Choosing explanatory variables

Design choices in statistical models

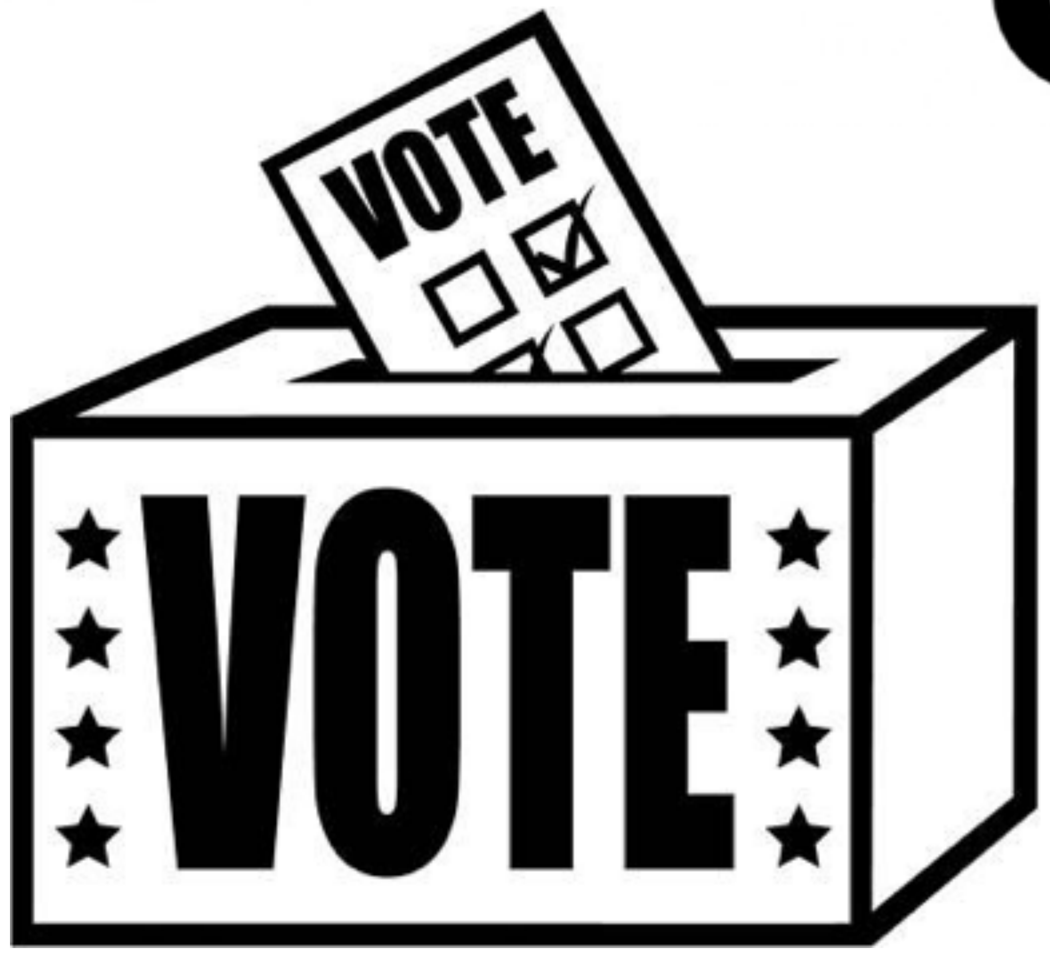
- The data to use for training
- The response variable
- The explanatory variables
- The model architecture: `lm()`, `rpart()`, and others

```
> model_1 <- lm(wage ~ educ + exper, data = CPS85)

> model_2 <- rpart(wage ~ educ + exper, data = CPS85)
```

Response and explanatory variables are specified in the formula

Applying statistical models



Applying statistical models

- Make predictions about an outcome
- Run experiments to study relationships between variables
- Explore data to identify relationships among variables

Basic choices in model architecture

- Categorical response variable (e.g. yes or no, infected or not)

  → Use `rpart()`

- Numerical response variable (e.g. unemployment rate)

  → Use `lm()` or `rpart()`

Gradual, proportional

Dichotomous, discontinuous

Comparing prediction results for variable selection

```
# Specify two models
> base_model <- lm(wage ~ sector + exper, data = CPS85)
> augmented_model <- lm(wage ~ sector + exper + age, data = CPS85)
```

- Train both models and compare them
- If `augmented_model` predicts better, include `age`

Let’s practice!

Cross validation

Training and testing data

name	sex	height
Josi	M	172
Nicole	F	163
Lore	F	170
Anna	F	166
Tom	M	179
Jen	F	151
Leo	M	186
Wes	M	183

Training and testing data

name	sex	height
Nicole	F	163
Anna	F	166
Tom	M	179
Wes	M	183

name	sex	height
Josi	M	172
Lore	F	170
Jen	F	151
Leo	M	186

Training and testing data

name	sex	height
Nicole	F	163
Anna	F	166
Tom	M	179
Wes	M	183

Training

name	sex	height
Josi	M	172
Lore	F	170
Jen	F	151
Leo	M	186

Testing

Using training and testing data

```
# Train base and extended models
> mod_1 <- lm(wage ~ sector + exper, data = Training_data)
> mod_2 <- lm(wage ~ sector + exper + age, data = Training_data)

# Calculate model outputs
> preds_1 <- predict(mod_1, newdata = Testing_data)
> preds_2 <- predict(mod_2, newdata = Testing_data)
```

Comparing model outputs to actual values

```
# Train base and extended models
> mod_1 <- lm(wage ~ sector + exper, data = Training_data)
> mod_2 <- lm(wage ~ sector + exper + age, data = Training_data)

# Calculate model outputs
> preds_1 <- predict(mod_1, newdata = Testing_data)
> preds_2 <- predict(mod_2, newdata = Testing_data)

# Compare model output to actual data
> errors_1 <- Testing_data$wage - preds_1
> errors_2 <- Testing_data$wage - preds_2
```

Mean square error (MSE)

```
# Prediction errors for mod_1
> head(errors_1)
        2         3         4         5         7         8 
-1.347412 -2.343323  1.969980  4.374695  3.554991  8.064577 

# Squared prediction errors for mod_1
> head(errors_1^2)
        2         3         4         5         7         8 
 1.815519  5.491162  3.880823 19.137959 12.637958 65.037399 

# MSE for mod_1
> mean(errors_1^2)
[1] 21.39825

# MSE for mod_2
> mean(errors_2^2)
[1] 18.91559
```

Let's practice!