R\_Sydney\_Property\_Analysis

Harry

20/03/2021

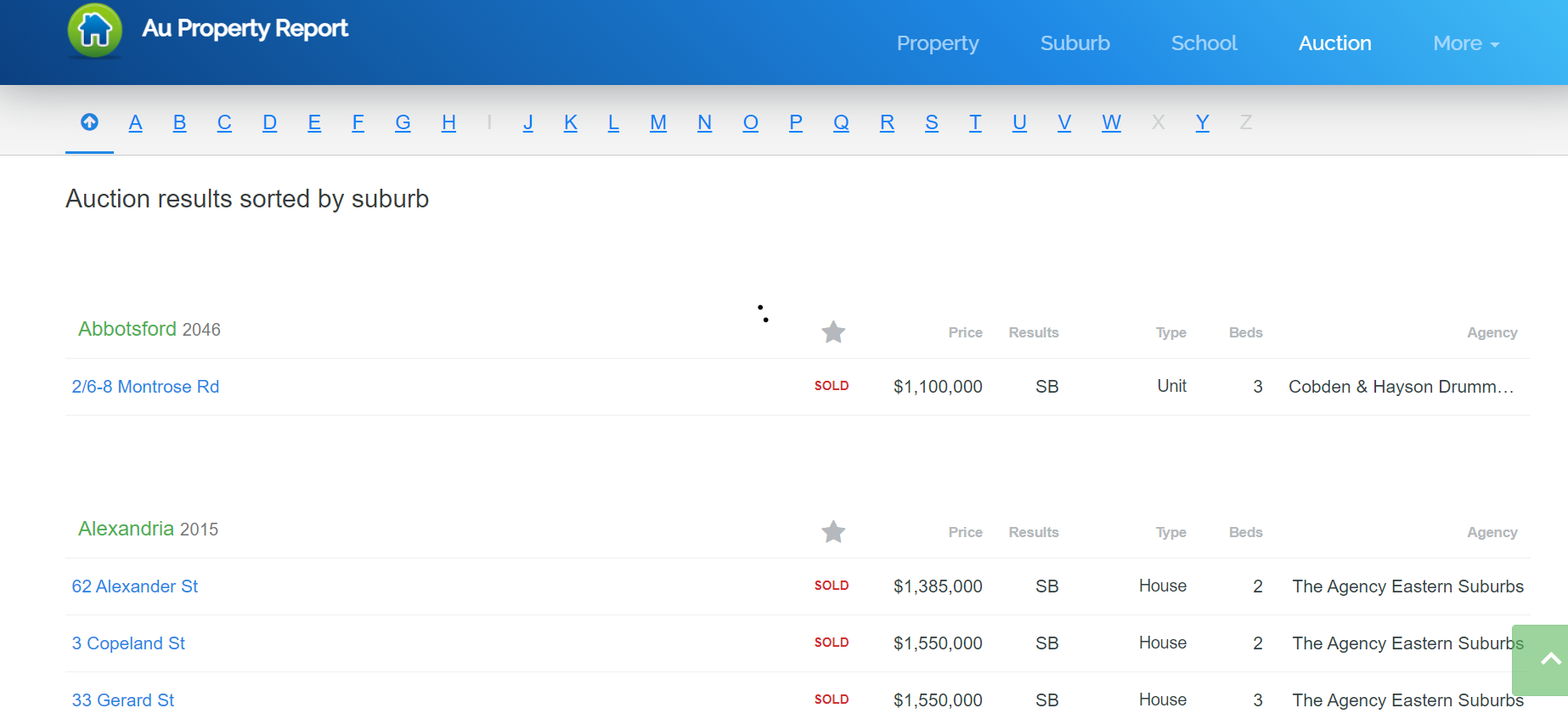
## Introduction

The purpose of this document is to investigate whether commonly listed property features influence the price of Sydney properties based on sold property information obtained from the following website: <https://www.aupropertyreport.com/auction-results/NSW/2021-03-14/>. We explore the process of collecting, cleaning, manipulating and visualising data using R and briefly Python.

