failurefull <- 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.

**structure and summary**

str(failurefull)

## 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>

summary(failurefull)

## 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

**Remove ID colum**

failurefull = failurefull %>% select( -id)

## Factor Conversion and reclassification

#I'm going to convert attributes 2 & 3 to factors as well even though they are numeric because I assume all of the attributes are indicative of the material type given the labels on attributes 0 & 1 (i.e. "material\_8")   
  
failurefull = failurefull %>% mutate\_if(is.character,as\_factor)%>%   
 mutate(attribute\_2 = as\_factor(attribute\_2))%>%  
 mutate(attribute\_3 = as\_factor(attribute\_3))%>%  
 mutate(failure = fct\_recode(failure, "No" = "0", "Yes" = "1"))

## Warning: Unknown levels in `f`: 0, 1

#Let's explore the values of == 0 for measurements 0, 1, & 2 which shows no missingness.   
  
failurefull %>% filter(measurement\_0 == 0)  
  
#538 rows where measurement\_0 == 0   
  
failurefull %>% filter(measurement\_1 == 0)  
  
#378 rows where measurement\_1 == 0   
  
failurefull %>% filter(measurement\_2 == 0)  
  
# 418 rows where measurement\_2 == 0  
  
#given every other measurement has misssingness and these seem like NA's rather than true zeroes. Let's convert.

## Convert zeroes to NA

failurefull = failurefull %>% mutate(measurement\_0 = na\_if(measurement\_0, "0"))%>%  
 mutate(measurement\_1 = na\_if(measurement\_1, "0"))%>%  
 mutate(measurement\_2 = na\_if(measurement\_2, "0"))

skim(failurefull)

Data summary

|  |  |
| --- | --- |
| Name | failurefull |
| Number of rows | 26570 |
| Number of columns | 25 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| Column type frequency: |  |
| factor | 6 |
| numeric | 19 |
| \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_ |  |
| 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 |
| attribute\_2 | 0 | 1 | FALSE | 4 | 6: 10455, 5: 5765, 8: 5250, 9: 5100 |
| attribute\_3 | 0 | 1 | FALSE | 4 | 8: 11015, 9: 5343, 6: 5112, 5: 5100 |
| failure | 0 | 1 | FALSE | 2 | No: 20921, Yes: 5649 |

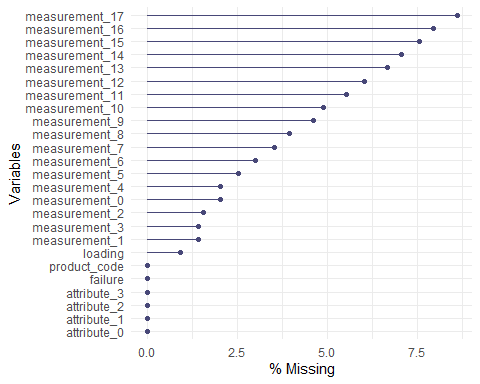
**Variable type: numeric**

| skim\_variable | n\_missing | complete\_rate | mean | sd | p0 | p25 | p50 | p75 | p100 | hist |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| loading | 250 | 0.99 | 127.83 | 39.03 | 33.16 | 99.99 | 122.39 | 149.15 | 385.86 | ▃▇▂▁▁ |
| measurement\_0 | 538 | 0.98 | 7.57 | 4.02 | 1.00 | 5.00 | 7.00 | 10.00 | 29.00 | ▇▇▂▁▁ |
| measurement\_1 | 378 | 0.99 | 8.35 | 4.11 | 1.00 | 5.00 | 8.00 | 11.00 | 29.00 | ▆▇▂▁▁ |
| measurement\_2 | 418 | 0.98 | 6.36 | 3.24 | 1.00 | 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 | ▁▅▇▁▁ |

#These complete\_rates for measurements 0, 1, & 2 seem more plausible given the 90 - 98.5% complete\_rate of the other measurements. I wonder if these values will be more predictive of failure than before converting to NA?

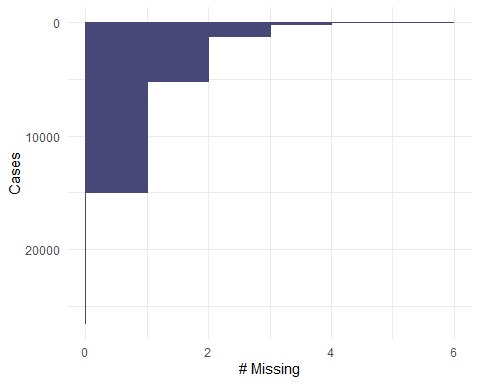
**Visualizing Missingness**

gg\_miss\_var(failurefull, show\_pct = TRUE)



