Final Project

Peter Antonaros

Packages/Libraries & Setup

```
#Set cache for seed
knitr::opts_chunk$set(cache = T)
#Memory allocation for Java ~10gb and Garbage Collection
options(java.parameters = c("-XX:+UseConcMarkSweepGC", "-Xmx10000m"))
#Packages to load
pacman::p_load(
  ggplot2,
  tidyverse,
  data.table,
  R.utils,
  magrittr,
  dplyr,
  testthat,
  YARF,
  lubridate,
  missForest,
  parallel,
  doParallel,
  caret,
  glmnet
)

#Set CPU cores for YARF
num_of_cores = 8
set_YARF_num_cores(num_of_cores)
```

```
## YARF can now make use of 8 cores.
```

```
#Initialize rJava
library(rJava)
gc()
```

```
##           used  (Mb) gc trigger  (Mb) max used  (Mb)
## Ncells 2560643 136.8    4437645 237.0  4437645 237.0
## Vcells 4280398  32.7   10146329  77.5  7143700  54.6
```

```
.jinit()
```

```
## [1] 0
```

The Data

```r
#Set our file path & read in file
housingDataFilePath = "/home/peterjr/RepoCollections/MATH_342W_FinalProject/Datasets/housing_data_2016_2
#Keep a unaltered "True" copy
housingDataTrue = data.table(fread(housingDataFilePath))

housingData = housingDataTrue

housingData
```

```
##                                 HITId                      HITTypeId
##    1: 3OID399FXG7F26JWONXF0Y86J90FD4 36BILMLQB75QQNBTYKGYCZWDN8TVAU
##    2: 3MQY1YVHS3K2MF90MWR2LPQH7KJ2B0 36BILMLQB75QQNBTYKGYCZWDN8TVAU
##    3: 3DGDV62G7O94Q9AA5193G9V6OOY2PL 36BILMLQB75QQNBTYKGYCZWDN8TVAU
##    4: 3087LXLJ6MGL3MI2CB9KLRONPKRF0B 36BILMLQB75QQNBTYKGYCZWDN8TVAU
##    5: 3FULMHZ7OUX88KSKHZ0ZSKY93XJ4MN 36BILMLQB75QQNBTYKGYCZWDN8TVAU
##   ---
## 2226:                           <NA>                           <NA>
## 2227:                           <NA>                           <NA>
## 2228:                           <NA>                           <NA>
## 2229:                           <NA>                           <NA>
## 2230:                           <NA>                           <NA>
##                                                                  Title
##    1: Find Information about Housing To Help a Student Project -- Very easy
##    2: Find Information about Housing To Help a Student Project -- Very easy
##    3: Find Information about Housing To Help a Student Project -- Very easy
##    4: Find Information about Housing To Help a Student Project -- Very easy
##    5: Find Information about Housing To Help a Student Project -- Very easy
##   ---
## 2226:                                                                  <NA>
## 2227:                                                                  <NA>
## 2228:                                                                  <NA>
## 2229:                                                                  <NA>
## 2230:                                                                  <NA>
##                                       Description Keywords Reward
##    1: Go to a link and copy information into the HIT       NA  $0.05
##    2: Go to a link and copy information into the HIT       NA  $0.05
##    3: Go to a link and copy information into the HIT       NA  $0.05
##    4: Go to a link and copy information into the HIT       NA  $0.05
##    5: Go to a link and copy information into the HIT       NA  $0.05
##   ---
## 2226:                                            <NA>       NA   <NA>
## 2227:                                            <NA>       NA   <NA>
## 2228:                                            <NA>       NA   <NA>
## 2229:                                            <NA>       NA   <NA>
## 2230:                                            <NA>       NA   <NA>
##                     CreationTime MaxAssignments
##    1: Wed Feb 15 22:13:37 PST 2017              1
##    2: Wed Feb 15 22:13:37 PST 2017              1
##    3: Wed Feb 15 22:13:41 PST 2017              1
##    4: Wed Feb 15 22:13:33 PST 2017              1
##    5: Wed Feb 15 22:13:38 PST 2017              1
##   ---
```

```
## 2226:                         <NA>             NA
## 2227:                         <NA>             NA
## 2228:                         <NA>             NA
## 2229:                         <NA>             NA
## 2230:                         <NA>             NA
##                                    RequesterAnnotation
##    1: BatchId:2689947;OriginalHitTemplateId:920937336;
##    2: BatchId:2689947;OriginalHitTemplateId:920937336;
##    3: BatchId:2689947;OriginalHitTemplateId:920937336;
##    4: BatchId:2689947;OriginalHitTemplateId:920937336;
##    5: BatchId:2689947;OriginalHitTemplateId:920937336;
##   ---
## 2226:                                             <NA>
## 2227:                                             <NA>
## 2228:                                             <NA>
## 2229:                                             <NA>
## 2230:                                             <NA>
##       AssignmentDurationInSeconds AutoApprovalDelayInSeconds
##    1:                         900                         60
##    2:                         900                         60
##    3:                         900                         60
##    4:                         900                         60
##    5:                         900                         60
##   ---
## 2226:                          NA                         NA
## 2227:                          NA                         NA
## 2228:                          NA                         NA
## 2229:                          NA                         NA
## 2230:                          NA                         NA
##                        Expiration NumberOfSimilarHITs LifetimeInSeconds
##    1: Wed Feb 22 22:13:37 PST 2017                  NA                NA
##    2: Wed Feb 22 22:13:37 PST 2017                  NA                NA
##    3: Wed Feb 22 22:13:41 PST 2017                  NA                NA
##    4: Wed Feb 22 22:13:33 PST 2017                  NA                NA
##    5: Wed Feb 22 22:13:38 PST 2017                  NA                NA
##   ---
## 2226:                         <NA>                  NA                NA
## 2227:                         <NA>                  NA                NA
## 2228:                         <NA>                  NA                NA
## 2229:                         <NA>                  NA                NA
## 2230:                         <NA>                  NA                NA
##                        AssignmentId       WorkerId AssignmentStatus
##    1: 32KTQ2V7RDFCSAWQOW1SXC5AZIC9MB A231MNJJDDF3LS         Approved
##    2: 35LDD5557A4W96FHSTSHNLJQAB7MKZ A394B5QVCVKU7A         Approved
##    3: 3FFJ6VRIL1O8OXIM3LK7C8X0F5U0I6 A231MNJJDDF3LS         Approved
##    4: 3S4AW7T80BIRPM8T7P4MGRF5DL74L7  AHXBZXWIZJSVG         Approved
##    5: 3JMSRU9HQIUCDTHGAZI5CMPYH7REVS A231MNJJDDF3LS         Approved
##   ---
## 2226:                           <NA>           <NA>             <NA>
## 2227:                           <NA>           <NA>             <NA>
## 2228:                           <NA>           <NA>             <NA>
## 2229:                           <NA>           <NA>             <NA>
## 2230:                           <NA>           <NA>             <NA>
##                        AcceptTime                   SubmitTime
```

```
##    1: Thu Feb 16 05:32:36 PST 2017 Thu Feb 16 05:35:37 PST 2017
##    2: Wed Feb 15 22:19:51 PST 2017 Wed Feb 15 22:21:52 PST 2017
##    3: Thu Feb 16 03:17:01 PST 2017 Thu Feb 16 03:19:01 PST 2017
##    4: Thu Feb 16 04:54:24 PST 2017 Thu Feb 16 04:57:04 PST 2017
##    5: Wed Feb 15 23:54:29 PST 2017 Wed Feb 15 23:56:45 PST 2017
##   ---
## 2226:                         <NA>                         <NA>
## 2227:                         <NA>                         <NA>
## 2228:                         <NA>                         <NA>
## 2229:                         <NA>                         <NA>
## 2230:                         <NA>                         <NA>
##                   AutoApprovalTime            ApprovalTime RejectionTime
##    1: Thu Feb 16 05:36:37 PST 2017 2017-02-16 13:37:11 UTC            NA
##    2: Wed Feb 15 22:22:52 PST 2017 2017-02-16 06:23:11 UTC            NA
##    3: Thu Feb 16 03:20:01 PST 2017 2017-02-16 11:20:11 UTC            NA
##    4: Thu Feb 16 04:58:04 PST 2017 2017-02-16 12:58:11 UTC            NA
##    5: Wed Feb 15 23:57:45 PST 2017 2017-02-16 07:58:11 UTC            NA
##   ---
## 2226:                         <NA>                    <NA>            NA
## 2227:                         <NA>                    <NA>            NA
## 2228:                         <NA>                    <NA>            NA
## 2229:                         <NA>                    <NA>            NA
## 2230:                         <NA>                    <NA>            NA
##       RequesterFeedback WorkTimeInSeconds LifetimeApprovalRate
##    1:                NA               181      100% (187/187)
##    2:                NA               121          100% (8/8)
##    3:                NA               120      100% (187/187)
##    4:                NA               160      100% (115/115)
##    5:                NA               136      100% (187/187)
##   ---
## 2226:                NA                NA                <NA>
## 2227:                NA                NA                <NA>
## 2228:                NA                NA                <NA>
## 2229:                NA                NA                <NA>
## 2230:                NA                NA                <NA>
##       Last30DaysApprovalRate Last7DaysApprovalRate
##    1:         100% (187/187)        100% (187/187)
##    2:             100% (8/8)            100% (8/8)
##    3:         100% (187/187)        100% (187/187)
##    4:         100% (115/115)        100% (103/103)
##    5:         100% (187/187)        100% (187/187)
##   ---
## 2226:                   <NA>                  <NA>
## 2227:                   <NA>                  <NA>
## 2228:                   <NA>                  <NA>
## 2229:                   <NA>                  <NA>
## 2230:                   <NA>                  <NA>
##                                                                                     U
##    1: http://www.mlsli.com/homes-for-sale/address-not-available-from-broker-Flushing-NY-11355-1492383
##    2:               http://www.mlsli.com/homes-for-sale/30-11-Parsons-Blvd-Flushing-NY-11354-1552428
##    3:                   http://www.mlsli.com/homes-for-sale/102-14-Lewis-Ave-Corona-NY-11368-1570845
##    4:             http://www.mlsli.com/homes-for-sale/144-48-Roosevelt-Ave-Flushing-NY-11354-1553227
##    5:                 http://www.mlsli.com/homes-for-sale/245-27-76th-Ave-Bellerose-NY-11426-1612806
##   ---
```

```
## 2226:                                                                                      <
## 2227:                                                                                      <
## 2228:                                                                                      <
## 2229:                                                                                      <
## 2230:                                                                                      <
##       approx_year_built cats_allowed common_charges community_district_num
##    1:              1955           no           $767                     25
##    2:              1955           no           <NA>                     25
##    3:              2004           no           $167                     24
##    4:              2002           no           $275                     25
##    5:              1949          yes           <NA>                     26
##   ---
## 2226:              1987           no           $480                     25
## 2227:              1983          yes           $956                     25
## 2228:              2010           no           $250                     24
## 2229:              2010           no           $250                     24
## 2230:              1982           no           $792                     25
##       coop_condo date_of_sale dining_room_type dogs_allowed fuel_type
##    1:      co-op    2/16/2016            combo           no       gas
##    2:      co-op    2/16/2016           formal           no       oil
##    3:      condo    2/17/2016            combo           no      <NA>
##    4:      condo    2/17/2016            combo           no       gas
##    5:      co-op    2/18/2016            combo          yes       gas
##   ---
## 2226:      condo         <NA>            combo           no       gas
## 2227:      condo         <NA>           formal           no       gas
## 2228:      condo         <NA>           formal           no       gas
## 2229:      condo         <NA>           formal           no       gas
## 2230:      condo         <NA>           formal           no       gas
##                                  full_address_or_zip_code garage_exists
##    1:                                  Flushing NY, 11355          <NA>
##    2: 30-11 Parsons Blvd,  Flushing NY, 11354 ( Sold )      Share          <NA>
##    3:                       102-14 Lewis Ave,  Corona NY, 11368          <NA>
##    4:                 144-48 Roosevelt Ave,  Flushing NY, 11354          <NA>
##    5:                     245-27 76th Ave,  Bellerose NY, 11426          <NA>
##   ---
## 2226:                         Not AvailableFlushing NY, 11355          <NA>
## 2227:                      One Bay Club Dr,  Bayside NY, 11360           yes
## 2228:                                  Ridgewood NY, 11385          <NA>
## 2229:                                  Ridgewood NY, 11385          <NA>
## 2230:                    Two Bay Club Drive,  Bayside NY, 11360           yes
##       kitchen_type maintenance_cost         model_type num_bedrooms
##    1:       eat in             <NA>   Mitchell Garden 3            2
##    2:       eat in             $604          Jr-4 Model            1
##    3:   efficiency             <NA>         Apt In Bldg            1
##    4:       eat in             <NA>    144-48 Roosevelt            3
##    5:       eat in             $660                C-1            2
##   ---
## 2226:        combo             <NA> Colden Luxury Condo            2
## 2227:        eatin             <NA>        2 Br Deluxe            2
## 2228:        combo             <NA>             Modern            3
## 2229:        combo             <NA>              Condo            3
## 2230:        combo             <NA>          2 Bedroom            2
##       num_floors_in_building num_full_bathrooms num_half_bathrooms
```

```
##    1:                       6                  1                  NA
##    2:                       7                  1                  NA
##    3:                       1                  1                  NA
##    4:                      NA                  2                  NA
##    5:                       2                  1                  NA
##   ---
## 2226:                       7                  1                  NA
## 2227:                      NA                  2                  NA
## 2228:                      NA                  2                  NA
## 2229:                       4                  2                  NA
## 2230:                      NA                  2                  NA
##       num_total_rooms parking_charges pct_tax_deductibl sale_price sq_footage
##    1:               5            <NA>                NA   $228,000         NA
##    2:               4            <NA>                NA   $235,500        890
##    3:               3            <NA>                NA   $137,550        550
##    4:               5            <NA>                NA   $545,000         NA
##    5:               4            <NA>                39   $241,700        675
##   ---
## 2226:               4            <NA>                NA       <NA>         NA
## 2227:               5             $99                NA       <NA>         NA
## 2228:               6            <NA>                NA       <NA>       1500
## 2229:               6            <NA>                NA       <NA>       1600
## 2230:               5            <NA>                NA       <NA>       1134
##       total_taxes walk_score listing_price_to_nearest_1000
##    1:        <NA>         82                          <NA>
##    2:        <NA>         89                          <NA>
##    3:      $5,500         90                          <NA>
##    4:      $2,260         94                          <NA>
##    5:        <NA>         71                          <NA>
##   ---
## 2226:      $3,588         97                          $628
## 2227:      $5,100         82                          $988
## 2228:        $250         96                          $850
## 2229:        $250         96                          $850
## 2230:      $3,785         82                          $899
##                                                                                                         u
##    1:                                                                                                  <
##    2:                                                                                                  <
##    3:                                                                                                  <
##    4:                                                                                                  <
##    5:                                                                                                  <
##   ---
## 2226: http://www.mlsli.com/homes-for-sale/address-not-available-from-broker-Flushing-NY-11355-1694274
## 2227:                    http://www.mlsli.com/homes-for-sale/One-Bay-Club-Dr-Bayside-NY-11360-1962742
## 2228: http://www.mlsli.com/homes-for-sale/address-not-available-from-broker-Ridgewood-NY-11385-9216955
## 2229: http://www.mlsli.com/homes-for-sale/address-not-available-from-broker-Ridgewood-NY-11385-9210133
## 2230:                 http://www.mlsli.com/homes-for-sale/Two-Bay-Club-Drive-Bayside-NY-11360-1402975
```

```
#Relevant columns begin at the column labeled (URL)
```

