Introduction To Regression

R Open Labs Workshop Series

Evan Wyse

2020-03-02

Download slides at

http://bit.ly/duke_lib_regression

Agenda

- What is regression?
- Fitting a model in R
- Interpreting Output
- Model Diagnostics
- Checking Assumptions
- Interactive Exercises + Q&A

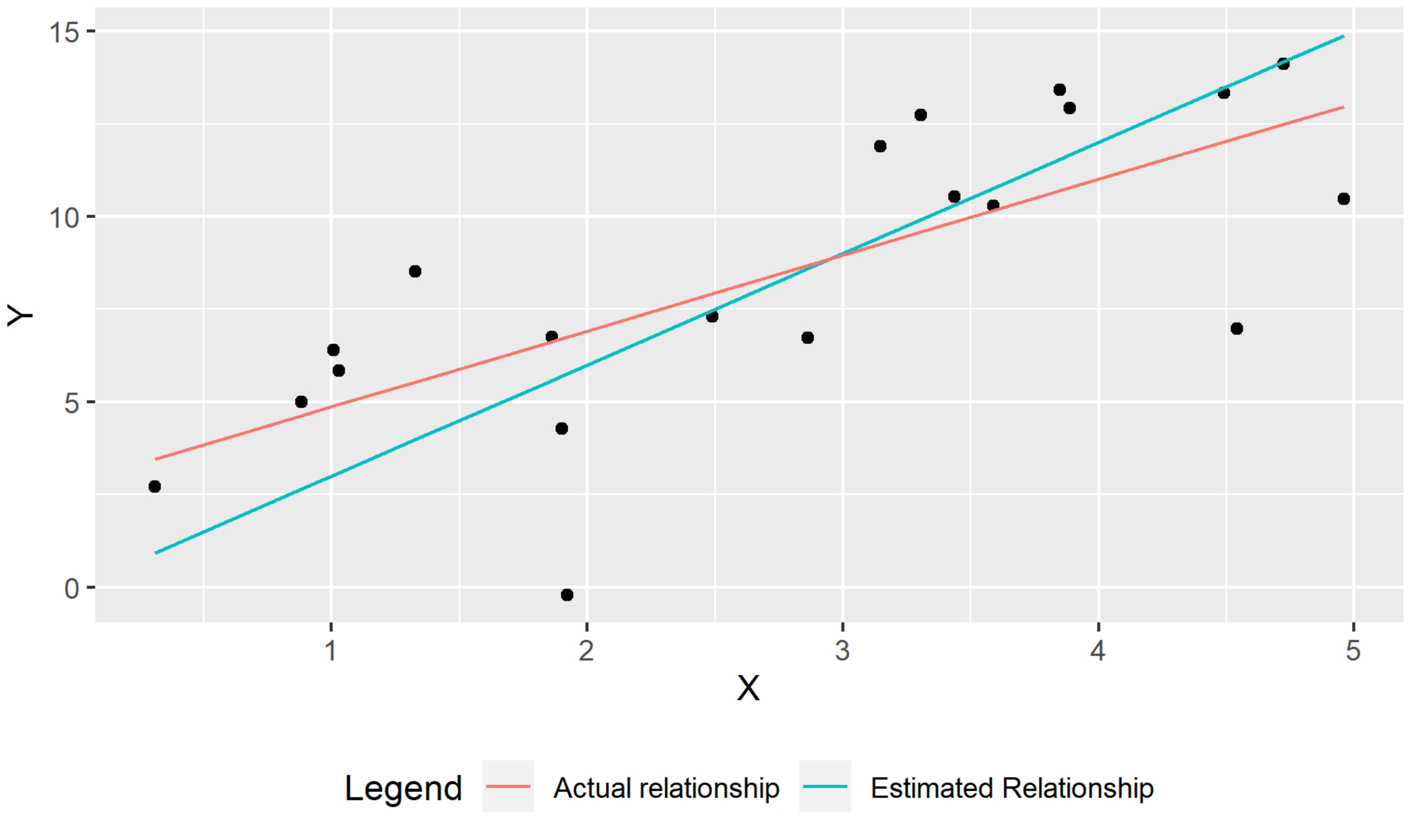
Disclaimer

- Regression is a complicated and deep subject. While this talk is a solid introduction, there are some significant caveats to its use. There is a whole undergraduate course at Duke on regression (STA 210). As such, it's probably not a good idea to publish a paper based on what a statistics grad student taught you in an hour.
- These slides make significant use of the course material from STA 210, taught by Professor Maria Tackett
    - You can access course materials here - they provide significantly more detail than is available here

Simple Linear Regression

- We observe a dataset $\mathbf{Y}$ composed of n observations, $Y_1 \ldots Y_n$, and an explanatory variable $X_1 \ldots X_n$
- Suspect that there is an (imperfect) linear relationship between $\mathbf{Y}$ and $\mathbf{X}$, thus our model is $Y_i = \beta_0 + \beta_1 x_{i1} + \epsilon$
  - ϵ is an error term - we assume that it's drawn from a normal (bell-curve) distribution with an unknown variance σ^2
- We don't know what β_0, β_1, or σ^2 are - but we'd like to estimate them
  - We'll call our estimate for the unknown β and σ^2 as $\hat{\beta}$ and $\hat{\sigma^2}$ respectively

