MSiA 400 Lab Data Cleansing

Oct 13, 2014

Young Woong Park

Missing Data

- Missing data: arises in almost every study
- How do we deal with missing data?
  - Delete observations with missing field
  - Disregard observations with missing field when analyzing
  - Impute (fill) missing data
- Symbol for missing data in R: NA
  - Note: all capital! Not “na”, “Na”,…
- Checking missing data in R

```
> x = c(1,2,NA,4,5,NA);
> is.na(x);
```

FALSE FALSE TRUE FALSE FALSE TRUE

Missing Data Handling in R

- Simple way: Delete or disregard observations with any missing attribute
- Import heightmissing.txt

```
> height = read.table("../heightmissing.txt", header=T);
> summary(height);
```

- Can we calculate mean of male height?

```
> mean(height$Male)
```

- Disregarding missing observations

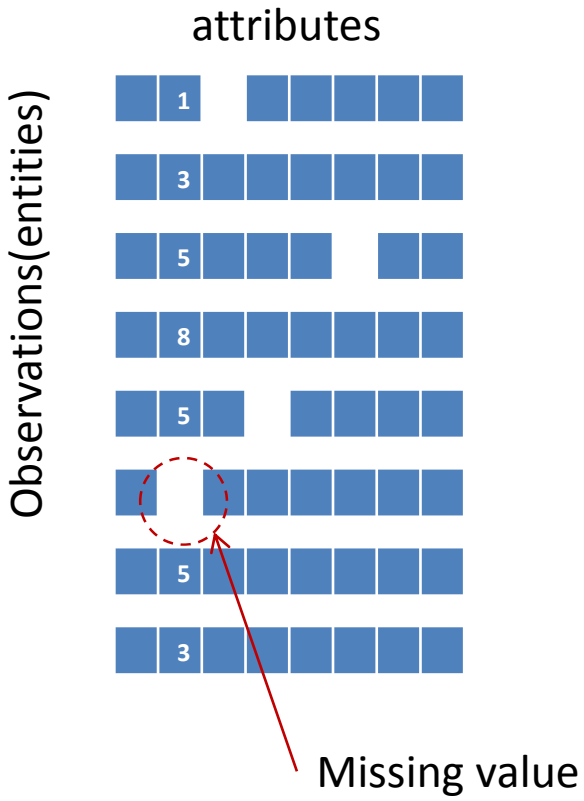
```
> mean(height$Male, na.rm = T)
```

- Removing missing observations

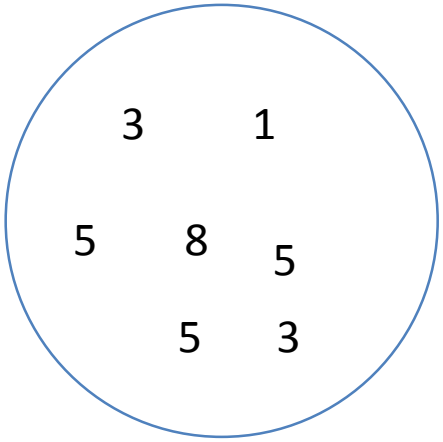
```
> height.omit = na.omit(height)
```

Imputing Missing Data: Random sampling

- Random imputation
  - Impute missing values by random sampling from the data



Q: What is the probability to pick 5?



Sample Space

Imputing Missing Data: Random sampling

- Define a function for random imputation in R

```
> random.imp <- function (a){
    missing <- is.na(a)              ## T/F matrix
    n.missing <- sum(missing)        ## number of observations with missing values
    a.obs <- a[!missing]
    imputed <- a
    imputed[missing] <- sample (a.obs, n.missing, replace=TRUE)
    return (imputed)
  }
```

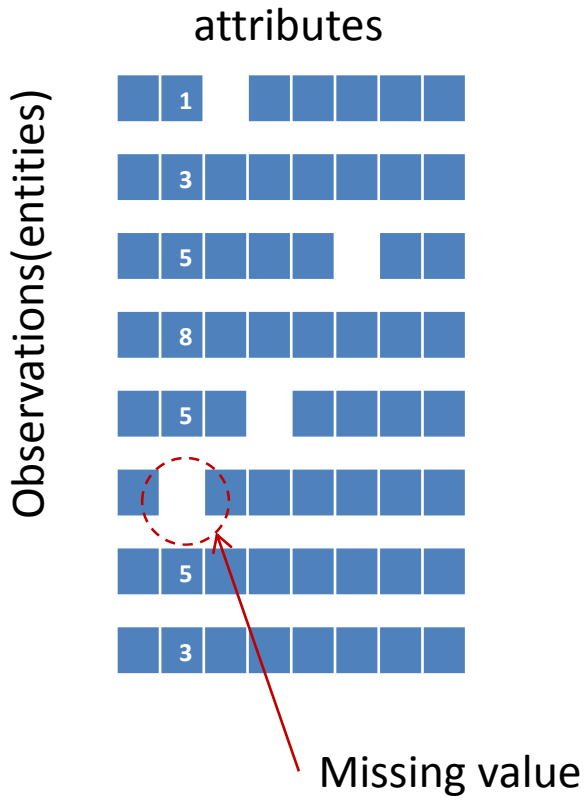
Ref: www.stat.columbia.edu/~gelman/arm/missing.pdf

- Use the function!

```
> height.rndimp = random.imp(height)
```

Imputing Missing Data: Most Common

- Most Common Value for categorical attributes: Fill with mode



Q: What is the mode for the attribute?

