Introduction to R programming for data science

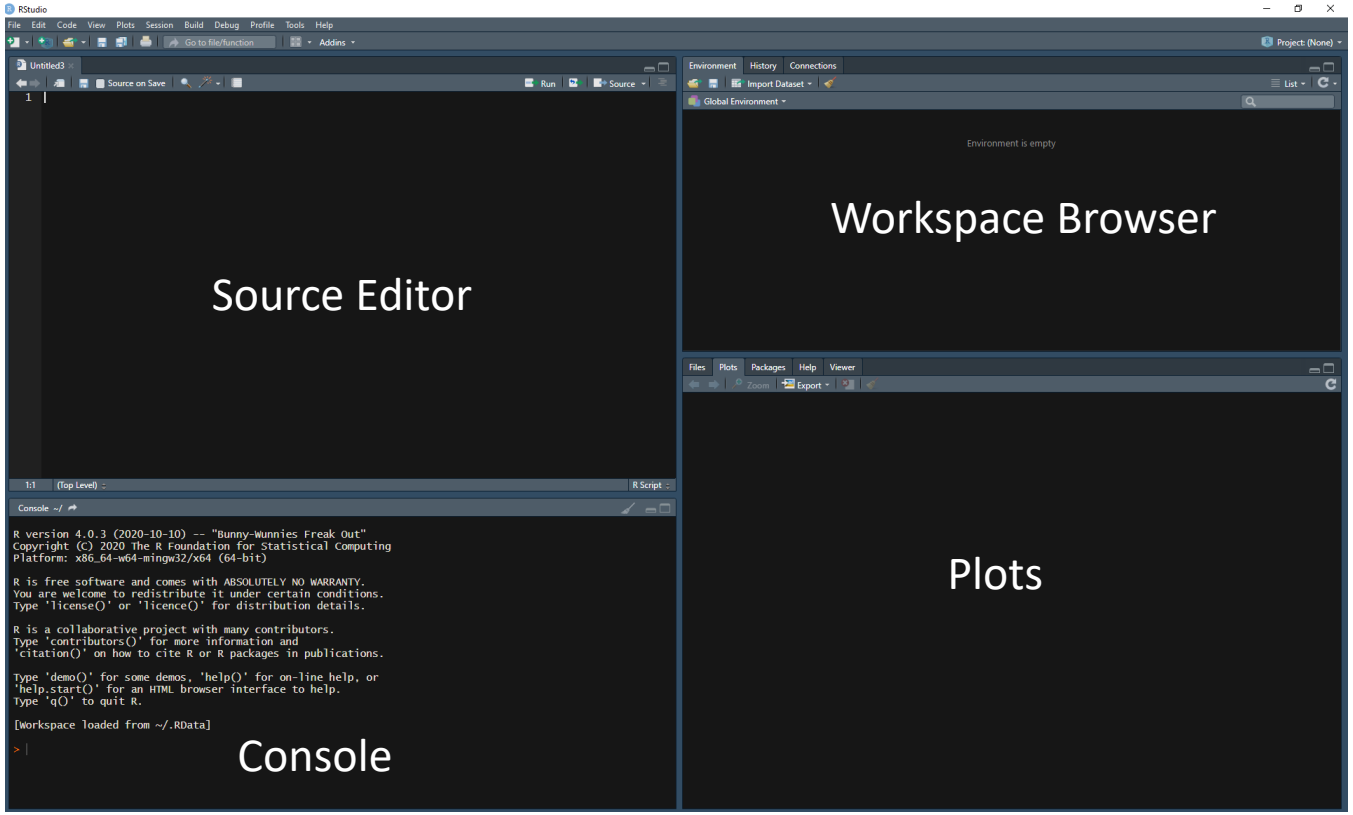
KIMIA VAHDAT (KVAHDAT@NCSU.EDU)

EDWARD P. FITTS DEPARTMENT OF
INDUSTRIAL AND SYSTEMS ENGINEERING

Questions we cover today:

- How to read data into R?
- How to preprocess and clean the data in R?
- How to build linear regression models in R?
- Making predictions and interpret the results.
- How to build a classification model in R?
- How to make sure my model is good enough?

How does RStudio look like?



How to read data into R?

.csv	• read.csv(filename, header=T, sep=“,”)
.data or .txt	• read.table(filename, header=F, sep=“”) • sep=“\t”: tab delimited • sep=“ “: space delimited
.RData	• load(filename)
.xlsx or .xls	• Need package “readxl” • read_excel(filename, sheet=“sheetname”)
Any BIG data	• Need package “data.table” • fread(file=“filename”)

How to preprocess and clean the data in R?