## Setting up the R environment

## Data Collection / webscraping

The following fields are collected: 1. Price 2. Agent Company 3. Address 4. Number of bedrooms 5. Property type - House or unit or other? 6. Auction Result - sold before auction or sold at auction 7. Link to further information on property 8. Postcode



# Get URL  
  
URL <- seq(Start\_date %>% ymd, End\_date %>% ymd, 91) %>% as.character() %>%   
 purrr::map(function(n)paste0("https://www.aupropertyreport.com/auction-results/NSW/", n,"/") %>% read\_html())  
  
# Page <- read\_html("https://www.aupropertyreport.com/auction-results/NSW/2021-03-14/")  
Page <- URL  
  
Get\_Prop\_info <- function(Page){tibble(  
 Price = Page %>%   
 html\_nodes("td:nth-child(2)") %>%   
 html\_text(),  
  
 Agent = Page %>%   
 html\_nodes("td:nth-child(6)") %>%   
 html\_text(),  
  
 Address = Page %>%   
 html\_nodes("td:nth-child(1)") %>%   
 html\_text(),  
  
 Beds = Page %>%   
 html\_nodes("td:nth-child(5)") %>%   
 html\_text(),  
  
 Type = Page %>%   
 html\_nodes("td:nth-child(4)") %>%   
 html\_text(),  
  
 Result = Page %>%   
 html\_nodes("td:nth-child(3)") %>%   
 html\_text(),  
   
 Link\_info = Page %>%   
 html\_nodes("td a") %>%   
 html\_attr("href")  
)}  
  
Property\_data <- URL %>%   
 purrr::map(function(n)Get\_Prop\_info(n)) %>%   
 bind\_rows()  
  
Get\_PostCode <- function(Page){tibble(  
 Address = Page %>%   
 html\_nodes("tr > :nth-child(1)") %>%   
 html\_text()  
) %>%   
 mutate(  
 PostCode = Address %>% substr(nchar(Address)-4,nchar(Address)) %>% as.numeric()  
 )}  
  
Full\_Address <- URL %>%   
 purrr::map(function(n)Get\_PostCode(n)) %>%   
 bind\_rows()   
  
PostCode <- Full\_Address %>%   
 pull(PostCode)

