

# Coursework MAP501 2022

Student ID: B928510

## Contents

Preamble	1
1. Data Preparation	2
2. Linear Regression	4
3. Logistic Regression	20
4. Multinomial Regression	25
5. Poisson/quasipoisson Regression	28

You will submit your coursework in the form of a single R notebook (i.e. `.Rmd` file) which can be rendered (“knitted”) to an `.pdf` document. Specifically, submit on Learn:

- your R notebook (i.e. the `.Rmd` file),
- the rendered `.pdf` version of your notebook. You might find it easier to knit to html, then print the html file to a pdf.

The coursework will be marked on the basis of correctness of code, interpretation of outputs and commentary as indicated. Therefore, please ensure that all code and outputs are visible in the knit document.

## Preamble

```
library(readxl) # not in original coursework doc -> to read csv file.
library(here) # not in original coursework doc -> to get data file.
library(tidyverse) # not in original coursework doc.
library(janitor) # not in original coursework doc -> used to clean column names.
library(lindia)
library(rio)
library(dplyr)
library(tidyr)
library(magrittr)
library(ggplot2)
library(pROC)
library(car)
library(nnet)
```

```
library(caret)
library(lme4)
library(AmesHousing)
```

```
Ames<-make_ames()
```

## 1. Data Preparation

- a. Import the soccer.csv dataset as “footballer\_data”. (2 points)

```
# Read in data & clean column names.
footballer_data <- read_csv(here("data", "soccer.csv"))
```

- b. Ensure all character variables are treated as factors and where variable names have a space, rename the variables without these. (3 points)

```
# Rename variables with spaces.
footballer_data <- footballer_data %>%
  clean_names()

# Convert all character variables to factors.
footballer_data <- footballer_data %>%
  mutate(across(
    c(full_name, birthday_gmt, position, current_club, nationality), as.factor))
footballer_data
```

```
# A tibble: 570 x 45
  full_~1   age birth~2 birth~3 posit~4 curre~5 minut~6 minut~7 minut~8 natio~9
  <fct>   <dbl>   <dbl> <fct>   <fct>   <fct>   <dbl>   <dbl>   <dbl> <fct>
1 Aaron ~    32  6.30e8 15/12/~ Defend~ West H~   1589    888    701 England
2 Aaron ~    35  5.46e8 16/04/~ Midfie~ Burnley   1217    487    730 England
3 Aaron ~    31  6.53e8 15/09/~ Midfie~ Hudder~   2327   1190   1137 Austra~
4 Aaron ~    31  6.62e8 26/12/~ Midfie~ Arsenal   1327    689    638 Wales
5 Aaron ~    22  9.68e8 07/09/~ Forward Hudder~    69     14     55 England
6 Aaron ~    24  8.81e8 26/11/~ Midfie~ Crysta~   3135   1605   1530 England
7 Abdelh~    25  8.49e8 28/11/~ Midfie~ Hudder~    49      0     49 Morocco
8 Abdoul~    29  7.26e8 01/01/~ Midfie~ Watford   3062   1566   1496 France
9 Abouba~    27  7.95e8 07/03/~ Forward Fulham    687    468    219 France
10 Adalbe~    25  8.65e8 31/05/~ Forward Watford     0      0      0 Venezu~
# ... with 560 more rows, 35 more variables: appearances_overall <dbl>,
#   appearances_home <dbl>, appearances_away <dbl>, goals_overall <dbl>,
#   goals_home <dbl>, goals_away <dbl>, assists_overall <dbl>,
#   assists_home <dbl>, assists_away <dbl>, penalty_goals <dbl>,
#   penalty_misses <dbl>, clean_sheets_overall <dbl>, clean_sheets_home <dbl>,
#   clean_sheets_away <dbl>, conceded_overall <dbl>, conceded_home <dbl>,
#   conceded_away <dbl>, yellow_cards_overall <dbl>, ...
```

- c. Remove the columns birthday and birthday\_GMT. (2 points)

```
# Removes birthday column and birthday_gmt column.
```

```
footballer_data <- footballer_data %>%
  select(-c(birthday, birthday_gmt))
footballer_data
```

```
# A tibble: 570 x 43
```

	full_~1	age	posit~2	curre~3	minut~4	minut~5	minut~6	natio~7	appea~8	appea~9
	<fct>	<dbl>	<fct>	<fct>	<dbl>	<dbl>	<dbl>	<fct>	<dbl>	<dbl>
1	Aaron ~	32	Defend~	West H~	1589	888	701	England	20	11
2	Aaron ~	35	Midfie~	Burnley	1217	487	730	England	16	7
3	Aaron ~	31	Midfie~	Hudder~	2327	1190	1137	Austra~	29	15
4	Aaron ~	31	Midfie~	Arsenal	1327	689	638	Wales	28	14
5	Aaron ~	22	Forward	Hudder~	69	14	55	England	2	1
6	Aaron ~	24	Midfie~	Crysta~	3135	1605	1530	England	35	18
7	Abdelh~	25	Midfie~	Hudder~	49	0	49	Morocco	2	0
8	Abdoul~	29	Midfie~	Watford	3062	1566	1496	France	35	18
9	Abouba~	27	Forward	Fulham	687	468	219	France	13	8
10	Adalbe~	25	Forward	Watford	0	0	0	Venezu~	0	0

```
# ... with 560 more rows, 33 more variables: appearances_away <dbl>,
# goals_overall <dbl>, goals_home <dbl>, goals_away <dbl>,
# assists_overall <dbl>, assists_home <dbl>, assists_away <dbl>,
# penalty_goals <dbl>, penalty_misses <dbl>, clean_sheets_overall <dbl>,
# clean_sheets_home <dbl>, clean_sheets_away <dbl>, conceded_overall <dbl>,
# conceded_home <dbl>, conceded_away <dbl>, yellow_cards_overall <dbl>,
# red_cards_overall <dbl>, goals_involved_per_90_overall <dbl>, ...
```

d. Remove the cases with age<=15 and age>40. (2 points)

```
footballer_data <- footballer_data %>%
  filter(age > 15 & age <= 40)
```

```
# The minimum and maximum age was printed to check range was correct.
```

```
footballer_data %>%
  summarise (
    min_age = min(age),
    max_age = max(age)
  )
```

```
footballer_data
```

```
# A tibble: 1 x 2
```

	min_age	max_age
	<dbl>	<dbl>
1	20	40

```
# A tibble: 565 x 43
```

	full_~1	age	posit~2	curre~3	minut~4	minut~5	minut~6	natio~7	appea~8	appea~9
	<fct>	<dbl>	<fct>	<fct>	<dbl>	<dbl>	<dbl>	<fct>	<dbl>	<dbl>
1	Aaron ~	32	Defend~	West H~	1589	888	701	England	20	11
2	Aaron ~	35	Midfie~	Burnley	1217	487	730	England	16	7
3	Aaron ~	31	Midfie~	Hudder~	2327	1190	1137	Austra~	29	15
4	Aaron ~	31	Midfie~	Arsenal	1327	689	638	Wales	28	14
5	Aaron ~	22	Forward	Hudder~	69	14	55	England	2	1

6	Aaron ~	24	Midfie~	Crysta~	3135	1605	1530	England	35	18
7	Abdelh~	25	Midfie~	Hudder~	49	0	49	Morocco	2	0
8	Abdoul~	29	Midfie~	Watford	3062	1566	1496	France	35	18
9	Abouba~	27	Forward	Fulham	687	468	219	France	13	8
10	Adalbe~	25	Forward	Watford	0	0	0	Venezu~	0	0

```
# ... with 555 more rows, 33 more variables: appearances_away <dbl>,
#   goals_overall <dbl>, goals_home <dbl>, goals_away <dbl>,
#   assists_overall <dbl>, assists_home <dbl>, assists_away <dbl>,
#   penalty_goals <dbl>, penalty_misses <dbl>, clean_sheets_overall <dbl>,
#   clean_sheets_home <dbl>, clean_sheets_away <dbl>, conceded_overall <dbl>,
#   conceded_home <dbl>, conceded_away <dbl>, yellow_cards_overall <dbl>,
#   red_cards_overall <dbl>, goals_involved_per_90_overall <dbl>, ...
```

## 2. Linear Regression

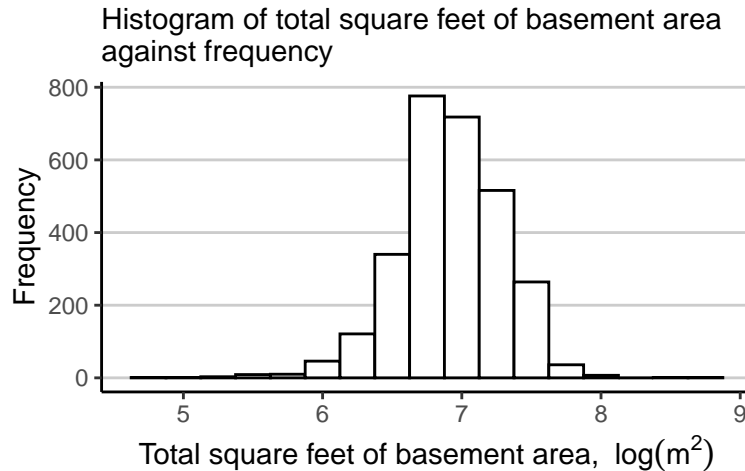
In this problem, you are going to investigate the response variable Total\_Bsmt\_SF in “Ames” dataset through linear regression.

