Analysis of Ecosystems homework week 4

Distributions

The solution

```
# Load just these packages, as per the template
pacman::p_load(s20x, plyr, ggplot2)
```

Data preparation

```
## Classes 'tbl_df', 'tbl' and 'data.frame':    234 obs. of  11 variables:
##  $ manufacturer: chr  "audi" "audi" "audi" "audi" ...
##  $ model       : chr  "a4" "a4" "a4" "a4" ...
##  $ displ       : num  1.8 1.8 2 2 2.8 2.8 3.1 1.8 1.8 2 ...
##  $ year        : int  1999 1999 2008 2008 1999 1999 2008 1999 1999 2008 ...
##  $ cyl         : int  4 4 4 4 6 6 6 4 4 4 ...
##  $ trans       : chr  "auto(l5)" "manual(m5)" "manual(m6)" "auto(av)" ...
##  $ drv         : chr  "f" "f" "f" "f" ...
##  $ cty         : int  18 21 20 21 16 18 18 18 16 20 ...
##  $ hwy         : int  29 29 31 30 26 26 27 26 25 28 ...
##  $ fl          : chr  "p" "p" "p" "p" ...
##  $ class       : chr  "compact" "compact" "compact" "compact" ...
```

Continuous data

Distribution

Make a graph

```
( hp.gg <-
    ggplot(mpg, aes(x=cty)) + theme_bw() +
    geom_histogram(aes(y=..density..),
                   binwidth=1,
                   colour="black", fill="lightgreen") +
    geom_density(alpha=.2, fill="#FF6666") +
    xlab("City fuel economy (mpg)") )
```

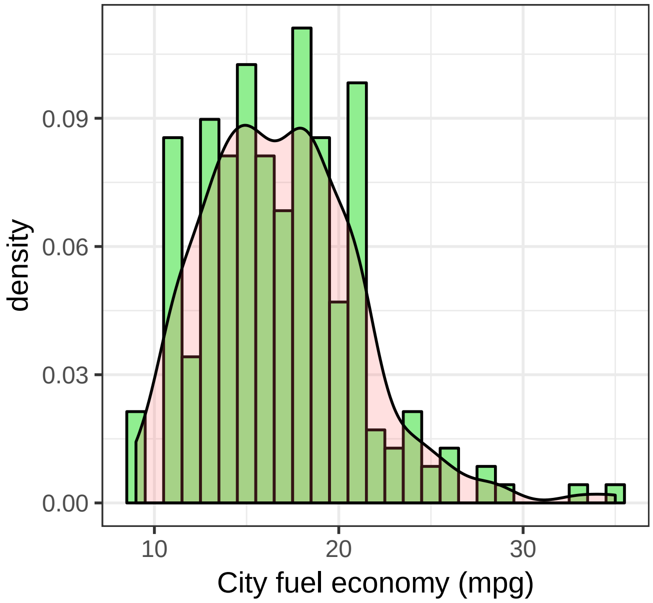

Figure 1: Histogram and density plot for city fuel economy.

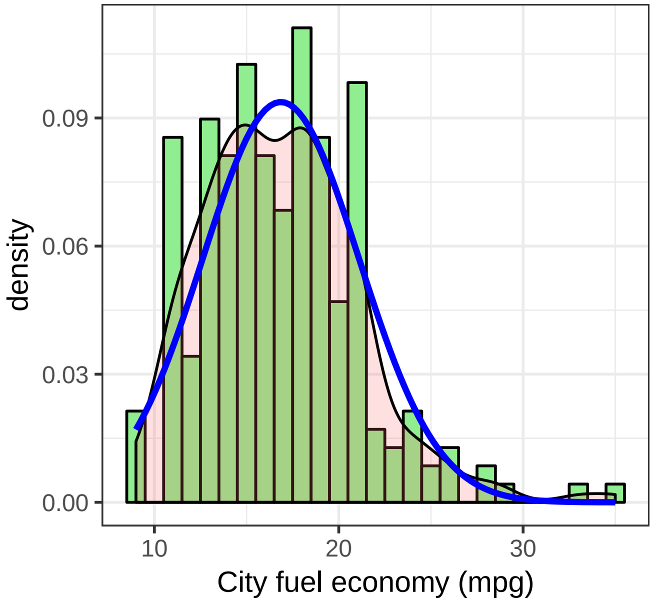

Figure 2: Histogram and density plot for city fuel economy with normal Probability Density Function.

