Homework 1 Exercise 5

Daniel Winkler

First, a column-vector Y of the log houseprices is created.

```r
y <- as.matrix(house, ncol = 1)
y <- matrix(as.numeric(y[,2]), ncol = 1)
y <- log(y)
```

Next, a few helper functions are created. `trendf(t)` creates a matrix with the intercept and the trend up to power t as the columns (μ_t). As nothing else is changing in the varying models this function is used in the `model(t)` function which estimates an OLS model with the trend to power t and seasonal dummies (not yet defined!). `sums(model)` will return the relevant summaries for model selection.

```r
# Create trend to the power of t
trendf <- function(t){
  x <- matrix(1, nrow(y))
  trend <- 1:nrow(y)
  for(i in 1:t){
    x <- as.matrix(cbind(x, trend^i))
  }
  powers <- 1:t
  trend_names <- paste0("trend^", powers)
  colnames(x) <- c("intercept", trend_names)
  return(x)
}

model <- function(t){
  x_ <- trendf(t)
  mod <- lm(y ~ x_ + season - 1)
  return(mod)
  }

sums <- function(mod){
  a <-AIC(mod)
  b <- BIC(mod)
  s <- summary(mod)
  return(list(AIC = a, BIC = b, summary = s))
}
```

Now, the seasonal dummmies for months 1 to 11 are created (γ_t) by stacking the identity matrix with 12 diagonal elements (I_{12}) to fit the rows of the dependent variable.

```r
# Seasonal dummy for months 1 to 11
season <- diag(1, 12)
for(i in 1:as.integer(nrow(y)/12)){
  season <- rbind(season, diag(1,12))
}
season <- season[1:nrow(y), 1:11]
```

Next, we estimate the 4 models and find the ones preferred by the AIC and BIC. The AIC indicates that we should choose the third, i.e. trend to the power of three, and the BIC indicates that we should use the second, i.e. squared trend, model. The bias corrected AIC is calculated according to Shumway & Stoffer (p. 53, Eq. 2.20) and preferrs the 4th model.

```
m1 <- model(1)
m2 <- model(2)
m3 <- model(3)
m4 <- model(4)

models <- list(m1, m2, m3, m4)
AICs <- sapply(models, function(x) AIC(x))
AICcs <- sapply(models, function(x) log(sum(resid(x)^2)/nrow(y)) +
                    (nrow(y) + ncol(model.matrix(x))/
                        (nrow(y) - ncol(model.matrix(x)) - 2)))
BICs <- sapply(models, function(x) BIC(x))
# AIC preferrs model:
which(AICs==min(AICs))
```

```
## [1] 3
```

```
# AICc preferrs model:
which(AICcs==min(AICcs))
```

```
## [1] 4
```

```
# BIC preferrs model:
which(BICs==min(BICs))
```

```
## [1] 2
```