Initial Data Preparation I (Dropping Irrelevant Columns & Storing Possible Ones for Later Use)

```r
#Dropping Mturk columns that are not relevant to our housing model
housingData[,c(1:27):=NULL]

#Save the urls in case they are needed
housingURLS = housingData[,.(URL)]

#Dropping URL from the data table
housingData[,URL:=NULL]
#Dropping other useless url column from data table (ALL NA's)
housingData[,url:=NULL]
#Dropping model_type because similar information is contained in other columns
housingData[,model_type:=NULL]

housingData
```

```
##       approx_year_built cats_allowed common_charges community_district_num
##    1:              1955           no           $767                     25
##    2:              1955           no           <NA>                     25
##    3:              2004           no           $167                     24
##    4:              2002           no           $275                     25
##    5:              1949          yes           <NA>                     26
##   ---
## 2226:              1987           no           $480                     25
## 2227:              1983          yes           $956                     25
## 2228:              2010           no           $250                     24
## 2229:              2010           no           $250                     24
## 2230:              1982           no           $792                     25
##       coop_condo date_of_sale dining_room_type dogs_allowed fuel_type
##    1:      co-op    2/16/2016            combo           no       gas
##    2:      co-op    2/16/2016           formal           no       oil
##    3:      condo    2/17/2016            combo           no      <NA>
##    4:      condo    2/17/2016            combo           no       gas
##    5:      co-op    2/18/2016            combo          yes       gas
##   ---
## 2226:      condo         <NA>            combo           no       gas
## 2227:      condo         <NA>           formal           no       gas
## 2228:      condo         <NA>           formal           no       gas
## 2229:      condo         <NA>           formal           no       gas
## 2230:      condo         <NA>           formal           no       gas
##                                   full_address_or_zip_code garage_exists
##    1:                                     Flushing NY, 11355          <NA>
##    2: 30-11 Parsons Blvd,  Flushing NY, 11354 ( Sold )      Share          <NA>
##    3:                        102-14 Lewis Ave,  Corona NY, 11368          <NA>
##    4:                  144-48 Roosevelt Ave,  Flushing NY, 11354          <NA>
##    5:                      245-27 76th Ave,  Bellerose NY, 11426          <NA>
##   ---
## 2226:                          Not AvailableFlushing NY, 11355          <NA>
## 2227:                        One Bay Club Dr,  Bayside NY, 11360           yes
## 2228:                                     Ridgewood NY, 11385          <NA>
## 2229:                                     Ridgewood NY, 11385          <NA>
## 2230:                     Two Bay Club Drive,  Bayside NY, 11360           yes
##       kitchen_type maintenance_cost num_bedrooms num_floors_in_building
##    1:       eat in             <NA>            2                      6
```

```
##    2:       eat in           $604          1                   7
##    3:   efficiency           <NA>          1                   1
##    4:       eat in           <NA>          3                  NA
##    5:       eat in           $660          2                   2
##   ---
## 2226:        combo           <NA>          2                   7
## 2227:        eatin           <NA>          2                  NA
## 2228:        combo           <NA>          3                  NA
## 2229:        combo           <NA>          3                   4
## 2230:        combo           <NA>          2                  NA
##       num_full_bathrooms num_half_bathrooms num_total_rooms parking_charges
##    1:                  1                 NA               5            <NA>
##    2:                  1                 NA               4            <NA>
##    3:                  1                 NA               3            <NA>
##    4:                  2                 NA               5            <NA>
##    5:                  1                 NA               4            <NA>
##   ---
## 2226:                  1                 NA               4            <NA>
## 2227:                  2                 NA               5             $99
## 2228:                  2                 NA               6            <NA>
## 2229:                  2                 NA               6            <NA>
## 2230:                  2                 NA               5            <NA>
##       pct_tax_deductibl sale_price sq_footage total_taxes walk_score
##    1:                NA  $228,000          NA        <NA>         82
##    2:                NA  $235,500         890        <NA>         89
##    3:                NA  $137,550         550     $5,500          90
##    4:                NA  $545,000          NA     $2,260          94
##    5:                39  $241,700         675        <NA>         71
##   ---
## 2226:                NA       <NA>         NA     $3,588          97
## 2227:                NA       <NA>         NA     $5,100          82
## 2228:                NA       <NA>       1500        $250         96
## 2229:                NA       <NA>       1600        $250         96
## 2230:                NA       <NA>       1134     $3,785          82
##       listing_price_to_nearest_1000
##    1:                          <NA>
##    2:                          <NA>
##    3:                          <NA>
##    4:                          <NA>
##    5:                          <NA>
##   ---
## 2226:                          $628
## 2227:                          $988
## 2228:                          $850
## 2229:                          $850
## 2230:                          $899
```

Initial Data Preparation II (Writing some notes about Columns)

```
#Getting the column names to write some notes about each column
names(housingData)
```

```
##  [1] "approx_year_built"             "cats_allowed"
```

```
##  [3] "common_charges"                "community_district_num"
##  [5] "coop_condo"                    "date_of_sale"
##  [7] "dining_room_type"              "dogs_allowed"
##  [9] "fuel_type"                     "full_address_or_zip_code"
## [11] "garage_exists"                 "kitchen_type"
## [13] "maintenance_cost"              "num_bedrooms"
## [15] "num_floors_in_building"        "num_full_bathrooms"
## [17] "num_half_bathrooms"            "num_total_rooms"
## [19] "parking_charges"               "pct_tax_deductibl"
## [21] "sale_price"                    "sq_footage"
## [23] "total_taxes"                   "walk_score"
## [25] "listing_price_to_nearest_1000"
```

```
#Getting some general information about the table
summary(housingData)
```

```
##  approx_year_built cats_allowed       common_charges     community_district_num
##  Min.   :1893      Length:2230        Length:2230        Min.   : 3.00
##  1st Qu.:1950      Class :character   Class :character   1st Qu.:25.00
##  Median :1958      Mode  :character   Mode  :character   Median :26.00
##  Mean   :1963                                            Mean   :26.33
##  3rd Qu.:1970                                            3rd Qu.:28.00
##  Max.   :2017                                            Max.   :32.00
##  NA's   :40                                              NA's   :19
##   coop_condo         date_of_sale       dining_room_type   dogs_allowed
##  Length:2230        Length:2230        Length:2230        Length:2230
##  Class :character   Class :character   Class :character   Class :character
##  Mode  :character   Mode  :character   Mode  :character   Mode  :character
##
##
##
##
##   fuel_type          full_address_or_zip_code garage_exists
##  Length:2230        Length:2230              Length:2230
##  Class :character   Class :character         Class :character
##  Mode  :character   Mode  :character         Mode  :character
##
##
##
##
##  kitchen_type       maintenance_cost    num_bedrooms   num_floors_in_building
##  Length:2230        Length:2230        Min.   :0.000   Min.   : 1.000
##  Class :character   Class :character   1st Qu.:1.000   1st Qu.: 3.000
##  Mode  :character   Mode  :character   Median :2.000   Median : 6.000
##                                        Mean   :1.653   Mean   : 7.785
##                                        3rd Qu.:2.000   3rd Qu.: 7.000
##                                        Max.   :6.000   Max.   :34.000
##                                        NA's   :115     NA's   :650
##  num_full_bathrooms num_half_bathrooms num_total_rooms  parking_charges
##  Min.   :1.000      Min.   :0.0000     Min.   : 0.000   Length:2230
##  1st Qu.:1.000      1st Qu.:1.0000     1st Qu.: 3.000   Class :character
##  Median :1.000      Median :1.0000     Median : 4.000   Mode  :character
##  Mean   :1.231      Mean   :0.9535     Mean   : 4.139
##  3rd Qu.:1.000      3rd Qu.:1.0000     3rd Qu.: 5.000
```

```
##  Max.   :3.000      Max.   :2.0000     Max.   :14.000
##                     NA's   :2058       NA's   :2
##  pct_tax_deductibl  sale_price          sq_footage     total_taxes
##  Min.   :20.0      Length:2230        Min.   : 100.0   Length:2230
##  1st Qu.:40.0      Class :character   1st Qu.: 743.0   Class :character
##  Median :50.0      Mode  :character   Median : 881.0   Mode  :character
##  Mean   :45.4                         Mean   : 955.4
##  3rd Qu.:50.0                         3rd Qu.:1100.0
##  Max.   :75.0                         Max.   :6215.0
##  NA's   :1754                         NA's   :1210
##    walk_score    listing_price_to_nearest_1000
##  Min.   : 7.00   Length:2230
##  1st Qu.:77.00   Class :character
##  Median :89.00   Mode  :character
##  Mean   :83.92
##  3rd Qu.:95.00
##  Max.   :99.00
##
```

Column Name | Information | Notes to Self about column

"approx_year_built" | Integer representing the year the house was built | 40 NA's

"cats_allowed" | Binary decision (0,1) are cats allowed in the home or not | Check for NA's & Factor

"common_charges" | Some sort of charges in dollars ($) | Remove the dollar symbol & Convert to integer & Check for NA's

"community_district_num" | Integer representing the district number of community home is a part of | 19 NA's

"coop_condo" | String representing "Co-op" or "Condo" | Lowercase everything | Check for levels & Factor

"date_of_sale" | String representing the date the home was sold |

"dining_room_type" | String representing "formal" or "combo" dining room type | Lowercase everything & Check for NA's & Factor

"dogs_allowed" | Binary decision (0,1) are dogs allowed in the home or not | Factor this & Check for NA's

"fuel_type" | String representing "gas", "oil", or "other" energy source for the home | Lowercase everything & Check for NA's & factor

"full_address_or_zip_code" | String representing the address of the home |

"garage_exists" | String representing "Yes" if the home has a garage | Check for NA's & Factor this & Missingness column

"kitchen_type" | String representing "Eat-In", "Efficiency", or "Combo" kitchen type | Lowercase everything & Factor this & Check for NA's

"maintenance_cost" | Cost of maintenece for the home in dollars ($) | Remove the dollar symbol & Convert to integer & Check for NA's

"num_bedrooms" | Integer representing number of bedrooms present in the home | 115 NA's

"num_floors_in_building" | Integer representing number of floors present in building containing home | 650 NA's

“num_full_bathrooms” | Integer representing the number of full bathrooms present in the home | No NA’s

“num_half_bathrooms” | Integer representing the number of half bathrooms present in the home | 2058 NA’s

“num_total_rooms” | Integer representing the number of total rooms present in the home | 2 NA’s

“parking_charges” | Parking charges in dollars ($) | Remove the dollar symbol & Convert to integer & Check for NA’s

“pct_tax_deductibl” | Integer representing percent of tax deduction | 1754 NA’s

“sale_price” | Sale price of the home in dollars ($) | Remove the dollar symbol & Convert to integer & Check for NA’s

“sq_footage” | Integer representing the total square footage of the home | 1210 NA’s

“total_taxes” | Taxes on the home in dollars ($) | Remove the dollar symbol & Convert to integer & Check for NA’s

“walk_score” | Integer representing a walking score for the home |

“listing_price_to_nearest_1000” | Listing price to the nearest 1000 for the home in dollars ($) | Remove the dollar symbol & Convert to integer & Check for NA’s