Regression Visualized

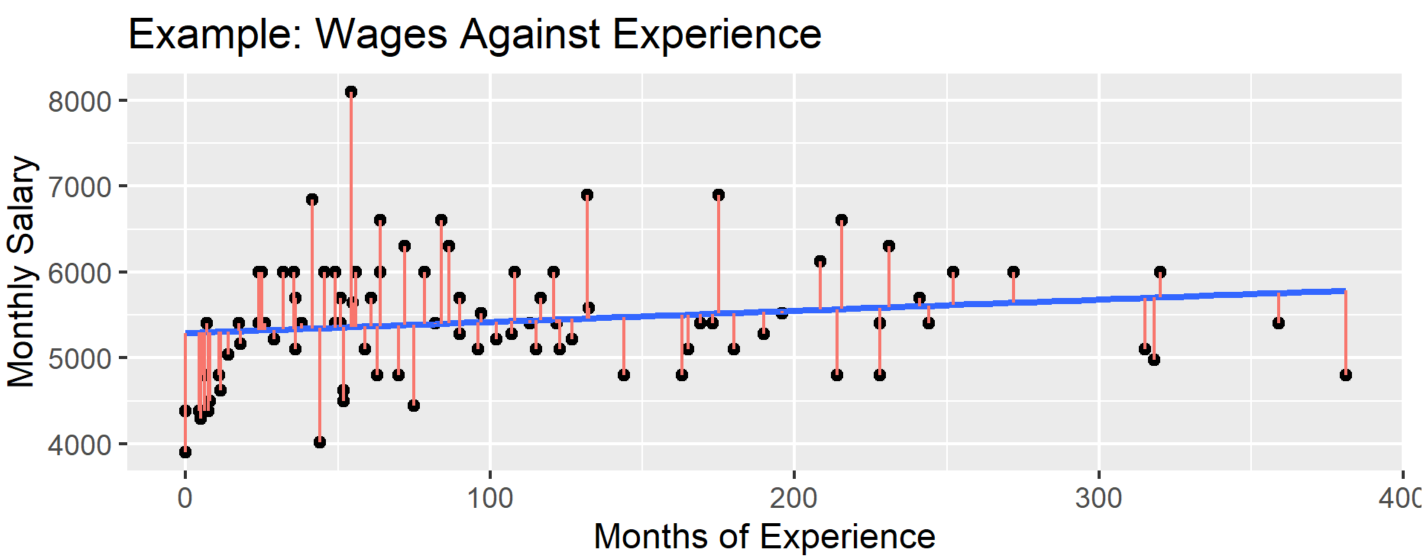


Expanding To Multiple Predictors

- Dataset of n observations of a response variable $\mathbf{Y}$, believed to be driven by p explanatory variables $\mathbf{X}$ plus an intercept
- Each $Y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_p x_{ip} + \epsilon$
- We can write this in matrix notation as $\mathbf{Y} = \mathbf{X}\beta + \epsilon$
- This allows us to estimate the individual impact that changes to a specific variable will have on future observations while controlling for the impact of other (correlated) variables

Ordinary Least Squares (OLS) Regression

- Collectively, the standard technique for regression with one or more is called ordinary least squares (OLS)
- OLS finds the vector (straight line) that minimizes the squared vertical distance between the line and each of the data points -- We refer to this squared distance as the sum of squared error. We want to minimize it.



Categorical Data

- Frequently, somes variables are discrete categories (gender, race, education level, etc)
- R will assume you'd like to regress an explanatory variable categorically if the column is stored as a `factor`, and generate the categories automatically for you
- We can capture this using linear regression by adding $k-1$ binary (taking values 1 or 0) variables into our model for a variable with k different levels
- If X_j is a categorical variable:
    - $X_j = 0 \implies X_j\beta_j = 0$
    - $X_j = 1 \implies X_j\beta_j = \beta_j$

Example: Wage Data

- In the 1970s Harris Trust and Savings Bank was sued for discrimination on the basis of gender. The following dataset is a collection of wages for bank employees

Variables

Explanatory

- `Educ`: Education, either 'HighSchool', 'Bachelors', or 'Graduate'
- `Exper`: months of previous work experience (before hire at bank)
- `Sex`: "Male" or "Female"
- `Senior`: months worked at bank since hire
- `Age`: age in months

Response

- `Bsal`: annual salary at time of hire

Glimpse of data

```
glimpse(wages)
```

```
## Observations: 93
## Variables: 6
## $ Bsal      <int> 5040, 6300, 6000, 6000, 6000, 6840, 8100, 6000, 6000, 6
## $ Sex       <fct> Male, Male, Male, Male, Male, Male, Male, Male, Male, M
## $ Senior    <int> 96, 82, 67, 97, 66, 92, 66, 82, 88, 75, 89, 91, 66, 86
## $ Age       <int> 329, 357, 315, 354, 351, 374, 369, 363, 555, 416, 481,
## $ Exper     <dbl> 14.0, 72.0, 35.5, 24.0, 56.0, 41.5, 54.5, 32.0, 252.0,
## $ Education <fct> Graduate, Graduate, Graduate, Bachelor, Bachelor, Gradu
```

Fitting a model

- R allows you to use formula objects to interact with your data using column names

```
model <- lm(Bsal ~ Education + Exper + Sex + Age, data=wages)
broom::tidy(model) %>% kable(format="markdown", digits=3) # View t
```

term	estimate	std.error	statistic	p.value
(Intercept)	4541.806	307.768	14.757	0.000
EducationGraduate	378.285	131.869	2.869	0.005
EducationHighSchool	-256.727	180.654	-1.421	0.159
Exper	0.051	1.150	0.045	0.964
SexMale	746.467	141.848	5.262	0.000
Age	1.109	0.786	1.411	0.162

- Note that R has automatically converted the 'Sex' and 'Education' variables to categorical variables and added categories as necessary
  - The 'missing' category is captured by the intercept

Additional Syntax in R

- Can also use `Bsal ~ .` to regress a column named `Bsal` against everything else in the data frame
- Can use `summary` function to obtain an easy-to-read output

```
model2 <- lm(Bsal ~ ., data=wages)
summary(model)
```

```
##
## Call:
## lm(formula = Bsal ~ Education + Exper + Sex + Age, data = wages)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -1050.48  -389.96   -24.56   321.94  2021.29
##
## Coefficients:
##                        Estimate Std. Error t value Pr(>|t|)
## (Intercept)          4541.80562  307.76782  14.757  < 2e-16
## EducationGraduate     378.28523  131.86917   2.869  0.00517
## EducationHighSchool -256.72742  180.65427  -1.421  0.15886
## Exper                   0.05148    1.15002   0.045  0.96440
## SexMale               746.46733  141.84809   5.262 1.01e-06
```

