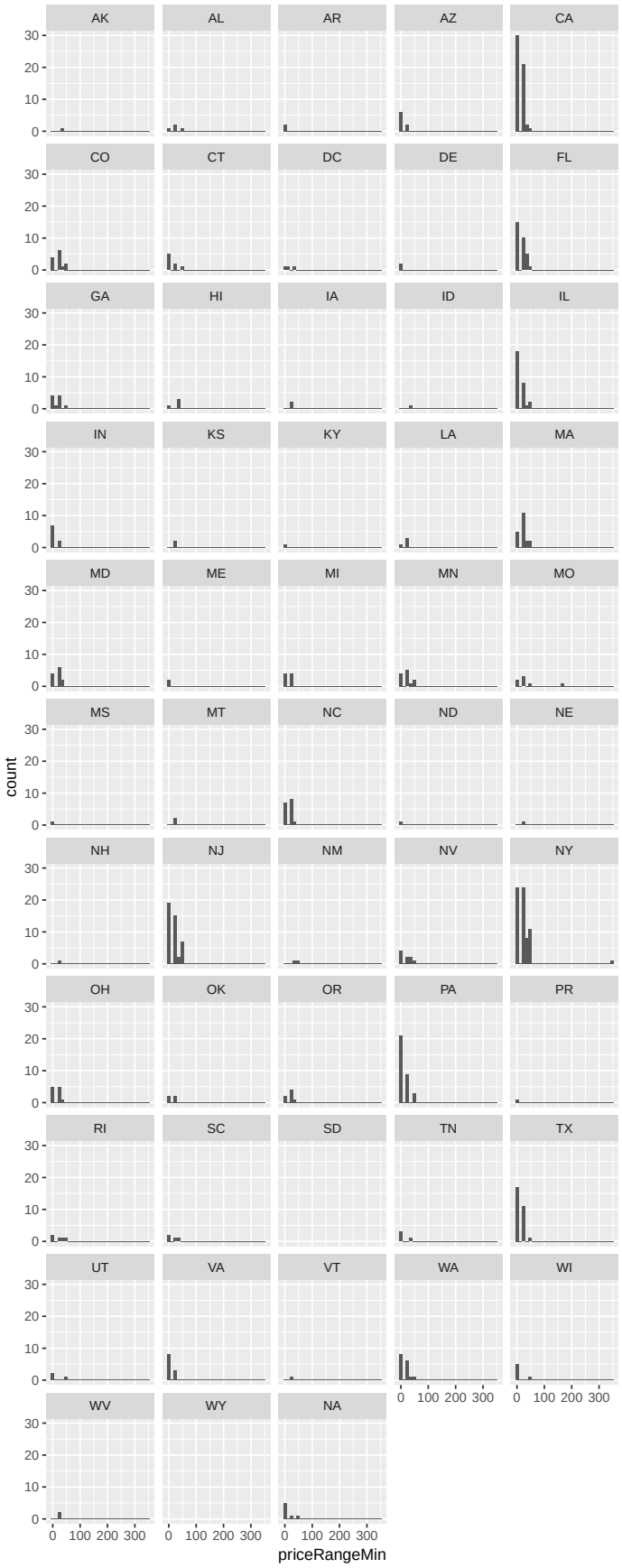
```
usa <- map_data("usa")

ggplot(data_restaurants) +
  geom_polygon(data = usa, aes(x=long, y = lat)) +
  geom_point(aes(x=longitude, y=latitude, colour=priceRangeMin))
```



Optional exercise (+++) - Create facets.

```
ggplot(data_restaurants) +
  geom_histogram(aes(x = priceRangeMin)) +
  facet_wrap(~ state, ncol = 5)
```

```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## Warning: Removed 452 rows containing non-finite values (stat_bin).
```


AK
AL
AR
AZ
CA
CO
CT
DC
DE
FL
GA
HI
IA
ID
IL
IN
KS
KY
LA
MA
MD
ME
MI
MN
MO
MS
MT
NC
ND
NE
NH
NJ
NM
NV
NY
OH
OK
OR
PA
PR
RI
SC
SD
TN
TX
UT
VA
VT
WA
WI
WV
WY
NA
count
0
10
20
30
0
100
200
300
priceRangeMin