Data Cleaning I (Symbol Removal & Establishing Column Types)

```
#First lets deal with the String columns that have $ symbols and convert to integer

#Extract dollar sign columns as subset to operate on
dollarSymbolSubset = housingData[,.(common_charges,maintenance_cost,parking_charges,sale_price,total_ta

#Remove dollar signs based on pattern matching
dollarSymbolSubset[] = lapply(dollarSymbolSubset,gsub,pattern="$",fixed=TRUE,replacement="")

#Also Remove any commas that may appear for large values
dollarSymbolSubset[] = lapply(dollarSymbolSubset,gsub,pattern=",",fixed=TRUE,replacement="")

#Replace the columns in housing Data with the new dollarSymbolSubset
housingData[,c("common_charges","maintenance_cost","parking_charges","sale_price","total_taxes","listin
            dollarSymbolSubset[,c("common_charges","maintenance_cost","parking_charges","sale_price","t

#Now we need to convert these columns in housing data to integer type
housingData[,c("common_charges","maintenance_cost","parking_charges","sale_price","total_taxes","listin


######################################################################
#Second lets deal with changing cats_allowed and dogs_allowed to factors

housingData[,sum(is.na(cats_allowed))] # No NA values for cats_allowed
```

```
## [1] 0
```

```
housingData[,sum(is.na(dogs_allowed))] # No NA values for dogs_allowed
```

```
## [1] 0
```

```r
#Changing to factors for cats and dogs allowed
unique(housingData[,cats_allowed]) # 3 "unique" values
```

```
## [1] "no"  "yes" "y"
```

```r
#Lets deal with the y instead of a yes
housingData$cats_allowed[grepl("y", housingData$cats_allowed)] = "yes"
length(unique(housingData[,cats_allowed])) # 2 unique values
```

```
## [1] 2
```

```r
#Lets do the same for dogs
unique(housingData[,dogs_allowed]) # 3 "unique" values"
```

```
## [1] "no"    "yes"   "yes89"
```

```r
housingData$dogs_allowed[grepl("yes89", housingData$dogs_allowed)] = "yes"
length(unique(housingData[,cats_allowed])) # 2 unique values
```

```
## [1] 2
```

```r
#Factor them
housingData[,c("cats_allowed","dogs_allowed")] = lapply(housingData[,c("cats_allowed","dogs_allowed")],

levels(housingData$cats_allowed) #Check levels
```

```
## [1] "no"  "yes"
```

```r
levels(housingData$dogs_allowed) #Check levels
```

```
## [1] "no"  "yes"
```

```r
############################################################################
#Third lets deal with other String columns to be factored (track NA's for later)

housingData[,sum(is.na(coop_condo))] # No NA values for coop_condo
```

```
## [1] 0
```

```r
length(unique(housingData[,coop_condo])) # 2 unique values
```

```
## [1] 2
```

```r
#Factor it
housingData[,coop_condo := factor(coop_condo)]
levels(housingData$coop_condo)
```

```
## [1] "co-op" "condo"
```

```r
housingData[,sum(is.na(dining_room_type))] # 448 NA values for dining_room_type
```

```
## [1] 448
```

```r
length(unique(housingData[,dining_room_type])) # 6 unique values including NA
```

```
## [1] 6
```

```r
length(which(housingData$dining_room_type == "none")) #none occurs 2 times
```

```
## [1] 2
```

```r
length(which(housingData$dining_room_type == "dining area")) #dining area occurs 2 times
```

```
## [1] 2
```

```r
#Lets deal with the issue of "dining area" as the room type and consider it as type other
housingData$dining_room_type[grepl("dining area", housingData$dining_room_type)] = "other"
housingData$dining_room_type[grepl("none", housingData$dining_room_type)] = "other"
length(unique(housingData[,dining_room_type])) # 4 unique values including NA
```

```
## [1] 4
```

```r
housingData[,dining_room_type := factor(dining_room_type)]
levels(housingData$dining_room_type)
```

```
## [1] "combo"  "formal" "other"
```

```r
housingData[,sum(is.na(fuel_type))] # 112 NA values for dining_room_type
```

```
## [1] 112
```

```r
length(unique(housingData[,fuel_type])) # 7 "unique" values including NA
```

```
## [1] 7
```

```r
#Lets deal with the capitalization issues for fuel_typenone
housingData[,fuel_type := tolower(fuel_type)]
housingData$fuel_type[grepl("none", housingData$fuel_type)] = "other"
length(unique(housingData[,fuel_type])) # r unique values including NA
```

```
## [1] 5
```

```r
housingData[,fuel_type := factor(fuel_type)]
levels(housingData$fuel_type)
```

```
## [1] "electric" "gas"      "oil"      "other"
```

```r
housingData[,sum(is.na(kitchen_type))]# 16 NA values for dining_room_type
```

```
## [1] 16
```

```r
length(unique(housingData[,kitchen_type])) # 14 "unique" values including NA
```

```
## [1] 14
```

```r
#Lets deal with the upper case lower case kitchen type differences
housingData[,kitchen_type:=tolower(kitchen_type)] # Lowercase everything to pattern match
length(unique(housingData[,kitchen_type])) # 11 "unique" values including NA
```

```
## [1] 11
```

```r
#Lets now deal with spaces creating more unique values
housingData[,kitchen_type := lapply(kitchen_type,gsub,pattern="eat in",fixed=TRUE,replacement="eatin")]
length(unique(housingData[,kitchen_type])) # 10 "unique" values including NA
```

```
## [1] 10
```

```r
#Lets lets deal with the misspellings of efficiency kitchen
housingData$kitchen_type[grepl("effic", housingData$kitchen_type)] = "efficiency"
length(unique(housingData[,kitchen_type])) # 6 unique values including NA
```

```
## [1] 6
```

```r
#Finally lets deal with 1955 and replace that with NA -> I am assuming here 1955 is wrong and not a typ
housingData[, kitchen_type := sapply(kitchen_type, function(x) replace(x, which(x=="1955"), NA))]
length(unique(housingData[,kitchen_type])) # t unique values including NA (no 1955 -> NA)
```

```
## [1] 5
```

```r
housingData[,kitchen_type := factor(kitchen_type)]
levels(housingData$kitchen_type)
```

```
## [1] "combo"      "eatin"      "efficiency" "none"
```

```r
###########################################################################
#Fourth lets deal with the Garage column (track NA's for later)

housingData[,sum(is.na(garage_exists))] # 1826 NA values for garage exists
```

```
## [1] 1826
```

```r
length(unique(housingData[,garage_exists])) # 7 "unique" values
```

```
## [1] 7
```

```r
#Lets deal with the capitalization and misspelling of yes
housingData[,garage_exists := tolower(garage_exists)]
housingData$garage_exists[grepl("y", housingData$garage_exists)] = "yes"
length(unique(housingData[,garage_exists])) # 5 unique values including NA
```

```
## [1] 5
```

```r
#Lets treat underground and ug as yes
housingData$garage_exists[grepl("u", housingData$garage_exists)] = "yes"
length(unique(housingData[,garage_exists])) # 3 unique values including NA
```

```
## [1] 3
```

```r
#Lets treat 1 as a yes
housingData$garage_exists[grepl("1", housingData$garage_exists)] = "yes"
length(unique(housingData[,garage_exists])) # 2 unique values including NA
```

```
## [1] 2
```

```r
#Fill NA's in garage with No's -> Use 1s in missingness to indicate this later om.
housingData[, c("garage_exists")][is.na(housingData[, c("garage_exists")])] = "no"

housingData[,c("garage_exists")] = lapply(housingData[,c("garage_exists")], as.factor)
#setattr(housingData$garage_exists,"levels",c("no","yes"))
#housingData[,garage_exists := factor(garage_exists)]
levels(housingData$garage_exists)
```

```
## [1] "no"  "yes"
```

```r
###########################################################################
#Fifth lets take the date column treat is a an unordered factor

#In order to limit the total number of levels in Date, lets just grabs the months
#We sacrifice some granularity, but hopefully this generalize better

housingData$date_of_sale = format(as.Date(housingData$date_of_sale, format="%m/%d/%Y"),"%m")
housingData[,date_of_sale:= factor(date_of_sale,ordered=FALSE)]
length(unique(housingData[,date_of_sale])) #13 including NA which is what we want
```

```
## [1] 13
```

```r
#Lets take a look at our data set now

ncol(housingData)
```

```
## [1] 25
```

```
summary(housingData)
```

```
##  approx_year_built cats_allowed common_charges   community_district_num
##  Min.   :1893      no :1402     Min.   :  70.0   Min.   : 3.00
##  1st Qu.:1950      yes: 828     1st Qu.: 280.0   1st Qu.:25.00
##  Median :1958                   Median : 390.0   Median :26.00
##  Mean   :1963                   Mean   : 441.8   Mean   :26.33
##  3rd Qu.:1970                   3rd Qu.: 551.5   3rd Qu.:28.00
##  Max.   :2017                   Max.   :2499.0   Max.   :32.00
##  NA's   :40                     NA's   :1684     NA's   :19
##  coop_condo    date_of_sale  dining_room_type dogs_allowed    fuel_type
##  co-op:1661   12     :  58   combo :957       no :1684     electric:  62
##  condo: 569   06     :  53   formal:620       yes: 546     gas     :1348
##               01     :  50   other :205                    oil     : 664
##               11     :  47   NA's  :448                    other   :  44
##               05     :  46                                 NA's    : 112
##               (Other): 274
##               NA's   :1702
##  full_address_or_zip_code garage_exists     kitchen_type maintenance_cost
##  Length:2230              no :1826      combo     :399   Min.   : 155.0
##  Class :character         yes: 404      eatin     :942   1st Qu.: 630.5
##  Mode  :character                       efficiency:849   Median : 767.0
##                                         none      : 23   Mean   : 858.9
##                                         NA's      : 17   3rd Qu.: 985.5
##                                                          Max.   :4659.0
##                                                          NA's   :623
##   num_bedrooms   num_floors_in_building num_full_bathrooms num_half_bathrooms
##  Min.   :0.000   Min.   : 1.000         Min.   :1.000      Min.   :0.0000
##  1st Qu.:1.000   1st Qu.: 3.000         1st Qu.:1.000      1st Qu.:1.0000
##  Median :2.000   Median : 6.000         Median :1.000      Median :1.0000
##  Mean   :1.653   Mean   : 7.785         Mean   :1.231      Mean   :0.9535
##  3rd Qu.:2.000   3rd Qu.: 7.000         3rd Qu.:1.000      3rd Qu.:1.0000
##  Max.   :6.000   Max.   :34.000         Max.   :3.000      Max.   :2.0000
##  NA's   :115     NA's   :650                               NA's   :2058
##  num_total_rooms  parking_charges pct_tax_deductibl   sale_price
##  Min.   : 0.000   Min.   :  6.0   Min.   :20.0      Min.   : 55000
##  1st Qu.: 3.000   1st Qu.: 60.0   1st Qu.:40.0      1st Qu.:171500
##  Median : 4.000   Median : 99.0   Median :50.0      Median :259500
##  Mean   : 4.139   Mean   :107.6   Mean   :45.4      Mean   :314957
##  3rd Qu.: 5.000   3rd Qu.:149.0   3rd Qu.:50.0      3rd Qu.:428875
##  Max.   :14.000   Max.   :837.0   Max.   :75.0      Max.   :999999
##  NA's   :2        NA's   :1671    NA's   :1754      NA's   :1702
##    sq_footage      total_taxes     walk_score    listing_price_to_nearest_1000
##  Min.   : 100.0   Min.   :  11   Min.   : 7.00   Min.   :  65.0
##  1st Qu.: 743.0   1st Qu.: 281   1st Qu.:77.00   1st Qu.: 229.8
##  Median : 881.0   Median :2411   Median :89.00   Median : 329.5
##  Mean   : 955.4   Mean   :2226   Mean   :83.92   Mean   : 385.6
##  3rd Qu.:1100.0   3rd Qu.:3500   3rd Qu.:95.00   3rd Qu.: 525.0
##  Max.   :6215.0   Max.   :9300   Max.   :99.00   Max.   :1000.0
##  NA's   :1210     NA's   :1646                   NA's   :534
```

Data Manipulation I (Creating New Features)

```r
#First lets just add up all the charges into a single column
#Assign new column totalCharges to be the row sum of the chargeCols ignoring NA's
housingData[, totalCharges := rowSums(.SD,na.rm=TRUE), .SDcols = c("common_charges","maintenance_cost",
```

```
##       approx_year_built cats_allowed common_charges community_district_num
##    1:              1955           no            767                     25
##    2:              1955           no             NA                     25
##    3:              2004           no            167                     24
##    4:              2002           no            275                     25
##    5:              1949          yes             NA                     26
##   ---
## 2226:              1987           no            480                     25
## 2227:              1983          yes            956                     25
## 2228:              2010           no            250                     24
## 2229:              2010           no            250                     24
## 2230:              1982           no            792                     25
##       coop_condo date_of_sale dining_room_type dogs_allowed fuel_type
##    1:      co-op           02            combo           no       gas
##    2:      co-op           02           formal           no       oil
##    3:      condo           02            combo           no      <NA>
##    4:      condo           02            combo           no       gas
##    5:      co-op           02            combo          yes       gas
##   ---
## 2226:      condo         <NA>            combo           no       gas
## 2227:      condo         <NA>           formal           no       gas
## 2228:      condo         <NA>           formal           no       gas
## 2229:      condo         <NA>           formal           no       gas
## 2230:      condo         <NA>           formal           no       gas
##                                        full_address_or_zip_code garage_exists
##    1:                                        Flushing NY, 11355            no
##    2: 30-11 Parsons Blvd,  Flushing NY, 11354 ( Sold )      Share            no
##    3:                          102-14 Lewis Ave,  Corona NY, 11368            no
##    4:                    144-48 Roosevelt Ave,  Flushing NY, 11354            no
##    5:                        245-27 76th Ave,  Bellerose NY, 11426            no
##   ---
## 2226:                            Not AvailableFlushing NY, 11355            no
## 2227:                          One Bay Club Dr,  Bayside NY, 11360           yes
## 2228:                                           Ridgewood NY, 11385            no
## 2229:                                           Ridgewood NY, 11385            no
## 2230:                       Two Bay Club Drive,  Bayside NY, 11360           yes
##       kitchen_type maintenance_cost num_bedrooms num_floors_in_building
##    1:        eatin               NA            2                      6
##    2:        eatin              604            1                      7
##    3:   efficiency               NA            1                      1
##    4:        eatin               NA            3                     NA
##    5:        eatin              660            2                      2
##   ---
## 2226:        combo               NA            2                      7
## 2227:        eatin               NA            2                     NA
## 2228:        combo               NA            3                     NA
## 2229:        combo               NA            3                      4
## 2230:        combo               NA            2                     NA
##       num_full_bathrooms num_half_bathrooms num_total_rooms parking_charges
```

```
##    1:                 1                NA               5               NA
##    2:                 1                NA               4               NA
##    3:                 1                NA               3               NA
##    4:                 2                NA               5               NA
##    5:                 1                NA               4               NA
##   ---
## 2226:                 1                NA               4               NA
## 2227:                 2                NA               5               99
## 2228:                 2                NA               6               NA
## 2229:                 2                NA               6               NA
## 2230:                 2                NA               5               NA
##       pct_tax_deductibl sale_price sq_footage total_taxes walk_score
##    1:                NA     228000         NA          NA         82
##    2:                NA     235500        890          NA         89
##    3:                NA     137550        550        5500         90
##    4:                NA     545000         NA        2260         94
##    5:                39     241700        675          NA         71
##   ---
## 2226:                NA         NA         NA        3588         97
## 2227:                NA         NA         NA        5100         82
## 2228:                NA         NA       1500         250         96
## 2229:                NA         NA       1600         250         96
## 2230:                NA         NA       1134        3785         82
##       listing_price_to_nearest_1000 totalCharges
##    1:                            NA          767
##    2:                            NA          604
##    3:                            NA         5667
##    4:                            NA         2535
##    5:                            NA          660
##   ---
## 2226:                           628         4068
## 2227:                           988         6155
## 2228:                           850          500
## 2229:                           850          500
## 2230:                           899         4577
```

```
housingData[,sum(is.na(totalCharges))] # No NA's here which is good since
```

```
## [1] 0
```

```
##############################################################################
#Second lets extract the zip codes and assign them to their own column

#Lets use a regular expression to extract the zip code out of this field
housingData[,zip_code := substr(str_extract(full_address_or_zip_code,"[0-9]{5}"),1,5)]
housingData[,zip_code := as.numeric(zip_code)]
#We can now drop the full_address column since we wont need that
housingData[,full_address_or_zip_code := NULL]


###############################################################################
#Third lets add up full and half bathrooms
#Lets divide the half bathroom columns by 2 so that when we add them it is more granular
```

```r
housingData[,num_half_bathrooms:=num_half_bathrooms/2]
#Assign a new column to represent the total number of bathrooms
housingData[,totalBathrooms :=rowSums(.SD,na.rm=TRUE), .SDcols = c("num_full_bathrooms","num_half_bathro
```

```
##       approx_year_built cats_allowed common_charges community_district_num
##    1:              1955           no            767                     25
##    2:              1955           no             NA                     25
##    3:              2004           no            167                     24
##    4:              2002           no            275                     25
##    5:              1949          yes             NA                     26
##   ---
## 2226:              1987           no            480                     25
## 2227:              1983          yes            956                     25
## 2228:              2010           no            250                     24
## 2229:              2010           no            250                     24
## 2230:              1982           no            792                     25
##       coop_condo date_of_sale dining_room_type dogs_allowed fuel_type
##    1:      co-op           02            combo           no       gas
##    2:      co-op           02           formal           no       oil
##    3:      condo           02            combo           no      <NA>
##    4:      condo           02            combo           no       gas
##    5:      co-op           02            combo          yes       gas
##   ---
## 2226:      condo         <NA>            combo           no       gas
## 2227:      condo         <NA>           formal           no       gas
## 2228:      condo         <NA>           formal           no       gas
## 2229:      condo         <NA>           formal           no       gas
## 2230:      condo         <NA>           formal           no       gas
##       garage_exists kitchen_type maintenance_cost num_bedrooms
##    1:            no        eatin               NA            2
##    2:            no        eatin              604            1
##    3:            no   efficiency               NA            1
##    4:            no        eatin               NA            3
##    5:            no        eatin              660            2
##   ---
## 2226:            no        combo               NA            2
## 2227:           yes        eatin               NA            2
## 2228:            no        combo               NA            3
## 2229:            no        combo               NA            3
## 2230:           yes        combo               NA            2
##       num_floors_in_building num_full_bathrooms num_half_bathrooms
##    1:                      6                  1                 NA
##    2:                      7                  1                 NA
##    3:                      1                  1                 NA
##    4:                     NA                  2                 NA
##    5:                      2                  1                 NA
##   ---
## 2226:                      7                  1                 NA
## 2227:                     NA                  2                 NA
## 2228:                     NA                  2                 NA
## 2229:                      4                  2                 NA
## 2230:                     NA                  2                 NA
##       num_total_rooms parking_charges pct_tax_deductibl sale_price sq_footage
```

```
##    1:              5             NA              NA    228000          NA
##    2:              4             NA              NA    235500         890
##    3:              3             NA              NA    137550         550
##    4:              5             NA              NA    545000          NA
##    5:              4             NA              39    241700         675
##   ---
## 2226:              4             NA              NA        NA          NA
## 2227:              5             99              NA        NA          NA
## 2228:              6             NA              NA        NA        1500
## 2229:              6             NA              NA        NA        1600
## 2230:              5             NA              NA        NA        1134
##       total_taxes walk_score listing_price_to_nearest_1000 totalCharges
##    1:          NA         82                            NA          767
##    2:          NA         89                            NA          604
##    3:        5500         90                            NA         5667
##    4:        2260         94                            NA         2535
##    5:          NA         71                            NA          660
##   ---
## 2226:        3588         97                           628         4068
## 2227:        5100         82                           988         6155
## 2228:         250         96                           850          500
## 2229:         250         96                           850          500
## 2230:        3785         82                           899         4577
##       zip_code totalBathrooms
##    1:    11355              1
##    2:    11354              1
##    3:    11368              1
##    4:    11354              2
##    5:    11426              1
##   ---
## 2226:    11355              1
## 2227:    11360              2
## 2228:    11385              2
## 2229:    11385              2
## 2230:    11360              2
```

```
##########################################################################
#Fourth lets bring in some extra data that shows median income by zipcode
queensIncomeDataFilePath = "/home/peterjr/RepoCollections/MATH_342W_FinalProject/Datasets/income_queens
queensIncomeData = data.table(read.csv(queensIncomeDataFilePath))

#Grab columns we want and remove the first row description of columns
queensIncomeData = queensIncomeData[-1,.(GEO_ID,S1901_C01_012E)]

#Change Data Type
queensIncomeData[,zip_code := as.numeric(GEO_ID)]


#Rename median income column
setnames(queensIncomeData, "S1901_C01_012E", "median_income")

queensIncomeData[,median_income := as.numeric(median_income)]
```

```
## Warning in eval(jsub, SDenv, parent.frame()): NAs introduced by coercion
```

```r
#Drop the geo_id column
queensIncomeData[,GEO_ID := NULL]


###########################################################################
#Fifth lets join this to our housing data on the zipcode
#We are doing a left join because I want everything in housing preserved -> median income can be impute

housingData = left_join(housingData,queensIncomeData,by.x = "zip_code",by.y = "zip_code")
```

```
## Joining, by = "zip_code"
```