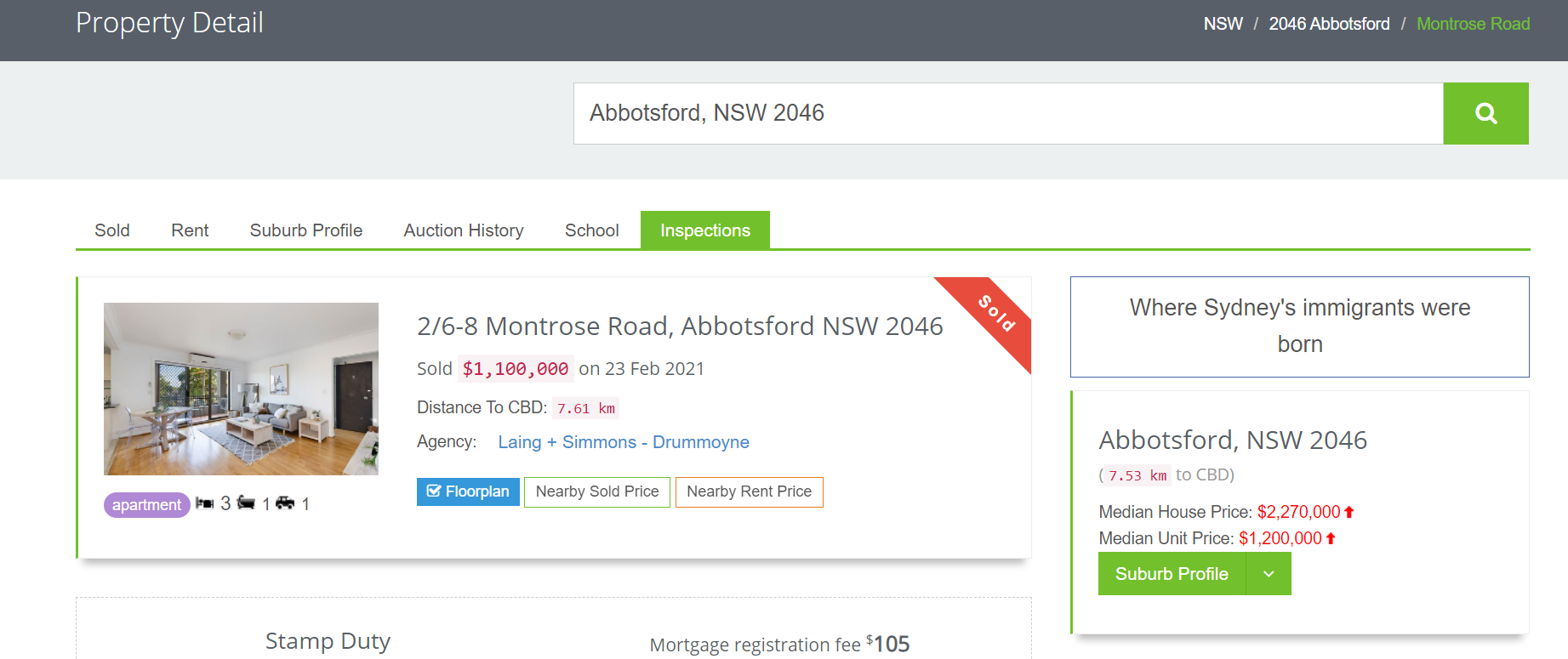
# Python and R integration

Post code information is used by the website to group properties together. Only the first property per postcode were web scraped into the tabular data above. To ensure postcodes are available for all properties, we use python to loop through the data such that missing postcodes are assigned the correct previous record’s postcode.

PostCode = r.PostCode  
x = []  
for n in PostCode:  
 if n > 1:  
 y = n  
 x.append(y)  
 else:  
 x.append(y)

# Collecting additional data

Within each link, we can click on the property sold to discover more information. This new page contains commonly listed property features that can be scraped: 9. Number of bathrooms 10. Number of car spots in the garage 11. Size of land in square metres 12. Distance to the CBD in kms 13. Date of property sold 14. Nearest train station name 15.Distance to the nearest train station in kms.



Note that for some of the above fields, some properties may not disclose information, resulting in missing values.

# Get post codes from Python variable  
Full\_Address1 <- Full\_Address %>%   
 bind\_cols(Post\_Code = py$x) %>%   
 select(-PostCode)  
   
# Add post codes to property data  
Property\_data1 <- Property\_data %>%   
 left\_join(Full\_Address1, by = c("Address")) %>%   
 mutate(Index = 1:dim(Property\_data[1]))  
  
# Obtaining additional information function  
Add\_info <- function(link){  
 Info = link %>%  
 read\_html()  
   
 House\_rooms = Info %>%   
 html\_nodes(".overflow-hidden .col-md-4 li") %>%   
 html\_text()  
   
 Land\_size = Info %>%   
 html\_nodes(".col-md-8 .list-unstyled li:nth-child(2)") %>%   
 html\_text()  
   
 Dist\_CBD = Info %>%   
 html\_nodes(".list-unstyled code") %>%   
 html\_text()  
   
 Date\_sold = Info %>%   
 html\_nodes(".overflow-hidden h5") %>%   
 html\_text()  
   
 Nearest\_train\_name = Info %>%   
 html\_nodes(".margin-bottom-30 .list-unstyled .margin-bottom-10:nth-child(1)") %>%   
 html\_text()  
   
 Nearest\_train\_dist = Info %>%   
 html\_nodes(".margin-bottom-10:nth-child(1) .rounded-2x") %>%   
 html\_text()  
   
 Additional = list(House\_rooms,Land\_size,Dist\_CBD,Date\_sold,Nearest\_train\_name,Nearest\_train\_dist)  
   
 return(Additional)  
   
}  
  
# Loop into each property and scrape additional information  
Additional\_info <- list()  
for (info in Property\_data1$Link\_info){  
 Additional = Add\_info(info) %>%   
 list()  
 Additional\_info = Additional\_info %>%   
 append(Additional)  
 Sys.sleep(1)  
}  
  
# For each additional information obtained, manipulate it into tabular format  
Additional\_info2 <- 1:length(Additional\_info) %>%   
 purrr::map(function(n)tibble(  
 House\_rooms = Additional\_info[[n]][[1]] %>% as.character(),  
   