3. Data transformation

Basic exercise I - Subset data

a) Make a selection of all restaurants in 'Los Angeles'.

```
filter(data_restaurants, city=='Los Angeles')
```

```
## # A tibble: 8 x 14
##   id     address categories city  country latitude longitude name  zipcode
##   <chr>  <chr>   <chr>      <chr> <chr>      <dbl>     <dbl> <chr> <chr>
## 1 AVwc_~ 148 S ~ American ~ Los ~ US          34.1     -118. The ~ 90049
## 2 AVwc_~ 5142 H~ Pizza Pla~ Los ~ US          34.1     -118. Brav~ 90027
## 3 AVwck~ 8136 W~ Motion Pi~ Los ~ US          34.1     -118. Doug~ 90048
## 4 AVwcr~ 11633 ~ Italian R~ Los ~ US          34.1     -118. Tosc~ 90049
## 5 AVwcy~ 11628 ~ Caterers,~ Los ~ US          34.0     -118. Nort~ 90025
## 6 AVwdk~ 505 S ~ Caf,Resta~ Los ~ US          34.1     -118. Mang~ 90071
## 7 AVwdK~ 10835 ~ Latin Ame~ Los ~ US          34.0     -118. Bamb~ 90034
## 8 AVweK~ 300 S ~ Restaurant Los ~ US          34.1     -118. Culi~ 90048
## # ... with 5 more variables: priceRangeCurrency <chr>,
## #   priceRangeMin <dbl>, priceRangeMax <dbl>, menuPageURL <chr>,
## #   state <chr>
```

b) Make a selection of all restaurants in 'Los Angeles' where the variable `priceRangeMin` isn't missing.

```
filter(data_restaurants, city=='Los Angeles' & !is.na(priceRangeMin))
```

```
## # A tibble: 4 x 14
##   id     address categories city  country latitude longitude name  zipcode
##   <chr>  <chr>   <chr>      <chr> <chr>      <dbl>     <dbl> <chr> <chr>
## 1 AVwc_~ 148 S ~ American ~ Los ~ US          34.1     -118. The ~ 90049
## 2 AVwck~ 8136 W~ Motion Pi~ Los ~ US          34.1     -118. Doug~ 90048
## 3 AVwdK~ 10835 ~ Latin Ame~ Los ~ US          34.0     -118. Bamb~ 90034
## 4 AVweK~ 300 S ~ Restaurant Los ~ US          34.1     -118. Culi~ 90048
## # ... with 5 more variables: priceRangeCurrency <chr>,
## #   priceRangeMin <dbl>, priceRangeMax <dbl>, menuPageURL <chr>,
## #   state <chr>
```

c) Make a selection of all restaurants in 'Los Angeles' where the variable `priceRangeMin` is not missing. Return only the `address` and `name` of the restaurants.

```
data_restaurants_filtered <- filter(data_restaurants, city=='Los Angeles' & !is.na(priceRangeMin))

select(data_restaurants_filtered, address, name)
```

```
## # A tibble: 4 x 2
##   address              name
##   <chr>                <chr>
## 1 148 S Barrington Ave The Brentwood
## 2 8136 W 3rd St        Doughboys
## 3 10835 Venice Blvd    Bamboo Restaurant
## 4 300 S Doheny Dr      Culina
```

