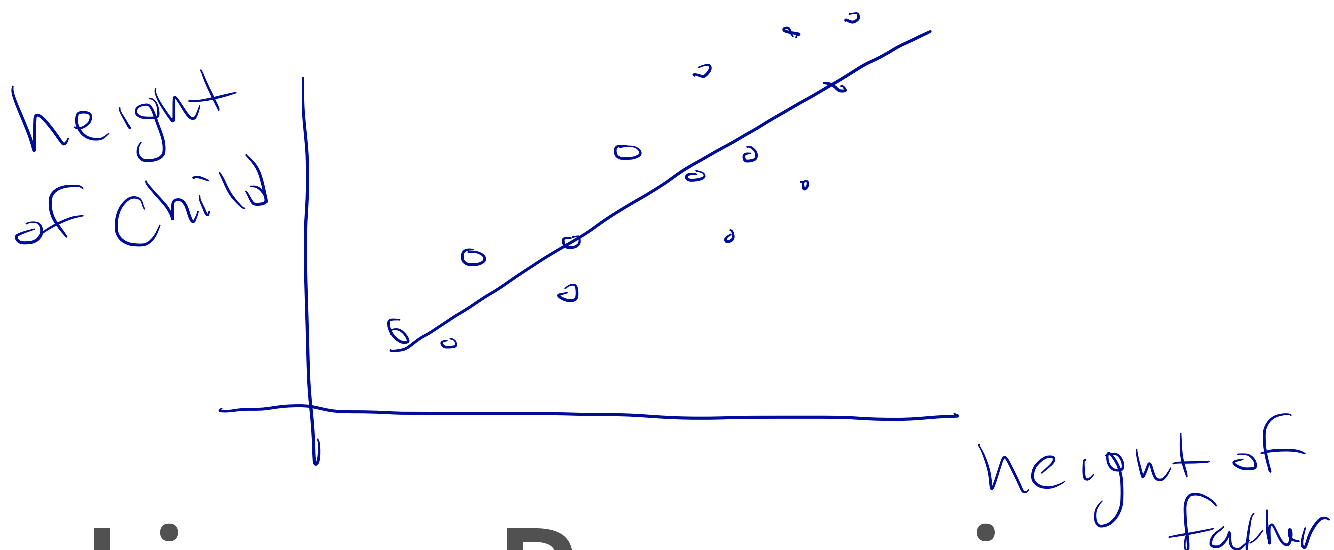
Classification trees outcome is binary.
(e.g., Yes/No, Sick/Not Sick, etc.)

Linear regression outcome variable is continuous.



Class 9 - Linear Regression

This Class

- Relationships between two continuous variables
- Linear Relationships: The equation of a straight line
- Relationships between two variables
- Linear regression models
- Estimating the coefficients: Least Squares
- Interpreting the slope with a continuous explanatory variable
- Prediction/Supervised learning using a linear regression model
- R^2 - Coefficient of Determination
- Introduction to Multiple Regression
- RMSE - Root mean Square error.

Relationships between two variables

Advertising Example

- Suppose that we are statistical consultants hired by a client to provide advice on how to improve sales of a particular product.
- The `Advertising` data set consists of the sales of that product in 200 different markets, along with advertising budgets for the product in each of those markets for three different media: TV, radio, and newspaper.

```
glimpse(Advertising)
```

```
## Observations: 200
## Variables: 4
## $ TV        <dbl> 230.1, 44.5, 17.2, 151.5, 180.8, 8.7, 57.5, 120.2, 8...
## $ radio     <dbl> 37.8, 39.3, 45.9, 41.3, 10.8, 48.9, 32.8, 19.6, 2.1,...
## $ newspaper <dbl> 69.2, 45.1, 69.3, 58.5, 58.4, 75.0, 23.5, 11.6, 1.0,...
## $ sales     <dbl> 22.1, 10.4, 9.3, 18.5, 12.9, 7.2, 11.8, 13.2, 4.8, 1...
```

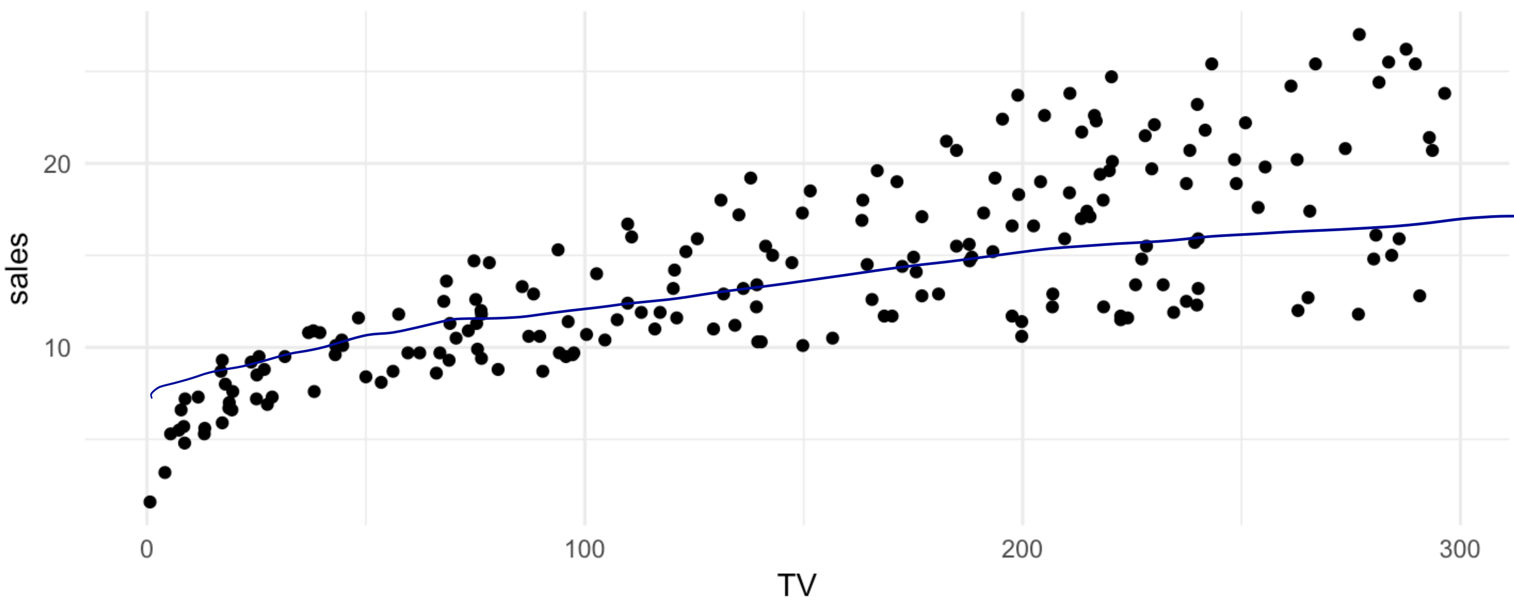
Advertising Example

- It is not possible for our client to directly increase sales of the product, but they can control the advertising expenditure in each of the three media.
- Therefore, if we determine that there is an association between advertising and sales, then we can instruct our client to adjust advertising budgets, thereby indirectly increasing sales.

Increasing sales through advertising

What is the relationship between `sales` and `TV` budget?

```
Advertising %>% ggplot(aes(x = TV, y = sales)) + geom_point() + theme_minimal()
```



positive
linear relationship

Question: How to use Sales and TV budget to estimate slope and y-intercept?

