Data Visualization with R (ggplot)

February 24, 2018

loading ggplot2

Let's get started right away by the loading ggplot2 package and reading in our dataset.

Data sets and R Code is available

`https://github.com/kiat/R-Examples`

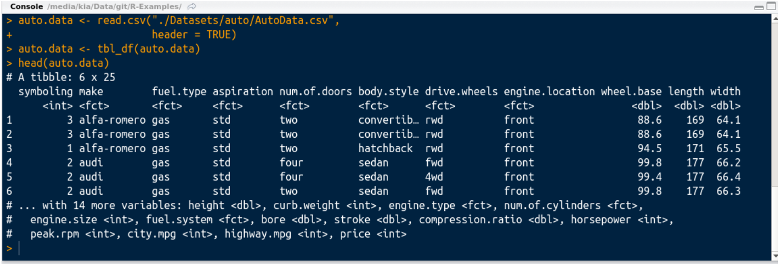
```
### Install packages if you don't have them yet
### Typical install:
# install.packages('gpplot2')
# install.packages('dplyr')

### Load packages
# Load packages
library(ggplot2)
library(stats)
library(base)
library(dplyr)


# setwd("YOUR-WORKING-PATH")

# Load personal copy
# library(ggplot2,lib.loc="/path/to/myfolder")
# library(dplyr,lib.loc="/path/to/myfolder")
# Read In data
auto.data <- read.csv("./data/auto/AutoData.csv",
                      header = TRUE)
# tbl_df() isn't necessary here
# It helps to display the data more clearly
auto.data <- tbl_df(auto.data)
```

Auto Data



```
> auto.data <- read.csv("./Datasets/auto/AutoData.csv",
+                       header = TRUE)
> auto.data <- tbl_df(auto.data)
> head(auto.data)
# A tibble: 6 x 25
  symboling make        fuel.type aspiration num.of.doors body.style drive.wheels engine.location wheel.base length width
      <int> <fct>       <fct>     <fct>      <fct>        <fct>      <fct>        <fct>                <dbl>  <dbl> <dbl>
1         3 alfa-romero gas       std        two          convertib… rwd          front                 88.6    169  64.1
2         3 alfa-romero gas       std        two          convertib… rwd          front                 88.6    169  64.1
3         1 alfa-romero gas       std        two          hatchback  rwd          front                 94.5    171  65.5
4         2 audi        gas       std        four         sedan      fwd          front                 99.8    177  66.2
5         2 audi        gas       std        four         sedan      4wd          front                 99.4    177  66.4
6         2 audi        gas       std        two          sedan      fwd          front                 99.8    177  66.3
# ... with 14 more variables: height <dbl>, curb.weight <int>, engine.type <fct>, num.of.cylinders <fct>,
#   engine.size <int>, fuel.system <fct>, bore <dbl>, stroke <dbl>, compression.ratio <dbl>, horsepower <int>,
#   peak.rpm <int>, city.mpg <int>, highway.mpg <int>, price <int>
> 
```

Run the following to get a quick glimpse of the data

```r
# Find the dimensions
dim(auto.data)
# Look at the structure
str(auto.data)
# Examine the top
head(auto.data)
# Find out about a function
?str
```

Data Exploration

- When looking at a new data set, exploration is key.
- What types of variables do we have?
- What types of relationships do you expect to see between variables?
- Does your intuition check out? If not, why not?
- Do we observe anomalous behavior?

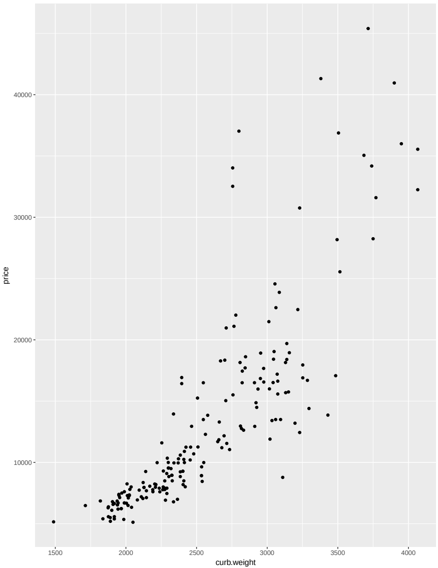
Scatter Plots

One of the simpler plots we can make is a scatter plot between to continuous variables.

```
# qplot is convenient front end for the more powerful,
# but slightly more complicated ggplot() function.

qplot(curb.weight,price,data=auto.data)
```

Scatter Plots



Power of ggplot

The true power of ggplot comes from its ability to easily visualize relationships between many variables.

