# Simple Linear Regression

## Fitting Models to Data Not Data to Models
## Model Fitting Series - With Applications in R

Jesse Ghashti

October 7, 2025

Centre for Scholarly Communication
The University of British Columbia | Okanagan Campus | Syilx Okanagan Nation Territory

# Welcome, feel free to get ready while we wait to start...

## Install Required Packages

```r
packages <- c("dplyr", "mgcv", "ggplot2", "gratia")
toInstall <- packages[!(packages %in%
                            installed.packages()[,"Package"])]
if(length(toInstall)) install.packages(toInstall)

library(dplyr)    # for data wrangling
library(mgcv)     # for modeling
library(ggplot2) # for plotting
library(gratia)   # for ggplot-based model graphics
```

# Workshop Series Overview

| Session | Topic | Date/Time |
|---|---|---|
| **1** | **Simple Linear Regression** | **Oct 7, 9:00 AM** |
| 2 | Fitting Linear Models in R | Oct 8, 10:30 AM |
| 3 | Multiple Linear Regression in R | Oct 16, 4:00 PM |
| 4 | Interaction Terms & Hierarchical Linear Models | Oct 21, 11:00 AM |
| 5 | Generalized Linear Models | Oct 23, 4:00 PM |
| 6 | Generalized Additive Models (GAMs) | Oct 28, 11:00 AM |
| 7 | Interpreting & Predicting from GAMs | Oct 29, 10:30 AM |
| 8 | Hierarchical GAMs | Nov 4, 12:00 PM |
| 9 | Penalized Models | Nov 18, 11:00 AM |
| 10 | Survival Models | Nov 25, 11:00 AM |
| 11 | Nonparametric Models | Dec 2, 11:00 AM |

# New Here?

<——— New to R? Check out the Fun-damentals of R series!

GitHub with code for todays workshop (copy and paste code available in these slides as well) ———>

# Today's Learning Objectives

**Definition 1:** *Workshop Goals*

By the end of this session, you will be able to:

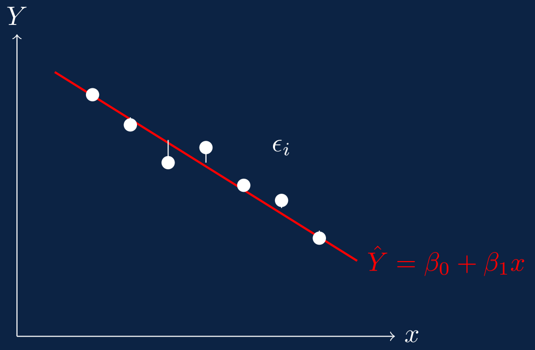
1. **Visualize** simple linear regression models
2. **Explain** the assumptions of linear models
3. **Understand** how coefficients are estimated (Least Squares)
4. **Identify** when linear models break down
5. **Interpret** R output and diagnostic plots

**Approach:** Visual and spatial learning with hands-on R coding

# What is Simple Linear Regression?

**Definition 2:** *Linear Model*

A statistical model that assumes a linear relationship between:
– **Y**: Response variable (dependent)
– **x**: Predictor variable (independent)




Mathematically:

$$Y_i = \beta_0 + \beta_1 x_i + \epsilon_i,$$

- $\beta_0$: Intercept
- $\beta_1$: Slope
- $\epsilon_i$: Random error

ex 1 Height vs Age

ex 2 Sales vs Advertising

# How Do We Estimate the Coefficients?

**Definition 3:** *Least Squares Method*

Find $\hat{\beta}_0$ and $\hat{\beta}_1$ that minimize the sum of squared residuals (SSE)

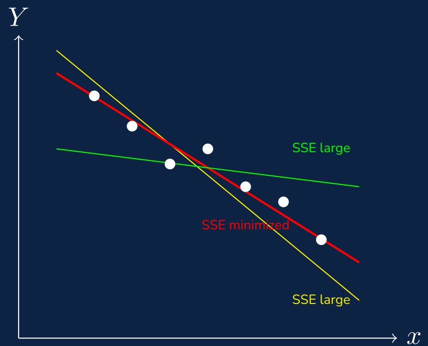
**Objective Function:**

$$\text{SSE} = \sum_{i=1}^{n}(Y_i - \hat{Y}_i)^2 = \sum_{i=1}^{n}(Y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i)^2$$