Increasing sales through advertising

- In general, as the budget for `TV` increases `sales` increases.
- Although, sometimes increasing the `TV` budget didn't increase `sales`.
- The relationship between these two variables is approximately linear.

Linear Relationships

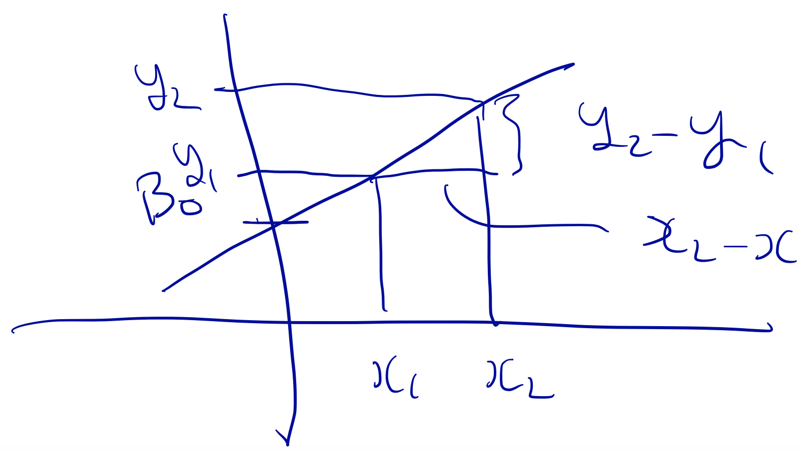
A perfect linear relationship between an independent variable x and dependent variable y has the mathematical form:

$$y = \beta_0 + \beta_1 x.$$

iopβ_0 is called the y-intercept and β_1 is called the slope.

When $x = 0 \Rightarrow y = \beta_0$ y-intercept

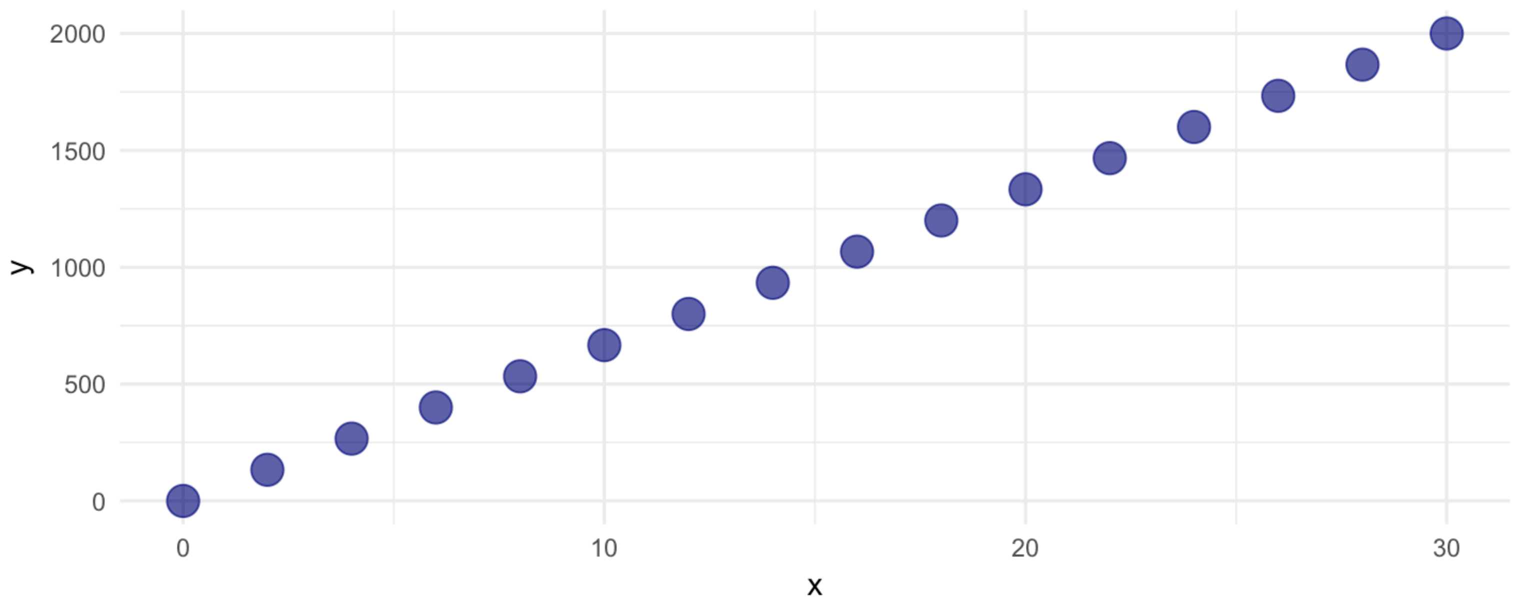
$$\beta_1 = \frac{y_2 - y_1}{x_2 - x_1}$$



Linear Relationships: The equation of a straight line

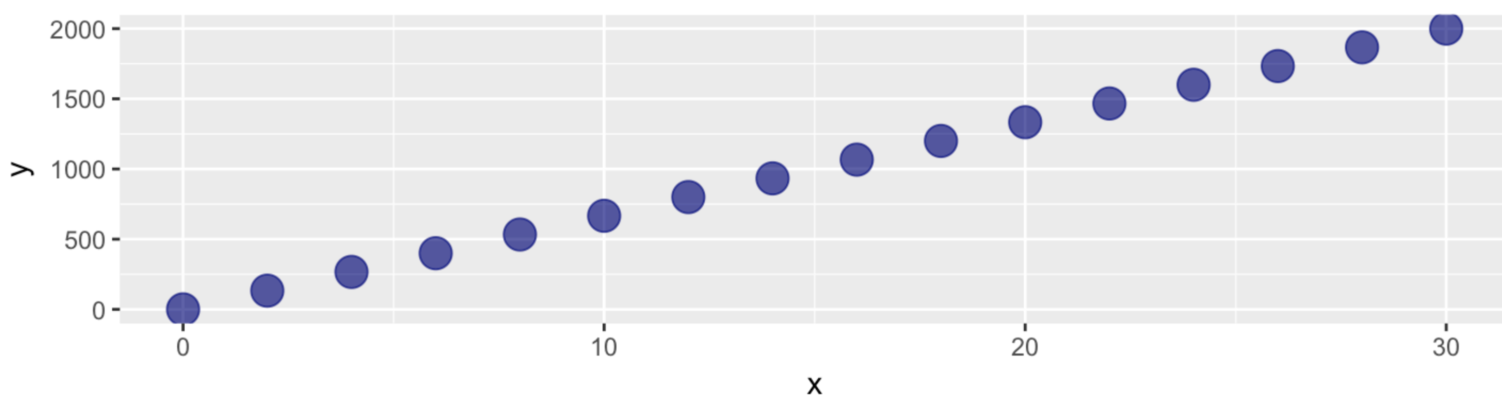
Linear Relationships: The equation of a straight line

If the relationship between y and x is perfectly linear then the scatter plot could look like:



Linear Relationships: The equation of a straight line

What is the equation of straight line that fits these points?