Imputing Missing Data: Most Common

- Finding the mode

```
> x = c(1,1,NA,3,4,4,5,5,5,5,6,NA)                    ## what is the mode?
> Mode <- function(x) {
   ux <- unique(x)                                    ## list the unique values (no duplicate)
   ux[which.max(tabulate(match(x, ux)))]
}
> mode.x = Mode(x);
```

- Define a function for most common value imputation

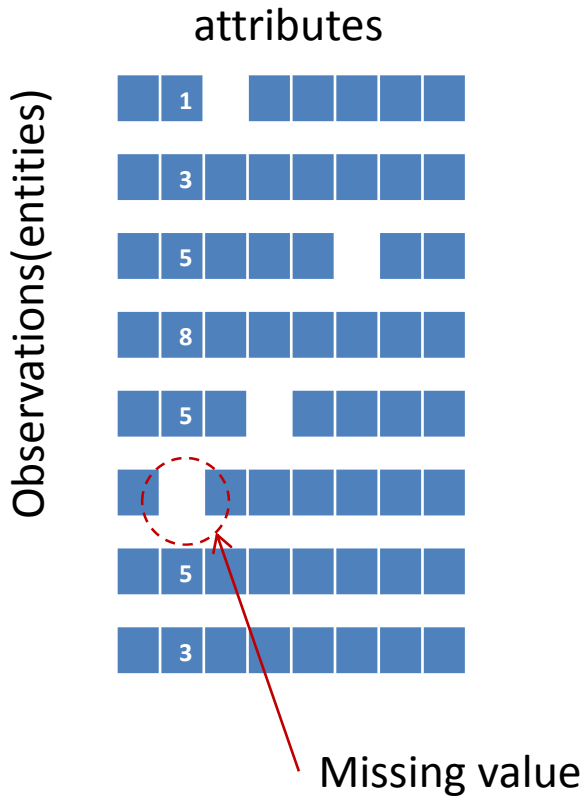
```
> mcv.imp <- function (a, modevalue){
      missing <- is.na(a)
      n.missing <- sum(missing)
      a.obs <- a[!missing]
      imputed <- a
      imputed[missing] <- modevalue
     return (imputed)
  }
```

- Use the function!

```
> x.mcv = mcv.imp(x,mode.x);
```

Imputing Missing Data: Average Value

- Average Value for numerical attributes: Fill with the average



Q: What is the average for the attribute?

Imputing Missing Data: Average Value

- Define a function for average value imputation in R

```
> avg.imp <- function (a, avg){
      missing <- is.na(a)
      n.missing <- sum(missing)
      a.obs <- a[!missing]
      imputed <- a
      imputed[missing] <- avg
     return (imputed)
  }
```

- Use the function!

```
> mavg = mean(na.omit(height$Male));
> favg = mean(na.omit(height$Female));
> mheight.avgimp = avg.imp(height$Male,mavg);
> fheight.avgimp = avg.imp(height$Female,favg);
```

