361 PSET 3 CM

2025-02-01

1.

```r
d = readRDS('C:/Users/camer/Downloads/games.rds')

d = d %>%
  filter(lg=='nba',
         season %in% 2022,
         season.type=='reg') %>%
  select(date, away, home, ascore, hscore, season, gid)
head(d)
```

```
##         date away home ascore hscore season      gid
## 1 2021-10-19  BKN  MIL    104    127   2022 22100001
## 2 2021-10-19  GSW  LAL    121    114   2022 22100002
## 3 2021-10-20  OKC  UTA     86    107   2022 22100011
## 4 2021-10-20  SAC  POR    124    121   2022 22100013
## 5 2021-10-20  DEN  PHX    110     98   2022 22100012
## 6 2021-10-20  ORL  SAS     97    123   2022 22100010
```

```r
dg = d %>%
  group_by(away, home) %>%
  summarise(games = n()) %>%
  ungroup() %>%
  complete(away, home, fill=list(games=0)) ## new function!
```

```
## `summarise()` has grouped output by 'away'. You can override using the
## `.groups` argument.
```

```r
head(dg)
```

```
## # A tibble: 6 x 3
##   away  home  games
##   <chr> <chr> <int>
## 1 ATL   ATL       0
## 2 ATL   BKN       1
## 3 ATL   BOS       2
## 4 ATL   CHA       2
## 5 ATL   CHI       2
## 6 ATL   CLE       2
```

```r
dg$games = as.character(dg$games)

# Creating a tile plot:
g = ggplot(dg,
           aes(x = home,
               y = away,
               fill = games)) +

  geom_tile(linewidth = 0.5,
            show.legend = T,
            color = pubdarkgray) +

  scale_fill_manual(values = c(pubgradgray,
                               pubblue,
                               pubdarkblue)) +

    labs(title = "Games Between Pairs of Teams",
         x = "Home Team",
         y = "Away Team",
         fill = "# of Games")

g %>%
  pub(type = 'grid') +
  theme(axis.text.x.top = element_text(angle = 90,
                                       vjust = .5,
                                       hjust = 0),
        axis.title.x = element_text(size = 12),
        axis.title.y = element_text(size = 12))
```
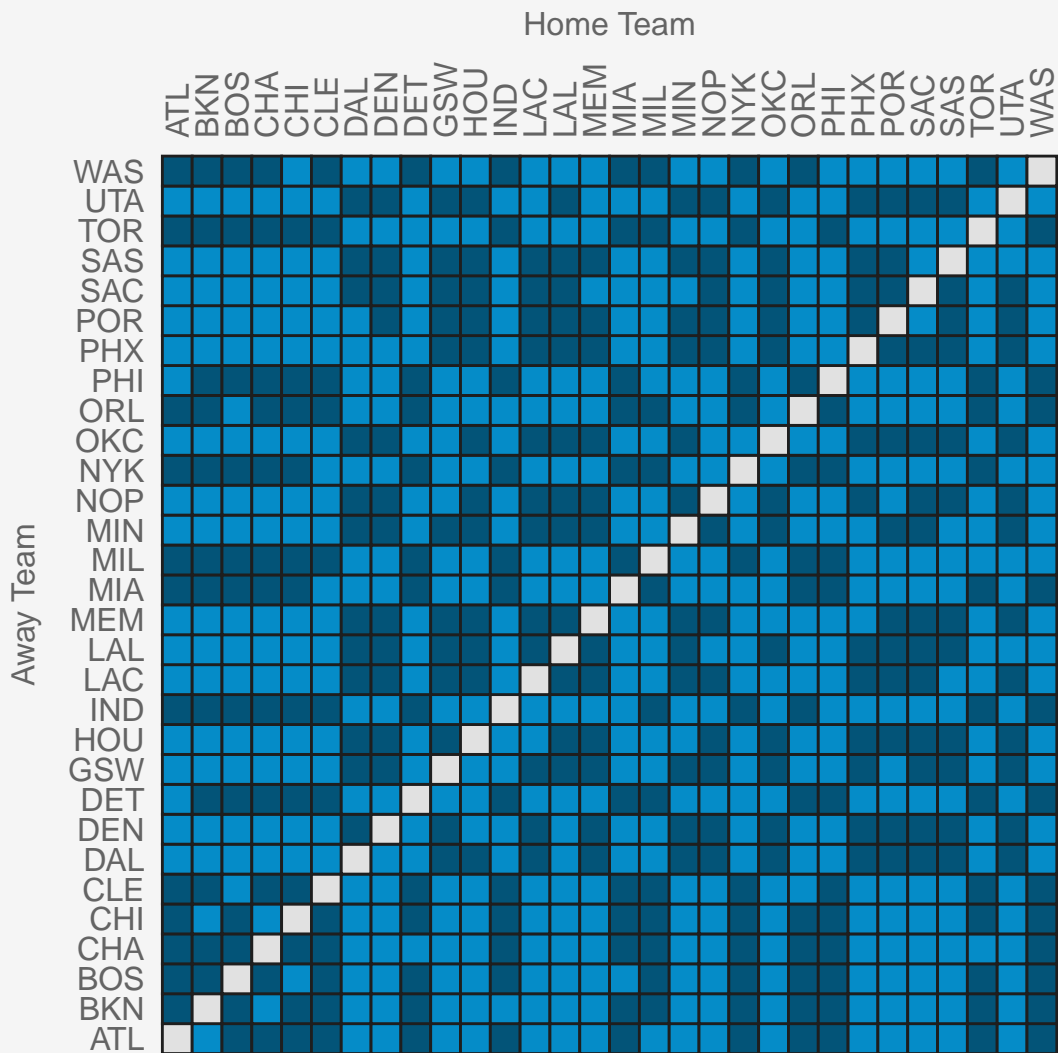
```
## [1] 85.89991
## [1] 80
## [1] 20
## [1] 185.8999
```

Games Between Pairs of Teams

of Games   ☐ 0   ■ 1   ■ 2

Home Team

2.

```r
tms = read.csv('C:/Users/camer/Downloads/nba.teams.csv')


## capitalize the first letter of each conf and div
## create a factor to specify the order we want for divisions
tms = tms %>%
  arrange(conf, div) %>%
  mutate(conf = paste0(toupper(substr(conf, 1, 1)),
```

```
                            substr(conf, 2, nchar(conf))),

         div  = paste0(toupper(substr(div , 1, 1)),
                            substr(div , 2, nchar( div))),

         div  = factor(div,
                     levels = unique(div)))
head(tms)
```

```
##   team      div conf
## 1  BKN Atlantic East
## 2  BOS Atlantic East
## 3  NYK Atlantic East
## 4  PHI Atlantic East
## 5  TOR Atlantic East
## 6  CHI  Central East
```

```
# Saving the order of teams ahead:
teams.order = tms %>%
  select(team) %>%
  unlist()
head(teams.order)
```

```
## team1 team2 team3 team4 team5 team6
## "BKN" "BOS" "NYK" "PHI" "TOR" "CHI"
```

```
# Joining both groups of data:
dg2 = dg %>%
  left_join(tms %>% select(team, div),
            by = c('home' = 'team')) %>%
  left_join(tms %>% select(team, div),
            by = c('away' = 'team'),
            suffix = c('.h', '.a'))


# We now need to change the order of the teams:
dg2 = dg2 %>%
  mutate(away = factor(away, levels = rev(teams.order)),
         home = factor(home, levels = teams.order))
head(dg2)
```

```
## # A tibble: 6 x 5
##   away  home  games div.h     div.a
##   <fct> <fct> <chr> <fct>     <fct>
## 1 ATL   ATL   0     Southeast Southeast
## 2 ATL   BKN   1     Atlantic  Southeast
## 3 ATL   BOS   2     Atlantic  Southeast
## 4 ATL   CHA   2     Southeast Southeast
## 5 ATL   CHI   2     Central   Southeast
## 6 ATL   CLE   2     Central   Southeast
```