**Solution:**

$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(x_i - \bar{x})^2}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{x}$$



# The Five Key Assumptions

1. **Certainty in x**: No measurement error in predictor
2. **Linearity**: $E(Y|x) = \beta_0 + \beta_1 x$
3. **Homoscedasticity**: Var$(Y|x) = \sigma^2$ (constant)
4. **Independence**: $Y_i \perp Y_j$ for $i \neq j$
5. **Normality**: $\epsilon_i \sim N(0, \sigma^2)$

**Why do these matter?**

– Violations affect validity of inference (p-values, confidence intervals)

– Can lead to biased or inefficient estimates

– May require different modelling approaches

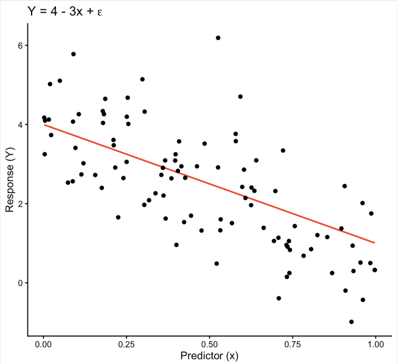
# Let's Generate Some Data!

```r
set.seed(10042025)
d0 <- tibble(
  x = runif(n = 100),                    # uniform predictor
  mu = 4 - 3 * x,                        # true mean
  e = rnorm(n = length(x),               # normal error
            mean = 0, sd = 1),
  Y = mu + e                             # observed values
)
head(d0)
```

**True Model:** $Y = 4 - 3x + \epsilon$ where $\epsilon \sim N(0, 1)$

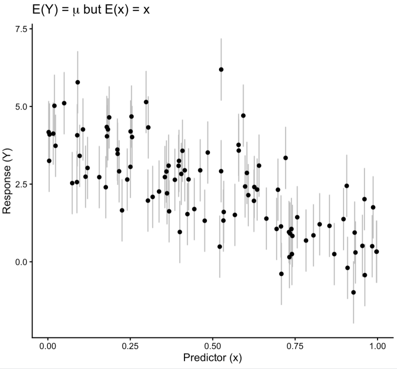
# Visualizing the Data and True Relationship

```r
ggplot(d0) +
  geom_line(aes(x, mu),
            col = 'red',
            lwd = 1) +
  geom_point(aes(x, Y)) +
  labs(x = 'Predictor (x)',
       y = 'Response (Y)',
       title = expression(
       'Y = 4 - 3x +'~epsilon)
       ) +
  theme_classic(base_size = 15)
```



# Assumption 1: Certainty in x

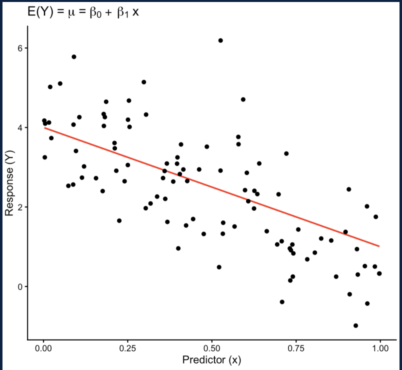
```r
# Visualize uncertainty in Y only
ggplot(d0) +
  geom_errorbar(
    aes(x, ymin = Y - 1,
           ymax = Y + 1),
    color = 'grey') +
  geom_point(aes(x, Y)) +
  labs(x = 'Predictor (x)',
    y = 'Response (Y)',
    title = expression(
      'E(Y) ='~mu~
      'but E(x) = x'))
```




We assume x is measured perfectly, all uncertainty is in Y.

