R+ Prüfung

180303015

Ömer Durkut

1-Wie würden Sie „Grammar of Graphics“ kurz definieren?

Angewandt auf Visualisierungen ist eine Grammatik von Grafiken eine Grammatik, die verwendet wird, um eine breite Palette von statistischen Grafiken zu beschreiben und zu erstellen. Der Ansatz der mehrschichtigen Grafikgrammatik ist in ggplot2 implementiert, einer weit verbreiteten Grafikbibliothek für R.

2- Definieren Sie „die kategorische und ordinale Variable“ und erläutern Sie die

Verschiedenheit mit Beispielen zwischen den beiden Begriffen.

Ordinale Variablen in R:

Mit der Funktion factor() kann Man den nominalen Variablen auch eine Ordnung zuweisen und sie so zu ordinalen Variablen machen. Dies geschieht, indem der Parameter order auf TRUE gesetzt wird und den Argumentebenen ein Vektor mit der gewünschten Ebenenhierarchie zugewiesen wird

İnput:

```
temperature_vector <- c("High", "Low", "High", "Low", "Medium")

factor_temperature_vector <- factor(temperature_vector, order = TRUE,
                                    levels = c("Low", "Medium", "High"))

factor_temperature_vector
```

output:

```
factor_temperature_vector
[1] High   Low    High   Low    Medium
```

Levels: Low < Medium < High

Kategorische Variable in R:

Kategoriale Variablen (auch Faktor- oder qualitative Variablen genannt) sind Variablen, die Beobachtungen in Gruppen klassifizieren. Sie haben eine begrenzte Anzahl verschiedener Werte, die als Ebenen bezeichnet werden.

Ein gutes Beispiel für kategoriale Variablen ist die Variable 'Geschlecht'. Ein menschliches Individuum kann entweder "männlich" oder "weiblich" sein. So sind hier "männlich" und "weiblich", im vereinfachten Sinn, die beiden Werte der kategorialen Variable "Geschlecht" und jede Beobachtung kann entweder dem Wert "männlich" oder "weiblich" zugewiesen werden.

3. Laden Sie das Programmpaket (Package) von „tidyverse“ auf das R herunter und führen

Sie das aus!

İnput:

```
library(tidyverse)
```

output:

```
-- Attaching packages ---------------------------------
----- tidyverse 1.3.0 --

v ggplot2 3.3.2     v purrr   0.3.4

v tibble  3.0.4     v dplyr   1.0.2

v tidyr   1.1.2     v stringr 1.4.0

v readr   1.4.0     v forcats 0.5.0

-- Conflicts ------------------------------------------
tidyverse_conflicts() --

x dplyr::filter() masks stats::filter()
```

```
x dplyr::lag()    masks stats::lag()
```

4. Es gibt einen Datensatz im R, der „iris" heißt. Prüfen Sie diesen Datensatz und lesen Sie

die R-Dokumente. Worum geht es? Zeigen Sie die Einzelheiten des Datensatzes an,

indem Sie geeigneten Codes benutzen. Z.B., wie viele Beobachtungen und Variablen gibt

es in dem Datensatz, gibt es die kategorischen, kontinuierlichen und/oder ordinalen

Variablen und wie heißen sie? (Schreiben Sie die Codes die Sie benutzen haben um zu

beantworten) (Datensatz = iris)

Der Iris-Datensatz ist ein in R integrierter Datensatz, der Messungen zu 4 verschiedenen Attributen (in Zentimetern) für 50 Blumen von 3 verschiedenen Arten enthält.

```
data(iris)
head(iris)
summary(iris)
dim(iris)
names(iris)
```

Run (Ctrl-Enter)

```
  Sepal.Length Sepal.Width Petal.Length Petal.Width Species
1          5.1         3.5          1.4         0.2  setosa
2          4.9         3.0          1.4         0.2  setosa
3          4.7         3.2          1.3         0.2  setosa
4          4.6         3.1          1.5         0.2  setosa
5          5.0         3.6          1.4         0.2  setosa
6          5.4         3.9          1.7         0.4  setosa
  Sepal.Length    Sepal.Width     Petal.Length    Petal.Width
 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.   :0.100
 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st Qu.:0.300
 Median :5.800   Median :3.000   Median :4.350   Median :1.300
 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean   :1.199
 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd Qu.:1.800
 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.   :2.500
       Species
 setosa    :50
 versicolor:50
 virginica :50



[1] 150   5
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length" "Petal.Width"  "Species"
```

