Inferential Stats - Tooth Growth Analysis

Introduction

A preliminary exploration of the `ToothGrowth` data in the `datasets` package.

We have been asked to compare the effects of supplements by hypothesis tests and/or confidence intervals.

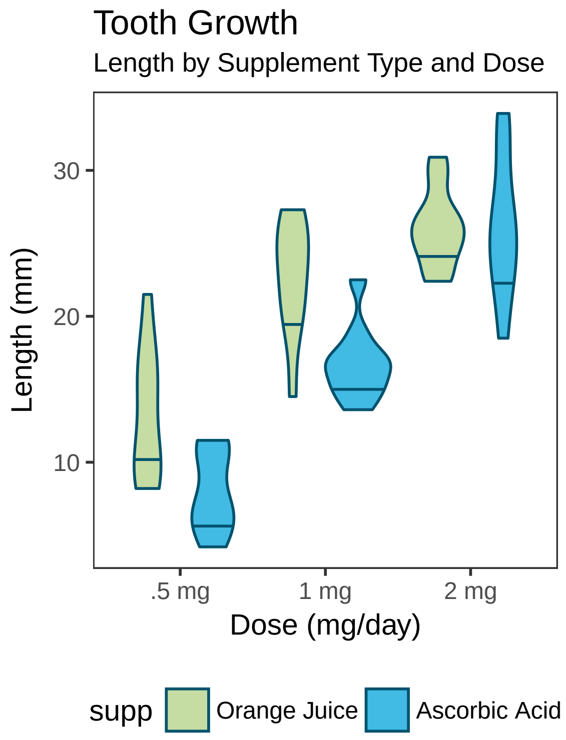
Basic Summary of Data

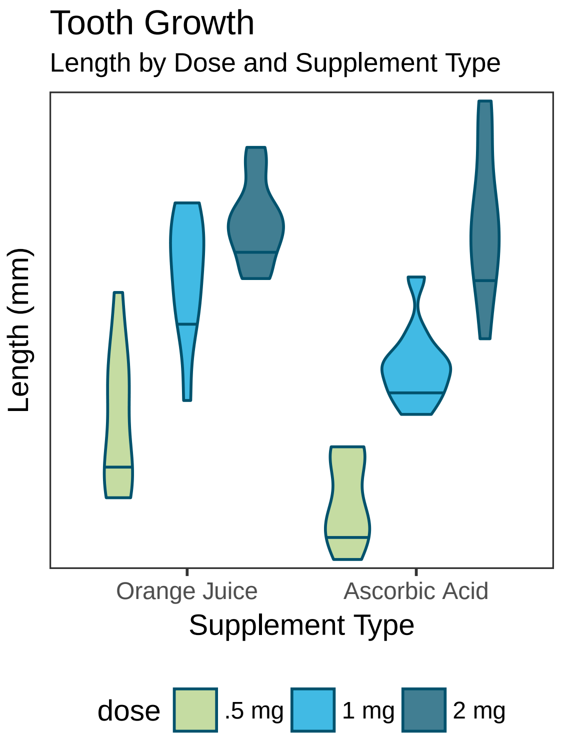
The experiment splits the animals into six groups of 10, each group receiving a different dose and/or method of delivery as follows:

Group	Dose (mg)	Supplement
A	0.5	Orange Juice
B	1.0	Orange Juice
C	2.0	Orange Juice
D	0.5	Ascorbic Acid
E	1.0	Ascorbic Acid
F	2.0	Ascorbic Acid

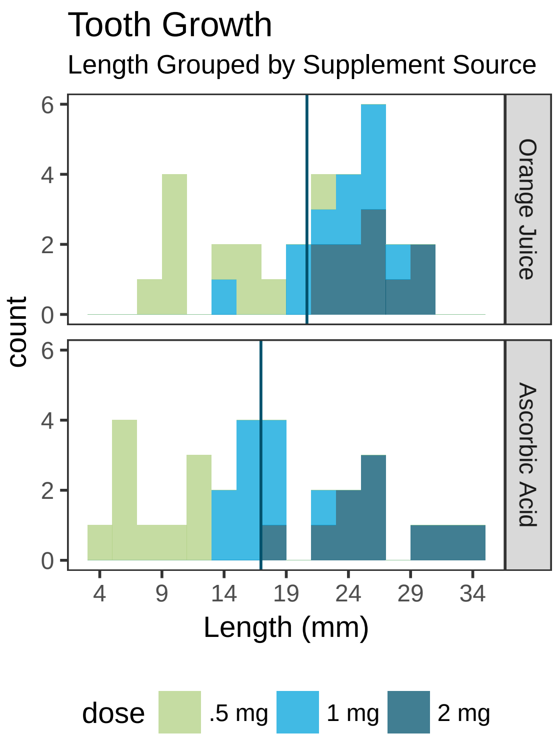
To simplify the calculations, I will treat the dose size as a factor. I explored the data using box plots and histograms to determine if the mean growth was effected by dosage, delivery mechanism, or some combination thereof.