 Land\_size = Additional\_info[[n]][[2]] %>% as.character(),  
   
 Dist\_CBD = Additional\_info[[n]][[3]] %>% as.character(),  
   
 Date\_sold = Additional\_info[[n]][[4]] %>% as.character(),  
   
 Nearest\_train\_name = Additional\_info[[n]][[5]] %>% as.character(),  
   
 Nearest\_train\_dist = Additional\_info[[n]][[6]] %>% as.character(),  
   
 Index = n  
 )) %>%   
 bind\_rows()

## Data Cleaning process

Cleaning the data to specify the correct formats. This means: 1. Missing numbers will be left as is 2. Data types for each field are assigned 3. For numbers, this means removing commas and dollar signs and transforming it from text to number data type 4. Other cases involves removing units (kms) and words.

glimpse(Property\_data1)  
  
# Joining main data set with additional information data set  
Property\_data2 <- Property\_data1 %>%   
 mutate(Index = 1:dim(Property\_data1)[1]) %>%   
 left\_join(Additional\_info2, by = c("Index"))  
  
glimpse(Property\_data2)  
  
# Cleaning data  
Property\_data3 <- Property\_data2 %>%  
 mutate(Price = Price %>% str\_remove\_all("[$,]") %>% as.numeric(),  
 Agent = Agent %>% as.factor(),  
 Beds = Beds %>% as.factor(),  
 Type = Type %>% as.factor(),  
 Result = Result %>% as.factor(),  
 Post\_Code = Post\_Code %>% as.factor()  
 ) %>%   
 separate(House\_rooms,c("Empty","Type2","Beds2","Bathrooms2","Cars2"), fill = "right") %>%   
 mutate(Date\_sold = Date\_sold %>% substr(nchar(Date\_sold)-10,nchar(Date\_sold)) %>% dmy,  
 Nearest\_train\_name = Nearest\_train\_name %>% substr(nchar(Nearest\_train\_dist)+3,nchar(Nearest\_train\_name))) %>%   
 mutate(Dist\_CBD = Dist\_CBD %>% substr(1,nchar(Dist\_CBD)-3) %>% as.numeric(),  
 Nearest\_train\_dist = Nearest\_train\_dist %>% substr(1,nchar(Nearest\_train\_dist)-3) %>% as.numeric())  
  
# Viewing data  
glimpse(Property\_data3)  
summary(Property\_data3 %>% filter(Price %>% is.na == FALSE))  
mean\_price = mean(Property\_data3$Price,na.rm = TRUE)  
  
# saving the data  
setwd(dir\_proj)  
write\_rds(Property\_data3,file = "Property\_data.rds")

## Exploratory data analysis

Create graphs to visually inspect whether there is a relationship between commonly list features for property advertisements vs the price sold. 1. Histogram of property prices 2. Property prices vs date sold 3. Property prices vs Number of beds violin plot 4. Property price heatmap

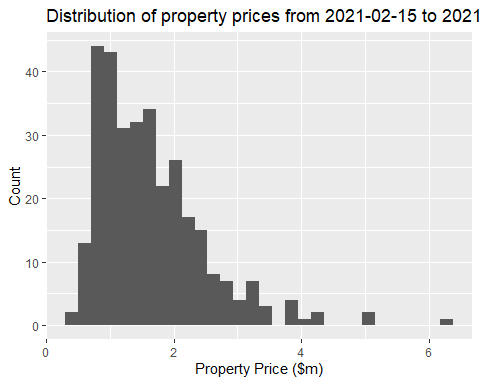
# Optional: reading in data  
setwd(dir\_proj)  
Property\_data3 <- read\_rds("Property\_data.rds")  
  