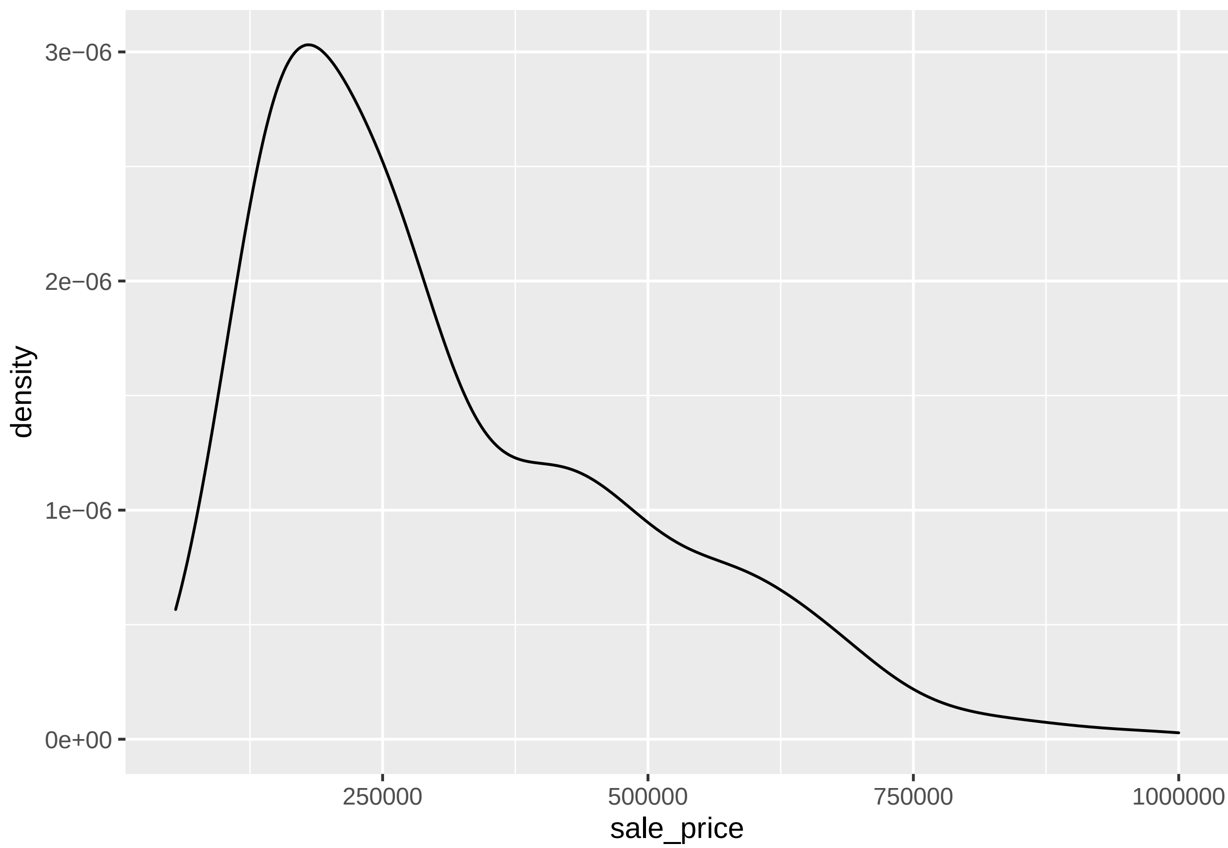
```r
housingData[,sum(is.na(median_income))] # 64 NA values, not bad since most are getting filled, should b
```

```
## [1] 64
```

Initial Data Exploration I (Basic Visualization & Stats)

```r
###########################################################################
#First lets take a look at sale_price. It is important we understand this since it is our response
sale_density = ggplot(housingData)+
  geom_density(aes(x=sale_price)) # From here we can see a concentration around ~ 225k

sale_density
```

```
## Warning: Removed 1702 rows containing non-finite values (stat_density).
```




```r
##########################################################################
#Second lets take a look at some basic statistics about sale_price
sd(housingData$sale_price,na.rm = TRUE)
```

```
## [1] 179526.6
```

```r
median(housingData$sale_price,na.rm = TRUE)
```

```
## [1] 259500
```

```r
mean(housingData$sale_price,na.rm = TRUE) # Mean higher than median makes sense with tail in graph abov
```

```
## [1] 314956.6
```

```r
min(housingData$sale_price,na.rm = TRUE)
```

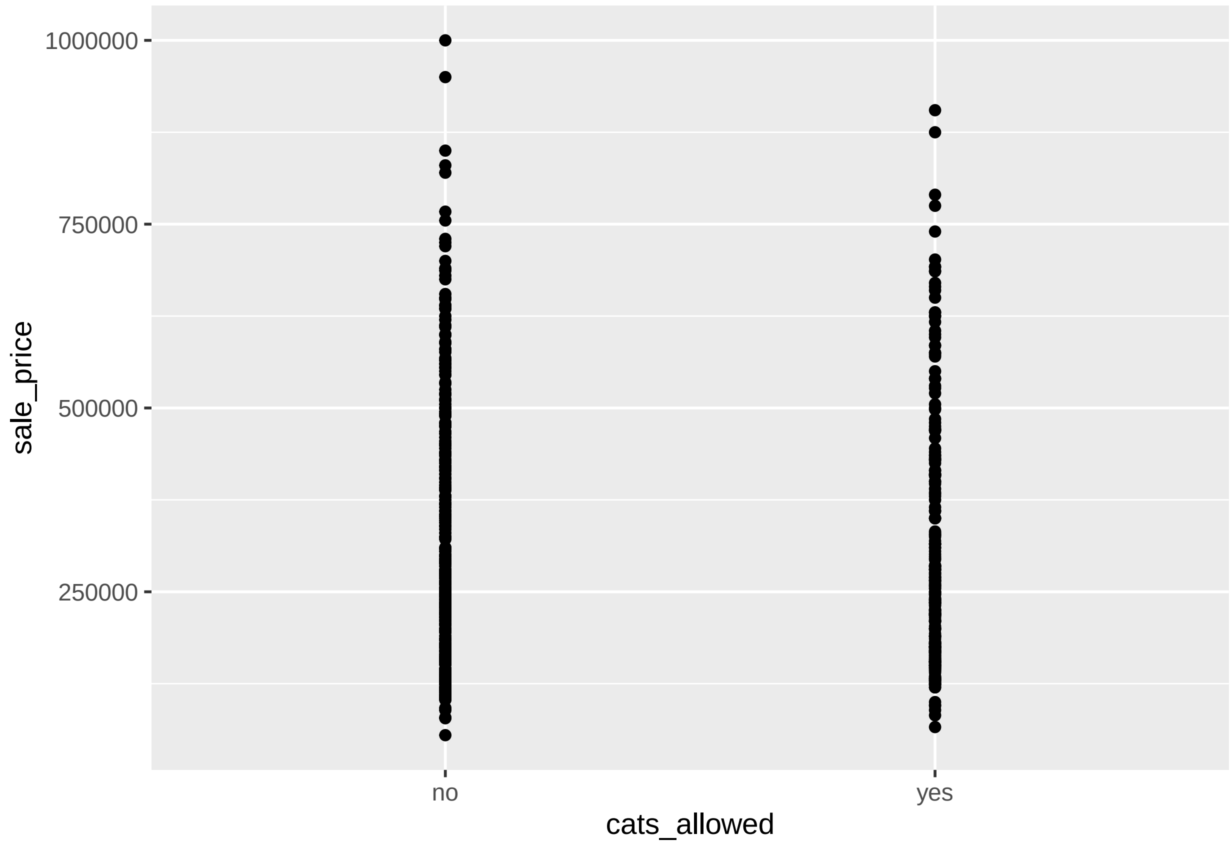
```
## [1] 55000
```

```r
max(housingData$sale_price,na.rm = TRUE)
```

```
## [1] 999999
```

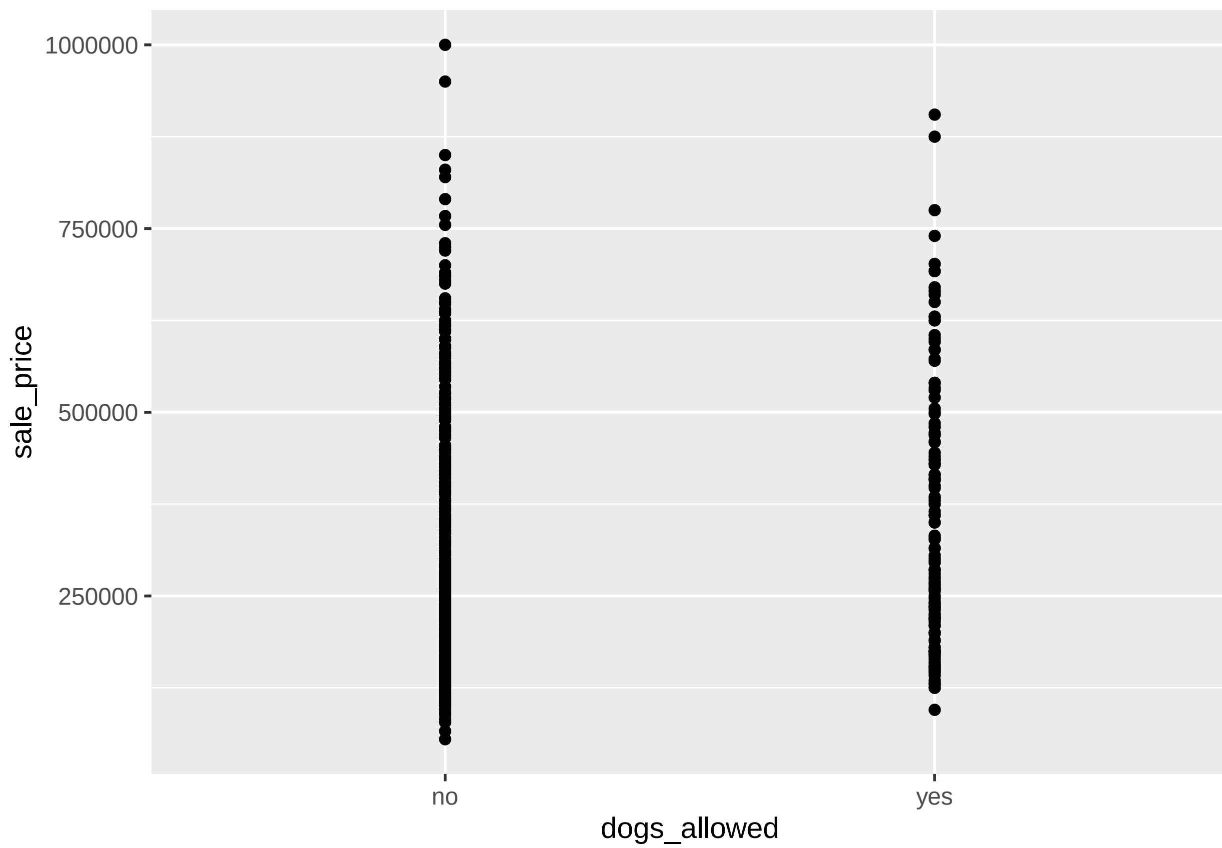
```r
##########################################################################
#Third lets look at some of the columns against sale_price
#I am looking at columns that I need will have the biggest influence on sale_price

bedrooms_sale = ggplot(housingData)+
  geom_point(aes(x=num_bedrooms, y=sale_price))# Looking at num_bedrooms VS sale_price


cats_sale = ggplot(housingData)+
  geom_point(aes(x=cats_allowed, y=sale_price)) # Looking at cats_allowed VS sale_price


dogs_sale = ggplot(housingData)+
  geom_point(aes(x=dogs_allowed, y=sale_price)) # Looking at dogs_allowed VS sale_price

#This is a feature we created from num_full_bathrooms + (num_half_bathrooms)/2
bathroooms_sale = ggplot(housingData)+
  geom_point(aes(x=totalBathrooms, y=sale_price)) # Looking at totalBathrooms VS sale_price

#This is a feature we created by adding up all of the chargest columns
charges_sale = ggplot(housingData)+
  geom_point(aes(x=totalCharges, y=sale_price)) # Looking at totalCharges VS sale_price

walk_sale = ggplot(housingData)+
  geom_point(aes(x=walk_score, y=sale_price)) # Looking at walk_score VS sale_price

cats_sale
```

```
## Warning: Removed 1702 rows containing missing values (geom_point).
```



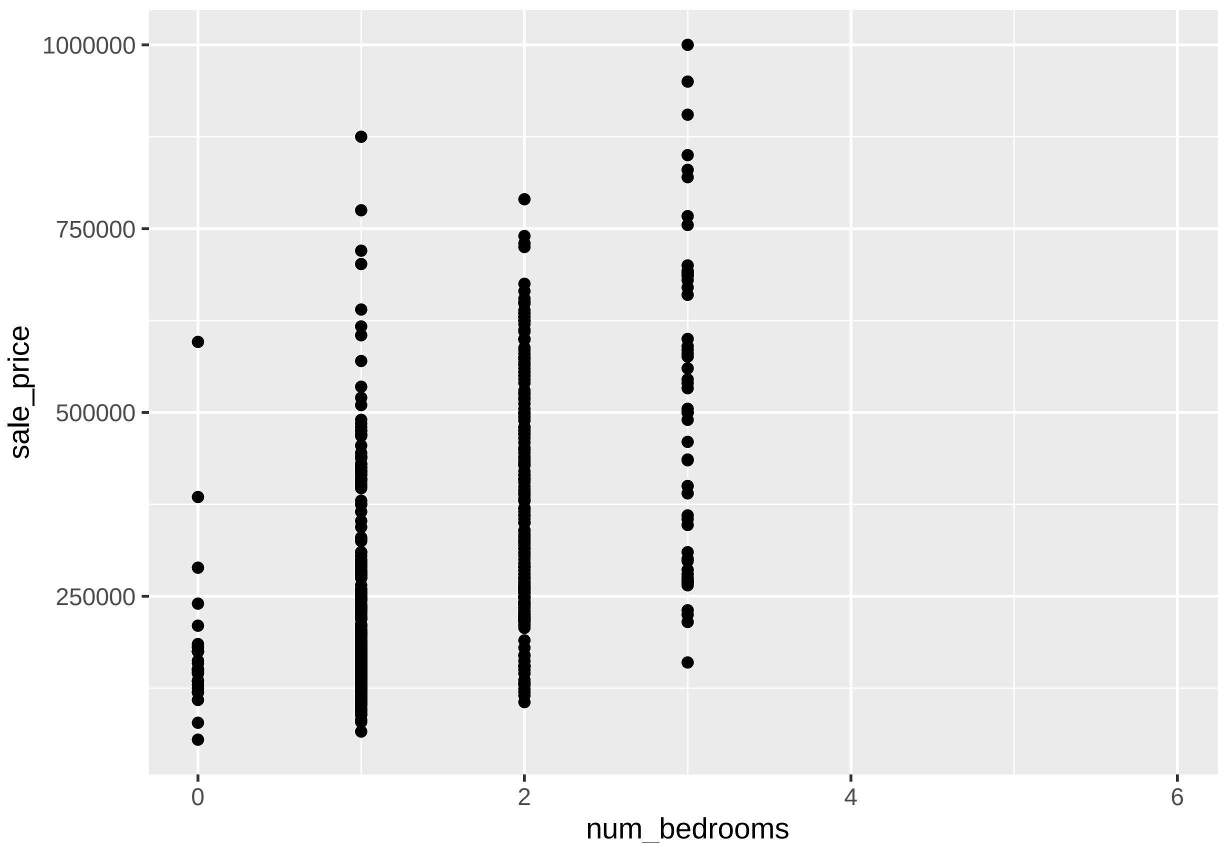

```
dogs_sale
```

```
## Warning: Removed 1702 rows containing missing values (geom_point).
```