First four observations:

```
## # A tibble: 4 x 2
##         x        y
##     <dbl>    <dbl>
## 1       0   0.0000
## 2       2 133.3333
## 3       4 266.6667
## 4       6 400.0000
```

x_1, x_2 ; y_1, y_2

$$m = \frac{133 - 0}{2 - 0} = \frac{133}{2}$$

$$y = \frac{133}{2} x$$

Fitting a straight line to data

Use analytic geometry to find the equation of the straight line: pick two any points $(x^{(1)}, y^{(1)})$ and $(x^{(2)}, y^{(2)})$ on the line.

The slope is:

$$m = \frac{y^{(1)} - y^{(2)}}{x^{(1)} - x^{(2)}}.$$

So the equation of the line with slope m passing through $(x^{(1)}, y^{(1)})$ is

$$y - y^{(1)} = m(x - x^{(1)}) \Rightarrow y = mx + b,$$

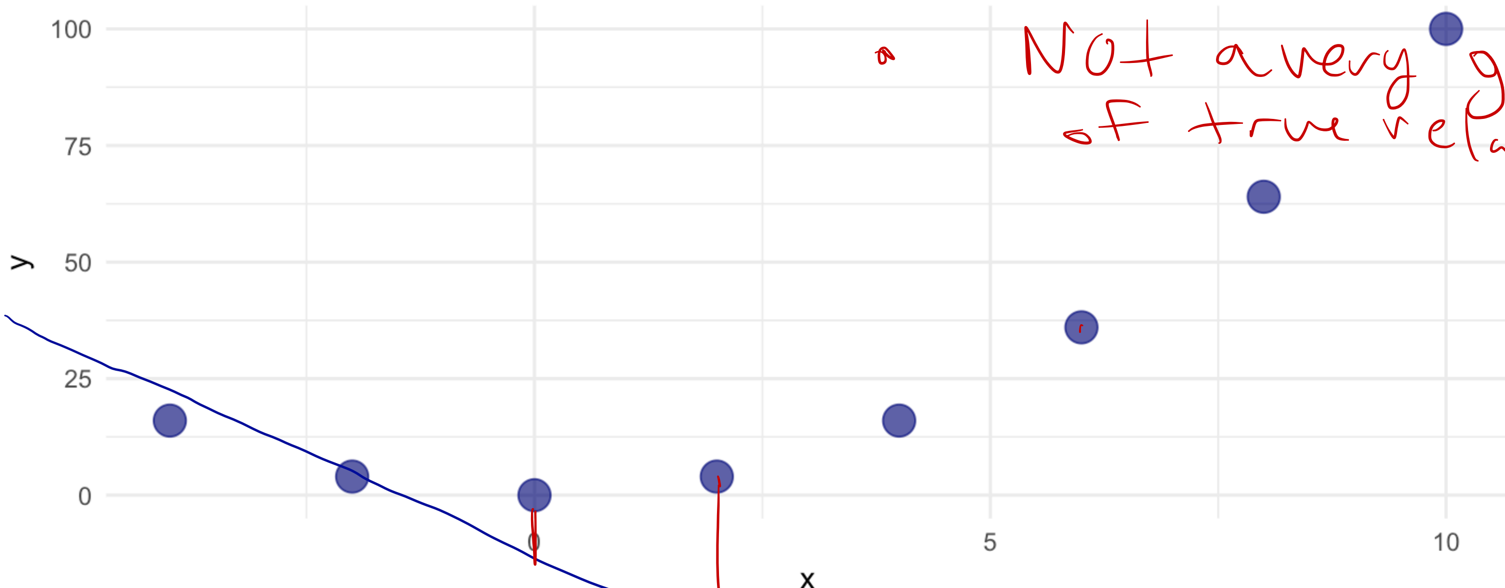
where $b = y^{(1)} - mx^{(1)}$.

Linear Relationships: The equation of a straight line

- Calculations can be done even though relationship is non linear.

What is the equation of the 'best' straight line that fits these points?

- Not a very good approx. of true relationship.



line that passes through $(-2, 4)$ is

$y = -6x - 8$

$m = \frac{16-4}{-4-(-2)} = -6$

```
## # A tibble: 4 x 2
##       x     y
##   <dbl> <dbl>
## 1    -4    16
## 2    -2     4
## 3     0     0
```

Relationships between two variables

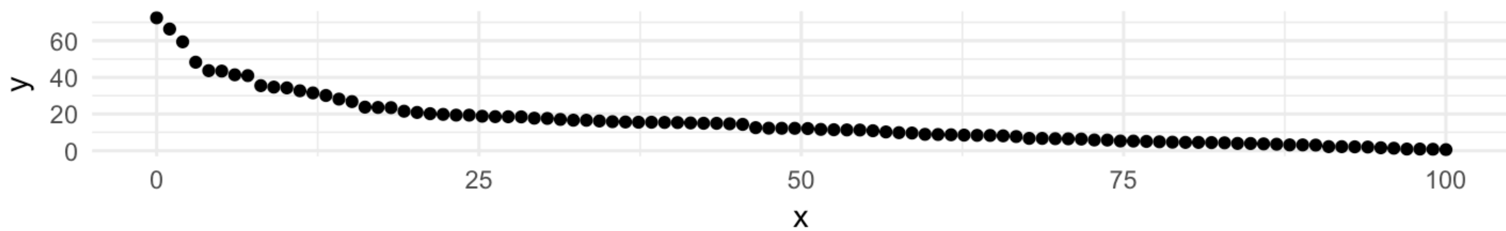
Relationships between two variables

- Sometimes the relationship between two variables in non-linear.
- If the realtionship is non-linear then fitting a straight line to the data is not useful in describing the relationship.

Example of Non-linear relationships

- Let y be life expectancy of a component, and x the age of the component.
- There is a relationship between y and x, but it is not linear.

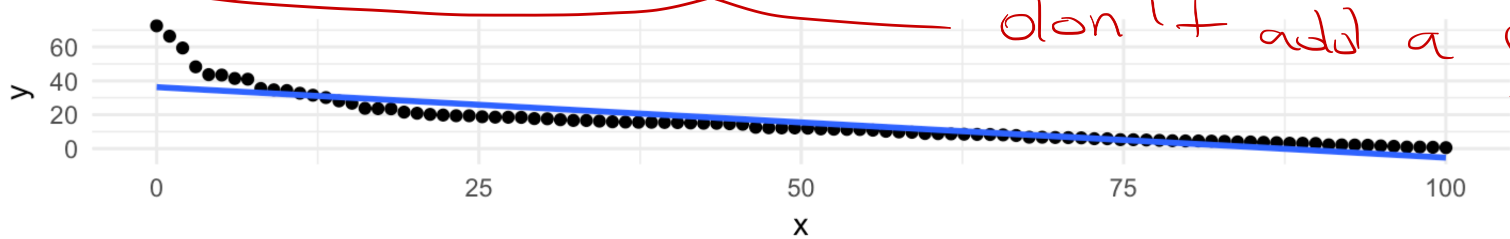
```
p <- data_frame(x = age, y = life_exp) %>%
  ggplot(aes(x = x, y = y)) + geom_point() + theme_minimal()
p
```




```
p + geom_smooth(method = "lm", se = F)
```

adds a linear regression line

don't add a confidence interval to regression line.



Tidy the Advertising Data

- Each market is an observation, but each column is the amount spent on TV, radio, newspaper advertising.

