title: “Vicki Hartley Course Project Phase 1 Delivery 2” output: word\_document: toc: true toc\_depth: 3

train = read\_csv("train.csv")

## Rows: 26570 Columns: 26  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (4): product\_code, attribute\_0, attribute\_1, failure  
## dbl (22): id, loading, attribute\_2, attribute\_3, measurement\_0, measurement\_...  
##   
## ℹ Use `spec()` to retrieve the full column specification for this data.  
## ℹ Specify the column types or set `show\_col\_types = FALSE` to quiet this message.

glimpse(train)

## Rows: 26,570  
## Columns: 26  
## $ id <dbl> 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, 10, 11, 12, 13, 14, 15, 1…  
## $ product\_code <chr> "A", "A", "A", "A", "A", "A", "A", "A", "A", "A", "A", …  
## $ loading <dbl> 80.10, 84.89, 82.43, 101.07, 188.06, 75.35, 161.71, 177…  
## $ attribute\_0 <chr> "material\_7", "material\_7", "material\_7", "material\_7",…  
## $ attribute\_1 <chr> "material\_8", "material\_8", "material\_8", "material\_8",…  
## $ attribute\_2 <dbl> 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9, 9…  
## $ attribute\_3 <dbl> 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5, 5…  
## $ measurement\_0 <dbl> 7, 14, 12, 13, 9, 11, 12, 4, 9, 10, 10, 7, 9, 6, 10, 8,…  
## $ measurement\_1 <dbl> 8, 3, 1, 2, 2, 4, 2, 8, 6, 4, 3, 3, 0, 9, 4, 5, 5, 6, 6…  
## $ measurement\_2 <dbl> 4, 3, 5, 6, 8, 0, 4, 8, 5, 7, 6, 2, 6, 10, 10, 9, 6, 3,…  
## $ measurement\_3 <dbl> 18.040, 18.213, 18.057, 17.295, 19.346, 17.564, 17.303,…  
## $ measurement\_4 <dbl> 12.518, 11.540, 11.652, 11.188, 12.950, 13.721, 12.643,…  
## $ measurement\_5 <dbl> 15.748, 17.717, 16.738, 18.576, 16.990, 16.594, 17.476,…  
## $ measurement\_6 <dbl> 19.292, 17.893, 18.240, 18.339, 15.746, NA, 17.679, 15.…  
## $ measurement\_7 <dbl> 11.739, 12.748, 12.718, 12.583, 11.306, 11.592, 12.957,…  
## $ measurement\_8 <dbl> 20.155, 17.889, 18.288, 19.060, 18.093, 20.810, 17.916,…  
## $ measurement\_9 <dbl> 10.672, 12.448, 12.715, 12.471, 10.337, 10.622, 11.370,…  
## $ measurement\_10 <dbl> 15.859, 17.947, 15.607, 16.346, 17.082, 14.904, 17.714,…  
## $ measurement\_11 <dbl> 17.594, 17.915, NA, 18.377, 19.932, 19.107, 19.924, 20.…  
## $ measurement\_12 <dbl> 15.193, 11.755, 13.798, 10.020, 12.428, 13.327, 11.560,…  
## $ measurement\_13 <dbl> 15.029, 14.732, 16.711, 15.250, 16.182, 15.354, 16.653,…  
## $ measurement\_14 <dbl> NA, 15.425, 18.631, 15.562, 12.760, 19.251, 17.734, 16.…  
## $ measurement\_15 <dbl> 13.034, 14.395, 14.094, 16.154, 13.153, NA, NA, 15.176,…  
## $ measurement\_16 <dbl> 14.684, 15.631, 17.946, 17.172, 16.412, 17.625, 16.637,…  
## $ measurement\_17 <dbl> 764.100, 682.057, 663.376, 826.282, 579.885, 832.902, 6…  
## $ failure <chr> "No", "No", "No", "No", "No", "No", "Yes", "Yes", "No",…

*Here is a summary and the structure of the data*

summary(train)