- By adjusting x axis range and number of bars, create a useful histogram of Total\_Bsmt\_SF on the full dataset. Ensure that plot titles and axis labels are clear. (4 points)

```
Ames
# Where 'Total_Bsmt_SF' is 'Total square feet of basement area'.
# Since it is linear regression we want the histogram to be 'bell-shaped'
# i.e., follow a Gaussian distribution.
# Plotting area as log(area) provided the most useful histogram...
# to linearize the data.
# Bin width was estimated via trial and error.

figure_1 <- Ames %>%
  ggplot(aes(x = log(Total_Bsmt_SF))) +
  geom_histogram(colour = "black", fill = "white", binwidth = 0.25) +
  labs (
    subtitle = "Histogram of total square feet of basement area\nagainst frequency",
    x = "Total square feet of basement area, "~log(m^2),
    y = "Frequency"
  ) +
  theme_classic() +
  theme(
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),
    panel.grid.major.y = element_line(colour = "grey80"),
    panel.grid.minor.y = element_blank(),
    panel.background = element_blank()
  )

figure_1
```



```
# A tibble: 2,930 x 81
  MS_Sub~1 MS_Zo~2 Lot_F~3 Lot_A~4 Street Alley Lot_S~5 Land_~6 Utili~7 Lot_C~8
    <fct>    <fct>    <dbl>    <int> <fct> <fct> <fct>    <fct>    <fct>
1 One_Sto~ Reside~    141   31770 Pave  No_A~ Slight~ Lvl    AllPub Corner
2 One_Sto~ Reside~     80   11622 Pave  No_A~ Regular Lvl    AllPub Inside
3 One_Sto~ Reside~     81   14267 Pave  No_A~ Slight~ Lvl    AllPub Corner
4 One_Sto~ Reside~     93   11160 Pave  No_A~ Regular Lvl    AllPub Corner
5 Two_Sto~ Reside~     74   13830 Pave  No_A~ Slight~ Lvl    AllPub Inside
6 Two_Sto~ Reside~     78   9978 Pave  No_A~ Slight~ Lvl    AllPub Inside
7 One_Sto~ Reside~     41   4920 Pave  No_A~ Regular Lvl    AllPub Inside
8 One_Sto~ Reside~     43   5005 Pave  No_A~ Slight~ HLS    AllPub Inside
9 One_Sto~ Reside~     39   5389 Pave  No_A~ Slight~ Lvl    AllPub Inside
10 Two_Sto~ Reside~     60   7500 Pave  No_A~ Regular Lvl    AllPub Inside
# ... with 2,920 more rows, 71 more variables: Land_Slope <fct>,
#   Neighborhood <fct>, Condition_1 <fct>, Condition_2 <fct>, Bldg_Type <fct>,
#   House_Style <fct>, Overall_Qual <fct>, Overall_Cond <fct>,
#   Year_Built <int>, Year_Remod_Add <int>, Roof_Style <fct>, Roof_Matl <fct>,
#   Exterior_1st <fct>, Exterior_2nd <fct>, Mas_Vnr_Type <fct>,
#   Mas_Vnr_Area <dbl>, Exter_Qual <fct>, Exter_Cond <fct>, Foundation <fct>,
#   Bsmt_Qual <fct>, Bsmt_Cond <fct>, Bsmt_Exposure <fct>, ...
```

```
# A frequency density plot was created.
```

```
# This was overlaid by a distribution curve to validate...
```

```
# the normal distribution of 'Total_Bsmt_SF'.
```

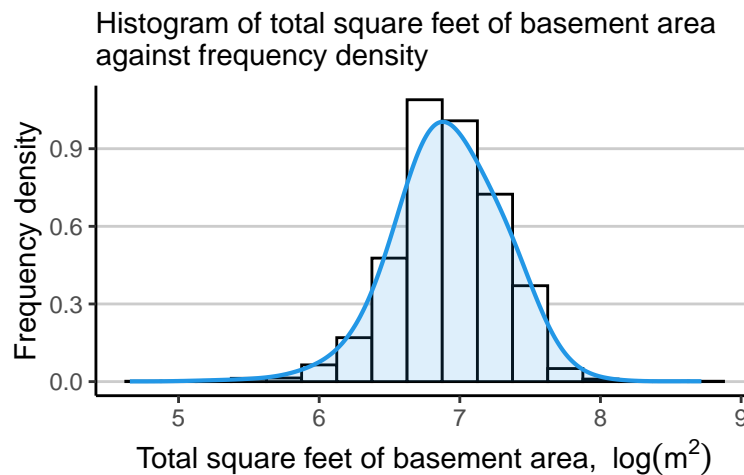
```
figure_2 <- Ames %>%
  ggplot(aes(x = log(Total_Bsmt_SF))) +
  geom_histogram(aes(y = ..density..), colour = 1, fill = "white", binwidth = 0.25) +
  geom_density(lwd = 0.75, colour = 4, fill = 4, alpha = 0.15, adjust = 2.75) +
  labs (
    subtitle = "Histogram of total square feet of basement area\nagainst frequency density",
    x = "Total square feet of basement area, "~log(m^2),
    y = "Frequency density"
  ) +
  theme_classic() +
  theme(
    panel.grid.major.x = element_blank(),
```

```

panel.grid.minor.x = element_blank(),
panel.grid.major.y = element_line(colour = "grey80"),
panel.grid.minor.y = element_blank(),
panel.background = element_blank(),
)

```

figure\_2



b. Using “Ames” dataset to create a new dataset called “Ames2” in which you remove all cases corresponding to:

- (i) MS\_Zoning categories of A\_agr (agricultural), C\_all (commercial) and I\_all (industrial),
- (ii) BsmtFin\_Type\_1 category of “No\_Basement”.
- (iii) Bldg\_Type category of “OneFam”

and drop the unused levels from the dataset “Ames2”. (4 points)

```

# Checked levels before removal.
levels(Ames$MS_Zoning)
levels(Ames$BsmtFin_Type_1)
levels(Ames$Bldg_Type)

```

```

[1] "Floating_Village_Residential" "Residential_High_Density"
[3] "Residential_Low_Density"      "Residential_Medium_Density"
[5] "A_agr"                       "C_all"
[7] "I_all"
[1] "ALQ"      "BLQ"      "GLQ"      "LwQ"      "No_Basement"
[6] "Rec"      "Unf"
[1] "OneFam"   "TwoFmCon" "Duplex"   "Twnhs"    "TwnhsE"

```

```

# (i), (ii) and (iii)
Ames2 <- Ames %>%
  filter(
    MS_Zoning != "A_agr",
    MS_Zoning != "C_all",

```

```

MS_Zoning != "I_all",
BsmtFin_Type_1 != "No_Basement",
Bldg_Type != "OneFam"
)

Ames2$MS_Zoning <- droplevels(Ames2$MS_Zoning)
Ames2$BsmtFin_Type_1 <- droplevels(Ames2$BsmtFin_Type_1)
Ames2$Bldg_Type <- droplevels(Ames2$Bldg_Type)

# Checked levels after removal.
levels(Ames2$MS_Zoning)
levels(Ames2$BsmtFin_Type_1)
levels(Ames2$Bldg_Type)

```

```

[1] "Floating_Village_Residential" "Residential_High_Density"
[3] "Residential_Low_Density"      "Residential_Medium_Density"
[1] "ALQ" "BLQ" "GLQ" "LwQ" "Rec" "Unf"
[1] "TwoFmCon" "Duplex" "Twnhs" "TwnhsE"