Interpreting the output

- **estimate**: the estimated value of the β coefficient for that explanatory variable.
  - For most coefficients, the way to interpret this is "*for every 1 unit increase in X, we observe a* β *unit increase in* Y."
  - For the **intercept**: the interpretation is "*the expected (average) value for* Y *if all the* X *variables are* 0". If we have categorical variables, the baseline category is included here.
- **std.error**: The standard error estimate for the coefficient
- **statistic**: The t-statistic for deviation
- **p.value**: The p-value implied by the t-statistic
  - The interpretation of the p-value for a particular coefficient $\hat{\beta}_j$ is "the probability of calculating a $\hat{\beta}_j$ this extreme or more extreme **assuming the null hypothesis is true** (in this case, null hypothesis is $\beta_j = 0$)

Prediction

```
x_star <- data.frame(Age=329, Education='HighSchool', Exper=14.0,
predict(model, x_star, interval='prediction', level=0.95)
```

```
##        fit      lwr      upr
## 1 5397.236 4218.215 6576.257
```

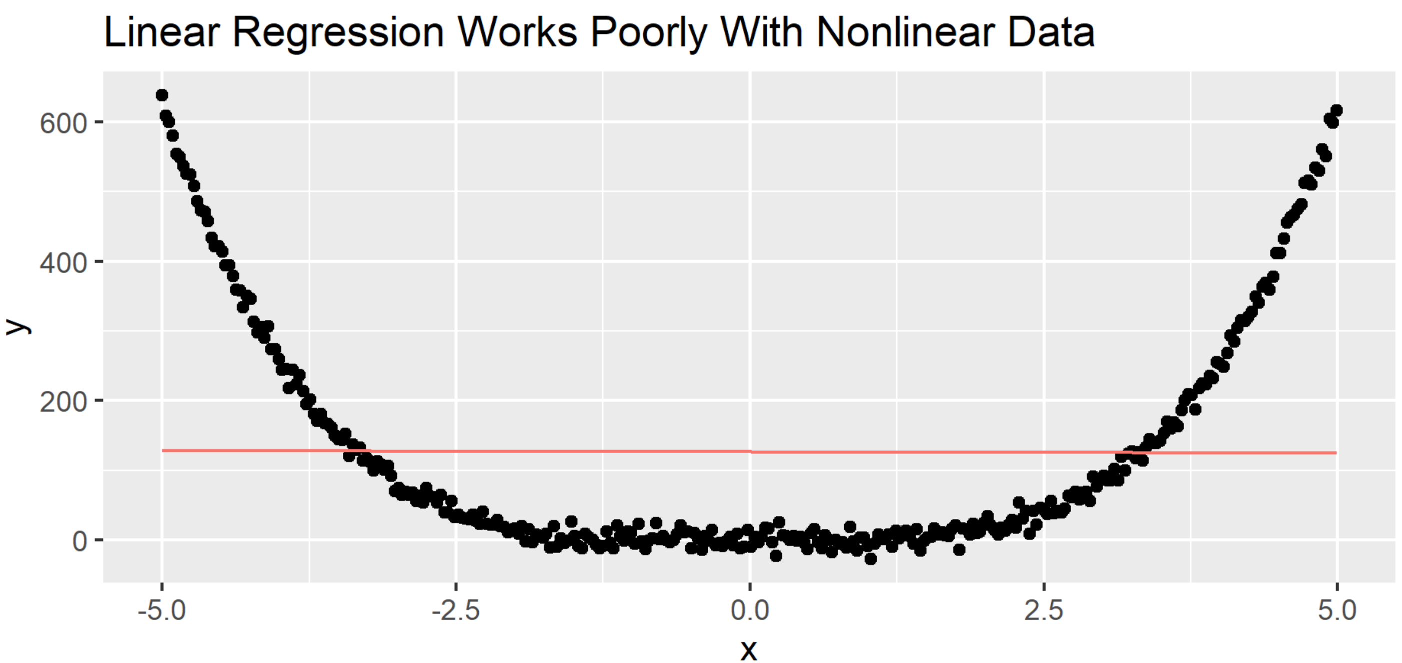
- Code above shows how to obtain an estimate ('fit') as well as the lower and upper bounds of the 95% prediction interval
- Types of uncertainty estimates for predictions:
  - **Confidence interval** (interval='confidence') captures the uncertainty inherent in estimating β - this is our best guess for the average value of Y at X
  - **Prediction interval** (interval='prediction') captures the uncertainty in obtaining $\hat{\beta}$, **plus** the uncertainty from the error inherent in Y

