Abalone age prediciton

Rohith Kumar Sajja

March 6, 2020

This is an R Markdown Notebook to predict age of abalone.

The goal of this project is to predict the number of rings in the abalone and thus predict the age.

1. Data Description

```
# Installing and loading required packages
library(e1071)
library(ggplot2)
library(dplyr)
```

```
##
## Attaching package: 'dplyr'
```

```
## The following objects are masked from 'package:stats':
##
##     filter, lag
```

```
## The following objects are masked from 'package:base':
##
##     intersect, setdiff, setequal, union
```

```
library(knitr)
library(GGally)
```

```
## Registered S3 method overwritten by 'GGally':
##   method from
##   +.gg   ggplot2
```

```
##
## Attaching package: 'GGally'
```

```
## The following object is masked from 'package:dplyr':
##
##     nasa
```

```
library(AppliedPredictiveModeling)
library(caret)
```

```
## Loading required package: lattice
```

```r
# Loading the abalone dataset
abalone_0 <- read.csv('abalone.csv', na.strings = '?')

# Structure of the dataset
print(str(abalone_0))
```

```
## 'data.frame':    4177 obs. of  9 variables:
##  $ Sex           : Factor w/ 3 levels "F","I","M": 3 3 1 3 2 2 1 1 3 1 ...
##  $ Length        : num  0.455 0.35 0.53 0.44 0.33 0.425 0.53 0.545 0.475 0.55 ...
##  $ Diameter      : num  0.365 0.265 0.42 0.365 0.255 0.3 0.415 0.425 0.37 0.44 ...
##  $ Height        : num  0.095 0.09 0.135 0.125 0.08 0.095 0.15 0.125 0.125 0.15 ...
##  $ Wholeweight   : num  0.514 0.226 0.677 0.516 0.205 ...
##  $ Shuckedweight: num  0.2245 0.0995 0.2565 0.2155 0.0895 ...
##  $ Visceraweight: num  0.101 0.0485 0.1415 0.114 0.0395 ...
##  $ Shellweight   : num  0.15 0.07 0.21 0.155 0.055 0.12 0.33 0.26 0.165 0.32 ...
##  $ Rings         : int  15 7 9 10 7 8 20 16 9 19 ...
## NULL
```

```r
# Summary statistics
summary(abalone_0)
```

```
##  Sex           Length          Diameter          Height        Wholeweight
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000   Min.   :0.0020
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415
##  M:1528   Median :0.545   Median :0.4250   Median :0.1400   Median :0.7995
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395   Mean   :0.8287
##           3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650   3rd Qu.:1.1530
##           Max.   :0.815   Max.   :0.6500   Max.   :1.1300   Max.   :2.8255
##  Shuckedweight    Visceraweight     Shellweight         Rings
##  Min.   :0.0010   Min.   :0.0005   Min.   :0.0015   Min.   : 1.000
##  1st Qu.:0.1860   1st Qu.:0.0935   1st Qu.:0.1300   1st Qu.: 8.000
##  Median :0.3360   Median :0.1710   Median :0.2340   Median : 9.000
##  Mean   :0.3594   Mean   :0.1806   Mean   :0.2388   Mean   : 9.934
##  3rd Qu.:0.5020   3rd Qu.:0.2530   3rd Qu.:0.3290   3rd Qu.:11.000
##  Max.   :1.4880   Max.   :0.7600   Max.   :1.0050   Max.   :29.000
```

Encode the Rings into 3 age buckets in a new column Age.