The main ingredients we'll be using are:

1. aesthetics
2. facets
3. geoms

ggplot - Aesthetics

Aesthetics control many of the plot's visual properties

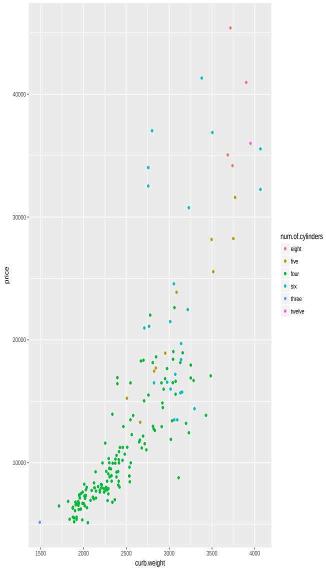
Importantly these visual properties may be mapped directly to variables

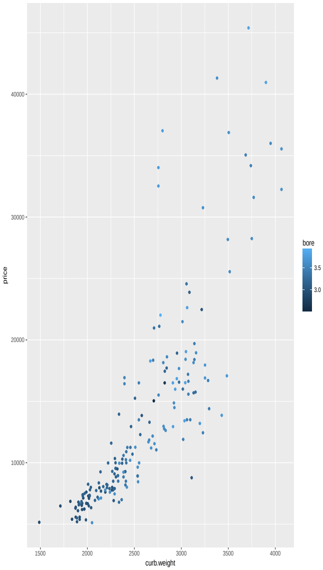
Scatter Plots

```
# map color to factor/categorical variable
qplot(curb.weight,
      price,
      data=auto.data,
      color=num.of.cylinders)

# map color to continuous variable
qplot(curb.weight,
      price,
      data=auto.data,
      color=bore)
```

Scatter Plots





Aesthetics

There are many other aesthetics besides color. Some we'll encounter are:

Not all aesthetics work with both categorical and continuous variables (like color did)

Also only a certain subset of aesthetics will be available for each plot type (geom)

1. color
2. size
3. shape
4. fill

Aesthetics

See how the following aesthetics behave with the scatter plot. Feel free to change the variables in the scatter plot

```
qplot(curb.weight,
        price,
        data=auto.data,
        size=horsepower)

qplot(curb.weight,
        price,
        data=auto.data,
        shape=drive.wheels)
```

Facets

Facets represent another way of visualizing the effect of factor/categorical variables

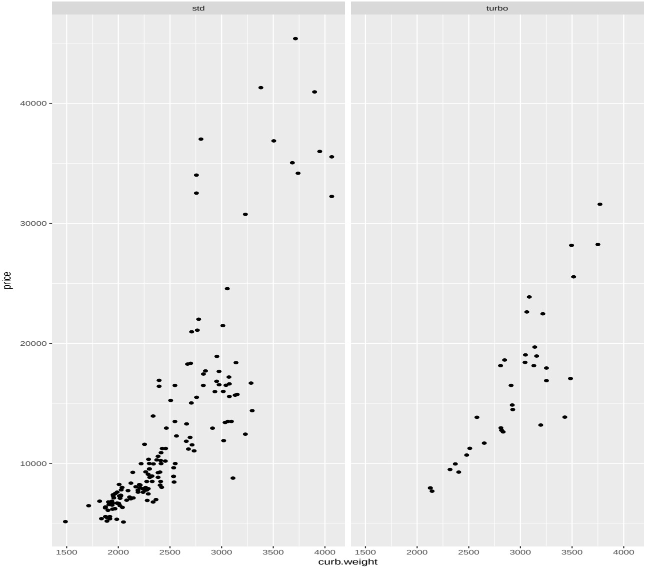
Facets enable us to get a separate plot for each level/category

Facets Example

Try out a faceting example:

```
qplot(curb.weight,
                price,
                data=auto.data) + facet_wrap(˜aspiration)
```