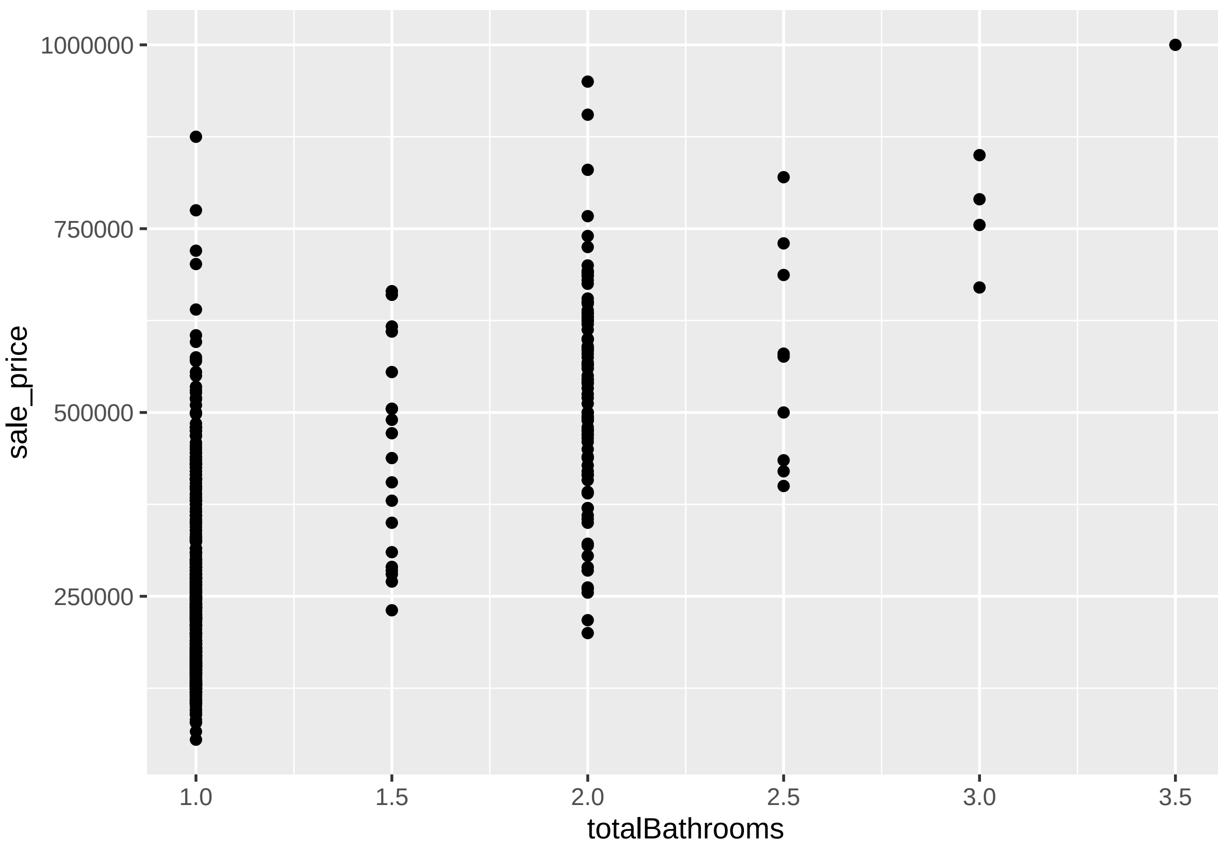

```
bedrooms_sale
```

```
## Warning: Removed 1702 rows containing missing values (geom_point).
```



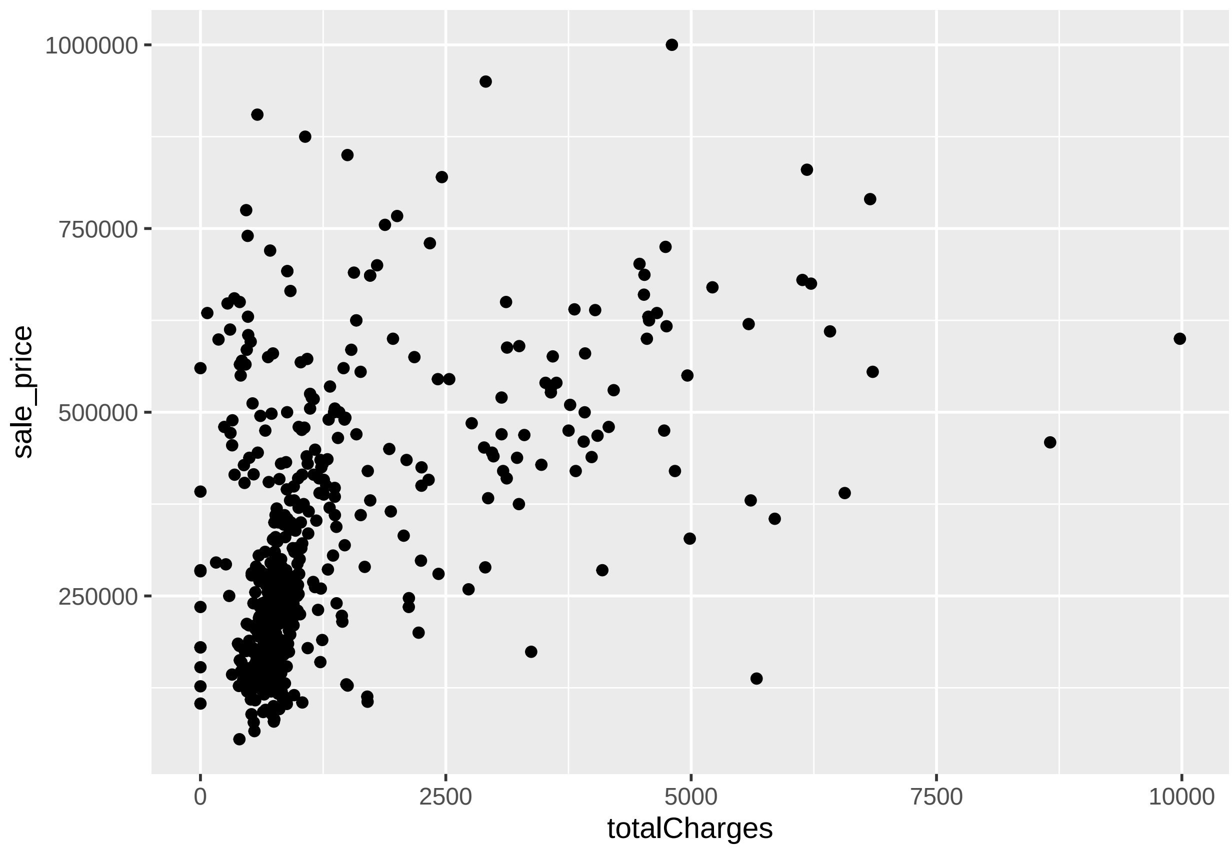

```
bathroooms_sale
```

```
## Warning: Removed 1702 rows containing missing values (geom_point).
```



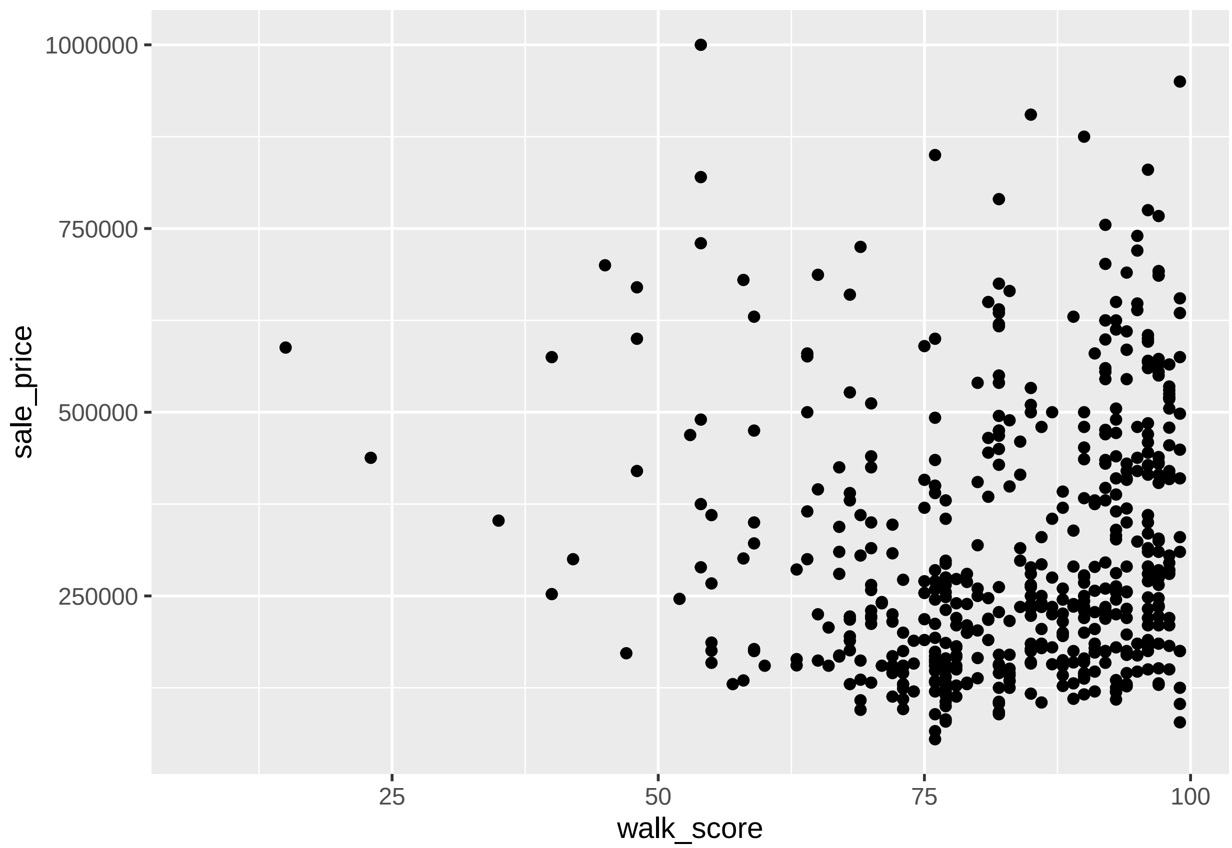

```
charges_sale
```

```
## Warning: Removed 1702 rows containing missing values (geom_point).
```



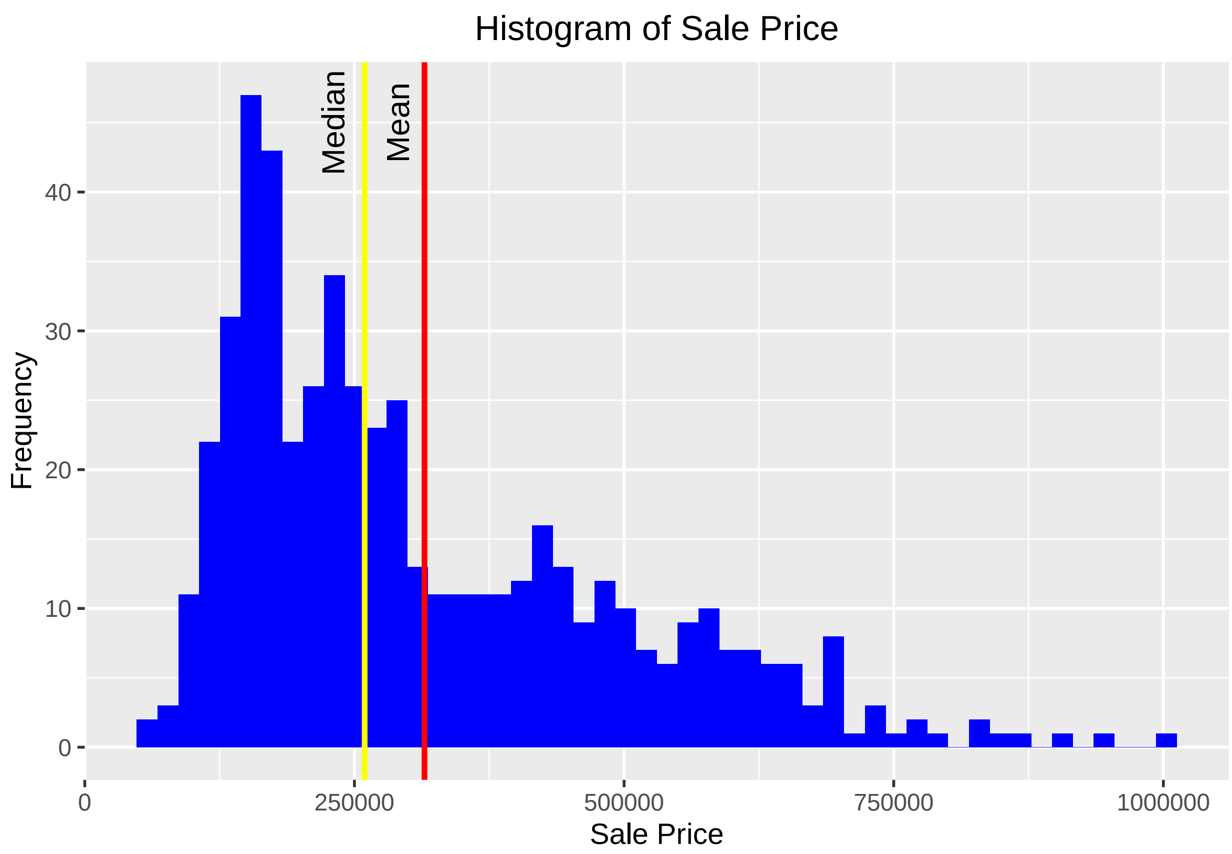

```
walk_sale
```

```
## Warning: Removed 1702 rows containing missing values (geom_point).
```



Initial Data Exploration II (Better visualizations)

```
ggplot(data=subset(housingData, !is.na(sale_price))) +
  aes(x = sale_price) +
  geom_histogram(bins = 50L, fill = "blue")+
  geom_vline(data = subset(housingData, !is.na(sale_price)), aes(xintercept = mean(sale_price)), color =
  annotate("text", x=290000, y=45, label=paste("Mean"),size=4.1,angle=90)+
  geom_vline(data = subset(housingData, !is.na(sale_price)), aes(xintercept = median(sale_price)), colo
  annotate("text", x=230000, y=45, label=paste("Median"),size=4.1,angle=90)+
  labs(x = "Sale Price", y = "Frequency")+
  ggtitle("Histogram of Sale Price")+
  theme(plot.title = element_text(hjust = 0.5))
```




```
#Uncomment the following line if we want to save this picture to our notebook directory
#gsave("SalePriceHist.png",width=6, height=4,dpi=400)
```

Establishing a Missingness Table

```
##########################################################################
#First lets grab the columns that are of interest to us
housingData = housingData[,.(approx_year_built,cats_allowed,community_district_num,coop_condo,date_of_sa
                             dogs_allowed,fuel_type,garage_exists,kitchen_type,num_bedrooms,num_floors_
                             sale_price,sq_footage,walk_score,totalCharges,zip_code,median_income)]

#########################################################################
#Second lets build up our missing table 0/1 where 1 indicates a NA value in the housingData

#Create a missing data table and fill with zeros
colNames = names(housingData)
missRows = nrow(housingData)
missCols = ncol(housingData)
missingData = setNames(data.table(matrix(0,nrow = missRows, ncol = missCols)), colNames)
setnames(missingData,1:ncol(missingData), paste0(names(missingData)[1:ncol(missingData)], '_miss'))
#Data Set with 1s indicating missing in housingData
missingData[is.na(housingData)] = 1

#Due to the nature of the construction of the missing table, all columns in housingData have a correspo
#This may not be entirely accurate, since some of our columns in housingData have no NA's thus the *_mi
```

```r
#Remove missing columns where the sum is 0. Implies housingData did not have any NAs.
checkZero= function(x){
    if(sum(x)==0){
      TRUE
    }
}

length(missingData[,sapply(missingData,  checkZero)]) # 7 columns with no missingness, we will drop the
```

```
## [1] 7
```

```r
missingData = missingData[, colSums(missingData != 0) > 0, with = FALSE]

#Lets also drop missingness for sale_price. This will be made clear later, but since we plan on trainin
#our missing will be all 1's aka a zero variance feature.
missingData = missingData[,!c("sale_price_miss")]

#Lets mark the indices where sale price is missing for reasons that will be made clear later
salePriceMissingIndices = which(is.na(housingData$sale_price))
salePriceFilledIndices = which(!is.na(housingData$sale_price))
```

