ORF525, Assignment 1, Problem 3

Data preparation

Let's load the data and extract necessary features. The response variables are stored as vector Y, the design matrix is stored as X.

```
macro = read.csv("macro.csv", header=T)

month = macro[,1]
Month = strptime(month, "%m/%d/%Y")

PCE = macro[,4]

n = length(PCE) - 2
p = 7

Unrate = macro[,25]
IndPro = macro[,7]
HouSta = macro[,49]
M2Real = macro[,67]
FedFund= macro[,79]
CPI  =  macro[,107]
SPY  =  macro[,75]

Y = diff(log(PCE))[2:(n+1)]

X = matrix(0, nrow=n, ncol=p)
X[, 1] = Unrate[2:(n+1)]
X[, 2] = diff(log(IndPro))[1:n]
X[, 3] = diff(log(M2Real))[1:n]
X[, 4] = diff(log(CPI))[1:n]
X[, 5] = diff(log(SPY))[1:n]
X[, 6] = HouSta[2:(n+1)]
X[, 7] = FedFund[2:(n+1)]
```

Then we need to split the data into train and test parts:

```
Y.train = Y[1:(n-10*12)]
Y.test = Y[(n-10*12+1):n]
X.train = X[1:(n-10*12), ]
X.test = X[(n-10*12+1):n, ]
N.train = n - 10*12
N.test = 10*12
```

Let's create the train and test dataframes:

```
data_train = data.frame(logPCE=Y.train, X.train)
data_test = data.frame(X.test)
```

(a)

Let's fit the linear model:

```r
model_lm = lm(logPCE ~ ., data=data_train)
summary(model_lm)
```

```
##
## Call:
## lm(formula = logPCE ~ ., data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0303806 -0.0028201  0.0000478  0.0031962  0.0184955
##
## Coefficients:
##                Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.586e-03  1.657e-03   0.957   0.3389
## X1           1.966e-04  1.760e-04   1.117   0.2643
## X2           2.023e-02  2.985e-02   0.678   0.4983
## X3           1.798e-01  7.090e-02   2.536   0.0115 *
## X4          -1.939e-02  1.194e-01  -0.162   0.8711
## X5          -7.537e-04  6.873e-03  -0.110   0.9127
## X6           5.000e-07  7.879e-07   0.635   0.5260
## X7          -1.516e-04  9.156e-05  -1.656   0.0982 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005528 on 563 degrees of freedom
## Multiple R-squared:  0.04708,    Adjusted R-squared:  0.03523
## F-statistic: 3.974 on 7 and 563 DF,  p-value: 0.000298
```

We see that $\hat{\sigma}^2 = 0.005528^2 \approx 3.1 \times 10^{-5}$, adjusted $R^2 = 0.03523$ We can also compute them manually:

```r
Y.pred_train = model_lm$fitted.values
    # same as X.train %*% model_lm$coefficients[2:9] + rep(model_lm$coefficients[1], N.train)
    #    or predict(model_lm, newdata=data_train)
RSS = sum((Y.pred_train - Y.train)^2)
    # same as sum(model_lm$residuals^2)
TSS = sum((Y.train - ave(Y.train))^2)

sigma_hat_2 = RSS/(N.train-(p+1))
    # same as sigma(model_lm)^2
sprintf("hat sigma ^2 = %f", sigma_hat_2)
```

```
## [1] "hat sigma ^2 = 0.000031"
```

```r
R2_adj = 1 - (N.train-1)*RSS/((N.train-(p+1))*TSS)
sprintf("Adjusted R^2 = %f", R2_adj)
```

```
## [1] "Adjusted R^2 = 0.035231"
```

Insignificant variables, as can be seen from the summary, e.g. for significance level 0.05, are all except $X3$.