Facets Example



Facets

Note facet_wrap gives a separate plot for each category

Also note how we incorporated the behavior of facet_wrap:
via the + operator

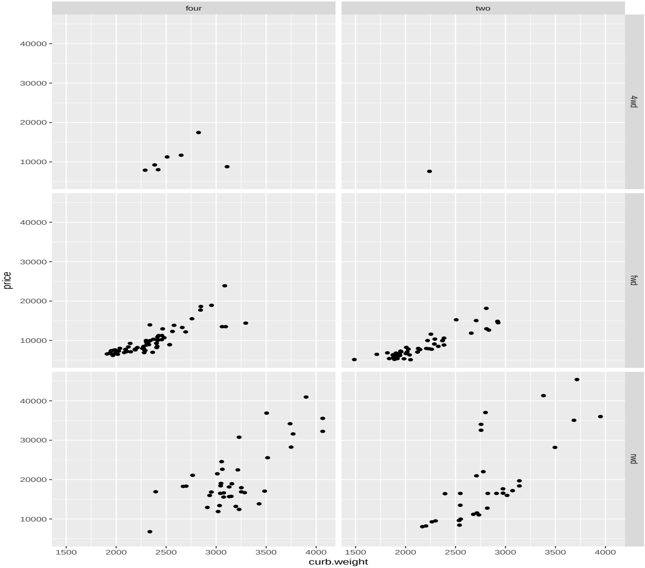
This is one of the main strengths of ggplot: plots are built up in intuitive layers

Facets

Also available is facet_grid for examining the interaction between two categorical variables:

```
qplot(curb.weight,
      price,
      data=auto.data) +
      facet_grid(drive.wheels˜num.of.doors)
```

Facets Grid Example



Facets

Try the following:

```
qplot(curb.weight,
      price,
      data=auto.data) + facet_grid(.~drive.wheels)

qplot(curb.weight,
      price,
      data=auto.data) + facet_grid(drive.wheels~.)

qplot(curb.weight,
      price,
      data=auto.data,
      color=num.of.doors) + facet_grid(drive.wheels~.)
```

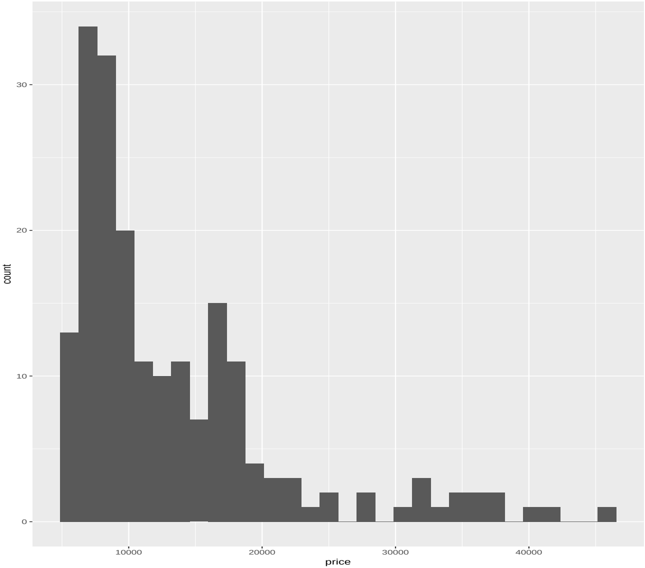
geom_histogram

Let's check out another geom: geom_histogram

```
# geom_histogram operates with a single continuous variable.
# Let's look at price
qplot(price,
        data=auto.data,
        geom='histogram')

# or via qplot's defaults
qplot(price,data=auto.data)
```

Histogram



geom_histogram

Note the warning concerning binwidth

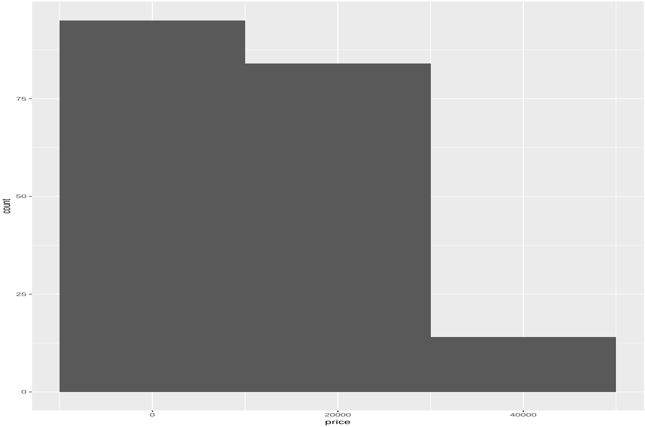
The binwidth chosen can dramatically impact how we visually interpret the distribution