Checking Assumptions of Linear Regression

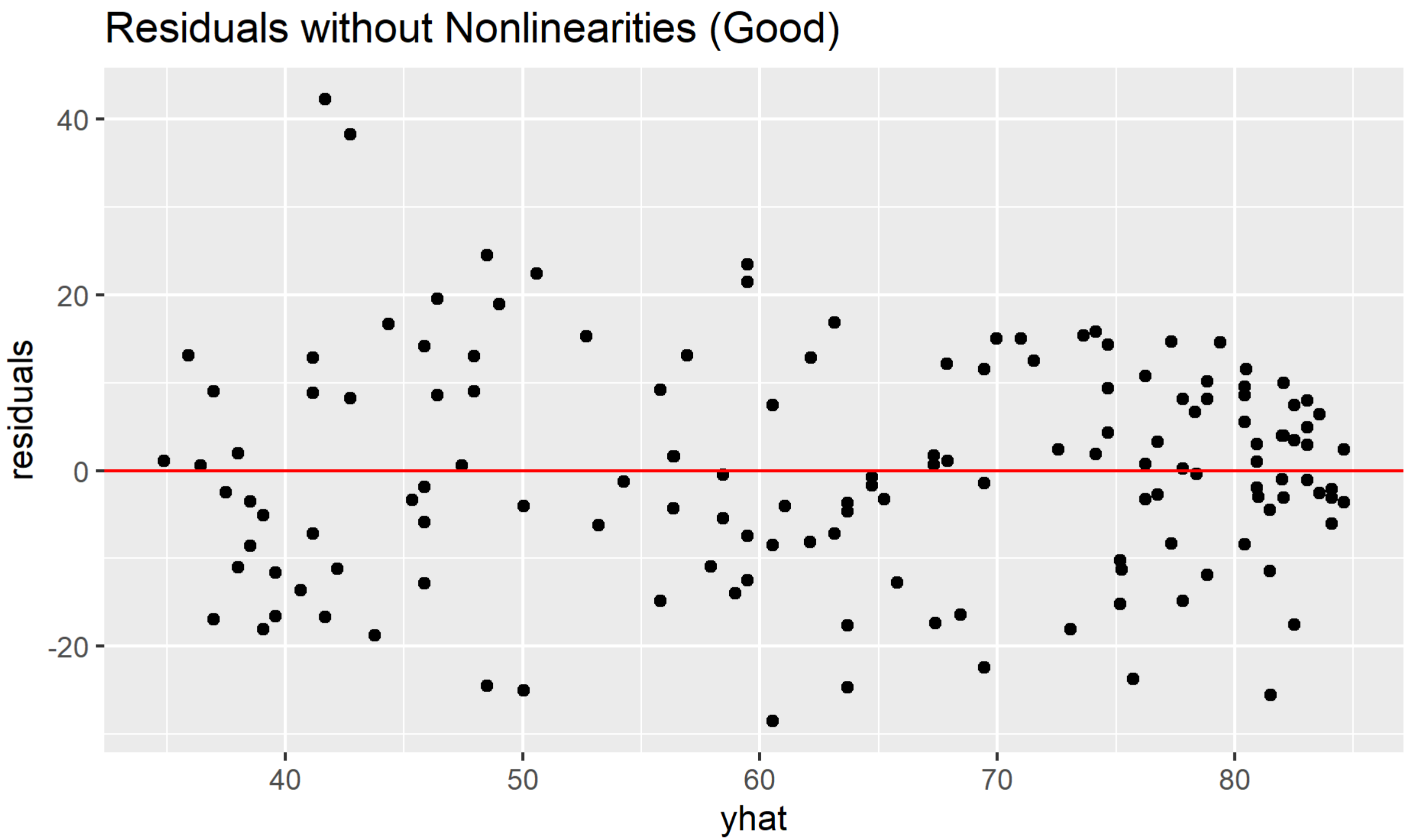
- OLS only gives unbiased estimates if four assumptions are satisfied
  - **Linearity**: Y cannont depend on $\mathbf{X}$ in a nonlinear way
  - **Normality**: The error must be normally distributed, and centered at 0. Note: $\mathbf{X}$ can be distributed however you want - it's **just the error** ϵ that needs to be normally distributed
  - **Constant Error** The amount of error can't change as the predicted value changes
  - **Independence**: Each individual Y_i can't depend on any of the other Y_i's except via their individual X values
- If these assumptions don't hold, the estimates $\hat{\beta}, \hat{\sigma}^2$ (and the p-values) are not guaranteed to be accurate

Assumption 1: Linearity

- **How to check:** Plot the predicted value $\hat{Y}$ against residuals
  - Values should be centered around 0 at every value of $\hat{Y}$
- You can fix this by transforming Y or X to make the relationship linear - but remember then that your predictors, confidence intervals, etc, are all going to be in the transformed space