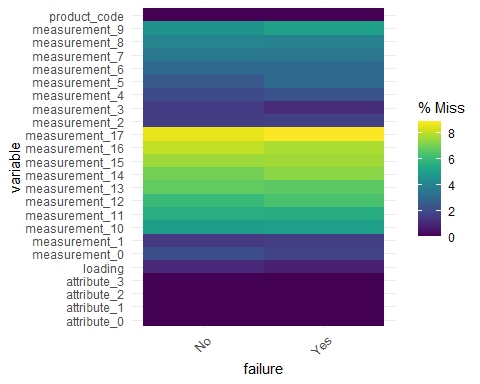
#measurements 11 through 17 have > 5% missingnes but all variables are 90%+ complete and so column-deletion is probably too aggressive here.

gg\_miss\_case(failurefull)



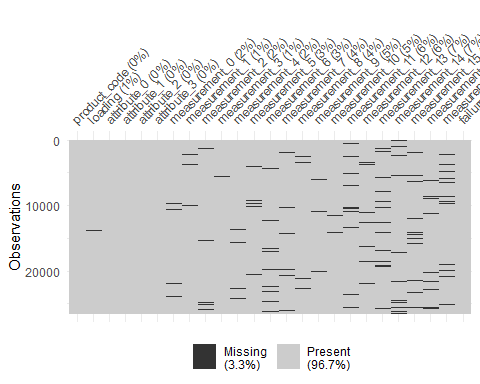
#nearly 15,000 observations have 1 missing value. Over 5,000 observations have 2 missing values.

gg\_miss\_fct(x = failurefull, fct = failure)



#there doesn't appear to be a dramatic difference in missingness of any values for products that failed. We aren't missing critical data.

vis\_miss(failurefull)



#The missingness appears to be random with no discernible patterns. Row-wise deletion would eliminate a significant amount of data. Similarly, column-wise deletion would completely eliminate variables that are 90%+ complete. For these reasons, I am going to pursue imputation to handle missingness. However, I am going to use row-wise deletion for missingness of the loading variable because it is a key measurement of the product's usefulness.

failurefull = failurefull %>% drop\_na(loading)  
  
#row-wise deletion for the loading variable before proceeding with imputation.

**Imputation**

imp\_vals = mice(failurefull, m=3, method='pmm', printFlag=FALSE)

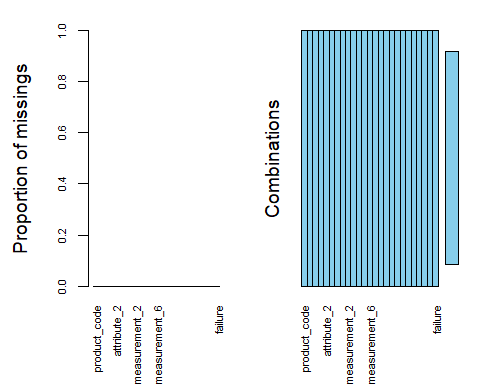
## Warning: Number of logged events: 539

#m is the number of imputations, 3 is used here rather than 5 due to dataset size.   
#pmm is "predictive mean matching" = imputation method for numeric data  
#printFlag reduces amount of output

summary(imp\_vals)

failurefull\_complete = complete(imp\_vals)  
  
vim\_plot = aggr(failurefull\_complete, numbers = TRUE, prop = c(TRUE, FALSE),cex.axis=.7)

## Warning in plot.aggr(res, ...): not enough horizontal space to display  
## frequencies



#data is complete, no missingness.

**Exploratory Analysis**

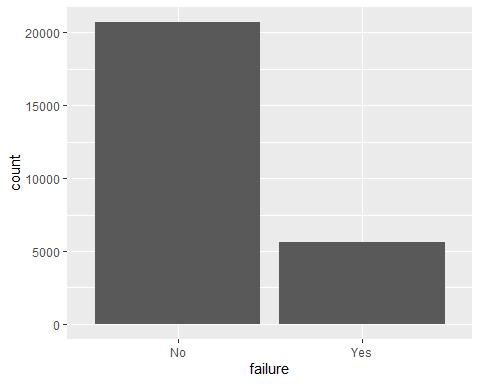
str(failurefull$failure)

## Factor w/ 2 levels "No","Yes": 1 1 1 1 1 1 2 2 1 1 ...

summary(failurefull$failure)

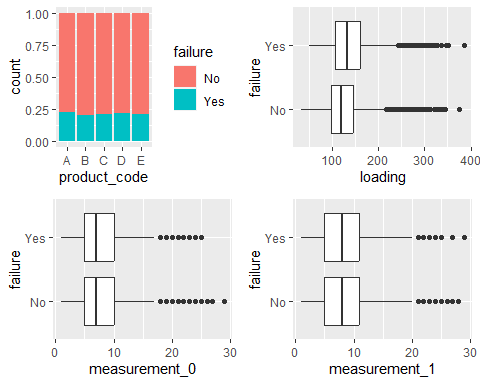
## No Yes   
## 20715 5605

ggplot(failurefull, aes(x = failure)) + geom\_bar()