1. Young - less than 6 rings (<7.5 years old)
2. Adult - from 6 to 13 rings (7.5 to 14.5 years old)
3. Old - more than 13 rings (>14.5 years old)

```r
# Transforming the dataset to include age bucket
abalone <- abalone_0 %>%
  mutate(Age=case_when(
    Rings %in% 1:5 ~ "Young",
    Rings %in% 6:13 ~ "Adult",
    Rings %in% 14:30 ~ "Old"
  ))
```

```r
# Converting Age into factor
abalone$Age <- as.factor(abalone$Age)

# Structure of modified data set
str(abalone)
```

```
## 'data.frame':    4177 obs. of  10 variables:
##  $ Sex          : Factor w/ 3 levels "F","I","M": 3 3 1 3 2 2 1 1 3 1 ...
##  $ Length       : num  0.455 0.35 0.53 0.44 0.33 0.425 0.53 0.545 0.475 0.55 ...
##  $ Diameter     : num  0.365 0.265 0.42 0.365 0.255 0.3 0.415 0.425 0.37 0.44 ...
##  $ Height       : num  0.095 0.09 0.135 0.125 0.08 0.095 0.15 0.125 0.125 0.15 ...
##  $ Wholeweight  : num  0.514 0.226 0.677 0.516 0.205 ...
##  $ Shuckedweight: num  0.2245 0.0995 0.2565 0.2155 0.0895 ...
##  $ Visceraweight: num  0.101 0.0485 0.1415 0.114 0.0395 ...
##  $ Shellweight  : num  0.15 0.07 0.21 0.155 0.055 0.12 0.33 0.26 0.165 0.32 ...
##  $ Rings        : int  15 7 9 10 7 8 20 16 9 19 ...
##  $ Age          : Factor w/ 3 levels "Adult","Old",..: 2 1 1 1 1 1 2 2 1 2 ...
```

```r
# Summary statistics of modified data set
summary(abalone)
```

```
##  Sex          Length         Diameter          Height        Wholeweight
##  F:1307   Min.   :0.075   Min.   :0.0550   Min.   :0.0000   Min.   :0.0020
##  I:1342   1st Qu.:0.450   1st Qu.:0.3500   1st Qu.:0.1150   1st Qu.:0.4415
##  M:1528   Median :0.545   Median :0.4250   Median :0.1400   Median :0.7995
##           Mean   :0.524   Mean   :0.4079   Mean   :0.1395   Mean   :0.8287
##           3rd Qu.:0.615   3rd Qu.:0.4800   3rd Qu.:0.1650   3rd Qu.:1.1530
##           Max.   :0.815   Max.   :0.6500   Max.   :1.1300   Max.   :2.8255
##  Shuckedweight    Visceraweight     Shellweight         Rings
##  Min.   :0.0010   Min.   :0.0005   Min.   :0.0015   Min.   : 1.000
##  1st Qu.:0.1860   1st Qu.:0.0935   1st Qu.:0.1300   1st Qu.: 8.000
##  Median :0.3360   Median :0.1710   Median :0.2340   Median : 9.000
##  Mean   :0.3594   Mean   :0.1806   Mean   :0.2388   Mean   : 9.934
##  3rd Qu.:0.5020   3rd Qu.:0.2530   3rd Qu.:0.3290   3rd Qu.:11.000
##  Max.   :1.4880   Max.   :0.7600   Max.   :1.0050   Max.   :29.000
##     Age
##  Adult:3498
##  Old  : 490
##  Young: 189
##
##
##
```

```r
kable(abalone[1:10,], digits = 4,format = 'markdown')
```

Sex	Length	Diameter	Height	Wholeweight	Shuckedweight	Visceraweight	Shellweight	Rings	Age
M	0.455	0.365	0.095	0.5140	0.2245	0.1010	0.150	15	Old
M	0.350	0.265	0.090	0.2255	0.0995	0.0485	0.070	7	Adult
F	0.530	0.420	0.135	0.6770	0.2565	0.1415	0.210	9	Adult
M	0.440	0.365	0.125	0.5160	0.2155	0.1140	0.155	10	Adult
I	0.330	0.255	0.080	0.2050	0.0895	0.0395	0.055	7	Adult

Sex	Length	Diameter	Height	Wholeweight	Shuckedweight	Visceraweight	Shellweight	Rings	Age
I	0.425	0.300	0.095	0.3515	0.1410	0.0775	0.120	8	Adult
F	0.530	0.415	0.150	0.7775	0.2370	0.1415	0.330	20	Old
F	0.545	0.425	0.125	0.7680	0.2940	0.1495	0.260	16	Old
M	0.475	0.370	0.125	0.5095	0.2165	0.1125	0.165	9	Adult
F	0.550	0.440	0.150	0.8945	0.3145	0.1510	0.320	19	Old

```r
# Number of missing values
nrow(abalone[!complete.cases(abalone),])
```