4

```r
# Now to create the graph:
g = ggplot(dg2,
           aes(x = home,
               y = away,
               fill = games)) +
  geom_tile(linewidth = 0.5,
            show.legend = T,
            color = pubdarkgray) +
  scale_fill_manual(values = c(pubdarkgray,
                               publightblue,
                               pubblue)) +
  labs(title = "Number of Games Between All Pairs of Teams",
       x = "Home Team",
       y = "Away Team",
       fill = "# of Games")

g %>%
  pub(type = 'grid',
      base_size = 10) +
  theme(axis.text.x.top = element_text(angle = 90,
                                       vjust = 0.5))
```
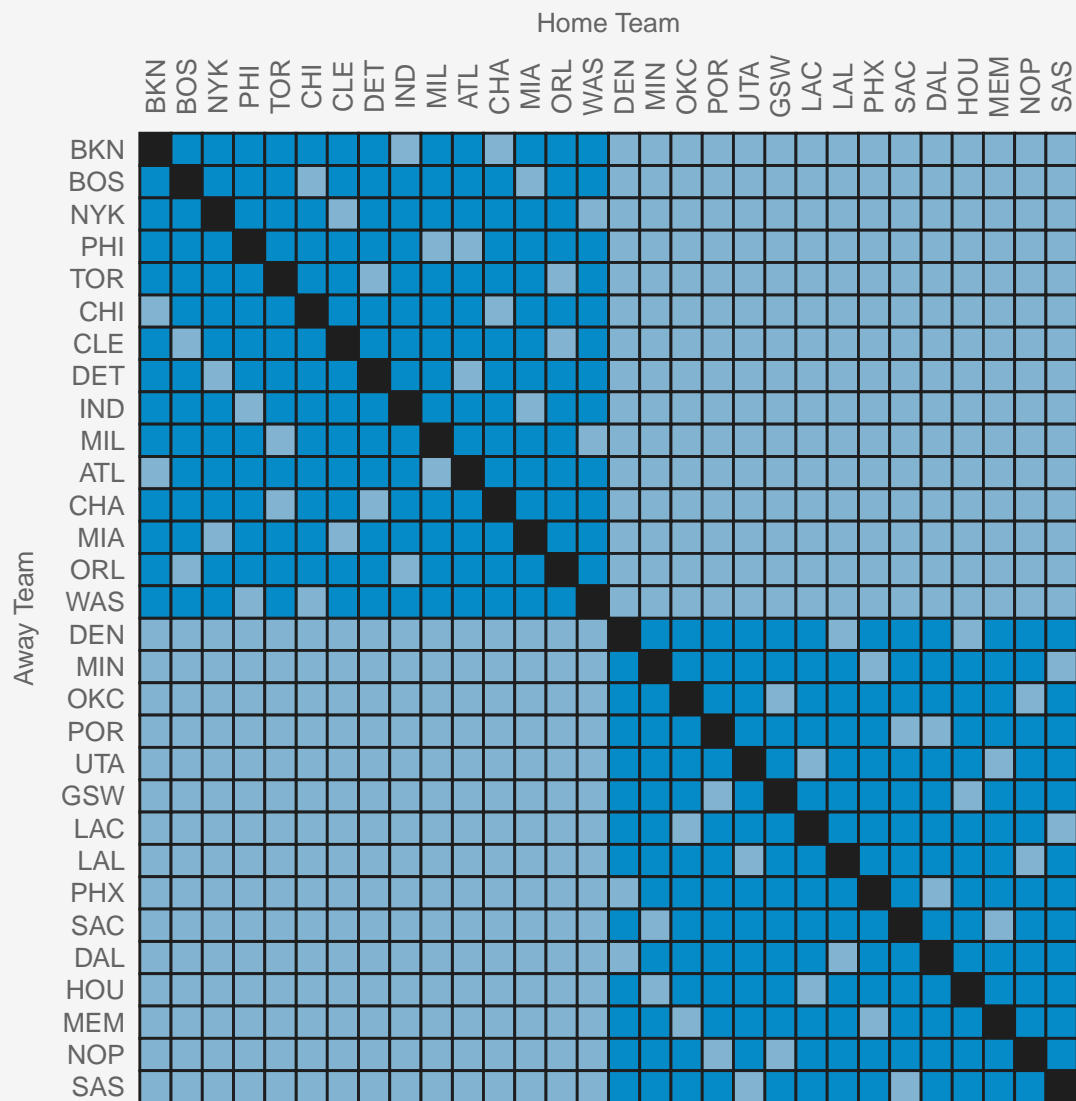
```
## [1] 85.89991
## [1] 80
## [1] 20
## [1] 185.8999
```