```

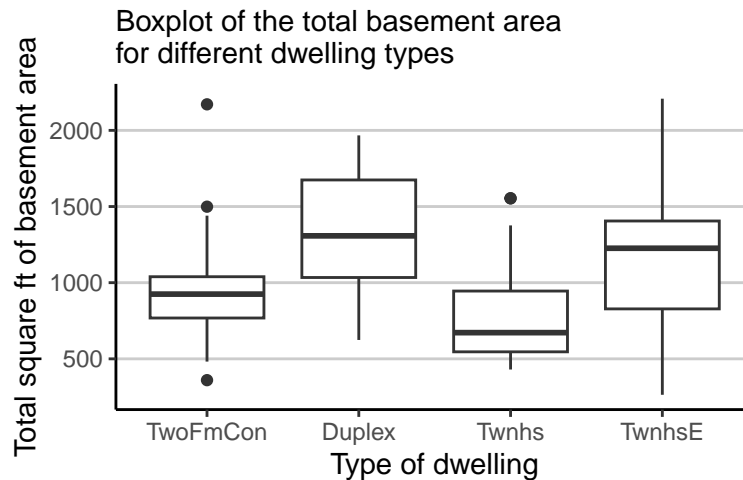
- c. Choose an appropriate plot to investigate the relationship between Bldg\_Type and Total\_Bsmt\_SF in Ames2. (2 points)

```

# Box plot since the relationship is categorical-numerical.
figure_3 <- Ames2 %>%
  ggplot(aes(x = Bldg_Type, y = Total_Bsmt_SF)) +
  geom_boxplot() +
  labs (
    subtitle = "Boxplot of the total basement area\nfor different dwelling types",
    x = "Type of dwelling",
    y = "Total square ft of basement area"
  ) +
  theme_classic() +
  theme(
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),
    panel.grid.major.y = element_line(colour = "grey80"),
    panel.grid.minor.y = element_blank(),
    panel.background = element_blank(),
  )

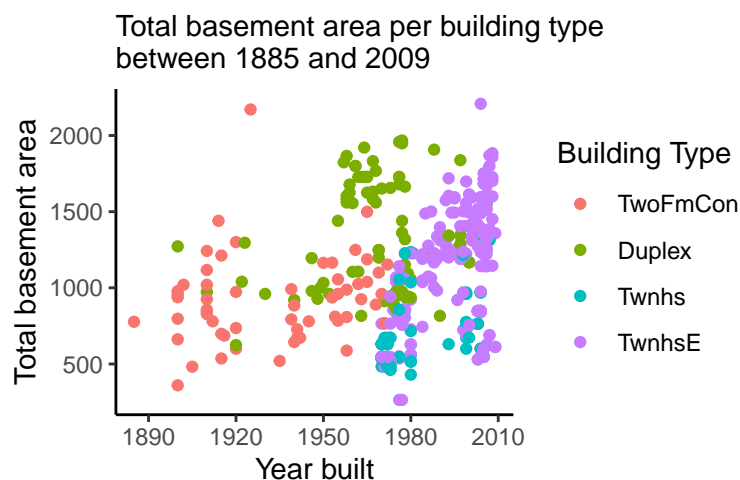
figure_3

```



- d. Choose an appropriate plot to investigate the relationship between `Year_Built` and `Total_Bsmt_SF` in `Ames2`. Color points according to the factor `Bldg_Type`. Ensure your plot has a clear title, axis labels and legend. What do you notice about how Basement size has changed over time? Were there any slowdowns in construction over this period? When? Can you think why? (4 points)

```
Ames2 %>%
  ggplot(aes(x = Year_Built, y = Total_Bsmt_SF, colour = Bldg_Type)) +
  geom_point() +
  labs(
    subtitle = "Total basement area per building type\nbetween 1885 and 2009",
    x = "Year built",
    y = "Total basement area",
    colour = "Building Type"
  ) +
  theme_classic()
```



*# As one can see it is difficult to detect a trend in the relationship...  
 # between Year\_Built and Total\_Bsmt\_SF from a raw scatter plot.  
 # Since there are vertically overlapping points, we can compare to average...*

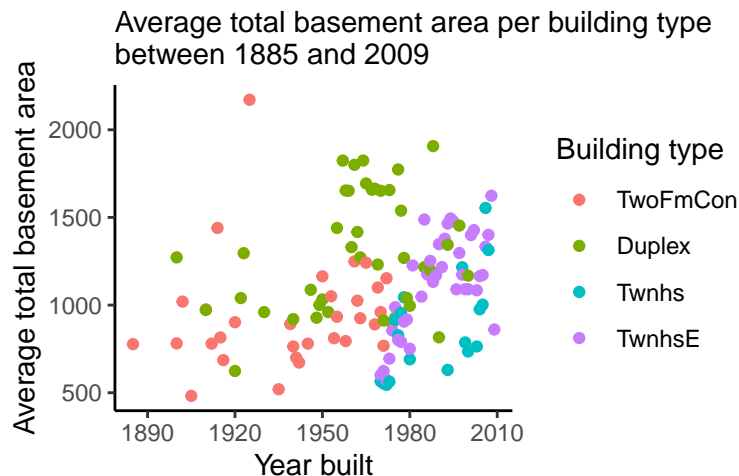


```
# Total_Bsmt_SF for each Bldg_Type built in a given year.

ave_by_year_built <- Ames2 %>%
  group_by(Year_Built, Bldg_Type) %>%
  summarise(mean(Total_Bsmt_SF))

figure_4 <- ave_by_year_built %>%
  ggplot(aes(x = Year_Built, y = `mean(Total_Bsmt_SF)`, colour = Bldg_Type)) +
  geom_point() +
  labs(
    subtitle = "Average total basement area per building type\nbetween 1885 and 2009",
    x = "Year built",
    y = "Average total basement area",
    colour = "Building type"
  ) +
  theme_classic()

figure_4
```



```
# The plot shows the basement size for all building types over time...
# in Ames Iowa, has gradually increased.
# Between 1930 and 1935 the rate of basement construction appears...
# to have slowed down i.e., the number of basements constructed...
# is at a lower frequency than before and after this period.
# Reason: This period also included the Great Depression from 1929 to 1933...
# so most businesses and people did not have the means for land acquisition or
# construction.
# The same occurs between 2007 and 2010 which coincides with...
# the 2007-2008 global financial crisis.

# (Link to sources to support these inferences are below.)
```

Sources:

1. Financial Crisis
2. List of Recessions in the United States

- e. Why do we make these plots? Comment on your findings from these plots (1 sentence is fine). (2 points)

```
# The box plots are to compare the distribution of numeric values...
# in each category. The scatter plot was to determine a trend...
# between the two variables and decide which model may be best fit...
# for prediction i.e., if there is a linear trend this may indicate...
# a linear regression model would best fit the data set.

# Findings:
# - Generally between 1885 and 2009 the basement size has increased.
# - The number of TwoFmCon constructed completely stopped around 1975 and...
#   around the time Twnhs and TwnhsE basements began to be constructed.
# - Twnhs has the lowest median total basement area with 50% of its areas...
#   ranging between 550 to 850 ft^2.
# - Duplex has the highest median total basement area with 50% of its... areas
#   areas ranging between 1100 to 1700 ft^2. This is the largest spread...
#   in the data set despite TwnshE having more outliers.
```

- f. Now choose an appropriate plot to investigate the relationship between Bldg\_Type and Year\_Built in Ames2. Why should we consider this? What do you notice? (3 points)

```
# This must be considered to compare frequency of each category...
# at different periods of time.

# We should consider this to avoid over generalizing which...
# could increase the residual error of predictions due to not taking...
# the different frequencies (and therefore sample sizes)...
# of each category into account.

# For example if we wanted to investigate Total_Bsmt_SF...
# based on Year_Built and Bldg_Type between 1980 and 2008, we now know...
# we should drop the 'TwoFmCon' level because there were zero TwoFmCon...
# buildings constructed during that time period.
# Duplex could also be dropped if the frequency...
# is found to be too small i.e., <30 or small relative to the...
# frequencies of other categories.

# We notice that the 'TwoFmCon' building was only constructed between...
# 1885 to 1965. This is around the same time, Twnhs and TwnhsE building...
# types began construction and at a noticeably higher frequency.
# Duplex building were dispersed more consistently between ...
# 1900 and 1990.
```

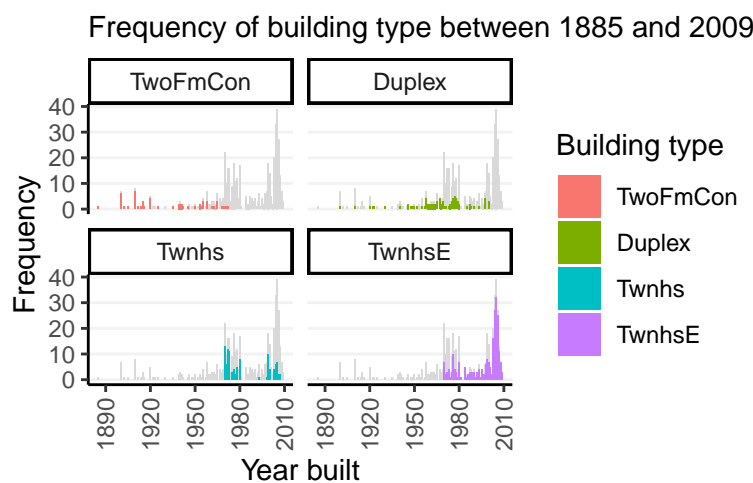
```
figure_5 <- Ames2 %>%
  ggplot(aes(x = Year_Built, fill = Bldg_Type)) +
  geom_bar(data = select(Ames2, !Bldg_Type), fill = "grey85") +
  geom_bar() +
  facet_wrap(facets = vars(Bldg_Type)) +
  labs(
    subtitle = "Frequency of building type between 1885 and 2009",
    x = "Year built",
    y = "Frequency",
```

```

    fill = "Building type"
  ) +
  theme_classic() +
  theme(
    panel.grid.major.x = element_blank(),
    panel.grid.minor.x = element_blank(),
    panel.grid.major.y = element_line(colour = "grey95"),
    panel.grid.minor.y = element_blank(),
    panel.background = element_blank(),
    axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1)
  )

