Determining Factors Influencing House Sale Prices

Kile Gilde, Jaan Bernberg, Kai Lukowiak,
Michael Muller, Ilya Kats
DATA 621, Master of Science in Data Science,
City University of New York

Abstract

TBD: Since it summarizes the work, it will be written at the end. 250 words or less summarizing the problem, methodology, and major outcomes.

Key Words

house prices, regression, linear models, assessed value

Introduction

This project stems out of the Business Analytics and Data Mining class in the Master of Science in Data Science program at CUNY. This paper is the result of the final class group project in applying regression methods to real-world data. Our team chose housing data because it promissed to be an interesting and useful subject. In addition, this research is based on a well studied data set which makes it an excellent educational resource allowing our team to study various approaches.

The data set was prepared by Dean De Cock in an effort to create a real-world data set to test and practice regression methods (De Cock 2011). It describes the sale of individual residential property in Ames, Iowa from 2006 to 2010. Ames, Iowa was founded in 1864 as a station stop. It has the population of about 60,000 people and covers about 24.27 sq mi. It was ranked ninth on the *Best Places to Live* list (CNNMoney 2010).

The data came directly from the Assessor's Office in the form of a data dump from their records system and it included information for calculation of assessed values in the city's assessment process. The data is recent and it covers the period of housing bubble collapse that led to the subprime mortgage crisis. 2008 saw one of the largest housing price drops in history.

Each of over 2,900 total observations in the data represent attributes of a residental property sold. For properties that exchanged ownership multiple times during the collection period (2006 through 2010), only the last sale is included in the data since it represents the most current value of the property. The attributes that make up the sale price of a house can seem daunting given a myriad of factors that can impact its value. There are about 80 variables included in the data set. Most variables describe physical attributes of the property. There is a variety of variable types - discrete, continous, categorical (both nominal and ordinal).

Data set was downloaded from Kaggle.com which gave us the ability to compare our results with results of other teams working with this data set (Kaggle 2016).

Literature Review

- 1 page
- Discuss how other researchers have addressed similar problems, what their achievements are, and what the advantage and drawbacks of each reviewed approach are.

- Explain how your investigation is similar or different to the state-of-theart.

Pardoe, I. https://ww2.amstat.org/publications/jse/v16n2/datasets.pardoe.html

- One interesting implementation - half-bathrooms are counted as 0.1 out of belief that buyers do not value them as much as full bathrooms.

Luttik, J. https://www.sciencedirect.com/science/article/pii/S0169204600000396 The value of trees, water and open space as reflected by house prices in the Netherlands

Methodology

- 2-3 pages
- Discuss high-level exploratory data analysis - how data was prepared
- Discuss high-level regression modeling
- Discuss high-level model building and model selection

Data Description

The data set includes 2,919 observation and 79 indepedent variables. Out of those 36 are numeric, such as lot area or pool area in square feet, and 43 are categorical, such as garage type (attached to home, built-in, carport, etc.) or utilities (gas, sewer, both, etc.).

Data Imputation

Original data set included no complete observations (*see table 1*). However, many `NA` values found in the data carry useable information. For example, `NA` in the `PoolQC` variable (pool quality) implies that the property has no pool. Often this logic carried across multiple variables - for example, `NA` in `GarageQual` (garage quality), `GarageCond` (garage condition) and `GarageType` variables all imply that the property has no garage. This type of missing values was replaced with a new category - *No Pool*, *No Garage* or similar. After this substitution there remained only 58 observations with true missing values.

...

Experimentation and Results

- 4-5 pages
- Key figures and tables may be included here
- Additional figures and tables should be added to appendices
- Discuss data prepatation details not mentioned under Methodology
- Discuss model building and selection
- Discuss model validation
- Discuss results of statistical analysis
- Describe final model (coefficients, interpretation)
- Discuss upload of results to Kaggle

Discussion

- 1-2 pages
- Discuss limitations

- Discuss areas for future work
- Discuss detailed findings
- May be combined with Conclusion section below

Conclusion

- 1 paragraph
- Quick summary of findings

Appendix A. Figures

Figure 2.