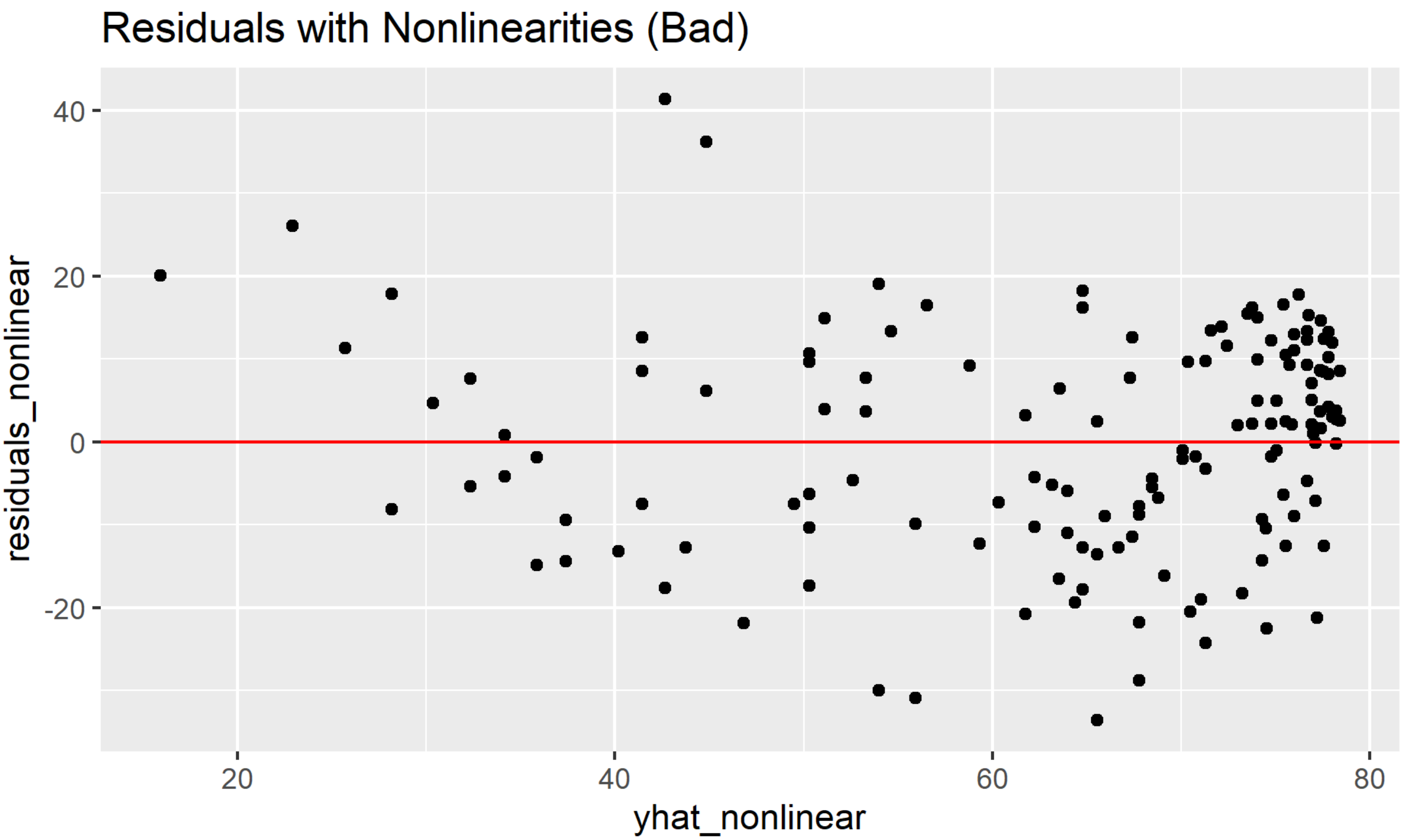


Assumption 1: Linearity



- DON'T worry if the data is bunched in some areas left-to-right
- DO worry if the data appears to be bunched above/below the line

Assumption 1: Linearity

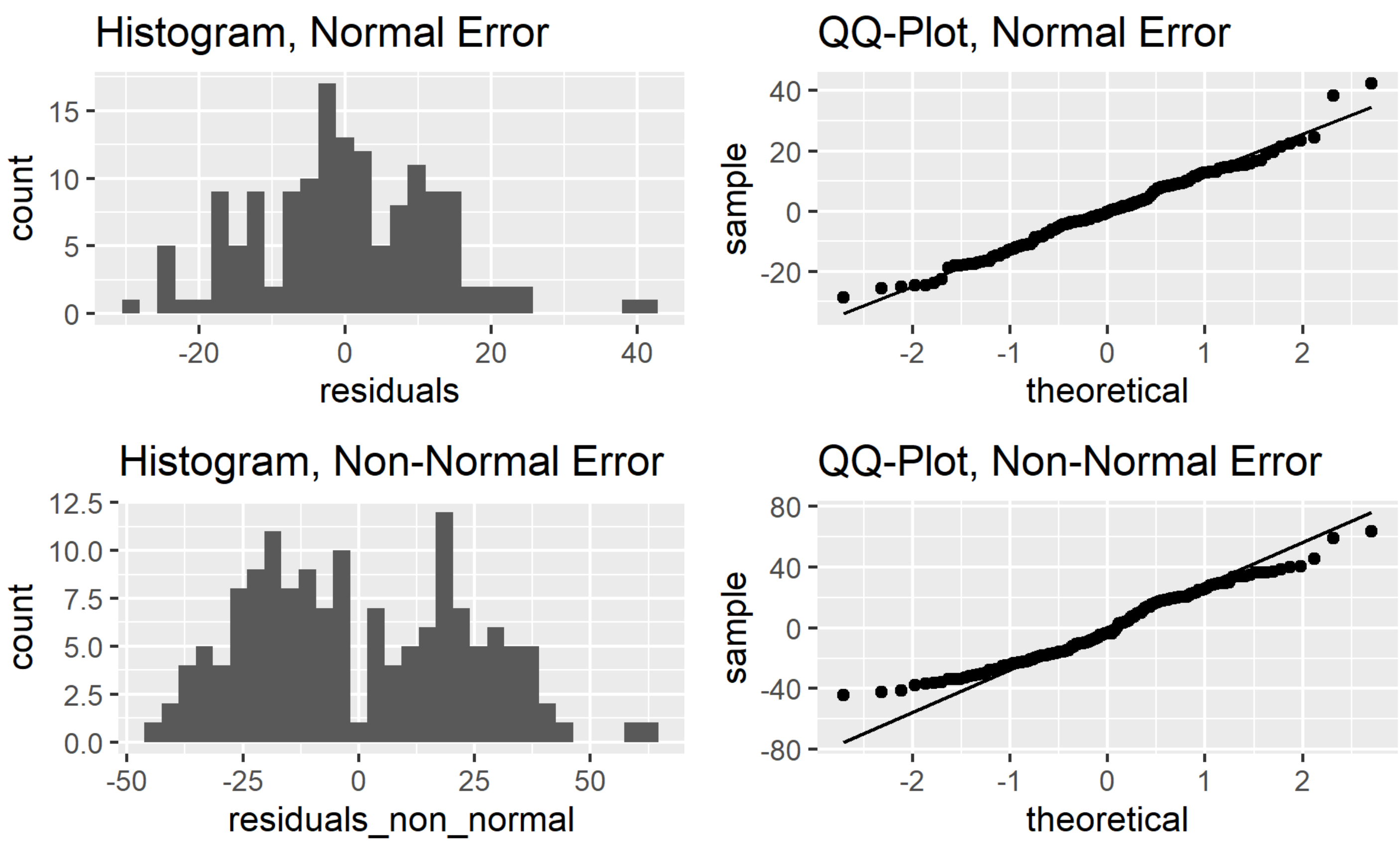


- DON'T worry if the data is bunched in some areas left-to-right
- DO worry if the data appears to be bunched above/below the line