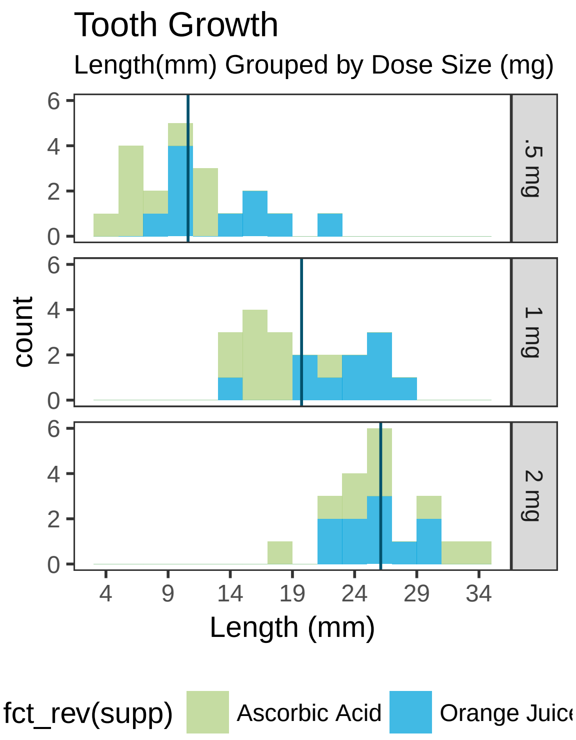
Violin Plots





Histograms grouping figures by length





Comparison via Confidence Intervals

I have decided to use confidence intervals to compare the effects of different dosages. A visual comparison seems to suggest that larger doses of Vitamin C do lead to increased tooth growth, and that the delivery method seems to have a larger effect at lower dosages.

By testing the confidence intervals of the three dosages, we can establish a connection between doses of vitamin C and the rate of tooth growth in guinea pigs.

Comparison via Hypothesis Testing

Assumptions and Conclusions

Appendix

```
meanlen <- plyr::ddply(tooth.growth, "supp", summarise, len.mean=mean(len))
fill <- c("#b2d183","#00a3dc","#01526d")

p <-  ggplot(data = tooth.growth, aes(x = dose, y = len, fill = supp)) +
      geom_violin(color = fill[3], draw_quantiles = .2, alpha = .75) +
      ggtitle("Tooth Growth",
              subtitle = "Length by Supplement Type and Dose") +
      theme_bw() + scale_fill_manual(values = fill) +
      theme(panel.grid.minor = element_blank(),
            panel.grid.major = element_blank(),
            panel.background = element_rect(color = 'white'),
            legend.position = 'bottom', legend.direction = 'horizontal') +
      scale_x_discrete(name = "Dose (mg/day)") +
      scale_y_continuous(name = "Length (mm)")

q <-  ggplot(data = tooth.growth, aes(x = supp, y = len, fill = dose)) +
      geom_violin(color = fill[3], draw_quantiles = .2, alpha = .75) +
      ggtitle("Tooth Growth",
              subtitle = "Length by Dose and Supplement Type") +
      theme_bw() + scale_fill_manual(values = fill) +
      theme(panel.grid.minor = element_blank(),
            panel.grid.major = element_blank(),
            panel.background = element_rect(color = 'white'),
            legend.position = 'bottom', legend.direction = 'horizontal') +
      scale_x_discrete(name = "Supplement Type") +
      scale_y_discrete(name = "Length (mm)")

r <-  ggplot(data = tooth.growth , aes(len) ) +
      geom_histogram(binwidth = 2, aes(fill = dose), alpha = .75) +
      facet_grid(supp ~ .) +
      ggtitle("Tooth Growth",
              subtitle = "Length Grouped by Supplement Source") +
      theme_bw() + scale_fill_manual(values = fill) +
      geom_vline(data = meanlen, aes(xintercept=len.mean),
               color = fill[3], linetype="solid", size=.5) +
      theme(panel.grid.minor = element_blank(),
            panel.grid.major = element_blank(),
            panel.background = element_rect(color = 'white'),
            legend.position = 'bottom', legend.direction = 'horizontal') +
        scale_x_continuous(name = "Length (mm)")

meanlen <- plyr::ddply(tooth.growth, "dose", summarise, len.mean=mean(len))
fill <- c("#b2d183","#00a3dc","#01526d")

s <-  ggplot(data = tooth.growth , aes(len) ) +
      geom_histogram(binwidth = 2, aes(fill = fct_rev(supp)), alpha = .75) +
      facet_grid(dose ~ .) +
      ggtitle("Tooth Growth",
              subtitle = "Length(mm) Grouped by Dose Size (mg)") +
      theme_bw() + scale_fill_manual(values = fill) +
      geom_vline(data = meanlen, aes(xintercept=len.mean),
               color = fill[3], linetype="solid", size=.5) +
      theme(panel.grid.minor = element_blank(),
            panel.grid.major = element_blank(),
            panel.background = element_rect(color = 'white'),
            legend.position = 'bottom', legend.direction = 'horizontal') +
        scale_x_continuous(name = "Length (mm)", breaks = seq(4,34,2), labels = waiver())
```