Number of Games Between All Pairs of Teams

of Games ■0 ☐1 ■2

3.

```
# Similar to previous plot except adding in facet grid:
g = ggplot(dg2,
           aes(x = home,
               y = away,
               fill = games)) +
  geom_tile(linewidth = 0.5,
            show.legend = T,
            color = pubdarkgray) +
  facet_grid(div.a ~ div.h,
             scales = 'free') +
  scale_fill_manual(values = c(pubdarkgray,
```

```
                          pubblue,
                          pubdarkblue)) +
  labs(title = "Number of Games Between All Pairs of Teams",
       x = "Home Team",
       y = "Away Team",
       fill = "# of Games")

g %>%
  pub(type = 'grid',
      base_size = 10) +
  theme(axis.text.x.top = element_text(angle = 90,
                                       vjust = 0.5,
                                       hjust = 0),
        legend.key.width = unit(1/8, "in"),
        panel.spacing = unit(1/24, "in"))
```
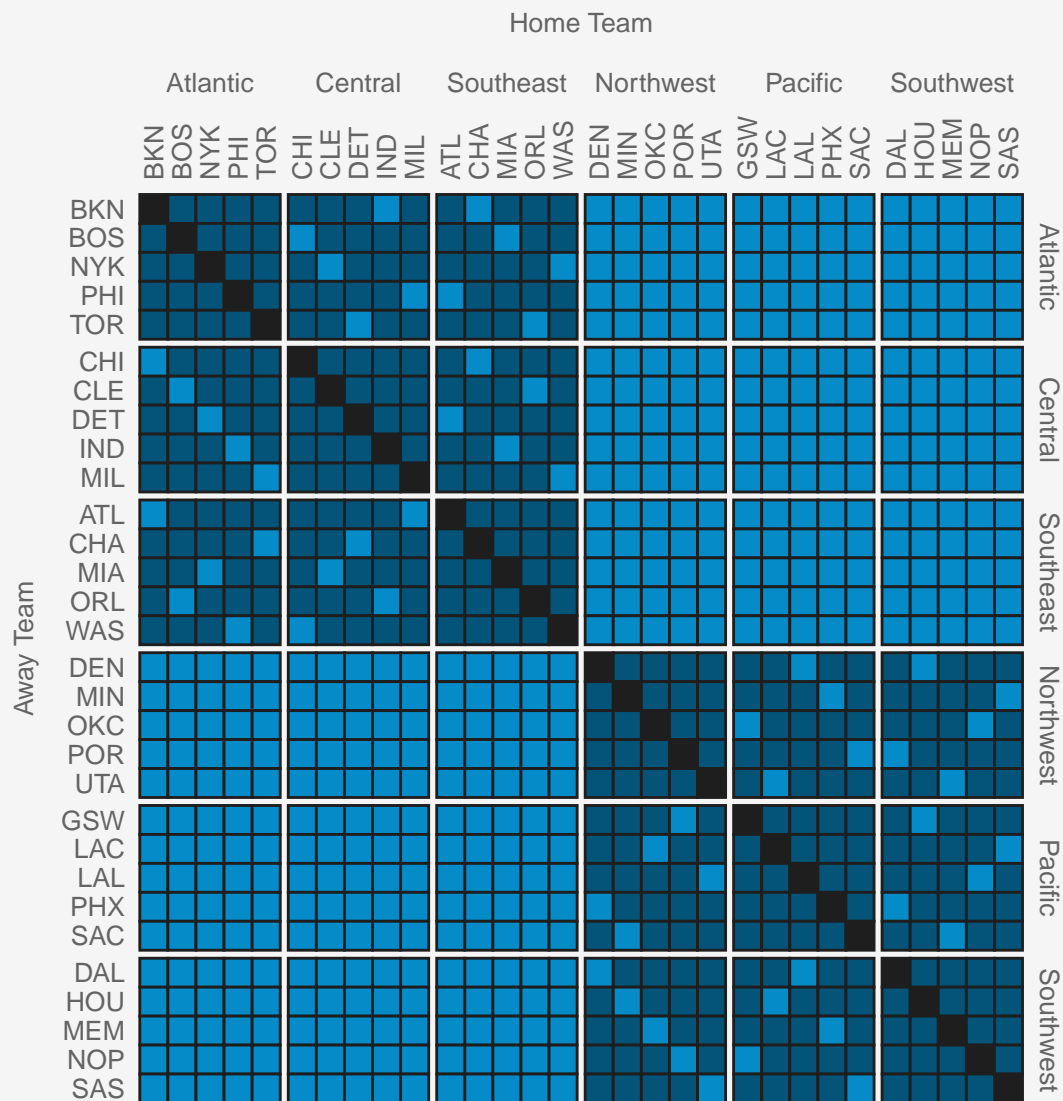
```
## [1] 75.94381
## [1] 80
## [1] 20
## [1] 175.9438
```