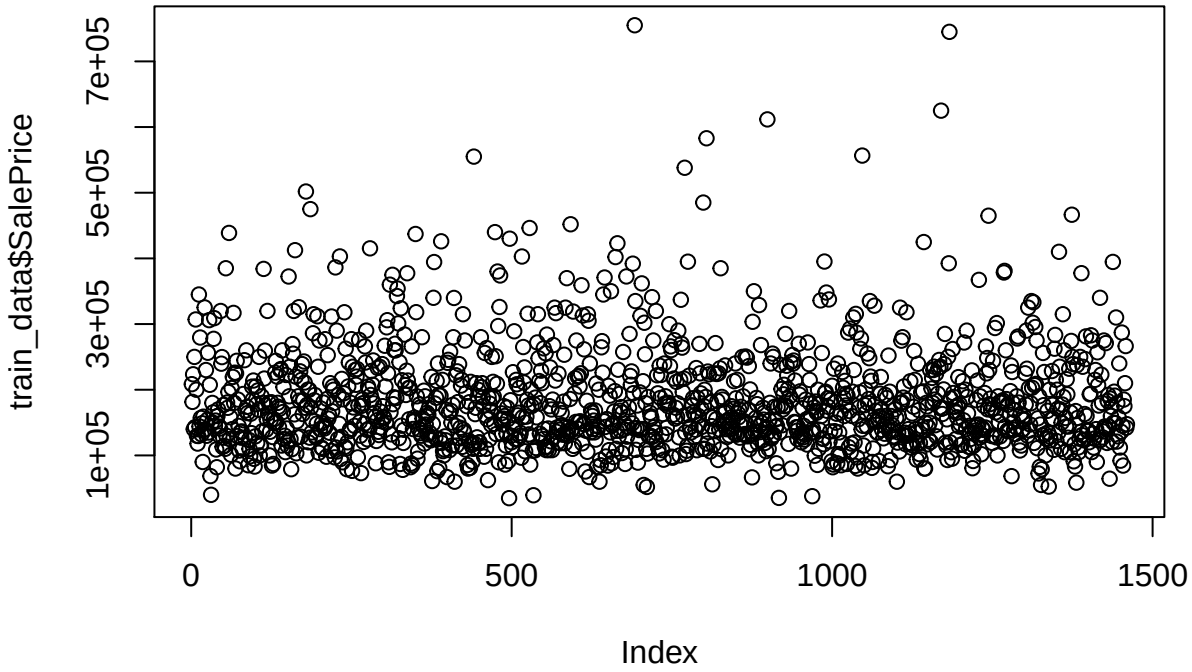
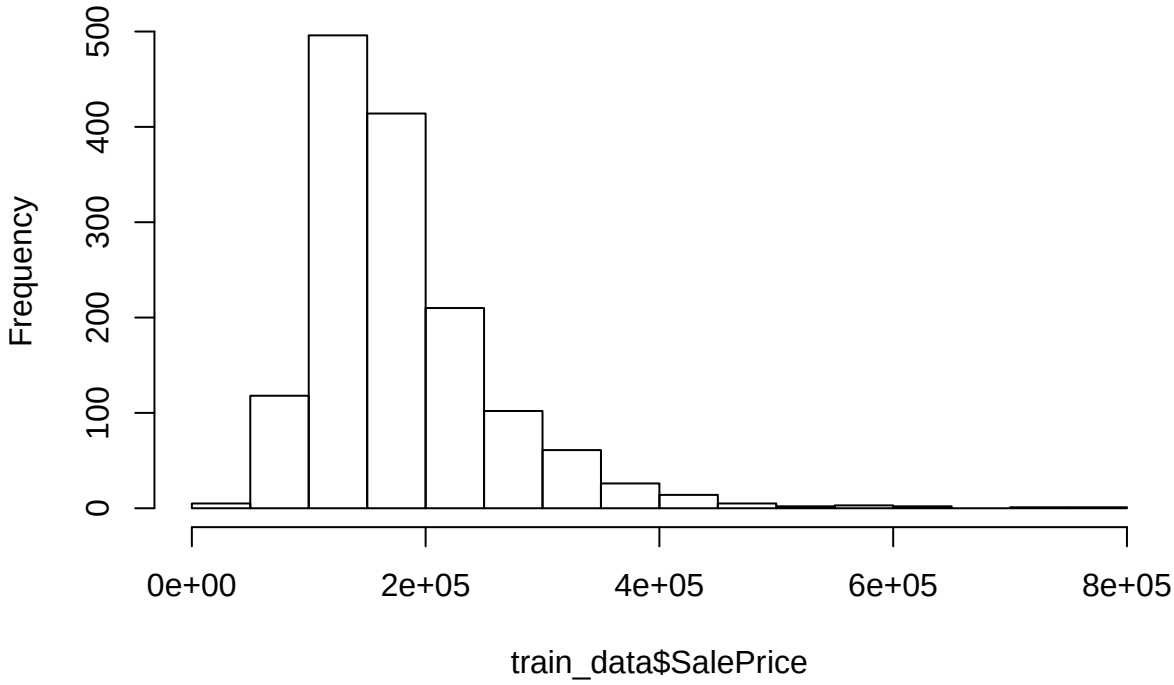