Interpret the graph

- The histogram (Fig. 1) represents the number of times each value on the X axis occurs in the dataset. It is interpeted as raw data (counts).
- The density estimate interpolates between the actual data to estimate the general shape of the sample data's distribution. It is interpreted as an estimation of the sample's distribution.
- These data are all positive, and slightly right-skewed (longer right tail).

Probability Density Function

Add PDF curve to the graph

```
hp.gg + stat_function(data=mpg,
                      fun = dnorm,
                      args=list(mean=mean(mpg$cty),
                                sd=sd(mpg$cty)),
                      colour="blue", size=1.1)
```

Interpret the graph

- The new curve (Fig. 2) represents the theoretical normal distribution based on the moments (mean and sd) from the sample data. It can be interpreted as the shape of the population's distribution from which these samples came, and is thus the distribution a statistical model that assumes a normal distribution will use.
- The normal distribution does not fit the data well. The tail on the right forces the symmetrical normal distribution to assume the population has many more low values that do not occur in the sample.

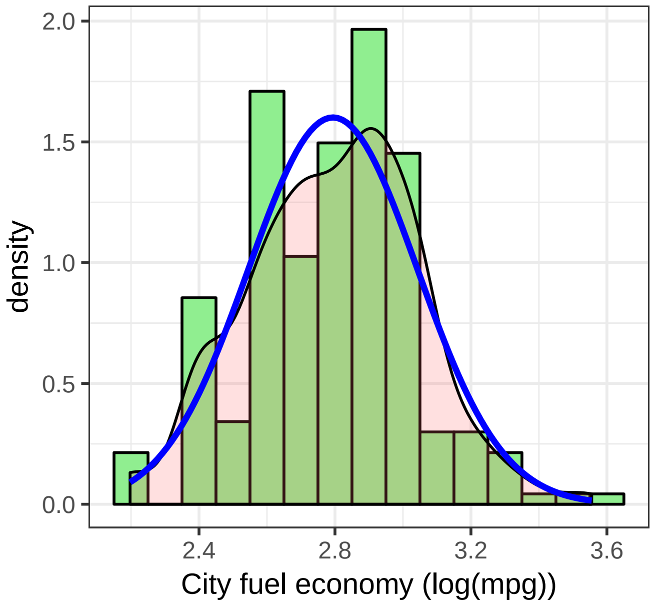

Figure 3: Histogram, density, and PDF of city fuel economy on a log scale.

Graph a different continuous distribution

Symmetry could be improved by log-transforming the data.

Transformation graph

```
 logmean <- mean(log(mpg$cty))
 logsd   <- sd(log(mpg$cty))
ggplot(mpg, aes(x=log(cty))) + theme_bw() +
  geom_histogram(aes(y=..density..),
                 binwidth=0.1,
                 colour="black", fill="lightgreen") +
  geom_density(alpha=.2, fill="#FF6666") +
  xlab("City fuel economy (log(mpg))") +
  stat_function(data=mpg,
                fun = dnorm,
                args=list(mean=logmean,
                          sd=logsd),
                colour="blue", size=1.1)
```

Interpretation

The log transformation increased the symmetry of the cty variable (Fig. 3). Now the data better meet the assumptions of models like ANOVA that assume data have a normal distribution.

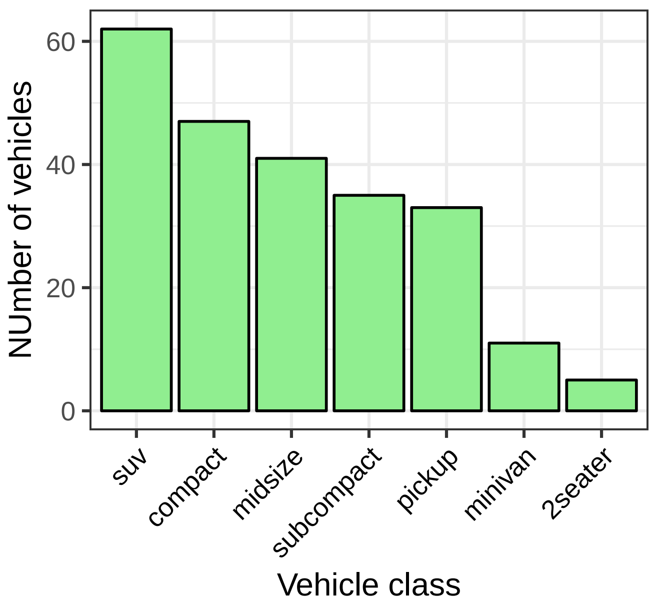

Figure 4: Number (counts) of vehicles by class.

