## Demonstrating Model Validation on Credit Dataset

Libraries

#install.packages("caret")  
library(tidyverse)

## Warning: package 'tidyverse' was built under R version 3.5.2

## -- Attaching packages ---------------------------------------------------------------- tidyverse 1.2.1 --

## v ggplot2 3.1.0 v purrr 0.2.5  
## v tibble 1.4.2 v dplyr 0.7.7  
## v tidyr 0.8.2 v stringr 1.3.1  
## v readr 1.1.1 v forcats 0.3.0

## -- Conflicts ------------------------------------------------------------------- tidyverse\_conflicts() --  
## x dplyr::filter() masks stats::filter()  
## x dplyr::lag() masks stats::lag()

library(caret) #for splitting for validation

## Warning: package 'caret' was built under R version 3.5.2

## Loading required package: lattice

##   
## Attaching package: 'caret'

## The following object is masked from 'package:purrr':  
##   
## lift

Read-in dataset

credit = read\_csv("CreditData.csv")

## Parsed with column specification:  
## cols(  
## AnnualIncome = col\_double(),  
## HouseholdSize = col\_integer(),  
## YrsEdAfterHS = col\_integer(),  
## HrWkTV = col\_integer(),  
## AnnualCharges = col\_double()  
## )

Get rid of missing data rows

credit = credit %>% drop\_na() #delete any row with an NA value  
str(credit) #check structure after the drop

## Classes 'tbl\_df', 'tbl' and 'data.frame': 5000 obs. of 5 variables:  
## $ AnnualIncome : num 21.8 65.5 54.2 73.7 110.4 ...  
## $ HouseholdSize: int 4 7 3 6 7 8 5 8 1 3 ...  
## $ YrsEdAfterHS : int 5 3 2 0 5 3 4 5 4 1 ...  
## $ HrWkTV : int 29 46 18 44 39 39 40 27 15 3 ...  
## $ AnnualCharges: num 10024 11249 6115 9786 12634 ...

We will do pure k-fold cross-validation with no training and testing set. Can easily combine with train/test split. If you do this, do k-fold on training set.

Model with best single variable (by correlation).

ctrl = trainControl(method = "cv",number = 10) #set up caret 10 fold cross validation  
  
set.seed(123) #set random number seed for cross validation  
modCV = train(AnnualCharges ~ AnnualIncome, credit, method = "lm", trControl = ctrl, metric="Rsquared")  
summary(modCV)

##   
## Call:  
## lm(formula = .outcome ~ ., data = dat)  
##   
## Residuals:  
## Min 1Q Median 3Q Max   
## -12284.4 -3938.1 14.4 3947.9 13232.5   
##   
## Coefficients:  
## Estimate Std. Error t value Pr(>|t|)   
## (Intercept) 3146.361 185.193 16.99 <2e-16 \*\*\*  
## AnnualIncome 121.355 2.529 47.98 <2e-16 \*\*\*  
## ---  
## Signif. codes: 0 '\*\*\*' 0.001 '\*\*' 0.01 '\*' 0.05 '.' 0.1 ' ' 1  
##   
## Residual standard error: 5027 on 4998 degrees of freedom  
## Multiple R-squared: 0.3153, Adjusted R-squared: 0.3152   
## F-statistic: 2302 on 1 and 4998 DF, p-value: < 2.2e-16

That’s it. We are done. If we had done a train/test split, we would then evaluate the quality of the model on the testing set.