# Assumption 2: Linearity

```r
# The mean of Y is linear in x
ggplot(d0) +
  geom_line(aes(x, mu),
            col = 'red',
            lwd = 1) +
  geom_point(aes(x, Y)) +
  labs(
    x = 'Predictor (x)',
    y = 'Response (Y)',
    title = expression(
      'E(Y) ='~mu~'='~
      beta[0]~+~beta[1]~x))
```

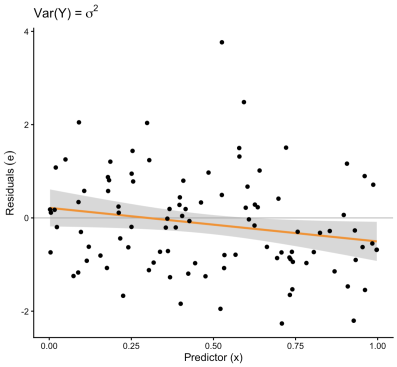



**What if it's not linear?**

- Polynomial terms: $x^2, x^3$
- Transformations: $\log(x), \sqrt{x}$

# Assumption 3: Homoscedasticity

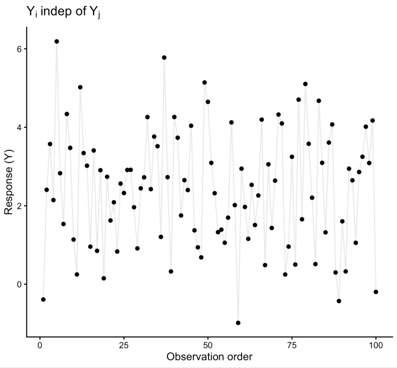
```r
# Plot residuals vs x
ggplot(d0) +
  geom_hline(yintercept = 0,
             color = 'grey') +
  geom_smooth(aes(x, e),
              col = 'darkorange',
              method = 'lm',
              se = TRUE) +
  geom_point(aes(x, e)) +
  labs(x = 'Predictor (x)',
    y = expression(
      'Residuals'~(e)),
    title = expression(
      'Var(Y) ='~sigma^2))
```




Look for constant spread of residuals across all x values

# Assumption 4: Independence

```r
# Check observation order
ggplot(d0,
       aes(seq(nrow(d0)), Y)) +
  geom_point() +
  geom_path(alpha = 0.1) +
  labs(x = 'Observation order',
    y = 'Response (Y)',
    title = expression(
      Y[i]~'indep of'~Y[j]))
```




✓ Random = Rad! ✓

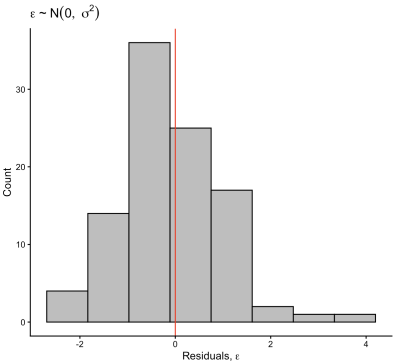
✗ Pattern = Problem! ✗

Common violations:

- Time series data, Spatial correlation,
- Repeated measures, Clustered data

# Assumption 5: Normality of Errors

```r
# Residuals histogram
ggplot(d0, aes(e)) +
  geom_histogram(color = 'black',
    fill = 'grey', bins = 8) +
  geom_vline(xintercept = 0,
    color = 'red') +
  labs(x = expression(
      'Residuals,'~e),
    y = 'Count',
    title = expression(
      epsilon~'~'~N(0,~sigma^2)))
```




**Being normal matters! (here)**

- Valid confidence intervals
- Accurate p-values
- Optimal predictions

# Fitting the Model

```r
# simple linear regression
m0 <- gam(Y ~ x, data = d0)
# model summary
summary(m0)
```

```
Parametric coefficients:
            Estimate Std. Error t value Pr(>|t|)
(Intercept)   4.2159     0.2001   21.03   < 2e-16 ***
x            -3.7210     0.3566  -10.43   < 2e-16 ***

R-sq.(adj) =  0.521   Deviance explained = 52.6%
```