Assumption 2: Normality

- ϵ must be distributed **normally** - i.e. from a bell curve
- **How to check**: Make a histogram and QQ-plot of the residuals, and examine if the data appears to be normally distributed
    - You should observe a roughly bell-shaped curve. Anything else indicates that the normality assumption is violated

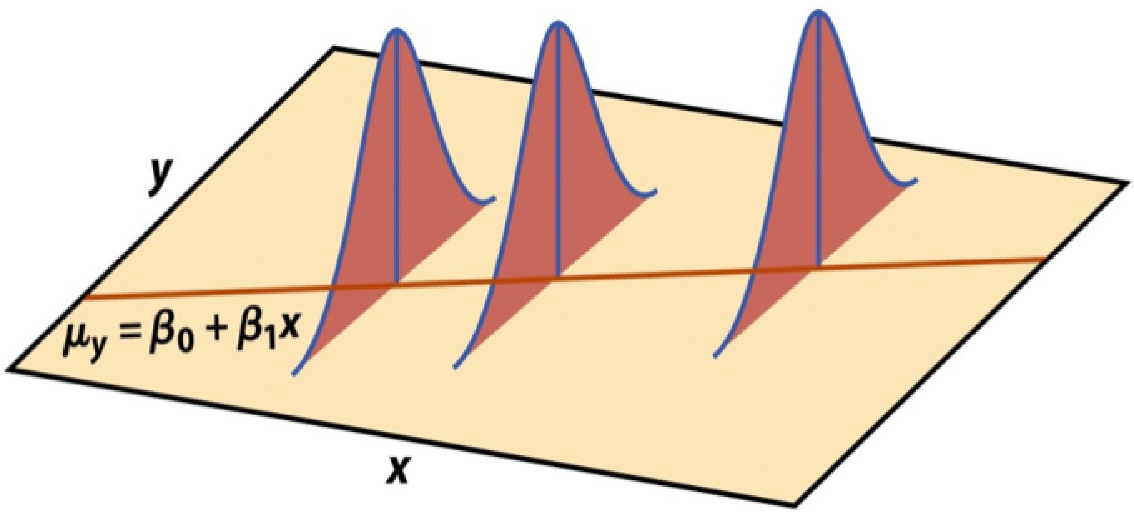
Assumption 2: Normality



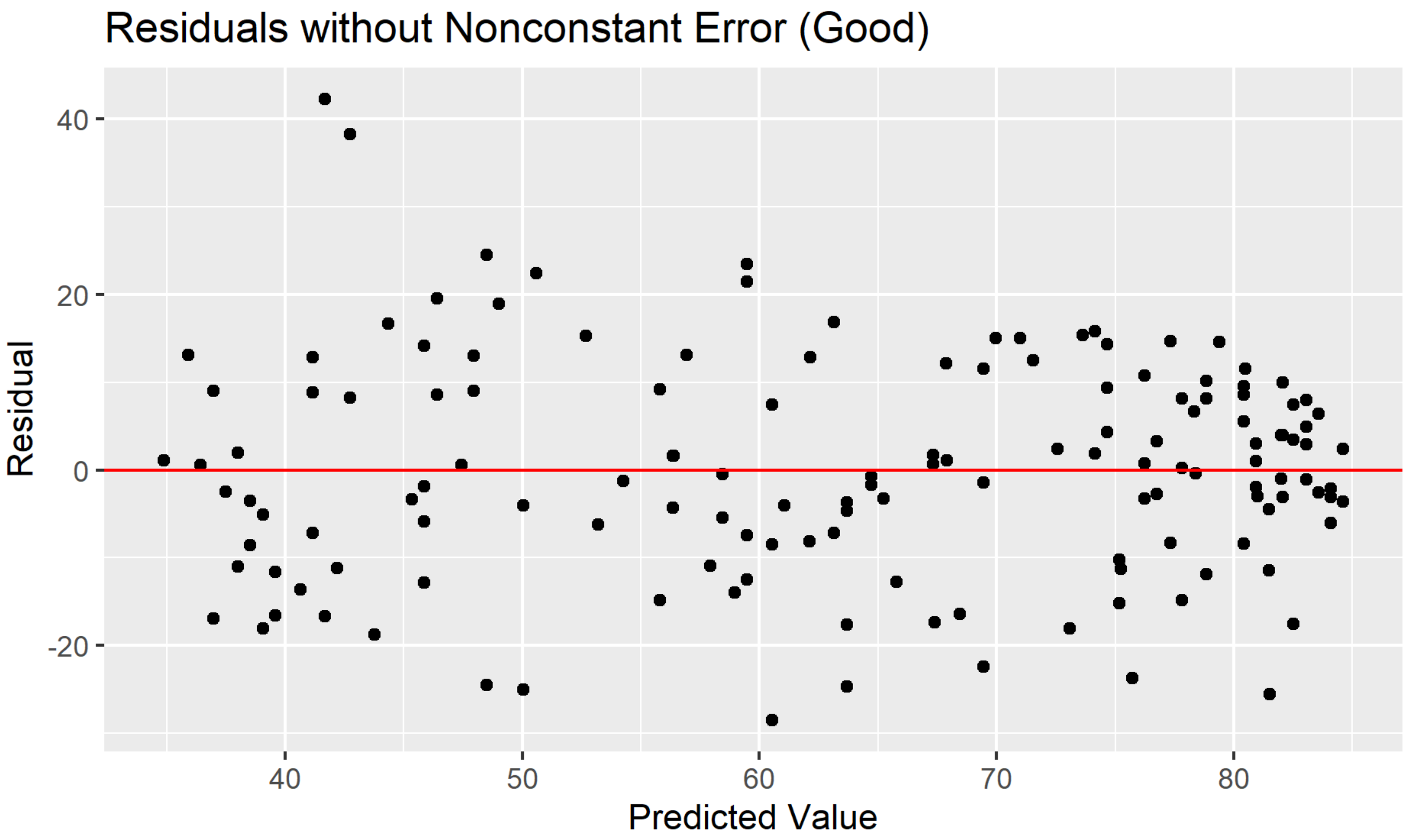
- DON'T worry if the histogram shows a somewhat spikey pattern - this happens a lot just due to inherent randomness if your sample