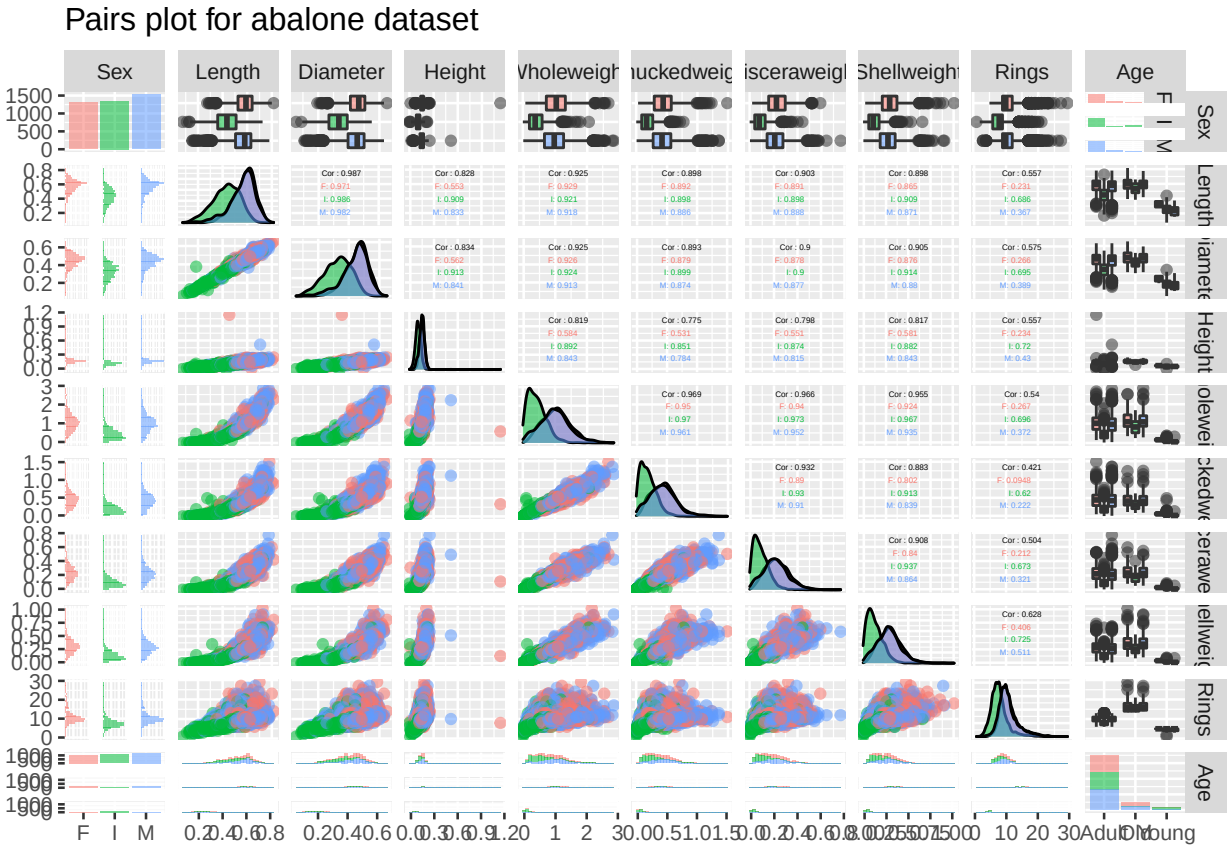
```
## [1] 0
```

2. Exploratory Data Analysis

2.1 Pair-wise correlation

```r
ggpairs(abalone, aes(colour = Sex, alpha = 0.1), title="Pairs plot for abalone dataset", upper = list(co
  theme_grey(base_size = 10)
```