(b)

Let's first use function *step* based on AIC:

```r
elimination_step = step(model_lm)
```

```
## Start:  AIC=-5928.12
## logPCE ~ X1 + X2 + X3 + X4 + X5 + X6 + X7
##
##        Df  Sum of Sq      RSS     AIC
## - X5    1 3.6800e-07 0.017204 -5930.1
## - X4    1 8.0500e-07 0.017205 -5930.1
## - X6    1 1.2305e-05 0.017216 -5929.7
## - X2    1 1.4030e-05 0.017218 -5929.7
## - X1    1 3.8147e-05 0.017242 -5928.9
## <none>               0.017204 -5928.1
## - X7    1 8.3814e-05 0.017288 -5927.3
## - X3    1 1.9657e-04 0.017400 -5923.6
##
## Step:  AIC=-5930.11
## logPCE ~ X1 + X2 + X3 + X4 + X6 + X7
##
##        Df  Sum of Sq      RSS     AIC
## - X4    1 7.2700e-07 0.017205 -5932.1
## - X6    1 1.2289e-05 0.017216 -5931.7
## - X2    1 1.4113e-05 0.017218 -5931.6
## - X1    1 3.7802e-05 0.017242 -5930.9
## <none>               0.017204 -5930.1
## - X7    1 8.3562e-05 0.017288 -5929.3
## - X3    1 1.9620e-04 0.017400 -5925.6
##
## Step:  AIC=-5932.08
## logPCE ~ X1 + X2 + X3 + X6 + X7
##
##        Df  Sum of Sq      RSS     AIC
## - X6    1 0.00001157 0.017216 -5933.7
## - X2    1 0.00001461 0.017219 -5933.6
## - X1    1 0.00003722 0.017242 -5932.8
## <none>               0.017205 -5932.1
## - X7    1 0.00011937 0.017324 -5930.1
## - X3    1 0.00032894 0.017534 -5923.3
##
## Step:  AIC=-5933.7
## logPCE ~ X1 + X2 + X3 + X7
##
##        Df  Sum of Sq      RSS     AIC
## - X2    1 0.00002271 0.017239 -5934.9
## - X1    1 0.00003308 0.017250 -5934.6
## <none>               0.017216 -5933.7
## - X7    1 0.00012764 0.017344 -5931.5
## - X3    1 0.00034643 0.017563 -5924.3
##
## Step:  AIC=-5934.95
## logPCE ~ X1 + X3 + X7
##
##        Df  Sum of Sq      RSS     AIC
## - X1    1 0.00003290 0.017272 -5935.9
## <none>               0.017239 -5934.9
## - X7    1 0.00014538 0.017385 -5932.2
## - X3    1 0.00035576 0.017595 -5925.3
```

```
##
## Step:  AIC=-5935.86
## logPCE ~ X3 + X7
##
##        Df  Sum of Sq      RSS     AIC
## <none>               0.017272 -5935.9
## - X7    1 0.00011531 0.017387 -5934.1
## - X3    1 0.00042710 0.017699 -5923.9
```

```
summary(elimination_step)
```

```
##
## Call:
## lm(formula = logPCE ~ X3 + X7, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0301015 -0.0029233  0.0001939  0.0031614  0.0187571
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.398e-03  5.501e-04   6.178 1.24e-09 ***
## X3           2.052e-01  5.476e-02   3.748 0.000197 ***
## X7          -1.440e-04  7.392e-05  -1.947 0.051990 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005514 on 568 degrees of freedom
## Multiple R-squared:  0.0433, Adjusted R-squared:  0.03993
## F-statistic: 12.85 on 2 and 568 DF,  p-value: 3.473e-06
```

We see that this approach eliminates all variables except $X3$ and $X7$. So, the model $\widehat{\mathcal{M}}$ consists of the features $X3, X7$.