Imputing Missing Data: Nearest Neighbor

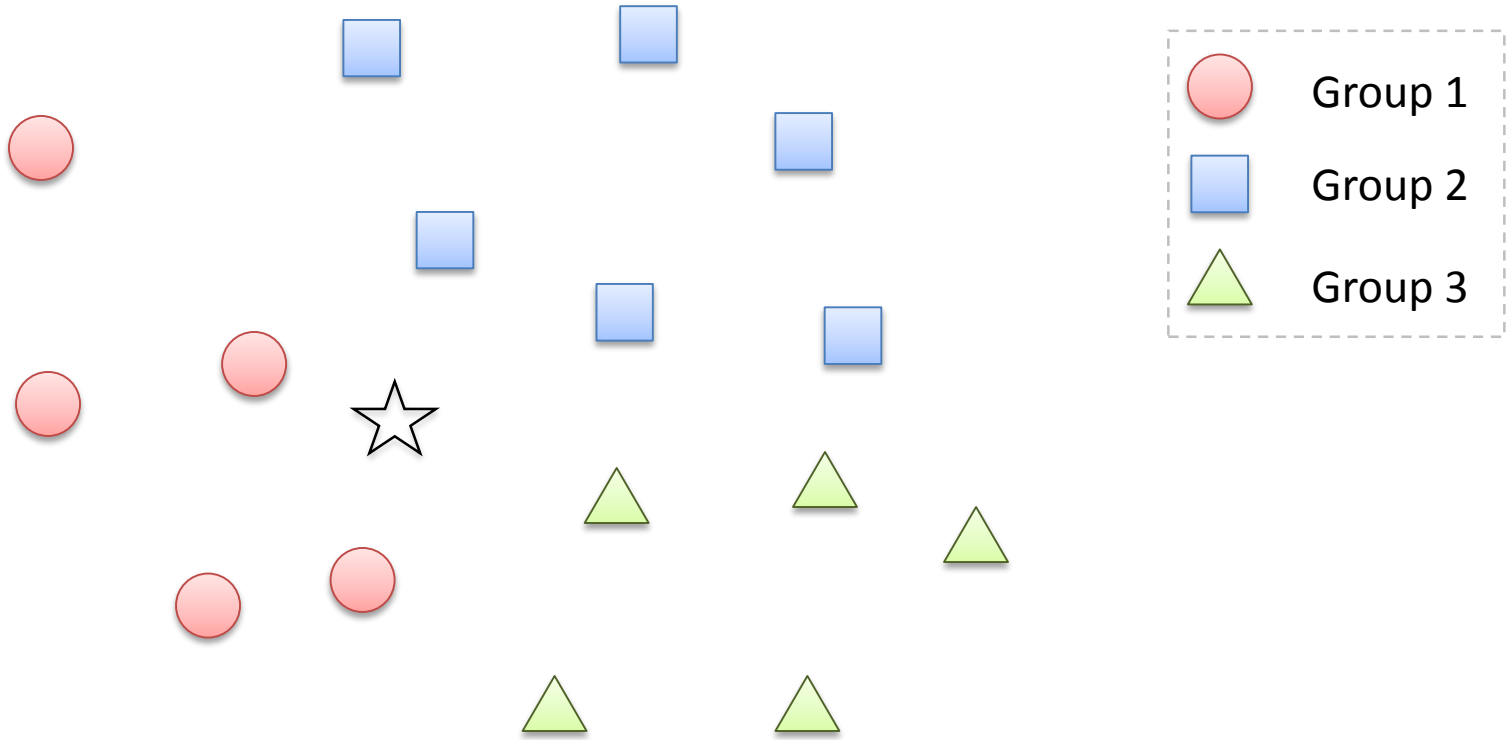
- K-Nearest Neighbor (k-NN)
  In pattern recognition, the k-nearest neighbor algorithm (k-NN) is a method for classifying objects based on closest training examples in the feature space.

  Ref: Wikipedia

- How is k-NN related to imputing missing data?

Intro: Nearest Neighborhood

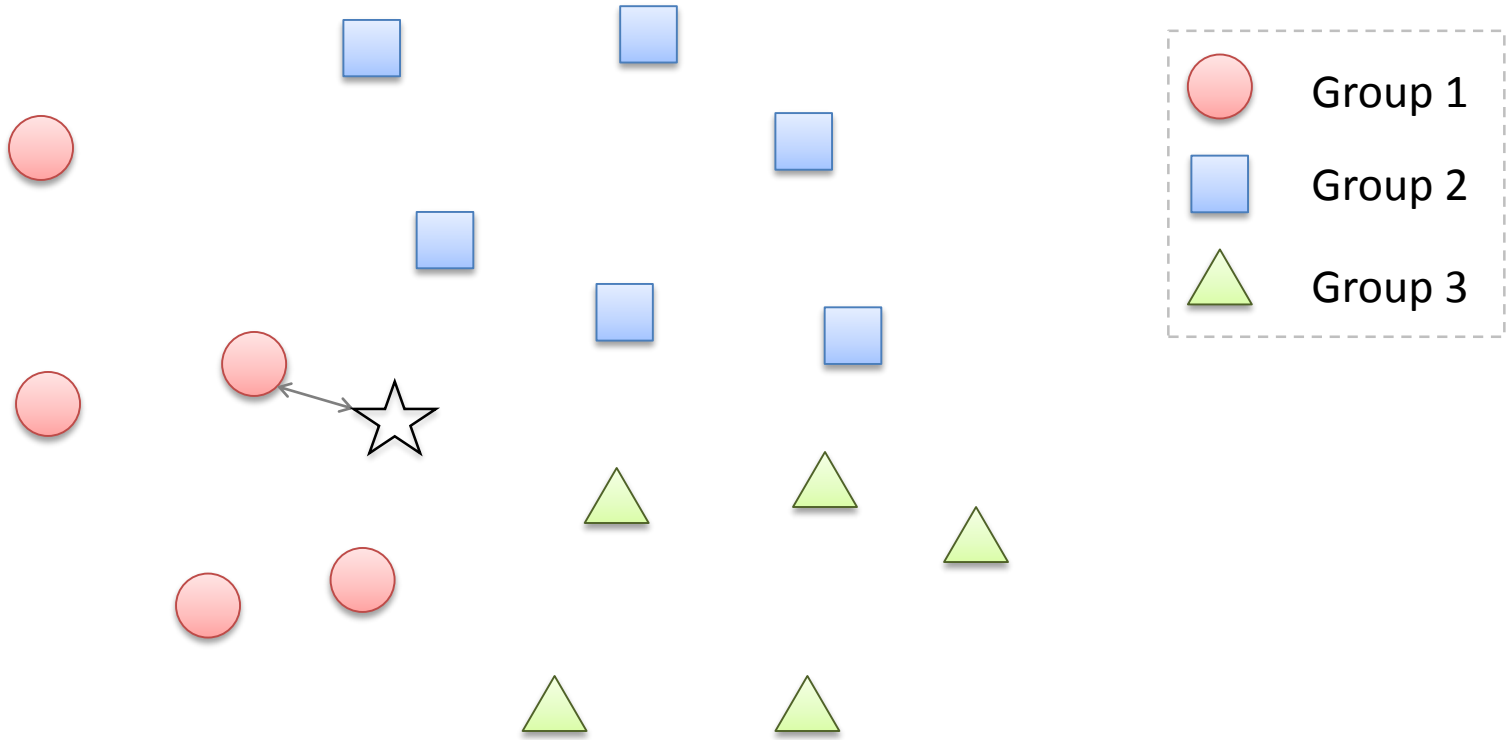
- Nearest Neighborhood



Is the new observation in Group 1? Group 2? Group 3?