- This is the most important and time consuming part of any data science project!
- You need a clean and tidy dataset to perform all machine learning models you want on it.
- We use three packages, “tidyverse”, “dplyr”, and “caret”, for the preprocessing and two packages, “ggplot2” and “GGally” for plotting the explanatory data analysis reports.

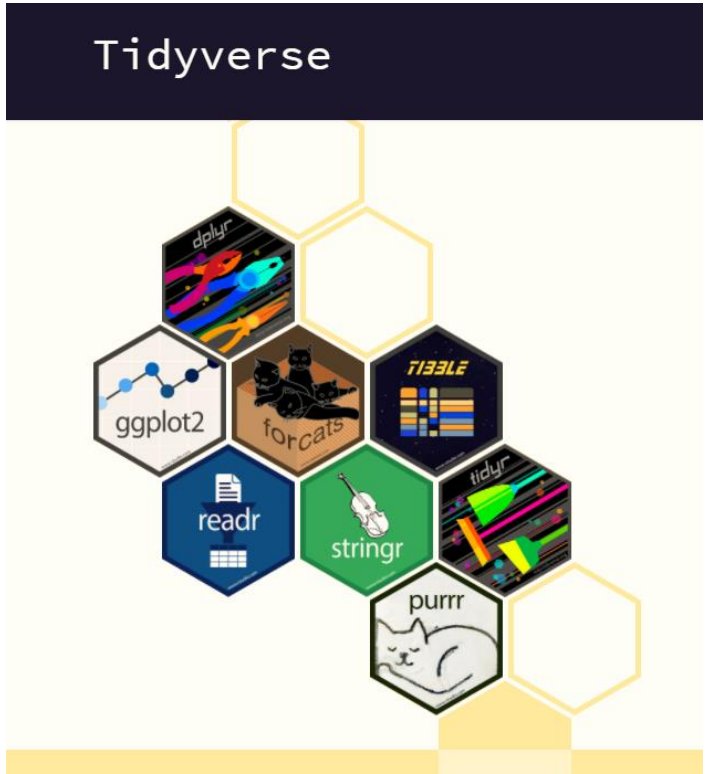
```
## Loading and installing packages
packages <- c("caret", "dplyr","tidyverse","ggplot2","GGally")
if (length(setdiff(packages, rownames(installed.packages()))) > 0) {
  install.packages(setdiff(packages, rownames(installed.packages())))
}

library(dplyr)
library(caret)
library(tidyverse)
library(ggplot2)
library(GGally)
```

First, let’s learn the syntax of tidyverse:

One of the great aspects of tidyverse is the operator %>%

This operator replaces parenthesis for functions and makes the code more readble.

- Example: log(sin(exp(10))) can be replaced with 10 %>% exp() %>% sin() %>% log()
- A list of functions used for cleaning and manipulating the data is:
  - filter(condition): filters the rows of a data based on the condition
  - select(column names): selects the column names from the data
  - group_by(variable_name): groups the data rows based on the selected variable
  - summarise(new_variable=function(old_variables)): summarizes the data using functions of existing variables; if used with group, then summarizes the data across groups
  - mutate(new_variable=function(old_variables)): adds new variables and preserves existing ones
  - transmute() adds new variables and drops existing ones.



Preprocessing checklist:

Missing values

- To check: anyNA()and is.na()
- Either remove all rows with missing values with na.omit(), or remove columns with most of their elements missing with:
  - data %>% select(!names)
- Or, replace the missing values with the average/median of the column using:
  - data %>% mutate(Var = replace(Var, is.na(Var), median(Var, na.rm = TRUE)))

Preprocessing checklist:

Convert categorical variables

- To check: glimpse() or summary()
- For variables with only two categories, like gender, convert one class to 0 and the other to 1.
- For variables with d categories, need to create d-1 new variables, each representing one of the categories using
  - dummyVars("~.", data, fullRank=TRUE)

Marital Status
Married
Single
Married
Divorced
Single

MS_Single	MS_Married	MS_Divorced
0	1	0
1	0	0
0	1	0
0	0	1
1	0	0

This is redundant!

Preprocessing checklist:

Near zero variance variables

- To check: nearZeroVar(data, freqCut=95/5, uniqueCut=10)
- Need to extremely careful with the parameters in this function, especially if variables are imbalanced.
- Another way is to compare the variance of the variables manually and count their unique values.
- Either remove those variables, if you have a lot of variables, or combine these variables into groups.