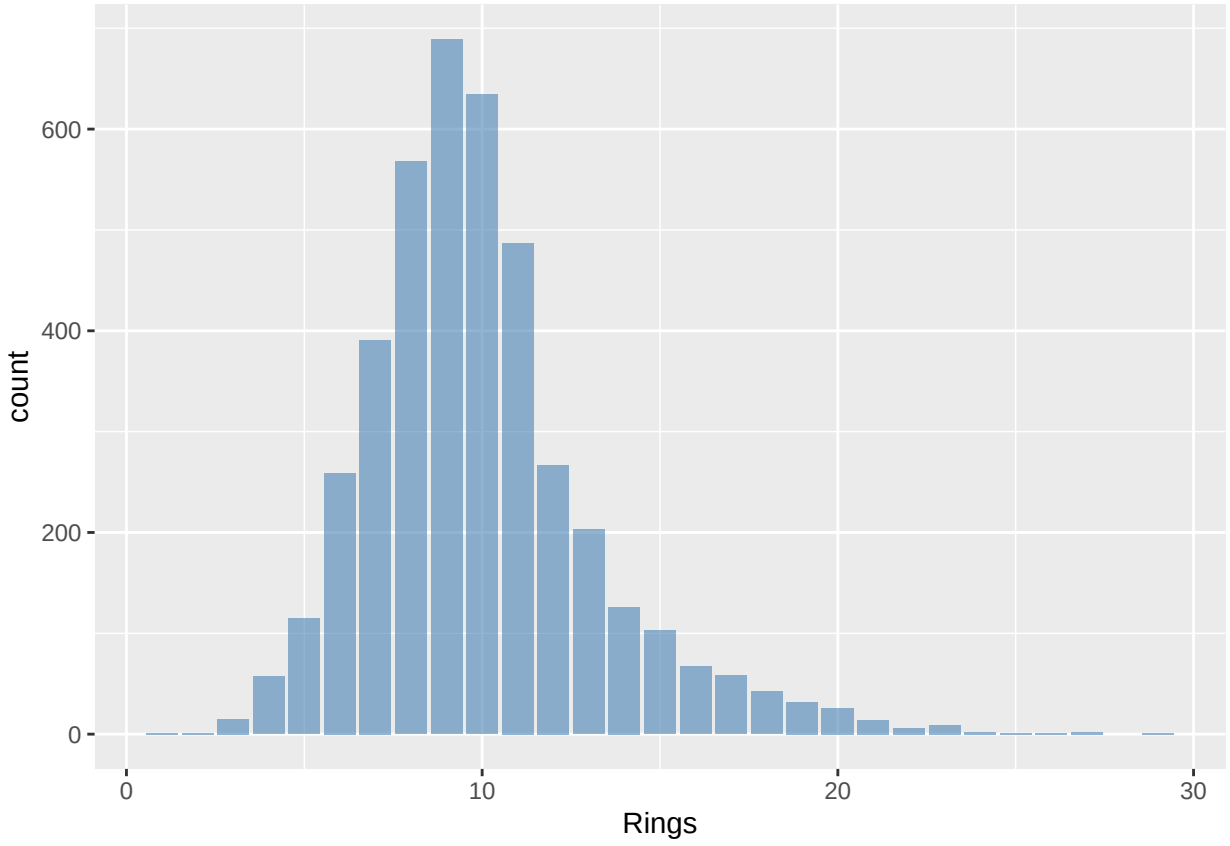
```
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
## `stat_bin()` using `bins = 30`. Pick better value with `binwidth`.
```



We can note that there is high correlation between a few measurements, such as Diameter and Length

2.2 Rings

```
# Plotting the frequency of abalone over number of rings
ggplot(abalone, aes(x=Rings))+geom_bar(fill="steelblue",alpha=0.6)
```

