# Data Visualization With R

Zack Treisman

Spring 2021

# Philosophy

Good visualizations are useful at every stage of the data analysis process from exploration to publication.

R has multiple graphics systems. We will use two:

- Base R graphics are often intuitive, but limited.
- `ggplot2` is robust and widely used. It takes some acclimatization.

Many older resources use `lattice` graphics, which started dropping in popularity as `ggplot2` took over.

Be wary of inference based on purely exploratory data analysis. If you look at your data until you find a pattern, and then test for that pattern, the significance levels of that test are inflated.

# Exploration

Data are in R, now what?

- Check numerical summaries.

```
summary(ReedfrogPred) # data in emdbook
```

```
##     density          pred         size         surv           propsurv
##  Min.   :10.00   no  :24   small:24   Min.   : 4.00   Min.   :0.1143
##  1st Qu.:10.00   pred:24   big  :24   1st Qu.: 9.00   1st Qu.:0.4964
##  Median :25.00                        Median :12.50   Median :0.8857
##  Mean   :23.33                        Mean   :16.31   Mean   :0.7216
##  3rd Qu.:35.00                        3rd Qu.:23.00   3rd Qu.:0.9200
##  Max.   :35.00                        Max.   :35.00   Max.   :1.0000
```

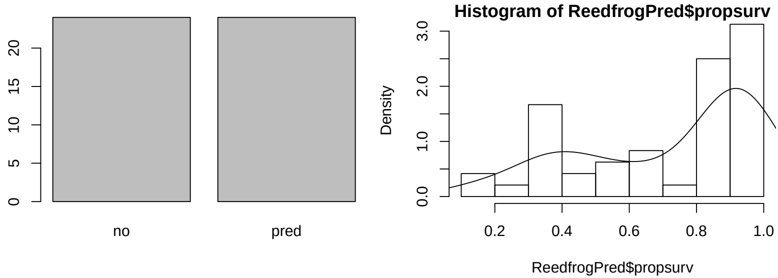
- Make some graphics!
  - Are there patterns that you expected to see?
  - Or didn't expect to see?
  - Are there problems with the data?

# Standard routines - one variable

Graphing the distribution of a single variable means representing how often it takes each possible value.

- Barplots for categorical variables. Same information as a table.
- Histograms for numeric variables. Can add a density estimate.

```
par(mfrow=c(1,2)) # show multiple base R plots at once
barplot(table(ReedfrogPred$pred))
hist(ReedfrogPred$propsurv,freq=F); lines(density(ReedfrogPred$propsurv))
```

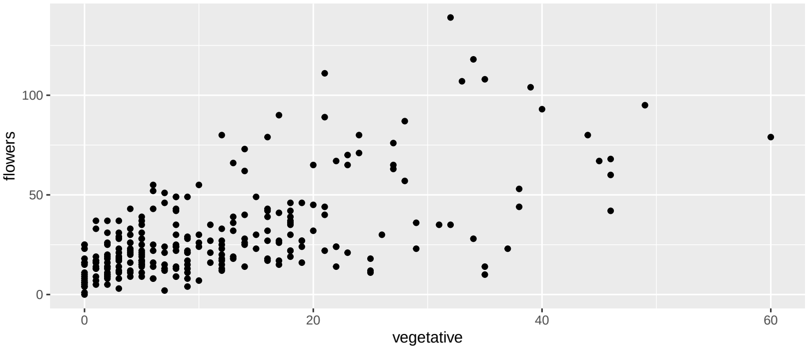


# Standard routines - two numeric variables

Scatterplots show codistribution.

- Put the response variable on the $y$ axis.

```
p <- ggplot(Lily_sum, aes(vegetative, flowers)) + # data are in emdbook
  geom_point()
p
```



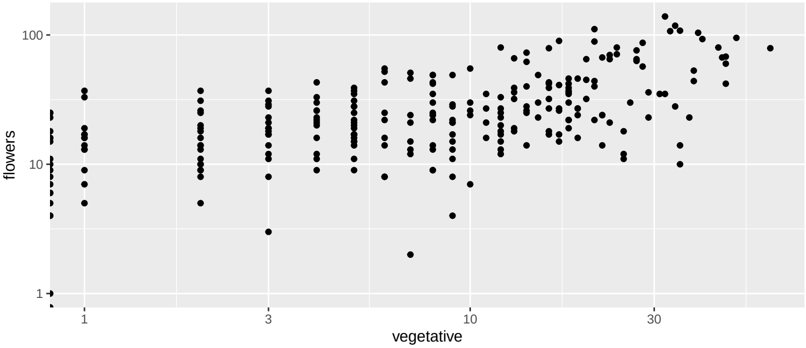
# Log scales

Sometimes the data look better with axes on log scales.

- Counts
- Dimesnional data

Recall log(0) is undefined so 0 values will produce warnings or errors.

```
p + scale_x_log10() +
  scale_y_log10()
```



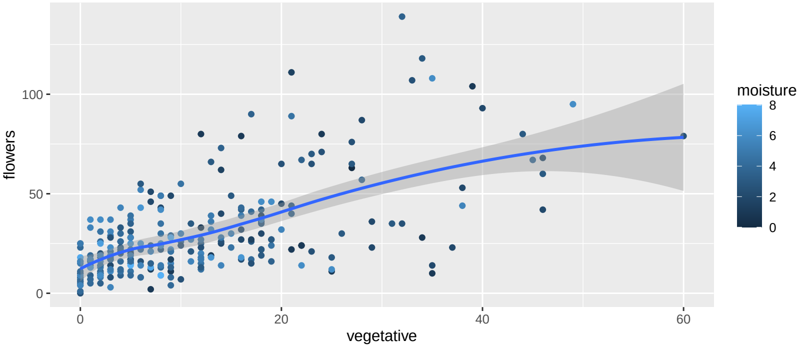
# Additional aesthetics

Map additional variables to color, size or shape (plotting symbol).

- Shape can only be a categorical variable.
- Size can only be a numerical variable.

Can also superimpose trendlines or other model graphs.

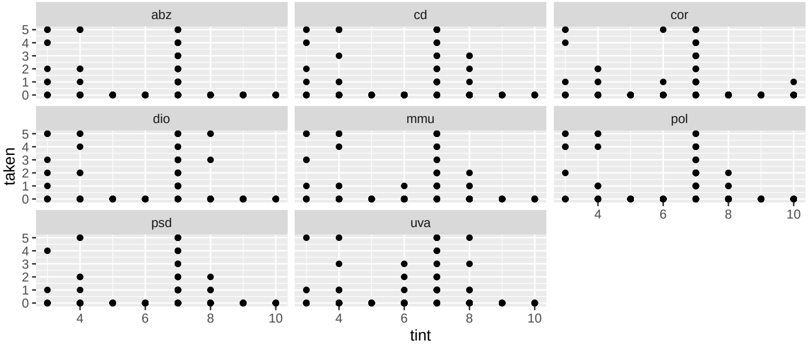
```
ggplot(Lily_sum, aes(vegetative, flowers, color = moisture)) +
  geom_point() +
  geom_smooth()
```



# Faceted plots

Categorical variables can also be represented by making multiple plots. Add facets to a ggplot and specify the variable or variables with facet_wrap(~varA) or facet_grid(varB~varA).

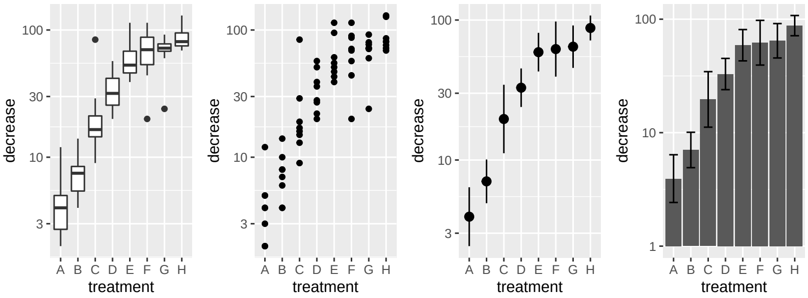
```
ggplot(SeedPred, aes(tint, taken))+ # data in emdbook
  geom_point()+
  facet_wrap(~species)
```



# Standard routines - numeric and categorical

If the response is numeric and all predictors are categorical, you have some options.

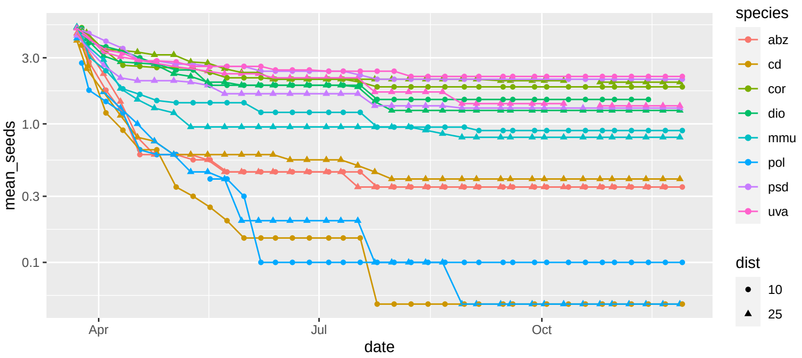
```
g0 <- ggplot(OrchardSprays,aes(x=treatment,y=decrease))+ # data in MASS
  scale_y_log10()
g_boxplot <- g0 + geom_boxplot()
g_point <- g0 +geom_point()
g_errbar <- g0 + stat_summary(fun.data=mean_cl_normal,geom="pointrange")
g_dyn <- g0 +  stat_summary(fun=mean,geom="bar")+
  stat_summary(fun.data=mean_cl_normal,geom="errorbar",width=0.5)
grid.arrange(g_boxplot,g_point,g_errbar,g_dyn, nrow=1)
```



# Dealing with non-standard tasks

Sometimes you need to reshape or summarize your data to plot what you want. To produce Figure 2.1 from Bolker (2008):

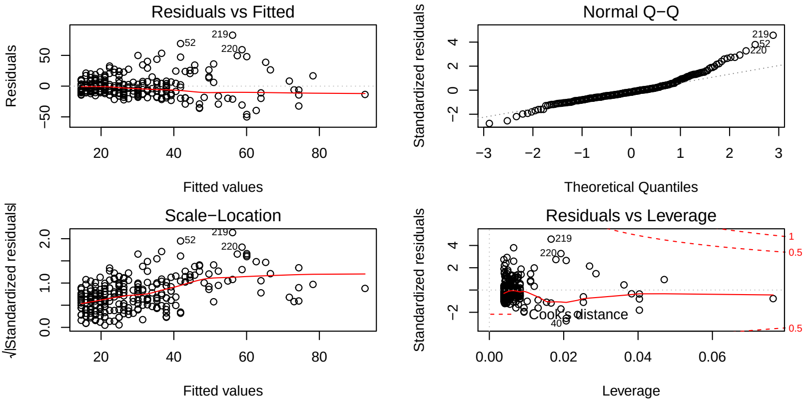
```
SeedPred$date <- ymd(SeedPred$date) # data in emdbook
daily_avgs <- SeedPred %>%
  group_by(date, species, dist) %>%
  summarise(mean_seeds = mean(seeds))
ggplot(daily_avgs, aes(date, mean_seeds, color=species, shape=dist)) +
  geom_point() + geom_line() + scale_y_log10()
```



# Diagnostics

Assessing the validity of a model is often done graphically.

```
lm1 <- lm(flowers~vegetative, data = Lily_sum)
par(mfrow=c(2, 2), mar = c(4, 4, 2, 2)) # see all 4 plots at once
plot(lm1)
```

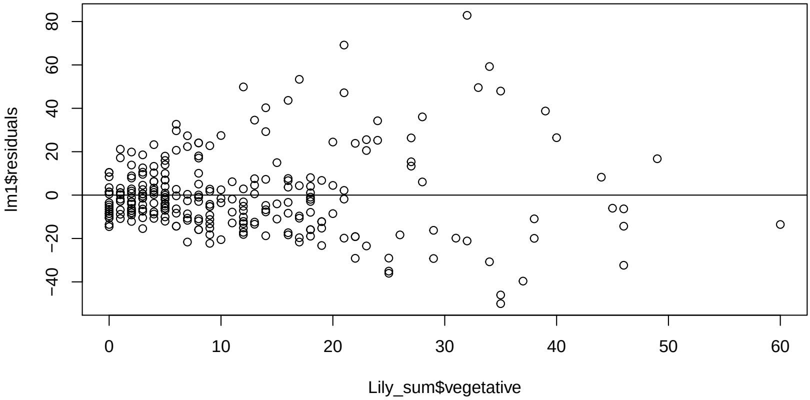



```
par(mfrow=c(1, 1), mar = c(4, 4, 0.75, 0.5)) # restore graphics parameters
```

# Residuals v. predictors

The `plot` method for `lm` doesn't show residuals against predictors.

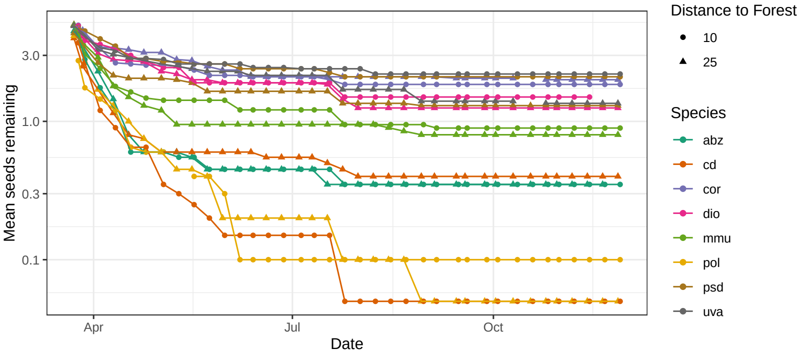
```
plot(lm1$residuals~Lily_sum$vegetative)
abline(h=0)
```



# Exposition

Fine tune and save your graphics.

```
emd2.1<-ggplot(daily_avgs,aes(date,mean_seeds,color=species,shape=dist))+
  geom_point() + geom_line() + scale_y_log10() +
  labs(y="Mean seeds remaining", x = "Date",
       color = "Species", shape = "Distance to Forest") +
  scale_color_brewer(palette = "Dark2") +
  theme_bw()
emd2.1
```




```
ggsave("figures/BolkerFig2.1.tiff", plot=emd2.1,
       width = 10, height = 6, units = "cm", dpi = 800)
```

# References

Bolker, Benjamin M. 2008. *Ecological Models and Data in R*. Princeton University Press.