Elimination by $|t|$-statistic can be performed as follows:

```
new_step <- function(fit, threshold)
{
  summary(fit)
  names = variable.names(fit)
  tvalue <- coef(summary(fit))[2:length(names), 't value']
  Pvalue <- coef(summary(fit))[2:length(names), "Pr(>|t|)"]
  names = names[-1]

  while (sum(Pvalue[1:length(names)] > threshold) != 0)
  {
    idx = which.min(abs(tvalue))
    print(paste('drop variable:', names[idx]))
    new_formula =as.formula(paste("logPCE", paste0(names[-idx], collapse='+'), sep='~') )
    fit <- update(fit, new_formula)
    print(summary(fit))

    names = variable.names(fit)
    tvalue <- coef(summary(fit))[2:length(names), 't value']
    Pvalue <- coef(summary(fit))[2:length(names), "Pr(>|t|)"]
    names = names[-1]
```

```r
  }
  return(fit)
}

threshold = 0.05
elimination_newstep = new_step(model_lm, threshold)
```

```
## [1] "drop variable: X5"
##
## Call:
## lm(formula = logPCE ~ X1 + X2 + X3 + X4 + X6 + X7, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0303849 -0.0028136  0.0000451  0.0031819  0.0184799
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.592e-03  1.655e-03   0.962   0.3366
## X1           1.943e-04  1.745e-04   1.113   0.2661
## X2           2.028e-02  2.982e-02   0.680   0.4967
## X3           1.795e-01  7.078e-02   2.536   0.0115 *
## X4          -1.837e-02  1.190e-01  -0.154   0.8774
## X6           4.997e-07  7.872e-07   0.635   0.5259
## X7          -1.513e-04  9.144e-05  -1.655   0.0985 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005523 on 564 degrees of freedom
## Multiple R-squared:  0.04706,    Adjusted R-squared:  0.03692
## F-statistic: 4.642 on 6 and 564 DF,  p-value: 0.0001282
##
## [1] "drop variable: X4"
##
## Call:
## lm(formula = logPCE ~ X1 + X2 + X3 + X6 + X7, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0304007 -0.0028534  0.0000667  0.0031809  0.0185054
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1.617e-03  1.646e-03   0.983  0.32619
## X1           1.922e-04  1.738e-04   1.106  0.26937
## X2           2.059e-02  2.973e-02   0.693  0.48879
## X3           1.861e-01  5.661e-02   3.287  0.00108 **
## X6           4.678e-07  7.591e-07   0.616  0.53794
## X7          -1.582e-04  7.989e-05  -1.980  0.04820 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005518 on 565 degrees of freedom
## Multiple R-squared:  0.04702,    Adjusted R-squared:  0.03858
```

```
## F-statistic: 5.575 on 5 and 565 DF,  p-value: 5.026e-05
##
## [1] "drop variable: X6"
##
## Call:
## lm(formula = logPCE ~ X1 + X2 + X3 + X7, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0302960 -0.0028606  0.0000865  0.0032174  0.0185383
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.423e-03  9.995e-04   2.424 0.015672 *
## X1           1.800e-04  1.726e-04   1.043 0.297435
## X2           2.494e-02  2.886e-02   0.864 0.387906
## X3           1.898e-01  5.625e-02   3.375 0.000789 ***
## X7          -1.628e-04  7.949e-05  -2.048 0.040975 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005515 on 566 degrees of freedom
## Multiple R-squared:  0.04638,    Adjusted R-squared:  0.03964
## F-statistic: 6.882 on 4 and 566 DF,  p-value: 2.058e-05
##
## [1] "drop variable: X2"
##
## Call:
## lm(formula = logPCE ~ X1 + X3 + X7, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0301542 -0.0028925  0.0001164  0.0032477  0.0185879
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  2.543e-03  9.896e-04   2.569  0.01044 *
## X1           1.795e-04  1.726e-04   1.040  0.29864
## X3           1.922e-01  5.618e-02   3.421  0.00067 ***
## X7          -1.722e-04  7.874e-05  -2.187  0.02918 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005514 on 567 degrees of freedom
## Multiple R-squared:  0.04512,    Adjusted R-squared:  0.04007
## F-statistic: 8.931 on 3 and 567 DF,  p-value: 8.657e-06
##
## [1] "drop variable: X1"
##
## Call:
## lm(formula = logPCE ~ X3 + X7, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
```

```
## -0.0301015 -0.0029233  0.0001939  0.0031614  0.0187571
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  3.398e-03  5.501e-04   6.178 1.24e-09 ***
## X3           2.052e-01  5.476e-02   3.748 0.000197 ***
## X7          -1.440e-04  7.392e-05  -1.947 0.051990 .
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005514 on 568 degrees of freedom
## Multiple R-squared:  0.0433, Adjusted R-squared:  0.03993
## F-statistic: 12.85 on 2 and 568 DF,  p-value: 3.473e-06
##
## [1] "drop variable: X7"
##
## Call:
## lm(formula = logPCE ~ X3, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0303728 -0.0031330  0.0002735  0.0033267  0.0187162
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0024504  0.0002567   9.547  < 2e-16 ***
## X3          0.2412533  0.0516629   4.670 3.77e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005528 on 569 degrees of freedom
## Multiple R-squared:  0.03691,    Adjusted R-squared:  0.03522
## F-statistic: 21.81 on 1 and 569 DF,  p-value: 3.766e-06
```

```r
summary(elimination_newstep)
```

```
##
## Call:
## lm(formula = logPCE ~ X3, data = data_train)
##
## Residuals:
##        Min         1Q     Median         3Q        Max
## -0.0303728 -0.0031330  0.0002735  0.0033267  0.0187162
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 0.0024504  0.0002567   9.547  < 2e-16 ***
## X3          0.2412533  0.0516629   4.670 3.77e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.005528 on 569 degrees of freedom
## Multiple R-squared:  0.03691,    Adjusted R-squared:  0.03522
## F-statistic: 21.81 on 1 and 569 DF,  p-value: 3.766e-06
```

