

### 5.3. OOP in R

#### 1. setClass:

**Example (R):**

```
setClass(
  "AnalysisTemplate",
  representation(getStatistics = "function",
                output = "function",
                createPlot = "function",
                doAnalysis = "function"),
  prototype = list
  (
    doAnalysis = function(response,
                          covariate,
                          getStatistics,
                          output,
                          createPlot)
  {
    statistics = getStatistics(response, covariate)$statistics
    output(statistics)
    createPlot(statistics)
  }
)
)
setClass(
  "LinearRegressionAnalysis",
  contains = "AnalysisTemplate",
  prototype = list
  (
    getStatistics = function(response, covariate)
    {
      list(statistics = lm(response ~ covariate))
    },
    output = function(obj)
    {
      write(obj$residuals,"D:\\residuals")
    },
    createPlot = function(obj)
```

```

    {
      par(mfrow = c(1, 2))
      plot(obj$fitted.values, obj$residuals, xlab = "Fitted Values",
           ylab = "Residuals")
      plot(obj$fitted.values + obj$residuals, obj$fitted.values,
           xlab = "Response", ylab = "Fitted Values")
    }
  )
)

lmResponse = c(58, 105, 88, 118, 117, 137, 157, 169, 149, 202)
lmCovariate = c(2, 6, 8, 8, 12, 16, 20, 20, 22, 26)
lmExample = new("LinearRegressionAnalysis")
lmExample@doAnalysis(lmResponse,
                      lmCovariate,
                      lmExample@getStatistics,
                      lmExample@output,
                      lmExample@createPlot)

```

## 2. setMethod:

### **Example (R):**

```

setClass(
  "EstimationStrategy",
  representation(locationEstimate = "function",
                dispersionEstimate = "function")
)

setClass(
  "UnbiasedEstimate",
  contains = "EstimationStrategy",
  prototype = list
  (
    locationEstimate = function(data)
    {
      mean(data)
    },
    dispersionEstimate = function(data)
    {
      var(data)
    }
  )
)

```

```

        }
    )
)

setClass(
  "RobustEstimate",
  contains = "EstimationStrategy",
  prototype = list
(
  locationEstimate = function(data)
  {
    median(data)
  },
  dispersionEstimate = function(data)
  {
    median(abs(data - median(data)))
  }
)
)

setGeneric(
  "meanSquareError",
  function(trueValue,
         data,
         estimationStrategy)
  standardGeneric("meanSquareError")
)

setMethod(
  "meanSquareError",
  signature("numeric",
           "matrix",
           "EstimationStrategy"),
  function(trueValue,
         data,
         estimationStrategy)
{
  list(sum((apply(data, 2, estimationStrategy@locationEstimate) -

```

```
    trueValue[1]) ^ 2) / ncol(data),
    sum((apply(data, 2, estimationStrategy@dispersionEstimate) -
        trueValue[2]) ^ 2) / ncol(data)))
}

)

data = as.matrix(rnorm(100))
unbiasedEstimate = new("UnbiasedEstimate")
meanSquareError(c(0, 1), data, unbiasedEstimate)
robustEstimate = new("RobustEstimate")
meanSquareError(c(0, 1), data, robustEstimate)
```