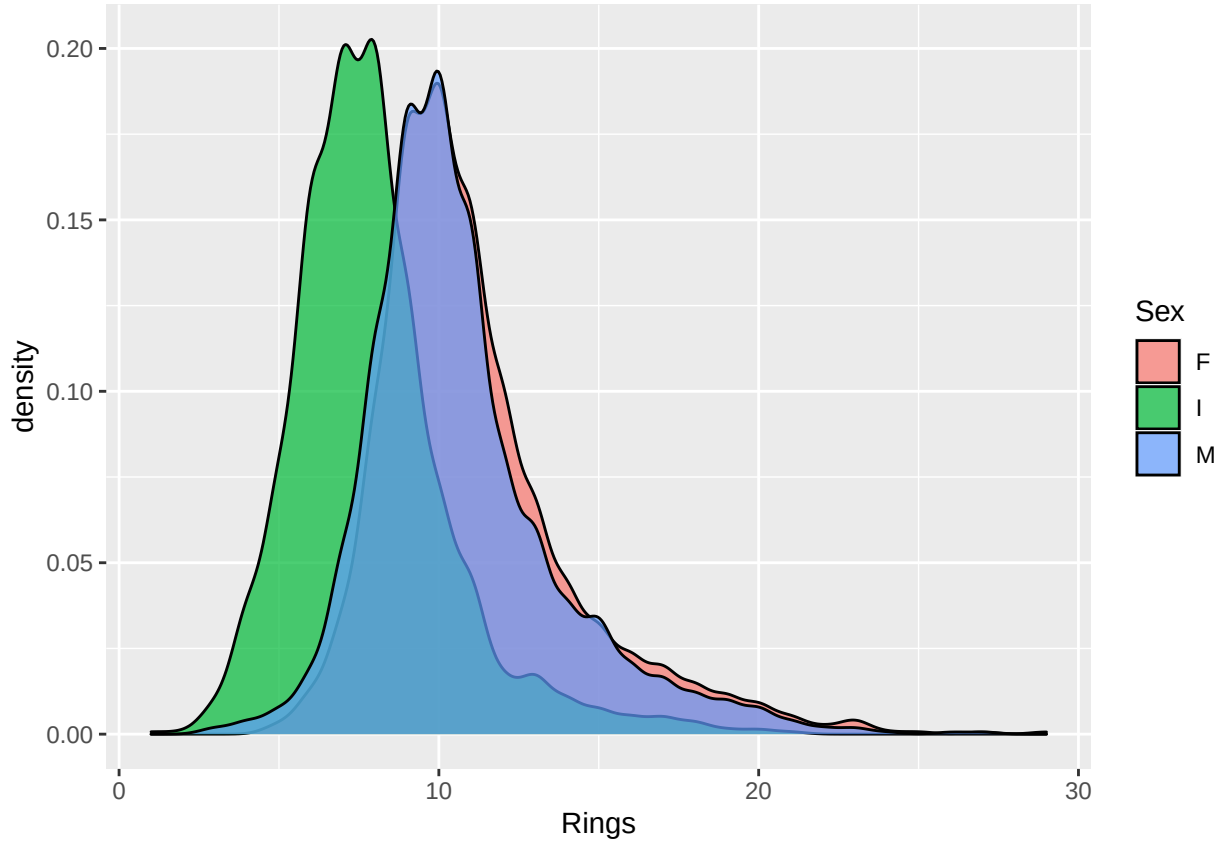


2.3 Sex

```
# Summary statistics
summary(abalone$Wholeweight)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0020  0.4415  0.7995  0.8287  1.1530  2.8255
```

```
# Density plot of Rings over Sex
ggplot(abalone) + aes(Rings, fill = Sex) +  geom_density(alpha = 0.7)
```



As we can see Females have higher number of rings and thus we could say that they live longer.

2.4 WholeWeight

```
# Summary statistics
summary(abalone$Wholeweight)
```

```
##    Min. 1st Qu.  Median    Mean 3rd Qu.    Max.
##  0.0020  0.4415  0.7995  0.8287  1.1530  2.8255
```

3. Model Selection

3.1 Testing and Evaluation metrics

```
# Algorithms using 10-fold cross validation
control <- trainControl(method="repeatedcv", number=10, repeats=3)
metric <- "RMSE"
seed <- 888
```

