Term work 1

```
#load BostonHousing Data
library(mlbench)
install.packages("dplyr")
library(dplyr)
library(ggplot2)
library(reshape2)
data("BostonHousing")
housing <- BostonHousing
str(housing)
#ggplot
housing %>%
  ggplot(aes(x = medv)) +
  stat_density() +
  labs(x = "Median Value ($1000s)", y = "Density", Ptle = "Density Plot of
Median Value House Price in Boston") +
  theme_minimal()
#summary
summary(housing$medv)
#predicted V/S original
housing %>%
  select(c(crim, rm, age, rad, tax, lstat, medv)) %>%
  melt( id.vars = "medv") %>%
  ggplot(aes(x = value, y = medv, colour = variable)) +
  geom_point(alpha = 0.7) +
  stat_smooth(aes(colour = "black")) +
  facet_wrap(~variable, scales = "free", ncol = 2) +
  labs(x = "Variable Value", y = "Median House Price ($1000s)") +
  theme_minimal()
library("caret")
set.seed(123) #random number genearPon
to_train <- createDataPartition(y = housing$medv, p = 0.75, list = FALSE)
to_test<-createDataPartition(y=housing$medv, p=0.25,list=FALSE)
train <- housing[to_train, ]
test <- housing[to_test, ]
first_lm <- lm( medv ~ crim +rm +tax +lstat, data = train)
lm1_rsqu <- summary(first_lm)$r.squared
print(paste("First linear model has an r-squared value of ", round(lm1_rsqu,
                                                    3), sep = ""))
#plot(first_lm)
second_lm <- lm(log(medv) ~ crim +rm + tax +lstat, data = train)
lm2_rsqu <- summary(second_lm)$r.squared
print(paste("Our second linear model has an r-squared value of ",
```

```
        round(lm2_rsqu, 3), sep = ""))
abs(mean(second_lm$residuals))
#Create a data frame of your predicted values and the original values
predicted <- predict(second_lm, newdata = test)
results <- data.frame(predicted = exp(predicted), original = test$medv)
#Plot this to visualize the performance of your model.
results %>%
  ggplot(aes(x = predicted, y = original)) +
  geom_point() +
  stat_smooth() +
  labs(x = "Predicted Values", y = "Original Values", Ptle = "Predicted vs.
Original Values") +
  theme_minimal()
```