## id product\_code loading attribute\_0   
## Min. : 0 Length:26570 Min. : 33.16 Length:26570   
## 1st Qu.: 6642 Class :character 1st Qu.: 99.99 Class :character   
## Median :13284 Mode :character Median :122.39 Mode :character   
## Mean :13284 Mean :127.83   
## 3rd Qu.:19927 3rd Qu.:149.15   
## Max. :26569 Max. :385.86   
## NA's :250   
## attribute\_1 attribute\_2 attribute\_3 measurement\_0   
## Length:26570 Min. :5.000 Min. :5.00 Min. : 0.000   
## Class :character 1st Qu.:6.000 1st Qu.:6.00 1st Qu.: 4.000   
## Mode :character Median :6.000 Median :8.00 Median : 7.000   
## Mean :6.754 Mean :7.24 Mean : 7.416   
## 3rd Qu.:8.000 3rd Qu.:8.00 3rd Qu.:10.000   
## Max. :9.000 Max. :9.00 Max. :29.000   
##   
## measurement\_1 measurement\_2 measurement\_3 measurement\_4   
## Min. : 0.000 Min. : 0.000 Min. :13.97 Min. : 8.008   
## 1st Qu.: 5.000 1st Qu.: 4.000 1st Qu.:17.12 1st Qu.:11.051   
## Median : 8.000 Median : 6.000 Median :17.79 Median :11.733   
## Mean : 8.233 Mean : 6.257 Mean :17.79 Mean :11.732   
## 3rd Qu.:11.000 3rd Qu.: 8.000 3rd Qu.:18.47 3rd Qu.:12.410   
## Max. :29.000 Max. :24.000 Max. :21.50 Max. :16.484   
## NA's :381 NA's :538   
## measurement\_5 measurement\_6 measurement\_7 measurement\_8   
## Min. :12.07 Min. :12.71 Min. : 7.968 Min. :15.22   
## 1st Qu.:16.44 1st Qu.:16.84 1st Qu.:11.045 1st Qu.:18.34   
## Median :17.13 Median :17.52 Median :11.712 Median :19.02   
## Mean :17.13 Mean :17.51 Mean :11.717 Mean :19.02   
## 3rd Qu.:17.80 3rd Qu.:18.18 3rd Qu.:12.391 3rd Qu.:19.71   
## Max. :21.43 Max. :21.54 Max. :15.419 Max. :23.81   
## NA's :676 NA's :796 NA's :937 NA's :1048   
## measurement\_9 measurement\_10 measurement\_11 measurement\_12   
## Min. : 7.537 Min. : 9.323 Min. :12.46 Min. : 5.167   
## 1st Qu.:10.757 1st Qu.:15.209 1st Qu.:18.17 1st Qu.:10.703   
## Median :11.430 Median :16.127 Median :19.21 Median :11.717   
## Mean :11.431 Mean :16.118 Mean :19.17 Mean :11.703   
## 3rd Qu.:12.102 3rd Qu.:17.025 3rd Qu.:20.21 3rd Qu.:12.709   
## Max. :15.412 Max. :22.479 Max. :25.64 Max. :17.663   
## NA's :1227 NA's :1300 NA's :1468 NA's :1601   
## measurement\_13 measurement\_14 measurement\_15 measurement\_16   
## Min. :10.89 Min. : 9.14 Min. : 9.104 Min. : 9.701   
## 1st Qu.:14.89 1st Qu.:15.06 1st Qu.:13.957 1st Qu.:15.268   
## Median :15.63 Median :16.04 Median :14.969 Median :16.436   
## Mean :15.65 Mean :16.05 Mean :14.996 Mean :16.461   
## 3rd Qu.:16.37 3rd Qu.:17.08 3rd Qu.:16.018 3rd Qu.:17.628   
## Max. :22.71 Max. :22.30 Max. :21.626 Max. :24.094   
## NA's :1774 NA's :1874 NA's :2009 NA's :2110   
## measurement\_17 failure   
## Min. : 196.8 Length:26570   
## 1st Qu.: 619.0 Class :character   
## Median : 701.0 Mode :character   
## Mean : 701.3   
## 3rd Qu.: 784.1   
## Max. :1312.8   
## NA's :2284

str(train)

## spc\_tbl\_ [26,570 × 26] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ id : num [1:26570] 0 1 2 3 4 5 6 7 8 9 ...  
## $ product\_code : chr [1:26570] "A" "A" "A" "A" ...  
## $ loading : num [1:26570] 80.1 84.9 82.4 101.1 188.1 ...  
## $ attribute\_0 : chr [1:26570] "material\_7" "material\_7" "material\_7" "material\_7" ...  
## $ attribute\_1 : chr [1:26570] "material\_8" "material\_8" "material\_8" "material\_8" ...  
## $ attribute\_2 : num [1:26570] 9 9 9 9 9 9 9 9 9 9 ...  
## $ attribute\_3 : num [1:26570] 5 5 5 5 5 5 5 5 5 5 ...  
## $ measurement\_0 : num [1:26570] 7 14 12 13 9 11 12 4 9 10 ...  
## $ measurement\_1 : num [1:26570] 8 3 1 2 2 4 2 8 6 4 ...  
## $ measurement\_2 : num [1:26570] 4 3 5 6 8 0 4 8 5 7 ...  
## $ measurement\_3 : num [1:26570] 18 18.2 18.1 17.3 19.3 ...  
## $ measurement\_4 : num [1:26570] 12.5 11.5 11.7 11.2 12.9 ...  
## $ measurement\_5 : num [1:26570] 15.7 17.7 16.7 18.6 17 ...  
## $ measurement\_6 : num [1:26570] 19.3 17.9 18.2 18.3 15.7 ...  
## $ measurement\_7 : num [1:26570] 11.7 12.7 12.7 12.6 11.3 ...  
## $ measurement\_8 : num [1:26570] 20.2 17.9 18.3 19.1 18.1 ...  
## $ measurement\_9 : num [1:26570] 10.7 12.4 12.7 12.5 10.3 ...  
## $ measurement\_10: num [1:26570] 15.9 17.9 15.6 16.3 17.1 ...  
## $ measurement\_11: num [1:26570] 17.6 17.9 NA 18.4 19.9 ...  
## $ measurement\_12: num [1:26570] 15.2 11.8 13.8 10 12.4 ...  
## $ measurement\_13: num [1:26570] 15 14.7 16.7 15.2 16.2 ...  
## $ measurement\_14: num [1:26570] NA 15.4 18.6 15.6 12.8 ...  
## $ measurement\_15: num [1:26570] 13 14.4 14.1 16.2 13.2 ...  
## $ measurement\_16: num [1:26570] 14.7 15.6 17.9 17.2 16.4 ...  
## $ measurement\_17: num [1:26570] 764 682 663 826 580 ...  
## $ failure : chr [1:26570] "No" "No" "No" "No" ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. id = col\_double(),  
## .. product\_code = col\_character(),  
## .. loading = col\_double(),  
## .. attribute\_0 = col\_character(),  
## .. attribute\_1 = col\_character(),  
## .. attribute\_2 = col\_double(),  
## .. attribute\_3 = col\_double(),  
## .. measurement\_0 = col\_double(),  
## .. measurement\_1 = col\_double(),  
## .. measurement\_2 = col\_double(),  
## .. measurement\_3 = col\_double(),  
## .. measurement\_4 = col\_double(),  
## .. measurement\_5 = col\_double(),  
## .. measurement\_6 = col\_double(),  
## .. measurement\_7 = col\_double(),  
## .. measurement\_8 = col\_double(),  
## .. measurement\_9 = col\_double(),  
## .. measurement\_10 = col\_double(),  
## .. measurement\_11 = col\_double(),  
## .. measurement\_12 = col\_double(),  
## .. measurement\_13 = col\_double(),  
## .. measurement\_14 = col\_double(),  
## .. measurement\_15 = col\_double(),  
## .. measurement\_16 = col\_double(),  
## .. measurement\_17 = col\_double(),  
## .. failure = col\_character()  
## .. )  
## - attr(\*, "problems")=<externalptr>