# View of data  
head(Property\_data3)

## # A tibble: 6 x 17  
## Price Agent Address Beds Type Result Post\_Code Index Type2 Beds2 Bathooms2  
## <dbl> <fct> <chr> <fct> <fct> <fct> <fct> <int> <chr> <chr> <chr>   
## 1 1100000 Cobd~ 2/6-8 ~ 3 Unit SB 2046 1 "" apar~ 3   
## 2 1385000 The ~ 62 Ale~ 2 House SB 2015 2 "" house 2   
## 3 1550000 The ~ 3 Cope~ 2 House SB 2015 3 "" house 2   
## 4 1550000 The ~ 33 Ger~ 3 House SB 2015 4 <NA> <NA> <NA>   
## 5 850000 MGM ~ 32/1 S~ 2 Unit S 2015 5 "" apar~ 2   
## 6 1920000 KORE~ 7 Tama~ 5 House S 2234 6 "" house 5   
## # ... with 6 more variables: Cars2 <chr>, Land\_size <chr>, Dist\_CBD <dbl>,  
## # Date\_sold <date>, Nearest\_train\_name <chr>, Nearest\_train\_dist <dbl>

# Histogram of property prices  
Property\_data3 %>%   
 ggplot(aes(Price/1000000)) +  
 geom\_histogram() +  
 ggtitle(paste0("Distribution of property prices from ",Start\_date," to ",End\_date)) +  
 xlab("Property Price ($m)") + ylab("Count")

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

## Warning: Removed 90 rows containing non-finite values (stat\_bin).

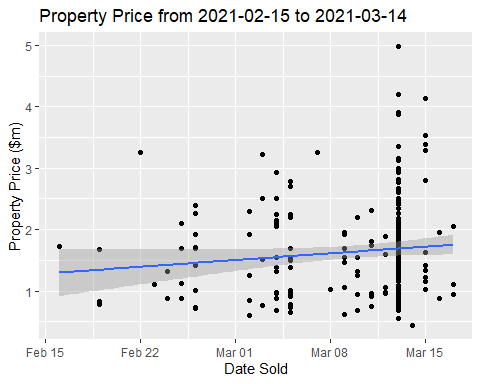


# Property price over time  
Property\_data3 %>%   
 filter(Date\_sold %>% year() >= Year) %>%   
 ggplot(aes(Date\_sold,Price/1000000)) +  
 geom\_point() +  
 geom\_smooth(method = "lm") +  
 ggtitle(paste0("Property Price from ",Start\_date," to ",End\_date)) +  
 xlab("Date Sold") + ylab("Property Price ($m)")

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

## Warning: Removed 57 rows containing non-finite values (stat\_smooth).

## Warning: Removed 57 rows containing missing values (geom\_point).



lm(Price ~ Date\_sold,data = Property\_data3) %>%   
 summary()

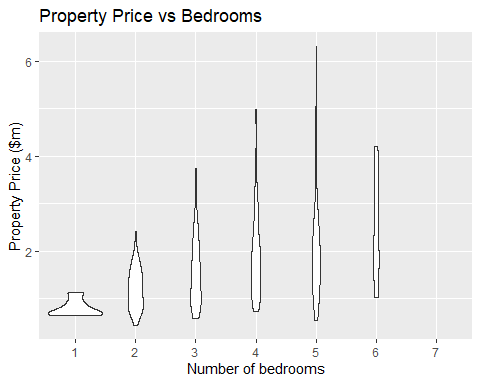
##   
## Call:  
## lm(formula = Price ~ Date\_sold, data = Property\_data3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1194098 -630134 -159462 431572 3347947   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)  
## (Intercept) 788664.13 1229961.60 0.641 0.522  
## Date\_sold 45.10 66.16 0.682 0.496  
##   
## Residual standard error: 781100 on 236 degrees of freedom  
## (170 observations deleted due to missingness)  
## Multiple R-squared: 0.001966, Adjusted R-squared: -0.002263   
## F-statistic: 0.4648 on 1 and 236 DF, p-value: 0.4961

Property\_data3 %>%  
 filter(Date\_sold %>% year() >= Year) %>%   
 group\_by(Month\_sold = paste0(year(Date\_sold)," ",case\_when(month(Date\_sold) == 1 ~ "Jan",  
 month(Date\_sold) == 2 ~ "Feb",  
 month(Date\_sold) == 3 ~ "Mar",  
 month(Date\_sold) == 4 ~ "Apr",  
 month(Date\_sold) == 5 ~ "May",  
 month(Date\_sold) == 6 ~ "Jun",  
 month(Date\_sold) == 7 ~ "Jul",  
 month(Date\_sold) == 8 ~ "Aug",  
 month(Date\_sold) == 9 ~ "Sep",  
 month(Date\_sold) == 10 ~ "Oct",  
 month(Date\_sold) == 11 ~ "Nov",  
 month(Date\_sold) == 12 ~ "Dec"))) %>%   
 summarise(Count = n())