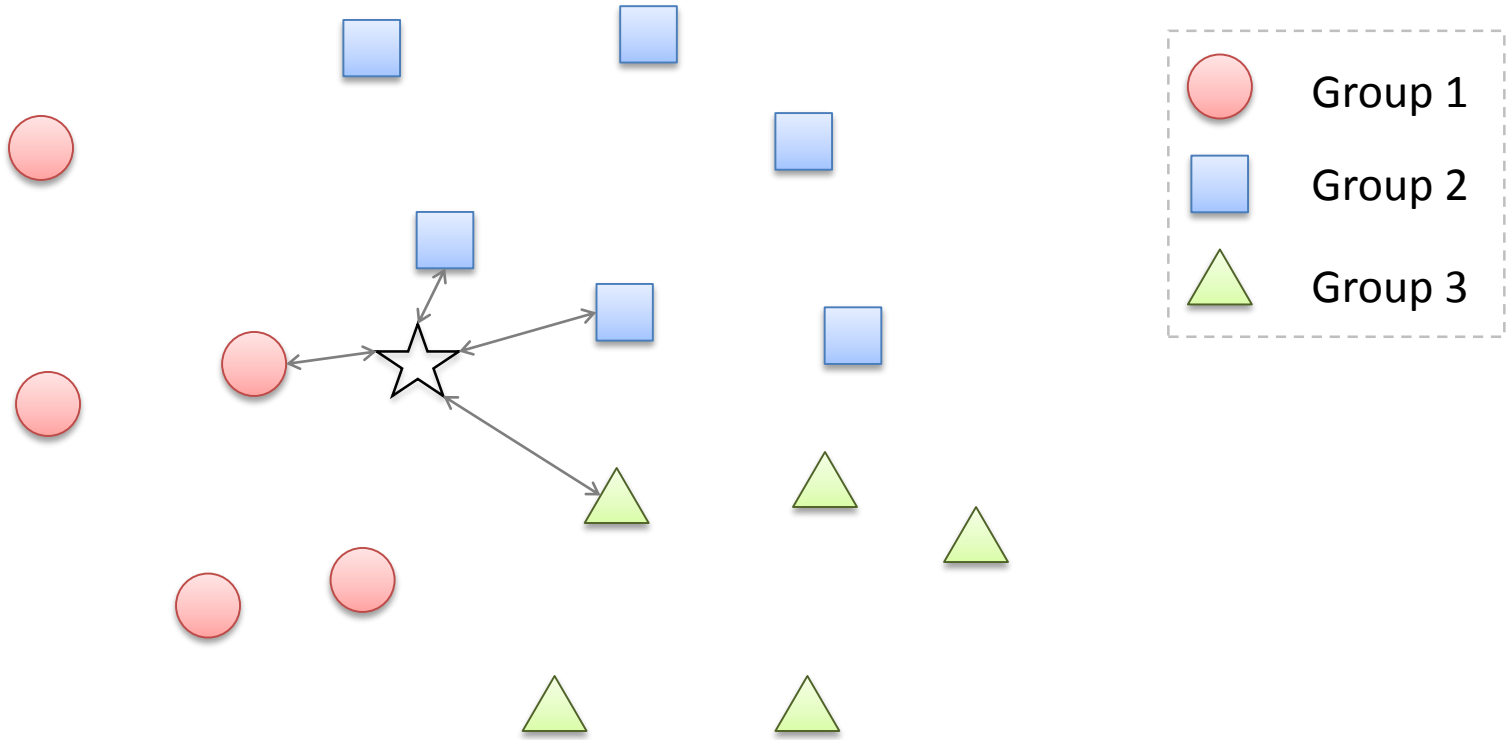
Intro: 1-Nearest Neighborhood

- 1-NN: find the closest neighbor and assign to the same group



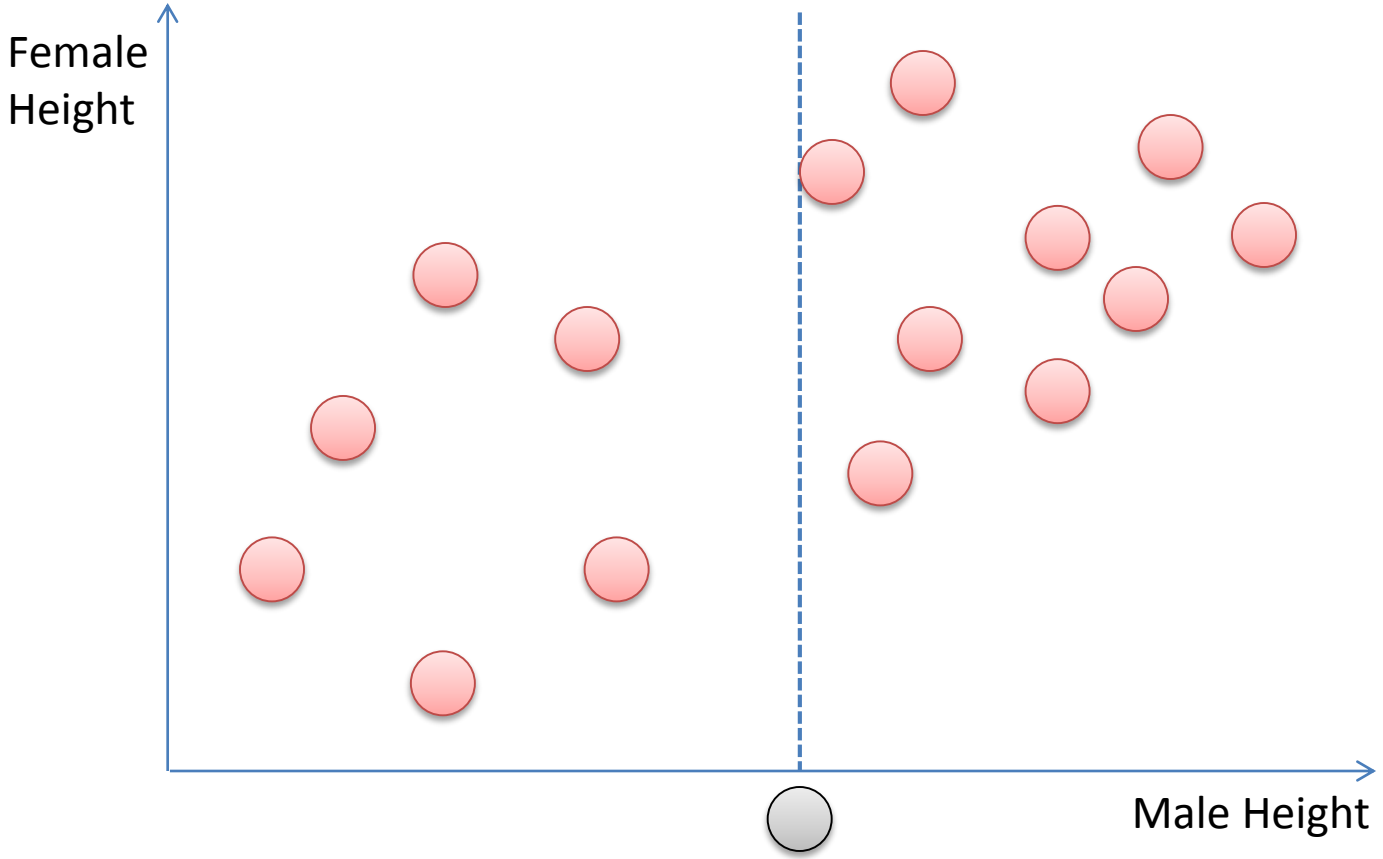
Intro: k-Nearest Neighborhood

- k-NN: find the k neighbors with smallest distances and assign to the major group



Imputation with 1-NN

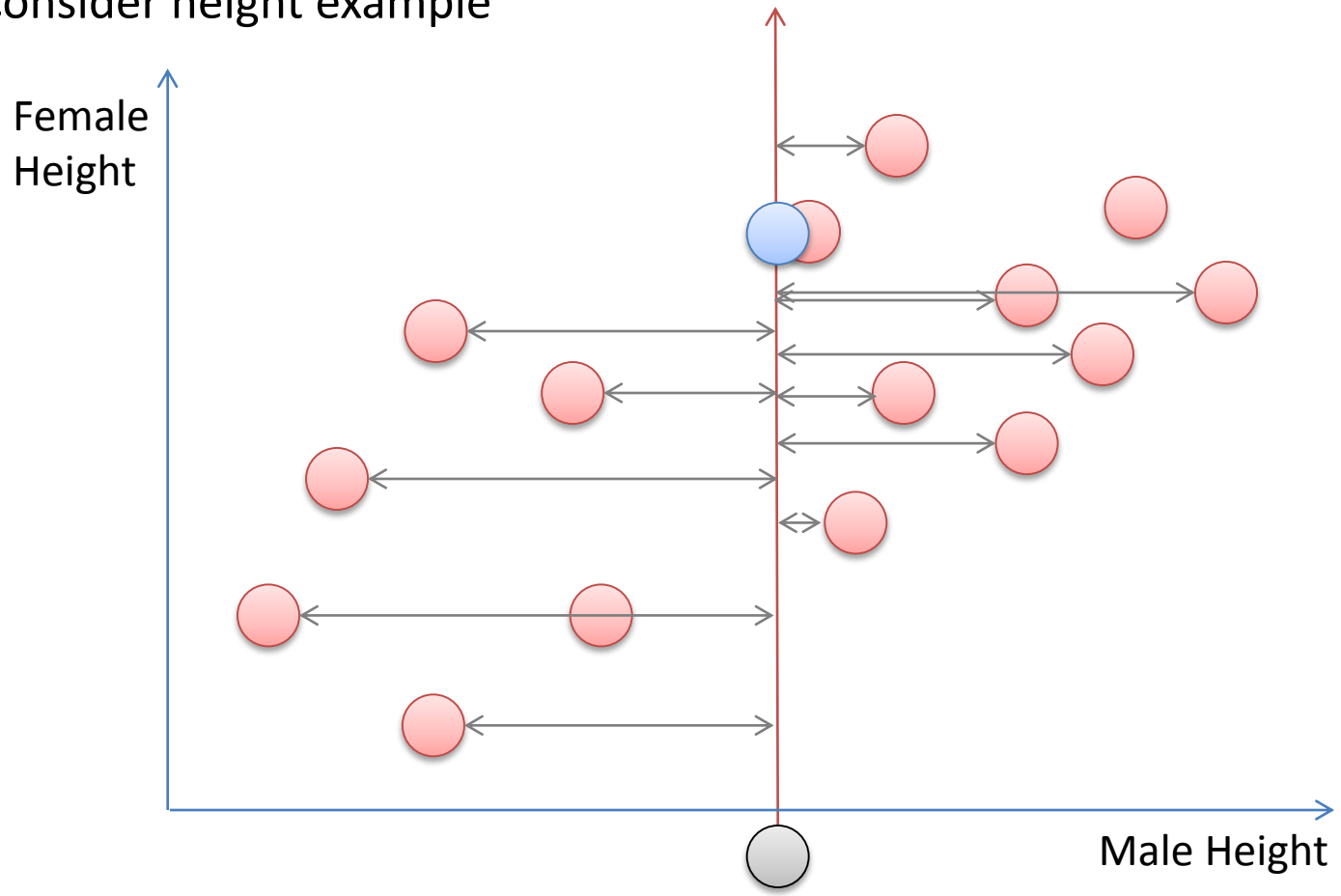
- Consider height example



- An observation with missing female height
- How can we impute the female height value?

Imputation with 1-NN

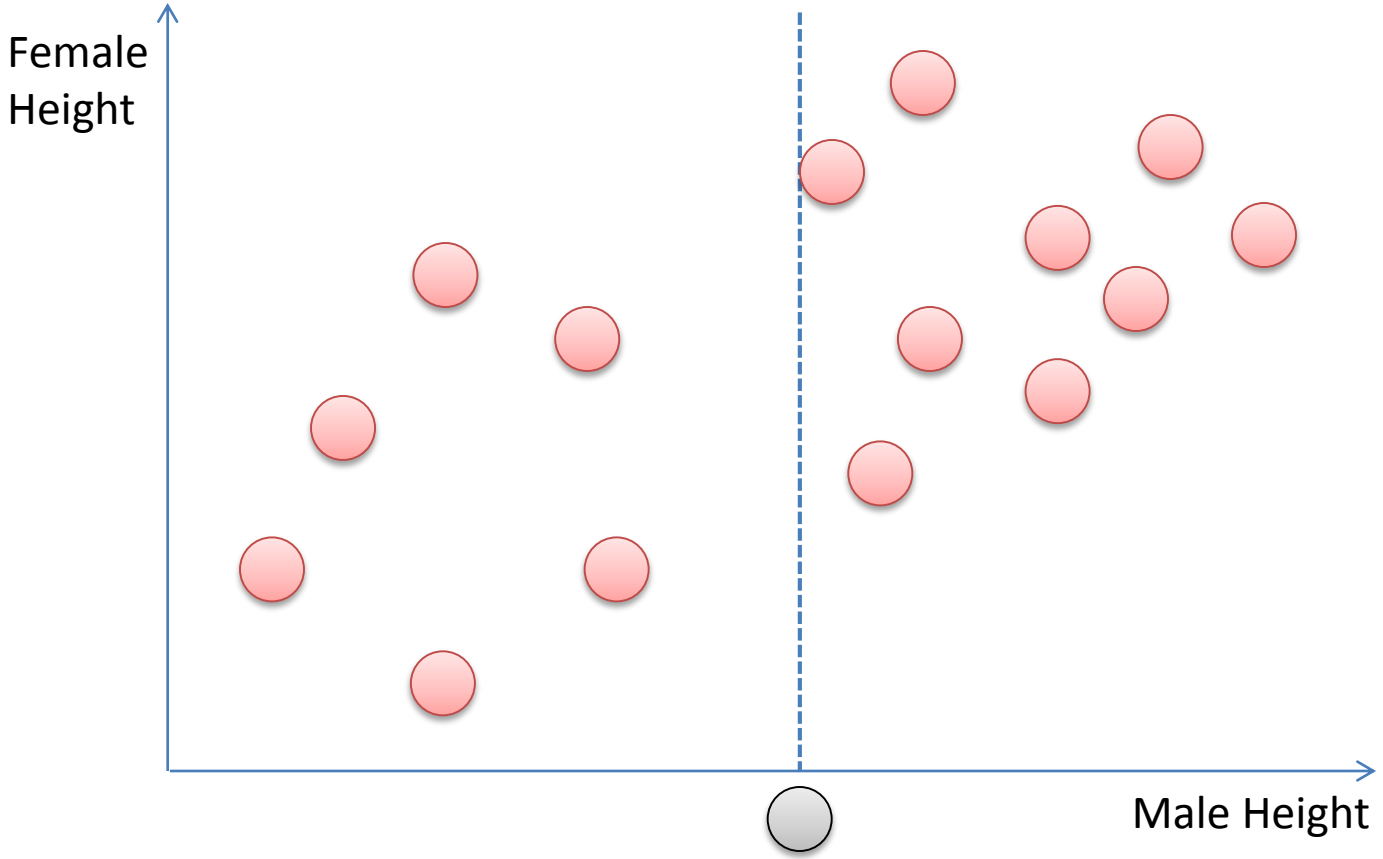
- Consider height example



- An observation with missing female height
- How can we impute the female height value?

Imputation with k-NN

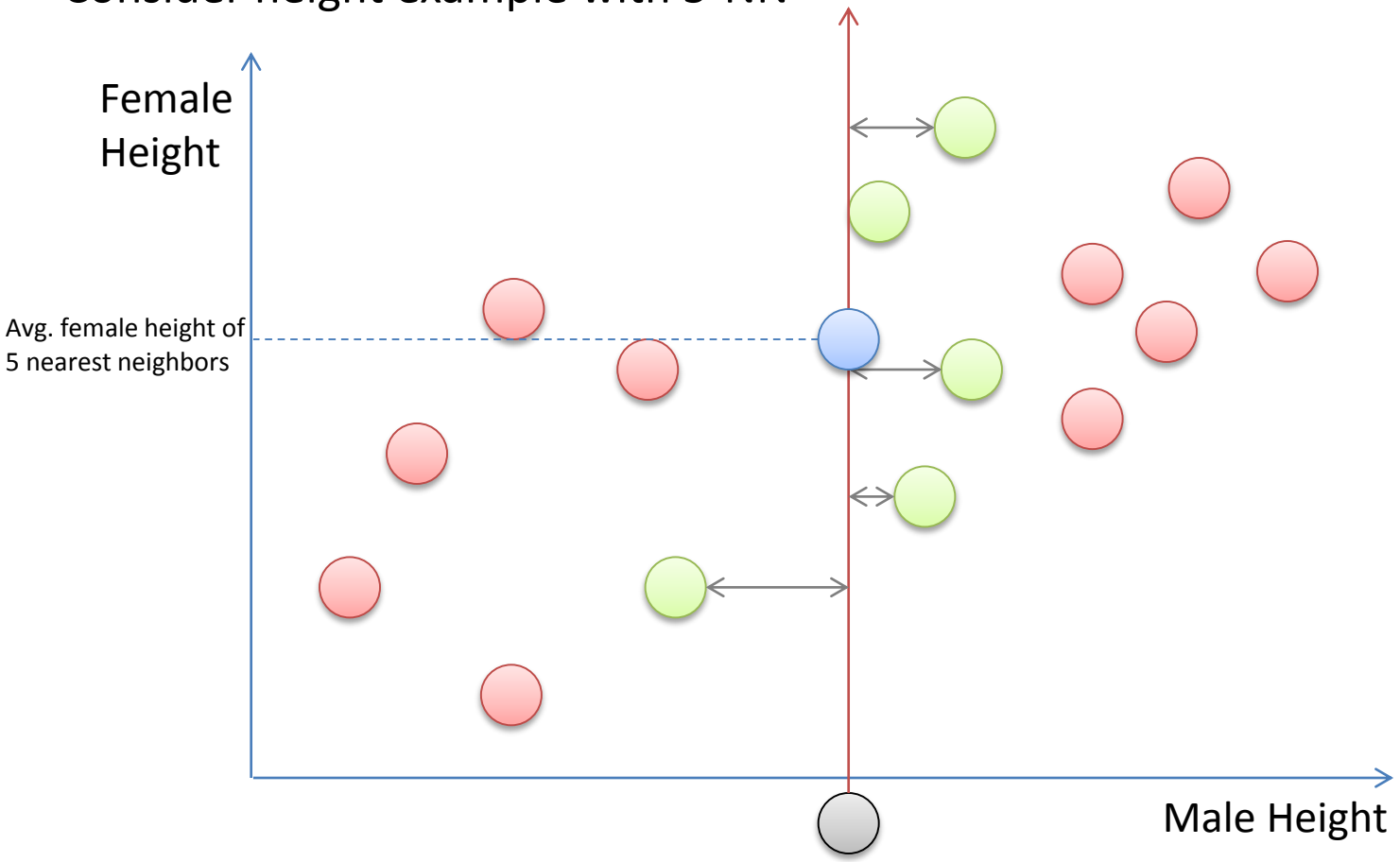
- Consider height example with 5-NN



- An observation with missing female height
- How can we impute the female height value?

Imputation with k-NN

- Consider height example with 5-NN



- An observation with missing female height
- How can we impute the female height value?