It's best to experiment with values to get a feel for the data

We can alter the binwidth by passing the option to qplot

```
qplot(price,
        data=auto.data,
        geom='histogram',
        binwidth=20000)
```

Histogram



Histogram

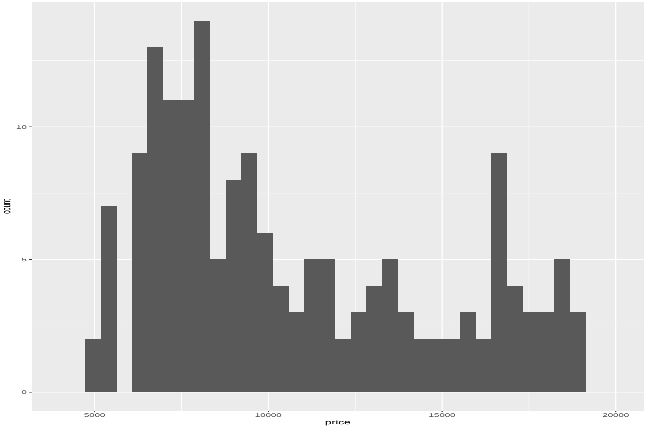
Note our price distribution is a bit skewed

Perhaps we are not interested in higher priced ($\geq$ 20,000 say) cars

We can limit our plot cars with lower price by setting limits

```
qplot(price,
      data=auto.data,
      geom='histogram',
      binwidth=450) +
          xlim(4000,20000)
```

Histogram



Histogram

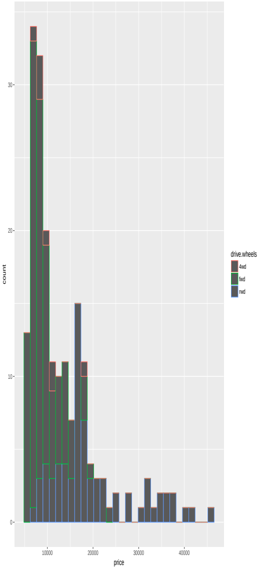
Just like our point geom, histogram too has aesthetics. Try the following

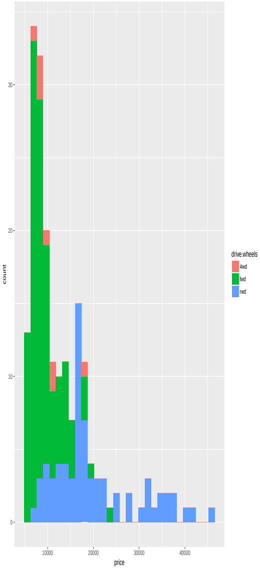
```
qplot(price,
      data=auto.data,

      color=drive.wheels)
qplot(price,
      data=auto.data,
      fill=drive.wheels)
```

Color and Fill Plots

Which one do like the best? Do you like either? How might we make it better?



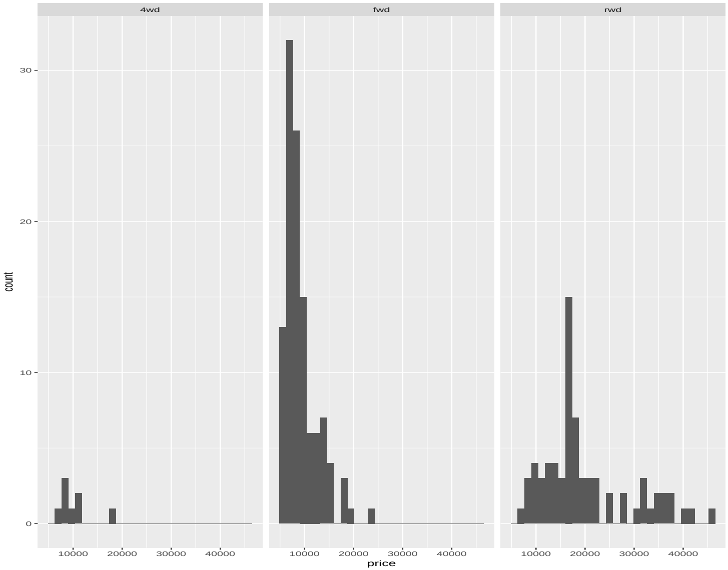


Histogram with facets

The colors help but the figure is a bit busy.
We can try faceting instead:

```
qplot(price,
        data=auto.data) +
        facet_wrap(~drive.wheels)
```