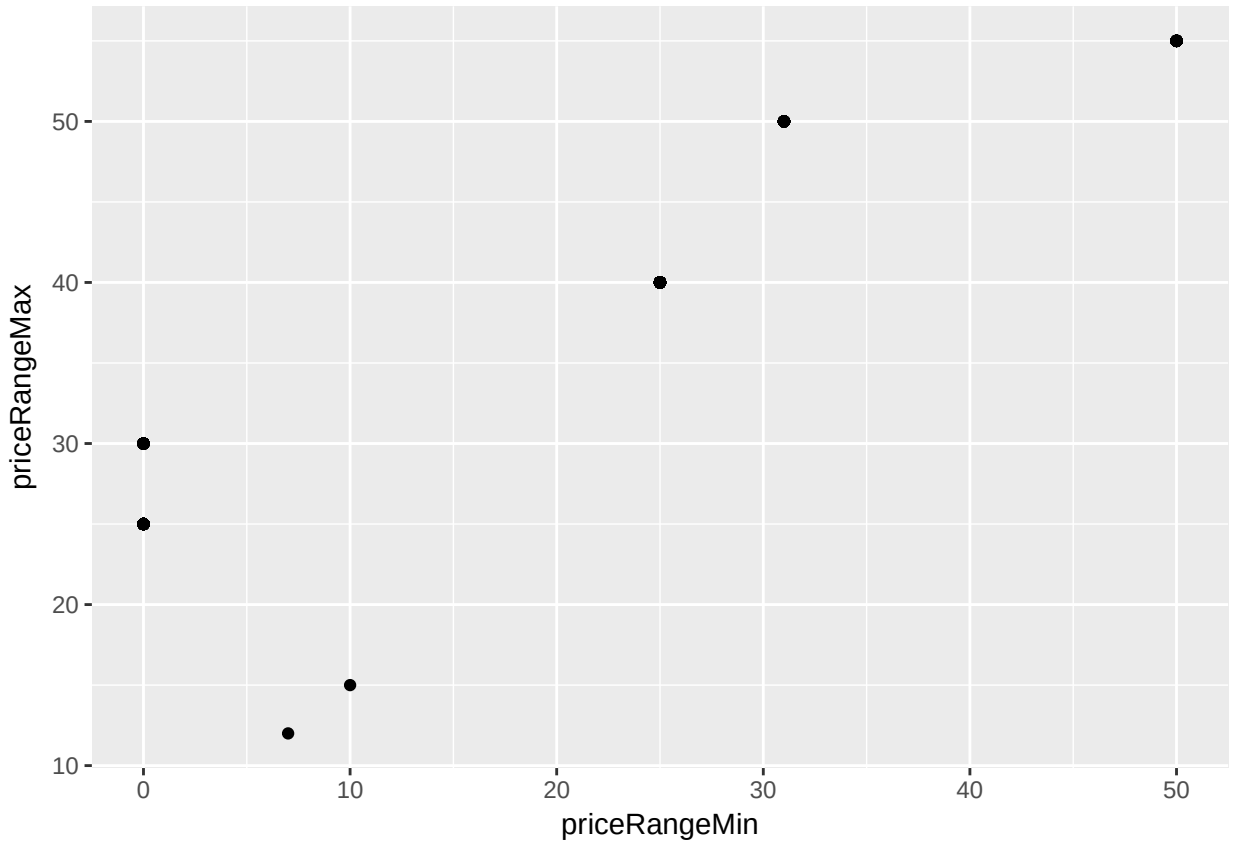
Basic exercise II - Compute the price range

```
data_restaurants_with_price_range <- mutate(data_restaurants,
  # compute the price range
  priceRangeDiff = priceRangeMax - priceRangeMin
)

head(data_restaurants_with_price_range)
```

```
## # A tibble: 6 x 15
##   id     address categories city  country latitude longitude name  zipcode
##   <chr>  <chr>   <chr>      <chr> <chr>      <dbl>     <dbl> <chr> <chr>
## 1 AVwc_~ Cascad~ Pizza Pla~ Bend  US          44.1    -121.  Litt~ 97701
## 2 AVwc_~ 148 S ~ American ~ Los ~ US          34.1    -118.  The ~ 90049
## 3 AVwc_~ 5142 H~ Pizza Pla~ Los ~ US          34.1    -118.  Brav~ 90027
## 4 AVwc_~ 801 Sa~ Bar,Beer ~ Hous~ US          29.8     -95.4 Luck~ 77003
## 5 AVwc_~ 478 So~ American ~ Hyan~ US          41.6     -70.3 Road~ 02601
## 6 AVwc_~ 1 N Un~ Universit~ Provo US          40.3    -112.  Brig~ 84602
## # ... with 6 more variables: priceRangeCurrency <chr>,
## #   priceRangeMin <dbl>, priceRangeMax <dbl>, menuPageURL <chr>,
## #   state <chr>, priceRangeDiff <dbl>
```