Figure 3.


Frequency
0
100
200
300
400
500
0e+00
2e+05
4e+05
6e+05
8e+05
train_data$SalePrice

Appendix B. Tables

Table 1: Number of NA values in original data.

Variable	No of NAs	Percent of Total Obs
PoolQC	2909	99.66
MiscFeature	2814	96.40
Alley	2721	93.22
Fence	2348	80.44
FireplaceQu	1420	48.65
LotFrontage	486	16.65
GarageYrBlt	159	5.45
GarageFinish	159	5.45
GarageQual	159	5.45
GarageCond	159	5.45
Garage_Age_Yrs	159	5.45
GarageType	157	5.38
BsmtCond	82	2.81
BsmtExposure	82	2.81
BsmtQual	81	2.77
BsmtFinType2	80	2.74
BsmtFinType1	79	2.71
MasVnrType	24	0.82
MasVnrArea	23	0.79
MSZoning	4	0.14
Utilities	2	0.07
BsmtFullBath	2	0.07
BsmtHalfBath	2	0.07
Functional	2	0.07
Exterior1st	1	0.03
Exterior2nd	1	0.03
BsmtFinSF1	1	0.03
BsmtFinSF2	1	0.03
BsmtUnfSF	1	0.03
TotalBsmtSF	1	0.03
Electrical	1	0.03
KitchenQual	1	0.03
GarageCars	1	0.03
GarageArea	1	0.03
SaleType	1	0.03

Table 2: Descriptive statistics.

	Count	Mean	SD	Median	Min	Max	Kurtosis
LotFrontage	2919	57.77	33.48	63	0	313	2.169
LotArea	2919	10168	7887	9453	1300	215245	264.3
OverallQual	2919	6.089	1.41	6	1	10	0.06295
OverallCond	2919	5.565	1.113	5	1	9	1.472
YearBuilt	2919	1971	30.29	1973	1872	2010	-0.5142
YearRemodAdd	2919	1984	20.89	1993	1950	2010	-1.347
MasVnrArea	2896	102.2	179.3	0	0	1600	9.228
BsmtFinSF1	2918	441.4	455.6	368.5	0	5644	6.884
BsmtFinSF2	2918	49.58	169.2	0	0	1526	18.79
BsmtUnfSF	2918	560.8	439.5	467	0	2336	0.3985
TotalBsmtSF	2918	1052	440.8	989.5	0	6110	9.125
X1stFlrSF	2919	1160	392.4	1082	334	5095	6.936
X2ndFlrSF	2919	336.5	428.7	0	0	2065	-0.4254
LowQualFinSF	2919	4.694	46.4	0	0	1064	174.5
GrLivArea	2919	1501	506.1	1444	334	5642	4.108
BsmtFullBath	2917	0.4299	0.5247	0	0	3	-0.738
BsmtHalfBath	2917	0.06136	0.2457	0	0	2	14.81
FullBath	2919	1.568	0.553	2	0	4	-0.5409
HalfBath	2919	0.3803	0.5029	0	0	2	-1.035
BedroomAbvGr	2919	2.86	0.8227	3	0	8	1.933
KitchenAbvGr	2919	1.045	0.2145	1	0	3	19.73
TotRmsAbvGrd	2919	6.452	1.569	6	2	15	1.162
Fireplaces	2919	0.5971	0.6461	1	0	4	0.07213
GarageYrBlt	2918	2412	1816	1984	1895	9999	13.51
GarageCars	2918	1.767	0.7616	2	0	5	0.2335
GarageArea	2918	472.9	215.4	480	0	1488	0.9334
WoodDeckSF	2919	93.71	126.5	0	0	1424	6.721
OpenPorchSF	2919	47.49	67.58	26	0	742	10.91
EnclosedPorch	2919	23.1	64.24	0	0	1012	28.31
X3SsnPorch	2919	2.602	25.19	0	0	508	149
ScreenPorch	2919	16.06	56.18	0	0	576	17.73
PoolArea	2919	2.252	35.66	0	0	800	297.9
MiscVal	2919	50.83	567.4	0	0	17000	562.7
MoSold	2919	6.213	2.715	6	1	12	-0.4574
YrSold	2919	2008	1.315	2008	2006	2010	-1.156
House_Age_Yrs	2919	36.48	30.34	35	-1	136	-0.5058
RemodAdd_Age_Yrs	2919	23.53	20.89	15	-2	60	-1.339
Garage_Age_Yrs	2918	28.07	25.8	25	-200	114	1.614