Number of Games Between All Pairs of Teams

4.

```r
# Screenshot of Github Repo:
```

5.

```r
library(dplyr)

# a. Random sample of 10,000 x values uniformly distributed on [0,1]:
```

```r
x = runif(10000, min = 0, max = 1)

# b. Random sample of 10,000 epsilon values from N(0,1):
epsilon = rnorm(10000, mean = 0, sd = 1)

# c. Use model y = 1 + 2x + epsilon:
y = 1 + 2*x + epsilon
df = data.frame(x, y)

# d. Fit model
lm_model = lm(y ~ x, data = df)
summary(lm_model)
```

```
##
## Call:
## lm(formula = y ~ x, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5692 -0.6804 -0.0092  0.6762  3.9483
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.00250    0.02017   49.69   <2e-16 ***
## x            2.01061    0.03487   57.66   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.001 on 9998 degrees of freedom
## Multiple R-squared:  0.2496, Adjusted R-squared:  0.2495
## F-statistic:  3325 on 1 and 9998 DF,  p-value: < 2.2e-16
```

```r
# e. Scatterplot with regression line
g = ggplot(df, aes(x, y)) +
  geom_jitter(color = pubdarkblue,
              width = 0.05,
              alpha = 0.05) +

  geom_smooth(method = "lm",
  color = pubred)
  theme_minimal() +
  labs(title = "Scatter Plot of y vs x with Regression Line",
       x = "x", y = "y")
```

```
## List of 138
##  $ line                      :List of 6
##   ..$ colour       : chr "black"
##   ..$ linewidth    : num 0.5
##   ..$ linetype     : num 1
##   ..$ lineend      : chr "butt"
##   ..$ arrow        : logi FALSE
##   ..$ inherit.blank: logi TRUE
```

```
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
## $ rect                          :List of 5
##   ..$ fill       : chr "white"
##   ..$ colour     : chr "black"
##   ..$ linewidth  : num 0.5
##   ..$ linetype   : num 1
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
## $ text                          :List of 11
##   ..$ family     : chr ""
##   ..$ face       : chr "plain"
##   ..$ colour     : chr "black"
##   ..$ size       : num 11
##   ..$ hjust      : num 0.5
##   ..$ vjust      : num 0.5
##   ..$ angle      : num 0
##   ..$ lineheight : num 0.9
##   ..$ margin     : 'margin' num [1:4] 0points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug      : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
## $ title                         : chr "Scatter Plot of y vs x with Regression Line"
## $ aspect.ratio                  : NULL
## $ axis.title                    : NULL
## $ axis.title.x                  :List of 11
##   ..$ family     : NULL
##   ..$ face       : NULL
##   ..$ colour     : NULL
##   ..$ size       : NULL
##   ..$ hjust      : NULL
##   ..$ vjust      : num 1
##   ..$ angle      : NULL
##   ..$ lineheight : NULL
##   ..$ margin     : 'margin' num [1:4] 2.75points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug      : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
## $ axis.title.x.top              :List of 11
##   ..$ family     : NULL
##   ..$ face       : NULL
##   ..$ colour     : NULL
##   ..$ size       : NULL
##   ..$ hjust      : NULL
##   ..$ vjust      : num 0
##   ..$ angle      : NULL
##   ..$ lineheight : NULL
##   ..$ margin     : 'margin' num [1:4] 0points 0points 2.75points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug      : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
## $ axis.title.x.bottom           : NULL
```

```
##  $ axis.title.y                        :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : NULL
##   ..$ hjust       : NULL
##   ..$ vjust       : num 1
##   ..$ angle       : num 90
##   ..$ lineheight  : NULL
##   ..$ margin      : 'margin' num [1:4] 0points 2.75points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y.left               : NULL
##  $ axis.title.y.right              :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : NULL
##   ..$ hjust       : NULL
##   ..$ vjust       : num 1
##   ..$ angle       : num -90
##   ..$ lineheight  : NULL
##   ..$ margin      : 'margin' num [1:4] 0points 0points 0points 2.75points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text                       :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : chr "grey30"
##   ..$ size        : 'rel' num 0.8
##   ..$ hjust       : NULL
##   ..$ vjust       : NULL
##   ..$ angle       : NULL
##   ..$ lineheight  : NULL
##   ..$ margin      : NULL
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x                     :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : NULL
##   ..$ hjust       : NULL
##   ..$ vjust       : num 1
##   ..$ angle       : NULL
##   ..$ lineheight  : NULL
##   ..$ margin      : 'margin' num [1:4] 2.2points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug       : NULL
```

```
##    ..$ inherit.blank: logi TRUE
##    ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.top              :List of 11
##    ..$ family     : NULL
##    ..$ face       : NULL
##    ..$ colour     : NULL
##    ..$ size       : NULL
##    ..$ hjust      : NULL
##    ..$ vjust      : num 0
##    ..$ angle      : NULL
##    ..$ lineheight : NULL
##    ..$ margin     : 'margin' num [1:4] 0points 0points 2.2points 0points
##    .. ..- attr(*, "unit")= int 8
##    ..$ debug      : NULL
##    ..$ inherit.blank: logi TRUE
##    ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.bottom           : NULL
##  $ axis.text.y                  :List of 11
##    ..$ family     : NULL
##    ..$ face       : NULL
##    ..$ colour     : NULL
##    ..$ size       : NULL
##    ..$ hjust      : num 1
##    ..$ vjust      : NULL
##    ..$ angle      : NULL
##    ..$ lineheight : NULL
##    ..$ margin     : 'margin' num [1:4] 0points 2.2points 0points 0points
##    .. ..- attr(*, "unit")= int 8
##    ..$ debug      : NULL
##    ..$ inherit.blank: logi TRUE
##    ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y.left             : NULL
##  $ axis.text.y.right            :List of 11
##    ..$ family     : NULL
##    ..$ face       : NULL
##    ..$ colour     : NULL
##    ..$ size       : NULL
##    ..$ hjust      : num 0
##    ..$ vjust      : NULL
##    ..$ angle      : NULL
##    ..$ lineheight : NULL
##    ..$ margin     : 'margin' num [1:4] 0points 0points 0points 2.2points
##    .. ..- attr(*, "unit")= int 8
##    ..$ debug      : NULL
##    ..$ inherit.blank: logi TRUE
##    ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.theta              : NULL
##  $ axis.text.r                  :List of 11
##    ..$ family     : NULL
##    ..$ face       : NULL
##    ..$ colour     : NULL
##    ..$ size       : NULL
##    ..$ hjust      : num 0.5
##    ..$ vjust      : NULL
```

```
##   ..$ angle       : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.2points 0points 2.2points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
## $ axis.ticks                   : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
## $ axis.ticks.x                 : NULL
## $ axis.ticks.x.top             : NULL
## $ axis.ticks.x.bottom          : NULL
## $ axis.ticks.y                 : NULL
## $ axis.ticks.y.left            : NULL
## $ axis.ticks.y.right           : NULL
## $ axis.ticks.theta             : NULL
## $ axis.ticks.r                 : NULL
## $ axis.minor.ticks.x.top       : NULL
## $ axis.minor.ticks.x.bottom    : NULL
## $ axis.minor.ticks.y.left      : NULL
## $ axis.minor.ticks.y.right     : NULL
## $ axis.minor.ticks.theta       : NULL
## $ axis.minor.ticks.r           : NULL
## $ axis.ticks.length            : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
## $ axis.ticks.length.x          : NULL
## $ axis.ticks.length.x.top      : NULL
## $ axis.ticks.length.x.bottom   : NULL
## $ axis.ticks.length.y          : NULL
## $ axis.ticks.length.y.left     : NULL
## $ axis.ticks.length.y.right    : NULL
## $ axis.ticks.length.theta      : NULL
## $ axis.ticks.length.r          : NULL
## $ axis.minor.ticks.length      : 'rel' num 0.75
## $ axis.minor.ticks.length.x    : NULL
## $ axis.minor.ticks.length.x.top   : NULL
## $ axis.minor.ticks.length.x.bottom: NULL
## $ axis.minor.ticks.length.y       : NULL
## $ axis.minor.ticks.length.y.left  : NULL
## $ axis.minor.ticks.length.y.right : NULL
## $ axis.minor.ticks.length.theta   : NULL
## $ axis.minor.ticks.length.r       : NULL
## $ axis.line                    : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
## $ axis.line.x                  : NULL
## $ axis.line.x.top              : NULL
## $ axis.line.x.bottom           : NULL
## $ axis.line.y                  : NULL
## $ axis.line.y.left             : NULL
## $ axis.line.y.right            : NULL
## $ axis.line.theta              : NULL
## $ axis.line.r                  : NULL
## $ legend.background            : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
```

```
## $ legend.margin                     : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##  ..- attr(*, "unit")= int 8
## $ legend.spacing                    : 'simpleUnit' num 11points
##  ..- attr(*, "unit")= int 8
## $ legend.spacing.x                  : NULL
## $ legend.spacing.y                  : NULL
## $ legend.key                        : list()
##  ..- attr(*, "class")= chr [1:2] "element_blank" "element"
## $ legend.key.size                   : 'simpleUnit' num 1.2lines
##  ..- attr(*, "unit")= int 3
## $ legend.key.height                 : NULL
## $ legend.key.width                  : NULL
## $ legend.key.spacing                : 'simpleUnit' num 5.5points
##  ..- attr(*, "unit")= int 8
## $ legend.key.spacing.x              : NULL
## $ legend.key.spacing.y              : NULL
## $ legend.frame                      : NULL
## $ legend.ticks                      : NULL
## $ legend.ticks.length               : 'rel' num 0.2
## $ legend.axis.line                  : NULL
## $ legend.text                       :List of 11
##  ..$ family       : NULL
##  ..$ face         : NULL
##  ..$ colour       : NULL
##  ..$ size         : 'rel' num 0.8
##  ..$ hjust        : NULL
##  ..$ vjust        : NULL
##  ..$ angle        : NULL
##  ..$ lineheight   : NULL
##  ..$ margin       : NULL
##  ..$ debug        : NULL
##  ..$ inherit.blank: logi TRUE
##  ..- attr(*, "class")= chr [1:2] "element_text" "element"
## $ legend.text.position              : NULL
## $ legend.title                      :List of 11
##  ..$ family       : NULL
##  ..$ face         : NULL
##  ..$ colour       : NULL
##  ..$ size         : NULL
##  ..$ hjust        : num 0
##  ..$ vjust        : NULL
##  ..$ angle        : NULL
##  ..$ lineheight   : NULL
##  ..$ margin       : NULL
##  ..$ debug        : NULL
##  ..$ inherit.blank: logi TRUE
##  ..- attr(*, "class")= chr [1:2] "element_text" "element"
## $ legend.title.position             : NULL
## $ legend.position                   : chr "right"
## $ legend.position.inside            : NULL
## $ legend.direction                  : NULL
## $ legend.byrow                      : NULL
## $ legend.justification              : chr "center"
## $ legend.justification.top          : NULL
```

```
##  $ legend.justification.bottom    : NULL
##  $ legend.justification.left      : NULL
##  $ legend.justification.right     : NULL
##  $ legend.justification.inside    : NULL
##  $ legend.location                : NULL
##  $ legend.box                     : NULL
##  $ legend.box.just                : NULL
##  $ legend.box.margin              : 'margin' num [1:4] 0cm 0cm 0cm 0cm
##   ..- attr(*, "unit")= int 1
##  $ legend.box.background          : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.box.spacing             : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
##   [list output truncated]
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi TRUE
##  - attr(*, "validate")= logi TRUE
```

```r
g %>% pub()
```

```
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
```

```
## [1] 49.99122
## [1] 80
## [1] 20
## [1] 149.9912
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```
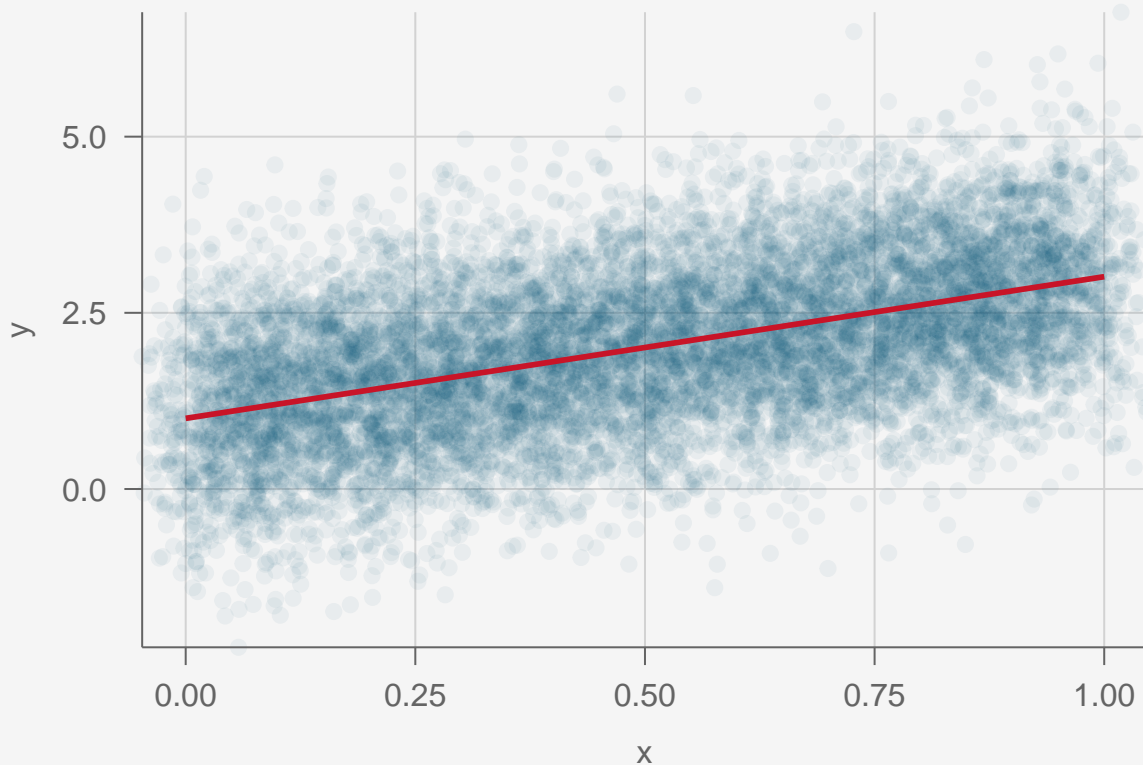
```
# The plot is as expected with an intercept of 1 and a slope of 2.
# The visualization looks similar to what I would expect with a clear linear relationship.
# The linear regression assumptions hold:
# - Linearity of y's and x's
# - Normality of Ei's: clearly show a normal distribution of (0,1)
# - Homoscedasticity of Ei's: constnat variance
# - Independence of Ei's: no dependence shown
```

6.