$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$

$Sales_i = \beta_0 + \beta_1 TV_i^2 + \varepsilon_i$

these values are constant

```
## # A tibble: 3 x 4
##      TV radio newspaper sales
##   <dbl> <dbl>     <dbl> <dbl>
## 1 230.1  37.8      69.2  22.1
## 2  44.5  39.3      45.1  10.4
## 3  17.2  45.9      69.3   9.3
```

- The data are not tidy since each column corresponds to the values of advertising budget for different media.

3 differenty types of advertsing TV, radio, newspaper.

Amount Ad Type Sales

Tidy the Advertising Data

- Tidy the data by creating a column for advertising budeget and another column for type of advertising.
- We can use the `gather` function in the `tidyr` library (part of the `tidyverse` library) to tidy the data.

```
Advertising_long <- Advertising %>%
  select(TV, radio, newspaper, sales) %>%
  gather(key = adtype, value = amount, TV, radio, newspaper)
head(Advertising_long)
```

```
## # A tibble: 6 x 3
##   sales adtype amount
##   <dbl> <chr>   <dbl>
## 1 22.1  TV      230
## 2 10.4  TV       44.5
## 3  9.30 TV       17.2
## 4 18.5  TV      152
## 5 12.9  TV      181
## 6  7.20 TV        8.70
```

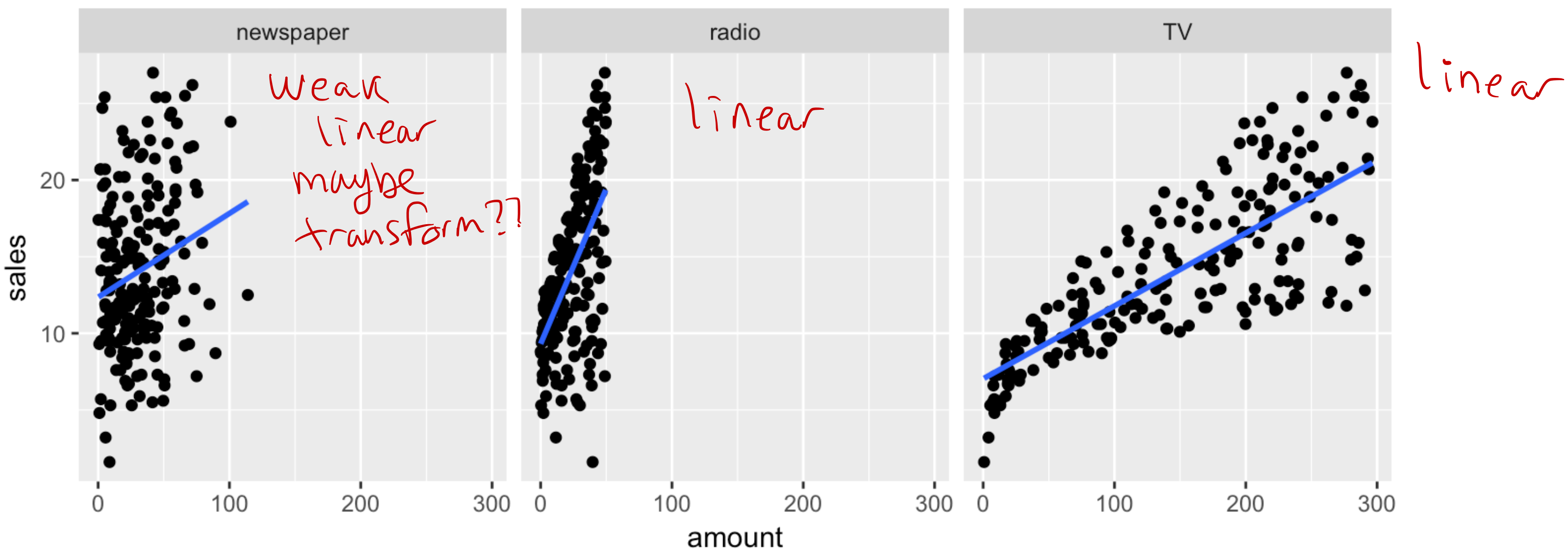
TV Radio newspaper Sales

wide format

long format data set

Advertising Data

```
Advertising_long %>%
  ggplot(aes(amount, sales)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  facet_grid(. ~ adtype)
```



- The advertising budgets (`newspaper`, `radio`, `TV`) are the input/independent/covariates and the dependent variable is sales.

Linear Regression Models

Simple Linear Regression

The simple linear regression model can describe the relationship between sales and amont spent on radio advertising through the model

amount spent on radio

Sales

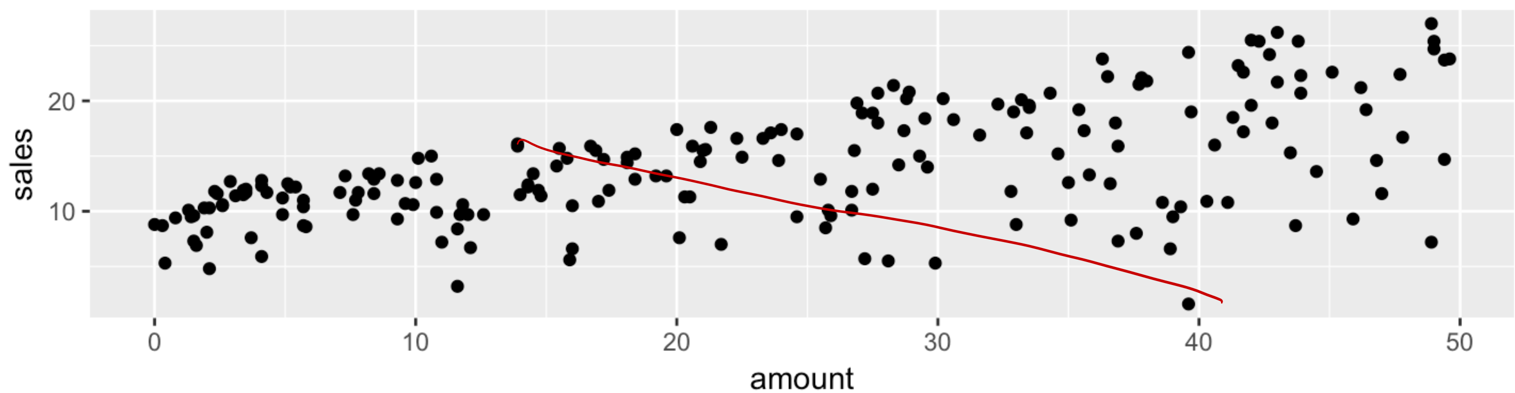
$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i,$$

200

||

where $i = 1, \ldots, n$ and n is the number of observations.

```
Advertising_long %>%
  filter(adtype == "radio") %>%
  ggplot(aes(amount, sales)) +
  geom_point()
```



Simple Linear Regression

The equation:

$$y_i = \beta_0 + \beta_1 x_i + \epsilon_i$$

is called a **regression model** and since we have only one independent variable it is called a *simple regression model*.

- y_i is called the dependent or target variable.
- β_0 is the intercept parameter.
- x_i is the independent variable, covariate, feature, or input.
- β_1 is called the slope parameter.
- ϵ_i is called the error parameter.

Statistical parameters that will be estimated from the data.