*Any character variables must be mutated to factors*

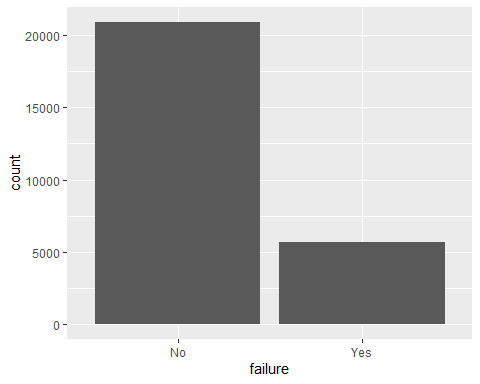
train = train %>% mutate(product\_code = as\_factor(product\_code))  
train = train %>% mutate(attribute\_0 = as\_factor(attribute\_0 ))  
train = train %>% mutate(attribute\_1 = as\_factor(attribute\_1 ))  
train = train %>% mutate(failure = as\_factor(failure )) %>%  
 mutate(failure = fct\_recode(failure, "No" = "0", "Yes" = "1"))

## Warning: There was 1 warning in `mutate()`.  
## ℹ In argument: `failure = fct\_recode(failure, No = "0", Yes = "1")`.  
## Caused by warning:  
## ! Unknown levels in `f`: 0, 1

str(train)

## tibble [26,570 × 26] (S3: tbl\_df/tbl/data.frame)  
## $ id : num [1:26570] 0 1 2 3 4 5 6 7 8 9 ...  
## $ product\_code : Factor w/ 5 levels "A","B","C","D",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ loading : num [1:26570] 80.1 84.9 82.4 101.1 188.1 ...  
## $ attribute\_0 : Factor w/ 2 levels "material\_7","material\_5": 1 1 1 1 1 1 1 1 1 1 ...  
## $ attribute\_1 : Factor w/ 3 levels "material\_8","material\_5",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ attribute\_2 : num [1:26570] 9 9 9 9 9 9 9 9 9 9 ...  
## $ attribute\_3 : num [1:26570] 5 5 5 5 5 5 5 5 5 5 ...  
## $ measurement\_0 : num [1:26570] 7 14 12 13 9 11 12 4 9 10 ...  
## $ measurement\_1 : num [1:26570] 8 3 1 2 2 4 2 8 6 4 ...  
## $ measurement\_2 : num [1:26570] 4 3 5 6 8 0 4 8 5 7 ...  
## $ measurement\_3 : num [1:26570] 18 18.2 18.1 17.3 19.3 ...  
## $ measurement\_4 : num [1:26570] 12.5 11.5 11.7 11.2 12.9 ...  
## $ measurement\_5 : num [1:26570] 15.7 17.7 16.7 18.6 17 ...  
## $ measurement\_6 : num [1:26570] 19.3 17.9 18.2 18.3 15.7 ...  
## $ measurement\_7 : num [1:26570] 11.7 12.7 12.7 12.6 11.3 ...  
## $ measurement\_8 : num [1:26570] 20.2 17.9 18.3 19.1 18.1 ...  
## $ measurement\_9 : num [1:26570] 10.7 12.4 12.7 12.5 10.3 ...  
## $ measurement\_10: num [1:26570] 15.9 17.9 15.6 16.3 17.1 ...  
## $ measurement\_11: num [1:26570] 17.6 17.9 NA 18.4 19.9 ...  
## $ measurement\_12: num [1:26570] 15.2 11.8 13.8 10 12.4 ...  
## $ measurement\_13: num [1:26570] 15 14.7 16.7 15.2 16.2 ...  
## $ measurement\_14: num [1:26570] NA 15.4 18.6 15.6 12.8 ...  
## $ measurement\_15: num [1:26570] 13 14.4 14.1 16.2 13.2 ...  
## $ measurement\_16: num [1:26570] 14.7 15.6 17.9 17.2 16.4 ...  
## $ measurement\_17: num [1:26570] 764 682 663 826 580 ...  
## $ failure : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 2 1 1 ...