İnput:

```
data(iris)

head(iris)

summary(iris)

dim(iris)

names(iris)
```

output:

```
  Sepal.Length Sepal.Width Petal.Length Petal.Width
Species

1          5.1         3.5          1.4         0.2
setosa

2          4.9         3.0          1.4         0.2
setosa

3          4.7         3.2          1.3         0.2
setosa
```

```
4          4.6          3.1           1.5          0.2
setosa

5          5.0          3.6           1.4          0.2
setosa

6          5.4          3.9           1.7          0.4
setosa


  Sepal.Length    Sepal.Width     Petal.Length
Petal.Width


 Min.   :4.300   Min.   :2.000   Min.   :1.000   Min.
:0.100

 1st Qu.:5.100   1st Qu.:2.800   1st Qu.:1.600   1st
Qu.:0.300

 Median :5.800   Median :3.000   Median :4.350   Median
:1.300

 Mean   :5.843   Mean   :3.057   Mean   :3.758   Mean
:1.199

 3rd Qu.:6.400   3rd Qu.:3.300   3rd Qu.:5.100   3rd
Qu.:1.800

 Max.   :7.900   Max.   :4.400   Max.   :6.900   Max.
:2.500


       Species

 setosa    :50

 versicolor:50

 virginica :50




[1] 150   5
```

```
[1] "Sepal.Length" "Sepal.Width"  "Petal.Length"
"Petal.Width"  "Species"
```

5. Wählen Sie Ihrem Wunsch nach zwei Variablen aus dem Datensatz von „iris“ aus und

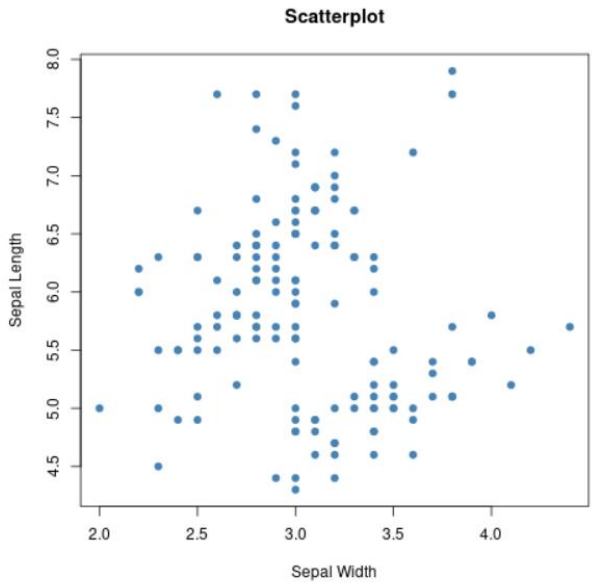
zeigen Sie alle zu diesen Variablen entsprechenden Beobachtungen in einem Plot an und

interpretieren Sie die Beziehungen zwischen diesen Variablen. (Datensatz = iris)

input:

```
plot(iris$Sepal.Width, iris$Sepal.Length,
     col='steelblue',
     main='Scatterplot',
     xlab='Sepal Width',
     ylab='Sepal Length',
     pch=19)
```

output:



6-Ich möchte die allgemeine Verteilung der Variablen von „sepal length and petal length“

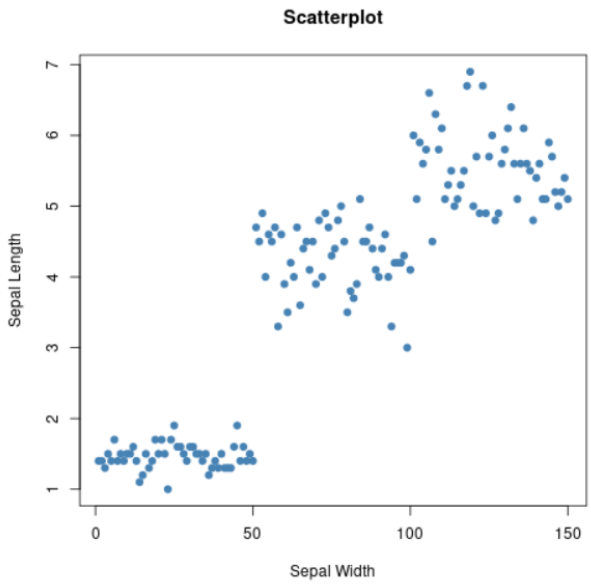
in dem „iris“ Datensatz laut der Arten (species) wahrnehmen, ohne die Beobachtungen

zu sehen. Wie würden Sie es visualisieren? (Datensatz = iris)

input:

```
plot(iris$Petal.Length,
     col='steelblue',
     main='Scatterplot',
     xlab='Sepal Width',
     ylab='Sepal Length',
     pch=19)
```

output:

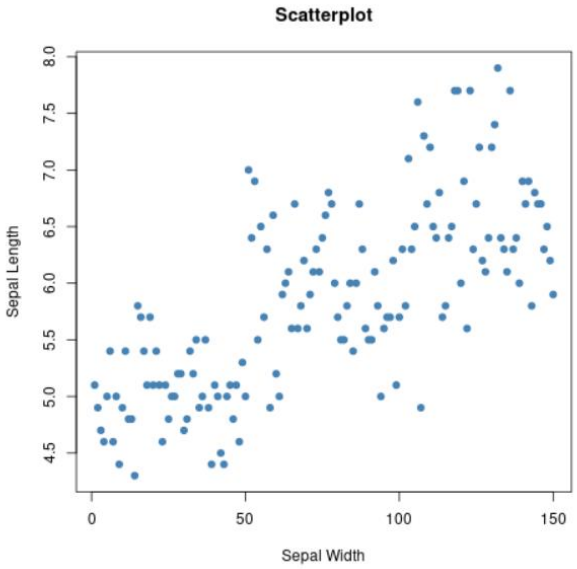


İnput:

```
plot(iris$Sepal.Length,
     col='steelblue',
     main='Scatterplot',
     xlab='Sepal Width',
```

```
     ylab='Sepal Length',
     pch=19)
```

output:



7. „mpg“ ist ein Datensatz, in dem die verschiedenen Autos und denen Eigenschaften wie

z.B. die Zahl der Zylinder, der Hubraum, die Klasse, die Marke usw. sich befinden.

Nehmen Sie nur die Autos mit „manual(m5)“ Transmission und versuchen Sie, eine

Grafik zu erstellen, in der die Autos nach ihrem Treibstofftyp getrennt sind. Die Autos in

jeder Gruppe sollen auch basierend auf ihrer Klasse präsentiert werden. Man sollte die

Anzahl der Autos mit dieser Grafik verglichen können. (Datensatz = mpg)