Multiple Linear Regression

In general, models of the form

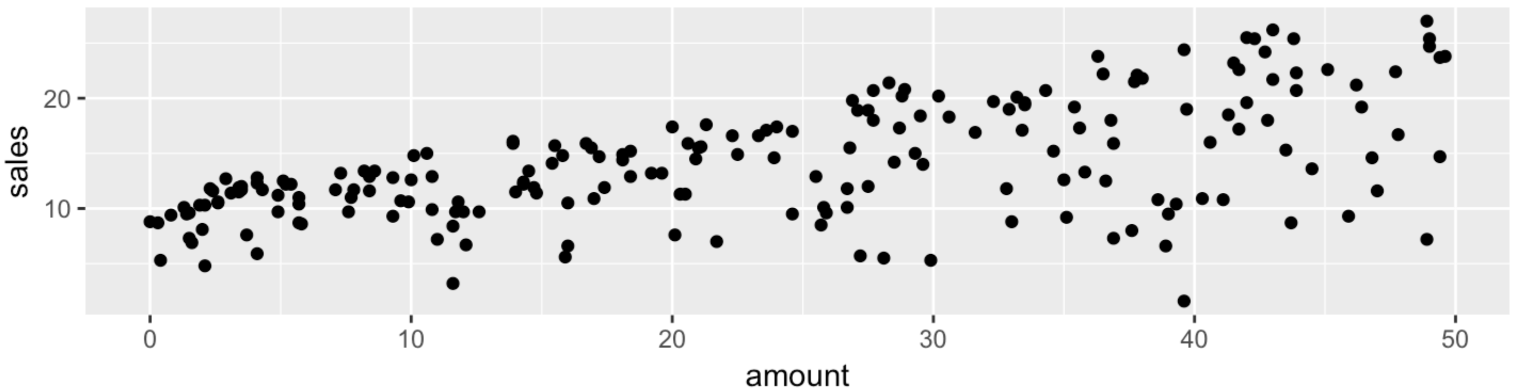
$$y_i = \beta_0 + \beta_1 x_{i1} + \beta_2 x_{i2} + \cdots + \beta_k x_{ik} + \epsilon_i,$$

where $i = 1, \ldots, n$, with $k > 1$ independent variables are called *multiple regression models*.

- The β_j's are called parameters and the ϵ_i's errors.
- The values of of neither β_j's nor ϵ_i's can ever be known, but they can be estimated.
- The "linear" in Linear Regression means that the equation is linear in the parameters β_j.
- This is a linear regression model: $y_i = \beta_0 + \beta_1 \sqrt{x_{i1}} + \beta_2 x_{i2}^2 + \epsilon_i$ Constants.
- This is not a linear regression model (i.e., a nonlinear regression model): $y_i = \beta_0 + \sin(\beta_1) x_{i1} + \beta_2 x_{i2} + \epsilon_i$

Least Squares

Fitting a straight line to Sales and Radio Advertising




```
## # A tibble: 6 x 2
##   sales amount
##   <dbl>  <dbl>
## 1  22.1   37.8
## 2  10.4   39.3
## 3   9.3   45.9
## 4  18.5   41.3
## 5  12.9   10.8
## 6   7.2   48.9
```

Fitting a straight line to Sales and Radio Advertising

```
head(Advertising_long %>%
  filter(adtype == "radio")) %>%
  select(sales,amount)
```

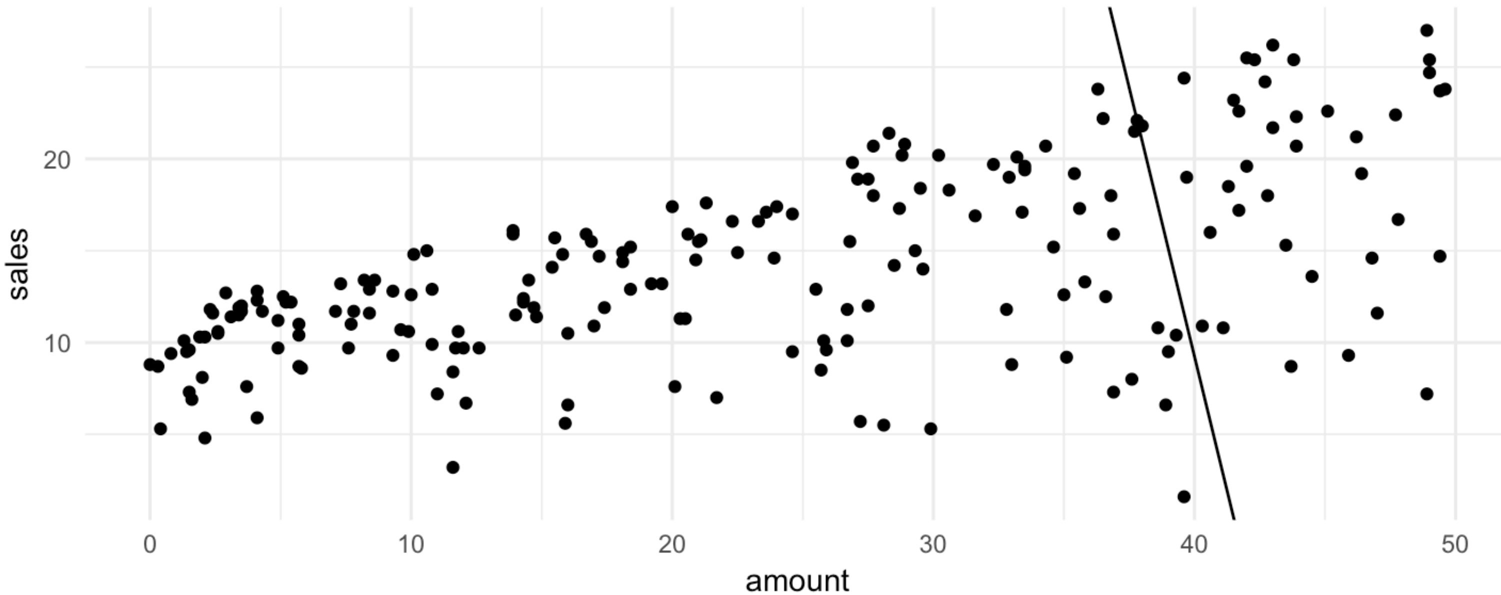
```
## # A tibble: 6 x 2
##   sales amount
##   <dbl>  <dbl>
## 1  22.1   37.8
## 2  10.4   39.3
## 3   9.3   45.9
## 4  18.5   41.3
## 5  12.9   10.8
## 6   7.2   48.9
```

$m = \frac{22.1-10.4}{37.8-39.8} =$ -5.85, $b = 22.1 - \frac{22.1-10.4}{37.8-39.8} \times 37.8 =$ 243.23. So, the equation of the straight line is:

$$y = 243.23 - 5.85x.$$

Fitting a straight line to Sales and Radio Advertising

The equation $y = 243.23 - 5.85x$ is shown on the scatter plot.