ggplot(train,aes(failure)) + geom\_bar() + theme\_get()



proportions <- table(train$failure)/length(train$failure)  
percentages <- proportions\*100

*I noticed there is missingness*

skim(train)

Data summary

|  |  |
| --- | --- |
| Name | train |
| Number of rows | 26570 |
| Number of columns | 26 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 4 |
| numeric | 22 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: factor**

| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| --- | --- | --- | --- | --- | --- |
| product\_code | 0 | 1 | FALSE | 5 | C: 5765, E: 5343, B: 5250, D: 5112 |
| attribute\_0 | 0 | 1 | FALSE | 2 | mat: 21320, mat: 5250 |
| attribute\_1 | 0 | 1 | FALSE | 3 | mat: 10865, mat: 10362, mat: 5343 |
| failure | 0 | 1 | FALSE | 2 | No: 20921, Yes: 5649 |

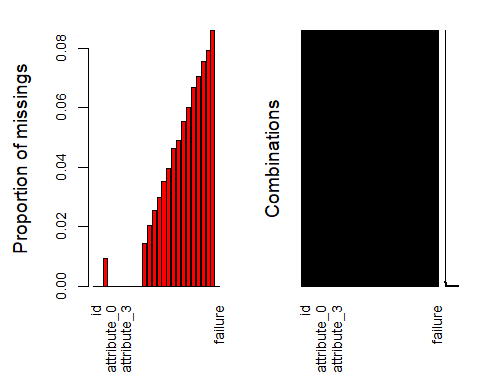
**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | 0 | 1.00 | 13284.50 | 7670.24 | 0.00 | 6642.25 | 13284.50 | 19926.75 | 26569.00 | ▇▇▇▇▇ |
| loading | 250 | 0.99 | 127.83 | 39.03 | 33.16 | 99.99 | 122.39 | 149.15 | 385.86 | ▃▇▂▁▁ |
| attribute\_2 | 0 | 1.00 | 6.75 | 1.47 | 5.00 | 6.00 | 6.00 | 8.00 | 9.00 | ▅▇▁▅▃ |
| attribute\_3 | 0 | 1.00 | 7.24 | 1.46 | 5.00 | 6.00 | 8.00 | 8.00 | 9.00 | ▃▃▁▇▃ |
| measurement\_0 | 0 | 1.00 | 7.42 | 4.12 | 0.00 | 4.00 | 7.00 | 10.00 | 29.00 | ▆▇▂▁▁ |
| measurement\_1 | 0 | 1.00 | 8.23 | 4.20 | 0.00 | 5.00 | 8.00 | 11.00 | 29.00 | ▅▇▃▁▁ |
| measurement\_2 | 0 | 1.00 | 6.26 | 3.31 | 0.00 | 4.00 | 6.00 | 8.00 | 24.00 | ▅▇▂▁▁ |
| measurement\_3 | 381 | 0.99 | 17.79 | 1.00 | 13.97 | 17.12 | 17.79 | 18.47 | 21.50 | ▁▃▇▃▁ |
| measurement\_4 | 538 | 0.98 | 11.73 | 1.00 | 8.01 | 11.05 | 11.73 | 12.41 | 16.48 | ▁▅▇▁▁ |
| measurement\_5 | 676 | 0.97 | 17.13 | 1.00 | 12.07 | 16.44 | 17.13 | 17.80 | 21.42 | ▁▁▇▃▁ |
| measurement\_6 | 796 | 0.97 | 17.51 | 1.00 | 12.71 | 16.84 | 17.52 | 18.18 | 21.54 | ▁▂▇▅▁ |
| measurement\_7 | 937 | 0.96 | 11.72 | 1.00 | 7.97 | 11.04 | 11.71 | 12.39 | 15.42 | ▁▃▇▃▁ |
| measurement\_8 | 1048 | 0.96 | 19.02 | 1.01 | 15.22 | 18.34 | 19.02 | 19.71 | 23.81 | ▁▅▇▂▁ |
| measurement\_9 | 1227 | 0.95 | 11.43 | 1.00 | 7.54 | 10.76 | 11.43 | 12.10 | 15.41 | ▁▃▇▃▁ |
| measurement\_10 | 1300 | 0.95 | 16.12 | 1.41 | 9.32 | 15.21 | 16.13 | 17.02 | 22.48 | ▁▂▇▂▁ |
| measurement\_11 | 1468 | 0.94 | 19.17 | 1.52 | 12.46 | 18.17 | 19.21 | 20.21 | 25.64 | ▁▂▇▃▁ |
| measurement\_12 | 1601 | 0.94 | 11.70 | 1.49 | 5.17 | 10.70 | 11.72 | 12.71 | 17.66 | ▁▂▇▃▁ |
| measurement\_13 | 1774 | 0.93 | 15.65 | 1.16 | 10.89 | 14.89 | 15.63 | 16.37 | 22.71 | ▁▇▇▁▁ |
| measurement\_14 | 1874 | 0.93 | 16.05 | 1.49 | 9.14 | 15.06 | 16.04 | 17.08 | 22.30 | ▁▂▇▃▁ |
| measurement\_15 | 2009 | 0.92 | 15.00 | 1.55 | 9.10 | 13.96 | 14.97 | 16.02 | 21.63 | ▁▃▇▂▁ |
| measurement\_16 | 2110 | 0.92 | 16.46 | 1.71 | 9.70 | 15.27 | 16.44 | 17.63 | 24.09 | ▁▃▇▂▁ |
| measurement\_17 | 2284 | 0.91 | 701.27 | 123.30 | 196.79 | 618.96 | 701.02 | 784.09 | 1312.79 | ▁▅▇▁▁ |