3.2 Model Generation (non-ensemble)

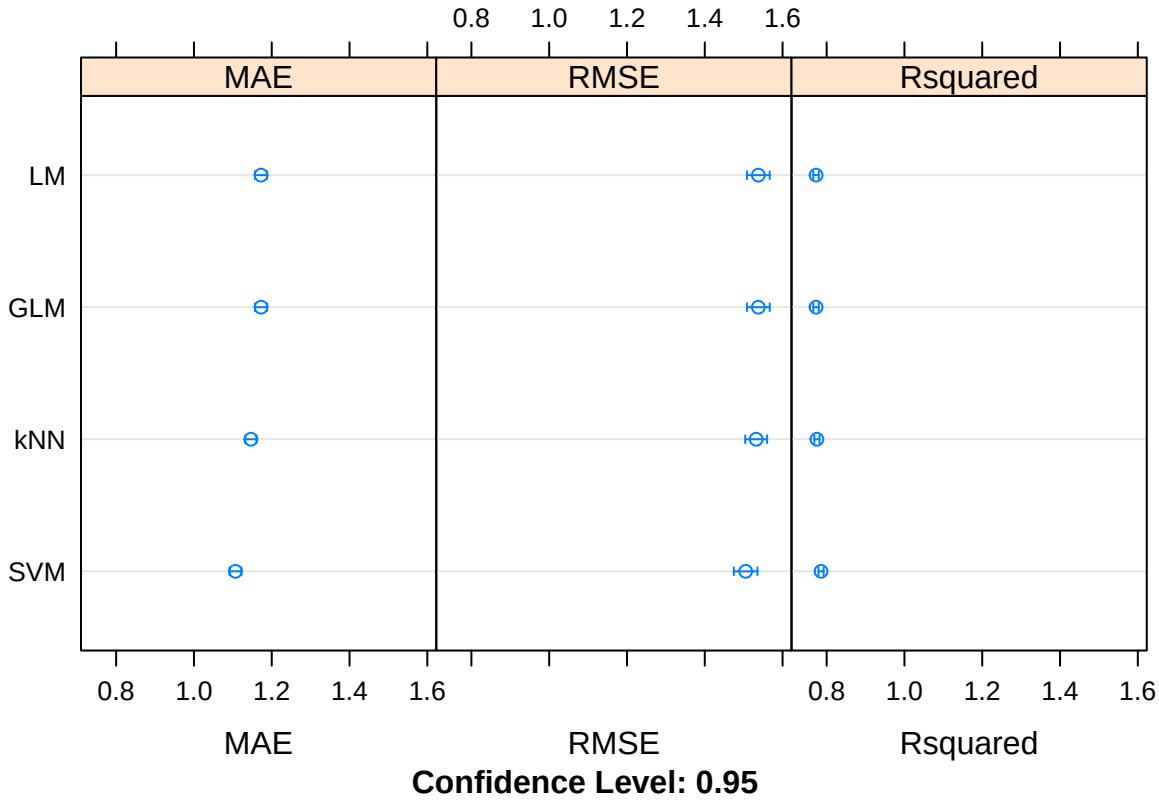
```
# GLM
set.seed(seed)
fit.glm <- train(Rings~., data=abalone, method="glm", metric=metric, trControl=control)
# LM
set.seed(seed)
fit.lm <- train(Rings~., data=abalone, method="lm", metric=metric, trControl=control)
# SVM
set.seed(seed)
fit.svm <- train(Rings~., data=abalone, method="svmRadial", metric=metric, trControl=control)
# kNN
set.seed(seed)
fit.knn <- train(Rings~., data=abalone, method="knn", metric=metric, trControl=control)
```

3.3 Algorithm Comparision

```
results <- resamples(list(SVM=fit.svm, kNN=fit.knn, GLM=fit.glm, LM=fit.lm))
summary(results)
```

```
##
## Call:
## summary.resamples(object = results)
##
## Models: SVM, kNN, GLM, LM
## Number of resamples: 30
##
## MAE
##          Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## SVM 0.9982476 1.087898 1.112499 1.106694 1.123501 1.190192    0
## kNN 1.0529506 1.127317 1.149675 1.146312 1.168410 1.233813    0
## GLM 1.0649650 1.151151 1.174471 1.172514 1.198601 1.245973    0
## LM  1.0649650 1.151151 1.174471 1.172514 1.198601 1.245973    0
##
## RMSE
##         Min.  1st Qu.   Median     Mean  3rd Qu.     Max. NA's
## SVM 1.308016 1.462884 1.492238 1.505026 1.531511 1.680163    0
## kNN 1.364875 1.485608 1.517420 1.532124 1.567766 1.703616    0
## GLM 1.358677 1.495171 1.527801 1.537665 1.572826 1.755688    0
## LM  1.358677 1.495171 1.527801 1.537665 1.572826 1.755688    0
##
## Rsquared
##          Min.   1st Qu.    Median      Mean   3rd Qu.      Max. NA's
## SVM 0.7400976 0.7765037 0.7869061 0.7855723 0.7973586 0.8204106    0
## kNN 0.7345130 0.7679448 0.7761924 0.7750216 0.7848360 0.8042621    0
## GLM 0.7418943 0.7571821 0.7717762 0.7727818 0.7871354 0.8082604    0
## LM  0.7418943 0.7571821 0.7717762 0.7727818 0.7871354 0.8082604    0
```

```
dotplot(results)
```



4. Results Validation

```
index <- best(fit.svm$results, metric, maximize=TRUE)
config <- fit.svm$results[index,]
print(config)
```

```
##       sigma    C     RMSE  Rsquared      MAE   RMSESD RsquaredSD      MAESD
## 1 0.1687842 0.25 1.521509 0.7840293 1.111526 0.090097 0.01859969 0.04325278
```