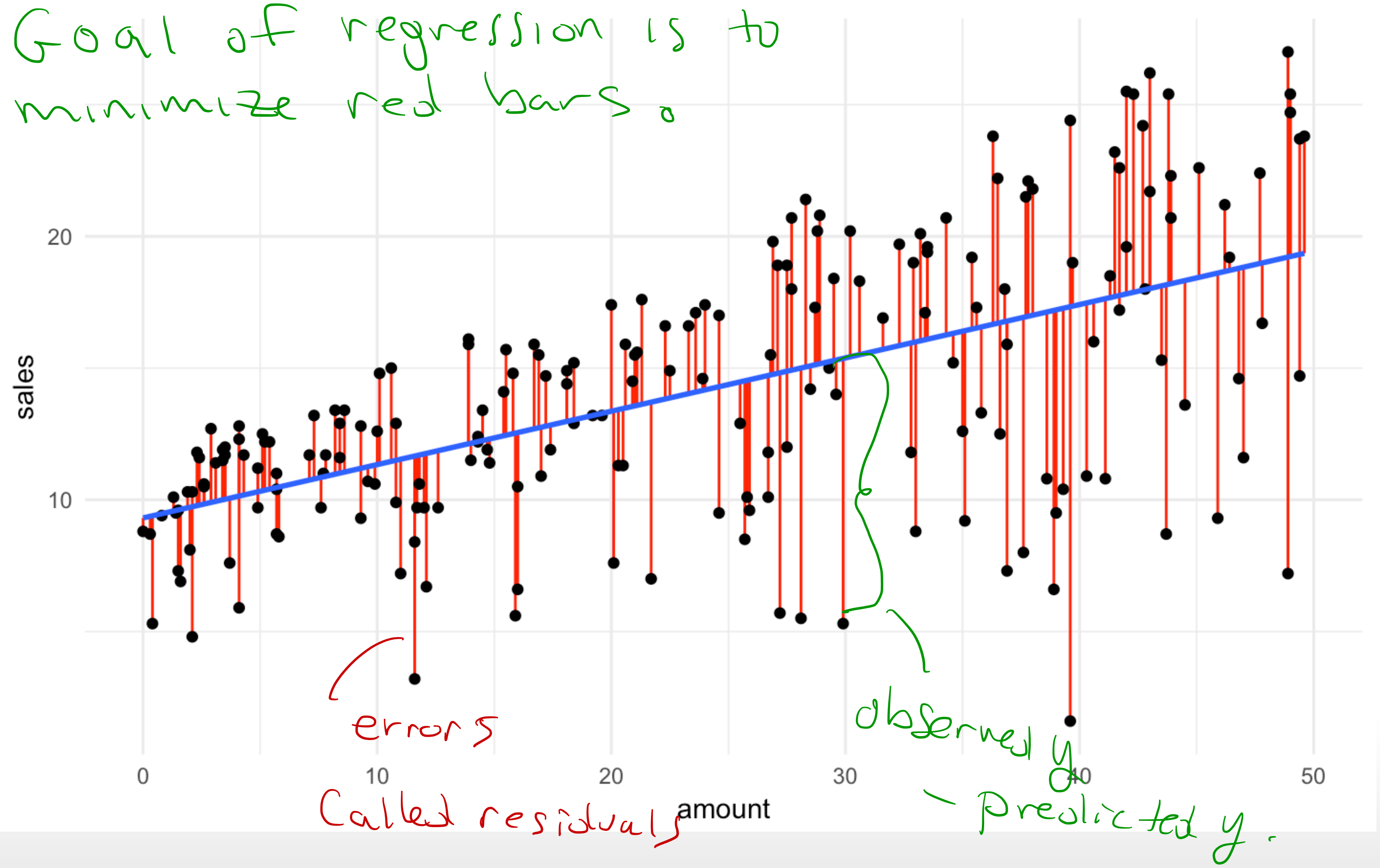


Fitting a straight line to Sales and Radio Advertising

- For a fixed value of `amount` spent on radio ads the corresponding `sales` has variation. It's neither strictly increasing nor decreasing.
- But, the overall pattern displayed in the scatterplot shows that *on average* `sales` increase as `amount` spent on radio ads increases.

Least Squares

The Least Squares approach is to find the y-intercept β_0 and slope β_1 of the straight line that is closest to as many of the points as possible.



Estimating the coefficients: Least Squares

To find the values of β_0 and slope β_1 that fit the data best we can minimize the sum of squared errors $\sum_{i=1}^{n} \epsilon_i^2$:

$$\sum_{i=1}^{n} \epsilon_i^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$$

So, we want to minimize a function of β_0, β_1

$$L(\beta_0, \beta_1) = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2,$$

where x_i's are numbers and therfore constants.

$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$

$\varepsilon_i^2 = (y_i - \beta_0 - \beta_1 x_i)^2$

$\sum_{i=1}^{n} \varepsilon_i^2 = \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i)^2$

find the values of β_0, β_1 such that $\sum_{i=1}^{n} \varepsilon_i^2$ is min.

Estimating the coefficients: Least Squares

- The derivative of $L(\beta_0, \beta_1)$ with respect to β_0 treats β_1 as a constant. This is also called the partial derivative and is denoted as $\frac{\partial L}{\partial \beta_0}$.

- To find the values of β_0 and β_1 that minimize $L(\beta_0, \beta_1)$ we set the partial derivatives to zero and solve:

$$\frac{\partial L}{\partial \beta_0} = -2 \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i) = 0, \qquad (1)$$

$$\frac{\partial L}{\partial \beta_1} = -2 \sum_{i=1}^{n} (y_i - \beta_0 - \beta_1 x_i) x_i = 0. \qquad (2)$$

The values of β_0 and β_1 that are solutions to above equations are denoted $\hat{\beta}_0$ and $\hat{\beta}_1$ respectively.

Find the values of β_0, β_1 called $\hat{\beta}_0$, $\hat{\beta}_1$ that satisfy (1) and (2).

Estimating the coefficients: Least Squares

It can be shown that

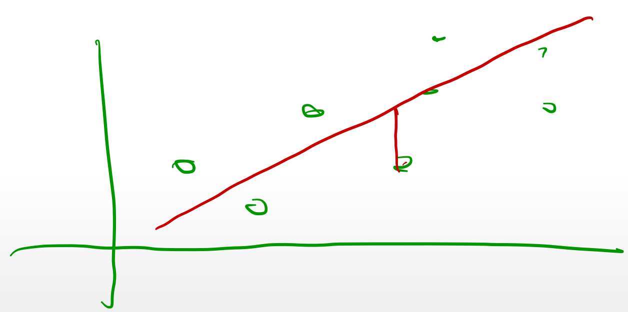
$$\hat{\beta}_0 = \bar{y} - \hat{\beta}_1 \bar{x}$$
$$\hat{\beta}_1 = \frac{\left(\sum_{i=1}^n y_i x_i\right) - n\bar{x}\bar{y}}{\left(\sum_{i=1}^n x_i^2\right) - n\bar{x}^2},$$

where, $\bar{y} = \sum_{i=1}^n y_i/n$, and $\bar{x} = \sum_{i=1}^n x_i/n$.

$\hat{\beta}_0$ and $\hat{\beta}_1$ are called the least squares estimators of β_0 and β_1.

these estimates use the data to estimate the slope and y-intercept. of "best" fitting line.