```
# a. Random sample of 10,000 x values uniformly distributed on [0,1]:
x2 = rbinom(n = 10000, size = 1, prob = 0.5)

# b. Random sample of 10,000 epsilon values from N(0,1):
epsilon2 = rnorm(10000, mean = 0, sd = 1)

# c. Use model y = 1 + 2x + epsilon:
y2 = 1 + 2*x2 + epsilon
df2 = data.frame(x2, y2)

# d. Fit SLR model
lm_model <- lm(y2 ~ x2)
summary(lm_model)
```

```
##
```

```
## Call:
## lm(formula = y2 ~ x2)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -3.5598 -0.6798 -0.0088  0.6744  3.9427
##
## Coefficients:
##             Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.01619    0.01407   72.23   <2e-16 ***
## x2           1.98311    0.02001   99.10   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1.001 on 9998 degrees of freedom
## Multiple R-squared:  0.4955, Adjusted R-squared:  0.4955
## F-statistic:  9820 on 1 and 9998 DF,  p-value: < 2.2e-16
```

```r
# e. Scatterplot with regression line
g = ggplot(df2, aes(x2, y2)) +
  geom_jitter(color = pubdarkblue,
              width = 0.05,
              alpha = 0.05) +

  geom_smooth(method = "lm",
  color = pubred)
  theme_minimal() +
  labs(title = "Scatter Plot of y vs x with Regression Line",
      x = "x", y = "y")
```

```
## List of 138
##  $ line                       :List of 6
##   ..$ colour      : chr "black"
##   ..$ linewidth   : num 0.5
##   ..$ linetype    : num 1
##   ..$ lineend     : chr "butt"
##   ..$ arrow       : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_line" "element"
##  $ rect                       :List of 5
##   ..$ fill        : chr "white"
##   ..$ colour      : chr "black"
##   ..$ linewidth   : num 0.5
##   ..$ linetype    : num 1
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_rect" "element"
##  $ text                       :List of 11
##   ..$ family      : chr ""
##   ..$ face        : chr "plain"
##   ..$ colour      : chr "black"
##   ..$ size        : num 11
##   ..$ hjust       : num 0.5
##   ..$ vjust       : num 0.5
##   ..$ angle       : num 0
```

```
##   ..$ lineheight   : num 0.9
##   ..$ margin       : 'margin' num [1:4] 0points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : logi FALSE
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ title                        : chr "Scatter Plot of y vs x with Regression Line"
##  $ aspect.ratio                 : NULL
##  $ axis.title                   : NULL
##  $ axis.title.x                 :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 2.75points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.top             :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 0
##   ..$ angle        : NULL
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 0points 2.75points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.x.bottom          : NULL
##  $ axis.title.y                 :List of 11
##   ..$ family       : NULL
##   ..$ face         : NULL
##   ..$ colour       : NULL
##   ..$ size         : NULL
##   ..$ hjust        : NULL
##   ..$ vjust        : num 1
##   ..$ angle        : num 90
##   ..$ lineheight   : NULL
##   ..$ margin       : 'margin' num [1:4] 0points 2.75points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug        : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.title.y.left            : NULL
##  $ axis.title.y.right           :List of 11
```