Imputation Using The MissForest Algorithm

```r
#########################################################################
#Lets impute our data set including sale price
imputeSet = housingData

#Setting up parallelization cluster
cluster = makePSOCKcluster(num_of_cores)
registerDoParallel(cluster)

#Initialize the missForest algorithm with 100 trees and parallelization
Ximp = missForest(imputeSet,verbose = TRUE, maxiter = 5, ntree = 100, parallelize = "variables")
```

```
##   parallelizing over the variables of the input data matrix 'xmis'
##   missForest iteration 1 in progress...done!
##     estimated error(s): 0.3817819 0.1703643
##     difference(s): 0.1066605 0.09372197
##     time: 4.328 seconds
##
##   missForest iteration 2 in progress...done!
##     estimated error(s): 0.3830449 0.162378
##     difference(s): 0.004361728 0.06008969
##     time: 4.205 seconds
##
##   missForest iteration 3 in progress...done!
##     estimated error(s): 0.3696682 0.1637841
##     difference(s): 0.001972756 0.04988789
##     time: 4.446 seconds
##
##   missForest iteration 4 in progress...done!
##     estimated error(s): 0.3765333 0.1659532
```

```
##     difference(s): 0.001750518 0.0478139
##     time: 3.999 seconds
##
##   missForest iteration 5 in progress...done!
##     estimated error(s): 0.3655974 0.1635159
##     difference(s): 0.00163494 0.04557175
##     time: 4.36 seconds
```

```
#Stop the cluster
stopCluster(cluster)
registerDoSEQ()

#Get our final imputed Dataset and bind it to the missiningness table
finalHousingData = cbind(Ximp$ximp,missingData)

finalHousingData
```

```
##       approx_year_built cats_allowed community_district_num coop_condo
##    1:              1955           no                     25      co-op
##    2:              1955           no                     25      co-op
##    3:              2004           no                     24      condo
##    4:              2002           no                     25      condo
##    5:              1949          yes                     26      co-op
##   ---
## 2226:              1987           no                     25      condo
## 2227:              1983          yes                     25      condo
## 2228:              2010           no                     24      condo
## 2229:              2010           no                     24      condo
## 2230:              1982           no                     25      condo
##       date_of_sale dining_room_type dogs_allowed fuel_type garage_exists
##    1:           02            combo           no       gas            no
##    2:           02           formal           no       oil            no
##    3:           02            combo           no       gas            no
##    4:           02            combo           no       gas            no
##    5:           02            combo          yes       gas            no
##   ---
## 2226:           10            combo           no       gas            no
## 2227:           02           formal           no       gas           yes
## 2228:           06           formal           no       gas            no
## 2229:           06           formal           no       gas            no
## 2230:           02           formal           no       gas           yes
##       kitchen_type num_bedrooms num_floors_in_building totalBathrooms
##    1:        eatin            2                6.00000              1
##    2:        eatin            1                7.00000              1
##    3:   efficiency            1                1.00000              1
##    4:        eatin            3                5.15500              2
##    5:        eatin            2                2.00000              1
##   ---
## 2226:        combo            2                7.00000              1
## 2227:        eatin            2               15.75667              2
## 2228:        combo            3                4.13500              2
## 2229:        combo            3                4.00000              2
## 2230:        combo            2               14.25578              2
##       num_total_rooms sale_price sq_footage walk_score totalCharges zip_code
```

```
##    1:               5   228000.0  1012.0988         82          767    11355
##    2:               4   235500.0   890.0000         89          604    11354
##    3:               3   137550.0   550.0000         90         5667    11368
##    4:               5   545000.0  1018.4111         94         2535    11354
##    5:               4   241700.0   675.0000         71          660    11426
##   ---
## 2226:               4   471478.4   968.8313         97         4068    11355
## 2227:               5   610995.5  1225.8587         82         6155    11360
## 2228:               6   575402.7  1500.0000         96          500    11385
## 2229:               6   578105.3  1600.0000         96          500    11385
## 2230:               5   585690.5  1134.0000         82         4577    11360
##       median_income approx_year_built_miss community_district_num_miss
##    1:         38451                      0                           0
##    2:         43660                      0                           0
##    3:         45980                      0                           0
##    4:         43660                      0                           0
##    5:         77487                      0                           0
##   ---
## 2226:         38451                      0                           0
## 2227:         82982                      0                           0
## 2228:         60526                      0                           0
## 2229:         60526                      0                           0
## 2230:         82982                      0                           0
##       date_of_sale_miss dining_room_type_miss fuel_type_miss kitchen_type_miss
##    1:                 0                     0              0                 0
##    2:                 0                     0              0                 0
##    3:                 0                     0              1                 0
##    4:                 0                     0              0                 0
##    5:                 0                     0              0                 0
##   ---
## 2226:                 1                     0              0                 0
## 2227:                 1                     0              0                 0
## 2228:                 1                     0              0                 0
## 2229:                 1                     0              0                 0
## 2230:                 1                     0              0                 0
##       num_bedrooms_miss num_floors_in_building_miss num_total_rooms_miss
##    1:                 0                           0                    0
##    2:                 0                           0                    0
##    3:                 0                           0                    0
##    4:                 0                           1                    0
##    5:                 0                           0                    0
##   ---
## 2226:                 0                           0                    0
## 2227:                 0                           1                    0
## 2228:                 0                           1                    0
## 2229:                 0                           0                    0
## 2230:                 0                           1                    0
##       sq_footage_miss zip_code_miss median_income_miss
##    1:               1             0                  0
##    2:               0             0                  0
##    3:               0             0                  0
##    4:               1             0                  0
##    5:               0             0                  0
##   ---
```

```
## 2226:                1             0                  0
## 2227:                1             0                  0
## 2228:                0             0                  0
## 2229:                0             0                  0
## 2230:                0             0                  0
```

```
Ximp$OOBerror
```

```
##     NRMSE       PFC
## 0.3655974 0.1635159
```

Establishing Holdout Set I

```
#Prior to any feature selection/modeling we want to establish a hold out set from our finalHousing Data
#We do this so that we can truly consider our hold out test set to be independent from any of the proce

holdout_K=5
holdout_prop = 1 / holdout_K

#Where sale price was NA prior to imputing  ~ 75% of ALL data
salePriceNA_Data = finalHousingData[salePriceMissingIndices,]

#This is crucial to note since our errors will be more honest albeit larger.
#If we test on imputed data we are essentially computing prediction error on a prediction rather than r
#Most likely this will result in worse error, but it will generalize better in the real world.

#Where sale price was not NA ~ 25% of ALL data
salePriceFilled_Data = finalHousingData[salePriceFilledIndices,]

#Training & Testing data (All Features)
finalHousingData_Train = salePriceNA_Data
finalHousingData_Test = salePriceFilled_Data

X_all_holdout = finalHousingData_Test[,!c("sale_price")]
y_all_holdout = finalHousingData_Test$sale_price

finalHousingData_Train
```

```
##       approx_year_built cats_allowed community_district_num coop_condo
##    1:              1983           no                     25      condo
##    2:              1930          yes                     28      co-op
##    3:              1912           no                     28      co-op
##    4:              1953          yes                     25      co-op
##    5:              1941           no                     28      condo
##   ---
## 1698:              1987           no                     25      condo
## 1699:              1983          yes                     25      condo
## 1700:              2010           no                     24      condo
## 1701:              2010           no                     24      condo
## 1702:              1982           no                     25      condo
##       date_of_sale dining_room_type dogs_allowed fuel_type garage_exists
##    1:           08            combo           no       gas           yes
```

```
##    2:           12           other          yes       oil            no
##    3:           12           combo           no  electric            no
##    4:           01           combo           no       gas            no
##    5:           06          formal           no       gas            no
##   ---
## 1698:           10           combo           no       gas            no
## 1699:           02          formal           no       gas           yes
## 1700:           06          formal           no       gas            no
## 1701:           06          formal           no       gas            no
## 1702:           02          formal           no       gas           yes
##       kitchen_type num_bedrooms num_floors_in_building totalBathrooms
##    1:        eatin         2.00              14.518722              2
##    2:        combo         1.00               3.000000              1
##    3:   efficiency         0.85               5.000000              1
##    4:        combo         3.00               8.972143              1
##    5:   efficiency         4.00               6.000000              1
##   ---
## 1698:        combo         2.00               7.000000              1
## 1699:        eatin         2.00              15.756667              2
## 1700:        combo         3.00               4.135000              2
## 1701:        combo         3.00               4.000000              2
## 1702:        combo         2.00              14.255778              2
##       num_total_rooms sale_price sq_footage walk_score totalCharges zip_code
##    1:               6   620625.0  1250.0000         82         4955    11360
##    2:               2   216285.0   450.0000         99          862    11375
##    3:               2   207496.7   566.9533         99          738    11375
##    4:               5   393730.0  1152.6725         49         1495    11357
##    5:               7   488639.0  1524.0000         94         5776    11375
##   ---
## 1698:               4   471478.4   968.8313         97         4068    11355
## 1699:               5   610995.5  1225.8587         82         6155    11360
## 1700:               6   575402.7  1500.0000         96          500    11385
## 1701:               6   578105.3  1600.0000         96          500    11385
## 1702:               5   585690.5  1134.0000         82         4577    11360
##       median_income approx_year_built_miss community_district_num_miss
##    1:         82982                      0                           0
##    2:         72982                      0                           0
##    3:         72982                      0                           0
##    4:         74255                      0                           0
##    5:         72982                      0                           0
##   ---
## 1698:         38451                      0                           0
## 1699:         82982                      0                           0
## 1700:         60526                      0                           0
## 1701:         60526                      0                           0
## 1702:         82982                      0                           0
##       date_of_sale_miss dining_room_type_miss fuel_type_miss kitchen_type_miss
##    1:                 1                     0              0                 0
##    2:                 1                     1              0                 0
##    3:                 1                     1              0                 0
##    4:                 1                     0              0                 0
##    5:                 1                     0              0                 0
##   ---
## 1698:                 1                     0              0                 0
```

```
## 1699:                 1                    0             0                0
## 1700:                 1                    0             0                0
## 1701:                 1                    0             0                0
## 1702:                 1                    0             0                0
##       num_bedrooms_miss num_floors_in_building_miss num_total_rooms_miss
##    1:                 0                           1                    0
##    2:                 0                           0                    0
##    3:                 1                           0                    0
##    4:                 0                           1                    0
##    5:                 0                           0                    0
##   ---
## 1698:                 0                           0                    0
## 1699:                 0                           1                    0
## 1700:                 0                           1                    0
## 1701:                 0                           0                    0
## 1702:                 0                           1                    0
##       sq_footage_miss zip_code_miss median_income_miss
##    1:               0             0                  0
##    2:               0             0                  0
##    3:               1             0                  0
##    4:               1             0                  0
##    5:               0             0                  0
##   ---
## 1698:               1             0                  0
## 1699:               1             0                  0
## 1700:               0             0                  0
## 1701:               0             0                  0
## 1702:               0             0                  0
```

```
finalHousingData_Test
```

```
##      approx_year_built cats_allowed community_district_num coop_condo
##   1:              1955           no                     25      co-op
##   2:              1955           no                     25      co-op
##   3:              2004           no                     24      condo
##   4:              2002           no                     25      condo
##   5:              1949          yes                     26      co-op
##  ---
## 524:              1950           no                     28      co-op
## 525:              1947           no                     28      co-op
## 526:              2010           no                     24      condo
## 527:              2006           no                     25      condo
## 528:              1958           no                     30      co-op
##      date_of_sale dining_room_type dogs_allowed fuel_type garage_exists
##   1:           02            combo           no       gas            no
##   2:           02           formal           no       oil            no
##   3:           02            combo           no       gas            no
##   4:           02            combo           no       gas            no
##   5:           02            combo          yes       gas            no
##  ---
## 524:           02            combo           no       gas            no
## 525:           02           formal           no       gas            no
## 526:           02            combo           no       gas            no
## 527:           02            combo           no  electric            no
```

```
## 528:           02           other           no     other            no
##      kitchen_type num_bedrooms num_floors_in_building totalBathrooms
##   1:        eatin            2               6.000000            1.0
##   2:        eatin            1               7.000000            1.0
##   3:   efficiency            1               1.000000            1.0
##   4:        eatin            3               5.155000            2.0
##   5:        eatin            2               2.000000            1.0
##  ---
## 524:        eatin            2               6.000000            1.0
## 525:        combo            1               6.151667            1.0
## 526:        eatin            2               4.000000            2.0
## 527:        combo            2               5.950714            2.0
## 528:        eatin            2               7.000000            1.5
##      num_total_rooms sale_price sq_footage walk_score totalCharges zip_code
##   1:               5     228000   1012.099         82          767    11355
##   2:               4     235500    890.000         89          604    11354
##   3:               3     137550    550.000         90         5667    11368
##   4:               5     545000   1018.411         94         2535    11354
##   5:               4     241700    675.000         71          660    11426
##  ---
## 524:               4     216000    889.805         83          850    11435
## 525:               5     232500   1000.000         94          680    11374
## 526:               5     428000    820.000         96          443    11368
## 527:               4     635000   1145.338         99           70    11355
## 528:               4     310000    972.426         96          659    11372
##      median_income approx_year_built_miss community_district_num_miss
##   1:         38451                      0                           0
##   2:         43660                      0                           0
##   3:         45980                      0                           0
##   4:         43660                      0                           0
##   5:         77487                      0                           0
##  ---
## 524:         55268                      0                           0
## 525:         55550                      0                           0
## 526:         45980                      0                           0
## 527:         38451                      0                           0
## 528:         52792                      0                           0
##      date_of_sale_miss dining_room_type_miss fuel_type_miss kitchen_type_miss
##   1:                 0                     0              0                 0
##   2:                 0                     0              0                 0
##   3:                 0                     0              1                 0
##   4:                 0                     0              0                 0
##   5:                 0                     0              0                 0
##  ---
## 524:                 0                     1              0                 0
## 525:                 0                     0              0                 0
## 526:                 0                     0              0                 0
## 527:                 0                     0              0                 0
## 528:                 0                     0              0                 0
##      num_bedrooms_miss num_floors_in_building_miss num_total_rooms_miss
##   1:                 0                           0                    0
##   2:                 0                           0                    0
##   3:                 0                           0                    0
##   4:                 0                           1                    0
```

```
##   5:                 0                           0                     0
##  ---
## 524:                 0                           0                     0
## 525:                 0                           1                     0
## 526:                 0                           0                     0
## 527:                 0                           1                     0
## 528:                 0                           0                     0
##      sq_footage_miss zip_code_miss median_income_miss
##   1:               1             0                  0
##   2:               0             0                  0
##   3:               0             0                  0
##   4:               1             0                  0
##   5:               0             0                  0
##  ---
## 524:               1             0                  0
## 525:               0             0                  0
## 526:               0             0                  0
## 527:               1             0                  0
## 528:               1             0                  0
```

Feature Importance

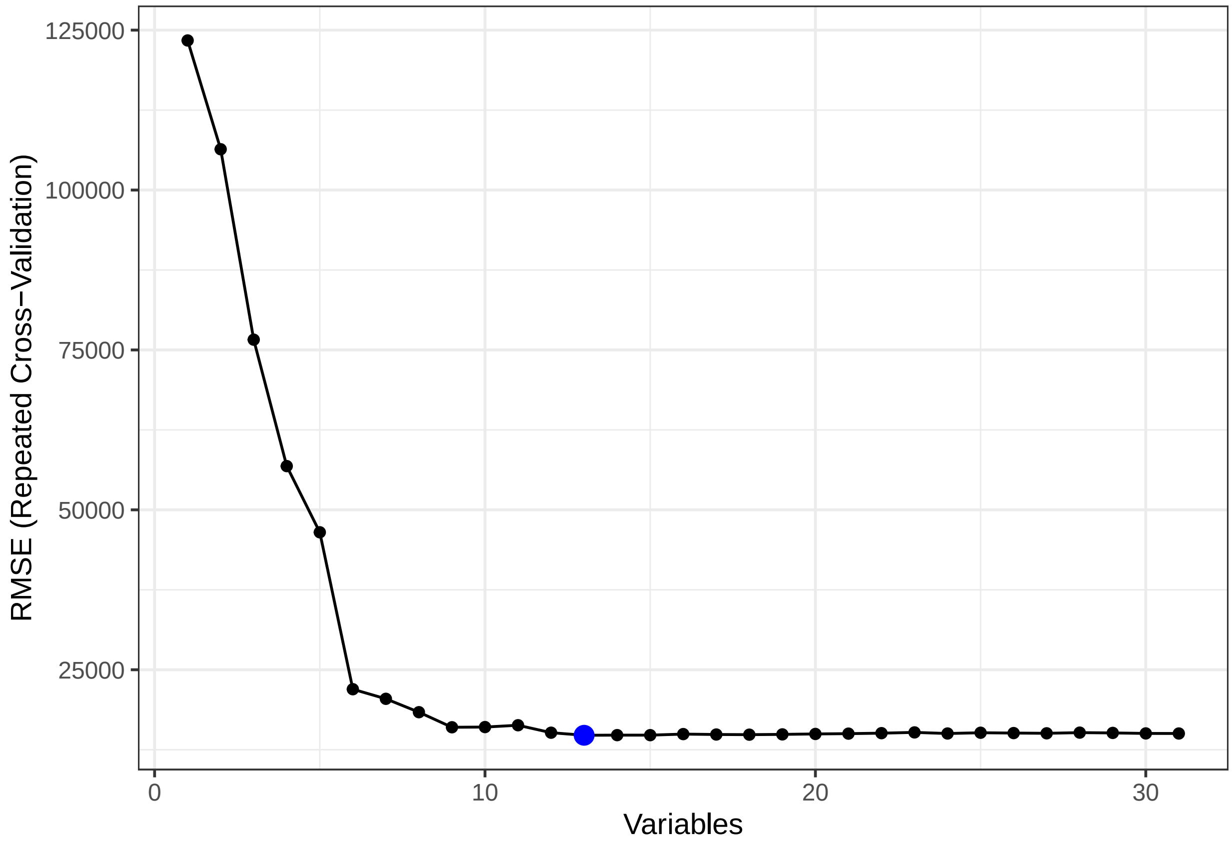
```
#Setting up parallelization cluster
cluster = makePSOCKcluster(num_of_cores)
registerDoParallel(cluster)

################################################################
#Evaluating Feature Importance

# 5 fold cross validation repeated 2 times
control_selection =  rfeControl(functions=rfFuncs, method="repeatedcv", number=5,repeats=2)

#We want to train it on the entire data just so we can see what subset of features are the best (exclud
trained_selection = rfe(data.matrix(finalHousingData_Train[,!c("sale_price")]),data.matrix(finalHousing
```

```
## Warning in rfout$mse/(var(y) * (n - 1)/n): Recycling array of length 1 in vector-array arithmetic is
##   Use c() or as.vector() instead.
```