#There are 5649 failures out of 26570 observations

p1 = ggplot(failurefull\_complete, aes(x= product\_code, fill = failure)) + geom\_bar(position = "fill")  
p2 = ggplot(failurefull\_complete, aes(x = loading, y = failure))+ geom\_boxplot()  
p3 = ggplot(failurefull\_complete, aes(x = measurement\_0, y = failure))+ geom\_boxplot()  
p4 = ggplot(failurefull\_complete, aes(x = measurement\_1, y = failure))+ geom\_boxplot()  
grid.arrange(p1, p2, p3, p4, ncol = 2)



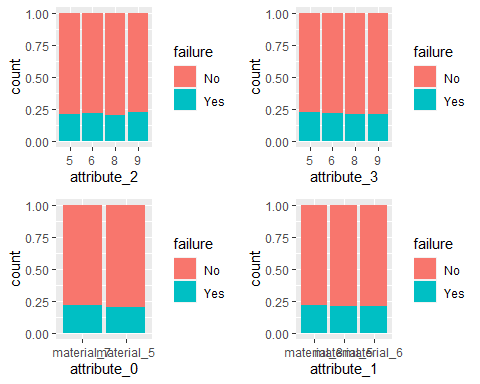
# The response variable is failure.   
#loading is predictive of failure   
#product code is predictive of failure   
#There is no discernible predictive effect of measurements 0 & 1 on failure.

#Above, we can see that the proportion of product failures are clustered around 20% for all product codes. Let's look at the interaction between product code and loading on failure.   
  
ggplot(failurefull\_complete, aes(x = loading, y= product\_code, color = failure))+ geom\_point(alpha = 0.3,position = "jitter")



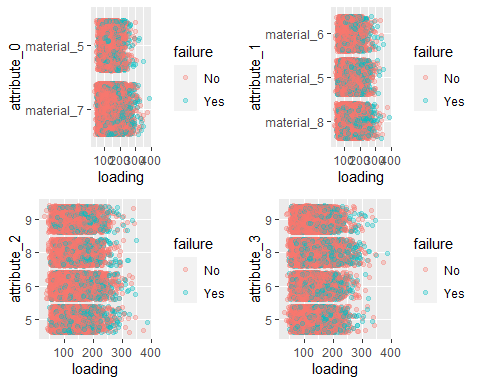
#clearly there is increased product failure at higher loading values. However, it's hard to tell much else.

p5 = ggplot(failurefull\_complete, aes(x = attribute\_2, fill = failure))+ geom\_bar(position = "fill")  
p6 = ggplot(failurefull\_complete, aes(x = attribute\_3, fill = failure))+ geom\_bar(position = "fill")  
p7 = ggplot(failurefull\_complete, aes( x = attribute\_0, fill = failure)) + geom\_bar(position = "fill")  
p8 = ggplot(failurefull\_complete, aes( x = attribute\_1, fill = failure)) + geom\_bar(position = "fill")  
  
grid.arrange(p5, p6, p7, p8, ncol = 2)



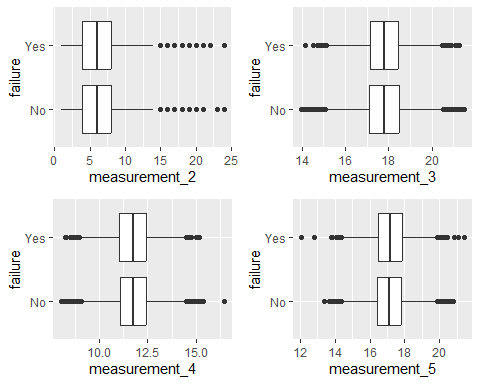
#Attributes 0, 2 & 3 appear to be predictors of failure.

#let's look at the interaction between loading amount across the various attributes. It appears that the various attributes strongly influence the loading value.  
  
p25 = ggplot(failurefull\_complete, aes(x = loading, y= attribute\_0, color = failure))+ geom\_point(alpha = 0.3,position = "jitter")  
p26 = ggplot(failurefull\_complete, aes(x = loading, y= attribute\_1, color = failure))+ geom\_point(alpha = 0.3,position = "jitter")  
p27 = ggplot(failurefull\_complete, aes(x = loading, y= attribute\_2, color = failure))+ geom\_point(alpha = 0.3,position = "jitter")   
p28 = ggplot(failurefull\_complete, aes(x = loading, y= attribute\_3, color = failure))+ geom\_point(alpha = 0.3,position = "jitter")  
  
grid.arrange(p25, p26, p27, p28, ncol = 2)