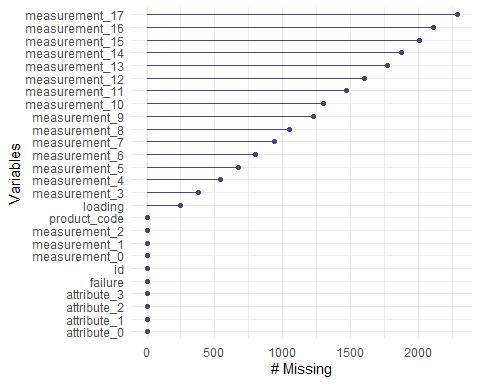
*Looking to see if there is overlap in missingness*

vim\_plot = aggr(train, numbers = TRUE, prop = c(TRUE, FALSE),cex.axis=.9)

## Warning in plot.aggr(res, ...): not enough vertical space to display  
## frequencies (too many combinations)

 Most of the missing data are on the measurement features

gg\_miss\_var(train)



*Best to imputate missing data for loading*

set.seed(1234)   
imp\_measurement\_17 = train = mice(train, m=5, method='pmm', printFlag=FALSE)

## Warning: Number of logged events: 756

train = complete(imp\_measurement\_17)

*Row-wise deletion ok for measurements since not many outliers in data*

train = train %>% drop\_na()   
skim(train)

Data summary

|  |  |
| --- | --- |
| Name | train |
| Number of rows | 26570 |
| Number of columns | 26 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 4 |
| numeric | 22 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Group variables | None |

**Variable type: factor**

| skim\_variable | n\_missing | complete\_rate | ordered | n\_unique | top\_counts |
| --- | --- | --- | --- | --- | --- |
| product\_code | 0 | 1 | FALSE | 5 | C: 5765, E: 5343, B: 5250, D: 5112 |
| attribute\_0 | 0 | 1 | FALSE | 2 | mat: 21320, mat: 5250 |
| attribute\_1 | 0 | 1 | FALSE | 3 | mat: 10865, mat: 10362, mat: 5343 |
| failure | 0 | 1 | FALSE | 2 | No: 20921, Yes: 5649 |

**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| id | 0 | 1 | 13284.50 | 7670.24 | 0.00 | 6642.25 | 13284.50 | 19926.75 | 26569.00 | ▇▇▇▇▇ |
| loading | 0 | 1 | 127.82 | 39.03 | 33.16 | 99.96 | 122.37 | 149.16 | 385.86 | ▃▇▂▁▁ |
| attribute\_2 | 0 | 1 | 6.75 | 1.47 | 5.00 | 6.00 | 6.00 | 8.00 | 9.00 | ▅▇▁▅▃ |
| attribute\_3 | 0 | 1 | 7.24 | 1.46 | 5.00 | 6.00 | 8.00 | 8.00 | 9.00 | ▃▃▁▇▃ |
| measurement\_0 | 0 | 1 | 7.42 | 4.12 | 0.00 | 4.00 | 7.00 | 10.00 | 29.00 | ▆▇▂▁▁ |
| measurement\_1 | 0 | 1 | 8.23 | 4.20 | 0.00 | 5.00 | 8.00 | 11.00 | 29.00 | ▅▇▃▁▁ |
| measurement\_2 | 0 | 1 | 6.26 | 3.31 | 0.00 | 4.00 | 6.00 | 8.00 | 24.00 | ▅▇▂▁▁ |
| measurement\_3 | 0 | 1 | 17.80 | 1.01 | 13.97 | 17.12 | 17.79 | 18.48 | 21.50 | ▁▃▇▃▁ |
| measurement\_4 | 0 | 1 | 11.71 | 1.01 | 8.01 | 11.01 | 11.72 | 12.40 | 16.48 | ▁▅▇▁▁ |
| measurement\_5 | 0 | 1 | 17.08 | 1.03 | 12.07 | 16.39 | 17.10 | 17.79 | 21.42 | ▁▂▇▃▁ |
| measurement\_6 | 0 | 1 | 17.51 | 1.00 | 12.71 | 16.84 | 17.52 | 18.18 | 21.54 | ▁▂▇▅▁ |
| measurement\_7 | 0 | 1 | 11.72 | 1.00 | 7.97 | 11.05 | 11.71 | 12.39 | 15.42 | ▁▃▇▃▁ |
| measurement\_8 | 0 | 1 | 19.12 | 1.12 | 15.22 | 18.38 | 19.07 | 19.81 | 23.81 | ▁▅▇▂▁ |
| measurement\_9 | 0 | 1 | 11.41 | 0.99 | 7.54 | 10.73 | 11.42 | 12.06 | 15.41 | ▁▃▇▂▁ |
| measurement\_10 | 0 | 1 | 16.08 | 1.40 | 9.32 | 15.17 | 16.11 | 16.98 | 22.48 | ▁▂▇▂▁ |
| measurement\_11 | 0 | 1 | 19.28 | 1.55 | 12.46 | 18.24 | 19.32 | 20.35 | 25.64 | ▁▂▇▃▁ |
| measurement\_12 | 0 | 1 | 11.70 | 1.51 | 5.17 | 10.72 | 11.69 | 12.70 | 17.66 | ▁▂▇▃▁ |
| measurement\_13 | 0 | 1 | 15.64 | 1.14 | 10.89 | 14.85 | 15.65 | 16.31 | 22.71 | ▁▇▇▁▁ |
| measurement\_14 | 0 | 1 | 16.09 | 1.52 | 9.14 | 15.08 | 16.08 | 17.13 | 22.30 | ▁▂▇▃▁ |
| measurement\_15 | 0 | 1 | 14.81 | 1.68 | 9.10 | 13.82 | 14.81 | 15.92 | 21.63 | ▁▃▇▂▁ |
| measurement\_16 | 0 | 1 | 16.72 | 2.13 | 9.70 | 15.26 | 16.53 | 17.85 | 24.09 | ▁▅▇▂▁ |
| measurement\_17 | 0 | 1 | 697.34 | 131.76 | 196.79 | 614.14 | 698.64 | 783.65 | 1312.79 | ▁▅▇▁▁ |