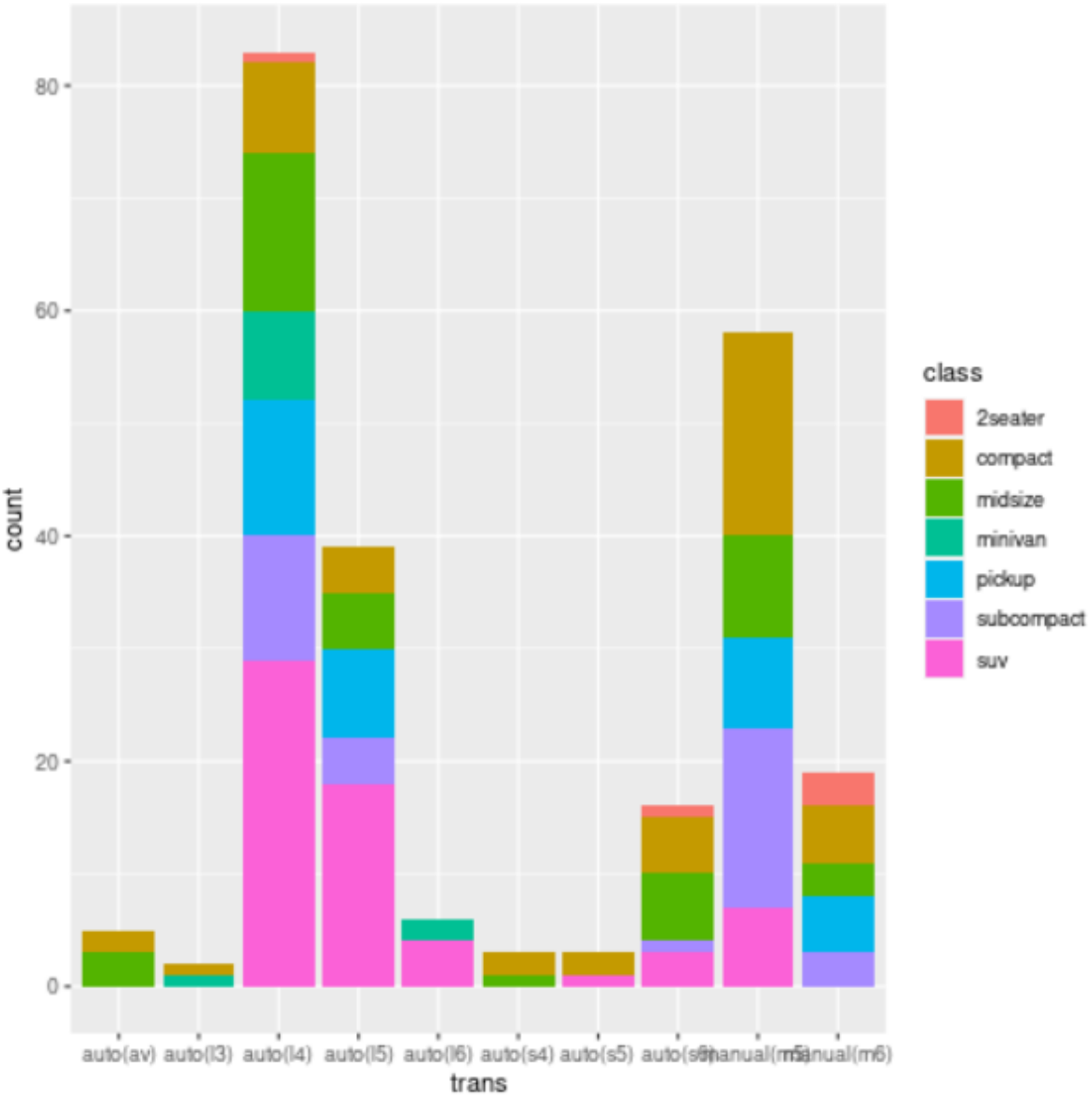
İnput:

```
library(tidyverse)

head(mpg)

mpg %>%
```

```
  filter(trans== "manual(m5)")
table(mpg$manufacturer) #Gives the list of manufacturer


ggplot(data=mpg) +
geom_bar(mapping=aes(x=trans,fill=class))
```

```
-- Attaching packages --------------------------------------- tidyverse 1.3.0 --
v ggplot2 3.3.2     v purrr   0.3.4
v tibble  3.0.4     v dplyr   1.0.2
v tidyr   1.1.2     v stringr 1.4.0
v readr   1.4.0     v forcats 0.5.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
# A tibble: 6 x 11
  manufacturer model displ  year   cyl trans      drv     cty   hwy fl    class
  <chr>        <chr> <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr>
1 audi         a4      1.8  1999     4 auto(l5)   f        18    29 p     compa~
2 audi         a4      1.8  1999     4 manual(m5) f        21    29 p     compa~
3 audi         a4      2    2008     4 manual(m6) f        20    31 p     compa~
4 audi         a4      2    2008     4 auto(av)   f        21    30 p     compa~
5 audi         a4      2.8  1999     6 auto(l5)   f        16    26 p     compa~
6 audi         a4      2.8  1999     6 manual(m5) f        18    26 p     compa~
# A tibble: 58 x 11
   manufacturer model     displ  year   cyl trans  drv     cty   hwy fl    class
   <chr>        <chr>     <dbl> <int> <int> <chr>  <chr> <int> <int> <chr> <chr>
 1 audi         a4          1.8  1999     4 manua~ f        21    29 p     comp~
 2 audi         a4          2.8  1999     6 manua~ f        18    26 p     comp~
 3 audi         a4 quatt~   1.8  1999     4 manua~ 4        18    26 p     comp~
 4 audi         a4 quatt~   2.8  1999     6 manua~ 4        17    25 p     comp~
 5 dodge        dakota p~   3.9  1999     6 manua~ 4        14    17 r     pick~
 6 dodge        dakota p~   5.2  1999     8 manua~ 4        11    17 r     pick~
 7 dodge        ram 1500~   5.2  1999     8 manua~ 4        11    16 r     pick~
 8 ford         explorer~   4    1999     6 manua~ 4        15    19 r     suv
 9 ford         f150 pic~   4.2  1999     6 manua~ 4        14    17 r     pick~
10 ford         f150 pic~   4.6  1999     8 manua~ 4        13    16 r     pick~
# ... with 48 more rows
```

```
      audi  chevrolet      dodge       ford      honda    hyundai       jeep
        18         19         37         25          9         14          8
land rover    lincoln    mercury     nissan    pontiac     subaru     toyota
         4          3          4         13          5         14         34
volkswagen
        27
```