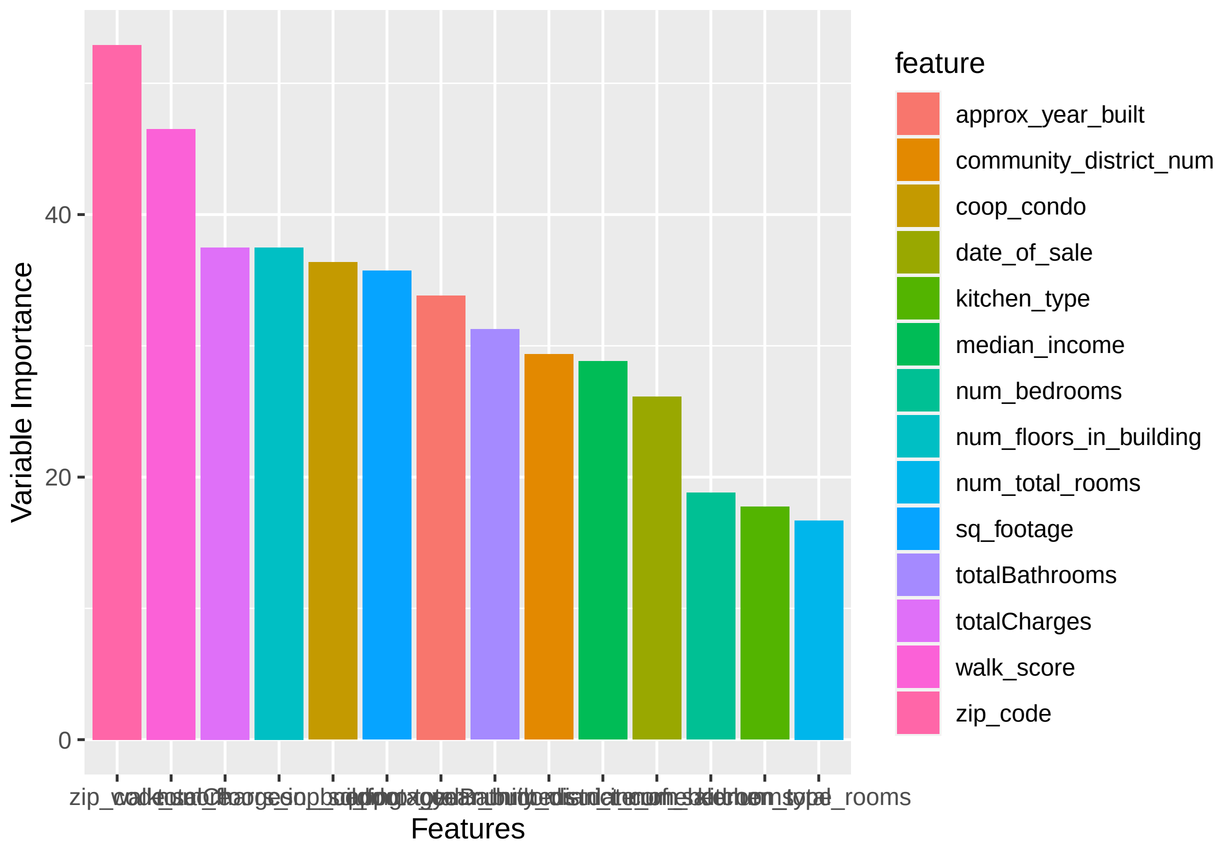
```
#Stop the cluster
stopCluster(cluster)
registerDoSEQ()

#Uncomment the following line to see a printout of the trained selection
#print(trained_selection)

predictors(trained_selection)
```

```
##  [1] "zip_code"                 "walk_score"               "totalCharges"
##  [4] "num_floors_in_building" "coop_condo"               "sq_footage"
##  [7] "approx_year_built"      "totalBathrooms"           "community_district_num"
## [10] "median_income"           "date_of_sale"             "num_bedrooms"
## [13] "kitchen_type"
```

```r
#Plot our RMSE by the number of variables
ggplot(data = trained_selection)+theme_bw()
```




```r
feat_Importance = data.frame(feature = row.names(varImp(trained_selection)), importance = varImp(trainec

ggplot(data = feat_Importance, aes(x=reorder(feature,-importance),y=importance ,fill = feature))+
  geom_bar(stat="identity")+
  labs(x = "Features", y = "Variable Importance")
```



Contending With Collinear Features

```
#Lets build a table consisting of only numeric values from finalHousingData
numericOnlyData2 = finalHousingData_Train[ , .SD, .SDcols = is.numeric]
ncol(numericOnlyData2) # total numeric columns
```

```
## [1] 24
```

```
#We expect at most p perfect collinearities in our pxp correlation matrix when i==j
#Greater than p columns indicates that there is perfect collinearity when i!=j

correlationMatrix2 = as.matrix(cor(numericOnlyData2))
```

```
## Warning in cor(numericOnlyData2): the standard deviation is zero
```

```
length(which(correlationMatrix2==1))
```

```
## [1] 24
```

Feature Selection (Using Results From Feature Importance & Collinearity Exploration)

```
#Here we implement feature selection based on the results provided from RFE and our test of perfect lin
#This was done in an effort to reduce the noise produced by irrelevant features in the  hopes of reduci

#Let's get a list of our features ranked by importance from the previous cell
varImp(trained_selection)
```

```
##                          Overall
## zip_code                52.92814
## walk_score              46.52162
## totalCharges            37.49589
## num_floors_in_building 37.48355
## coop_condo              36.36705
## sq_footage              35.72583
## approx_year_built       33.83540
## totalBathrooms          31.26524
## community_district_num 29.35821
## median_income           28.82297
## date_of_sale            26.12452
## num_bedrooms            18.80328
## kitchen_type            17.74777
## num_total_rooms         16.70658
```

```
#Thinking about this logically, it would be wise to retain sale_price_miss for the following reasons.
#For starters, sale_price was imputed and so it would be wise to retain a marker indicating this
#The sale price missing leads to there being no date of sale. No date of sale can just mean that is was
#This is a judgement call here and we choose to retain sale_price_miss

#Get the subset of features from the trained selection
subsetF = c(predictors(trained_selection))

#Create a secondary finalHousingData table with only our selected features & response
finalHousingDataImpFeat_Train = finalHousingData_Train[,..subsetF]

#Add back our sale price and sale price missingsince subsetF did not include sale_price as it was exclu
finalHousingDataImpFeat_Train[,sale_price := finalHousingData_Train[,c("sale_price")]]
```

Establishing Holdout Set II

```
#Since we are creating a secondary data set with only our selected features we want to use the same hol
#We do this so that we can truly consider our hold out test set on the sub features to be independent f

finalHousingDataImpFeat_Test = finalHousingData_Test[,..subsetF]
#Add back our sale price since subsetF did not include sale_price as it was excluded from feature impor
finalHousingDataImpFeat_Test[,sale_price := finalHousingData_Test[,c("sale_price")]] #This is our holdo

X_imp_holdout = finalHousingDataImpFeat_Test[,!c("sale_price")]
y_imp_holdout = finalHousingDataImpFeat_Test$sale_price

finalHousingDataImpFeat_Train
```

```
##       zip_code walk_score totalCharges num_floors_in_building coop_condo
##    1:    11360         82         4955              14.518722      condo
##    2:    11375         99          862               3.000000      co-op
##    3:    11375         99          738               5.000000      co-op
##    4:    11357         49         1495               8.972143      co-op
##    5:    11375         94         5776               6.000000      condo
##   ---
## 1698:    11355         97         4068               7.000000      condo
## 1699:    11360         82         6155              15.756667      condo
```

```
## 1700:    11385         96          500             4.135000      condo
## 1701:    11385         96          500             4.000000      condo
## 1702:    11360         82         4577            14.255778      condo
##       sq_footage approx_year_built totalBathrooms community_district_num
##    1:  1250.0000              1983              2                     25
##    2:   450.0000              1930              1                     28
##    3:   566.9533              1912              1                     28
##    4:  1152.6725              1953              1                     25
##    5:  1524.0000              1941              1                     28
##   ---
## 1698:   968.8313              1987              1                     25
## 1699:  1225.8587              1983              2                     25
## 1700:  1500.0000              2010              2                     24
## 1701:  1600.0000              2010              2                     24
## 1702:  1134.0000              1982              2                     25
##       median_income date_of_sale num_bedrooms kitchen_type sale_price
##    1:         82982           08         2.00        eatin   620625.0
##    2:         72982           12         1.00        combo   216285.0
##    3:         72982           12         0.85   efficiency   207496.7
##    4:         74255           01         3.00        combo   393730.0
##    5:         72982           06         4.00   efficiency   488639.0
##   ---
## 1698:         38451           10         2.00        combo   471478.4
## 1699:         82982           02         2.00        eatin   610995.5
## 1700:         60526           06         3.00        combo   575402.7
## 1701:         60526           06         3.00        combo   578105.3
## 1702:         82982           02         2.00        combo   585690.5
```

```
finalHousingDataImpFeat_Test
```

```
##      zip_code walk_score totalCharges num_floors_in_building coop_condo
##   1:    11355         82          767               6.000000      co-op
##   2:    11354         89          604               7.000000      co-op
##   3:    11368         90         5667               1.000000      condo
##   4:    11354         94         2535               5.155000      condo
##   5:    11426         71          660               2.000000      co-op
##  ---
## 524:    11435         83          850               6.000000      co-op
## 525:    11374         94          680               6.151667      co-op
## 526:    11368         96          443               4.000000      condo
## 527:    11355         99           70               5.950714      condo
## 528:    11372         96          659               7.000000      co-op
##      sq_footage approx_year_built totalBathrooms community_district_num
##   1:   1012.099              1955            1.0                     25
##   2:    890.000              1955            1.0                     25
##   3:    550.000              2004            1.0                     24
##   4:   1018.411              2002            2.0                     25
##   5:    675.000              1949            1.0                     26
##  ---
## 524:    889.805              1950            1.0                     28
## 525:   1000.000              1947            1.0                     28
## 526:    820.000              2010            2.0                     24
## 527:   1145.338              2006            2.0                     25
## 528:    972.426              1958            1.5                     30
```

```
##      median_income date_of_sale num_bedrooms kitchen_type sale_price
##   1:         38451           02            2        eatin     228000
##   2:         43660           02            1        eatin     235500
##   3:         45980           02            1   efficiency     137550
##   4:         43660           02            3        eatin     545000
##   5:         77487           02            2        eatin     241700
##  ---
## 524:         55268           02            2        eatin     216000
## 525:         55550           02            1        combo     232500
## 526:         45980           02            2        eatin     428000
## 527:         38451           02            2        combo     635000
## 528:         52792           02            2        eatin     310000
```

Quick Check on our Full Feature Set and Important Feature Set

```
#Ensure the rows in both are the same...columns will obviously be different since *ImpFeat* contains le
setequal(dim(finalHousingData_Train)[1], dim(finalHousingDataImpFeat_Train)[1])
```

```
## [1] TRUE
```

```
setequal(dim(finalHousingData_Test)[1], dim(finalHousingDataImpFeat_Test)[1])
```

```
## [1] TRUE
```

Splitting Sets Into Train & Test

```
#Let's leave ~20% of our total data for testing
K=5
prop = 1 /K

#All Feature Set
#Training & Testing data (All Features)
trainIndices_all = sample(1 : nrow(finalHousingData_Train), round((1 - prop) * nrow(finalHousingData_Tra
testIndices_all =  setdiff(1 : nrow(finalHousingData_Train), trainIndices_all)

finalHousingData_subTrain = finalHousingData_Train[trainIndices_all,]
finalHousingData_subTest = finalHousingData_Train[testIndices_all,]
X_train_all= finalHousingData_subTrain[,!c("sale_price")]
y_train_all = finalHousingData_subTrain$sale_price

X_test_all = finalHousingData_subTest[,!c("sale_price")]
y_test_all = finalHousingData_subTest$sale_price

########################################################################
#Important Feature Set
#Training & Testing data (Important Features)
trainIndices_imp = sample(1 : nrow(finalHousingDataImpFeat_Train), round((1 - prop) * nrow(finalHousing
testIndices_imp = setdiff(1 : nrow(finalHousingDataImpFeat_Train), trainIndices_imp)

finalHousingDataImpFeat_subTrain = finalHousingDataImpFeat_Train[trainIndices_imp,]
finalHousingDataImpFeat_subTest = finalHousingDataImpFeat_Train[testIndices_imp]
X_train_imp= finalHousingDataImpFeat_subTrain[,!c("sale_price")]
```

```
y_train_imp = finalHousingDataImpFeat_subTrain$sale_price

X_test_imp = finalHousingDataImpFeat_subTest[,!c("sale_price")]
y_test_imp = finalHousingDataImpFeat_subTest$sale_price
```

Quick Check For Above Cell

```
setequal((dim(finalHousingData_subTrain)[1]+dim(finalHousingData_subTest)[1]), dim(finalHousingData_Tra
```

```
## [1] TRUE
```

```
setequal((dim(finalHousingDataImpFeat_subTrain)[1]+dim(finalHousingDataImpFeat_subTest)[1]), dim(finalHo
```

```
## [1] TRUE
```

Linear Regression Model (Full Data Set)

```
#Lets run a traditional OLS with all of our features

lin_mod_all = lm(y_train_all~.,X_train_all,x = TRUE, y = TRUE)

#Test set performance
yHats_OLS_test_all = predict(lin_mod_all,X_test_all)
```

```
## Warning in predict.lm(lin_mod_all, X_test_all): prediction from a rank-deficient
## fit may be misleading
```

```
oosRMSE_OLS_test_all = sqrt(sum((y_test_all-yHats_OLS_test_all)^2)/length(y_test_all))

#Hold out set performance
yHats_OLS_holdout_all = predict(lin_mod_all,X_all_holdout)
```

```
## Warning in predict.lm(lin_mod_all, X_all_holdout): prediction from a rank-
## deficient fit may be misleading
```

```
oosRMSE_OLS_holdout_all = sqrt(sum((y_all_holdout-yHats_OLS_holdout_all)^2)/length(y_all_holdout))

oosRMSE_OLS_test_all
```

```
## [1] 49236.87
```

```
oosRMSE_OLS_holdout_all
```

```
## [1] 116163.6
```

```
#Notice we are being warned about a rank deficiency in our full feature data set. This is expected sinc
#We should not trust the first value because of this
```