Assumption 3: Constant Error

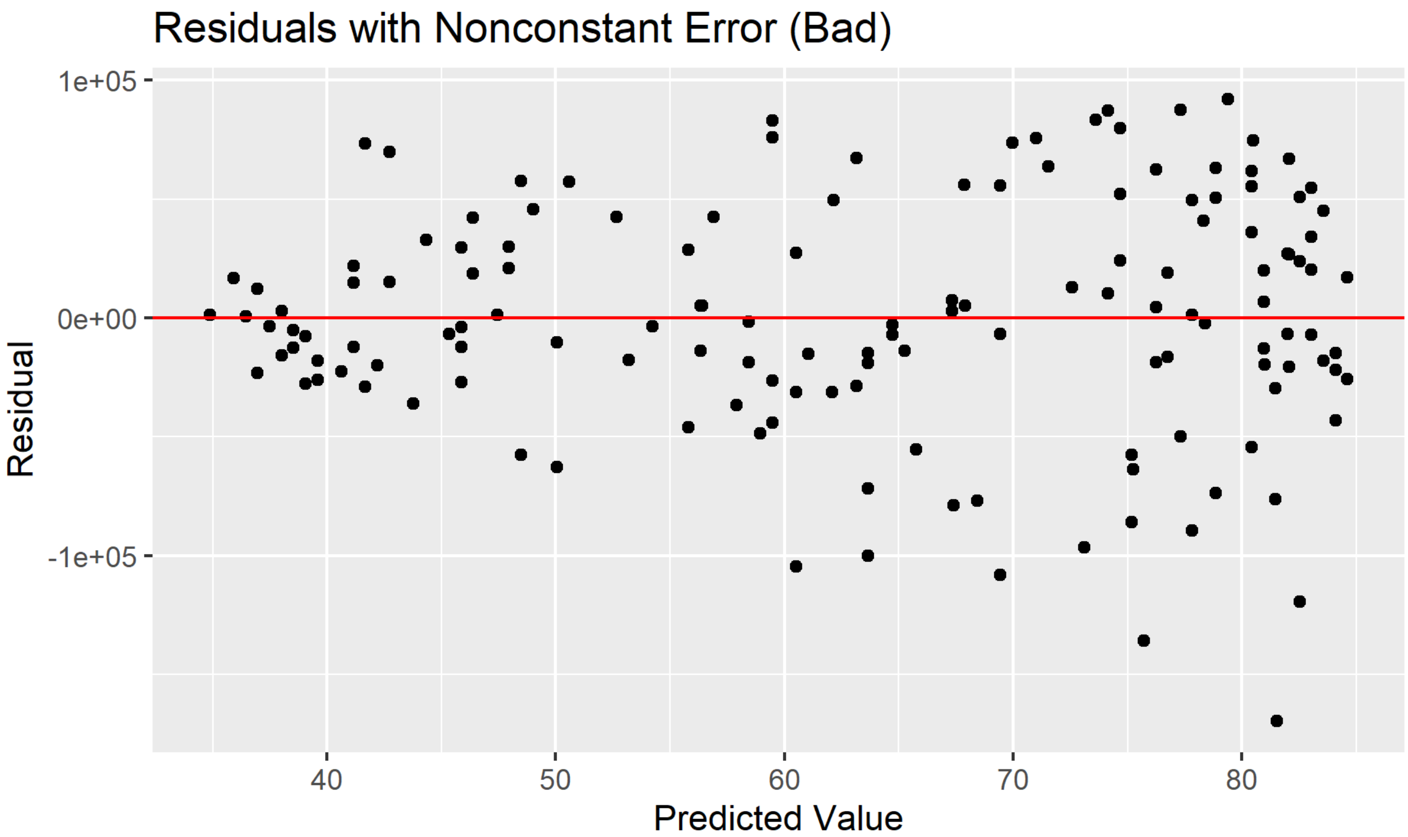
- The expected squared error σ^2 can't change as X changes
- **How to check**: Plot the predicted value $\hat{Y}$ against residuals. The spread above/below zero shouldn't change.



Assumption 3: Constant Error



Assumption 3: Constant Error



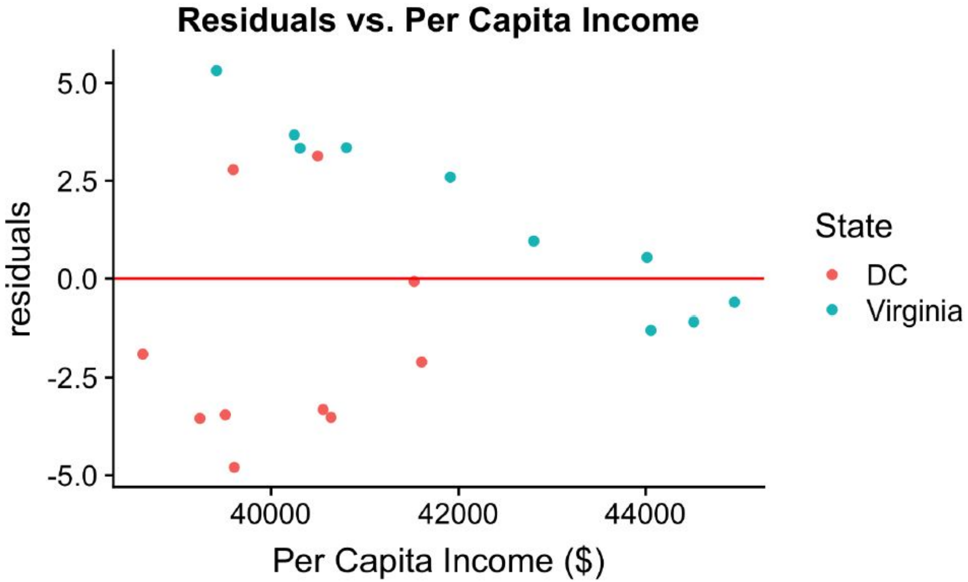
- Note how the residuals get larger as the predicted value increases. This is bad.