```
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : NULL
##   ..$ hjust       : NULL
##   ..$ vjust       : num 1
##   ..$ angle       : num -90
##   ..$ lineheight  : NULL
##   ..$ margin      : 'margin' num [1:4] 0points 0points 0points 2.75points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text                    :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : chr "grey30"
##   ..$ size        : 'rel' num 0.8
##   ..$ hjust       : NULL
##   ..$ vjust       : NULL
##   ..$ angle       : NULL
##   ..$ lineheight  : NULL
##   ..$ margin      : NULL
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x                  :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : NULL
##   ..$ hjust       : NULL
##   ..$ vjust       : num 1
##   ..$ angle       : NULL
##   ..$ lineheight  : NULL
##   ..$ margin      : 'margin' num [1:4] 2.2points 0points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.x.top              :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : NULL
##   ..$ hjust       : NULL
##   ..$ vjust       : num 0
##   ..$ angle       : NULL
##   ..$ lineheight  : NULL
##   ..$ margin      : 'margin' num [1:4] 0points 0points 2.2points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
```

```
##  $ axis.text.x.bottom                : NULL
##  $ axis.text.y                       :List of 11
##   ..$ family        : NULL
##   ..$ face          : NULL
##   ..$ colour        : NULL
##   ..$ size          : NULL
##   ..$ hjust         : num 1
##   ..$ vjust         : NULL
##   ..$ angle         : NULL
##   ..$ lineheight    : NULL
##   ..$ margin        : 'margin' num [1:4] 0points 2.2points 0points 0points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug         : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.y.left                  : NULL
##  $ axis.text.y.right                 :List of 11
##   ..$ family        : NULL
##   ..$ face          : NULL
##   ..$ colour        : NULL
##   ..$ size          : NULL
##   ..$ hjust         : num 0
##   ..$ vjust         : NULL
##   ..$ angle         : NULL
##   ..$ lineheight    : NULL
##   ..$ margin        : 'margin' num [1:4] 0points 0points 0points 2.2points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug         : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.text.theta                   : NULL
##  $ axis.text.r                       :List of 11
##   ..$ family        : NULL
##   ..$ face          : NULL
##   ..$ colour        : NULL
##   ..$ size          : NULL
##   ..$ hjust         : num 0.5
##   ..$ vjust         : NULL
##   ..$ angle         : NULL
##   ..$ lineheight    : NULL
##   ..$ margin        : 'margin' num [1:4] 0points 2.2points 0points 2.2points
##   .. ..- attr(*, "unit")= int 8
##   ..$ debug         : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ axis.ticks                        : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ axis.ticks.x                      : NULL
##  $ axis.ticks.x.top                  : NULL
##  $ axis.ticks.x.bottom               : NULL
##  $ axis.ticks.y                      : NULL
##  $ axis.ticks.y.left                 : NULL
##  $ axis.ticks.y.right                : NULL
##  $ axis.ticks.theta                  : NULL
```

```
## $ axis.ticks.r                   : NULL
## $ axis.minor.ticks.x.top         : NULL
## $ axis.minor.ticks.x.bottom      : NULL
## $ axis.minor.ticks.y.left        : NULL
## $ axis.minor.ticks.y.right       : NULL
## $ axis.minor.ticks.theta         : NULL
## $ axis.minor.ticks.r             : NULL
## $ axis.ticks.length              : 'simpleUnit' num 2.75points
##   ..- attr(*, "unit")= int 8
## $ axis.ticks.length.x            : NULL
## $ axis.ticks.length.x.top        : NULL
## $ axis.ticks.length.x.bottom     : NULL
## $ axis.ticks.length.y            : NULL
## $ axis.ticks.length.y.left       : NULL
## $ axis.ticks.length.y.right      : NULL
## $ axis.ticks.length.theta        : NULL
## $ axis.ticks.length.r            : NULL
## $ axis.minor.ticks.length        : 'rel' num 0.75
## $ axis.minor.ticks.length.x      : NULL
## $ axis.minor.ticks.length.x.top  : NULL
## $ axis.minor.ticks.length.x.bottom: NULL
## $ axis.minor.ticks.length.y      : NULL
## $ axis.minor.ticks.length.y.left : NULL
## $ axis.minor.ticks.length.y.right : NULL
## $ axis.minor.ticks.length.theta  : NULL
## $ axis.minor.ticks.length.r      : NULL
## $ axis.line                      : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
## $ axis.line.x                    : NULL
## $ axis.line.x.top                : NULL
## $ axis.line.x.bottom             : NULL
## $ axis.line.y                    : NULL
## $ axis.line.y.left               : NULL
## $ axis.line.y.right              : NULL
## $ axis.line.theta                : NULL
## $ axis.line.r                    : NULL
## $ legend.background              : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
## $ legend.margin                  : 'margin' num [1:4] 5.5points 5.5points 5.5points 5.5points
##   ..- attr(*, "unit")= int 8
## $ legend.spacing                 : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
## $ legend.spacing.x               : NULL
## $ legend.spacing.y               : NULL
## $ legend.key                     : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
## $ legend.key.size                : 'simpleUnit' num 1.2lines
##   ..- attr(*, "unit")= int 3
## $ legend.key.height              : NULL
## $ legend.key.width               : NULL
## $ legend.key.spacing             : 'simpleUnit' num 5.5points
##   ..- attr(*, "unit")= int 8
## $ legend.key.spacing.x           : NULL
## $ legend.key.spacing.y           : NULL
```

```
##  $ legend.frame                    : NULL
##  $ legend.ticks                    : NULL
##  $ legend.ticks.length             : 'rel' num 0.2
##  $ legend.axis.line                : NULL
##  $ legend.text                     :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : 'rel' num 0.8
##   ..$ hjust       : NULL
##   ..$ vjust       : NULL
##   ..$ angle       : NULL
##   ..$ lineheight  : NULL
##   ..$ margin      : NULL
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.text.position            : NULL
##  $ legend.title                    :List of 11
##   ..$ family      : NULL
##   ..$ face        : NULL
##   ..$ colour      : NULL
##   ..$ size        : NULL
##   ..$ hjust       : num 0
##   ..$ vjust       : NULL
##   ..$ angle       : NULL
##   ..$ lineheight  : NULL
##   ..$ margin      : NULL
##   ..$ debug       : NULL
##   ..$ inherit.blank: logi TRUE
##   ..- attr(*, "class")= chr [1:2] "element_text" "element"
##  $ legend.title.position           : NULL
##  $ legend.position                 : chr "right"
##  $ legend.position.inside          : NULL
##  $ legend.direction                : NULL
##  $ legend.byrow                    : NULL
##  $ legend.justification            : chr "center"
##  $ legend.justification.top        : NULL
##  $ legend.justification.bottom     : NULL
##  $ legend.justification.left       : NULL
##  $ legend.justification.right      : NULL
##  $ legend.justification.inside     : NULL
##  $ legend.location                 : NULL
##  $ legend.box                      : NULL
##  $ legend.box.just                 : NULL
##  $ legend.box.margin               : 'margin' num [1:4] 0cm 0cm 0cm 0cm
##   ..- attr(*, "unit")= int 1
##  $ legend.box.background           : list()
##   ..- attr(*, "class")= chr [1:2] "element_blank" "element"
##  $ legend.box.spacing              : 'simpleUnit' num 11points
##   ..- attr(*, "unit")= int 8
##   [list output truncated]
##  - attr(*, "class")= chr [1:2] "theme" "gg"
##  - attr(*, "complete")= logi TRUE
```