Basic exercise III - Filter outliers

```
data_restautants_wo_outliers <- filter(data_restaurants, priceRangeMin < 100, priceRangeMax < 100)

ggplot(data_restautants_wo_outliers, aes(priceRangeMin, priceRangeMax)) +
  geom_point()
```



Reading exercise - Pipe operator

Optional exercise (+) - Exclude variables

Create a tibble of the restaurant dataset without the latitude and longitude.

Use tidyverse and a maximum of 75 characters. (Our best result is 42 characters.)

```
select(data_restaurants, -latitude, -longitude)
```

```
## # A tibble: 989 x 12
##    id     address categories  city  country name  zipcode priceRangeCurre~
##    <chr>  <chr>   <chr>       <chr> <chr>   <chr> <chr>   <chr>
##  1 AVwc_~ Cascad~ Pizza Place Bend  US      Litt~ 97701   <NA>
##  2 AVwc_~ 148 S ~ American R~ Los ~ US      The ~ 90049   USD
##  3 AVwc_~ 5142 H~ Pizza Place Los ~ US      Brav~ 90027   <NA>
##  4 AVwc_~ 801 Sa~ Bar,Beer G~ Hous~ US      Luck~ 77003   USD
##  5 AVwc_~ 478 So~ American R~ Hyan~ US      Road~ 02601   <NA>
##  6 AVwc_~ 1 N Un~ University~ Provo US      Brig~ 84602   <NA>
##  7 AVwc_~ 9595 S~ Sporting G~ Spri~ US      Luke~ 77380   USD
##  8 AVwc_~ 200 E ~ Italian Re~ Chic~ US      Fran~ 60611   USD
##  9 AVwc_~ 145 E ~ Bagels,Bak~ West~ US      Coun~ 19380   <NA>
## 10 AVwc_~ 925 Bl~ Restaurant  San ~ US      Buca~ 95123   USD
## # ... with 979 more rows, and 4 more variables: priceRangeMin <dbl>,
## #   priceRangeMax <dbl>, menuPageURL <chr>, state <chr>
```

Optional exercise (++) - Summarise results

```
summarise(data_menus,
          mean_min_price = mean(menus.amountMin, na.rm = T),
          mean_max_price = mean(menus.amountMax, na.rm = T),
          mean_diff_price = mean(menus.amountMax - menus.amountMin, na.rm = T)
)
```

```
## # A tibble: 1 x 3
##   mean_min_price mean_max_price mean_diff_price
##            <dbl>          <dbl>           <dbl>
## 1           11.4           12.5            1.05
```

Optional exercise (+++) - Join datasets

```
data_pizza_with_restaurant <- left_join(data_menus, data_restaurants, by="id")
head(data_pizza_with_restaurant)
```

```
## # A tibble: 6 x 20
##   id         menus.amountMax menus.amountMin menus.currency menus.dateSeen
##   <chr>                <dbl>           <dbl> <chr>          <chr>
## 1 AVwc_6KEI~            22.5            15.5 USD            2016,31+Mar
## 2 AVwc_6KEI~            19.0            19.0 USD            2016,31+Mar
## 3 AVwc_6qRB~            12.0            12.0 USD            2015,23+Oct
## 4 AVwc_6qRB~            13.0            13.0 USD            2015,23+Oct
## 5 AVwc_6qRB~            13.0            13.0 USD            2015,23+Oct
## 6 AVwc_6qRB~            15.0            15.0 USD            2015,23+Oct
## # ... with 15 more variables: menus.description <chr>, menus.name <chr>,
## #   address <chr>, categories <chr>, city <chr>, country <chr>,
## #   latitude <dbl>, longitude <dbl>, name <chr>, zipcode <chr>,
## #   priceRangeCurrency <chr>, priceRangeMin <dbl>, priceRangeMax <dbl>,
## #   menuPageURL <chr>, state <chr>
```

Multiple options give the same results. Explain why!