Preprocessing checklist:

Highly correlated variables

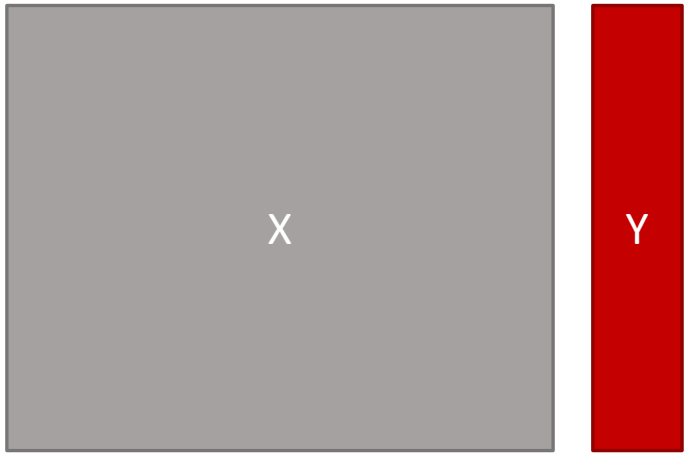
- To check: cor (data, method="pearson")
- If the data is very large, use corrplot package to visualize the correlations and decide based on that.
- A heatmap is the most common way to illustrate correlations.
- For two highly correlated variables, either remove one of them, or combine them.

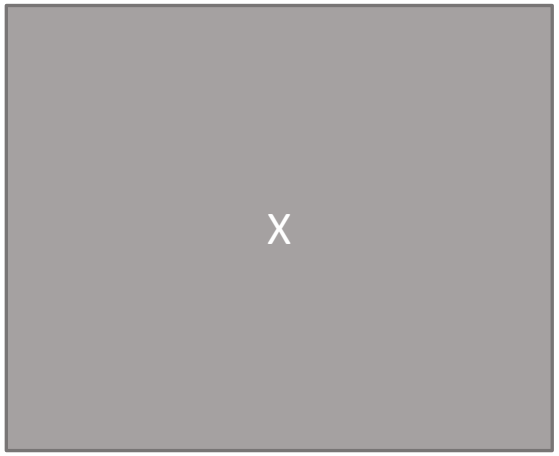
Preprocessing checklist:

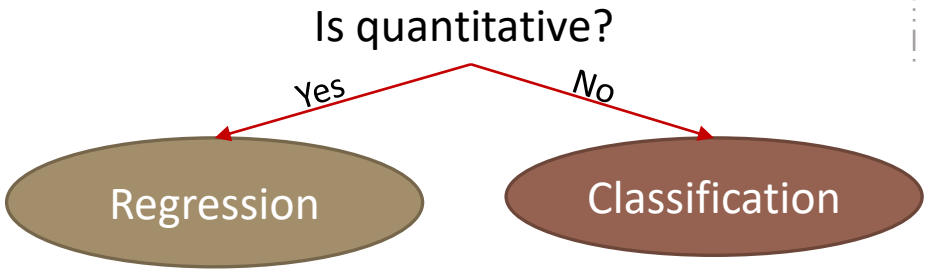
Variable selection

- If the number of variables in a data is large, it is better to identify the important variables and only build models on them.
- There are many methods for variable selection:
  - Backward/Forward selection, Criterion based selection (such as AIC and BIC) and many others.

Supervised vs. Unsupervised Learning







How to build linear regression models in R?

Linear Regression:

$$y = \beta_0 + X_1\beta_1 + X_2\beta_2 + \cdots + X_p\beta_p + \epsilon$$

lm(y~., data) or lm(y~X1+X2+X3,data)

Polynomial Regression:

$$y = \beta_0 + X_1\beta_1 + X_1^2\beta_2 + X_1X_2\beta_3 + \cdots + \epsilon$$

lm(y~X1+(X1-mean(X1))^2+(X2-mean(X2))(X1-mean(X1),data)

Note: when adding higher degree variables to a linear model, normalize them to avoid variance inflation!

