library(tidyverse)

## ── Attaching packages ─────────────────────────────────────── tidyverse 1.3.1 ──

## ✔ ggplot2 3.3.6 ✔ purrr 0.3.4  
## ✔ tibble 3.1.7 ✔ dplyr 1.0.9  
## ✔ tidyr 1.2.0 ✔ stringr 1.4.0  
## ✔ readr 2.1.2 ✔ forcats 0.5.1

## ── Conflicts ────────────────────────────────────────── tidyverse\_conflicts() ──  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ dplyr::lag() masks stats::lag()

library(tidymodels)

## ── Attaching packages ────────────────────────────────────── tidymodels 0.2.0 ──

## ✔ broom 0.8.0 ✔ rsample 0.1.1  
## ✔ dials 0.1.1 ✔ tune 0.2.0  
## ✔ infer 1.0.0 ✔ workflows 0.2.6  
## ✔ modeldata 0.1.1 ✔ workflowsets 0.2.1  
## ✔ parsnip 0.2.1 ✔ yardstick 0.0.9  
## ✔ recipes 0.2.0

## ── Conflicts ───────────────────────────────────────── tidymodels\_conflicts() ──  
## ✖ scales::discard() masks purrr::discard()  
## ✖ dplyr::filter() masks stats::filter()  
## ✖ recipes::fixed() masks stringr::fixed()  
## ✖ dplyr::lag() masks stats::lag()  
## ✖ yardstick::spec() masks readr::spec()  
## ✖ recipes::step() masks stats::step()  
## • Search for functions across packages at https://www.tidymodels.org/find/

library(GGally) #ggcorr and ggpairs

## Registered S3 method overwritten by 'GGally':  
## method from   
## +.gg ggplot2

library(ggcorrplot) #correlation plot alternative  
library(gridExtra) #create grids of plots

##   
## Attaching package: 'gridExtra'

## The following object is masked from 'package:dplyr':  
##   
## combine

ameso = read\_csv("ames\_student.csv")

## Rows: 2053 Columns: 81  
## ── Column specification ────────────────────────────────────────────────────────  
## Delimiter: ","  
## chr (47): MS\_SubClass, MS\_Zoning, Street, Alley, Lot\_Shape, Land\_Contour, Ut...  
## dbl (34): Lot\_Frontage, Lot\_Area, Year\_Built, Year\_Remod\_Add, Mas\_Vnr\_Area, ...  
##   
## 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.

str(ameso)