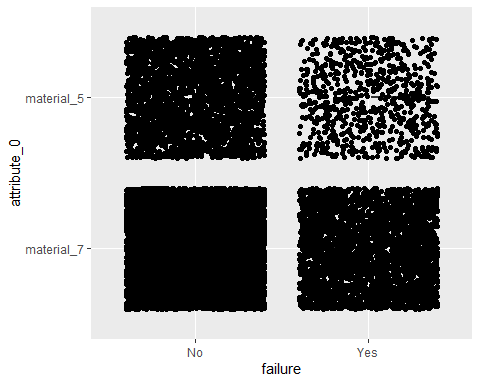
*Split the data into testing (30%) and training (70%) stratified by the “failure” variable*

set.seed(123)  
train\_split = initial\_split (train, prop = 0.70, strata = failure)  
train\_train = training(train\_split)  
test\_train = testing(train\_split)  
str(train\_train)

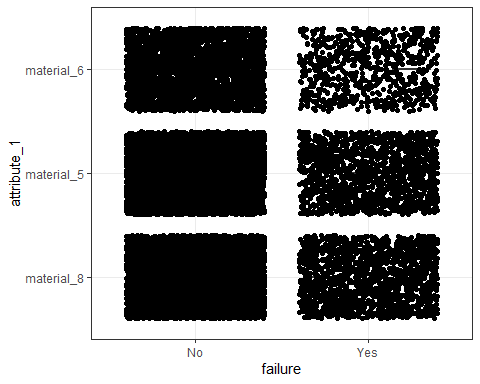
## 'data.frame': 18598 obs. of 26 variables:  
## $ id : num 0 1 3 4 5 8 10 12 14 16 ...  
## $ product\_code : Factor w/ 5 levels "A","B","C","D",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ loading : num 80.1 84.9 101.1 188.1 75.3 ...  
## $ attribute\_0 : Factor w/ 2 levels "material\_7","material\_5": 1 1 1 1 1 1 1 1 1 1 ...  
## $ attribute\_1 : Factor w/ 3 levels "material\_8","material\_5",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ attribute\_2 : num 9 9 9 9 9 9 9 9 9 9 ...  
## $ attribute\_3 : num 5 5 5 5 5 5 5 5 5 5 ...  
## $ measurement\_0 : num 7 14 13 9 11 9 10 9 10 10 ...  
## $ measurement\_1 : num 8 3 2 2 4 6 3 0 4 5 ...  
## $ measurement\_2 : num 4 3 6 8 0 5 6 6 10 6 ...  
## $ measurement\_3 : num 18 18.2 17.3 19.3 17.6 ...  
## $ measurement\_4 : num 12.5 11.5 11.2 12.9 13.7 ...  
## $ measurement\_5 : num 15.7 17.7 18.6 17 16.6 ...  
## $ measurement\_6 : num 19.3 17.9 18.3 15.7 18 ...  
## $ measurement\_7 : num 11.7 12.7 12.6 11.3 11.6 ...  
## $ measurement\_8 : num 20.2 17.9 19.1 18.1 20.8 ...  
## $ measurement\_9 : num 10.7 12.4 12.5 10.3 10.6 ...  
## $ measurement\_10: num 15.9 17.9 16.3 17.1 14.9 ...  
## $ measurement\_11: num 17.6 17.9 18.4 19.9 19.1 ...  
## $ measurement\_12: num 15.2 11.8 10 12.4 13.3 ...  
## $ measurement\_13: num 15 14.7 15.2 16.2 15.4 ...  
## $ measurement\_14: num 19 15.4 15.6 12.8 19.3 ...  
## $ measurement\_15: num 13 14.4 16.2 13.2 14.3 ...  
## $ measurement\_16: num 14.7 15.6 17.2 16.4 17.6 ...  
## $ measurement\_17: num 764 682 826 580 833 ...  
## $ failure : Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 1 1 1 1 ...