Appendix C. R Code

```
knitr::opts_chunk$set(error = F, message = F, # tidy = T,
                      cache = T, warning = T,
                      results = 'hide', # suppress code output
                      echo = F,         # suppress code
                      fig.show = 'hide' # suppress plots
                      )

install_load <- function(pkg){
  # Load packages & Install them if needed.
  # CODE SOURCE: https://gist.github.com/stevenworthington/3178163
  new.pkg <- pkg[!(pkg %in% installed.packages()[, "Package"])]
  if (length(new.pkg)) install.packages(new.pkg, dependencies = TRUE)
  sapply(pkg, require, character.only = TRUE, quietly = TRUE, warn.conflicts = FALSE)
}

# Required packages
packages <- c("tidyverse", "RCurl",               # Data input and manipulation
              "knitr", "knitcitations", "pander", # Data output
              "ggthemes",                         # Plotting
              "mice", "VIM"                       # Imputation
              )
install_load(packages)
##Read data
url_train <- paste0("https://raw.githubusercontent.com/kaiserxc/DATA621FinalProject/",
                    "master/house-prices-advanced-regression-techniques/train.csv")
url_test <-  paste0("https://raw.githubusercontent.com/kaiserxc/DATA621FinalProject/",
                    "master/house-prices-advanced-regression-techniques/test.csv")

stand_read <- function(url){
  return(read.csv(text = getURL(url)))
}

o_train <-
  stand_read(url_train) %>%
  mutate(d_name = 'train')
o_test <- stand_read(url_test) %>%
  mutate(SalePrice = NA, d_name = 'test')

full_set <- rbind(o_train, o_test)
# x <- plot_missing(full_set)
na_review <- function(df){
  # returns df of vars w/ NA qty desc.
  na_qty <- colSums(is.na(df)) %>% as.data.frame(stringsAsFactors=F)
  colnames(na_qty) <- c("NA_qty")
  na_qty <- cbind('Variable' = rownames(na_qty), na_qty) %>%
    select(Variable, NA_qty)
  rownames(na_qty) <- NULL

  na_qty <- na_qty %>%
    arrange(desc(NA_qty)) %>% filter(NA_qty > 0) %>%
    mutate(Variable = as.character(Variable)) %>%
```

```
    mutate(Pct_of_Tot =  round(NA_qty/nrow(df), 4) * 100)

  return(na_qty)
}

first_pass <- full_set %>%
  # first_pass is train.csv and test.csv combined for NA reviews
  # and imputation planning and calculated columns
  mutate(House_Age_Yrs = YrSold - YearBuilt,
         RemodAdd_Age_Yrs = YrSold - YearRemodAdd,
         Garage_Age_Yrs = YrSold - GarageYrBlt)
naVars <- na_review(first_pass %>% select(-SalePrice))
colnames(naVars) <- c("Variable", "No of NAs", "Percent of Total Obs")
pander(naVars, keep.trailing.zeros=TRUE,
       caption="Number of NA values in original data.",
       justify=c("left", "right", "right"))


set_aside <- c(2600, 2504, 2421, 2127, 2041, 2186, 2525, 1488, 949, 2349,
               2218, 2219, 333)
#View(first_pass[is.na(first_pass$PoolQC), ]) # 2600, 2504, 2421
#View(first_pass[is.na(first_pass$GarageFinish), ]) # 2127
#View(first_pass[is.na(first_pass$GarageQual), ]) # 2127
#View(first_pass[is.na(first_pass$GarageCond), ]) # 2127
#View(first_pass[is.na(first_pass$BsmtCond), ]) # 2041, 2186, 2525
#View(first_pass[is.na(first_pass$BsmtExposure), ]) # 1488, 949, 2349