Best means that the sum of squared errors are minimized



$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i$$

$$L(\beta_0, \beta_1) = \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i\right)^2$$

find $\hat{\beta}_0$, $\hat{\beta}_1$ such that L is minimized.

$$\frac{dL}{d\beta_0} = \frac{\partial L}{\partial \beta_0} = \frac{d}{d\beta_0} \sum_{i=1}^{n} \left(y_i - \beta_0 - \beta_1 x_i\right)^2$$

$$= \sum_{i=1}^{n} \frac{d}{d\beta_0} \left(y_i - \beta_0 - \beta_1 x_i\right)^2$$

$$= \sum_{i=1}^{n} 2\left(y_i - \beta_0 - \beta_1 x_i\right)(-1)$$

Setting $\frac{\partial L}{\partial \beta_0} = 0$ $\quad \sum_{i=1}^{n} 2\left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i\right)(-1) = 0$

$$\Rightarrow \sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i\right) = 0$$

$$\sum_{i=1}^{n} \left(y_i - \hat{\beta}_0 - \hat{\beta}_1 x_i \right) = 0$$

$$\Rightarrow \sum_{i=1}^{n} y_i - \sum_{i=1}^{n} \hat{\beta}_0 - \sum_{i=1}^{n} \hat{\beta}_1 x_i = 0$$

$$\Rightarrow \sum_{i=1}^{n} y_i - n\hat{\beta}_0 - \hat{\beta}_1 \sum_{i=1}^{n} x_i = 0$$

$$\Rightarrow \sum_{i=1}^{n} y_i - \hat{\beta}_1 \sum_{i=1}^{n} x_i = n\hat{\beta}_0$$

$$\Rightarrow \frac{\sum_{i=1}^{n} y_i - \hat{\beta}_1 \sum_{i=1}^{n} x_i}{n} = \hat{\beta}_0$$

$$\Rightarrow \frac{\sum y_i}{n} - \hat{\beta}_1 \frac{\sum x_i}{n} = \hat{\beta}_0 \Rightarrow \boxed{\bar{y} - \hat{\beta}_1 \bar{x} = \hat{\beta}_0}$$

Estimating the Coefficients Using R - Formula syntax in R

The R syntax for defining relationships between inputs such as amount spent on `newspaper` advertising and outputs such as `sales` is:

```
sales ~ newspaper
```

The tilde `~` is used to define the what the output variable (or outcome, on the left-hand side) is and what the input variables (or predictors, on the right-hand side) are.

A formula that has three inputs can be written as

```
sales ~ newspaper + TV + radio
```

this notation is also used in classification trees.

Estimating the Coefficients Using `lm()`

linear model.

y_i x_i

```
mod_paper <- lm(sales ~ newspaper, data = Advertising)
mod_paper_summary <- summary(mod_paper)
mod_paper_summary$coefficients
```

```
##               Estimate Std. Error   t value     Pr(>|t|)
## (Intercept) 12.3514071 0.62142019 19.876096 4.713507e-49
## newspaper    0.0546931 0.01657572  3.299591 1.148196e-03
```

- `(Intercept)` is the estimate of $\hat{\beta}_0$.
- `newspaper` is the estimate of $\hat{\beta}_1$.

$\hat{\beta}_1$

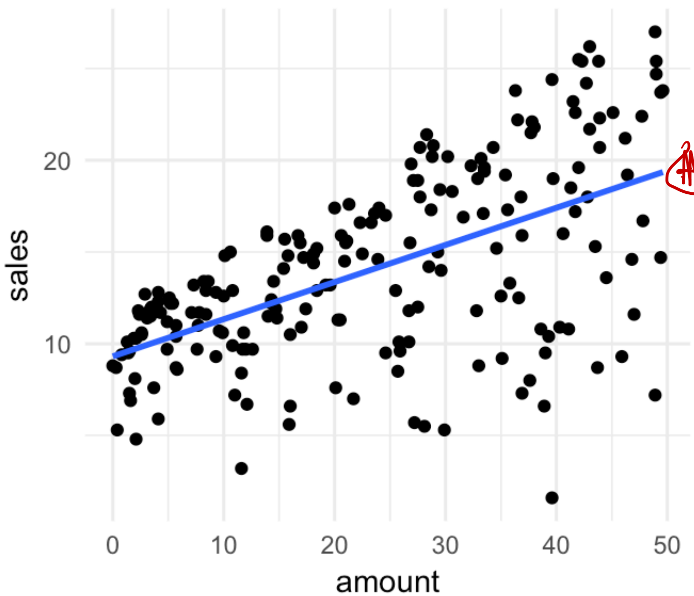
$\hat{\beta}_0$

Estimating the Coefficients Using R

```
Advertising_long %>%
  filter(adtype == "radio") %>%
  ggplot(aes(amount, sales)) +
  geom_point() +
  geom_smooth(method = "lm", se = FALSE) +
  theme_minimal()
```

adds the linear regression line to the Scatter plot

$y_i = 12.35 + 0.05 \cdot x$.



- The blue line is the estimated regression line with intercept 12.35 and slope 0.05.
- `geom_smooth(method = "lm", se = FALSE)` adds the linear regression to the

Interpreting the Slope and Intercept with a Continuous Explanatory Variable

The estimated linear regression of `sales` on `newspaper` is:

$$y_i = 12.35 + 0.05x_i,$$

measured in dollars.

where y_i is sales in the i^{th} market and x_i is the dollar amount spent on newspaper advertising in the i^{th} market.

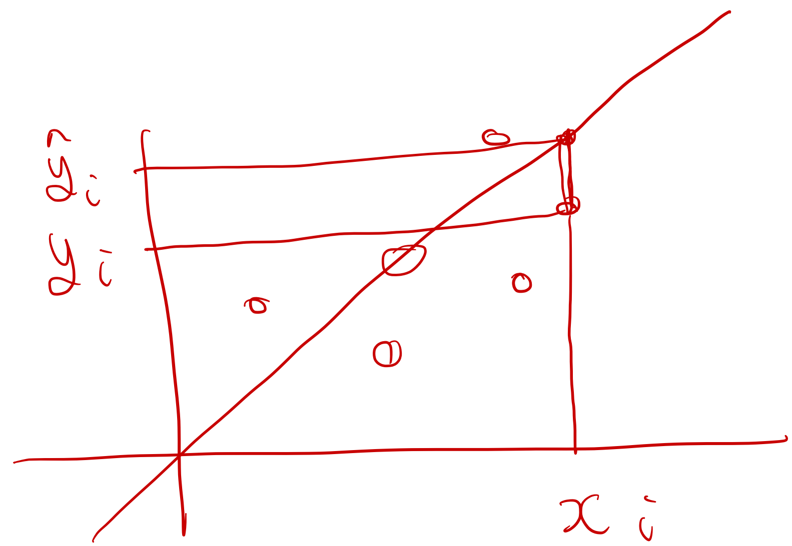
- The **slope** $\hat{\beta}_1$ is the amount of change in y for a unit change in x.
- Sales increase by 0.05 for each dollar spent on advertising.
- The **intercept** $\hat{\beta}_0$ is the average of y when $x_i = 0$.
- The average sales is 12.35 when the amount spent on advertising is zero.

wt. of new born $= \hat{\beta}_0 + \hat{\beta}_1 x$.

Prediction using a Linear Regression Model

After a linear regression model is estimated from data it can be used to calculate predicted values using the regression equation

$$\hat{y}_i = \hat{\beta}_0 + \hat{\beta}_1 x_i .$$



$\hat{y}_i$ is the predicted value of the i^{th} response y_i .

The i^{th} residual is

$$e_i = y_i - \hat{y}_i .$$

$$y_i - \hat{y}_i$$

Prediction using a Linear Regression Model

The amount spent on newspaper advertising in the first market is:

```
Advertising %>% filter(row_number() == 1)
```

```
## # A tibble: 1 x 4
##      TV radio newspaper sales
##   <dbl> <dbl>     <dbl> <dbl>
## 1 230.1  37.8      69.2  22.1
```

observed

- The predicted sales using the regression model is: $12.35 + 0.05 \times 69.2 =$ 16.14.
- The observed sales for region is 22.1.
- The **error** or **residual** is $y_1 - \hat{y}_1$ $=$ 5.96.