```
##  - attr(*, "validate")= logi TRUE
```

```
g %>% pub()
```

```
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
## 'geom_smooth()' using formula = 'y ~ x'
```
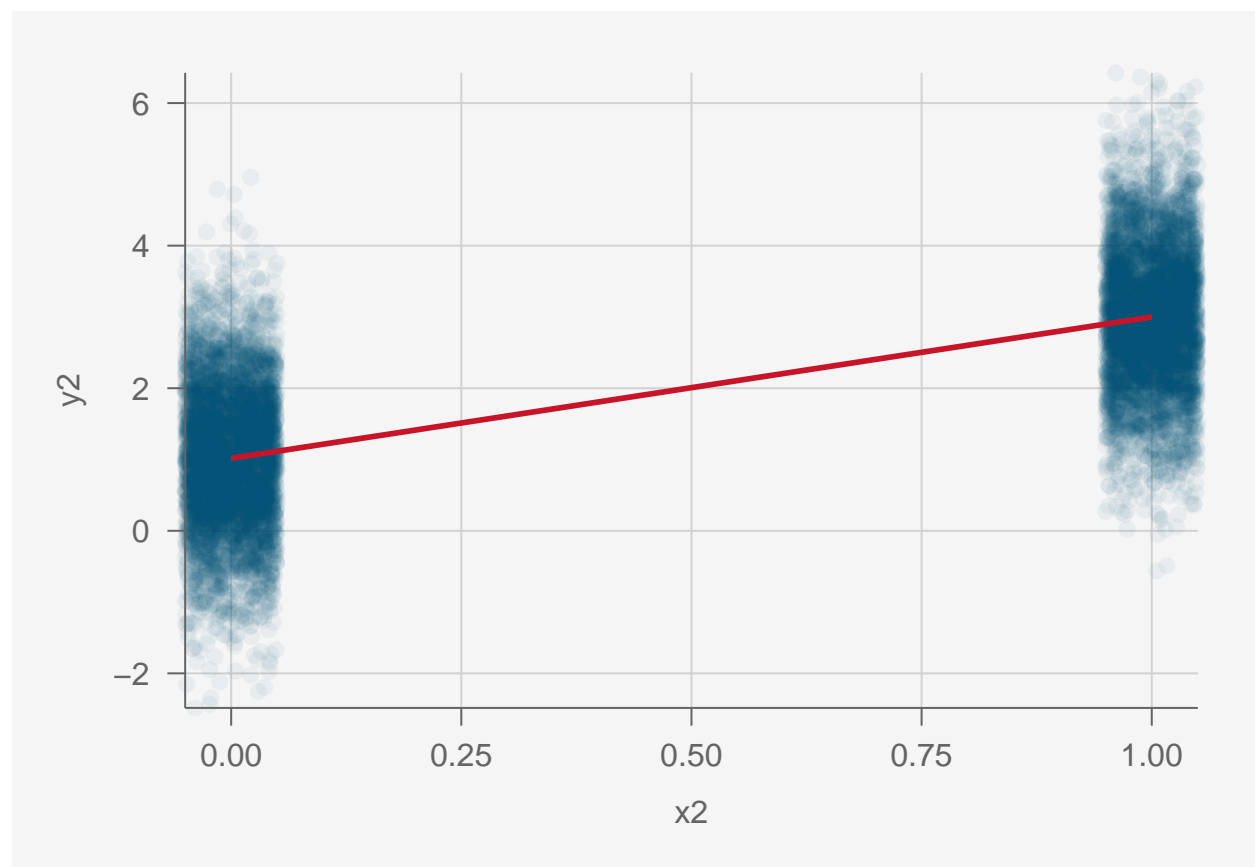
```
## [1] 31.97542
## [1] 80
## [1] 20
## [1] 131.9754
```

```
## 'geom_smooth()' using formula = 'y ~ x'
```



```
# The plot is as expected with two clusters of data at x=0 and x=1.
# This is because the bernoulli distribution only takes two outcomes.
# The visualization looks similar to what I would expect with two distinct means.
# The linear regression assumptions hold:
# - Linearity of y's and x's: linear relationship with two means
```

23

```r
# - Normality of Ei's: clearly show a normal distribution of (0,1) at each cluster
# - Homoscedasticity of Ei's: variance looks consistent between two groups
# - Independence of Ei's: no dependence shown
# The simulation assumed these, so this makes sense.
```

7.

1. $\mathbf{E(y|x) = E(B0 + B1x + E(e) = B0 + B1x*}$

2. $\mathbf{V(y|x) = V(B0 + B1x + E) = V(E) = sigma\char`^2}$

3. **Y is normally distributed. This is because it is constant and E is normal.**

For any given x, $y\sim N(B0+B1x,$ sigma$\char`^2)$

8.

```r
# T / F:
# 1. FALSE. The Ei's (residuals) must be normally distributed but not Y. Y should only be normally dist
#
# 2. FALSE. Confidence interval is always narrower than the Prediction Interval because the Prediction
#
# 3. TRUE. Bj is approximately normal for a large n due to the CLT. Additionally, it has a
# t-distribution, which is always normal for large n number of observations.
#
# 4. FALSE. These fitted values are associated withb different values. Mu should be
# approximately normal for a given x.
#
# 5. TRUE. PIs wider than CIs. This is because there is more uncertainty due to a
# combination of the uncertainty in mu and sigma^2.
```

8.

```r
df = readRDS('C:/Users/camer/Downloads/Connecticut_State_Parcel_Layer_2023_-3653519729566038598.rds')
head(df,2)
```

```
##   OBJECTID Town.Name                   Link                       Owner
## 1        1  BROOKLYN     09190-CT-019-16-85-4          HAYNES ALYSSA & SEAN
## 2        2   POMFRET 61030-CT-112-18-D-003.00 CONNECTICUT LABORERS TRAINING
##     Co.Owner        Location Mailing.Address Mailing.City Mailing.State
## 1               135  TATNIC RD   135 TATNIC RD     BROOKLYN            CT
## 2 ACADEMY INC 37 DEERFIELD RD       PO BOX 77  POMFRET CTR            CT
##   Assessed.Total Assessed.Land Assessed.Building Pre.Year.Assessed.Total
## 1         157700         34500            123200                  157700
## 2        6575000        506100           5962900                 6575000
##   Appraised.Land Appraised.Building Appraised.Outbuilding
## 1          49200             176000                     0
## 2         723000            8518300                 51100
##   Appraised.Extra.Feature Valuation.Year Zone Zone.Description Model Condition
## 1                      NA           2023   RA                      1        A
```