## # A tibble: 2 x 2  
## Month\_sold Count  
## <chr> <int>  
## 1 2021 Feb 32  
## 2 2021 Mar 257

# Property price vs number of bedrooms  
Property\_data3 %>%   
 ggplot(aes(x=Beds,y=Price/1000000)) +  
 geom\_violin() +  
 ggtitle("Property Price vs Bedrooms") +  
 xlab("Number of bedrooms") + ylab("Property Price ($m)")

## Warning: Removed 90 rows containing non-finite values (stat\_ydensity).



lm(Price ~ Beds,data = Property\_data3) %>%   
 summary()

##   
## Call:  
## lm(formula = Price ~ Beds, data = Property\_data3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1555869 -554885 -115239 381420 4201131   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 798333 321572 2.483 0.013570 \*   
## Beds2 417311 334527 1.247 0.213165   
## Beds3 809055 329784 2.453 0.014704 \*   
## Beds4 1067890 334527 3.192 0.001556 \*\*   
## Beds5 1310536 343775 3.812 0.000166 \*\*\*  
## Beds6 1638833 454772 3.604 0.000365 \*\*\*  
## Beds7 1054167 643145 1.639 0.102207   
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 787700 on 311 degrees of freedom  
## (90 observations deleted due to missingness)  
## Multiple R-squared: 0.1512, Adjusted R-squared: 0.1349   
## F-statistic: 9.236 on 6 and 311 DF, p-value: 0.000000002554

Property\_data3 %>%  
 filter(Date\_sold %>% year() >= Year) %>%   
 group\_by(Number\_of\_bedrooms = Beds) %>%   
 summarise(Count = n())

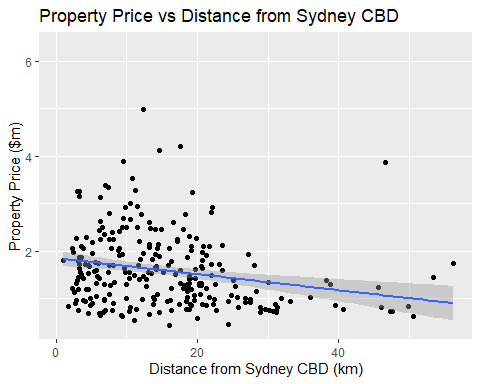
## # A tibble: 7 x 2  
## Number\_of\_bedrooms Count  
## <fct> <int>  
## 1 1 6  
## 2 2 65  
## 3 3 103  
## 4 4 65  
## 5 5 44  
## 6 6 4  
## 7 7 2

# Relationship between property price vs distance from the CBD  
Property\_data3 %>%   
 ggplot(aes(x=Dist\_CBD,y=Price/1000000)) +  
 geom\_point() +  
 geom\_smooth(method = "lm") +  
 ggtitle("Property Price vs Distance from Sydney CBD") +  
 xlab("Distance from Sydney CBD (km)") + ylab("Property Price ($m)")

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

## Warning: Removed 151 rows containing non-finite values (stat\_smooth).

## Warning: Removed 151 rows containing missing values (geom\_point).



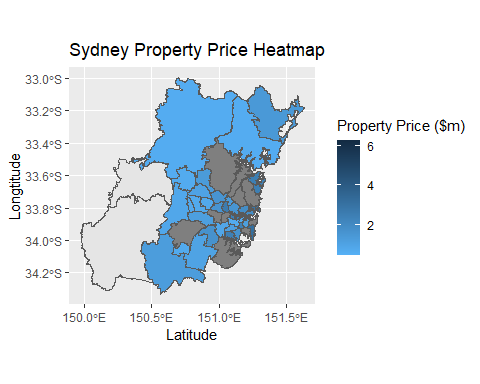
lm(Price ~ Dist\_CBD,data = Property\_data3) %>%   
 summary()