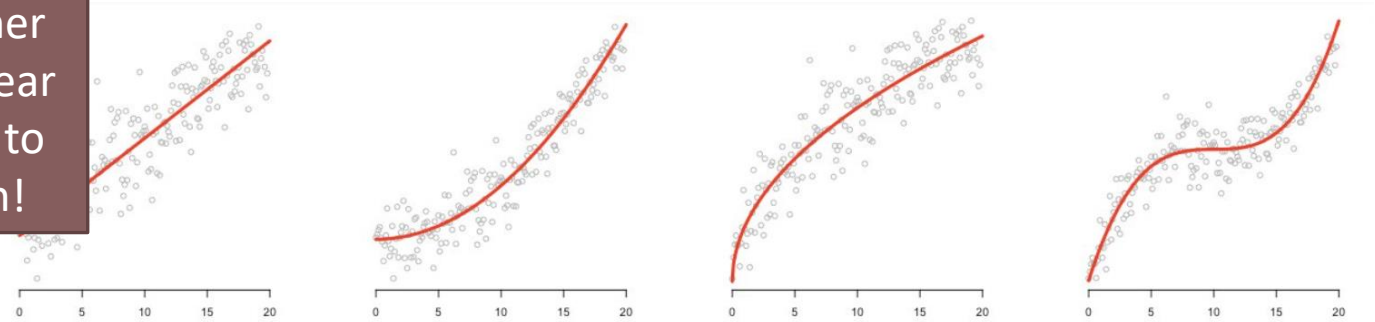


Photo source: https://twitter.com/DrHughHarvey/status/1011257076147019776/photo/1

Always keep in mind these assumptions:

Linear Relationship

- X and Y have a linear relationship.
- To check: the scatter plot

Independence

- The data points or residuals are independent.
- To check: plot residuals vs. fitted values

Homoscedasticity

- The residuals have constant variance.
- To check: plot residuals vs. fitted values

Normality

- The residuals follow a Normal distribution.
- To check: plot the QQ-Plot

Making predictions and interpret the results.

```
Call:
lm(formula = V1 ~ ., data = car)

Residuals:
    Min      1Q  Median      3Q     Max 
-9.5903 -2.1565 -0.1169  1.8690 13.0604 

Coefficients:
              Estimate Std. Error t value Pr(>|t|)    
(Intercept) -17.218435   4.644294  -3.707  0.00024 ***
V2           -0.493376   0.323282  -1.526  0.12780    
V3            0.019896   0.007515   2.647  0.00844 ** 
V4           -0.016951   0.013787  -1.230  0.21963    
V5           -0.006474   0.000652  -9.929  < 2e-16 ***
V6            0.080576   0.098845   0.815  0.41548    
V7            0.750773   0.050973  14.729  < 2e-16 ***
V8            1.426141   0.278136   5.127 4.67e-07 ***
---
Signif. codes:  0 ‘***’ 0.001 ‘**’ 0.01 ‘*’ 0.05 ‘.’ 0.1 ‘ ’ 1

Residual standard error: 3.328 on 384 degrees of freedom
Multiple R-squared:  0.8215,	Adjusted R-squared:  0.8182 
F-statistic: 252.4 on 7 and 384 DF,  p-value: < 2.2e-16
```

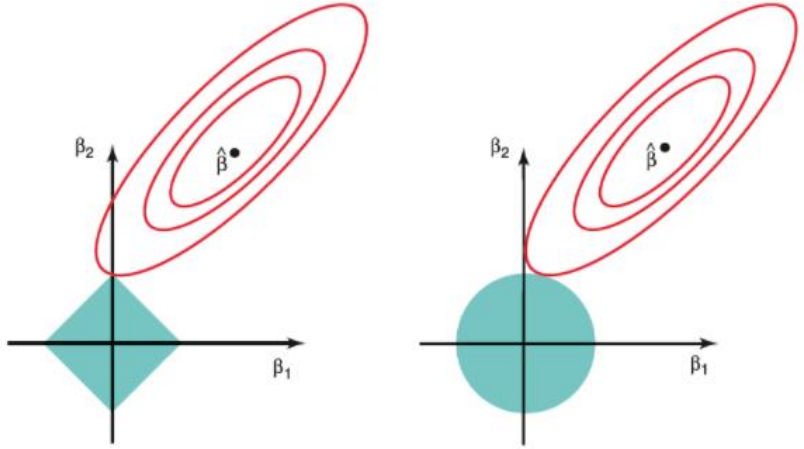
LASSO Regression

A regularization method, where it adds a penalty term to the objective function of LR.

$$Loss = \left|\left|y - \hat{y}\right|\right|_2^2 + \lambda\left|\left|\beta\right|\right|_1$$

In R, use “glmnet” package to build a LASSO regression model.

You need to fit the hyperparameter lambda!



How to build classification models in R?

Generalized linear models (glm):

$$g(y) = \beta_0 + X_1\beta_1 + X_2\beta_2 + \cdots + X_p\beta_p + \epsilon$$

$g(y)$ is the link function. For binary response we use the logit link function which is:

$g(y) = log\frac{P(y=1)}{1-P(y=1)}$

You can use the code: glm(y~., data, family="binomial") to build a model on the data.

And predict with predict(model, type="response")→ This gives the probabilities

How to evaluate a classification model?

AUC: Area under the curve

- Computes the area under the ROC curve, closer to 1 the better

Sensitivity or true positive rate

- What proportion of the positives were correctly selected

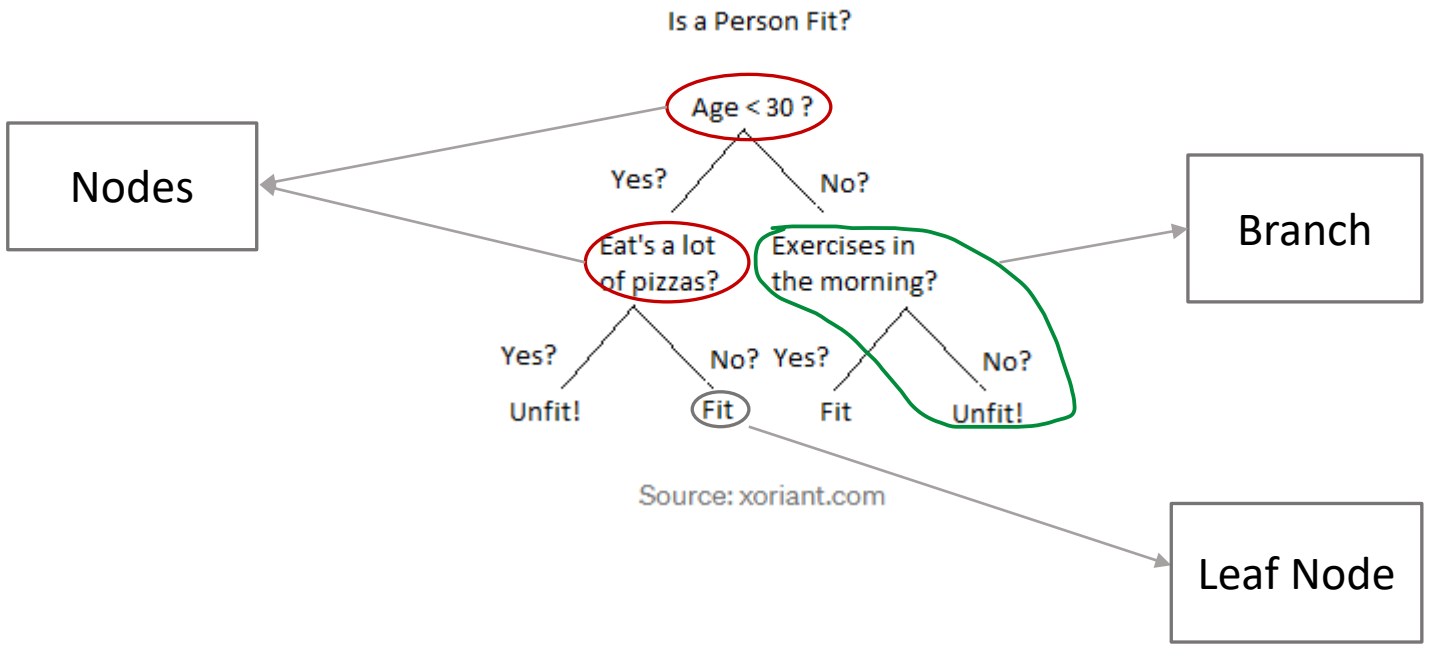
Specificity or true negative rate

- What proportion of the negatives were correctly selected

Accuracy

- What proportion of all predictions were correctly identified
- Might mislead in imbalanced datasets

Classification Trees

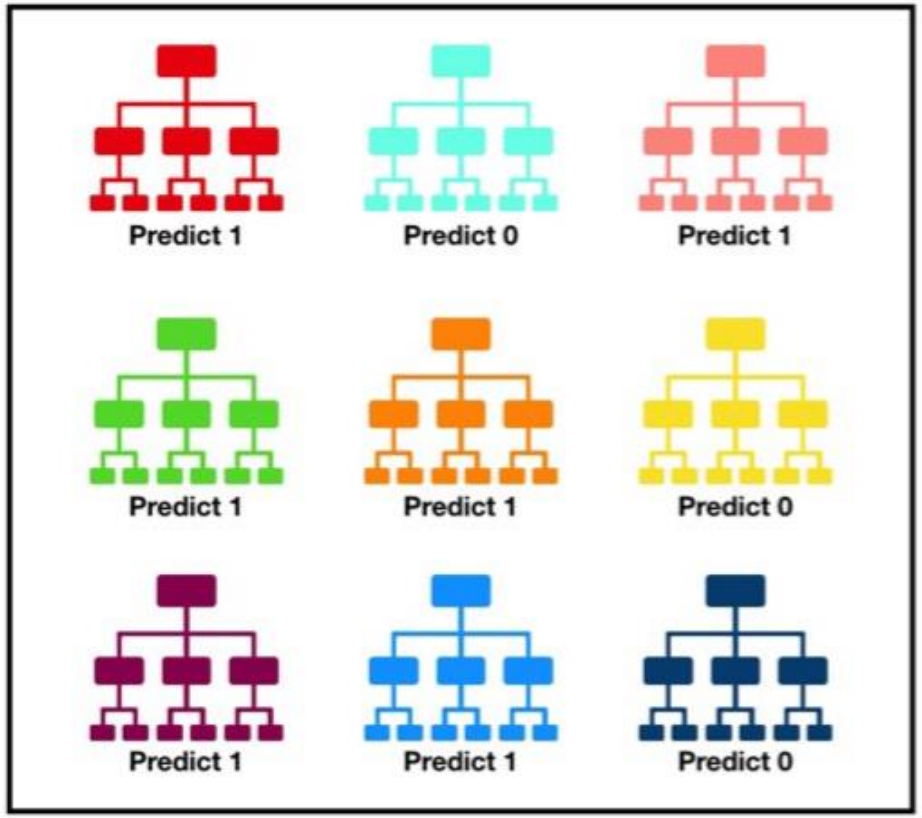


Random Forest

Mtry: number of variables to select each time

Ntree: number of trees to grow

Maxdepth: how much deep the trees should be?



What is overfitting?

Overfitting means your predicted values or $\hat{y}$ are very dependent on the small sample of the data you trained the model on.

An overfitted model does not generalize to other samples of the data!

Also referred to as bias-variance tradeoff.

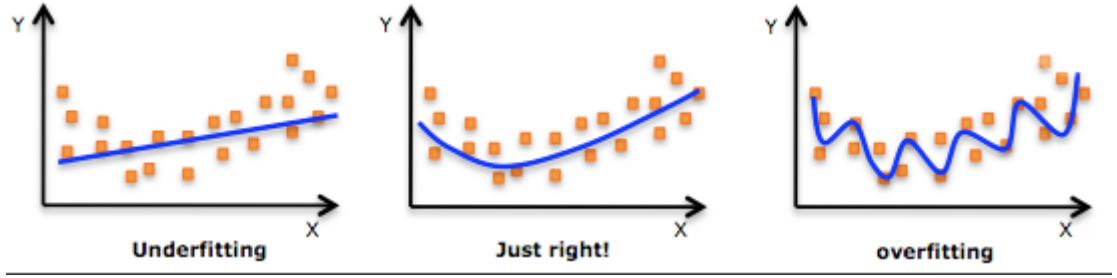


Photo source: https://www.quora.com/What-are-the-key-trade-offs-between-overfitting-and-underfitting

Our goal is to predict the response for the future data without overfitting the data on hand.

How to make sure my model is good enough?

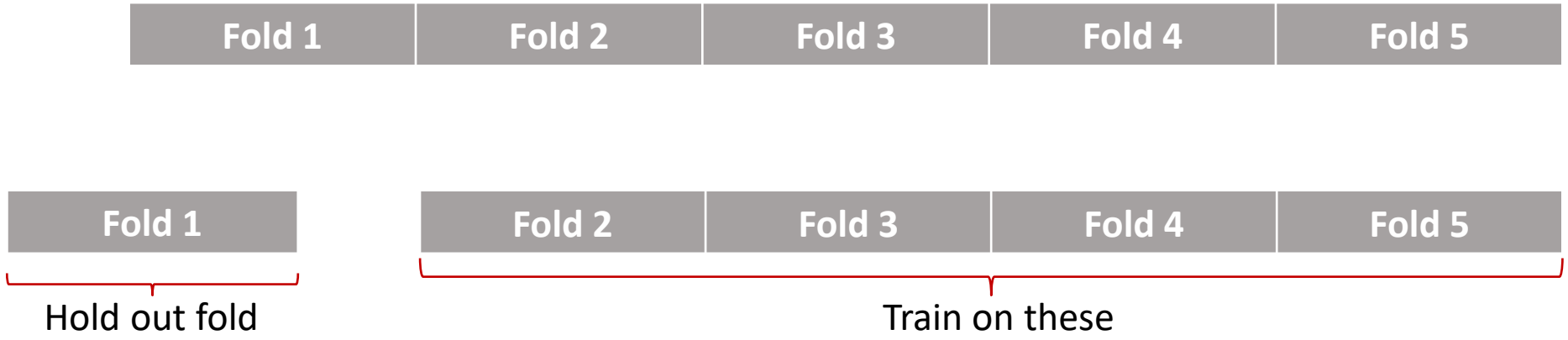
Criterion based

- Compare different models using AIC, BIC or adjusted R-squared

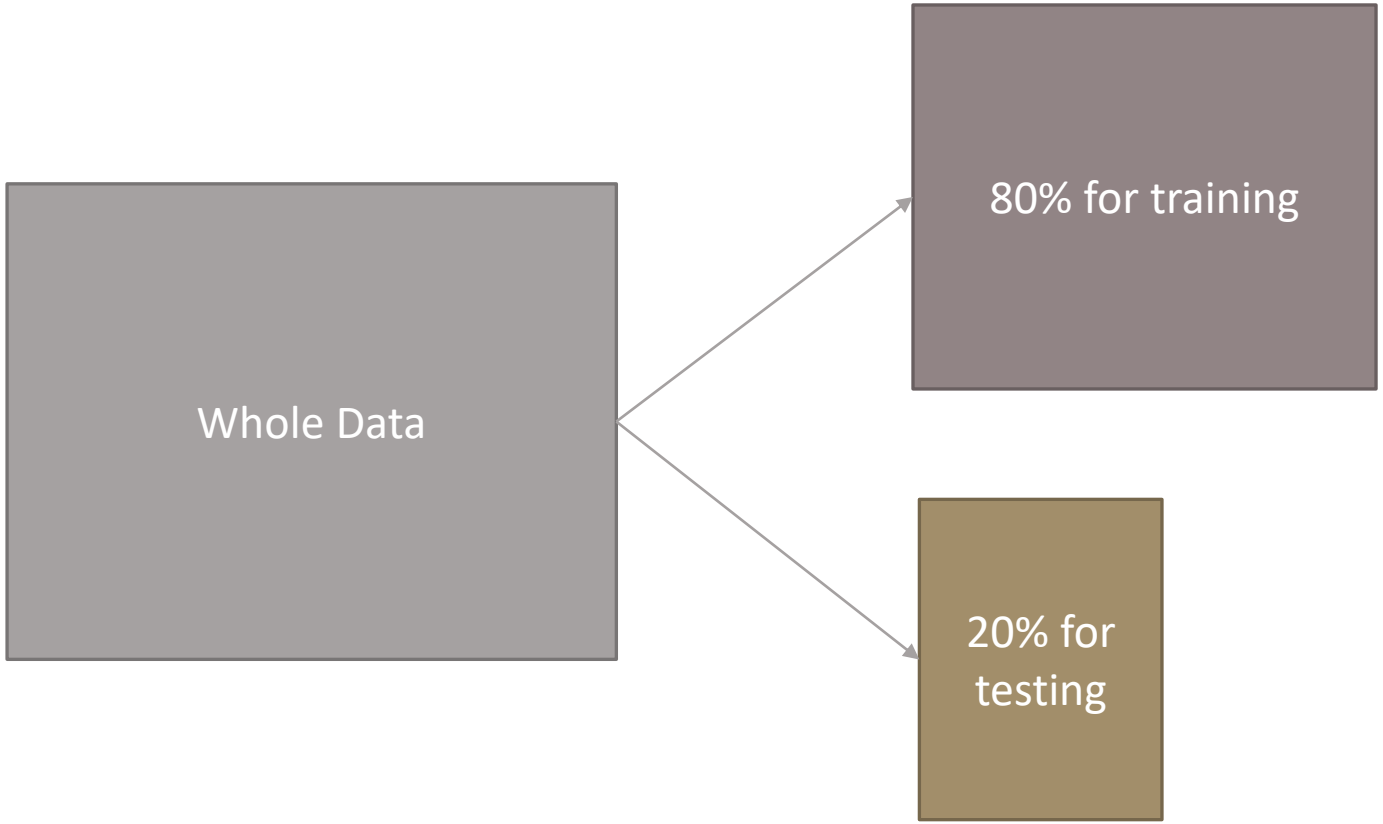
Sampling based

- K fold cross validation
- Leave one out cross validation
- Sample splitting
- Bootstrapping

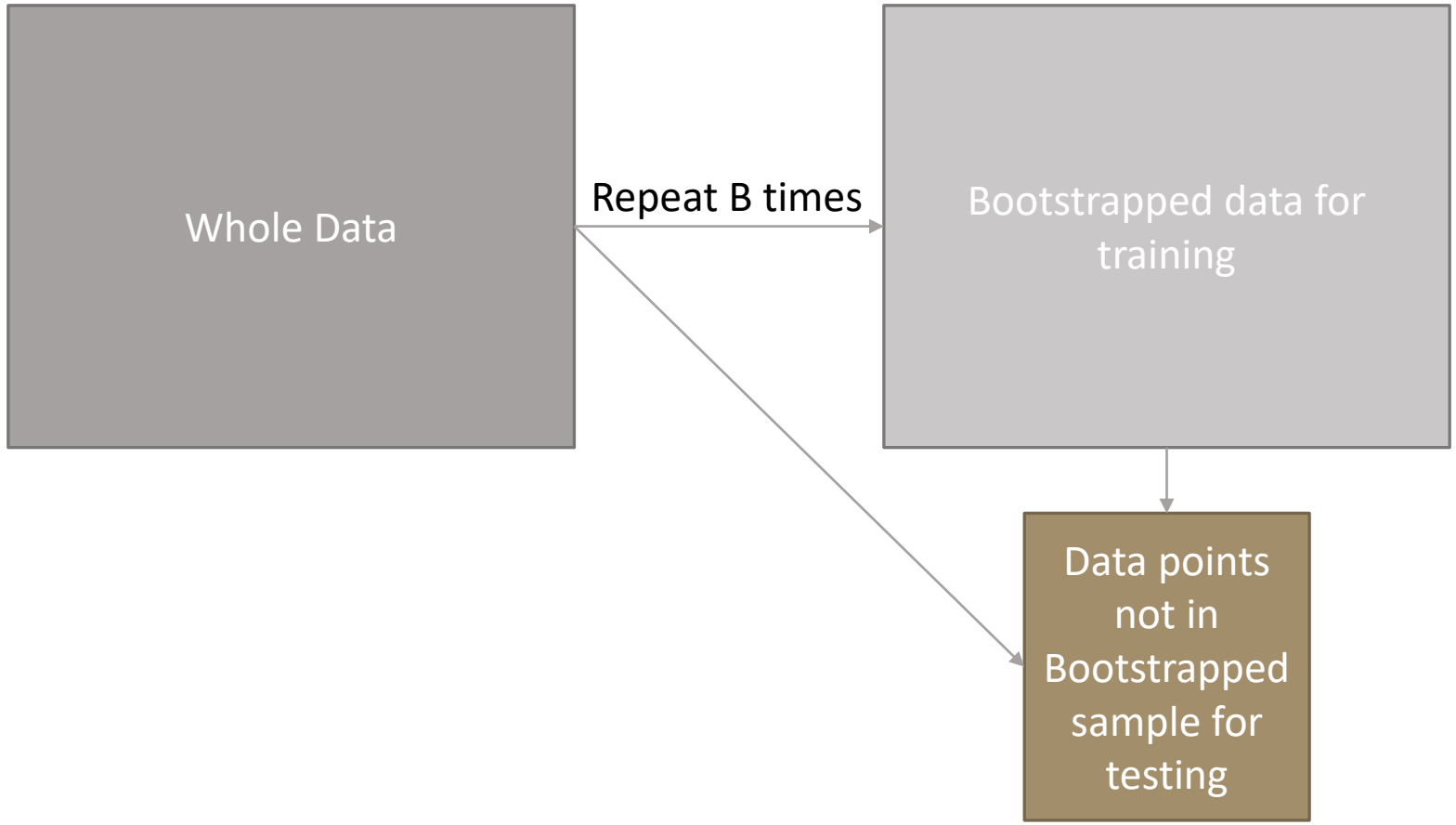
K-fold cross validation



Sample splitting



Bootstrapping



Let’s practice coding now!

Any Questions?

FEEL FREE TO CONTACT ME AT KVAHDAT@NCSU.EDU