```
## 2                          NA          2021  PSR                              96        VG
##   Condition.Description  AYB  EYB Living.Area Effective.Area Total.Rooms
## 1             Average 2009 2013        1224           1602           5
## 2           Very Good 1895 1989       43105          45508          NA
##   Number.of.Bedroom Number.of.Baths Number.of.Half.Baths Occupancy Sale.Price
## 1                 3               2                    0         1     350000
## 2                NA              NA                   NA         1          0
##   Sale.Date Qualified Prior.Sale.Date Prior.Book.Page Prior.Sale.Price Editor
## 1                   Q                         656/332           270000      Z
## 2                   U                                                NA      z
##   Edit.Date Collection.Year Planning.Region State.Use State.Use.Description
## 1  3/7/2011            2023     Northeast CT      1010    Single Fam  MDL-01
## 2 3/17/2011            2023     Northeast CT        NA    PVT SCHOOL  MDL-96
##                               GlobalID Shape__Area Shape__Length
## 1 3d975e1f-76d8-4fb2-aa29-64f8a68c3d92    24859.03      1420.435
## 2 7bbd9dd8-5161-4b83-a8b1-12e771e06482   183081.64      1815.231
```

```r
df = df %>%
  filter(Assessed.Total != 0,
         grepl('Single Fam|SINGLE FAM|One Fam|ONE FAM',
               State.Use.Description),
         Qualified == 'Q',
         !is.na(Living.Area),
         Living.Area != 0,
         Town.Name == 'NEW HAVEN')
nrow(df)
```

```
## [1] 4060
```

We can aim to answer the question what is the relationship between Total Rooms and Assessed
Value? I hypothesize there will be a positive relationship, but it would be good to confirm
this in the data.

9.

```r
library(tidyr)

# Creating the plot and including an LS fit line
g <- ggplot(df) +
  aes(x = Total.Rooms, y = Assessed.Total) +
  geom_point(colour = "red") +
  geom_smooth(method = "lm", color = "blue", se = FALSE) +  # LS fit line
  theme_minimal()

g %>% pub()
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 3 rows containing non-finite outside the scale range
## (`stat_smooth()`).
```

```
## `geom_smooth()` using formula = 'y ~ x'
```

```
## Warning: Removed 3 rows containing non-finite outside the scale range
## ('stat_smooth()').

## 'geom_smooth()' using formula = 'y ~ x'

## Warning: Removed 3 rows containing non-finite outside the scale range
## ('stat_smooth()').

## 'geom_smooth()' using formula = 'y ~ x'

## Warning: Removed 3 rows containing non-finite outside the scale range
## ('stat_smooth()').

## 'geom_smooth()' using formula = 'y ~ x'

## Warning: Removed 3 rows containing non-finite outside the scale range
## ('stat_smooth()').

## 'geom_smooth()' using formula = 'y ~ x'

## Warning: Removed 3 rows containing non-finite outside the scale range
## ('stat_smooth()').

## 'geom_smooth()' using formula = 'y ~ x'

## Warning: Removed 3 rows containing non-finite outside the scale range
## ('stat_smooth()').

## [1] 137.3577
## [1] 80
## [1] 20
## [1] 237.3577

## 'geom_smooth()' using formula = 'y ~ x'

## Warning: Removed 3 rows containing non-finite outside the scale range
## ('stat_smooth()').

## Warning: Removed 3 rows containing missing values or values outside the scale range
## ('geom_point()').
```
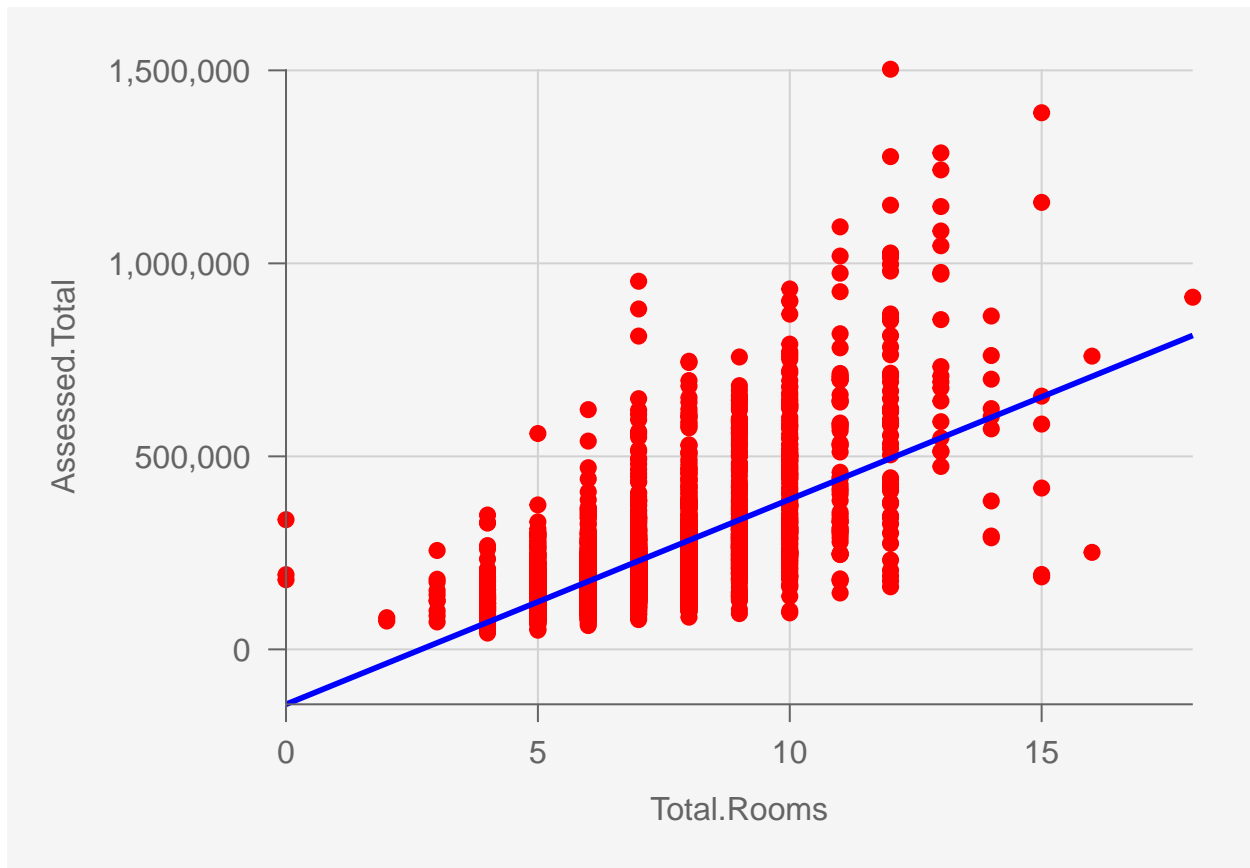
10.

We can see that Total Rooms and price are correlated which make sense. Additionally, there appears to be a positive linear relationship between the rooms discrete variable and the price (continuous variable).