Imputation with k-NN in R

- load package imputation

```
> library(imputation)
```

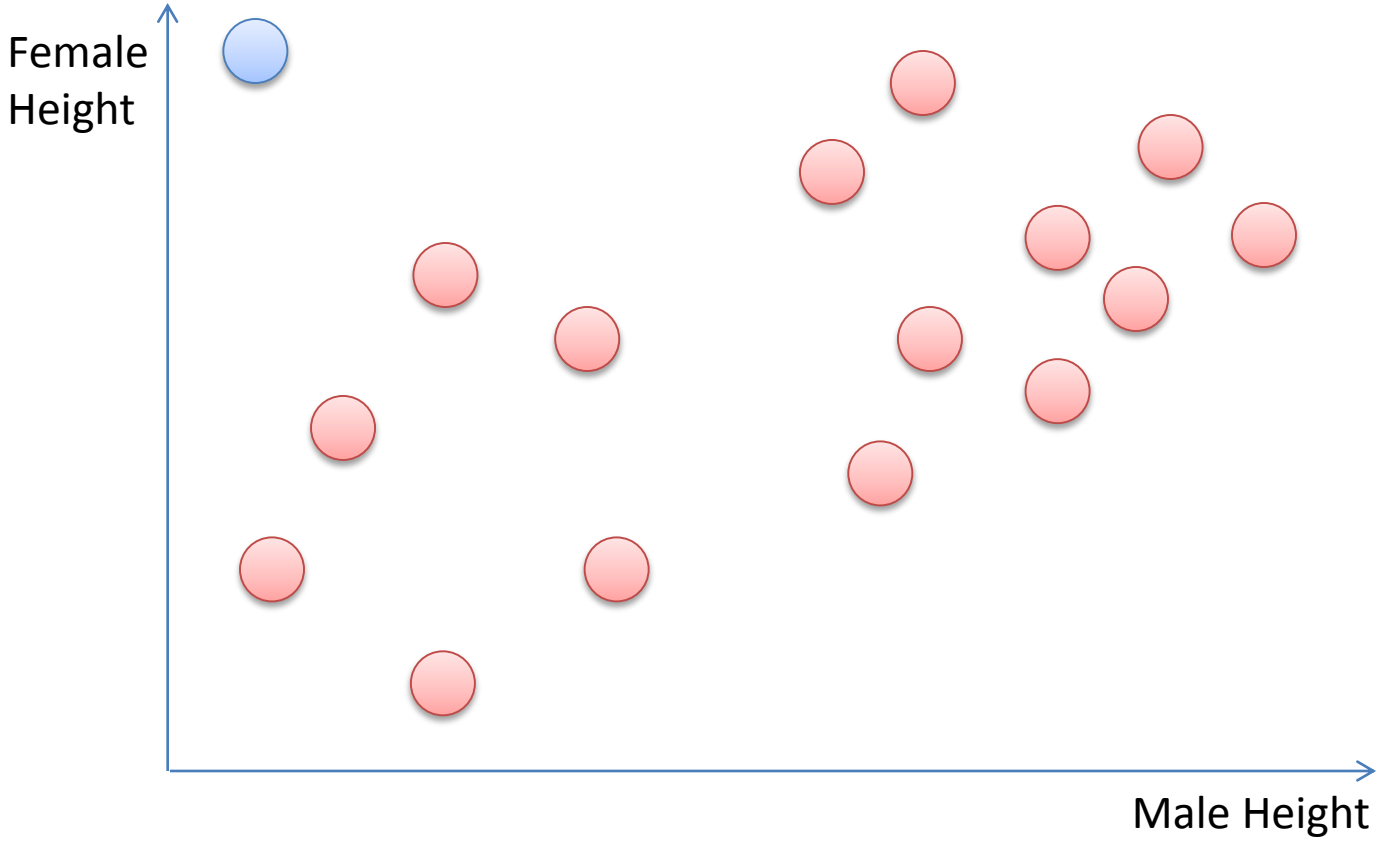
- Use Function

```
> height.1NN = kNNImpute(height,1);
> height.5NN = kNNImpute(height,5);
```

Package ‘imputation’ was removed from the CRAN repository.
You can manually download and install the latest version, but it has lots of dependencies
So, do not install this.
Instead, using the concept of KNN, you may implement your function

Outliers

- Consider height example



Should we consider outliers in analysis?

Outliers Elimination in R

- Removing observations that are outside of $\pm 3\sigma$ range from mean

Step1 Calculate mean, standard deviation, upper and lower limits

```
> male.mean = mean(height.omit$Male)
> female.mean = mean(height.omit$Female)
> male.std = sd(height.omit$Male)
> female.std = sd(height.omit$Female)
> male.lb = male.mean -male.std          # To see the effect clearly, use ±σ
> male.ub = male.mean + male.std         # To see the effect clearly, use ±σ
> female.lb = female.mean - female.std   # To see the effect clearly, use ±σ
> female.ub = female.mean + female.std   # To see the effect clearly, use ±σ
```

Step2 Remove observations outside of the range

```
> subset(height.omit, Male<male.ub & Male>male.lb & Female<female.ub & Female>female.lb)
```

Outliers Elimination in R with a package

- load package outliers

```
> library(outliers)
```

- Simple outlier detection: observation farthest from the sample average

```
> outlier(height.omit);                    ## returns the observation
```

- Outlier elimination

```
> height.out = rm.outlier(height.omit);
> height.out = rm.outlier(height.omit, median=T);   # use median instead of mean
```