Compare to true values: $\beta_0 = 4$, $\beta_1 = -3$

# Understanding Model Output

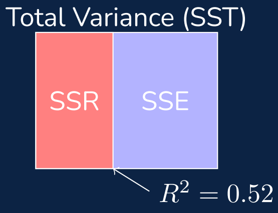
```r
# Extract coefficients
beta0_hat <- coef(m0)[1]
beta1_hat <- coef(m0)[2]

print(beta0_hat) # 4.216
print(beta1_hat) # -3.721
```

**Interpretation:**

- $\hat{\beta}_0 = 4.216$: Expected $Y$ when $x = 0$
- $\hat{\beta}_1 = -3.721$: Change in $Y$ per unit increase in $x$

**What is $R^2$?**

$$R^2 = \frac{\text{SSR}}{\text{SST}} = \frac{\text{Explained Var}}{\text{Total Var}}$$




52% of variance in $Y$ explained by $x$.

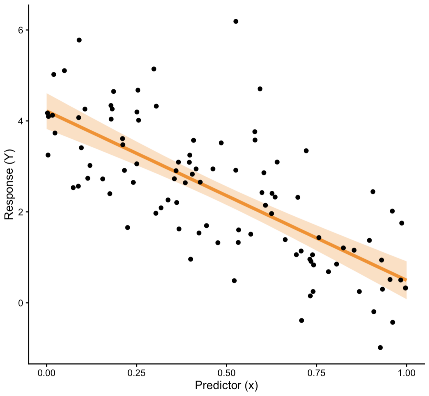
# Making Predictions

```r
# new data
newd0 <- tibble(x = seq(0, 1, by = 0.01))
# get predictions with standard errors
pred0 <- bind_cols(
  newd0, predict(m0,
              newdata = newd0,
              se.fit = TRUE) %>%
    as.data.frame()) %>%
  rename(mu_hat = fit) %>%
  mutate(lwr_95 = mu_hat +
                se.fit * qnorm(0.025),
    upr_95 = mu_hat +
                se.fit * qnorm(0.975))
```

```
# A tibble: 101 × 5
       x mu_hat se.fit lwr_95 upr_95
   <dbl>  <dbl>  <dbl>  <dbl>  <dbl>
 1  0      4.22  0.201   3.82   4.61
 2  0.01   4.18  0.197   3.79   4.57
 3  0.02   4.14  0.194   3.76   4.52
 4  0.03   4.10  0.191   3.73   4.48
 5  0.04   4.07  0.188   3.70   4.44
 6  0.05   4.03  0.185   3.67   4.39
 7  0.06   3.99  0.183   3.63   4.35
 8  0.07   3.96  0.180   3.60   4.31
 9  0.08   3.92  0.177   3.57   4.26
10  0.09   3.88  0.174   3.54   4.22
```

# Visualizing Predictions

```r
ggplot() +
  geom_ribbon(aes(x, ymin = lwr_95,
                     ymax = upr_95),
                     pred0,
                 fill = 'darkorange',
                 alpha = 0.3) +
  geom_line(aes(x, mu_hat),
            pred0,
            color = 'darkorange',
            linewidth = 2) +
  geom_point(aes(x, Y), d0) +
  labs(x = 'Predictor (x)',
       y = 'Response (Y)')
```




Orange band: 95% CI for the mean
Points: Observed data

# Types of Variance

Three components of variance:

1. **SST**: Total variation in Y (ignoring x)
2. **SSR**: Variation explained by the model
3. **SSE**: Unexplained variation (residuals)

$$\text{SST} = \text{SSR} + \text{SSE}$$

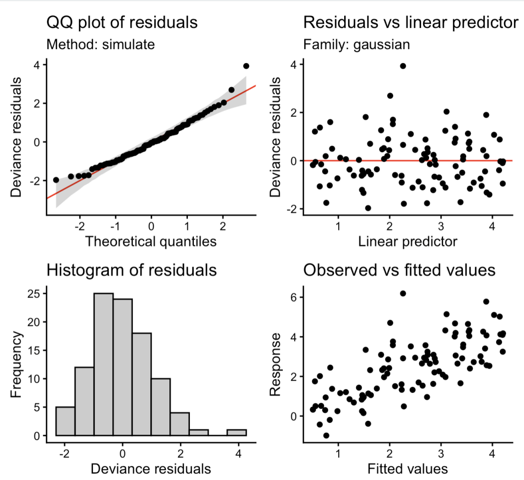
```r
d0$mu_hat <- predict(m0)
SST <- sum((d0$Y - mean(d0$Y))^2)
SSR <- sum((d0$mu_hat - mean(d0$Y))^2)
SSE <- sum((d0$Y - d0$mu_hat)^2)
SST
SSR + SSE
# R^2 = SSR / SST
SSR / SST

summary(m0)$r.sq # adjusted R^2
summary(m0)$dev.expl # R^2
```

# Model Diagnostics with gratia

```r
appraise(
  model = m0,
  method = 'simulate',
  n_simulate = 1000
)
```

What to look for:

- QQ-plot: Points on diagonal line
- Residuals vs Fitted: Random scatter
- Histogram: Bell-shaped
- Residuals vs x: No pattern



# When Linear Models Break

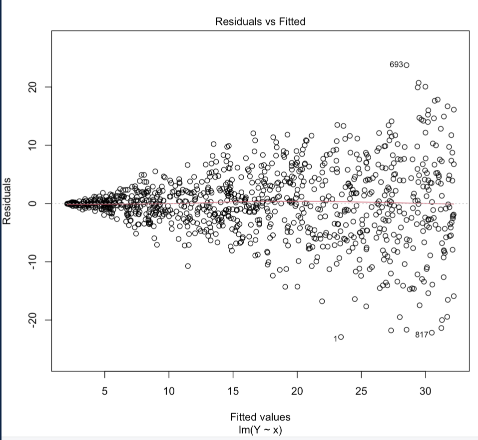
```r
m_bad <- gam(weight ~ Time, data = ChickWeight)
summary(m_bad)
```

| Problem | Detection | Solution |
|---|---|---|
| Non-linearity | Curved pattern in residuals | Transform x or Y, add polynomial terms |
| Heteroscedasticity | Funnel shape in residuals | Transform Y (log, sqrt), use weights |
| Non-normality | QQ-plot deviation | Transform Y, robust methods |
| Outliers | Large residuals | Investigate, robust regression |
| Autocorrelation | Pattern over time/space | Time series models, spatial models |

**Remember:** All models are wrong, but some are useful! - George Box

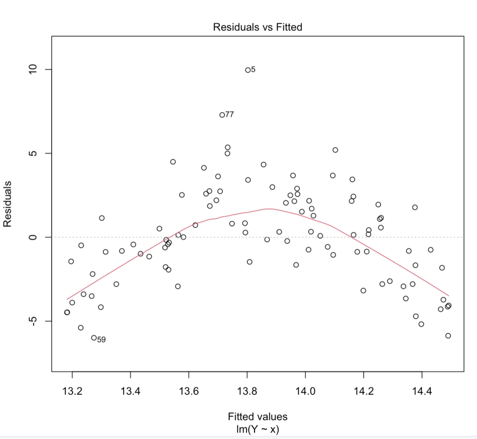
# What's the Problem? I

```r
set.seed(10042025)
dhet <- tibble(
  x = runif(1000, 0, 10),
  Y = 2 + 3*x +
      rnorm(1000, 0, sd = x))
mhet <- lm(Y ~ x,
           data = dhet)
plot(mhet, which = 1)
```



# What's the Problem? II

```r
set.seed(10042025)
dnlin <- tibble(
  x = runif(100, 0, 5),
  Y = 10 + 5*x - x^2 +
      rnorm(100, 0, 2))
mlin <- lm(Y ~ x,
           data = dnlin)
plot(mlin, which = 1)
```



# Exercise: Your Turn! (15 minutes)

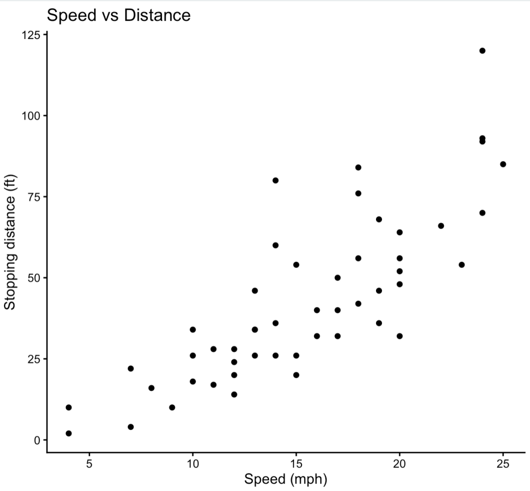
## Practice Exercise

**Dataset:** `cars` is built-in R dataset
– Speed of cars (mph) and stopping distances (ft)

```r
data(cars)
head(cars)

# 1. Explore the data (plot speed vs dist)
# 2. Fit a linear model (dist ~ speed)
# 3. Check assumptions using diagnostic plots
# 4. Interpret the coefficients
# 5. Predict stopping distance at 25 mph
# 6. Calculate and interpret R-squared
```

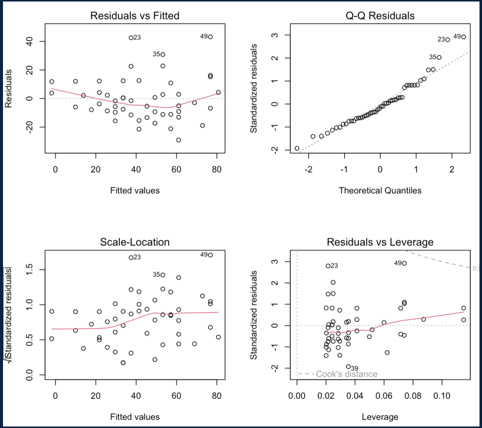
# Exercise Solution - Part 1

```r
# 1. Explore the data
ggplot(cars, aes(speed, dist)) +
  geom_point() +
  labs(x = "Speed (mph)",
       y = "Stopping distance (ft)",
       title = "Speed vs Distance")
```



# Exercise Solution - Part 2/3/4

```r
# 2. Fit the model
# alternative to 'gam'
carsmod <- lm(dist~speed,
              data = cars)
summary(carsmod)
# 3. Check Assumptions
par(mfrow = c(2, 2))
plot(carsmod)
# 4. Interpretation
coef(carsmod)
```




**Coefficients:**

– For each 1 mph increase in speed, stopping distance increases by 3.93 feet.

– The intercept seems to be meaningless here, why?

# Exercise Solution - Part 5/6

```r
# 5. Prediction at 25 mph
newDat <- data.frame(speed = 25)
predict(carsmod, newdata = newDat,
        interval = "confidence")
# 80.7 feet (95% CI: 71.6 - 89.9)

# 6. R-squared
summary(carsmod)$r.squared
# speed explains 65.1% of variation in stopping distance
```

# Key Takeaways

1. **Simple linear regression** models the relationship between two variables
2. **Least squares** finds the best-fitting line by minimizing SSE
3. **Five key assumptions** must be checked:
    - Certainty in x
    - Linearity
    - Homoscedasticity
    - Independence
    - Normality
4. **Diagnostic plots** help identify assumption violations
5. $R^2$ measures proportion of variance explained
6. When assumptions fail, we must reconsider.

**A good portion of this workshop series focuses on how to address that last point.**

# What's Next?

## Additional Resources:

- *An Introduction to Statistical Learning* (James et al.)
- *Linear Models with R* (Faraway)

## Additional Questions? Book an Appointment!

## Next Workshop:

**Multiple Linear Regression**
**October 8, 10:30 AM**
– Multiple predictors
– Multicollinearity
– Variable Transformations