##   
## Call:  
## lm(formula = Price ~ Dist\_CBD, data = Property\_data3)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -1144575 -512856 -170167 413202 3333141   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 1854935 82711 22.427 < 0.0000000000000002 \*\*\*  
## Dist\_CBD -16917 4400 -3.845 0.000152 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 749700 on 255 degrees of freedom  
## (151 observations deleted due to missingness)  
## Multiple R-squared: 0.0548, Adjusted R-squared: 0.05109   
## F-statistic: 14.78 on 1 and 255 DF, p-value: 0.0001523

# Add NSW map  
setwd(dir\_proj)  
#NSW\_map <- read\_sf("MB\_2016\_NSW.shp")  
AUST\_map <- read\_sf("SA3\_2016\_AUST.shp")  
PostCode\_Mapping <- read\_csv("australian\_postcodes.csv")

##   
## -- Column specification --------------------------------------------------------  
## cols(  
## .default = col\_double(),  
## postcode = col\_character(),  
## locality = col\_character(),  
## state = col\_character(),  
## dc = col\_character(),  
## type = col\_character(),  
## status = col\_character(),  
## sa3name = col\_character(),  
## sa4name = col\_character(),  
## region = col\_character(),  
## SA2\_NAME\_2016 = col\_character(),  
## SA3\_NAME\_2016 = col\_character(),  
## SA4\_NAME\_2016 = col\_character()  
## )  
## i Use `spec()` for the full column specifications.

# Reduce AUST map to NSW  
NSW <- AUST\_map %>%   
 filter(STE\_NAME16 == "New South Wales") %>%   
 filter(GCC\_NAME16 %in% c("Greater Sydney","No usual address (NSW)"))  
  
# Map Post codes to SA3 codes  
Property\_map <- Property\_data3 %>%   
 left\_join(PostCode\_Mapping, by = c("Post\_Code" = "postcode")) %>%   
 select(names(Property\_data3),SA3\_CODE\_2016) %>%   
 mutate(SA3\_CODE\_2016 = SA3\_CODE\_2016 %>% as.character()) %>%   
 left\_join(NSW, by = c("SA3\_CODE\_2016" = "SA3\_CODE16")) %>%   
 mutate(SA3\_CODE16 = SA3\_CODE\_2016) %>%   
 select(names(NSW),Price) %>%   
 filter(GCC\_NAME16 %in% c("Greater Sydney","No usual address (NSW)")) %>%   
 select(SA3\_CODE16,Price)  
  
# Filtered map with price  
NSW\_Property <- NSW %>%   
 filter(SA3\_CODE16 %in% Property\_map$SA3\_CODE16) %>%   
 inner\_join(Property\_map, by = c("SA3\_CODE16")) %>%   
 select(names(NSW),Price)   
  
# Heat map of Sydney  
ggplot() +  
 geom\_sf(data = NSW) +  
 geom\_sf(data = NSW\_Property, aes(fill = Price/1000000)) +  
 scale\_fill\_gradient(low = "#56B1F7",high = "#132B43", na = "Property Price ($m)") +  
 ggtitle("Sydney Property Price Heatmap") +  
 xlab("Latitude") + ylab("Longtitude")



## Conclusion

Summarising the graphs and statistics produced for Sydney property price vs commonly listed attributes: 1. Sydney property prices are right skewed distributed with a mean of $1,645,043. 2. Over the investigation period, there does not seem to be a significant increase or decrease in property prices sold based on a p-vale of 0.496 due to the short period of investigation and data collected. 3. Property prices vs number of bedrooms has a positive relationship. Most properties sold within the investigation period have 2 to 5 bedrooms, with the mode at 3 bedrooms with 103 records. 4. There is a negative relationship between property price and distance from the Sydney CBD, which means properties are more expensive when you’re located closer to the CBD. This relationship is statistically significant with a p value of 0.0001523 when testing the coefficient for distance to CBD. In our linear model, a 1km increase in distance from CBD results in a $16917 decrease in property price. 5. Our heat map loosely validated point 4, where for our investigation period, higher property prices are concentrated more towards the CBD.