Linear Regression Model (Sub Data Set)

```r
#Lets run a traditional OLS with all of our features

lin_mod_imp = lm(y_train_imp~.,X_train_imp,x = TRUE, y = TRUE)

#Test set performance
yHats_OLS_test_imp = predict(lin_mod_imp,X_test_imp)

oosRMSE_OLS_test_imp = sqrt(sum((y_test_imp-yHats_OLS_test_imp)^2)/length(y_test_imp))

#Hold out set performance
yHats_OLS_holdout_imp = predict(lin_mod_imp,X_imp_holdout)

oosRMSE_OLS_holdout_imp = sqrt(sum((y_imp_holdout-yHats_OLS_holdout_imp)^2)/length(y_imp_holdout))


SSR_olsImp_Holdout = sum((y_imp_holdout - yHats_OLS_holdout_imp) ^ 2)  ## residual sum of squares
SST_olsImp_Holdout = sum((y_imp_holdout - mean(y_imp_holdout)) ^ 2)  ## total sum of squares
Rsq_olsImp_Holdout = 1 - SSR_olsImp_Holdout/SST_olsImp_Holdout

lin_mod_imp$coefficients
```

```
##            (Intercept)               zip_code             walk_score
##          -8.802830e+05           1.161252e+00           1.020549e+03
##           totalCharges num_floors_in_building       coop_condocondo
##           1.879219e-01           6.152039e+03           1.292500e+05
##              sq_footage       approx_year_built        totalBathrooms
##           1.382113e+02           3.838950e+02           6.927525e+04
## community_district_num          median_income         date_of_sale02
##           8.987998e+02           2.854961e-01          -2.596055e+04
##         date_of_sale03         date_of_sale04         date_of_sale05
##          -2.279629e+04          -6.189772e+04          -3.282046e+04
##         date_of_sale06         date_of_sale07         date_of_sale08
##          -3.314503e+04          -2.199238e+04           4.304721e+02
##         date_of_sale09         date_of_sale10         date_of_sale11
##          -2.614294e+04          -1.044975e+04          -9.933625e+03
##         date_of_sale12           num_bedrooms      kitchen_typeeatin
##          -1.028978e+04           3.437368e+04           7.894143e+03
## kitchen_typeefficiency        kitchen_typenone
##          -1.861083e+04           1.084809e+04
```

```r
oosRMSE_OLS_test_imp
```

```
## [1] 47812.54
```

```r
oosRMSE_OLS_holdout_imp
```

```
## [1] 95162.5
```

```r
Rsq_olsImp_Holdout
```

```
## [1] 0.7184877
```

Cross Validated Linear Model (Full & Sub Data Set)

```
train_cv = trainControl(method = "cv", number = K)

#Create a model that is cross validated on the training portion of our all feature data
ols_all_cv = train(sale_price~., data=data.matrix(finalHousingData_subTrain),method="lm", trControl = tr
```

```
## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading

## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading
```

```
#Create a model that is cross validated on the training portion of our important feature data
ols_imp_cv = train(sale_price~., data=data.matrix(finalHousingDataImpFeat_subTrain),method="lm", trCont

#Predict for both models
yHats_OLS_all_cvTest = predict(ols_all_cv,data.matrix(X_test_all))
```

```
## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading
```

```
yHats_OLS_imp_cvTest = predict(ols_imp_cv,data.matrix(X_test_imp))


#Test set performance
oosRMSE_OLS_all_cvTest = sqrt(sum((y_test_all-yHats_OLS_all_cvTest)^2)/length(y_test_all)) #Here there
oosRMSE_OLS_imp_cvTest = sqrt(sum((y_test_imp-yHats_OLS_imp_cvTest)^2)/length(y_test_imp)) #It is done

#Predict for both models
yHats_OLS_all_cvHoldout = predict(ols_all_cv,data.matrix(X_all_holdout))
```

```
## Warning in predict.lm(modelFit, newdata): prediction from a rank-deficient fit
## may be misleading
```

```
yHats_OLS_imp_cvHoldout = predict(ols_imp_cv,data.matrix(X_imp_holdout))

#Hold out set performance
oosRMSE_OLS_all_cvHoldout = sqrt(sum((y_all_holdout-yHats_OLS_all_cvHoldout)^2)/length(y_all_holdout))
oosRMSE_OLS_imp_cvHoldout = sqrt(sum((y_imp_holdout-yHats_OLS_imp_cvHoldout)^2)/length(y_imp_holdout))

SSR_olsImp_cvHoldout = sum((y_imp_holdout - yHats_OLS_imp_cvHoldout) ^ 2)  ## residual sum of squares
SST_olsImp_cvHoldout = sum((y_imp_holdout - mean(y_imp_holdout)) ^ 2)  ## total sum of squares
```

```r
Rsq_olsImp_cvHoldout = 1 - SSR_olsImp_cvHoldout/SST_olsImp_cvHoldout


oosRMSE_OLS_all_cvTest
```

```
## [1] 52000.75
```

```r
oosRMSE_OLS_all_cvHoldout
```

```
## [1] 125056.4
```

```r
oosRMSE_OLS_imp_cvTest
```

```
## [1] 51348.13
```

```r
oosRMSE_OLS_imp_cvHoldout
```

```
## [1] 96693.23
```

```r
Rsq_olsImp_cvHoldout
```

```
## [1] 0.7093584
```

```r
#Notice we are being warned about a rank deficiency in our full feature data set. This is expected sinc
#We should not trust the first two values because of this
```

Linear Regression Model Cross Validated Lasso (Full Dataset)

```r
#This is mainly for fun to see how a cross validated Lasso Regression Model can tame the rank deficienc
lin_mod_lasso = cv.glmnet(data.matrix(X_train_all),y_train_all,nfolds=K,alpha = 1)
opt_Lambda = lin_mod_lasso$lambda.min

#Test Performance
yHats_LassoTest = predict(lin_mod_lasso, data.matrix(X_test_all),s = opt_Lambda)

oosRMSE_Lasso_Test = sqrt(sum((y_test_all-yHats_LassoTest)^2)/length(y_test_all))

#Holdout Set Performance
yHats_LassoHoldout = predict(lin_mod_lasso, data.matrix(X_all_holdout),s = opt_Lambda)

oosRMSE_Lasso_Holdout = sqrt(sum((y_all_holdout-yHats_LassoHoldout)^2)/length(y_all_holdout))

SSR_lasso_cvHoldout = sum((y_imp_holdout - yHats_LassoHoldout) ^ 2)  ## residual sum of squares
SST_lasso_cvHoldout = sum((y_imp_holdout - mean(y_imp_holdout)) ^ 2)  ## total sum of squares
Rsq_lasso_cvHoldout = 1 - SSR_lasso_cvHoldout/SST_lasso_cvHoldout

oosRMSE_Lasso_Test
```

```
## [1] 53042.36
```

```r
oosRMSE_Lasso_Holdout
```

```
## [1] 104030
```

```r
Rsq_lasso_cvHoldout
```

```
## [1] 0.6635792
```

```r
#At this point we will stop using the full feature data and stick with our important feature data set
```

Regression Tree Model (Important Feature Data Set)

```r
#Lets fit a regression tree to our important feature set
regTree_mod = YARFCART(X_train_imp, y_train_imp, calculate_oob_error = FALSE)
```

```
## YARF initializing with a fixed 1 trees...
## YARF factors created...
## YARF after data preprocessed... 28 total features...
## Beginning YARF regression model construction...done.
```

```r
#Test performance
yHats_RegTree_Test = predict(regTree_mod,X_test_imp)

oosRMSE_RegTree_Test = sqrt(sum((y_test_imp-yHats_RegTree_Test)^2)/length(y_test_imp))

#Holdout Set Performance
yHats_RegTree_Holdout = predict(regTree_mod,X_imp_holdout)

oosRMSE_RegTree_Holdout = sqrt(sum((y_imp_holdout-yHats_RegTree_Holdout)^2)/length(y_imp_holdout))

SSR_regTree_Holdout = sum((y_imp_holdout - yHats_RegTree_Holdout) ^ 2)  ## residual sum of squares
SST_regTree_Holdout = sum((y_imp_holdout - mean(y_imp_holdout)) ^ 2)  ## total sum of squares
Rsq_regTree_Holdout = 1 - SSR_regTree_Holdout/SST_regTree_Holdout

#Uncomment the following line to save an illustration of the tree
#illustrate_trees(regTree_mod, max_depth=5, open_file=TRUE)

oosRMSE_RegTree_Test
```

```
## [1] 20698.89
```

```r
oosRMSE_RegTree_Holdout
```

```
## [1] 76429.7
```

```r
Rsq_regTree_Holdout
```

```
## [1] 0.8184108
```

Random Forest Model (Important Feature Data Set)

```r
#Lets fit a random Forest to our important feature set
rf_mod = YARF(X_train_imp, y_train_imp, calculate_oob_error = FALSE)
```

```
## YARF initializing with a fixed 500 trees...
## YARF factors created...
## YARF after data preprocessed... 28 total features...
## Beginning YARF regression model construction...done.
```

```r
#Test performance
yHats_rf_Test = predict(rf_mod,X_test_imp)

oosRMSE_rf_Test = sqrt(sum((y_test_imp-yHats_rf_Test)^2)/length(y_test_imp))

#Holdout Set Performance
yHats_rf_Holdout = predict(rf_mod,X_imp_holdout)

oosRMSE_rf_Holdout = sqrt(sum((y_imp_holdout-yHats_rf_Holdout)^2)/length(y_imp_holdout))

SSR_rf_Holdout = sum((y_imp_holdout - yHats_rf_Holdout) ^ 2)  ## residual sum of squares
SST_rf_Holdout = sum((y_imp_holdout - mean(y_imp_holdout)) ^ 2)  ## total sum of squares
Rsq_rf_Holdout = 1 - SSR_rf_Holdout/SST_rf_Holdout

oosRMSE_rf_Test
```

```
## [1] 13210.72
```

```r
oosRMSE_rf_Holdout
```

```
## [1] 73465.9
```

```r
Rsq_rf_Holdout
```

```
## [1] 0.8322211
```

Bagged Random Forest Model (Important Feature Data Set)

```r
#Lets fit a bagged random forest to our important feature set
rfBag_mod = YARFBAG(X_train_imp, y_train_imp, calculate_oob_error = TRUE)
```

```
## YARF initializing with a fixed 500 trees...
## YARF factors created...
## YARF after data preprocessed... 28 total features...
## Beginning YARF regression model construction...done.
## Calculating OOB error...done.
```

```r
#Out of Bag Performance
oosRMSE_brf_Bag = rfBag_mod$rmse_oob

#Holdout Set Performance
yHats_brf_Holdout = predict(rfBag_mod,X_imp_holdout)
```

```r
oosRMSE_brf_Holdout = sqrt(sum((y_imp_holdout-yHats_brf_Holdout)^2)/length(y_imp_holdout))

SSR_rfBag_Holdout = sum((y_imp_holdout - yHats_brf_Holdout) ^ 2)  ## residual sum of squares
SST_rfBag_Holdout = sum((y_imp_holdout - mean(y_imp_holdout)) ^ 2)  ## total sum of squares
Rsq_rfBag_Holdout = 1 - SSR_rfBag_Holdout/SST_rfBag_Holdout

oosRMSE_brf_Bag
```

```
## [1] 19434.22
```

```r
oosRMSE_brf_Holdout
```

```
## [1] 72718.72
```

```r
Rsq_rfBag_Holdout
```

```
## [1] 0.8356165
```

Bagged Random Forest Model Optimization (Hyper-Parameter Tuning)

```r
#Hyper-Parameter Tuning
#Setting up parallelization cluster
cluster = makePSOCKcluster(num_of_cores)
registerDoParallel(cluster)


control_rf = trainControl(method='repeatedcv', number=K, repeats=2,search = 'random')

mtry = ncol(finalHousingDataImpFeat_subTrain) # Columns in our important feature set
nTree = 500
tunegrid = expand.grid(.mtry=seq(1,mtry))

rf_optimized = train(sale_price~.,
                      data=data.matrix(finalHousingDataImpFeat_subTrain),
                      method='rf',
                      metric='RMSE',
                      tuneGrid=tunegrid,
                      nTree = nTree,
                      trControl=control_rf
                     )

#Stop the cluster
stopCluster(cluster)
registerDoSEQ()

#Holdout Set Performance
yHats_bgfOpt_Holdout = predict(rf_optimized,data.matrix(X_imp_holdout))

oosRMSE_bgfOpt_Holdout = sqrt(sum((y_imp_holdout-yHats_bgfOpt_Holdout)^2)/length(y_imp_holdout))

SSR_bgfOpt_Holdout = sum((y_imp_holdout - yHats_bgfOpt_Holdout) ^ 2)  ## residual sum of squares
```

```r
SST_bgfOpt_Holdout = sum((y_imp_holdout - mean(y_imp_holdout)) ^ 2)  ## total sum of squares
Rsq_bgfOpt_Holdout = 1 - SSR_bgfOpt_Holdout/SST_bgfOpt_Holdout

print(rf_optimized)
```

```
## Random Forest
##
## 1362 samples
##   13 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold, repeated 2 times)
## Summary of sample sizes: 1089, 1091, 1089, 1089, 1090, 1088, ...
## Resampling results across tuning parameters:
##
##   mtry  RMSE      Rsquared   MAE
##    1    33450.78  0.9714715  26127.95
##    2    19905.75  0.9847797  14428.55
##    3    18109.15  0.9862737  12764.24
##    4    17816.12  0.9861786  12388.48
##    5    17768.79  0.9859781  12266.81
##    6    17979.31  0.9854403  12353.24
##    7    18178.53  0.9849586  12431.33
##    8    18467.35  0.9843565  12527.56
##    9    18914.51  0.9834826  12754.23
##   10    19217.54  0.9828363  12949.49
##   11    19686.39  0.9818814  13191.34
##   12    20358.54  0.9804927  13501.52
##   13    21066.04  0.9790198  13837.54
##   14    20948.49  0.9792665  13794.39
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 5.
```

```r
oosRMSE_bgfOpt_Holdout
```

```
## [1] 70851.02
```

```r
Rsq_bgfOpt_Holdout
```

```
## [1] 0.8439521
```

Final Shipped Model Trained On All Data

```r
#Hyper-Parameter Tuning
#Setting up parallelization cluster
cluster = makePSOCKcluster(num_of_cores)
registerDoParallel(cluster)

#Lets combine the Train and Test Portion of our important feature data set into a single entity
finalHousingData_ImpFeat = rbind(finalHousingDataImpFeat_Train,finalHousingDataImpFeat_Test)
```

```r
control_rf = trainControl(method='repeatedcv', number=K, repeats=2,search = 'random')

mtry = ncol(finalHousingData_ImpFeat) # Columns in our important feature set
nTree = 500
tunegrid = expand.grid(.mtry=seq(1,mtry))

rf_optimizedFinal = train(sale_price~.,
                      data=data.matrix(finalHousingData_ImpFeat),
                      method='rf',
                      metric='RMSE',
                      tuneGrid=tunegrid,
                      nTree = nTree,
                      trControl=control_rf
                     )

#Stop the cluster
stopCluster(cluster)
registerDoSEQ()

print(rf_optimizedFinal)
```

```
## Random Forest
##
## 2230 samples
##   13 predictor
##
## No pre-processing
## Resampling: Cross-Validated (5 fold, repeated 2 times)
## Summary of sample sizes: 1784, 1784, 1784, 1785, 1783, 1784, ...
## Resampling results across tuning parameters:
##
##   mtry  RMSE      Rsquared   MAE
##    1    44849.87  0.9369808  29319.55
##    2    34724.15  0.9516299  17739.55
##    3    33221.06  0.9544861  16407.39
##    4    32788.01  0.9553193  16234.95
##    5    32550.58  0.9557380  16187.83
##    6    32579.45  0.9554463  16356.35
##    7    32546.34  0.9554195  16450.45
##    8    32656.63  0.9550071  16644.05
##    9    32646.81  0.9549716  16820.57
##   10    32889.58  0.9541840  17060.57
##   11    33107.49  0.9535409  17376.83
##   12    33570.92  0.9521151  17850.41
##   13    34012.33  0.9507760  18332.96
##   14    34051.33  0.9506865  18348.73
##
## RMSE was used to select the optimal model using the smallest value.
## The final value used for the model was mtry = 7.
```