Histogram with facets



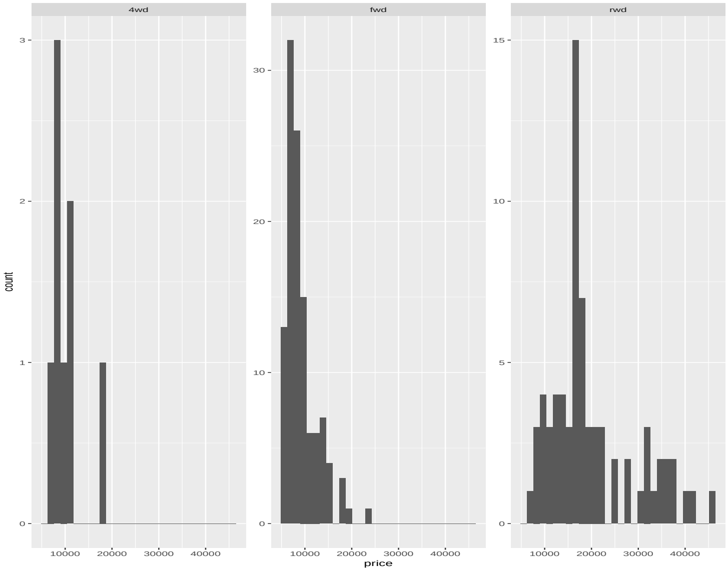
Histogram with facets

This helps us separate out the categorical variables much easier.

Note the counts vary quite a bit among the different classes, but yet the count axis is the same for all. We can change this by modifying the facet_wrap call:

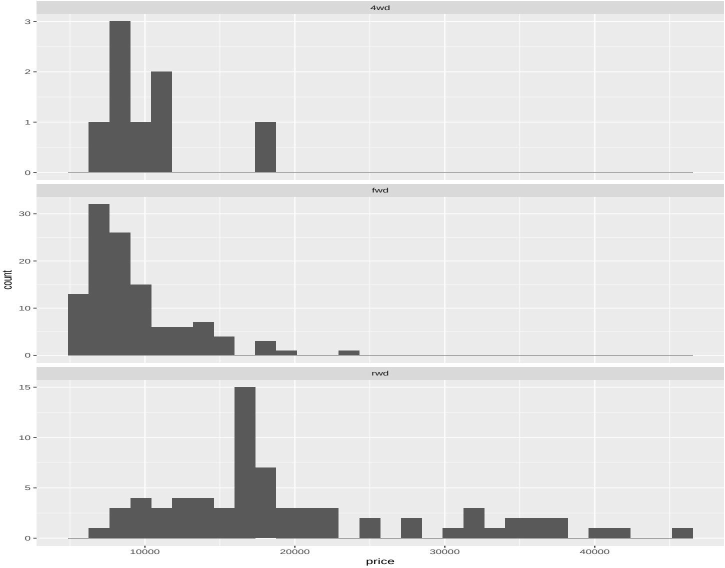
```
qplot(price,
        data=auto.data) +
        facet_wrap(~drive.wheels,
        scales = 'free_y')
```

Histogram with facets



Histogram with facets

More useful options. For example nrow=3



More geoms

- There are many other geoms besides point and histogram. Try ??geom to see a list.
- Different geoms operate with different (combinations of) data types (i.e. categorical or continuous).
- As is characteristic of ggplot, geoms can be layered to create plots of increasing detail/complexity.

Layering of ggplot, geoms

```
qplot(price,data=auto.data,
      geom='density')
qplot(price,
      ..density.., # don't use counts
      data=auto.data,
      geom='histogram') +
  geom_density()
qplot(height,price,
      data=auto.data,
      geom='density2d')
qplot(height,price,
      data=auto.data)+
  geom_density2d()
```

geoms boxplot

- Can you guess the geom for creating a boxplot?
- Create a boxplot displaying price for each of the drive.wheels categories

geoms boxplot

```
qplot(drive.wheels,
        price,
        data=auto.data,
        geom='boxplot')
```

References and Additional Info

- ggplot2 documentation: `http://docs.ggplot2.org/current/`
- Hadley's ggplot2 book: `http://ggplot2.org/book/`
- RStudio ggplot cheatsheet: `http://www.rstudio.com/wp-content/uploads/2015/03/ggplot2-cheatsheet.png`