#View(first_pass[is.na(first_pass$BsmtQual), ]) # 2218, 2219
#View(first_pass[is.na(first_pass$BsmtFinType2), ]) # 333
#View(first_pass[is.na(first_pass$MasVnrType), ]) #

#qty
# first_pass[first_pass$PoolArea == 0, ]      # 2,906
# first_pass[is.na(first_pass$PoolQC), ]
# first_pass[is.na(first_pass$Alley), ]       # 2,721
# first_pass[is.na(first_pass$Fence), ]       # 2,348
# first_pass[first_pass$Fireplaces == 0, ]    # 1,420
# first_pass[is.na(first_pass$GarageType),]   # 157
# first_pass[is.na(first_pass$GarageArea),]   # 1
# first_pass[is.na(first_pass$GarageFinish),] # 159
# first_pass[first_pass$GarageArea == 0, ]    # 158
# first_pass[first_pass$TotalBsmtSF == 0, ]   # 79
# first_pass[is.na(first_pass$Electrical),]   # 1
set_asideA <- '2600|2504|2421|2127|2041|2186|2525|1488|949|2349|2218|2219|333' # 13
set_asideB <- '|2550|524|2296|2593' # negative values in '_Age' columns

x <- first_pass %>%
  # exclude set_aside observations to fill in known NA's
  filter(!grepl(paste0(set_asideA, set_asideB), Id))

naVarsx <- na_review(x %>% select(-SalePrice))
# naVarsx
```

```r
nrow(x[x$PoolArea==0, ])   # 2,887
# x[is.na(x$MiscFeature),]   # 2,793
# x[is.na(x$Alley),]         # 2,700
# x[is.na(x$Fence),]         # 2,331
# x[is.na(x$FireplaceQu),]   # 1,414
# nrow(x[x$LotFrontage==0, ])# 486
# x[is.na(x$GarageArea),]    # 158
# x[x$TotalBsmtSF == 0, ]    # 78
obtain_data <- function(df){
  # like first_pass but with imputation that addresses
  # observations that have known NA's
  df %>%
    mutate(PoolQC = fct_explicit_na(PoolQC, na_level='NoP'),
           MiscFeature = fct_explicit_na(MiscFeature, na_level='NoM'),
           Alley = fct_explicit_na(Alley, na_level='NoA'),
           Fence = fct_explicit_na(Fence, na_level = 'NoF'),
           FireplaceQu = fct_explicit_na(FireplaceQu, na_level = 'NoFp'),
           LotFrontage = ifelse(is.na(LotFrontage), 0, LotFrontage),

           # Note GarageYrBlt set to 9999 may be a problem
           GarageYrBlt = ifelse(is.na(GarageYrBlt), 9999, GarageYrBlt),
           GarageFinish = fct_explicit_na(GarageFinish, na_level = 'NoG'),
           GarageQual = fct_explicit_na(GarageQual, na_level = 'NoG'),
           GarageCond = fct_explicit_na(GarageCond, na_level = 'NoG'),
           # NOTE: Garage_Age_Yrs: 0 doesn't seem appropriate...
           Garage_Age_Yrs = ifelse(is.na(Garage_Age_Yrs), 0, Garage_Age_Yrs),
           GarageType = fct_explicit_na(GarageType, na_level = 'NoG'),

           BsmtQual = fct_explicit_na(BsmtQual, na_level = 'NoB'),
           BsmtCond = fct_explicit_na(BsmtCond, na_level = 'NoB'),
           BsmtExposure = fct_explicit_na(BsmtExposure, na_level = 'NoB'),
           BsmtFinType1 = fct_explicit_na(BsmtFinType1, na_level = 'NoB'),
           BsmtFinType2 = fct_explicit_na(BsmtFinType2, na_level = 'NoB')
           )
}
probl_obs <- full_set %>%
  mutate(House_Age_Yrs = YrSold - YearBuilt,
         RemodAdd_Age_Yrs = YrSold - YearRemodAdd,
         Garage_Age_Yrs = YrSold - GarageYrBlt) %>%
  filter(grepl(paste0(set_asideA, set_asideB), Id))

known_obs <- full_set %>%
  filter(!grepl(paste0(set_asideA, set_asideB), Id)) %>%
  mutate(House_Age_Yrs = YrSold - YearBuilt,
         RemodAdd_Age_Yrs = YrSold - YearRemodAdd,