*It looks like some attributes could have an impact on failure*

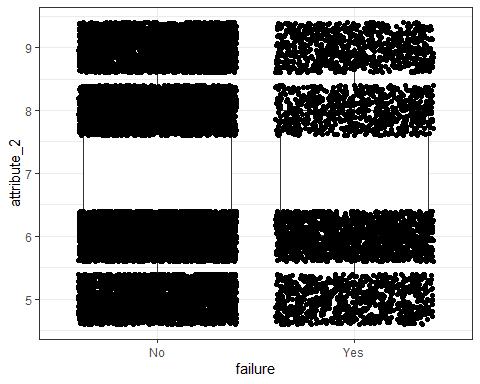
ggplot(train\_train,aes(x=failure,y=attribute\_0)) + geom\_boxplot() + theme\_get() + geom\_jitter()



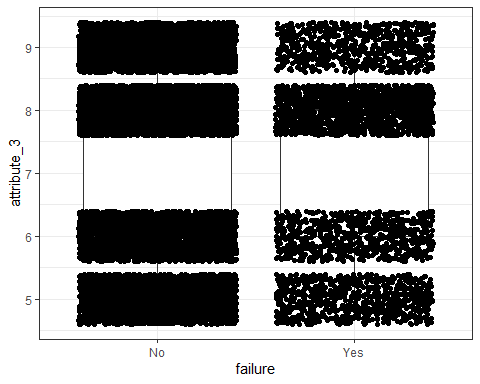
ggplot(train\_train,aes(x=failure,y=attribute\_1)) + geom\_boxplot() + theme\_bw() + geom\_jitter()



ggplot(train\_train,aes(x=failure,y=attribute\_2)) + geom\_boxplot() + theme\_bw() + geom\_jitter()

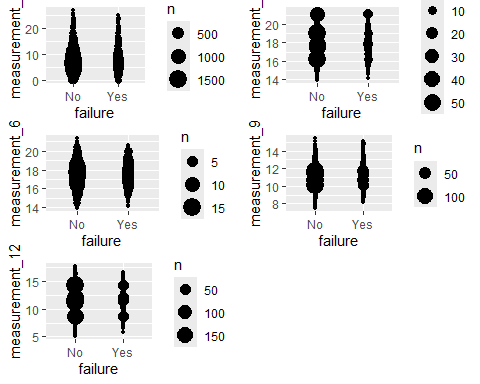


ggplot(train\_train,aes(x=failure,y=attribute\_3)) + geom\_boxplot() + theme\_bw() + geom\_jitter()

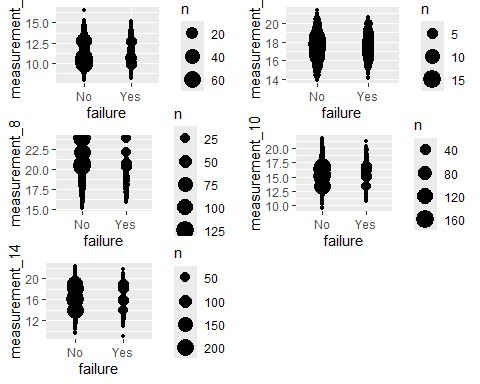


# It is hard to tell if measurements impact failure from this random set of measurments

p1 = ggplot(train\_train,aes(x=failure,y=measurement\_0)) + geom\_count()  
p2 = ggplot(train\_train,aes(x=failure,y=measurement\_3)) + geom\_count()  
p3 = ggplot(train\_train,aes(x=failure,y=measurement\_6)) + geom\_count()  
p4 = ggplot(train\_train,aes(x=failure,y=measurement\_9)) + geom\_count()  
p5 = ggplot(train\_train,aes(x=failure,y=measurement\_12)) + geom\_count()  
grid.arrange(p1,p2,p3,p4,p5, ncol = 2)



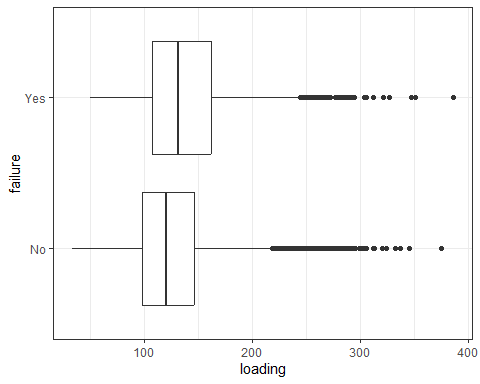
p1 = ggplot(train\_train,aes(x=failure,y=measurement\_4)) + geom\_count()  
p2 = ggplot(train\_train,aes(x=failure,y=measurement\_6)) + geom\_count()  
p3 = ggplot(train\_train,aes(x=failure,y=measurement\_8)) + geom\_count()  
p4 = ggplot(train\_train,aes(x=failure,y=measurement\_10)) + geom\_count()  
p5 = ggplot(train\_train,aes(x=failure,y=measurement\_14)) + geom\_count()  
grid.arrange(p1,p2,p3,p4,p5, ncol = 2)