```

figure\_5



*# TO SEE A CLEARER VERSION OF THIS PLOT PLEASE OPEN IT IN THE .RMD FILE.*

- g. Use the `lm` command to build a linear model, `linmod1`, of `Total_Bsmt_SF` as a function of the predictors `Bldg_Type` and `Year_Built` for the “Ames2” dataset. (2 points)

```

linmod1 <- lm(formula = Total_Bsmt_SF ~ Bldg_Type + Year_Built,
              data = Ames2)

```

- h. State and evaluate the assumptions of the model. (6 points)

*# Scatter plot of Year Built vs. Total\_Bsmt\_SF with linear trend line.  
 # The first one shows the overall trend line of all categories.  
 # The second one shows the separate trend line of each category.*

```

figure_6a <- ave_by_year_built %>%
  ggplot(aes(x = Year_Built, y = `mean(Total_Bsmt_SF)`, colour = Bldg_Type)) +
  geom_point() +
  labs(
    subtitle = "Average total basement area per building type\nbetween 1885 and 2009",
    x = "Year built",
    y = "Average total basement area",
  )

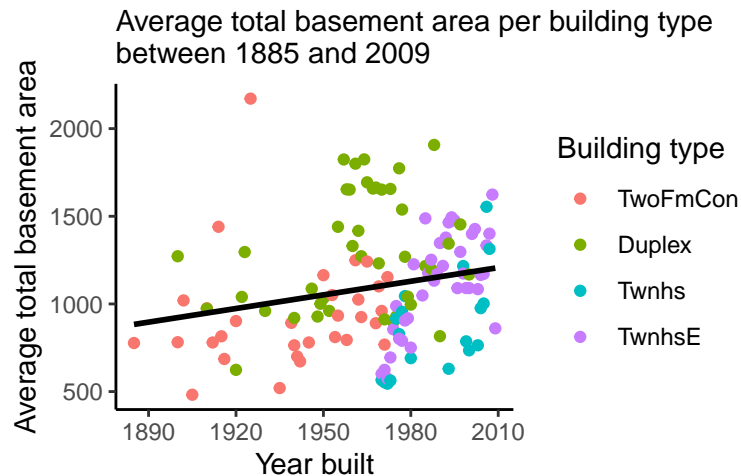
```

```

  colour = "Building type"
) +
geom_smooth(method = "lm", se = FALSE, colour = "black") +
theme_classic()

```

figure\_6a

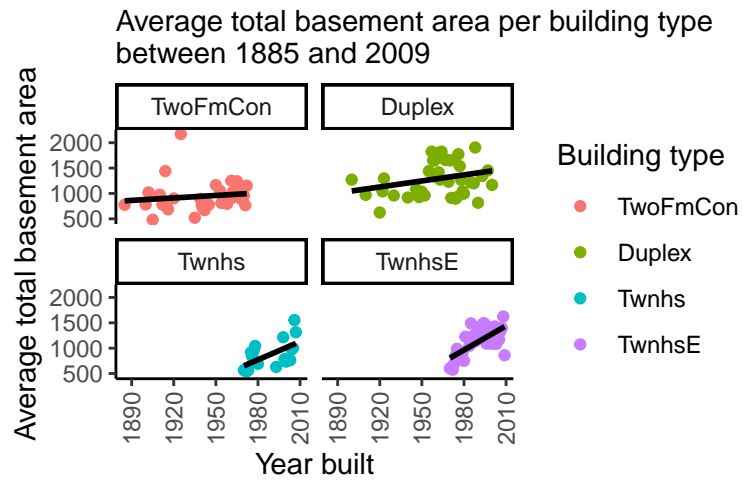


```

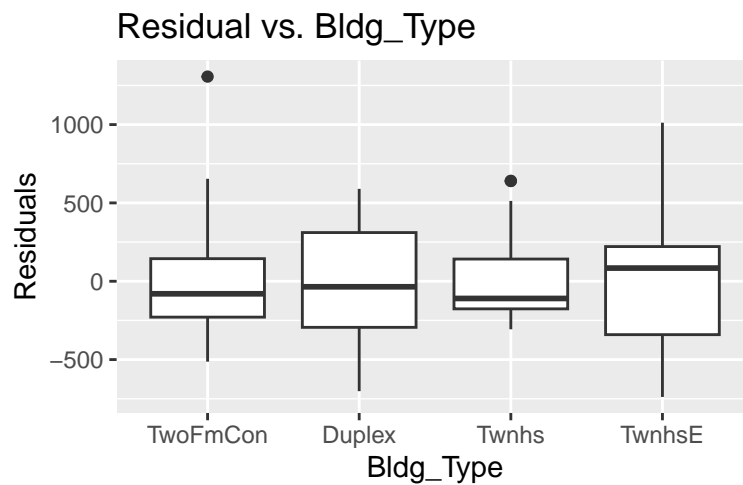
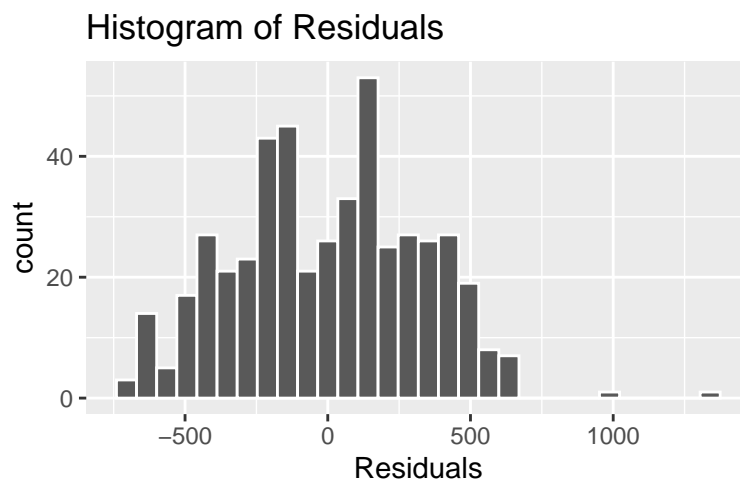
figure_6b <- ave_by_year_built %>%
  ggplot(aes(x = Year_Built, y = `mean(Total_Bsmt_SF)`, colour = Bldg_Type)) +
  geom_point() +
  facet_wrap(facets = vars(Bldg_Type)) +
  labs(
    subtitle = "Average total basement area per building type\nbetween 1885 and 2009",
    x = "Year built",
    y = "Average total basement area",
    colour = "Building type"
  ) +
  geom_smooth(method = "lm", se = FALSE, colour = "black") +
  theme_classic() +
  theme(axis.text.x = element_text(angle = 90, vjust = 0.5, hjust = 1))

```

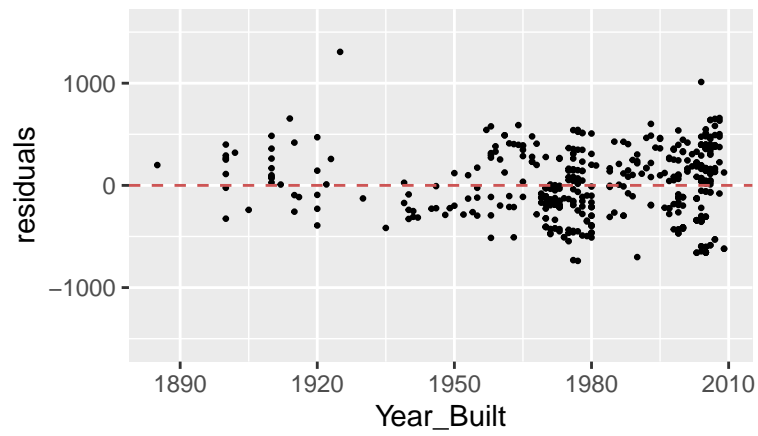
figure\_6b



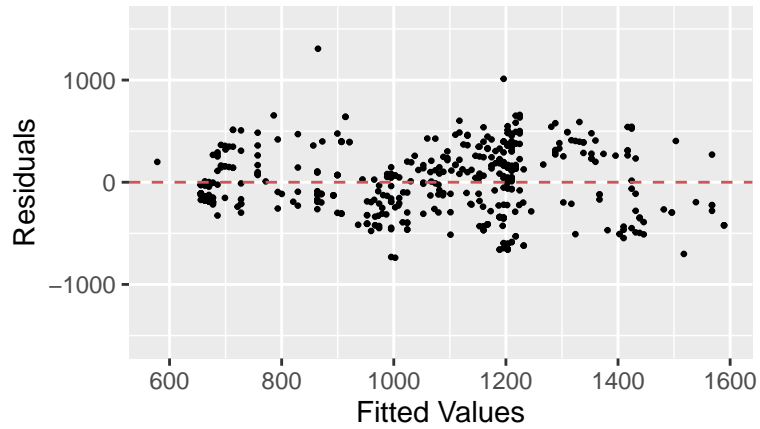
```
figure_7 <- linmod1 %>%
  gg_diagnose(max.per.page = 1)
```



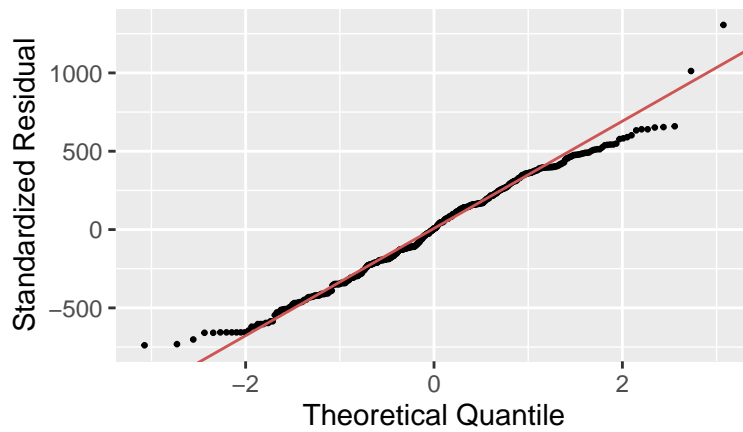
Residual vs. Year\_Built

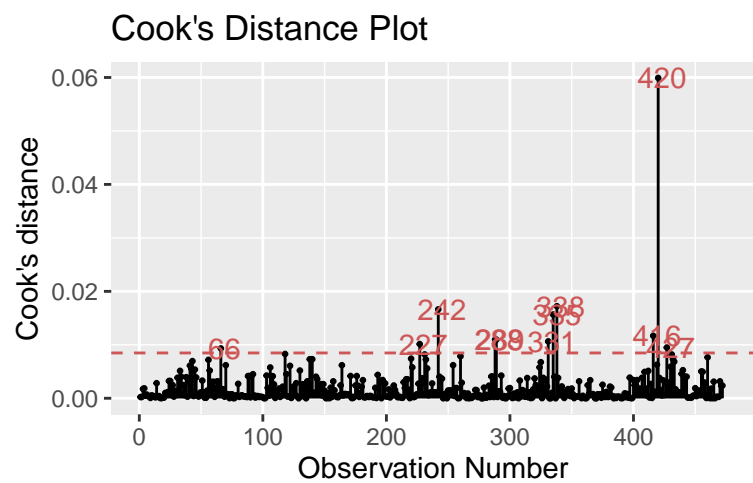
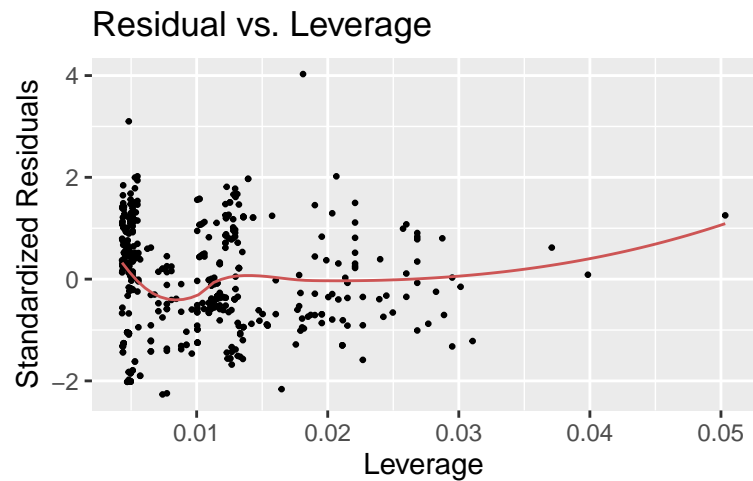
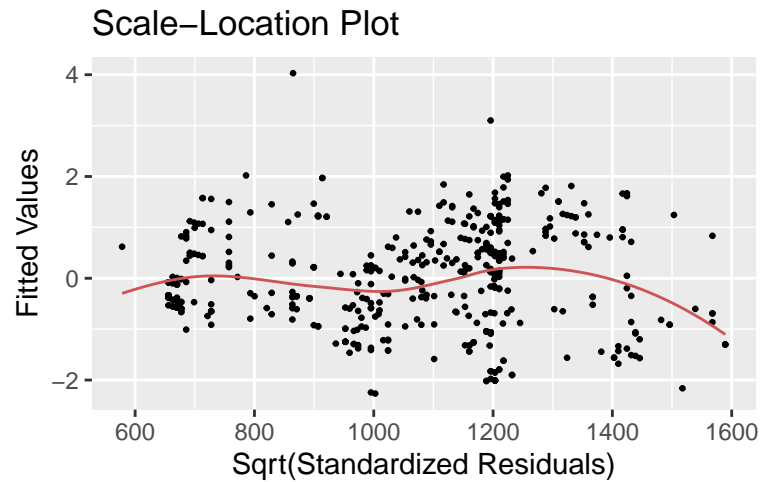


Residual vs. Fitted Value



Normal-QQ Plot





```
# The assumption is the model follows the assumptions...
# of a Gaussian distribution that is:
```

```

# 1) Linearity -> If there is 1 categorical variable and 1 or more...
#                 continuous variables linearity is assessed by...
#                 a scatter plot with the levels in different colors...
#                 hence figure_6a and figure_6b.
#                 Both figures show the trend is roughly...
#                 linear so this assumption is held.

##### All plots referred to in 2) and 3) are in figure_7. #####

# 2) Normality -> Aside from a few outliers in the histogram of...
#                 of residuals it appears normality assumption is...
#                 is upheld. The normal...
#                 QQ plot also shows this since most of the standardized...
#                 residuals are along the red line.
#

# 3) Homoscedasticity -> The model appears to violate this assumption.
#                         This is because the residuals in the plot...
#                         of Year Built vs Residuals show as the...
#                         Year Built increases the further the residuals...
#                         are from the zero-mean. A similar trend was...
#                         observed in Fitted Values vs Residuals.
#                         The boxplot of Bldg_Type vs Residuals also...
#                         show the IQR are not similar and every category...
#                         aside from 'Duplex' are far from the zero mean.

```

- i. Use the `lm` command to build a second linear model, `linmod2`, for `Total_Bsmt_SF` as a function of `Bldg_Type`, `Year_Built` and `Lot_Area`. (2 points)

```

# I assumed the question meant to the model using the...
# Ames2 dataset as well.
linmod2 <- lm(formula = Total_Bsmt_SF ~ Bldg_Type + Year_Built + Lot_Area,
              data = Ames2)

```

- j. Use Anova and Adjusted R-squared to compare these two models, and decide which is a better model. (6 points)

```

summary(linmod1)
summary(linmod2)
anova(linmod1, linmod2)

# Anova -> 8.099e-05 < 0.05 therefore linmod2 is a more statistically...
#         significant model compared to linmod1.
#         In context, this means adding the 'Lot_Area' predictor...
#         had a significant effect on the 'Total Basement Area in...
#         Square Feet'.
# Adjusted R-squared -> linmod1 has an Adjusted R-squared value...
#                       of 0.3282 whereas linmod2 has an...
#                       Adjusted R-squared value of 0.3489.
#                       Therefore linmod2 has a slightly better fit...
#                       than linmod1 as an additional 2.07% of the...
#                       variation in Total_Bsmt_SF can be explained by...
#                       its predictors.

```



```
# Therefore the better model is linmod2. That said the Adjusted R-squared...
# value is still low so its predictors are explanatory, there are...
# alot of uncontrolled factors that affect the Total_Bsmt_SF.
```

Call:

```
lm(formula = Total_Bsmt_SF ~ Bldg_Type + Year_Built, data = Ames2)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-738.53	-223.35	7.68	238.36	1306.23

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	-1.293e+04	1.833e+03	-7.054	6.30e-12	***
Bldg_TypeDuplex	1.870e+02	6.504e+01	2.875	0.00422	**
Bldg_TypeTwnhs	-5.314e+02	7.252e+01	-7.327	1.04e-12	***
Bldg_TypeTwnhsE	-2.349e+02	7.678e+01	-3.059	0.00235	**
Year_Built	7.166e+00	9.478e-01	7.560	2.15e-13	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 327.1 on 467 degrees of freedom

Multiple R-squared: 0.3339, Adjusted R-squared: 0.3282

F-statistic: 58.54 on 4 and 467 DF, p-value: < 2.2e-16

Call:

```
lm(formula = Total_Bsmt_SF ~ Bldg_Type + Year_Built + Lot_Area,
    data = Ames2)
```

Residuals:

	Min	1Q	Median	3Q	Max
	-810.32	-212.07	-5.72	233.88	1232.65

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	-1.176e+04	1.828e+03	-6.435	3.08e-10	***
Bldg_TypeDuplex	2.378e+02	6.529e+01	3.642	0.000301	***
Bldg_TypeTwnhs	-4.120e+02	7.745e+01	-5.319	1.62e-07	***
Bldg_TypeTwnhsE	-1.265e+02	8.035e+01	-1.575	0.115942	
Year_Built	6.509e+00	9.476e-01	6.868	2.09e-11	***
Lot_Area	7.793e-03	1.960e-03	3.977	8.10e-05	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 322.1 on 466 degrees of freedom

Multiple R-squared: 0.3558, Adjusted R-squared: 0.3489

F-statistic: 51.48 on 5 and 466 DF, p-value: < 2.2e-16

Analysis of Variance Table

Model 1: Total\_Bsmt\_SF ~ Bldg\_Type + Year\_Built

Model 2: Total\_Bsmt\_SF ~ Bldg\_Type + Year\_Built + Lot\_Area

	Res.Df	RSS	Df	Sum of Sq	F	Pr(>F)
1	467	49980160				
2	466	48339705	1	1640455	15.814	8.099e-05 ***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

- k. Construct a confidence interval and a prediction interval for the basement area of a Twnhs built in 1980, with a lot Area of 7300. Explain what these two intervals mean. (6 points)

```
# Confidence interval
predict(linmod2,
       newdata = data.frame(Bldg_Type = "Twnhs",
                           Year_Built = 1980,
                           Lot_Area = 7300),
       interval = "confidence")

# Prediction interval
predict(linmod2,
       newdata = data.frame(Bldg_Type = "Twnhs",
                           Year_Built = 1980,
                           Lot_Area = 7300),
       interval = "prediction")

# Explanation:
# The confidence interval is the uncertainty around the mean at each ...
# total basement area. The expected values of total basement area...
# are expected lie within this interval.
# Whereas the prediction interval is the range of likely values...
# the expected value of the total basement area could take.
```

	fit	lwr	upr
1	768.7589	702.1423	835.3755

	fit	lwr	upr
1	768.7589	132.3605	1405.157

- l. Now build a linear mixed model, linmod3, for Total\_Bsmt\_SF as a function of Year\_Built, MS\_Zoning and Bldg\_Type. Use Neighborhood as random effect. What is the critical number to pull out from this, and what does it tell us? (4 points)

```
linmod3 <- lmer(formula = Total_Bsmt_SF ~
               Year_Built + MS_Zoning + Bldg_Type + (1|Neighborhood),
               data = Ames2)

# Critical number: 187.4
# What it tells us: This is the standard deviation of the effect...
# 'Neighborhood' has on Year_Built and MS_Zoning. In other words...
# Neighbourhood causes 187.4 variance between the Year_Built and...
# MS_Zoning variables.
```

- m. Construct 95% confidence intervals around each parameter estimate for linmod3. What does this tell us about the significance of the random effect? (3 points)

```
confint(linmod3)
```

```
# .sig01 = Is the random effect.  
# Since the range of the confidence interval does not include 0...  
# the random effect is significant.
```

	2.5 %	97.5 %
.sig01	114.916972	253.19244
.sigma	244.221572	278.77404
(Intercept)	-9699.022207	-595.99553
Year_Built	0.691417	5.33084
MS_ZoningResidential_High_Density	-254.190137	549.38121
MS_ZoningResidential_Low_Density	-91.829020	665.77648
MS_ZoningResidential_Medium_Density	-266.136920	487.72182
Bldg_TypeDuplex	145.073466	377.00462
Bldg_TypeTwnhs	-249.148897	102.22240
Bldg_TypeTwnhsE	-68.266514	263.18536

- n. Write out the full mathematical expression for the model in linmod2 and for the model in linmod3. Round to the nearest integer in all coefficients with modulus  $> 10$  and to three decimal places for coefficients with modulus  $< 10$ . (4 points)

```
linmod2
```

Call:

```
lm(formula = Total_Bsmt_SF ~ Bldg_Type + Year_Built + Lot_Area,  
    data = Ames2)
```

Coefficients:

(Intercept)	Bldg_TypeDuplex	Bldg_TypeTwnhs	Bldg_TypeTwnhsE
-1.176e+04	2.378e+02	-4.120e+02	-1.265e+02
Year_Built	Lot_Area		
6.509e+00	7.793e-03		

```
linmod3
```

Linear mixed model fit by REML ['lmerMod']

Formula:

```
Total_Bsmt_SF ~ Year_Built + MS_Zoning + Bldg_Type + (1 | Neighborhood)
```

Data: Ames2

REML criterion at convergence: 6566.758

Random effects:

Groups	Name	Std.Dev.
Neighborhood	(Intercept)	187.4
	Residual	261.8

Number of obs: 472, groups: Neighborhood, 27

Fixed Effects:

	(Intercept)	Year_Built
	-4890.652	2.876
MS_ZoningResidential_High_Density		MS_ZoningResidential_Low_Density
	148.504	288.369

MS_ZoningResidential_Medium_Density	Bldg_TypeDuplex
109.234	264.530
Bldg_TypeTwnhs	Bldg_TypeTwnhsE
-63.140	105.171

Mathematical expression for linmod2:

$$E(\text{TotalBsmtSF}) = -11760 + (6.509 \times \text{YearBuilt}) + (0.008 \times \text{LotArea}) \\ + (238 \times \text{isDuplex}) + (-412 \times \text{isTwnhs}) + (-127 \times \text{isTwnhsE})$$

Mathematical expression for linmod3:

$$E(\text{TotalBsmtSF}) = -4981 + (2.876 \times \text{YearBuilt}) + (149 \times \text{isHighDensity}) + (288 \times \text{isLowDensity}) \\ + (109 \times \text{isMediumDensity}) + (265 \times \text{isDuplex}) + (-63 \times \text{isTwnhs}) + (105 \times \text{isTwnhsE}) + U \\ U \sim N(0, 187) \\ \text{TotalBsmtSF} \sim N(E(\text{TotalBsmtSF}), 262)$$

### 3. Logistic Regression

a. Do the following:

- (i) Create a new dataset called “Ames3” that contains all data in “Ames” dataset plus a new variable “excellent\_heating” that indicates if the heating quality and condition “Heating\_QC” is excellent or not. (2 points)

```
Ames3 <- Ames %>%
  mutate(
    excellent_heating = if_else(Heating_QC == "Excellent", "Yes", "No")
  )

# The code below checks whether this new variable is correct.
Ames3 %>%
  summarise (
    Heating_QC,
    excellent_heating
  )
```

```
# A tibble: 2,930 x 2
  Heating_QC excellent_heating
  <fct>      <chr>
1 Fair      No
2 Typical   No
3 Typical   No
4 Excellent Yes
5 Good      No
6 Excellent Yes
7 Excellent Yes
8 Excellent Yes
9 Excellent Yes
10 Good     No
# ... with 2,920 more rows
```

- (ii) In “Ames3” dataset, remove all cases “3” and “4” corresponding to the Fireplaces variable. Remove all cases where Lot\_Frontage is greater than 130 or smaller than 20. Drop the unused levels from the dataset. (2 points)

```
Ames3 <- Ames3 %>%
  filter(Fireplaces != 3 & Fireplaces != 4) %>%
  filter(Lot_Frontage >= 20 & Lot_Frontage <= 130) %>%
  droplevels()
```

- (iii) Save “Fireplaces” as factor in “Ames3” dataset (1 point)

```
Ames3 <- Ames3 %>%
  mutate (
    Fireplaces = as.factor(Fireplaces)
  )
```

- (iv) Construct a logistic regression model glmod for excellent\_heating as a function of Lot\_Frontage and Fireplaces for the dataset “Ames3”. (2 points)

```
glmod <- glm(formula = as.factor(excellent_heating) ~ Lot_Frontage + Fireplaces,
             family = "binomial",
             data = Ames3)
glmod
```

```
Call:  glm(formula = as.factor(excellent_heating) ~ Lot_Frontage + Fireplaces,
          family = "binomial", data = Ames3)
```

Coefficients:

(Intercept)	Lot_Frontage	Fireplaces1	Fireplaces2
-0.769387	0.007018	0.796183	0.494887

Degrees of Freedom: 2399 Total (i.e. Null); 2396 Residual

Null Deviance: 3324

Residual Deviance: 3213 AIC: 3221

- b. Construct confidence bands for the variable excellent\_heating as a function of Lot\_Frontage for each number of Fireplaces (hint: create a new data frame for each number of Fireplaces). Colour these with different transparent colours for each number of Fireplaces and plot them together on the same axes. Put the actual data on the plot, coloured to match the bands, and jittered in position to make it possible to see all points. Ensure you have an informative main plot title, axes labels and a legend. (7 points)

```
ilink <- family(glmod)$linkinv

newf <- with(
  Ames3,
  data.frame(
    Lot_Frontage = seq(min(Ames3$Lot_Frontage),
                      max(Ames3$Lot_Frontage),
                      length = 100),
```

```

Fireplaces))

newf <- cbind(newf,
              predict(
                glmod,
                newf,
                type = "link",
                se.fit = TRUE)[1:2])

newf <- transform(newf,
                  Fitted = ilink(fit),
                  Upper = ilink(fit + (1.96 * se.fit)),
                  Lower = ilink(fit - (1.96 * se.fit)))

#### Original (actual data without jitter) ####
figure_8a <- ggplot(Ames3,
                    aes(x = Lot_Frontage,
                        y = as.numeric(as.factor(excellent_heating)) - 1,
                        colour = Fireplaces)) + # so the points are coloured.
  geom_ribbon(data = newf,
             aes(
               ymin = Lower,
               ymax = Upper,
               x = Lot_Frontage,
               fill = Fireplaces,
               colour = Fireplaces),
             alpha = 0.2,
             inherit.aes = FALSE) +
  geom_line(data = newf,
            aes(y = Fitted,
                x = Lot_Frontage,
                group = Fireplaces,
                colour = Fireplaces)) +
  geom_point(alpha = 0.5) + # so all points can be seen.
  labs(
    title = "Without jitter",
    subtitle = "The probability and corresponding confidence interval of
Lot Frontage obtaining an excellent rating\n based on each Fireplace type",
    x = "Lot Frontage",
    y = "Probability of excellent heating") +
  theme_classic()

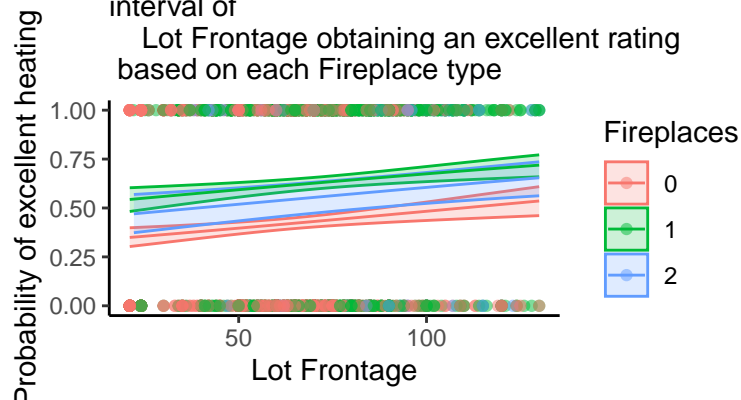
figure_8a

```

## Without jitter

The probability and corresponding confidence interval of

Lot Frontage obtaining an excellent rating based on each Fireplace type



# TO SEE A CLEARER VERSION OF THIS PLOT PLEASE OPEN IT IN THE .RMD FILE.

#### Final (actual data with jitter) ####

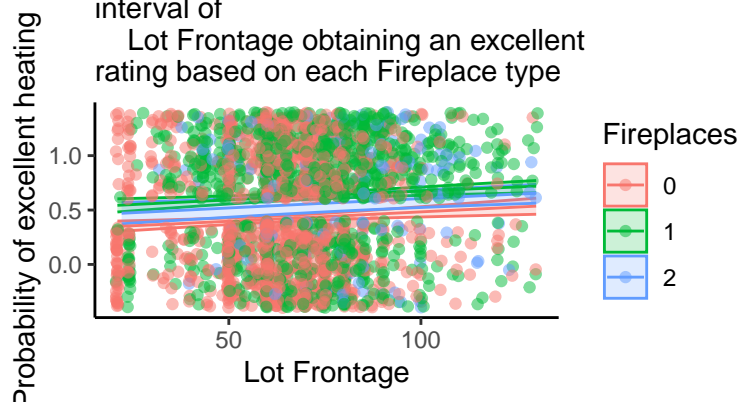
```
figure_8b <- ggplot(Ames3,
  aes(x = Lot_Frontage,
      y = as.numeric(as.factor(excellent_heating)) - 1,
      colour = Fireplaces)) + # so the points are coloured.
  geom_ribbon(data = newf,
    aes(
      ymin = Lower,
      ymax = Upper,
      x = Lot_Frontage,
      fill = Fireplaces,
      colour = Fireplaces),
    alpha = 0.2,
    inherit.aes = FALSE) +
  geom_line(data = newf,
    aes(y = Fitted,
      x = Lot_Frontage,
      group = Fireplaces,
      colour = Fireplaces)) +
  geom_point(position = "jitter", alpha = 0.5) + # so all points can be seen.
  labs(
    title = "With jitter",
    subtitle = "The probability and corresponding confidence\ninterval of
    Lot Frontage obtaining an excellent\nrating based on each Fireplace type",
    x = "Lot Frontage",
    y = "Probability of excellent heating") +
  theme_classic()

figure_8b
```

With jitter

The probability and corresponding confidence interval of

Lot Frontage obtaining an excellent rating based on each Fireplace type



*# TO SEE A CLEARER VERSION OF THIS PLOT PLEASE OPEN IT IN THE .RMD FILE.*

- c. Split the data using `set.seed(120)` and rebuild the model on 80% of the data. Cross validate on the remaining 20%. Plot the ROCs for both data and comment on your findings. (6 points)

```
set.seed(120)
training.samples <- c(Ames3$excellent_heating) %>%
  createDataPartition(p = 0.8, list = FALSE)
train.data <- Ames3[training.samples, ]
test.data <- Ames3[-training.samples, ]

train.model <- glm(formula = as.factor(excellent_heating) ~ Lot_Frontage + Fireplaces,
  family = "binomial",
  data = train.data)

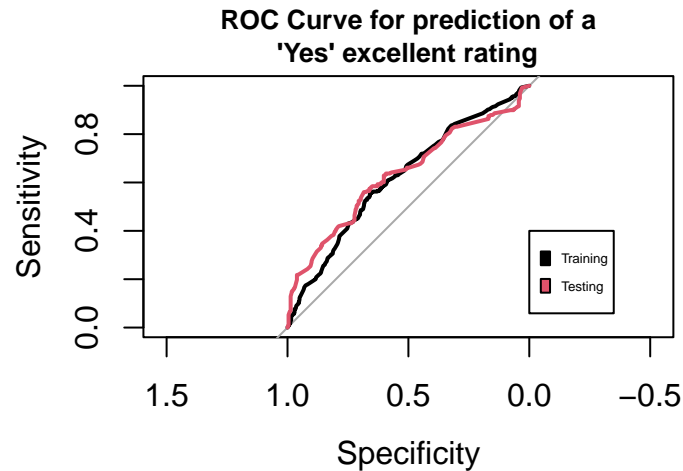
predtrain <- predict(train.model, type = "response")
predtest <- predict(train.model, newdata = test.data, type = "response")

roctrain <- roc(response = train.data$excellent_heating,
  predictor = predtrain,
  plot = TRUE,
  main = "ROC Curve for prediction of a\n'Yes' excellent rating",
  cex.main = 0.85,
  auc = TRUE)

roc(response = test.data$excellent_heating,
  predictor = predtest,
  plot = TRUE,
  auc = TRUE,
  add = TRUE,
  col = 2)

legend(0, 0.4, legend = c("Training", "Testing"), fill = 1:2, cex = 0.4)
```





*# The testing and training ROC curves are similar in shape and closely overlap.  
# Therefore this is a good indication the testing data is not...  
# over fitted to the training data.*

## 4. Multinomial Regression

- a. For the dataset “Ames”, create a model `multregmod` to predict `BsmtFin_Type_1` from `Total_Bsmt_SF` and `Year_Remod_Add`. (3 points)

```
multregmod <- multinom(formula = BsmtFin_Type_1 ~ Total_Bsmt_SF + Year_Remod_Add,
                        data = Ames)
multregmod
```

```
# weights: 28 (18 variable)
initial value 5701.516737
iter 10 value 4611.614897
iter 20 value 4159.256251
iter 30 value 4153.561922
iter 40 value 4150.324235
iter 50 value 4146.549269
iter 60 value 4144.509436
iter 70 value 4144.474970
final value 4144.474825
converged
Call:
multinom(formula = BsmtFin_Type_1 ~ Total_Bsmt_SF + Year_Remod_Add,
          data = Ames)
```

Coefficients:

	(Intercept)	Total_Bsmt_SF	Year_Remod_Add
BLQ	34.465254	6.282504e-05	-0.017706708
GLQ	-105.324418	1.030040e-03	0.052676145
LwQ	39.566891	1.243787e-05	-0.020550529
No_Basement	4.876103	-1.729079e-01	0.004007989
Rec	56.710979	1.596801e-06	-0.028929851

Unf                    -29.377212   -6.987213e-04      0.015514051

Residual Deviance: 8288.95

AIC: 8324.95

- b. Write out the formulas for this model in terms of  $P(\text{No\_Basement})$ ,  $P(\text{Unf})$ ,  $P(\text{Rec})$ ,  $P(\text{BLQ})$ ,  $P(\text{GLQ})$ ,  $P(\text{LwQ})$ ,  
You may round coefficients to 3 dp. (4 points)

The probabilities in terms of logit are:

$$\text{logit}(P(\text{BLQ})) = 34.465 + ((6.283 \times 10^{-5}) \times \text{TotalBmstSF}) + (-0.018 \times \text{YearRemodAdd})$$

$$\text{logit}(P(\text{GLQ})) = -105.324 + (0.001 \times \text{TotalBmstSF}) + (0.053 \times \text{YearRemodAdd})$$

$$\text{logit}(P(\text{LwQ})) = 39.567 + ((1.244 \times 10^{-5}) \times \text{TotalBmstSF}) + (-0.021 \times \text{YearRemodAdd})$$

$$\text{logit}(P(\text{NoBasement})) = 4.876 + (-0.173 \times \text{TotalBmstSF}) + (0.004 \times \text{YearRemodAdd})$$

$$\text{logit}(P(\text{Rec})) = 56.711 + ((1.597 \times 10^{-6}) \times \text{TotalBmstSF}) + (-0.029 \times \text{YearRemodAdd})$$

$$\text{logit}(P(\text{Unf})) = -29.377 + ((-6.99 \times 10^{-4}) \times \text{TotalBmstSF}) + (0.016 \times \text{YearRemodAdd})$$

Where:

```
# Checked levels to find the base case.
levels(Ames$BsmtFin_Type_1)
```

```
[1] "ALQ"      "BLQ"      "GLQ"      "LwQ"      "No_Basement"
[6] "Rec"      "Unf"
```

$$P(\text{ALQ}) = 1 - P(\text{BLQ}) - P(\text{GLQ}) - P(\text{LwQ}) - P(\text{NoBasement}) - P(\text{Rec}) - P(\text{Unf})$$

- c. Evaluate the performance of this model using a confusion matrix and by calculating the sum of sensitivities for the model. Comment on your findings. (4 points)

```
multitable <- table(Ames$BsmtFin_Type_1,
                    predict(multregmod,
                            type = "class"))

names(dimnames(multitable)) <- list("Actual", "Predicted")

multitable
```

	Predicted						
Actual	ALQ	BLQ	GLQ	LwQ	No_Basement	Rec	Unf
ALQ	1	0	117	0	0	18	293
BLQ	0	0	50	0	0	30	189
GLQ	1	0	579	0	0	2	277
LwQ	1	0	38	0	0	30	85
No_Basement	0	0	0	0	80	0	0
Rec	3	0	31	0	0	46	208
Unf	6	0	291	0	0	76	478

Evaluation:

First calculate the sensitivity and specificity for each category:

$$\text{Sensitivity}(\text{ALQ}) = \frac{1}{1 + 117 + 18 + 293} = 0.233\%$$

$$\text{Sensitivity}(\text{BLQ}) = \frac{0}{50 + 30 + 189} = 0\%$$

$$\text{Sensitivity}(\text{GLQ}) = \frac{579}{1 + 579 + 2 + 277} = 67.404\%$$

$$\text{Sensitivity}(\text{LwQ}) = \frac{0}{1 + 38 + 30 + 85} = 0\%$$

$$\text{Sensitivity}(\text{NoBasement}) = \frac{80}{80} = 100\%$$

$$\text{Sensitivity}(\text{Rec}) = \frac{46}{3 + 31 + 46 + 208} = 15.972\%$$

$$\text{Sensitivity}(\text{Unf}) = \frac{478}{6 + 291 + 76 + 478} = 56.169\%$$

2. Then calculating the sum of sensitivities for the model:

```
# Calculated manually:
sum_of_sensitivites <- 0.002331002 + 0.6740396 + 1 + 0.1597222 + 0.5616921

sum_of_sensitivites

# Checked by automated sum of sensitivities:
ss_check <- multitable[1,1]/sum(Ames$BsmtFin_Type_1=="ALQ") +
  multitable[2,2]/sum(Ames$BsmtFin_Type_1=="BLQ") +
  multitable[3,3]/sum(Ames$BsmtFin_Type_1=="GLQ") +
  multitable[4,4]/sum(Ames$BsmtFin_Type_1=="LwQ") +
  multitable[5,5]/sum(Ames$BsmtFin_Type_1=="No_Basement") +
  multitable[6,6]/sum(Ames$BsmtFin_Type_1=="Rec") +
  multitable[7,7]/sum(Ames$BsmtFin_Type_1=="Unf")

ss_check
# Since these two values are the same the sum of the sensitivities is correct.

# Comments: Sensitivities per category ->
# Generally the model does a better job at predicting GLQ, No_Basement and Unf.
# Particularly No_Basement which predictions were correct 100% of the time.
# In comparison ALQ and Rec were much less likely to be predicted correctly,
# with sensitivities of 0.002331002 and 0.1597222 respectively.
# The model was worst at predicting BLQ and LwQ with sensitivities of 0.

# Comments: Sum of sensitivities ->
# The sum of the sensitivities is 2.397785.
```

```
# This is less than 7 which would be the sum of sensitivities...
# for a perfect model. The model is not very good overall...
# since the sum of sensitivities is well below the...
# perfect model value. Instead only 34% (2.s.f)...
# of all categories are correctly predicted overall.
```

```
[1] 2.397785
[1] 2.397785
```

## 5. Poisson/quasipoisson Regression

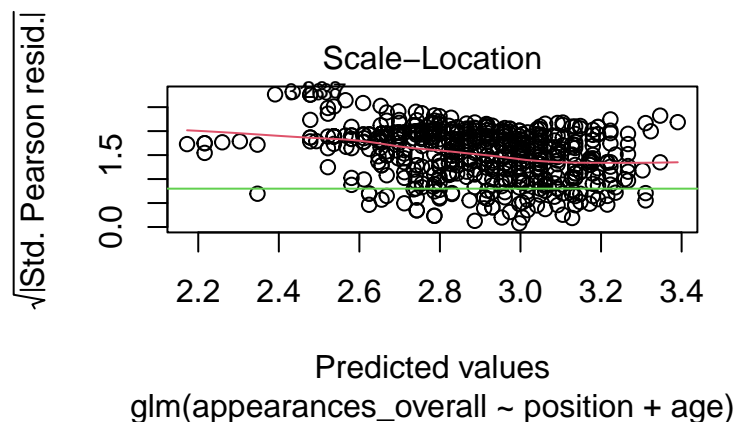
- a. For the “footballer\_data” dataset, create a model `appearances_mod` to predict the total number of overall appearances a player had based on position and age. (2 points)

```
appearances_mod <- glm(formula = appearances_overall ~ position + age,
                        data = footballer_data,
                        family = "poisson")
```

- b. Check the assumption of the model using a diagnostic plot and comment on your findings. (3 points)

```
# The assumption that needs to be checked is whether the variance = mean...
# i.e., the dispersion assumption.
```

```
plot(appearances_mod, which = 3)
abline(h = 0.8, col = 3)
```



```
# Findings:
# The red line is not flat, and it is consistently above 0.8.
# This suggests over dispersion in the data that decreases...
# somewhat linearly as the prediction increases. Therefore...
# the variance is not equal to the mean.

# Since the dispersion appears to roughly be a linear function...
```

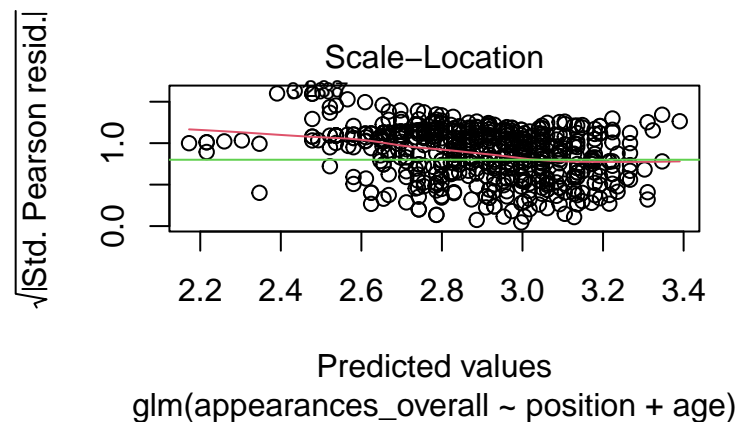
```

# of the mean, it is possible a quasipoisson model may...
# be a better fit and reduce the magnitude of over dispersion.

# This was tested below:
appearances_mod2 <- glm(formula = appearances_overall ~ position + age,
                        data = footballer_data,
                        family = "quasipoisson")

plot(appearances_mod2, which = 3)
abline(h = 0.8, col = 3)

```



```

# And as expected the over dispersion reduces. This indicates the...
# the quasipoisson is more suitable than the poisson model.

```

- c. What do the coefficients of the model tell us about? which position has the most appearances? How many times more appearances do forwards get on average than goalkeepers? (3 points)

```

# Used the quasipoisson model (since it has less over dispersion):
summary(appearances_mod2)

# The coefficients of the model tells us...
# Age: Every increase of appearances overall by 1...
#       is due to an increase in age by a rate of 0.043704.
# Position: And it tells us the rate of change of each...
#           positions relative to the base case position - the Defender.

# Position with the most appearances: Midfielder

# How many more times forwards appear than goalkeepers: 1.6 times
# Rounded up its 2 times.
forward_goalkeeper_ratio <- exp(0.110606)/exp(-0.364605)
forward_goalkeeper_ratio

```

Call:

```
glm(formula = appearances_overall ~ position + age, family = "quasipoisson",
     data = footballer_data)
```

Deviance Residuals:

Min	1Q	Median	3Q	Max
-7.5377	-3.5215	0.0351	2.1892	6.1853

Coefficients:

	Estimate	Std. Error	t value	Pr(> t )	
(Intercept)	1.575316	0.223980	7.033	5.90e-12	***
positionForward	0.110606	0.082097	1.347	0.17844	
positionGoalkeeper	-0.364605	0.121975	-2.989	0.00292	**
positionMidfielder	0.118259	0.069717	1.696	0.09039	.
age	0.043704	0.007153	6.110	1.87e-09	***

---

Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1

(Dispersion parameter for quasipoisson family taken to be 8.946343)

Null deviance: 6539.7 on 564 degrees of freedom  
 Residual deviance: 6114.4 on 560 degrees of freedom  
 AIC: NA

Number of Fisher Scoring iterations: 5

[1] 1.608354