         Garage_Age_Yrs = YrSold - GarageYrBlt)

full_set_clean <- rbind(obtain_data(known_obs), probl_obs) %>% arrange(Id)
#View(full_set_clean)
summary(full_set_clean)
naVarsy <- na_review(full_set_clean %>% select(-SalePrice))
sum(naVarsy$NA_qty) # 176
# unique(full_set_clean$Alley) # NoA  Grvl Pave <NA>, levels: Grvl Pave NoA
```

```r
# unique(full_set_clean$PoolQC) # NoP  Ex   <NA> Fa   Gd, levels: Ex Fa Gd NoP
# unique(full_set_clean$GarageYrBlt) # character!
var_types <- function(df){
  # returns df of Variable name and Type from df
  var_df <- sapply(df, class) %>% as.data.frame()
  colnames(var_df) <- c("Var_Type")
  var_df <- cbind(var_df, 'Variable' = rownames(var_df)) %>%
    select(Variable, Var_Type) %>%
    mutate(Variable = as.character(Variable),Var_Type = as.character(Var_Type))
  return(var_df)
}

var_review <- var_types(full_set_clean %>% select(-c(Id,SalePrice,d_name)))

fac_vars <- var_review %>% filter(Var_Type == 'factor') %>%
  select(Variable) %>% t() %>% as.character() # 43 total length(fac_vars)
num_vars <- var_review %>% filter(grepl('character|integer|numeric', Var_Type)) %>%
  select(Variable) %>% t() %>% as.character() # 39 total but see GarageYrBlt
#sum(complete.cases(full_set %>% select(-SalePrice)))       # 0
#sum(complete.cases(full_set_clean %>% select(-SalePrice))) # 2,861 ~ 98%
#nrow(full_set_clean) - 2861 # 58 NA
stat_info <- psych::describe(full_set_clean %>% select(num_vars, -Id, -d_name))
stat_tbl <- stat_info[c(2:nrow(stat_info)),c(2:5,8:9,13:ncol(stat_info)-1)]
colnames(stat_tbl) <- c("Count", "Mean", "SD", "Median", "Min", "Max", "Kurtosis")
pander(stat_tbl, caption="Descriptive statistics.",
       digits=4, emphasize.rownames=FALSE,
       justify=c("left", "right", "right", "right", "right", "right", "right", "right"))
summary(full_set_clean %>% select(fac_vars, -Id, -SalePrice, -d_name))
train_data <- full_set_clean %>% filter(d_name == 'train') %>% select(-d_name)
test_data <- full_set_clean %>% filter(d_name == 'test') %>% select(-d_name)
##View(train_data)
dim(train_data)
dim(test_data)
full_set_clean %>%
  filter(Garage_Age_Yrs < 0 | RemodAdd_Age_Yrs < 0 | Garage_Age_Yrs < 0)
# Ids c(524, 2296, 2550, 2593)
plot(train_data$SalePrice, main = "Figure 2.")
hist(train_data$SalePrice, main = "Figure 3.")
```

References

Chang, W., J. Cheng, JJ. Allaire, Y. Xie, and J. McPherson. 2015. "Shiny: Web Application Framework for R. R Package Version 0.12.1." Computer Program. http://CRAN.R-project.org/package=shiny.

CNNMoney. 2010. "Best Places to Live." http://money.cnn.com/magazines/moneymag/bplive/2010/snapshots/PL1901855.html.

De Cock, D. 2011. "Ames, Iowa: Alternative to the Boston Housing Data as an End of Semester Regression Project." Journal article. https://ww2.amstat.org/publications/jse/v19n3/decock.pdf.

Kaggle. 2016. "House Prices: Advanced Regression Techniques," August. https://www.kaggle.com/c/house-prices-advanced-regression-techniques.

R Core Team. 2015. "R: A Language and Environment for Statistical Computing." Journal Article. http://www.R-project.org.

Wickham, Hadley. 2009. *Ggplot2: Elegant Graphics for Data Analysis (Use R!)*. New York, NY. http://ggplot2.org.