This elimination procedure leaves only one significant variable, so $\widehat{\mathcal{M}}$ consists of $X3$.

(c)

Let's fit the linear lodel $\widehat{\mathcal{M}}$, predict the values on the test set, and compute root mean-squared error (rMSE) and mean absolute deviation error (MADE).

```
model_lm_eliminated = lm(logPCE ~ X3, data=data_train)
Y.pred_test = predict(model_lm_eliminated, newdata=data_test)

rMSE = sqrt(mean((Y.test-Y.pred_test)^2))
sprintf("(c) rMSE = %f", rMSE)
```

```
## [1] "(c) rMSE = 0.003711"
```

```
MADE = mean(abs(Y.test-Y.pred_test))
sprintf("(c) MADE = %f", MADE)
```

```
## [1] "(c) MADE = 0.002798"
```

```
R2_out = 1 - sum((Y.pred_test-Y.test)^2)/sum((Y.test-rep(mean(Y.train), N.test))^2)
sprintf("(c) Out-of-sample R^2 = %f", R2_out)
```

```
## [1] "(c) Out-of-sample R^2 = -0.286801"
```

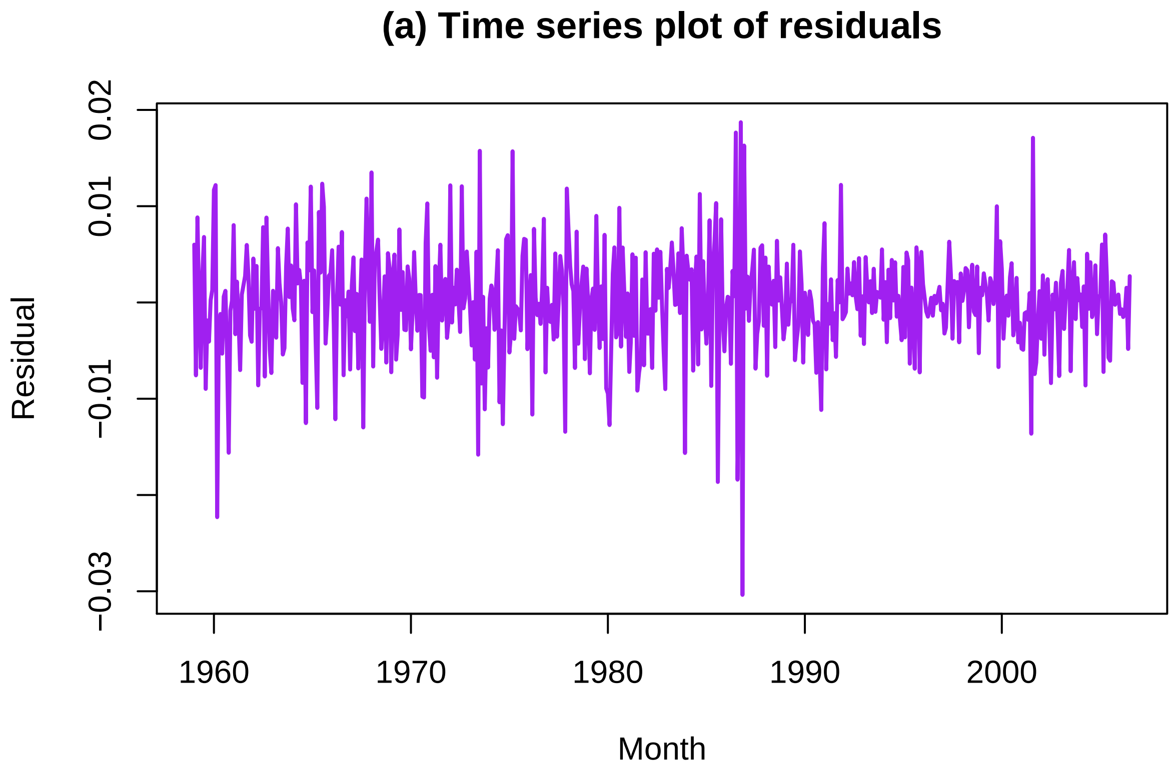
We see that for predition we have $rMSE = 0.003711$ and $MADE = 0.0.002798$. R^2 is negative, which means that the prediction by this model is worse than the prediction just by average of historical data.

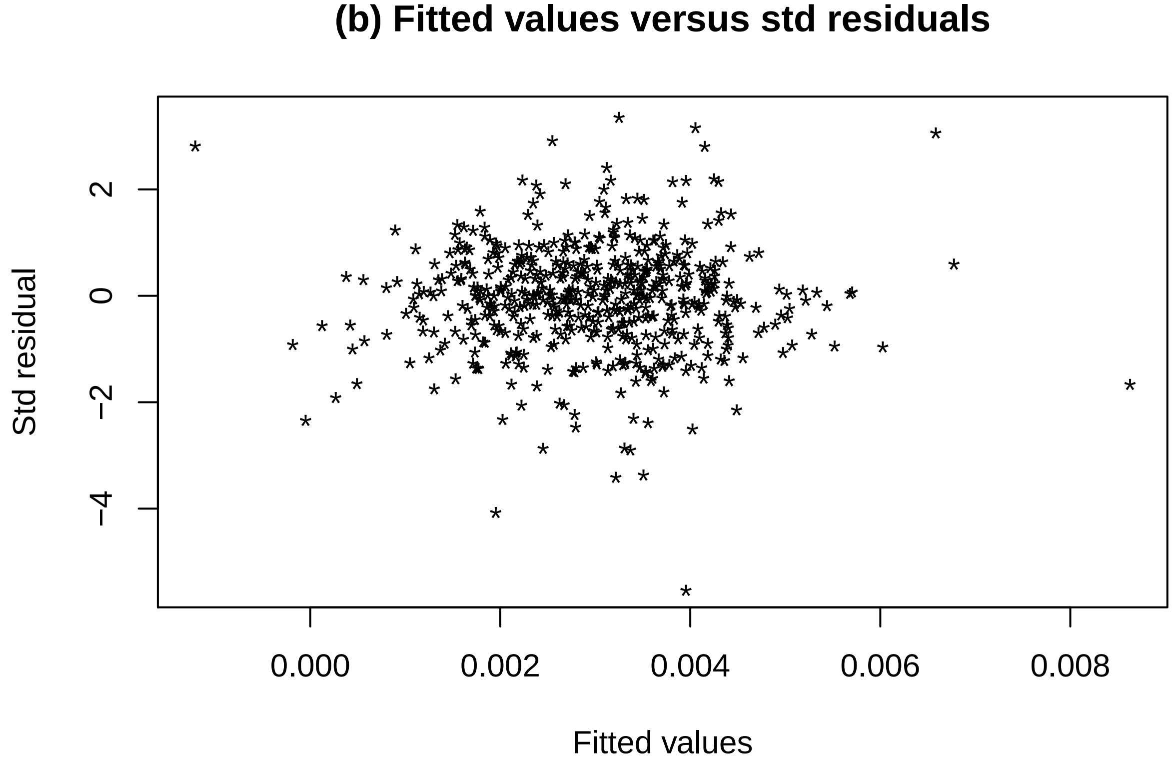
(d)