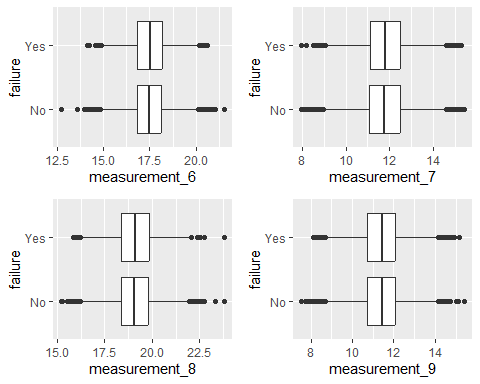
# This wasn't as useful as I had hoped!

p9 = ggplot(failurefull\_complete, aes(x = measurement\_2, y = failure))+ geom\_boxplot()  
p10 = ggplot(failurefull\_complete, aes(x = measurement\_3, y = failure))+ geom\_boxplot()  
p11 = ggplot(failurefull\_complete, aes(x = measurement\_4, y = failure))+ geom\_boxplot()  
p12 = ggplot(failurefull\_complete, aes(x = measurement\_5, y = failure))+ geom\_boxplot()  
grid.arrange(p9, p10, p11, p12, ncol = 2)



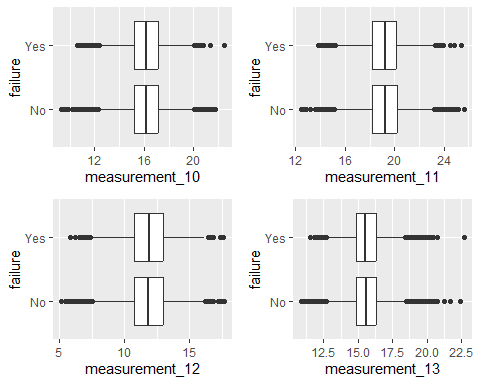
#No discernible predictive value of these measurements on failure.

p13 = ggplot(failurefull\_complete, aes(x = measurement\_6, y = failure))+ geom\_boxplot()  
p14 = ggplot(failurefull\_complete, aes(x = measurement\_7, y = failure))+ geom\_boxplot()  
p15 = ggplot(failurefull\_complete, aes(x = measurement\_8, y = failure))+ geom\_boxplot()  
p16 = ggplot(failurefull\_complete, aes(x = measurement\_9, y = failure))+ geom\_boxplot()  
grid.arrange(p13, p14, p15, p16, ncol = 2)



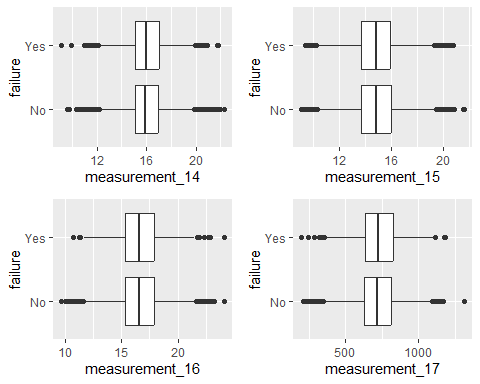
#No discernible predictive value of these measurements on failure.

p17 = ggplot(failurefull\_complete, aes(x = measurement\_10, y = failure))+ geom\_boxplot()  
p18 = ggplot(failurefull\_complete, aes(x = measurement\_11, y = failure))+ geom\_boxplot()  
p19 = ggplot(failurefull\_complete, aes(x = measurement\_12, y = failure))+ geom\_boxplot()  
p20 = ggplot(failurefull\_complete, aes(x = measurement\_13, y = failure))+ geom\_boxplot()  
grid.arrange(p17, p18, p19, p20, ncol=2)



#No discernible predictive value of these measurements on failure.

p21 = ggplot(failurefull\_complete, aes(x = measurement\_14, y = failure))+ geom\_boxplot()  
p22 = ggplot(failurefull\_complete, aes(x = measurement\_15, y = failure))+ geom\_boxplot()  
p23 = ggplot(failurefull\_complete, aes(x = measurement\_16, y = failure))+ geom\_boxplot()  
p24 = ggplot(failurefull\_complete, aes(x = measurement\_17, y = failure))+ geom\_boxplot()  
grid.arrange(p21, p22, p23, p24, ncol=2)



# There is a slight difference in the mean values of measurement\_17 for products that failed vs. products that did not fail.

**Loading appears to be the strongest predictor of failure. Attributes 0, 2 & 3 appear to be weak predictors of failure. Measurements 5, 8 and 17 are the only measurement that exhibit a slight discernible difference in mean values for product failure.**