8. Wenn die Blüten nach ihrer Kelchblattlänge (sepal.length) und Kelchblattbreite

(sepal.width) dargestellt werden, werden diejenigen, denen Blütenblattlänge (petal

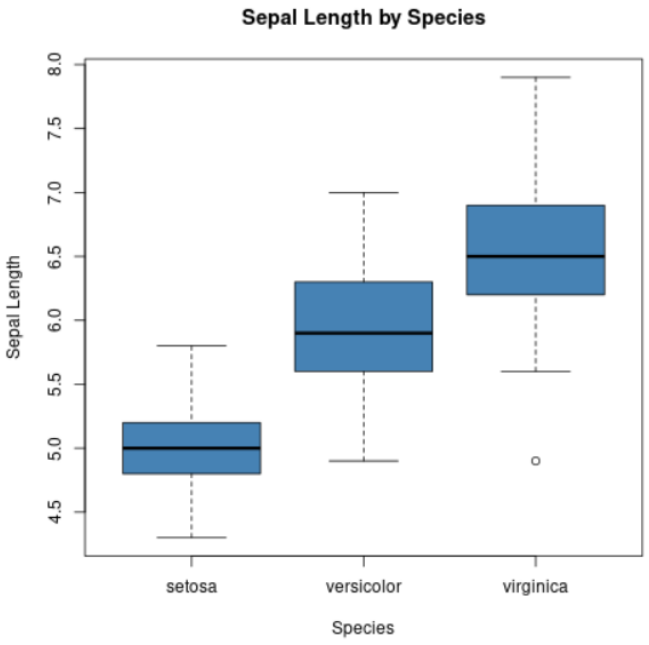
length) kürzer als 2 sind, in die Setosa-Gruppe eingeordnet. Wie würden Sie diesen

Fakt visualisieren? (Datensatz = iris)

İnput:

```
data(iris)
boxplot(Sepal.Length~Species,
        data=iris,
        main='Sepal Length by Species',
        xlab='Species',
        ylab='Sepal Length',
        col='steelblue',
        border='black')
```

output:



,

9. Datensatz „vehicles” wurde im R bereits für Sie zur Verfügung gestellt. Sie müssen aber

erstmal „fueleconomy“ Paketprogramm herunterladen. Machen Sie das bitte! Nachdem

Sie Paketprogramm heruntergeladen haben, führen Sie das aus. Sie sollen nun bereit sein,

an dem „vehicles“ zu arbeiten. Erklären Sie kurz, was gibt es im Datensatz (versuchen

Sie verstehen, worum es geht)? Wie würden Sie den durchschnittlichen Verbrauch der

Autos, denen Anzahl der Zylinder weniger als 6 und denen Motorhubräume („displ“)

kleiner als 2 sind, basierend auf dem Fahrmodus („drive“) anzeigen? (Datensatz =

vehicles)!

İnput:

```
library(tidyverse)
head(mpg)
str(mpg)
mpg %>%
  filter(cyl <= 6)
mpg %>%
  filter(displ <= 2)
```