## spec\_tbl\_df [2,053 × 81] (S3: spec\_tbl\_df/tbl\_df/tbl/data.frame)  
## $ MS\_SubClass : chr [1:2053] "One\_Story\_1946\_and\_Newer\_All\_Styles" "One\_Story\_1946\_and\_Newer\_All\_Styles" "One\_Story\_1946\_and\_Newer\_All\_Styles" "One\_Story\_1946\_and\_Newer\_All\_Styles" ...  
## $ MS\_Zoning : chr [1:2053] "Residential\_Low\_Density" "Residential\_High\_Density" "Residential\_Low\_Density" "Residential\_Low\_Density" ...  
## $ Lot\_Frontage : num [1:2053] 141 80 81 93 74 78 43 39 0 85 ...  
## $ Lot\_Area : num [1:2053] 31770 11622 14267 11160 13830 ...  
## $ Street : chr [1:2053] "Pave" "Pave" "Pave" "Pave" ...  
## $ Alley : chr [1:2053] "No\_Alley\_Access" "No\_Alley\_Access" "No\_Alley\_Access" "No\_Alley\_Access" ...  
## $ Lot\_Shape : chr [1:2053] "Slightly\_Irregular" "Regular" "Slightly\_Irregular" "Regular" ...  
## $ Land\_Contour : chr [1:2053] "Lvl" "Lvl" "Lvl" "Lvl" ...  
## $ Utilities : chr [1:2053] "AllPub" "AllPub" "AllPub" "AllPub" ...  
## $ Lot\_Config : chr [1:2053] "Corner" "Inside" "Corner" "Corner" ...  
## $ Land\_Slope : chr [1:2053] "Gtl" "Gtl" "Gtl" "Gtl" ...  
## $ Neighborhood : chr [1:2053] "North\_Ames" "North\_Ames" "North\_Ames" "North\_Ames" ...  
## $ Condition\_1 : chr [1:2053] "Norm" "Feedr" "Norm" "Norm" ...  
## $ Condition\_2 : chr [1:2053] "Norm" "Norm" "Norm" "Norm" ...  
## $ Bldg\_Type : chr [1:2053] "OneFam" "OneFam" "OneFam" "OneFam" ...  
## $ House\_Style : chr [1:2053] "One\_Story" "One\_Story" "One\_Story" "One\_Story" ...  
## $ Overall\_Qual : chr [1:2053] "Above\_Average" "Average" "Above\_Average" "Good" ...  
## $ Overall\_Cond : chr [1:2053] "Average" "Above\_Average" "Above\_Average" "Average" ...  
## $ Year\_Built : num [1:2053] 1960 1961 1958 1968 1997 ...  
## $ Year\_Remod\_Add : num [1:2053] 1960 1961 1958 1968 1998 ...  
## $ Roof\_Style : chr [1:2053] "Hip" "Gable" "Hip" "Hip" ...  
## $ Roof\_Matl : chr [1:2053] "CompShg" "CompShg" "CompShg" "CompShg" ...  
## $ Exterior\_1st : chr [1:2053] "BrkFace" "VinylSd" "Wd Sdng" "BrkFace" ...  
## $ Exterior\_2nd : chr [1:2053] "Plywood" "VinylSd" "Wd Sdng" "BrkFace" ...  
## $ Mas\_Vnr\_Type : chr [1:2053] "Stone" "None" "BrkFace" "None" ...  
## $ Mas\_Vnr\_Area : num [1:2053] 112 0 108 0 0 20 0 0 0 0 ...  
## $ Exter\_Qual : chr [1:2053] "Typical" "Typical" "Typical" "Good" ...  
## $ Exter\_Cond : chr [1:2053] "Typical" "Typical" "Typical" "Typical" ...  
## $ Foundation : chr [1:2053] "CBlock" "CBlock" "CBlock" "CBlock" ...  
## $ Bsmt\_Qual : chr [1:2053] "Typical" "Typical" "Typical" "Typical" ...  
## $ Bsmt\_Cond : chr [1:2053] "Good" "Typical" "Typical" "Typical" ...  
## $ Bsmt\_Exposure : chr [1:2053] "Gd" "No" "No" "No" ...  
## $ BsmtFin\_Type\_1 : chr [1:2053] "BLQ" "Rec" "ALQ" "ALQ" ...  
## $ BsmtFin\_SF\_1 : num [1:2053] 2 6 1 1 3 3 1 3 1 3 ...  
## $ BsmtFin\_Type\_2 : chr [1:2053] "Unf" "LwQ" "Unf" "Unf" ...  
## $ BsmtFin\_SF\_2 : num [1:2053] 0 144 0 0 0 0 0 0 0 0 ...  
## $ Bsmt\_Unf\_SF : num [1:2053] 441 270 406 1045 137 ...  
## $ Total\_Bsmt\_SF : num [1:2053] 1080 882 1329 2110 928 ...  
## $ Heating : chr [1:2053] "GasA" "GasA" "GasA" "GasA" ...  
## $ Heating\_QC : chr [1:2053] "Fair" "Typical" "Typical" "Excellent" ...  
## $ Central\_Air : chr [1:2053] "Y" "Y" "Y" "Y" ...  
## $ Electrical : chr [1:2053] "SBrkr" "SBrkr" "SBrkr" "SBrkr" ...  
## $ First\_Flr\_SF : num [1:2053] 1656 896 1329 2110 928 ...  
## $ Second\_Flr\_SF : num [1:2053] 0 0 0 0 701 678 0 0 0 0 ...  
## $ Low\_Qual\_Fin\_SF : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Gr\_Liv\_Area : num [1:2053] 1656 896 1329 2110 1629 ...  
## $ Bsmt\_Full\_Bath : num [1:2053] 1 0 0 1 0 0 0 1 1 1 ...  
## $ Bsmt\_Half\_Bath : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Full\_Bath : num [1:2053] 1 1 1 2 2 2 2 2 2 1 ...  
## $ Half\_Bath : num [1:2053] 0 0 1 1 1 1 0 0 0 1 ...  
## $ Bedroom\_AbvGr : num [1:2053] 3 2 3 3 3 3 2 2 3 2 ...  
## $ Kitchen\_AbvGr : num [1:2053] 1 1 1 1 1 1 1 1 1 1 ...  
## $ Kitchen\_Qual : chr [1:2053] "Typical" "Typical" "Good" "Excellent" ...  
## $ TotRms\_AbvGrd : num [1:2053] 7 5 6 8 6 7 5 5 6 5 ...  
## $ Functional : chr [1:2053] "Typ" "Typ" "Typ" "Typ" ...  
## $ Fireplaces : num [1:2053] 2 0 0 2 1 1 0 1 0 1 ...  
## $ Fireplace\_Qu : chr [1:2053] "Good" "No\_Fireplace" "No\_Fireplace" "Typical" ...  
## $ Garage\_Type : chr [1:2053] "Attchd" "Attchd" "Attchd" "Attchd" ...  
## $ Garage\_Finish : chr [1:2053] "Fin" "Unf" "Unf" "Fin" ...  
## $ Garage\_Cars : num [1:2053] 2 1 1 2 2 2 2 2 2 2 ...  
## $ Garage\_Area : num [1:2053] 528 730 312 522 482 470 506 608 420 506 ...  
## $ Garage\_Qual : chr [1:2053] "Typical" "Typical" "Typical" "Typical" ...  
## $ Garage\_Cond : chr [1:2053] "Typical" "Typical" "Typical" "Typical" ...  
## $ Paved\_Drive : chr [1:2053] "Partial\_Pavement" "Paved" "Paved" "Paved" ...  
## $ Wood\_Deck\_SF : num [1:2053] 210 140 393 0 212 360 0 237 483 192 ...  
## $ Open\_Porch\_SF : num [1:2053] 62 0 36 0 34 36 82 152 21 0 ...  
## $ Enclosed\_Porch : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Three\_season\_porch: num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Screen\_Porch : num [1:2053] 0 120 0 0 0 0 144 0 0 0 ...  
## $ Pool\_Area : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Pool\_QC : chr [1:2053] "No\_Pool" "No\_Pool" "No\_Pool" "No\_Pool" ...  
## $ Fence : chr [1:2053] "No\_Fence" "Minimum\_Privacy" "No\_Fence" "No\_Fence" ...  
## $ Misc\_Feature : chr [1:2053] "None" "None" "Gar2" "None" ...  
## $ Misc\_Val : num [1:2053] 0 0 12500 0 0 0 0 0 500 0 ...  
## $ Mo\_Sold : num [1:2053] 5 6 6 4 3 6 1 3 3 2 ...  
## $ Year\_Sold : num [1:2053] 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...  
## $ Sale\_Type : chr [1:2053] "WD" "WD" "WD" "WD" ...  
## $ Sale\_Condition : chr [1:2053] "Normal" "Normal" "Normal" "Normal" ...  
## $ Longitude : num [1:2053] -93.6 -93.6 -93.6 -93.6 -93.6 ...  
## $ Latitude : num [1:2053] 42.1 42.1 42.1 42.1 42.1 ...  
## $ Above\_Median : chr [1:2053] "Yes" "No" "Yes" "Yes" ...  
## - attr(\*, "spec")=  
## .. cols(  
## .. MS\_SubClass = col\_character(),  
## .. MS\_Zoning = col\_character(),  
## .. Lot\_Frontage = col\_double(),  
## .. Lot\_Area = col\_double(),  
## .. Street = col\_character(),  
## .. Alley = col\_character(),  
## .. Lot\_Shape = col\_character(),  
## .. Land\_Contour = col\_character(),  
## .. Utilities = col\_character(),  
## .. Lot\_Config = col\_character(),  
## .. Land\_Slope = col\_character(),  
## .. Neighborhood = col\_character(),  
## .. Condition\_1 = col\_character(),  
## .. Condition\_2 = col\_character(),  
## .. Bldg\_Type = col\_character(),  
## .. House\_Style = col\_character(),  
## .. Overall\_Qual = col\_character(),  
## .. Overall\_Cond = col\_character(),  
## .. Year\_Built = col\_double(),  
## .. Year\_Remod\_Add = col\_double(),  
## .. Roof\_Style = col\_character(),  
## .. Roof\_Matl = col\_character(),  
## .. Exterior\_1st = col\_character(),  
## .. Exterior\_2nd = col\_character(),  
## .. Mas\_Vnr\_Type = col\_character(),  
## .. Mas\_Vnr\_Area = col\_double(),  
## .. Exter\_Qual = col\_character(),  
## .. Exter\_Cond = col\_character(),  
## .. Foundation = col\_character(),  
## .. Bsmt\_Qual = col\_character(),  
## .. Bsmt\_Cond = col\_character(),  
## .. Bsmt\_Exposure = col\_character(),  
## .. BsmtFin\_Type\_1 = col\_character(),  
## .. BsmtFin\_SF\_1 = col\_double(),  
## .. BsmtFin\_Type\_2 = col\_character(),  
## .. BsmtFin\_SF\_2 = col\_double(),  
## .. Bsmt\_Unf\_SF = col\_double(),  
## .. Total\_Bsmt\_SF = col\_double(),  
## .. Heating = col\_character(),  
## .. Heating\_QC = col\_character(),  
## .. Central\_Air = col\_character(),  
## .. Electrical = col\_character(),  
## .. First\_Flr\_SF = col\_double(),  
## .. Second\_Flr\_SF = col\_double(),  
## .. Low\_Qual\_Fin\_SF = col\_double(),  
## .. Gr\_Liv\_Area = col\_double(),  
## .. Bsmt\_Full\_Bath = col\_double(),  
## .. Bsmt\_Half\_Bath = col\_double(),  
## .. Full\_Bath = col\_double(),  
## .. Half\_Bath = col\_double(),  
## .. Bedroom\_AbvGr = col\_double(),  
## .. Kitchen\_AbvGr = col\_double(),  
## .. Kitchen\_Qual = col\_character(),  
## .. TotRms\_AbvGrd = col\_double(),  
## .. Functional = col\_character(),  
## .. Fireplaces = col\_double(),  
## .. Fireplace\_Qu = col\_character(),  
## .. Garage\_Type = col\_character(),  
## .. Garage\_Finish = col\_character(),  
## .. Garage\_Cars = col\_double(),  
## .. Garage\_Area = col\_double(),  
## .. Garage\_Qual = col\_character(),  
## .. Garage\_Cond = col\_character(),  
## .. Paved\_Drive = col\_character(),  
## .. Wood\_Deck\_SF = col\_double(),  
## .. Open\_Porch\_SF = col\_double(),  
## .. Enclosed\_Porch = col\_double(),  
## .. Three\_season\_porch = col\_double(),  
## .. Screen\_Porch = col\_double(),  
## .. Pool\_Area = col\_double(),  
## .. Pool\_QC = col\_character(),  
## .. Fence = col\_character(),  
## .. Misc\_Feature = col\_character(),  
## .. Misc\_Val = col\_double(),  
## .. Mo\_Sold = col\_double(),  
## .. Year\_Sold = col\_double(),  
## .. Sale\_Type = col\_character(),  
## .. Sale\_Condition = col\_character(),  
## .. Longitude = col\_double(),  
## .. Latitude = col\_double(),  
## .. Above\_Median = col\_character()  
## .. )  
## - attr(\*, "problems")=<externalptr>

summary(ameso)