#Now we can look at loading as a predictor of failure

*The means for loading and failure are similar*

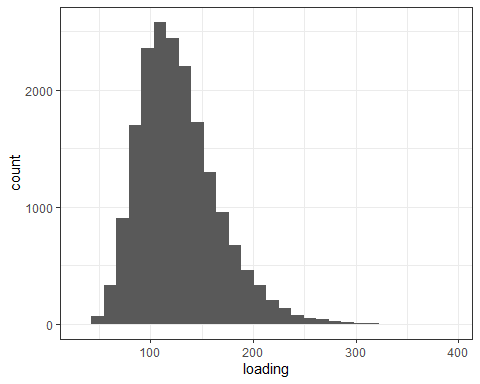
ggplot(train\_train,aes(x=loading,y=failure)) + geom\_boxplot() + theme\_bw()



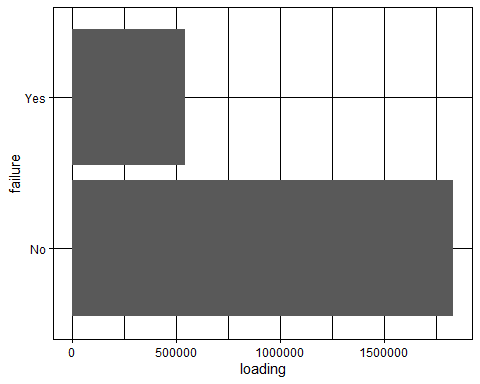
*loading appears to be normally distributed*

ggplot(train\_train, aes(x=loading)) + geom\_histogram() + theme\_bw()

## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.

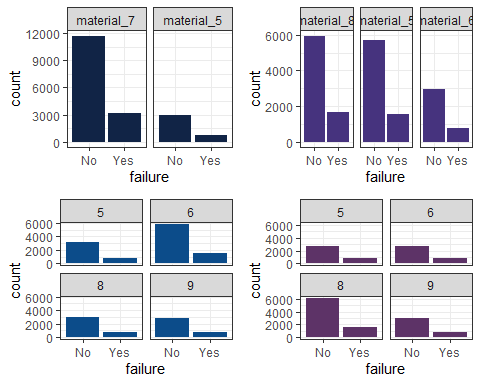


ggplot(train\_train,aes(x=loading,y=failure)) + geom\_col() + theme\_linedraw()



*It looks like certain attributes impact failure.* *specifically Material 7 within Attribute\_0 had around 3,100 failures which was the most of any material*

#esquisser()  
  
  
p1 = ggplot(train\_train) +  
 aes(x = failure) +  
 geom\_bar(fill = "#112446") +  
 theme\_bw() +  
 facet\_wrap(vars(attribute\_0))  
  
p2 =ggplot(train\_train) +  
 aes(x = failure) +  
 geom\_bar(fill = "#46337E") +  
 theme\_bw() +  
 facet\_wrap(vars(attribute\_1))  
  
p3 = ggplot(train\_train) +  
 aes(x = failure) +  
 geom\_bar(fill = "#0C4C8A") +  
 theme\_bw() +  
 facet\_wrap(vars(attribute\_2))  
  
p4 = ggplot(train\_train) +  
 aes(x = failure) +  
 geom\_bar(fill = "#5D3367") +  
 theme\_bw() +  
 facet\_wrap(vars(attribute\_3))  
  
grid.arrange(p1,p2,p3,p4, ncol = 2)



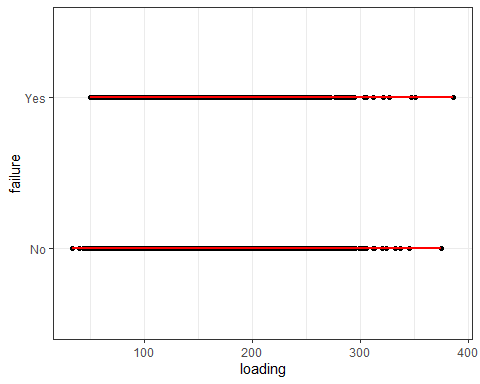
*The P value for loading is less than 0.05 so it statistically significant to predicting failure*

load\_recipe = recipe(loading ~ failure, train\_train) %>%  
 step\_dummy(all\_nominal\_predictors())  
  
lm\_model =   
 linear\_reg() %>%   
 set\_engine("lm")   
  
lm\_wflow =   
 workflow() %>%   
 add\_model(lm\_model) %>%   
 add\_recipe(load\_recipe)  
  
lm\_fit = fit(lm\_wflow, train\_train)  
summary(lm\_fit$fit$fit$fit)

##   
## Call:  
## stats::lm(formula = ..y ~ ., data = data)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -92.081 -27.859 -5.336 21.584 249.089   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 125.2410 0.3192 392.42 <2e-16 \*\*\*  
## failure\_Yes 11.8824 0.6922 17.17 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 38.62 on 18596 degrees of freedom  
## Multiple R-squared: 0.0156, Adjusted R-squared: 0.01555   
## F-statistic: 294.7 on 1 and 18596 DF, p-value: < 2.2e-16

ggplot(train\_train, aes(x=loading, y=failure)) + geom\_point() + geom\_smooth(method = "lm",se=FALSE, color="red") + theme\_bw()

## `geom\_smooth()` using formula = 'y ~ x'



train\_pred\_load = predict(lm\_fit, train\_train)  
head(train\_pred\_load)

## # A tibble: 6 × 1  
## .pred  
## <dbl>  
## 1 125.  
## 2 125.  
## 3 125.  
## 4 125.  
## 5 125.  
## 6 125.