Prediction using a Linear Regression Model

linear regression is saved as an object.

The predicted and residual values from a regression model can be obtained using the **`predict()`** and **`residual()`** functions.

```
mod_paper <- lm(sales ~ newspaper, data = Advertising)
sales_pred <- predict(mod_paper)
head(sales_pred)
```

use predict function

```
##        1        2        3        4        5        6 
## 16.13617 14.81807 16.14164 15.55095 15.54548 16.45339
```

```
sales_resid <- residuals(mod_paper)
head(sales_resid)
```

```
##         1         2         3         4         5         6 
##  5.963831 -4.418066 -6.841639  2.949047 -2.645484 -9.253389
```

Measure of Fit for Simple Regression

- The regression model is a good fit when the residuals are small.
- Thus, we can measure the quality of fit by the sum of squares of the residuals $\sum_{i=1}^{n}(y_i - \hat{y}_i)^2$.
- This quantity depends on the units in which y_i 's are measured. A measure of fit that does not depend on the units is:

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

- R^2 is often called the coeffcient of determination.
- $0 \leq R^2 \leq 1$, where 1 indicates a perfect match between the observed and predicted values and 0 indicates an poor match.

$R^2 = 1 \iff e_i = 0 \quad \forall i.$

$R^2 = 0$ indicates poor fit.

Measure of Fit for Simple Regression

The `summary()` method calculates R^2

```
mod_paper <- lm(sales ~ newspaper, data = Advertising)
mod_paper_summ <- summary(mod_paper)
mod_paper_summ$r.squared
```

```
## [1] 0.05212045
```

- $R^2 = 0.0521204$. This indicates a poor fit.

Using Linear Regression as a Machine Learning/Supervised Learning Tool

The **`diamonds`** data set contains the prices and other attributes of almost 54,000 diamonds. The variables are as follows:

```
## Observations: 53,940
## Variables: 10
## $ carat   <dbl> 0.23, 0.21, 0.23, 0.29, 0.31, 0.24, 0.24, 0.26, 0.22, ...   x_i
## $ cut     <ord> Ideal, Premium, Good, Premium, Good, Very Good, Very G...
## $ color   <ord> E, E, E, I, J, J, I, H, E, H, J, J, F, J, E, E, I, J, ...
## $ clarity <ord> SI2, SI1, VS1, VS2, SI2, VVS2, VVS1, SI1, VS2, VS1, SI...
## $ depth   <dbl> 61.5, 59.8, 56.9, 62.4, 63.3, 62.8, 62.3, 61.9, 65.1, ...
## $ table   <dbl> 55, 61, 65, 58, 58, 57, 57, 55, 61, 61, 55, 56, 61, 54...
## $ price   <int> 326, 326, 327, 334, 335, 336, 336, 337, 337, 338, 339,...   y_i
## $ x       <dbl> 3.95, 3.89, 4.05, 4.20, 4.34, 3.94, 3.95, 4.07, 3.87, ...
## $ y       <dbl> 3.98, 3.84, 4.07, 4.23, 4.35, 3.96, 3.98, 4.11, 3.78, ...
## $ z       <dbl> 2.43, 2.31, 2.31, 2.63, 2.75, 2.48, 2.47, 2.53, 2.49, ...
```

Question: Predict the price of diamonds based on carot size.

Predicting the Price of Diamonds

Let's select training and test sets.

```
set.seed(2)
diamonds_train <- diamonds %>%
  mutate(id = row_number()) %>%
  sample_frac(size = 0.8)

diamonds_test <- diamonds %>%
  mutate(id = row_number()) %>%
  # return all rows from diamonds where there are not
  # matching values in diamonds_train, keeping just
  # columns from diamonds.
  anti_join(diamonds_train, by = 'id')
```

use 80% of data to build/train model

use 20% of data to test model.

Predicting the Price of Diamonds

- Now fit a regression model on `diamonds_train`.

```
mod_train <- lm(price ~ carat, data = diamonds_train)
mod_train_summ <- summary(mod_train)
mod_train_summ$r.squared
```

fits a linear regression on training data.

```
## [1] 0.848017
```

- Evaluate the prediction error using root mean square error using the training model on `diamonds_test`.

$$\text{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}(y_i - \hat{y}_i)^2}$$

- RMSE can be used to compare different sizes of data sets on an eual footing and the square root ensures that RMSE is on the same scale as y.

Predicting the Price of Diamonds using Simple Linear Regression

- Calculate RMSE using test and training data.

```
y_test <- diamonds_test$price
yhat_test <- predict(mod_train, newdata = diamonds_test)
n_test <- length(diamonds_test$price)

# test RMSE
rmse <- sqrt(sum((y_test - yhat_test)^2) / n_test)
rmse
```

```
## [1] 1553.208
```

test RMSE is slightly higher but very similar to training data.

```
y_train <- diamonds_train$price
yhat_train <- predict(mod_train, newdata = diamonds_train)
n_train <- length(diamonds_train$price)

# train RMSE
sqrt(sum((y_train - yhat_train)^2) / n_train)
```

```
## [1] 1547.402
```

Predicting the Price of Diamonds using Multiple Linear Regression

• multiple regression since we have used more than one indep. variable

We will add other variables to the regression model to investigate if we can decrease the prediction error.

4 indep. variables

```
mrmod_train <- lm(price ~ carat + cut + color + clarity, data = diamonds_train)
mrmod_train_summ <- summary(mrmod_train)
mrmod_train_summ$r.squared
```

```
## [1] 0.9152898
```

R^2 for multiple linear reg. ↑ Compared to simple linear regression

```
y_test <- diamonds_test$price
yhat_test <- predict(mrmod_train, newdata = diamonds_test)
n_test <- length(diamonds_test$price)
mr_rmse <- sqrt(sum((y_test - yhat_test)^2) / n_test)
mr_rmse
```

```
## [1] 1149.881
```

RMSE for multiple linear regression ↓ Compared to simple linear regression.

- The simple linear regression model had $R^2 = 0.848017$ and RMSE = 1553.2083635.

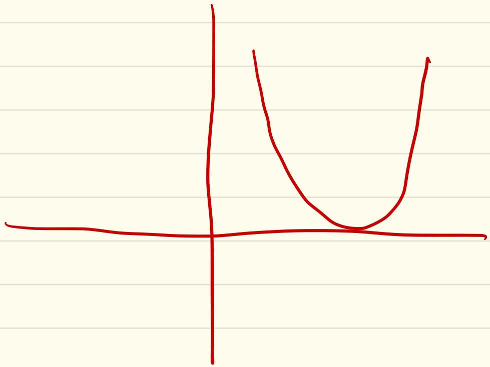
The multiple linear regression model is a better predictive model for the price of a diamond compared to the simple linear regression model with only one independent variable (carat).

$$y_i = \beta_0 + \beta_1 x_i + \varepsilon_i \quad , \quad i = 1, \dots, n$$

$$\sum_{i=1}^{n} \varepsilon_i^2 \quad \text{is minimized}$$

Why not?

$$\sum_{i=1}^{n} \varepsilon_i$$



$y = x^2$