Discrete data

Distribution

```
class.ct <- count(mpg, vars="class") # New d.f. of counts/group
colnames(class.ct)[[2]] <- "count"
class.ct$freq <- round(with(class.ct, # create a frequency column
                            count/sum(count)), 3)
class.ct <- class.ct[with(class.ct, order(-freq)), ] # Re-order by freq
str(class.ct)
```

```
## 'data.frame':    7 obs. of  3 variables:
##  $ class: chr  "suv" "compact" "midsize" "subcompact" ...
##  $ count: int  62 47 41 35 33 11 5
##  $ freq : num  0.265 0.201 0.175 0.15 0.141 0.047 0.021
```

Make a bar graph

```
ggplot(class.ct, aes(x=reorder(class,-count, max), y=count)) +
  geom_bar(stat="identity",
           colour="black", fill="lightgreen") +
  labs(x="Vehicle class",
       y="NUmber of vehicles") +
  theme_bw(12) +
  theme(axis.text.x = element_text(color="black", angle=45, hjust = 1))
```

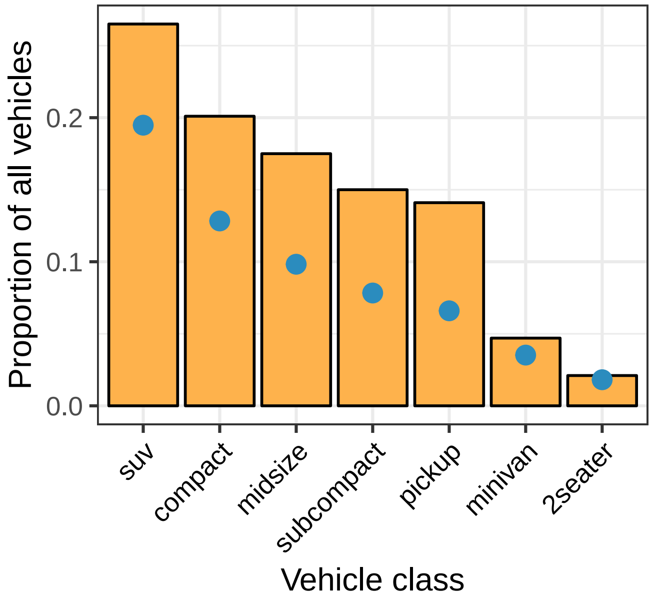

Figure 5: Relative frequency of vehicles by class, with negative binomial PMF. When the `prob=` argument is used in `dnbinom`, try `size=1` first, since 1.0 represents the sum of all probabilities.

Add PMF

```
ggplot(class.ct, aes(x=reorder(class,-freq, max), y=freq)) +
geom_bar(stat="identity",
         colour="black", fill="#feb24c") +
labs(x="Vehicle class",
     y="Proportion of all vehicles") +
theme_bw(12) +
theme(axis.text.x = element_text(color="black", angle=45, hjust = 1)) +
geom_point(data=transform(data.frame(x=1:7),
                                   y=dnbinom(1:7, size=1,
                                     prob=class.ct$freq)),
                          aes(x, y),
                          stat="identity",
                          color="#2b8cbe", size=3)
```

Interpret the graph

- The PMF (Fig. 5) is similar to the PDF (Fig. 2) in that it represents the theoretical distribution of these data—i.e., the probability that a given category would be selected from a random draw of the population from which these data were sampled. A difference is that it is applied to discrete data, so instead of a continuous line under which all probabilities sum to 1.0, the PMF is plotted as discrete probabilities for each category based on their relative frequency in the sample set.
- Random draws on a negative binomial distribution fit with the above parameters would likely draw SUVs most frequently and two-seater cars least frequently, as these are the relative values given by the PMF.