Assumption 4: Independence

- Each Y_i can't depend in some way on any other Y_j, beyond what's captured in X
- Common issues with this assumption are:
  - **Serial effect**: If data are collected over time, there is a chance of autocorrelation in the dataset
  - **Cluster effect**: If Y depends on some variable that's not included in your model

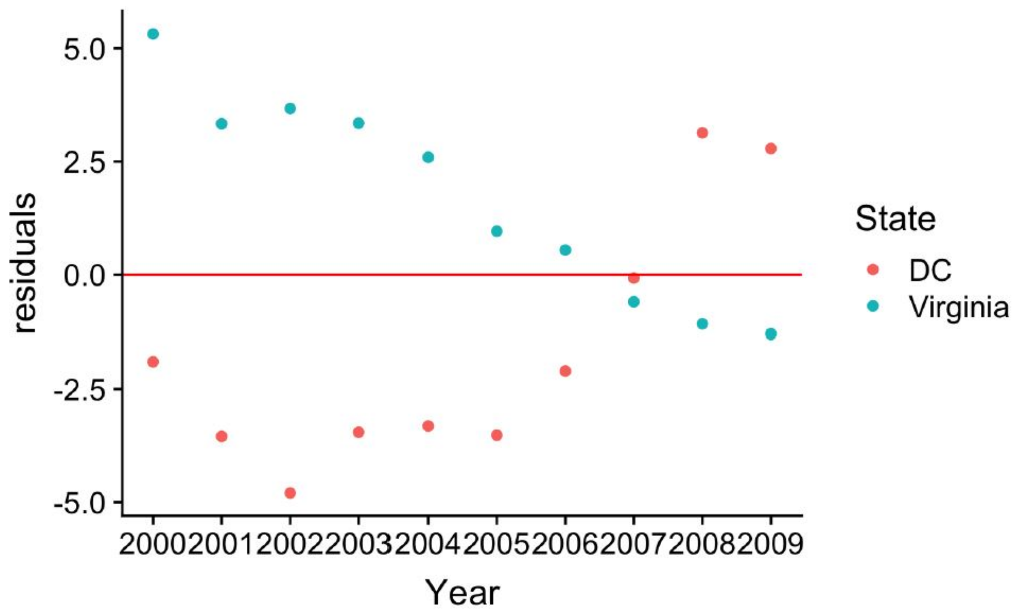
Example Residuals: Cluster Effect

```
ggplot(data=pew_data, mapping = aes(x=percapitaincome,y=residuals,
  geom_point() +
  geom_hline(yintercept=0,color="red") +
  labs(title="Residuals vs. Per Capita Income",
       x="Per Capita Income ($)")
```



Example Residuals: Serial Effect

```
ggplot(data=pew_data, aes(x=Year,y=residuals,color=State)) + geom_
  geom_hline(yintercept=0,color="red")+
  labs("Residuals vs. Year") +
  scale_x_continuous(breaks=seq(2000,2009,1))
```



Common Scenarios That Violate Assumptions

- **I'm predicting one or more time series**: Most time series suffer from some amount of *autocorrelation*, which violates the independence assumption. A common fix is to calculate the growth rate between each time step, and run your regression on that, though this isn't guaranteed to
- **I'm predicting an index value, like app ratings**: Because indexes are typically bounded, the normality assumption breaks down as we get closer to our bounds. Try dividing your data into , and using *multinomial regression*
- **I'm predicting the number of times something happens**: Similarly, as Y approaches 0, the assumption of normality breaks down . This isn't a huge problem if your observations aren't close to zero. Otherwise, consider Poisson regression for a more appropriate model.

Cautions

- Avoid extrapolation:
  - Relationships can change at different portions of the data
  - Almost all continuous functions are locally linear - but a nonlinear trend might emerge as you extend beyond the scope of your data
- Regression shows only correlation, not causation
  - Proving causality requires a carefully designed experiment or carefully accounting for confounding variables in an observational study
- Be careful of providing variables that are too correlated
  - You can use model selection techniques to help understand which variables you should retain
- Model selection is an iterative process
  - Don't be afraid to change your model based on the outcome of initial regressions

Important Topics We Didn't Cover:

- **Interaction terms**: What to do when some of your variables might produce an additional response when viewed together
- **Model selection**: How to know which variables to include in your model
- **Outlier detection**: Use of Cook's Distance, robust regression, and other techniques for handling outliers
- **Logistic Regression**: When your observed variable is a binary (yes/no) response
- **Multinomial Regression**: Similar to logistic regression, when your response is one or more discrete categories
- **Penalized regression**: Wide class of techniques used to obtain more stable estimates of β at the expense of an unbiased estimate
- **Poisson regression**: Used to model count-based data
- **Bayesian approaches to regression**: How to use priors to gain estimates of the distribution of $\hat{\beta}, \hat{\sigma^2}$

Try it out

- Download and save the .Rmd from here so we can step through exercises together