## MS\_SubClass MS\_Zoning Lot\_Frontage Lot\_Area   
## Length:2053 Length:2053 Min. : 0.00 Min. : 1300   
## Class :character Class :character 1st Qu.: 43.00 1st Qu.: 7500   
## Mode :character Mode :character Median : 62.00 Median : 9548   
## Mean : 57.38 Mean : 10258   
## 3rd Qu.: 78.00 3rd Qu.: 11600   
## Max. :313.00 Max. :215245   
## Street Alley Lot\_Shape Land\_Contour   
## Length:2053 Length:2053 Length:2053 Length:2053   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## Utilities Lot\_Config Land\_Slope Neighborhood   
## Length:2053 Length:2053 Length:2053 Length:2053   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## Condition\_1 Condition\_2 Bldg\_Type House\_Style   
## Length:2053 Length:2053 Length:2053 Length:2053   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## Overall\_Qual Overall\_Cond Year\_Built Year\_Remod\_Add  
## Length:2053 Length:2053 Min. :1875 Min. :1950   
## Class :character Class :character 1st Qu.:1953 1st Qu.:1965   
## Mode :character Mode :character Median :1972 Median :1993   
## Mean :1971 Mean :1984   
## 3rd Qu.:2000 3rd Qu.:2004   
## Max. :2010 Max. :2010   
## Roof\_Style Roof\_Matl Exterior\_1st Exterior\_2nd   
## Length:2053 Length:2053 Length:2053 Length:2053   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## Mas\_Vnr\_Type Mas\_Vnr\_Area Exter\_Qual Exter\_Cond   
## Length:2053 Min. : 0.0 Length:2053 Length:2053   
## Class :character 1st Qu.: 0.0 Class :character Class :character   
## Mode :character Median : 0.0 Mode :character Mode :character   
## Mean : 103.8   
## 3rd Qu.: 164.0   
## Max. :1600.0   
## Foundation Bsmt\_Qual Bsmt\_Cond Bsmt\_Exposure   
## Length:2053 Length:2053 Length:2053 Length:2053   
## Class :character Class :character Class :character Class :character   
## Mode :character Mode :character Mode :character Mode :character   
##   
##   
##   
## BsmtFin\_Type\_1 BsmtFin\_SF\_1 BsmtFin\_Type\_2 BsmtFin\_SF\_2   
## Length:2053 Min. :1.00 Length:2053 Min. : 0.00   
## Class :character 1st Qu.:3.00 Class :character 1st Qu.: 0.00   
## Mode :character Median :3.00 Mode :character Median : 0.00   
## Mean :4.21 Mean : 52.57   
## 3rd Qu.:7.00 3rd Qu.: 0.00   
## Max. :7.00 Max. :1526.00   
## Bsmt\_Unf\_SF Total\_Bsmt\_SF Heating Heating\_QC   
## Min. : 0.0 Min. : 0 Length:2053 Length:2053   
## 1st Qu.: 226.0 1st Qu.: 793 Class :character Class :character   
## Median : 460.0 Median : 988 Mode :character Mode :character   
## Mean : 561.2 Mean :1055   
## 3rd Qu.: 801.0 3rd Qu.:1304   
## Max. :2336.0 Max. :5095   
## Central\_Air Electrical First\_Flr\_SF Second\_Flr\_SF   
## Length:2053 Length:2053 Min. : 432 Min. : 0.0   
## Class :character Class :character 1st Qu.: 882 1st Qu.: 0.0   
## Mode :character Mode :character Median :1088 Median : 0.0   
## Mean :1168 Mean : 326.1   
## 3rd Qu.:1402 3rd Qu.: 701.0   
## Max. :5095 Max. :1862.0   
## Low\_Qual\_Fin\_SF Gr\_Liv\_Area Bsmt\_Full\_Bath Bsmt\_Half\_Bath   
## Min. : 0.000 Min. : 480 Min. :0.0000 Min. :0.00000   
## 1st Qu.: 0.000 1st Qu.:1137 1st Qu.:0.0000 1st Qu.:0.00000   
## Median : 0.000 Median :1447 Median :0.0000 Median :0.00000   
## Mean : 4.973 Mean :1499 Mean :0.4301 Mean :0.05796   
## 3rd Qu.: 0.000 3rd Qu.:1737 3rd Qu.:1.0000 3rd Qu.:0.00000   
## Max. :1064.000 Max. :5095 Max. :3.0000 Max. :2.00000   
## Full\_Bath Half\_Bath Bedroom\_AbvGr Kitchen\_AbvGr   
## Min. :0.000 Min. :0.0000 Min. :0.000 Min. :1.000   
## 1st Qu.:1.000 1st Qu.:0.0000 1st Qu.:2.000 1st Qu.:1.000   
## Median :2.000 Median :0.0000 Median :3.000 Median :1.000   
## Mean :1.564 Mean :0.3751 Mean :2.855 Mean :1.047   
## 3rd Qu.:2.000 3rd Qu.:1.0000 3rd Qu.:3.000 3rd Qu.:1.000   
## Max. :4.000 Max. :2.0000 Max. :6.000 Max. :3.000   
## Kitchen\_Qual TotRms\_AbvGrd Functional Fireplaces   
## Length:2053 Min. : 3.000 Length:2053 Min. :0.000   
## Class :character 1st Qu.: 5.000 Class :character 1st Qu.:0.000   
## Mode :character Median : 6.000 Mode :character Median :1.000   
## Mean : 6.442 Mean :0.603   
## 3rd Qu.: 7.000 3rd Qu.:1.000   
## Max. :15.000 Max. :4.000   
## Fireplace\_Qu Garage\_Type Garage\_Finish Garage\_Cars   
## Length:2053 Length:2053 Length:2053 Min. :0.000   
## Class :character Class :character Class :character 1st Qu.:1.000   
## Mode :character Mode :character Mode :character Median :2.000   
## Mean :1.774   
## 3rd Qu.:2.000   
## Max. :5.000   
## Garage\_Area Garage\_Qual Garage\_Cond Paved\_Drive   
## Min. : 0 Length:2053 Length:2053 Length:2053   
## 1st Qu.: 320 Class :character Class :character Class :character   
## Median : 478 Mode :character Mode :character Mode :character   
## Mean : 472   
## 3rd Qu.: 576   
## Max. :1488   
## Wood\_Deck\_SF Open\_Porch\_SF Enclosed\_Porch Three\_season\_porch  
## Min. : 0.00 Min. : 0.00 Min. : 0.00 Min. : 0.000   
## 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: 0.00 1st Qu.: 0.000   
## Median : 0.00 Median : 27.00 Median : 0.00 Median : 0.000   
## Mean : 93.52 Mean : 48.17 Mean : 23.02 Mean : 2.799   
## 3rd Qu.: 168.00 3rd Qu.: 72.00 3rd Qu.: 0.00 3rd Qu.: 0.000   
## Max. :1424.00 Max. :742.00 Max. :584.00 Max. :407.000   
## Screen\_Porch Pool\_Area Pool\_QC Fence   
## Min. : 0.00 Min. : 0.000 Length:2053 Length:2053   
## 1st Qu.: 0.00 1st Qu.: 0.000 Class :character Class :character   
## Median : 0.00 Median : 0.000 Mode :character Mode :character   
## Mean : 16.68 Mean : 1.339   
## 3rd Qu.: 0.00 3rd Qu.: 0.000   
## Max. :576.00 Max. :800.000   
## Misc\_Feature Misc\_Val Mo\_Sold Year\_Sold   
## Length:2053 Min. : 0.00 Min. : 1.000 Min. :2006   
## Class :character 1st Qu.: 0.00 1st Qu.: 4.000 1st Qu.:2007   
## Mode :character Median : 0.00 Median : 6.000 Median :2008   
## Mean : 60.12 Mean : 6.189 Mean :2008   
## 3rd Qu.: 0.00 3rd Qu.: 8.000 3rd Qu.:2009   
## Max. :17000.00 Max. :12.000 Max. :2010   
## Sale\_Type Sale\_Condition Longitude Latitude   
## Length:2053 Length:2053 Min. :-93.69 Min. :41.99   
## Class :character Class :character 1st Qu.:-93.66 1st Qu.:42.02   
## Mode :character Mode :character Median :-93.64 Median :42.03   
## Mean :-93.64 Mean :42.03   
## 3rd Qu.:-93.62 3rd Qu.:42.05   
## Max. :-93.58 Max. :42.06   
## Above\_Median   
## Length:2053   
## Class :character   
## Mode :character   
##   
##   
##

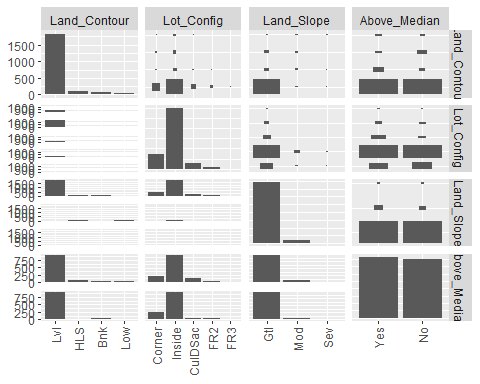
ames = ameso %>% mutate(MS\_SubClass = as\_factor(MS\_SubClass)) %>%  
 mutate(MS\_Zoning = as\_factor(MS\_Zoning)) %>%  
 mutate(Street = as\_factor(Street)) %>%  
 mutate(Alley = as\_factor(Alley)) %>%  
 mutate(Lot\_Shape = as\_factor(Lot\_Shape)) %>%  
 mutate(Land\_Contour = as\_factor(Land\_Contour)) %>%  
 mutate(Utilities = as\_factor(Utilities)) %>%  
 mutate(Lot\_Config = as\_factor(Lot\_Config)) %>%  
 mutate(Land\_Slope = as\_factor(Land\_Slope)) %>%  
 mutate(Neighborhood = as\_factor(Neighborhood)) %>%  
 mutate(Condition\_1 = as\_factor(Condition\_1)) %>%  
 mutate(Condition\_2 = as\_factor(Condition\_2)) %>%  
 mutate(Bldg\_Type = as\_factor(Bldg\_Type)) %>%  
 mutate(House\_Style = as\_factor(House\_Style)) %>%  
 mutate(Overall\_Qual = as\_factor(Overall\_Qual)) %>%  
 mutate(Overall\_Cond = as\_factor(Overall\_Cond)) %>%  
 mutate(Roof\_Style = as\_factor(Roof\_Style)) %>%  
 mutate(Roof\_Matl = as\_factor(Roof\_Matl)) %>%  
 mutate(Exterior\_1st = as\_factor(Exterior\_1st)) %>%  
 mutate(Exterior\_2nd = as\_factor(Exterior\_2nd)) %>%  
 mutate(Mas\_Vnr\_Type = as\_factor(Mas\_Vnr\_Type)) %>%  
 mutate(Exter\_Qual = as\_factor(Exter\_Qual)) %>%  
 mutate(Exter\_Cond = as\_factor(Exter\_Cond)) %>%  
 mutate(Foundation = as\_factor(Foundation)) %>%  
 mutate(Bsmt\_Qual = as\_factor(Bsmt\_Qual)) %>%  
 mutate(Bsmt\_Cond = as\_factor(Bsmt\_Cond)) %>%  
 mutate(Bsmt\_Exposure = as\_factor(Bsmt\_Exposure)) %>%  
 mutate(BsmtFin\_Type\_1 = as\_factor(BsmtFin\_Type\_1)) %>%  
 mutate(BsmtFin\_Type\_2 = as\_factor(BsmtFin\_Type\_2)) %>%  
 mutate(Heating = as\_factor(Heating)) %>%  
 mutate(Heating\_QC = as\_factor(Heating\_QC)) %>%  
 mutate(Central\_Air = as\_factor(Central\_Air)) %>%  
 mutate(Electrical = as\_factor(Electrical)) %>%  
 mutate(Kitchen\_Qual = as\_factor(Kitchen\_Qual)) %>%  
 mutate(Functional = as\_factor(Functional)) %>%  
 mutate(Fireplace\_Qu = as\_factor(Fireplace\_Qu)) %>%  
 mutate(Garage\_Type = as\_factor(Garage\_Type)) %>%  
 mutate(Garage\_Finish = as\_factor(Garage\_Finish)) %>%  
 mutate(Garage\_Cond = as\_factor(Garage\_Cond)) %>%  
 mutate(Garage\_Qual = as\_factor(Garage\_Qual)) %>%  
 mutate(Paved\_Drive = as\_factor(Paved\_Drive)) %>%  
 mutate(Pool\_QC = as\_factor(Pool\_QC)) %>%  
 mutate(Fence = as\_factor(Fence)) %>%  
 mutate(Misc\_Feature = as\_factor(Misc\_Feature)) %>%  
 mutate(Sale\_Type = as\_factor(Sale\_Type)) %>%  
 mutate(Sale\_Condition = as\_factor(Sale\_Condition)) %>%  
 mutate(Above\_Median = as\_factor(Above\_Median))  
 # mutate(Above\_Median = fct\_recode(Above\_Median, "0" = "No", "1" = "Yes" ))  
   
   
   
# mutate(Survived = fct\_recode(Survived, "No" = "0", "Yes" = "1" )) %>%  
# mutate(Pclass = as\_factor(Pclass)) %>% mutate(Sex = as\_factor(Sex))  
str(ames)

## tibble [2,053 × 81] (S3: tbl\_df/tbl/data.frame)  
## $ MS\_SubClass : Factor w/ 16 levels "One\_Story\_1946\_and\_Newer\_All\_Styles",..: 1 1 1 1 2 2 3 3 1 1 ...  
## $ MS\_Zoning : Factor w/ 7 levels "Residential\_Low\_Density",..: 1 2 1 1 1 1 1 1 1 1 ...  
## $ Lot\_Frontage : num [1:2053] 141 80 81 93 74 78 43 39 0 85 ...  
## $ Lot\_Area : num [1:2053] 31770 11622 14267 11160 13830 ...  
## $ Street : Factor w/ 2 levels "Pave","Grvl": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Alley : Factor w/ 3 levels "No\_Alley\_Access",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Lot\_Shape : Factor w/ 4 levels "Slightly\_Irregular",..: 1 2 1 2 1 1 1 1 1 2 ...  
## $ Land\_Contour : Factor w/ 4 levels "Lvl","HLS","Bnk",..: 1 1 1 1 1 1 2 1 1 1 ...  
## $ Utilities : Factor w/ 2 levels "AllPub","NoSewr": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Lot\_Config : Factor w/ 5 levels "Corner","Inside",..: 1 2 1 1 2 2 2 2 2 2 ...  
## $ Land\_Slope : Factor w/ 3 levels "Gtl","Mod","Sev": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Neighborhood : Factor w/ 28 levels "North\_Ames","Gilbert",..: 1 1 1 1 2 2 3 3 2 2 ...  
## $ Condition\_1 : Factor w/ 9 levels "Norm","Feedr",..: 1 2 1 1 1 1 1 1 1 1 ...  
## $ Condition\_2 : Factor w/ 8 levels "Norm","Feedr",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Bldg\_Type : Factor w/ 5 levels "OneFam","TwnhsE",..: 1 1 1 1 1 1 2 2 1 1 ...  
## $ House\_Style : Factor w/ 8 levels "One\_Story","Two\_Story",..: 1 1 1 1 2 2 1 1 1 1 ...  
## $ Overall\_Qual : Factor w/ 10 levels "Above\_Average",..: 1 2 1 3 2 1 4 4 1 3 ...  
## $ Overall\_Cond : Factor w/ 9 levels "Average","Above\_Average",..: 1 2 2 1 1 2 1 1 3 1 ...  
## $ Year\_Built : num [1:2053] 1960 1961 1958 1968 1997 ...  
## $ Year\_Remod\_Add : num [1:2053] 1960 1961 1958 1968 1998 ...  
## $ Roof\_Style : Factor w/ 6 levels "Hip","Gable",..: 1 2 1 1 2 2 2 2 2 2 ...  
## $ Roof\_Matl : Factor w/ 6 levels "CompShg","WdShake",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Exterior\_1st : Factor w/ 16 levels "BrkFace","VinylSd",..: 1 2 3 1 2 2 4 5 4 4 ...  
## $ Exterior\_2nd : Factor w/ 17 levels "Plywood","VinylSd",..: 1 2 3 4 2 2 5 6 5 5 ...  
## $ Mas\_Vnr\_Type : Factor w/ 5 levels "Stone","None",..: 1 2 3 2 2 3 2 2 2 2 ...  
## $ Mas\_Vnr\_Area : num [1:2053] 112 0 108 0 0 20 0 0 0 0 ...  
## $ Exter\_Qual : Factor w/ 4 levels "Typical","Good",..: 1 1 1 2 1 1 2 2 1 1 ...  
## $ Exter\_Cond : Factor w/ 5 levels "Typical","Good",..: 1 1 1 1 1 1 1 1 2 1 ...  
## $ Foundation : Factor w/ 6 levels "CBlock","PConc",..: 1 1 1 1 2 2 2 2 2 2 ...  
## $ Bsmt\_Qual : Factor w/ 6 levels "Typical","Good",..: 1 1 1 1 2 1 2 2 2 2 ...  
## $ Bsmt\_Cond : Factor w/ 6 levels "Good","Typical",..: 1 2 2 2 2 2 2 2 2 2 ...  
## $ Bsmt\_Exposure : Factor w/ 5 levels "Gd","No","Av",..: 1 2 2 2 2 2 2 2 2 1 ...  
## $ BsmtFin\_Type\_1 : Factor w/ 7 levels "BLQ","Rec","ALQ",..: 1 2 3 3 4 4 3 4 3 4 ...  
## $ BsmtFin\_SF\_1 : num [1:2053] 2 6 1 1 3 3 1 3 1 3 ...  
## $ BsmtFin\_Type\_2 : Factor w/ 7 levels "Unf","LwQ","BLQ",..: 1 2 1 1 1 1 1 1 1 1 ...  
## $ BsmtFin\_SF\_2 : num [1:2053] 0 144 0 0 0 0 0 0 0 0 ...  
## $ Bsmt\_Unf\_SF : num [1:2053] 441 270 406 1045 137 ...  
## $ Total\_Bsmt\_SF : num [1:2053] 1080 882 1329 2110 928 ...  
## $ Heating : Factor w/ 6 levels "GasA","GasW",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Heating\_QC : Factor w/ 5 levels "Fair","Typical",..: 1 2 2 3 4 3 3 3 3 4 ...  
## $ Central\_Air : Factor w/ 2 levels "Y","N": 1 1 1 1 1 1 1 1 1 1 ...  
## $ Electrical : Factor w/ 5 levels "SBrkr","FuseA",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ First\_Flr\_SF : num [1:2053] 1656 896 1329 2110 928 ...  
## $ Second\_Flr\_SF : num [1:2053] 0 0 0 0 701 678 0 0 0 0 ...  
## $ Low\_Qual\_Fin\_SF : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Gr\_Liv\_Area : num [1:2053] 1656 896 1329 2110 1629 ...  
## $ Bsmt\_Full\_Bath : num [1:2053] 1 0 0 1 0 0 0 1 1 1 ...  
## $ Bsmt\_Half\_Bath : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Full\_Bath : num [1:2053] 1 1 1 2 2 2 2 2 2 1 ...  
## $ Half\_Bath : num [1:2053] 0 0 1 1 1 1 0 0 0 1 ...  
## $ Bedroom\_AbvGr : num [1:2053] 3 2 3 3 3 3 2 2 3 2 ...  
## $ Kitchen\_AbvGr : num [1:2053] 1 1 1 1 1 1 1 1 1 1 ...  
## $ Kitchen\_Qual : Factor w/ 5 levels "Typical","Good",..: 1 1 2 3 1 2 2 2 1 2 ...  
## $ TotRms\_AbvGrd : num [1:2053] 7 5 6 8 6 7 5 5 6 5 ...  
## $ Functional : Factor w/ 8 levels "Typ","Mod","Min2",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Fireplaces : num [1:2053] 2 0 0 2 1 1 0 1 0 1 ...  
## $ Fireplace\_Qu : Factor w/ 6 levels "Good","No\_Fireplace",..: 1 2 2 3 3 1 2 3 2 4 ...  
## $ Garage\_Type : Factor w/ 7 levels "Attchd","BuiltIn",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Garage\_Finish : Factor w/ 4 levels "Fin","Unf","RFn",..: 1 2 2 1 1 1 3 3 1 2 ...  
## $ Garage\_Cars : num [1:2053] 2 1 1 2 2 2 2 2 2 2 ...  
## $ Garage\_Area : num [1:2053] 528 730 312 522 482 470 506 608 420 506 ...  
## $ Garage\_Qual : Factor w/ 6 levels "Typical","No\_Garage",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Garage\_Cond : Factor w/ 6 levels "Typical","No\_Garage",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Paved\_Drive : Factor w/ 3 levels "Partial\_Pavement",..: 1 2 2 2 2 2 2 2 2 2 ...  
## $ Wood\_Deck\_SF : num [1:2053] 210 140 393 0 212 360 0 237 483 192 ...  
## $ Open\_Porch\_SF : num [1:2053] 62 0 36 0 34 36 82 152 21 0 ...  
## $ Enclosed\_Porch : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Three\_season\_porch: num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Screen\_Porch : num [1:2053] 0 120 0 0 0 0 144 0 0 0 ...  
## $ Pool\_Area : num [1:2053] 0 0 0 0 0 0 0 0 0 0 ...  
## $ Pool\_QC : Factor w/ 5 levels "No\_Pool","Excellent",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Fence : Factor w/ 5 levels "No\_Fence","Minimum\_Privacy",..: 1 2 1 1 2 1 1 1 3 1 ...  
## $ Misc\_Feature : Factor w/ 5 levels "None","Gar2",..: 1 1 2 1 1 1 1 1 3 1 ...  
## $ Misc\_Val : num [1:2053] 0 0 12500 0 0 0 0 0 500 0 ...  
## $ Mo\_Sold : num [1:2053] 5 6 6 4 3 6 1 3 3 2 ...  
## $ Year\_Sold : num [1:2053] 2010 2010 2010 2010 2010 2010 2010 2010 2010 2010 ...  
## $ Sale\_Type : Factor w/ 10 levels "WD","New","COD",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Sale\_Condition : Factor w/ 6 levels "Normal","Partial",..: 1 1 1 1 1 1 1 1 1 1 ...  
## $ Longitude : num [1:2053] -93.6 -93.6 -93.6 -93.6 -93.6 ...  
## $ Latitude : num [1:2053] 42.1 42.1 42.1 42.1 42.1 ...  
## $ Above\_Median : Factor w/ 2 levels "Yes","No": 1 2 1 1 1 1 1 1 1 1 ...

should split columns into groups to make more managable - date and time -location -land -internal & external features -basement -General house stats?

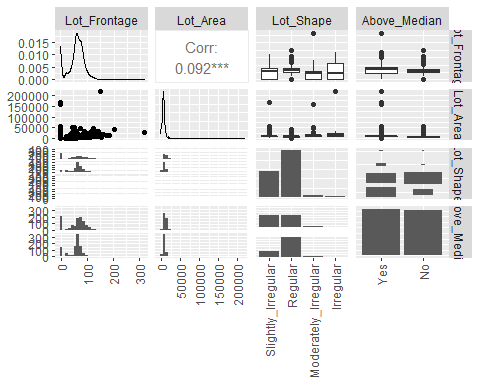
amesf\_date = ames %>%  
 select( Neighborhood,Year\_Built, Year\_Remod\_Add, Mo\_Sold, Year\_Sold, Sale\_Type, Sale\_Condition, Above\_Median)  
  
amesf\_exft = ames %>%  
 select(Neighborhood ,Garage\_Type, Garage\_Finish, Garage\_Cond, Garage\_Qual, Garage\_Cars, Garage\_Area, Paved\_Drive, Wood\_Deck\_SF, Open\_Porch\_SF, Enclosed\_Porch, Three\_season\_porch, Screen\_Porch, Pool\_Area, Pool\_QC, Fence, Misc\_Feature, Misc\_Val, Above\_Median)  
  
amesf\_house = ames %>%  
 select(Neighborhood,Bldg\_Type, House\_Style, Overall\_Qual, Overall\_Cond ,Above\_Median)  
  
amesf\_bas = ames %>%  
 select(Neighborhood,Foundation, Bsmt\_Qual, Bsmt\_Cond, Bsmt\_Exposure, BsmtFin\_Type\_1, BsmtFin\_SF\_1, BsmtFin\_Type\_2, BsmtFin\_SF\_2,Bsmt\_Unf\_SF, Total\_Bsmt\_SF ,Above\_Median)  
  
amesf\_exhs = ames %>%  
 select(Neighborhood,Roof\_Style, Roof\_Matl, Exterior\_1st, Exterior\_2nd, Mas\_Vnr\_Type, Mas\_Vnr\_Area, Exter\_Qual, Exter\_Cond ,Above\_Median)  
  
amesf\_inft = ames %>%  
 select(Neighborhood, Heating, Heating\_QC, Central\_Air, Electrical, First\_Flr\_SF, Second\_Flr\_SF, Low\_Qual\_Fin\_SF, Gr\_Liv\_Area, Bsmt\_Full\_Bath, Bsmt\_Half\_Bath, Bedroom\_AbvGr, Kitchen\_AbvGr, Kitchen\_Qual, TotRms\_AbvGrd, Functional, Fireplaces, Fireplace\_Qu ,Above\_Median)  
  
amesf\_land = ames %>%  
 select(Neighborhood, Lot\_Frontage, Lot\_Area, Lot\_Shape, Land\_Contour, Lot\_Config, Land\_Slope ,Above\_Median)  
  
amesf\_location = ames %>%  
 select( Neighborhood,MS\_Zoning, Street, Alley, Utilities, Condition\_1, Condition\_2, Longitude, Latitude ,Above\_Median)  
  
amesf\_ult = ames %>%  
 select(Neighborhood,Bsmt\_Qual, Garage\_Type, Garage\_Finish, Paved\_Drive, Lot\_Shape, Exterior\_1st, Exterior\_2nd, Mas\_Vnr\_Type, Exter\_Qual, Overall\_Qual, Fireplaces, Fireplace\_Qu, MS\_Zoning, Above\_Median)

ggpairs(amesf\_land, columns = 5:8)+ theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



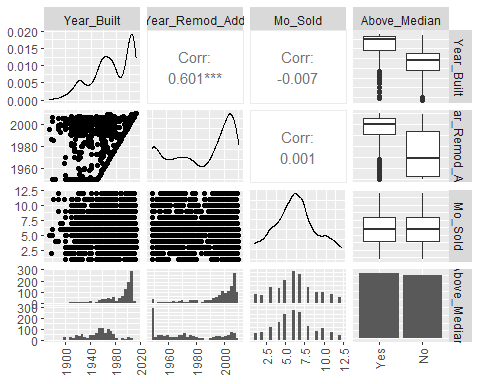
ggpairs(amesf\_land, columns = c(2,3,4,8))+ theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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



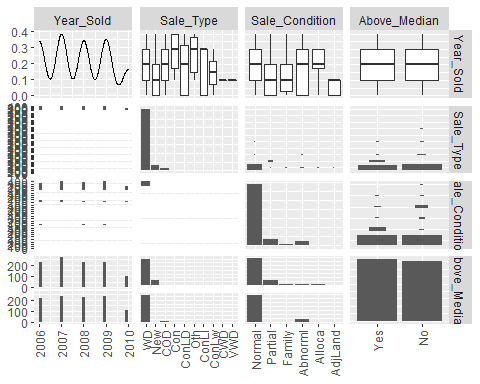
ggpairs(amesf\_date, columns = c(2:4,8)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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

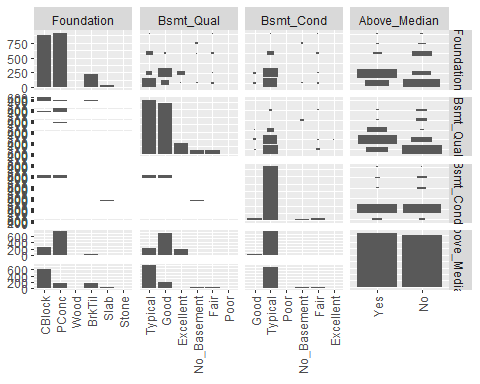


ggpairs(amesf\_date, columns = c(5:8)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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

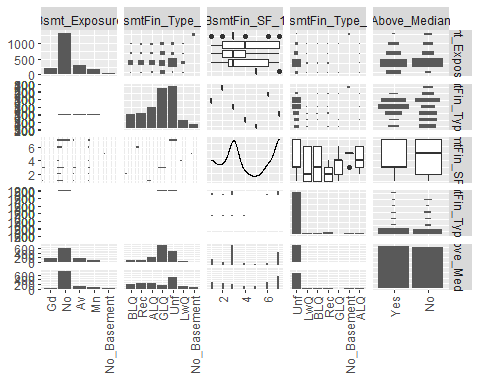


ggpairs(amesf\_bas, columns = c(2:4,12)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



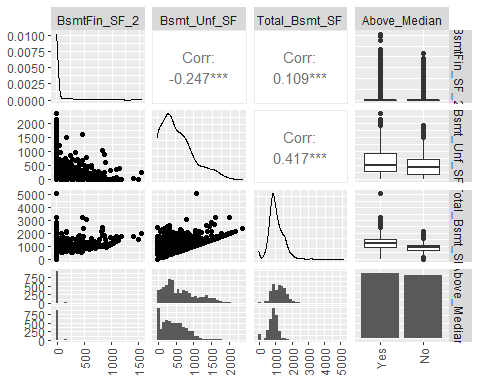
ggpairs(amesf\_bas, columns = c(5:8, 12)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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

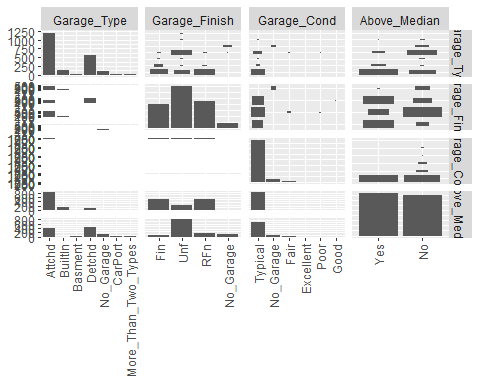


ggpairs(amesf\_bas, columns = c(9:12)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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

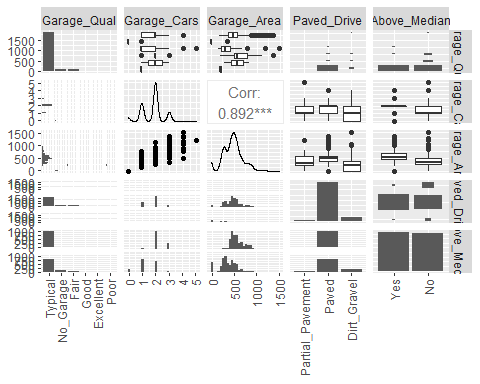


ggpairs(amesf\_exft, columns = c(2:4,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



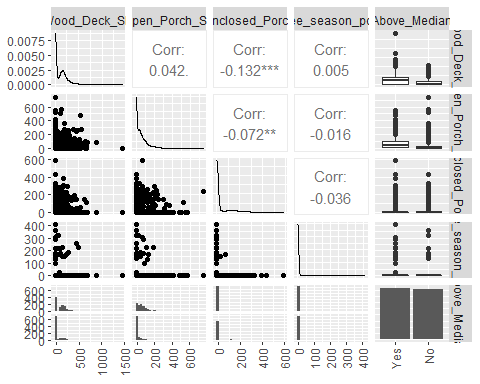
ggpairs(amesf\_exft, columns = c(5:8,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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



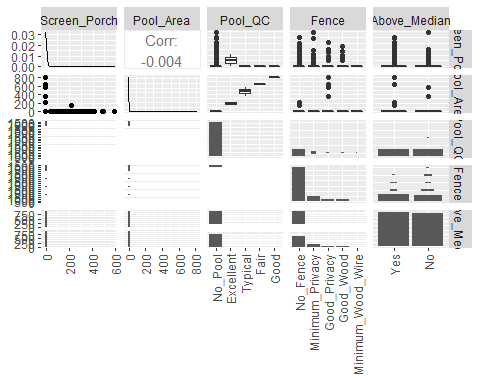
ggpairs(amesf\_exft, columns = c(9:12,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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



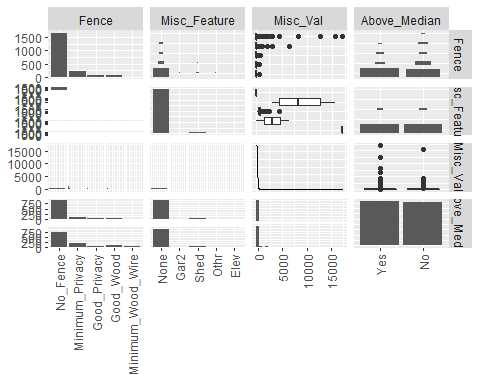
ggpairs(amesf\_exft, columns = c(13:16,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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

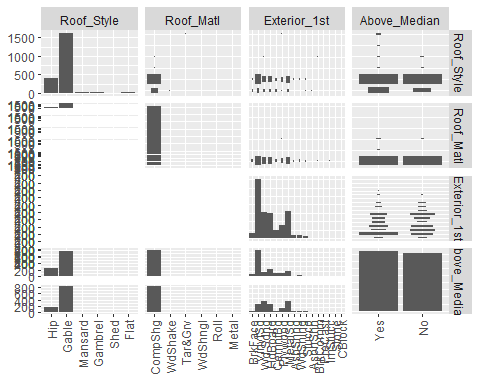


ggpairs(amesf\_exft, columns = c(16:19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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

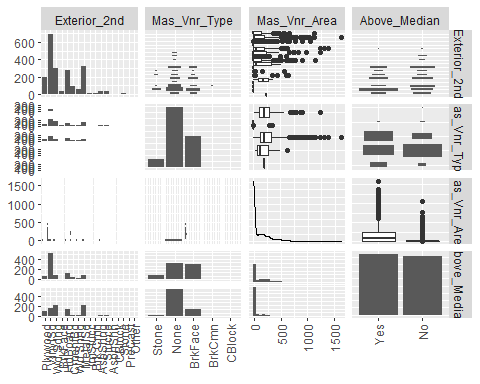


ggpairs(amesf\_exhs, columns = c(2:4,10), cardinality\_threshold = 16) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

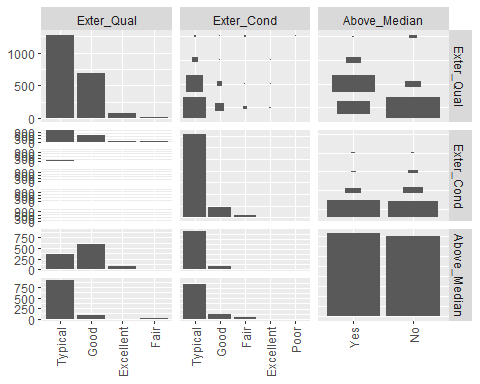


ggpairs(amesf\_exhs, columns = c(5:7,10), cardinality\_threshold = 17) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

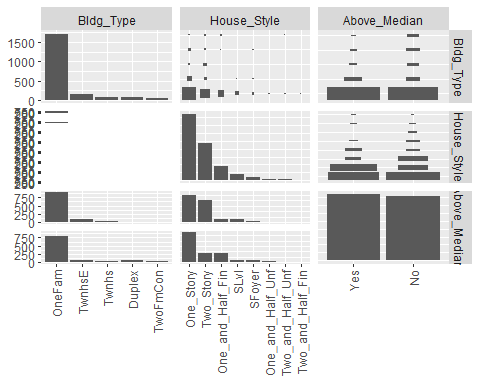
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



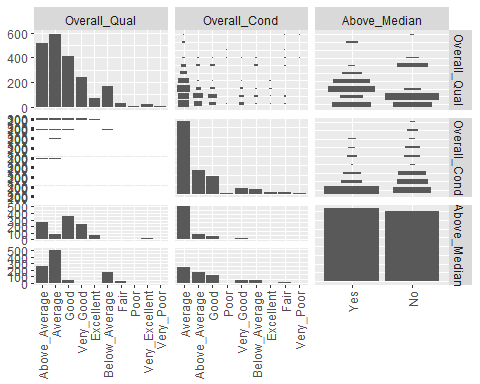
ggpairs(amesf\_exhs, columns = c(8:10)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



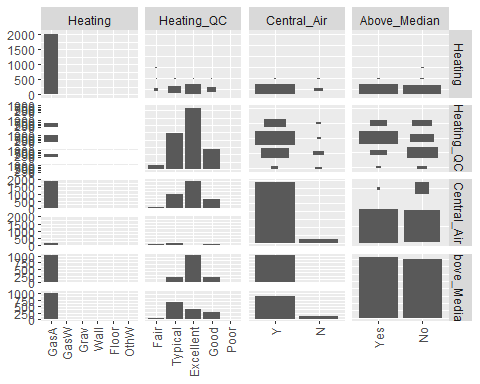
ggpairs(amesf\_house, columns = c(2,3,6)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



ggpairs(amesf\_house, columns = c(4:6)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

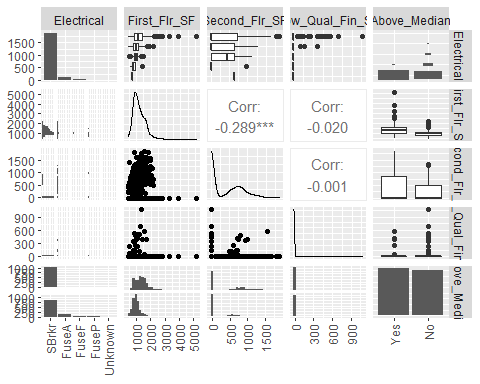


ggpairs(amesf\_inft, columns = c(2:4,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



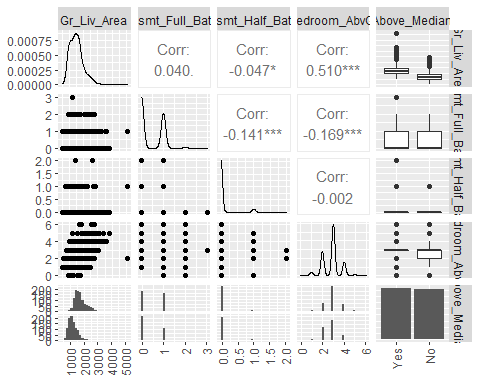
ggpairs(amesf\_inft, columns = c(5:8,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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



ggpairs(amesf\_inft, columns = c(9:12,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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



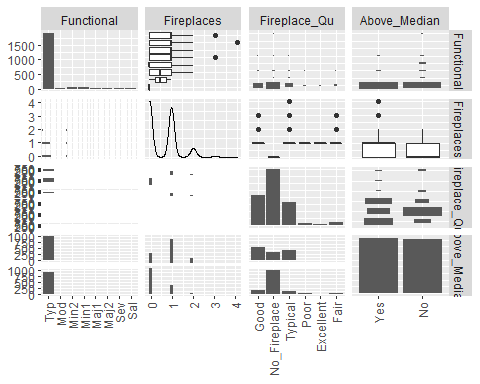
ggpairs(amesf\_inft, columns = c(13:16,19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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

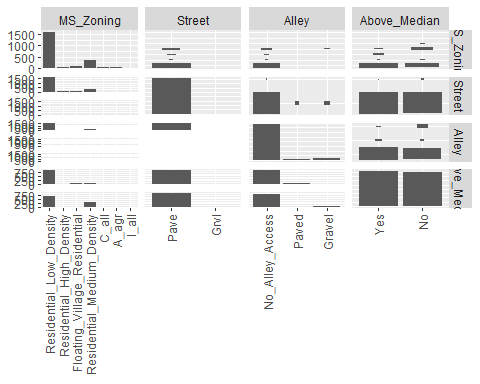


ggpairs(amesf\_inft, columns = c(16:19)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

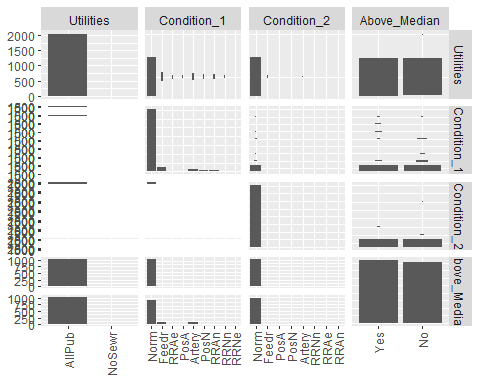
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



ggpairs(amesf\_location, columns = c(2:4,10)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

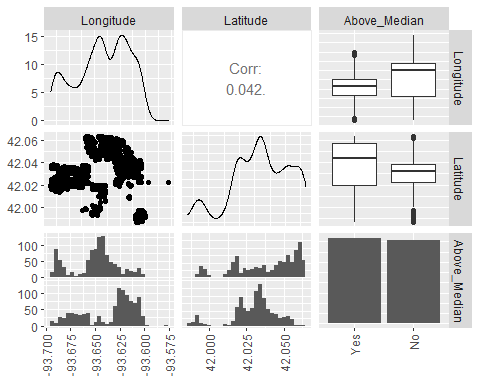


ggpairs(amesf\_location, columns = c(5:7,10)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

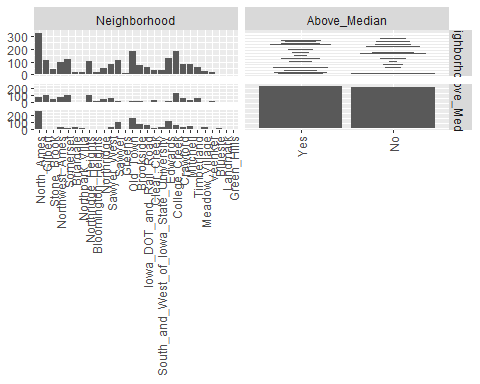


ggpairs(amesf\_location, columns = c(8:10)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

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



ggpairs(amesf\_location, columns = c(1,10), cardinality\_threshold = 28) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



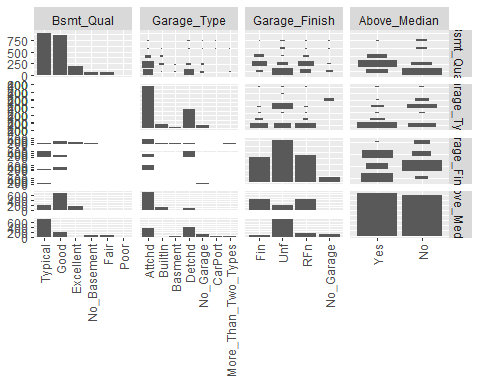
Most important variables:

Basement Quality Garage Type Garage Finish Paved Drive Lot Shape

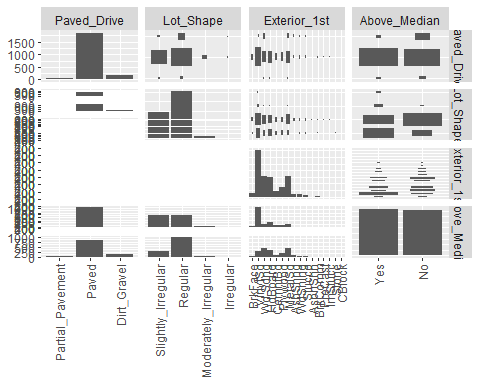
Exterior 1? Exterior 2?

Mas\_Vnr\_Type External Quality Overall Quality Fireplaces Fireplace Quality Zoning Neighborhood

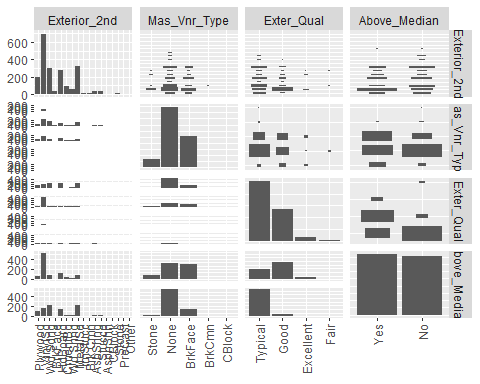
ggpairs(amesf\_ult, columns = c(2:4,15)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



ggpairs(amesf\_ult, columns = c(5:7,15), cardinality\_threshold = 17) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

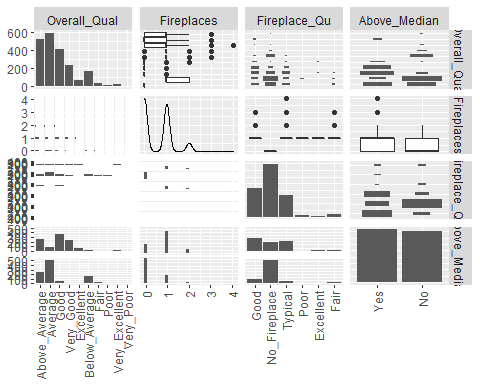


ggpairs(amesf\_ult, columns = c(8:10,15), cardinality\_threshold = 17) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

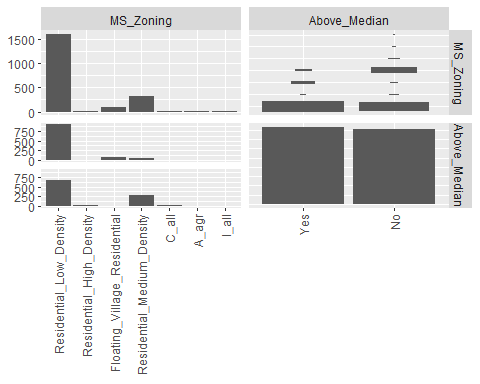


ggpairs(amesf\_ult, columns = c(11:13,15)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

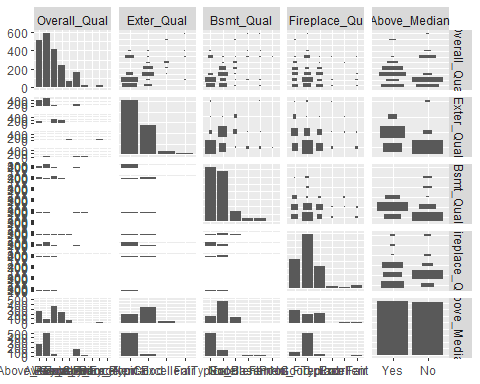
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.  
## `stat\_bin()` using `bins = 30`. Pick better value with `binwidth`.



ggpairs(amesf\_ult, columns = c(14:15)) + theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

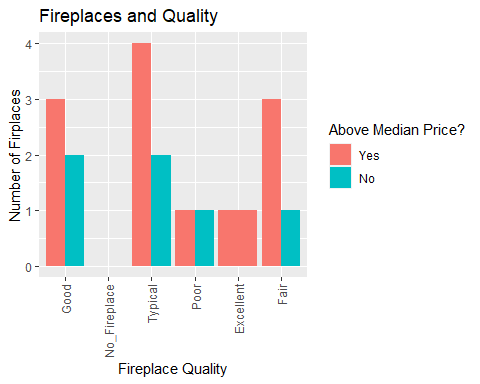


amespres = ames %>%  
 select(Overall\_Qual, Exter\_Qual, Bsmt\_Qual,Fireplace\_Qu, Above\_Median)  
 ggpairs(amespres)

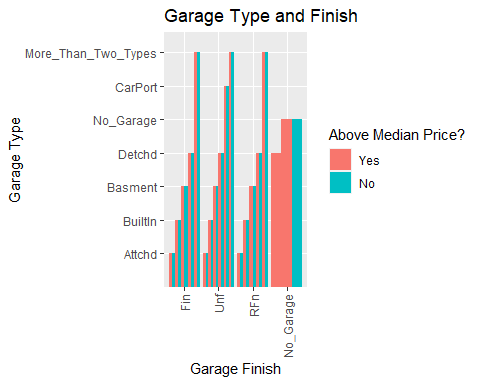


#ggplot(ames\_loc2, aes(x=Neighborhood, Y= n)) +   
# geom\_bar()  
#  
#ggplot(ames\_loc2\_y, aes(x=Neighborhood,y=n)) + geom\_point(alpha=0.1)   
#  
#ggplot(ames\_loc2\_y, aes(x)) + geom\_histogram()  
  
# Grouped  
#ggplot(data, aes(fill=condition, y=value, x=specie)) +   
 # geom\_bar(position="dodge", stat="identity")  
  
# Stacked  
#ggplot(data, aes(fill=condition, y=value, x=specie)) +   
 # geom\_bar(position="stack", stat="identity")  
  
# Stacked + percent  
#ggplot(data, aes(fill=condition, y=value, x=specie)) +   
 # geom\_bar(position="fill", stat="identity")

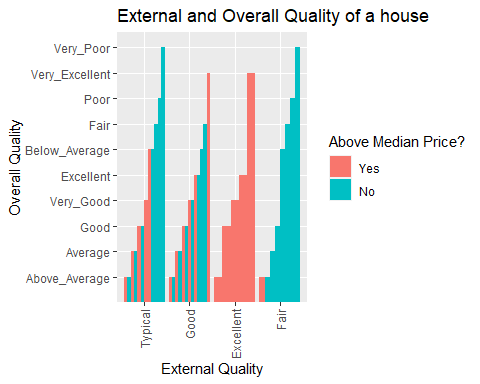
ggplot(ames, aes(y=Fireplaces, fill=Above\_Median, x=Fireplace\_Qu)) +  
 ggtitle("Fireplaces and Quality") +  
 xlab("Fireplace Quality") + ylab("Number of Firplaces") +   
 labs(fill = "Above Median Price?") +  
 geom\_bar(position="dodge", stat="identity")+   
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



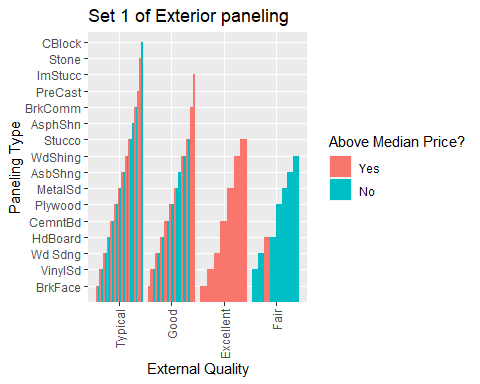
ggplot(ames, aes(fill=Above\_Median, y=Garage\_Type, x=Garage\_Finish)) +  
ggtitle("Garage Type and Finish") +  
 xlab("Garage Finish") + ylab("Garage Type") +   
 labs(fill = "Above Median Price?") +  
 geom\_bar(position="dodge", stat="identity")+   
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



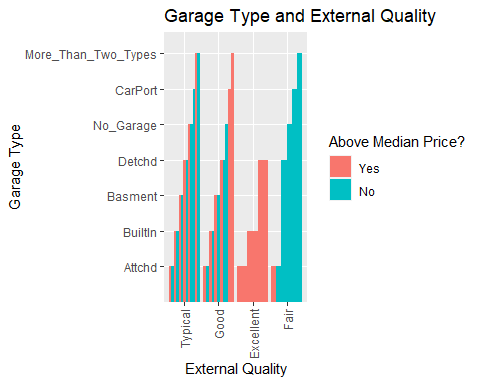
ggplot(ames, aes(fill=Above\_Median, x=Exter\_Qual, y=Overall\_Qual)) +   
 ggtitle("External and Overall Quality of a house") +  
 xlab("External Quality") + ylab("Overall Quality") +   
 labs(fill = "Above Median Price?") +  
 geom\_bar(position="dodge", stat="identity")+   
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



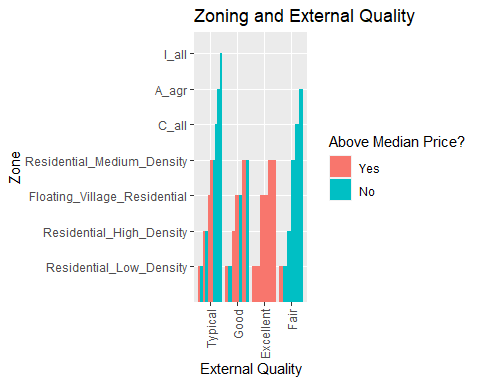
ggplot(ames, aes(fill=Above\_Median, y=Exterior\_1st, x=Exter\_Qual)) +  
 ggtitle("Set 1 of Exterior paneling") +  
 xlab("External Quality") + ylab("Paneling Type") +   
 labs(fill = "Above Median Price?") +  
 geom\_bar(position="dodge", stat="identity") +   
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



ggplot(ames, aes(fill=Above\_Median, y=Garage\_Type, x=Exter\_Qual)) +   
 geom\_bar(position="dodge", stat="identity") +   
 ggtitle("Garage Type and External Quality") +  
 xlab("External Quality") + ylab("Garage Type") +   
 labs(fill = "Above Median Price?") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



ggplot(ames, aes(fill=Above\_Median, x=Exter\_Qual, y=MS\_Zoning)) +   
 geom\_bar(position="dodge", stat="identity") +  
 ggtitle("Zoning and External Quality") +  
 xlab("External Quality") + ylab("Zone") +   
 labs(fill = "Above Median Price?") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))



ggplot(ames, aes(fill=Above\_Median, y=Neighborhood, x=Exter\_Qual)) +  
 ggtitle("Neighborhoods and External Quality") +  
 xlab("External Quality") + ylab("Neighborhood") +   
 labs(fill = "Above Median Price?") +  
 geom\_bar(position="dodge", stat="identity") +  
 theme(axis.text.x = element\_text(angle = 90, vjust = 0.5, hjust=1))