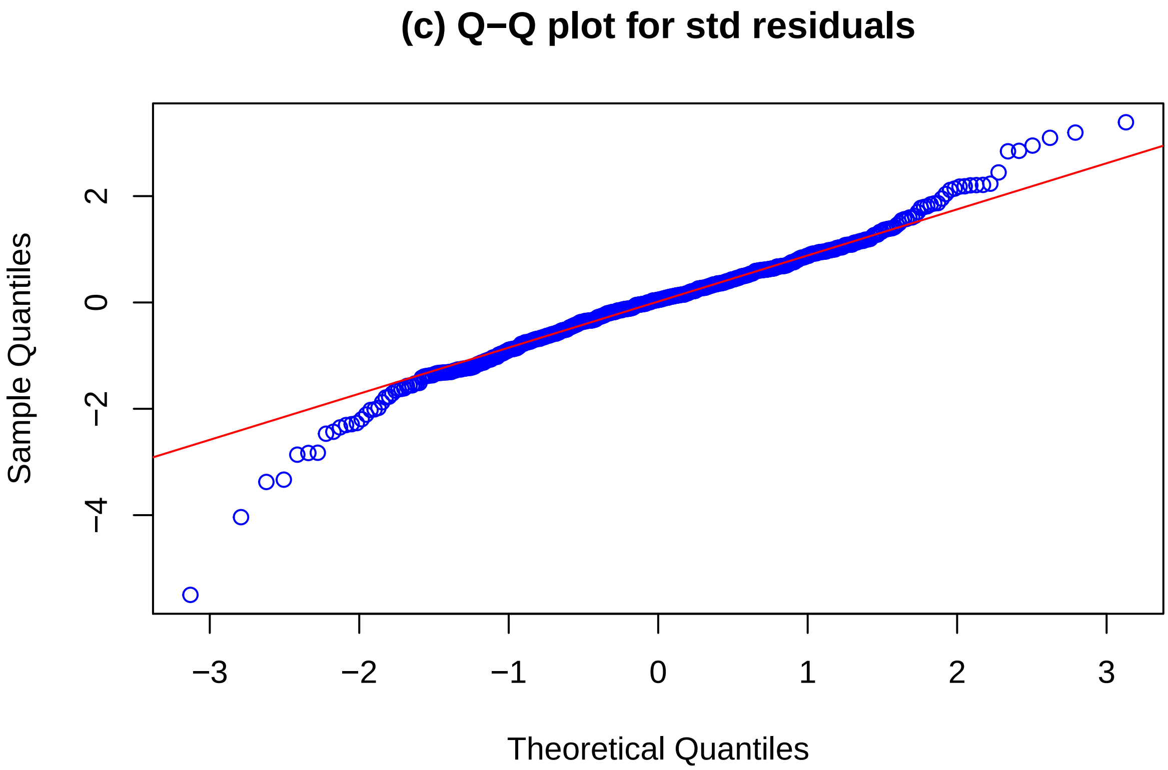
Take residuals and standardized residuals:

```
res = model_lm_eliminated$residuals
res_std = res/sd(res)
```

And now we build all the required plots:


(a) Time series plot of residuals
Residual
0.02
0.01
−0.01
−0.03
1960
1970
1980
1990
2000
Month


(b) Fitted values versus std residuals
Std residual
2
0
−2
−4
0.000
0.002
0.004
0.006
0.008
Fitted values


(c) Q−Q plot for std residuals
Sample Quantiles
Theoretical Quantiles
−4
−2
0
2
−3
−2
−1
0
1
2
3