output:

```
-- Attaching packages --------------------------------------- tidyverse 1.3.0 --
v ggplot2 3.3.2     v purrr   0.3.4
v tibble  3.0.4     v dplyr   1.0.2
v tidyr   1.1.2     v stringr 1.4.0
v readr   1.4.0     v forcats 0.5.0
-- Conflicts ------------------------------------------ tidyverse_conflicts() --
x dplyr::filter() masks stats::filter()
x dplyr::lag()    masks stats::lag()
# A tibble: 6 x 11
  manufacturer model displ  year   cyl trans      drv     cty   hwy fl    class
  <chr>        <chr> <dbl> <int> <int> <chr>      <chr> <int> <int> <chr> <chr>
1 audi         a4      1.8  1999     4 auto(l5)   f        18    29 p     compa~
2 audi         a4      1.8  1999     4 manual(m5) f        21    29 p     compa~
3 audi         a4      2    2008     4 manual(m6) f        20    31 p     compa~
4 audi         a4      2    2008     4 auto(av)   f        21    30 p     compa~
5 audi         a4      2.8  1999     6 auto(l5)   f        16    26 p     compa~
6 audi         a4      2.8  1999     6 manual(m5) f        18    26 p     compa~
tibble [234 x 11] (S3: tbl_df/tbl/data.frame)
```

```
 $ manufacturer: chr [1:234] "audi" "audi" "audi" "audi" ...
 $ model       : chr [1:234] "a4" "a4" "a4" "a4" ...
 $ displ       : num [1:234] 1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
 $ year        : int [1:234] 1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
 $ cyl         : int [1:234] 4 4 4 4 6 6 6 4 4 4 ...
 $ trans       : chr [1:234] "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
 $ drv         : chr [1:234] "f" "f" "f" "f" ...
 $ cty         : int [1:234] 18 21 20 21 16 18 18 18 16 20 ...
 $ hwy         : int [1:234] 29 29 31 30 26 26 27 26 25 28 ...
 $ fl          : chr [1:234] "p" "p" "p" "p" ...
 $ class       : chr [1:234] "compact" "compact" "compact" "compact" ...
# A tibble: 164 x 11
   manufacturer model    displ  year   cyl trans   drv     cty   hwy fl    class
   <chr>        <chr>    <dbl> <int> <int> <chr>   <chr> <int> <int> <chr> <chr>
 1 audi         a4         1.8  1999     4 auto(l~ f        18    29 p     comp~
 2 audi         a4         1.8  1999     4 manual~ f        21    29 p     comp~
 3 audi         a4         2    2008     4 manual~ f        20    31 p     comp~
 4 audi         a4         2    2008     4 auto(a~ f        21    30 p     comp~
 5 audi         a4         2.8  1999     6 auto(l~ f        16    26 p     comp~
 6 audi         a4         2.8  1999     6 manual~ f        18    26 p     comp~
 7 audi         a4         3.1  2008     6 auto(a~ f        18    27 p     comp~
 8 audi         a4 quat~   1.8  1999     4 manual~ 4        18    26 p     comp~
 9 audi         a4 quat~   1.8  1999     4 auto(l~ 4        16    25 p     comp~
10 audi         a4 quat~   2    2008     4 manual~ 4        20    28 p     comp~
# ... with 154 more rows
# A tibble: 43 x 11
```

```
   manufacturer model   displ  year   cyl trans   drv     cty   hwy fl    class
   <chr>        <chr>   <dbl> <int> <int> <chr>   <chr> <int> <int> <chr> <chr>
 1 audi         a4        1.8  1999     4 auto(l~ f        18    29 p     compa~
 2 audi         a4        1.8  1999     4 manual~ f        21    29 p     compa~
 3 audi         a4        2    2008     4 manual~ f        20    31 p     compa~
 4 audi         a4        2    2008     4 auto(a~ f        21    30 p     compa~
 5 audi         a4 qua~   1.8  1999     4 manual~ 4        18    26 p     compa~
 6 audi         a4 qua~   1.8  1999     4 auto(l~ 4        16    25 p     compa~
 7 audi         a4 qua~   2    2008     4 manual~ 4        20    28 p     compa~
 8 audi         a4 qua~   2    2008     4 auto(s~ 4        19    27 p     compa~
 9 honda        civic     1.6  1999     4 manual~ f        28    33 r     subco~
10 honda        civic     1.6  1999     4 auto(l~ f        24    32 r     subco~
# ... with 33 more rows
```

10-Schreiben Sie den das unten gegebene Plot erzeugenden Code und interpretieren Sie es!

Sie müssen „diamonds" Datensatz anwenden aber Achtung! Dieser Plot beinhaltet nur

die Diamanten mit "SI1 clarity" (Datensatz = diamonds)

input:

```
library(ggplot2)

p<-ggplot(data=filter(diamonds, clarity == "SI1"),
aes(x=carat, y=price))

p+facet_wrap(~cut)+geom_point(aes(color=color) )
```

output: