Fundamentals I: Economy

Gov 1347: Election Analytics

Kiara Hernandez

September 15, 2022

Harvard University

Today's goal

Can we predict midterm election outcomes using *only* the state of the economy?

1. **Describing how the economy relates to elections**
   - Bivariate correlation between X and Y (r_{XY})
2. **How to make a prediction by fitting a model to your data:**
   - Linear regression of Y on X
3. **How to evaluate your model:**
   - In-sample model fit
   - Out-of-sample model testing
   - Out-of-sample extrapolation
4. **How to improve your model:**
   - Measure for a single independent variable
   - Multiple independent variables

Before we start, quick recap of code for national map

left_join by district and state

```
# example from Blog 01
R_2014 <- h %>%
  filter(raceYear == 2014) %>% #State == "New Jersey") %>%
  select(raceYear, State, district_num, RepVotesMajorPercent, De
  # summarize party vote share by district
  group_by(district_num, State) %>%
  summarise(Rep_votes_pct = RepVotesMajorPercent) %>%
  # rename district and state variable to match shapefile
  rename(DISTRICT = district_num, STATENAME = State)

# merge
cd114$DISTRICT <- as.numeric(cd114$DISTRICT)
cd114 <- cd114 %>% left_join(R_2014, by=c("DISTRICT", "STATENAME
```

Use package ‘rmapshaper’ to plot - rmapshaper::ms_simplify()

```r
# plot with simplify
districts_simp <- rmapshaper::ms_simplify(cd114, keep = 0.01)
```

Add a layer to your ggplot to set geographic parameters: coord_sf()

```
ggplot() +
  geom_sf(data=districts_simp,aes(fill=Rep_votes_pct),
          inherit.aes=FALSE,alpha=0.9) +
  scale_fill_gradient(low = "white", high = "black", limits=c(0,
  coord_sf(xlim = c(-172.27, -66.57), ylim = c(18.55, 71.23), ex
  theme_void() +
  theme(axis.title.x=element_blank(),
        axis.text.x=element_blank(),
        axis.ticks.x=element_blank(),
        axis.title.y=element_blank(),
        axis.text.y=element_blank(),
        axis.ticks.y=element_blank())
```

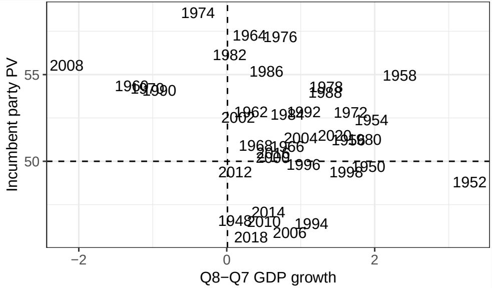
Describing how the economy relates to elections

Bivariate correlation of economy and PV

A **scatterplot** visualizes bivariate correlation between some X (independent variable or IV) and Y (dependent variable or DV).

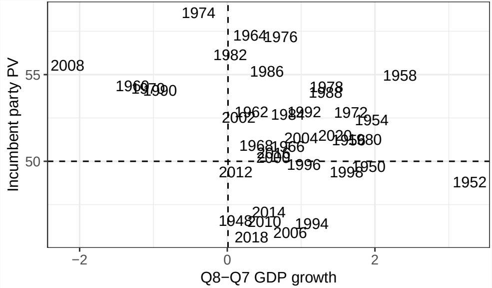
Bivariate correlation of economy and PV

A **scatterplot** visualizes bivariate correlation between some X (independent variable or IV) and Y (dependent variable or DV).



Bivariate correlation of economy and PV

A **scatterplot** visualizes bivariate correlation between some X (independent variable or IV) and Y (dependent variable or DV).

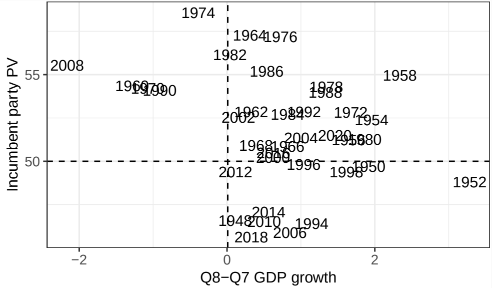


Bivariate correlation is formally measured from -1 to 1 as:

$$r_{XY} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}.$$

Bivariate correlation of economy and PV

A **scatterplot** visualizes bivariate correlation between some X (independent variable or IV) and Y (dependent variable or DV).



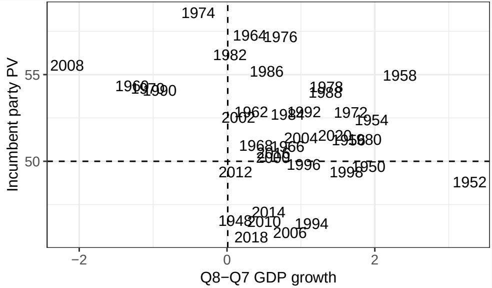
Bivariate correlation is formally measured from -1 to 1 as:

$$r_{XY} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}.$$

```
cor(dat2$GDP_growth_pct, dat2$H_incumbent_party_majorvote_pct)
```

Bivariate correlation of economy and PV

A **scatterplot** visualizes bivariate correlation between some X (independent variable or IV) and Y (dependent variable or DV).



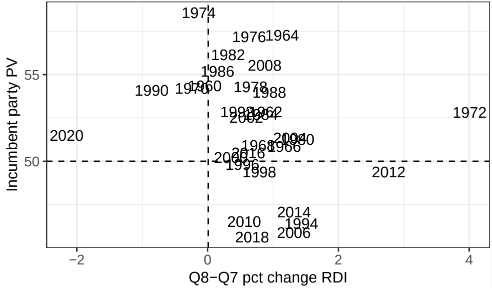
Bivariate correlation is formally measured from -1 to 1 as:

$$r_{XY} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}.$$

```
cor(dat2$GDP_growth_pct, dat2$H_incumbent_party_majorvote_pct)
```

```
## [1] -0.2840337
```

A **scatterplot** visualizes bivariate correlation between some X (independent variable or IV) and Y (dependent variable or DV).



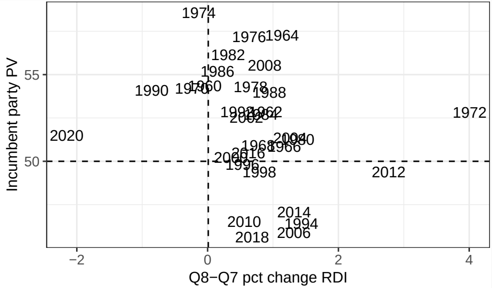
Bivariate correlation is formally measured from -1 to 1 as:

$$r_{XY} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}.$$

```
cor(data2$DSPIC_change_pct, data2$H_incumbent_party_majorvote_pc
```

Bivariate correlation of economy and PV

A **scatterplot** visualizes bivariate correlation between some X (independent variable or IV) and Y (dependent variable or DV).



Bivariate correlation is formally measured from -1 to 1 as:

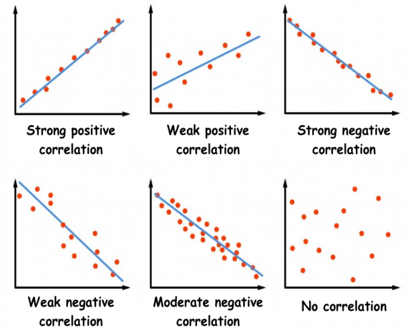
$$r_{XY} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}.$$

```
cor(data2$DSPIC_change_pct, data2$H_incumbent_party_majorvote_pc
```

```
## [1] -0.2361528
```

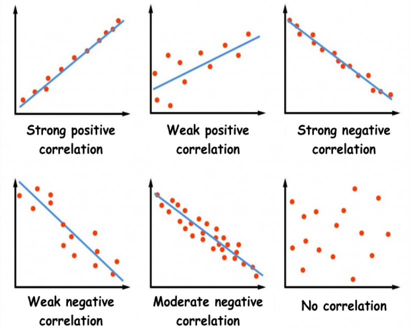
Summary of bivariate correlation

Strong bivariate correlation means X probably predicts Y well.



Summary of bivariate correlation

Strong bivariate correlation means X probably predicts Y well.



But, correlation can't tell us what the underlying model is to generate Y from X.

How to make a prediction by fitting a model to your data

How to make a prediction

Given some variable (DV) Y that you wish to predict:

How to make a prediction

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form of a function $Y = f(X)$ for some proposed X, the IV.

How to make a prediction

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form of a function $Y = f(X)$ for some proposed X, the IV.

STEP 2. Estimate the parameters of that function from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

How to make a prediction

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form of a function $Y = f(X)$ for some proposed X, the IV.

STEP 2. Estimate the parameters of that function from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Determine whether parameters are "good".

How to make a prediction

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form of a function $Y = f(X)$ for some proposed X, the IV.

STEP 2. Estimate the parameters of that function from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Determine whether parameters are "good".

STEP 4. Obtain a new observation of the IV, X_{new}.

How to make a prediction

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form of a function $Y = f(X)$ for some proposed X, the IV.

STEP 2. Estimate the parameters of that function from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Determine whether parameters are "good".

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = f(X_{new})$.

How to make a prediction

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form of a function $Y = f(X)$ for some proposed X, the IV.

STEP 2. Estimate the parameters of that function from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Determine whether parameters are "good".

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = f(X_{new})$.

STEP 6. Calculate uncertainty about estimate $\widehat{Y}_{new}$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form of a function $Y = f(X)$ for some proposed X, the IV.

STEP 2. Estimate the parameters of that function from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Determine whether parameters are "good".

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = f(X_{new})$.

STEP 6. Calculate uncertainty about estimate $\widehat{Y}_{new}$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form $Y = \underbrace{A}_{\textit{intercept}} + \underbrace{B}_{\textit{slope}} X$.

STEP 2. Estimate the parameters of that function from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Determine whether parameters are "good".

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = f(X_{new})$.

STEP 6. Calculate uncertainty about estimate $\widehat{Y}_{new}$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form $Y = \underbrace{A}_{intercept} + \underbrace{B}_{slope} X$.

STEP 2. Calculate estimates $\widehat{A}$ and $\widehat{B}$, the "best guesses" at the slopes and intercepts, from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Determine whether parameters are "good".

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = f(X_{new})$.

STEP 6. Calculate uncertainty about estimate $\widehat{Y}_{new}$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form $Y = \underbrace{A}_{\textit{intercept}} + \underbrace{B}_{\textit{slope}} X$.

STEP 2. Calculate estimates $\widehat{A}$ and $\widehat{B}$, the "best guesses" at the slopes and intercepts, from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Check in-sample model fit and perform out-of-sample testing.

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = f(X_{new})$.

STEP 6. Calculate uncertainty about estimate $\widehat{Y}_{new}$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form $Y = \underbrace{A}_{intercept} + \underbrace{B}_{slope} X$.

STEP 2. Calculate estimates $\widehat{A}$ and $\widehat{B}$, the "best guesses" at the slopes and intercepts, from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Check in-sample model fit and perform out-of-sample testing.

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = f(X_{new})$.

STEP 6. Calculate uncertainty about estimate $\widehat{Y}_{new}$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form $Y = \underbrace{A}_{intercept} + \underbrace{B}_{slope} X$.

STEP 2. Calculate estimates $\widehat{A}$ and $\widehat{B}$, the "best guesses" at the slopes and intercepts, from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Check in-sample model fit and perform out-of-sample testing.

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = \widehat{A} + \widehat{B} X_{new}$.

STEP 6. Calculate uncertainty about estimate $\widehat{Y}_{new}$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form $Y = \underbrace{A}_{intercept} + \underbrace{B}_{slope} X$.

STEP 2. Calculate estimates $\widehat{A}$ and $\widehat{B}$, the "best guesses" at the slopes and intercepts, from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Check in-sample model fit and perform out-of-sample testing.

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = \widehat{A} + \widehat{B}X_{new}$.

STEP 6. Calculate a **prediction interval** for $\widehat{Y}_{new}$ as $\widehat{Y}_{new} \pm 1.96^{*} \times \text{se}^{**}(\widehat{Y}_{new})$.

How to make a prediction using linear regression

Given some variable (DV) Y that you wish to predict:

STEP 1. Specify a model in the form $Y = \underbrace{A}_{\textit{intercept}} + \underbrace{B}_{\textit{slope}} X$.

STEP 2. Calculate estimates $\widehat{A}$ and $\widehat{B}$, the "best guesses" at the slopes and intercepts, from a sample $(x_1, y_1), \ldots, (x_n, y_n)$ observed of the variables.

STEP 3. Check in-sample model fit and perform out-of-sample testing.

STEP 4. Obtain a new observation of the IV, X_{new}.

STEP 5. Predict its DV Y_{new} value as $\widehat{Y}_{new} = \widehat{A} + \widehat{B} X_{new}$.

STEP 6. Calculate a **prediction interval** for $\widehat{Y}_{new}$ as $\widehat{Y}_{new} \pm 1.96^{*} \times \text{se}^{**}(\widehat{Y}_{new})$.

* If we truly believe our model and we additionally assume errors between all Y and predicted $\hat{Y}$ are normally distributed, scaling the standard deviation by 1.96 ensures that our predictive interval will contain the true Y_{new} 95% of the time.

** Standard Error.

Economy and PV: Fitting a model (STEP 1 & 2)

For now let's use just a single IV for two models: (1) Q8-Q7 GDP growth and (2) Q8-Q7 percent change RDI.

Economy and PV: Fitting a model (STEP 1 & 2)

For now let's use just a single IV for two models: (1) Q8-Q7 GDP growth and (2) Q8-Q7 percent change RDI. We can fit a linear regression model using `lm()`:

We can fit this linear regression model using `lm()`:

```
lm_econ <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
              data = dat2)
# # lm_rdi <- lm(H_incumbent_party_majorvote_pct ~ DSPIC_change_
#               data = data2)
```

Economy and PV: Fitting a model (STEP 1 & 2)

For now let's use just a single IV for two models: (1) Q8-Q7 GDP growth and (2) Q8-Q7 percent change RDI. We can fit a linear regression model using `lm()`:

We can fit this linear regression model using `lm()`:

```
lm_econ <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
              data = dat2)
# # lm_rdi <- lm(H_incumbent_party_majorvote_pct ~ DSPIC_change_
#               data = data2)
```

```
summary(lm_econ)
# summary(lm_rdi)
```

Economy and PV: Fitting a model (STEP 1 & 2)

For now let's use just a single IV for two models: (1) Q8-Q7 GDP growth and (2) Q8-Q7 percent change RDI. We can fit a linear regression model using `lm()`:

We can fit this linear regression model using `lm()`:

```
lm_econ <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
              data = dat2)
# # lm_rdi <- lm(H_incumbent_party_majorvote_pct ~ DSPIC_change_
#               data = data2)
```

```
summary(lm_econ)
# summary(lm_rdi)
```

```
summary(lm_econ)
```

```
##
## Call:
## lm(formula = H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
##     data = dat2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

Economy and PV: Fitting a model (STEP 1 & 2)

For now let's use just a single IV for two models: (1) Q8-Q7 GDP growth and (2) Q8-Q7 percent change RDI. We can fit a linear regression model using `lm()`:

We can fit this linear regression model using `lm()`:

```
lm_econ <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
               data = dat2)
# # lm_rdi <- lm(H_incumbent_party_majorvote_pct ~ DSPIC_change_
#                 data = data2)
```

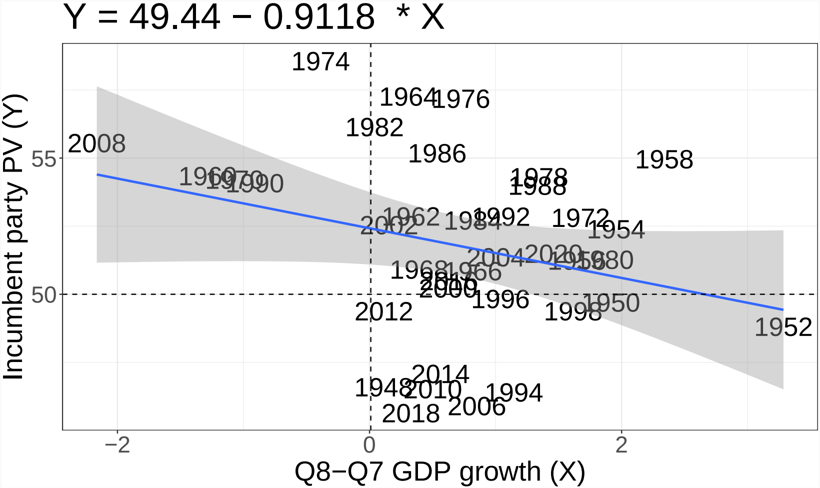
```
summary(lm_econ)
# summary(lm_rdi)
```

```
summary(lm_econ)
```

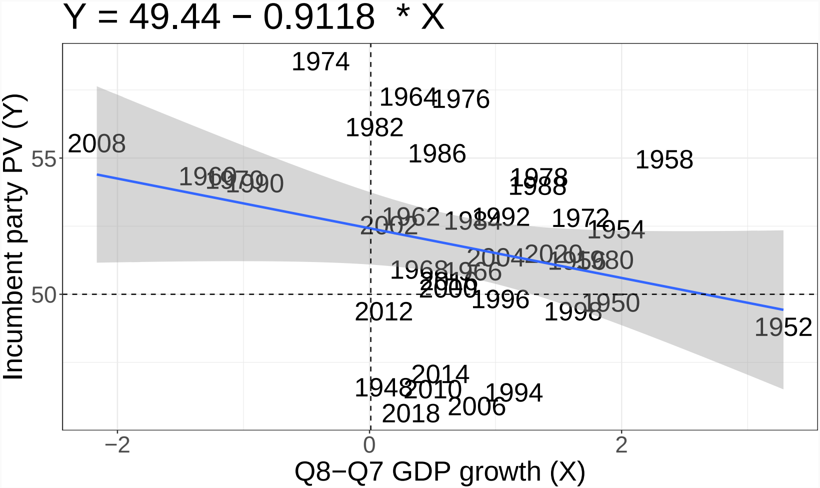
```
##
## Call:
## lm(formula = H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
##     data = dat2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
```

Economy and PV: Fitting a model (STEP 1 & 2)

Economy and PV: Fitting a model (STEP 1 & 2)



Economy and PV: Fitting a model (STEP 1 & 2)



How to evaluate your model (STEP 3)

Tools

Key: We want to evaluate the predictive power of our model.

Tools

Key: We want to evaluate the predictive power of our model.

- In-sample fit
    1. R^2
    2. In-sample error

Tools

Key: We want to evaluate the predictive power of our model.

- In-sample fit
  1. R^2
  2. In-sample error

- Out-of-sample testing
  1. Leave-one-out validation
  2. Cross-validation
  3. **Real** out-of-sample prediction (and see what happens...)

Model Fit: R^2

$$R^2 = 1 - \frac{\sum_{i=1}^{n}(y_i - \widehat{y_i})^2}{\sum_{i=1}^{n}(y_i - \bar{y})^2}$$

Model Fit: R^2

$$R^2 = 1 - \frac{\overbrace{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}^{\text{variance unexplained by model}}}{\underbrace{\sum_{i=1}^{n}(y_i - \bar{y})^2}_{\text{variance in data}}}$$

says how much variation of in Y values in the sample is captured by the fitted model's predicted values $\widehat{Y}$.

Model Fit: R^2

$$R^2 = 1 - \frac{\overbrace{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}^{\text{variance unexplained by model}}}{\underbrace{\sum_{i=1}^{n}(y_i - \bar{y})^2}_{\text{variance in data}}}$$

says how much variation of in Y values in the sample is captured by the fitted model's predicted values $\widehat{Y}$.

```
summary(lm_econ)$r.squared
```

```
## [1] 0.08067517
```

```
# summary(lm_rdi)$r.squared
```

Model Fit: R^2

$$R^2 = 1 - \frac{\overbrace{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}^{\text{variance unexplained by model}}}{\underbrace{\sum_{i=1}^{n}(y_i - \bar{y})^2}_{\text{variance in data}}}$$

says how much variation of in Y values in the sample is captured by the fitted model's predicted values $\widehat{Y}$.

```r
summary(lm_econ)$r.squared
```

```
## [1] 0.08067517
```

```r
# summary(lm_rdi)$r.squared
```

For a univariate linear regression, this is the same as the:

Model Fit: R^2

$$R^2 = 1 - \frac{\overbrace{\sum_{i=1}^{n}(y_i - \widehat{y}_i)^2}^{\text{variance unexplained by model}}}{\underbrace{\sum_{i=1}^{n}(y_i - \bar{y})^2}_{\text{variance in data}}}$$

says how much variation of in Y values in the sample is captured by the fitted model's predicted values $\widehat{Y}$.

```
summary(lm_econ)$r.squared
```

```
## [1] 0.08067517
```

```
# summary(lm_rdi)$r.squared
```

For a univariate linear regression, this is the same as the: square of bivariate correlation between X and Y (r^2_{XY}).

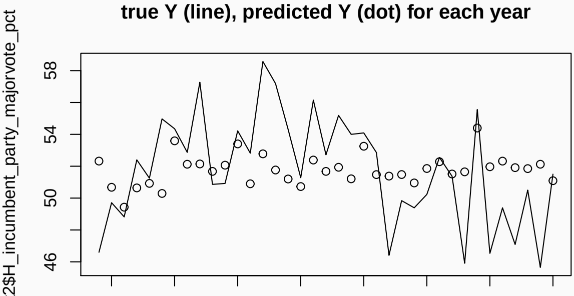
Model Fit: in-sample error and MSE

We can plot the in-sample error via **residuals**, which capture the difference between each observed value (y_i) and predicted value ($\widehat{y}_i = \hat{A} + \widehat{B}x_i$):

Model Fit: in-sample error and MSE

We can plot the in-sample error via **residuals**, which capture the difference between each observed value (y_i) and predicted value ($\widehat{y}_i = \hat{A} + \widehat{B}x_i$):

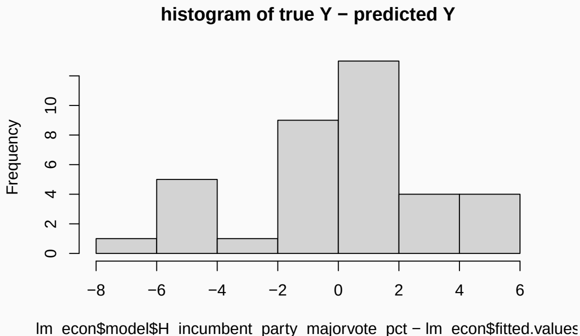
```
# GDP
plot(dat2$year, dat2$H_incumbent_party_majorvote_pct,
     type="l",
     main="true Y (line), predicted Y (dot) for each year")
points(dat2$year, predict(lm_econ, dat2))
```



Model Fit: in-sample error and MSE

We can plot the in-sample error via **residuals**, which capture the difference between each observed value (y_i) and predicted value ($\widehat{y}_i = \hat{A} + \widehat{B}x_i$):

```
# GDP
hist(lm_econ$model$H_incumbent_party_majorvote_pct -
        lm_econ$fitted.values,
     main="histogram of true Y - predicted Y")
```



Model Fit: in-sample error and MSE

We can summarise the error a single number, such as the **mean-squared error (MSE)**:

```
# GDP
mse_g <- mean((lm_econ$model$H_incumbent_party_majorvote_pct -
                  lm_econ$fitted.values)^2)
sqrt(mse_g)
```

```
## [1] 3.180479
```

```
# # RDI
# mse_r <- mean((lm_rdi$model$H_incumbent_party_majorvote_pct -
#                   lm_rdi$fitted.values)^2)
# sqrt(mse_r)
```

This is hard to interpret on its own, more useful in comparison with other models.

Model Testing

Model Testing

Checking in-sample model predictions is a good baseline evaluation, but it feels a bit like "cheating"

Model Testing

Checking in-sample model predictions is a good baseline evaluation, but it feels a bit like "cheating" $\rightsquigarrow$ can we take away the model's "answer key"?

Model Testing

Checking in-sample model predictions is a good baseline evaluation, but it feels a bit like "cheating" $\rightsquigarrow$ can we take away the model's "answer key"?

We can simulate **out-of-sample prediction** (also called out-of-sample testing) by withholding some observation, e.g. X_{2018}, before fitting:

Model Testing

Checking in-sample model predictions is a good baseline evaluation, but it feels a bit like "cheating" $\rightsquigarrow$ can we take away the model's "answer key"?

We can simulate **out-of-sample prediction** (also called out-of-sample testing) by withholding some observation, e.g. X_{2018}, before fitting:

```
# GDP
outsamp_mod1 <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_
                    dat2[dat2$year != 2018,])
outsamp_pred <- predict(outsamp_mod1,
                        dat2[dat2$year == 2018,])
outsamp_true <- dat2$H_incumbent_party_majorvote_pct[dat2$year =
```

Model Testing

Checking in-sample model predictions is a good baseline evaluation, but it feels a bit like "cheating" ⇝ can we take away the model's "answer key"?

We can simulate **out-of-sample prediction** (also called out-of-sample testing) by withholding some observation, e.g. X_{2018}, before fitting:

```
# GDP
outsamp_mod1 <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_
                   dat2[dat2$year != 2018,])
outsamp_pred <- predict(outsamp_mod1,
                       dat2[dat2$year == 2018,])
outsamp_true <- dat2$H_incumbent_party_majorvote_pct[dat2$year =
```

and see how well the model predicts the true Y_{2018} for the held-out observation X_{2018}:

```
outsamp_pred - outsamp_true
```

```
##         1
```

Model Testing

Checking in-sample model predictions is a good baseline evaluation, but it feels a bit like "cheating" ⇝ can we take away the model's "answer key"?

We can simulate **out-of-sample prediction** (also called out-of-sample testing) by withholding some observation, e.g. X_{2018}, before fitting:

```
# GDP
outsamp_mod1 <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_
                   dat2[dat2$year != 2018,])
outsamp_pred <- predict(outsamp_mod1,
                       dat2[dat2$year == 2018,])
outsamp_true <- dat2$H_incumbent_party_majorvote_pct[dat2$year =
```

and see how well the model predicts the true Y_{2018} for the held-out observation X_{2018}:

```
outsamp_pred - outsamp_true
```



```
##          1
```

Model Testing

Cross-validation: withold a *random subset* of the sample, fit model on rest of sample, and evaluate predictive performance on the held-out observations.

Model Testing

Cross-validation: withold a *random subset* of the sample, fit model on rest of sample, and evaluate predictive performance on the held-out observations.

```
# GDP
years_outsamp <- sample(dat2$year, 8)
mod <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
          dat2[!(dat2$year %in% years_outsamp),])
outsamp_pred <- predict(mod,
                newdata = dat2[dat2$year %in%
                                     years_outsamp,])
```

Model Testing

Cross-validation: withold a *random subset* of the sample, fit model on rest of sample, and evaluate predictive performance on the held-out observations.

```
# GDP
years_outsamp <- sample(dat2$year, 8)
mod <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
          dat2[!(dat2$year %in% years_outsamp),])
outsamp_pred <- predict(mod,
                newdata = dat2[dat2$year %in%
                                   years_outsamp,])
```

```
mean(outsamp_pred - dat2$H_incumbent_party_majorvote_pct[dat2$ye
                                                            %in% ye
```

```
## [1] 1.472607
```

Model Testing

Cross-validation: withold a *random subset* of the sample, fit model on rest of sample, and evaluate predictive performance on the held-out observations.

```
# GDP
years_outsamp <- sample(dat2$year, 8)
mod <- lm(H_incumbent_party_majorvote_pct ~ GDP_growth_pct,
          dat2[!(dat2$year %in% years_outsamp),])
outsamp_pred <- predict(mod,
                        newdata = dat2[dat2$year %in%
                                       years_outsamp,])
```

```
mean(outsamp_pred - dat2$H_incumbent_party_majorvote_pct[dat2$ye
                                                          %in% ye
```

```
## [1] 1.472607
```

But we don't want to do this just once.

Model Testing

Cross-validation involves repeatedly evaluating performance against many randomly held-out “out-of-sample” datasets:

```
years_outsamp <- sample(dat2$year, 8)
outsamp_mod <- lm(H_incumbent_party_majorvote_pct ~
                    GDP_growth_pct,
                  dat2[!(dat2$year %in% years_outsamp),])
outsamp_pred <- predict(outsamp_mod,
                newdata = dat2[dat2$year %in% years_outsamp,])
outsamp_true <- dat2$H_incumbent_party_majorvote_pct[dat2$year
                                                    %in% years_
mean(outsamp_pred - outsamp_true)
```

Model Testing

Cross-validation involves repeatedly evaluating performance against many randomly held-out "out-of-sample" datasets:

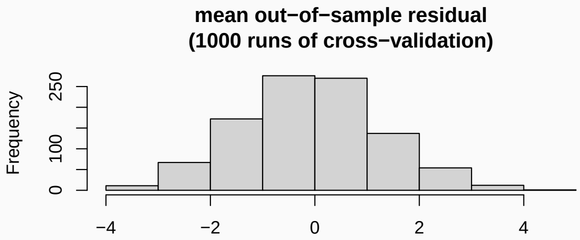
```
outsamp_errors <- sapply(1:1000, function(i){
    years_outsamp <- sample(dat2$year, 8)
  outsamp_mod <- lm(H_incumbent_party_majorvote_pct ~
                     GDP_growth_pct,
                 dat2[!(dat2$year %in% years_outsamp),])
  outsamp_pred <- predict(outsamp_mod,
               newdata = dat2[dat2$year %in% years_outsamp,])
  outsamp_true <- dat2$H_incumbent_party_majorvote_pct[dat2$year
                                                      %in% year
  mean(outsamp_pred - outsamp_true)
})
```

Model Testing

We can then look at a distribution of evaluations, rather than one single evaluation:

Model Testing

We can then look at a distribution of evaluations, rather than one single evaluation:




```
mean(abs(outsamp_errors))
```

```
## [1] 1.067424
```

Economy and PV: Out-of-sample prediction (STEP 4 & 5)

Ok, now let's say we're happy with our model.

Economy and PV: Out-of-sample prediction (STEP 4 & 5)

Ok, now let's say we're happy with our model. Plug in the 2nd quarter GDP growth this year:

```
GDP_new <- economy_df %>%
    subset(year == 2020 & quarter_cycle == 8) %>%
    select(GDP_growth_pct)

predict(lm_econ, GDP_new)
```

Economy and PV: Out-of-sample prediction (STEP 4 & 5)

Ok, now let's say we're happy with our model. Plug in the 2nd quarter GDP growth this year:

```
GDP_new <- economy_df %>%
    subset(year == 2020 & quarter_cycle == 8) %>%
    select(GDP_growth_pct)

predict(lm_econ, GDP_new)
```

```
##       1
## 51.0921
```

Economy and PV: Prediction uncertainty (STEP 6)

```
predict(lm_econ, GDP_new, interval="prediction")
```

Economy and PV: Prediction uncertainty (STEP 6)

```r
predict(lm_econ, GDP_new, interval="prediction")
```

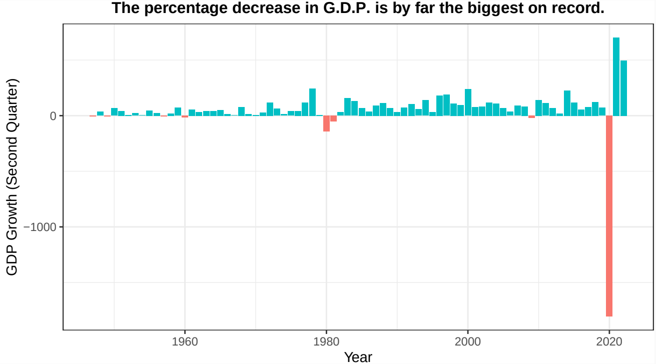
```
##       fit      lwr      upr
## 1 51.0921 44.31922 57.86497
```

What's wrong with a "fundamentals-only" forecast for 2020?

Replicating New York Times:

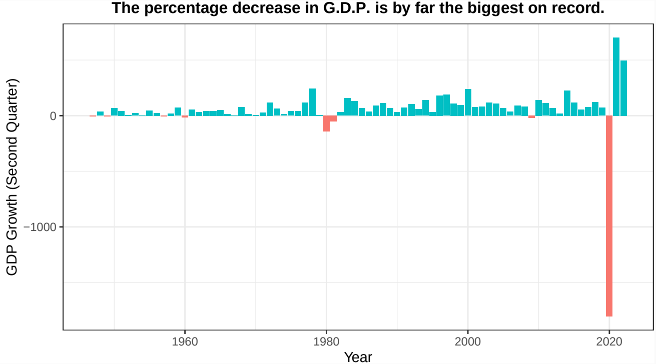
What's wrong with a "fundamentals-only" forecast for 2020?

Replicating New York Times:



What's wrong with a "fundamentals-only" forecast for 2020?

Replicating New York Times:



Extrapolation: Forecasting a DV from an observation of X_{new} *much smaller or bigger* than any $x_1, \ldots, x_n$ in sample used to fit model.

How to improve your model

Most obvious: Choice of measure for IV

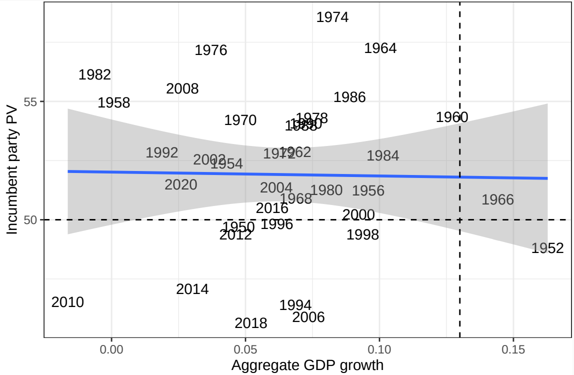
So many options for GDP growth:

- Ex. Q8 of election year vs aggregate GDP growth for 2 years (8 quarters).
- Latter makes sense but implies a stronger behavioral model

Most obvious: Choice of measure for IV

So many options for GDP growth:

- Ex. Q8 of election year vs aggregate GDP growth for 2 years (8 quarters).
- Latter makes sense but implies a stronger behavioral model (full information, rational calculus, retrospective voting).



Most obvious: Choice of measure for IV

Multiple measures of economic performance!

```
* GDP growth
* Real disposable income
* Unemployment
* Inflation
```

Most obvious: Choice of measure for IV

Multiple measures of economic performance!

```
* GDP growth
* Real disposable income
* Unemployment
* Inflation
```

Another option is to include multiple economic IVs $X_1, X_2, \ldots$ in our model, since they capture <u>different dimensions</u> of the economy.

Multiple IVs

What's one potential draw-back of throwing a "kitchen sink" of IVs into a model?

Multiple IVs

What's one potential draw-back of throwing a "kitchen sink" of IVs into a model? IVs may be **multicollinear**, that is highly correlated and therefore, in a sense, redundant

Multiple IVs

What's one potential draw-back of throwing a "kitchen sink" of IVs into a model? IVs may be **multicollinear**, that is highly correlated and therefore, in a sense, redundant ⇝ IVs are no longer independent variables!

```
cor(agg$GDP_growth_pct, agg$GDP_growth_total)
```

```
## [1] 0.149489
```

Multiple IVs

What's one potential draw-back of throwing a "kitchen sink" of IVs into a model? IVs may be **multicollinear**, that is highly correlated and therefore, in a sense, redundant $\rightsquigarrow$ IVs are no longer independent variables!

```
cor(agg$GDP_growth_pct, agg$GDP_growth_total)
```

```
## [1] 0.149489
```

We want models that capture the complexities of the real world, but that are also parsimonious (why? will explore this in the future).

Blog Extensions

1. **Model Evaluation.** Build multiple predictive models using national economic variables as predictors. Compare those models using the tools we learned today. How much is your 2022 prediction sensitive to the change of measure(s)? What does it tell us about the economic model of voting behavior?
2. **Heterogenous Predictive Power of the Economy.** Does the effect of the economy vary when we consider popular vote versus seat share as our outcome (dependent) variable? Does the predictive power of economy change across time? If so, why?
3. **Local Economy.** We can think of a behavioral model where voters base their decisions not on national economy but on their local economy (or both!). Build a predictive model for 2022 using unemployment data at the state level: `unemployment_state_monthly.csv`. You can use popular vote or seat share as your outcome variable. Does this improve predictive power compared to solely focusing on national economy?