1. Import the Titanic Dataset from the link Titanic Data Set.

Perform the following:

a. Preprocess the passenger names to come up with a list of titles that represent families and represent using appropriate visualization graph.

library('ggplot2') # visualization

library('ggthemes') # visualization

library('scales') # visualization

library('dplyr') # data manipulation

library('mice') # imputation

library('randomForest') # classification algorithm

train <- read.csv('../input/train.csv', stringsAsFactors = F)

test <- read.csv('../input/test.csv', stringsAsFactors = F)

full <- bind\_rows(train, test) # bind training & test data

# check data

str(full)

## 'data.frame': 1309 obs. of 12 variables:

## $ PassengerId: int 1 2 3 4 5 6 7 8 9 10 ...

## $ Survived : int 0 1 1 1 0 0 0 0 1 1 ...

## $ Pclass : int 3 1 3 1 3 3 1 3 3 2 ...

## $ Name : chr "Braund, Mr. Owen Harris" "Cumings, Mrs. John Bradley (Florence Briggs Thayer)" "Heikkinen, Miss. Laina" "Futrelle, Mrs. Jacques Heath (Lily May Peel)" ...

## $ Sex : chr "male" "female" "female" "female" ...

## $ Age : num 22 38 26 35 35 NA 54 2 27 14 ...

## $ SibSp : int 1 1 0 1 0 0 0 3 0 1 ...

## $ Parch : int 0 0 0 0 0 0 0 1 2 0 ...

## $ Ticket : chr "A/5 21171" "PC 17599" "STON/O2. 3101282" "113803" ...

## $ Fare : num 7.25 71.28 7.92 53.1 8.05 ...

## $ Cabin : chr "" "C85" "" "C123" ...

## $ Embarked : chr "S" "C" "S" "S" ...

# Grab title from passenger names

full$Title <- gsub('(.\*, )|(\\..\*)', '', full$Name)

# Show title counts by sex

table(full$Sex, full$Title)

##

## Capt Col Don Dona Dr Jonkheer Lady Major Master Miss Mlle Mme

## female 0 0 0 1 1 0 1 0 0 260 2 1

## male 1 4 1 0 7 1 0 2 61 0 0 0

##

## Mr Mrs Ms Rev Sir the Countess

## female 0 197 2 0 0 1

## male 757 0 0 8 1 0+

b. Represent the proportion of people survived from the family size using a graph.

# Use ggplot2 to visualize the relationship between family size & survival

ggplot(full[1:891,], aes(x = Fsize, fill = factor(Survived))) +

geom\_bar(stat='count', position='dodge') +

scale\_x\_continuous(breaks=c(1:11)) +

labs(x = 'Family Size') +

theme\_few()

# Discretize family size

full$FsizeD[full$Fsize == 1] <- 'singleton'

full$FsizeD[full$Fsize < 5 & full$Fsize > 1] <- 'small'

full$FsizeD[full$Fsize > 4] <- 'large'

# Show family size by survival using a mosaic plot

mosaicplot(table(full$FsizeD, full$Survived), main='Family Size by Survival', shade=TRUE)

c. Impute the missing values in Age variable using Mice Library, create two different graphs showing Age distribution before and after imputation.

3.2 Predictive imputation

Finally, as we noted earlier, there are quite a few missing Age values in our data. We are going to get a bit more fancy in imputing missing age values. Why? Because we can. We will create a model predicting ages based on other variables.

# Show number of missing Age values

sum(is.na(full$Age))

## [1] 263

We could definitely use rpart (recursive partitioning for regression) to predict missing ages, but I’m going to use the mice package for this task just for something different. You can read more about multiple imputation using chained equations in r here (PDF). Since we haven’t done it yet, I’ll first factorize the factor variables and then perform mice imputation.

# Make variables factors into factors

factor\_vars <- c('PassengerId','Pclass','Sex','Embarked',

'Title','Surname','Family','FsizeD')

full[factor\_vars] <- lapply(full[factor\_vars], function(x) as.factor(x))

# Set a random seed

set.seed(129)

# Perform mice imputation, excluding certain less-than-useful variables:

mice\_mod <- mice(full[, !names(full) %in% c('PassengerId','Name','Ticket','Cabin','Family','Surname','Survived')], method='rf')

##

## iter imp variable

## 1 1 Age Deck

## 1 2 Age Deck

## 1 3 Age Deck

## 1 4 Age Deck

## 1 5 Age Deck

## 2 1 Age Deck

## 2 2 Age Deck

## 2 3 Age Deck

## 2 4 Age Deck

## 2 5 Age Deck

## 3 1 Age Deck

## 3 2 Age Deck

## 3 3 Age Deck

## 3 4 Age Deck

## 3 5 Age Deck

## 4 1 Age Deck

## 4 2 Age Deck

## 4 3 Age Deck

## 4 4 Age Deck

## 4 5 Age Deck

## 5 1 Age Deck

## 5 2 Age Deck

## 5 3 Age Deck

## 5 4 Age Deck

## 5 5 Age Deck

# Save the complete output

mice\_output <- complete(mice\_mod)

Let’s compare the results we get with the original distribution of passenger ages to ensure that nothing has gone completely awry.

# Plot age distributions

par(mfrow=c(1,2))

hist(full$Age, freq=F, main='Age: Original Data',

col='darkgreen', ylim=c(0,0.04))

hist(mice\_output$Age, freq=F, main='Age: MICE Output',

col='lightgreen', ylim=c(0,0.04))