Finally, the data and fitted values as well as residuals are plotted. The models estimated do not adequatly acount for the variations in the data and thus the residuals are far from a white noise process. That is, systematic information is still found in the error term which could still be used and the covariance between error terms over time is non-zero. Despite that, all models show R^2s of over 99. However, the residuals can be interpreted as seasonally adjusted and de-trended data which can then be used to estimate an ARMA model. The fitted values for the 3rd and 4th model are virtually indistinguishable. The sum of squared differences between the fitted values of those models is 0.089. Thus, the 3rd model will be used for a quick ARMA exercise below. In terms of the fit, the first model finds a downward trend, while the others find close to a quadratic trend that first increases with a peak in the 1980s and decreases subsequently. In each plot of the fitted values the actual data is shown in red.
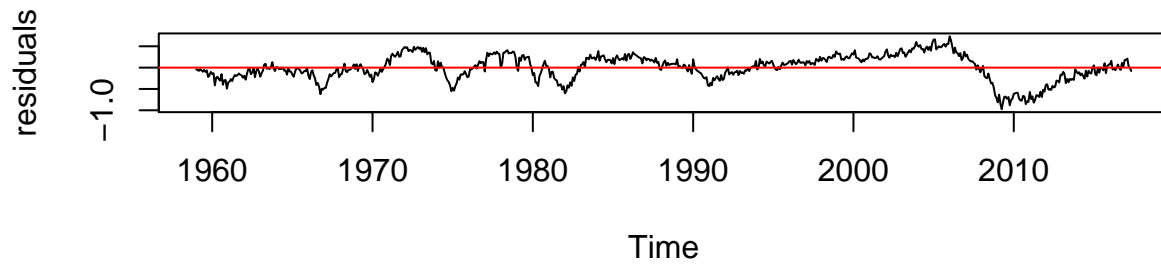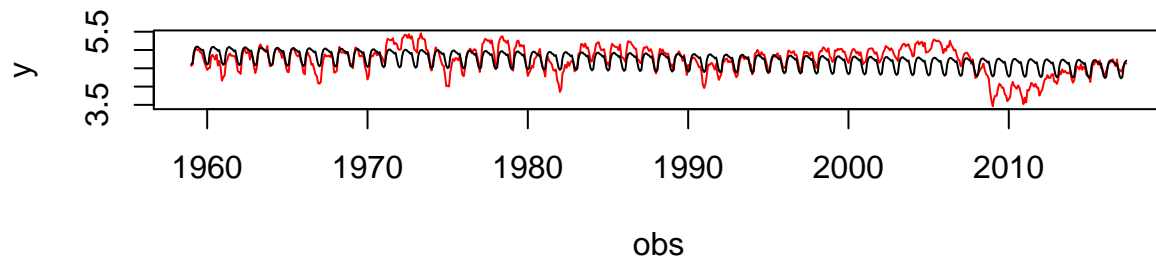
```
obs <- seq(as.Date("1959-01-01"), as.Date("2017-05-01"), by="months")
plot(y, x = obs, type = "l", main = "log(houseprices)", xlab = "time")
```
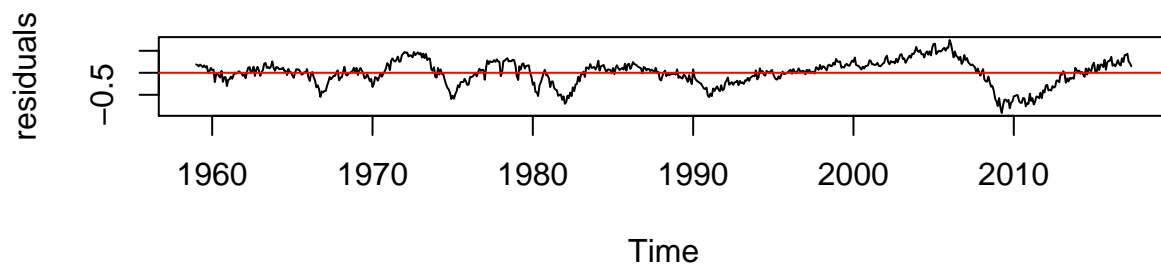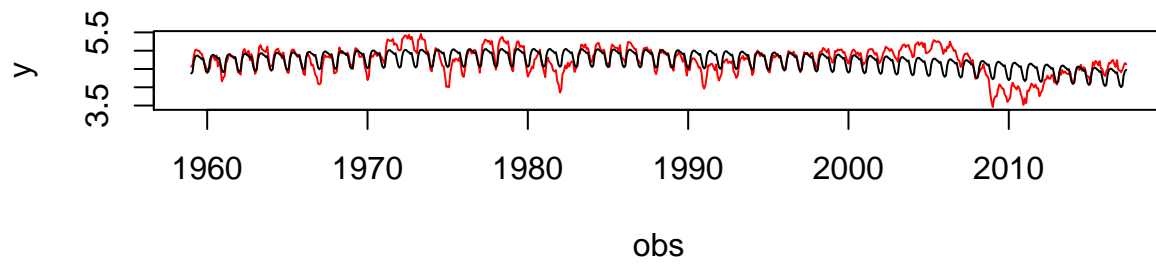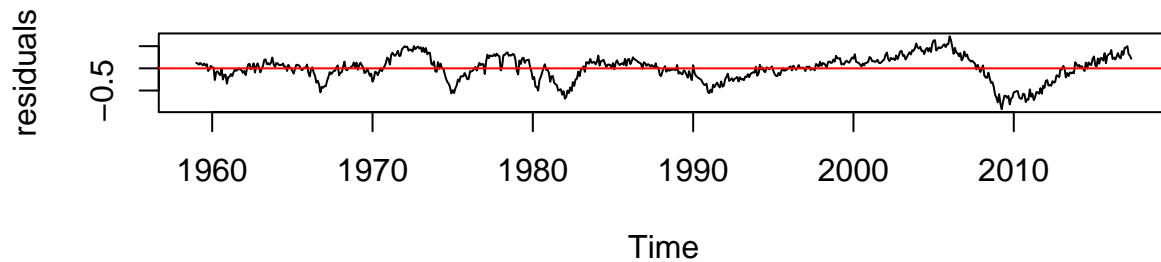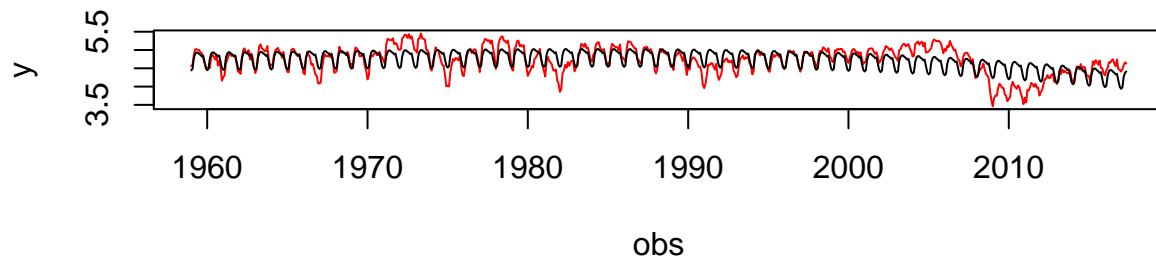
log(houseprices)
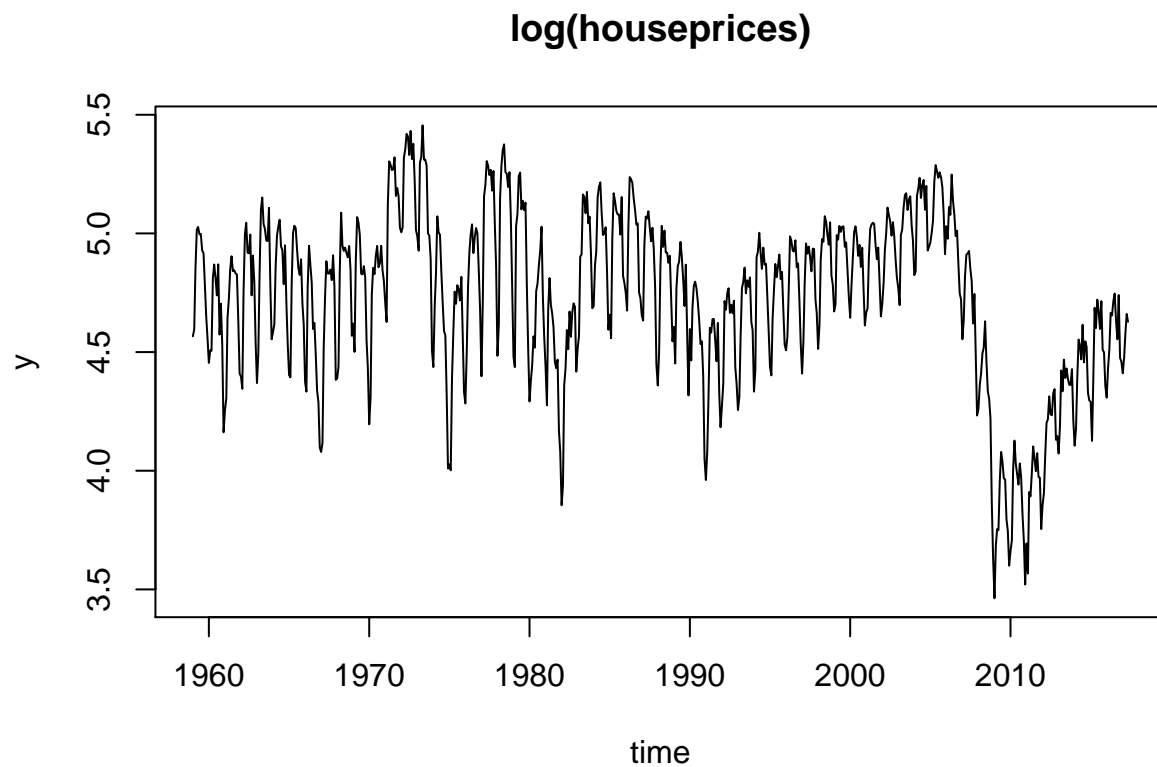

```
plotmod <- function(mod){
  par(mfrow = c(2,1))
  plot(y, x = obs, col = "red", type = "l")
  lines(fitted(mod), x = obs, type = "l", ylab = "fitted values", xlab = "Time")
  plot(resid(mod), x = obs, type = "l", ylab = "residuals", xlab = "Time")
  abline(h = 0, col = "red")
  }

plotmod(m1)
```
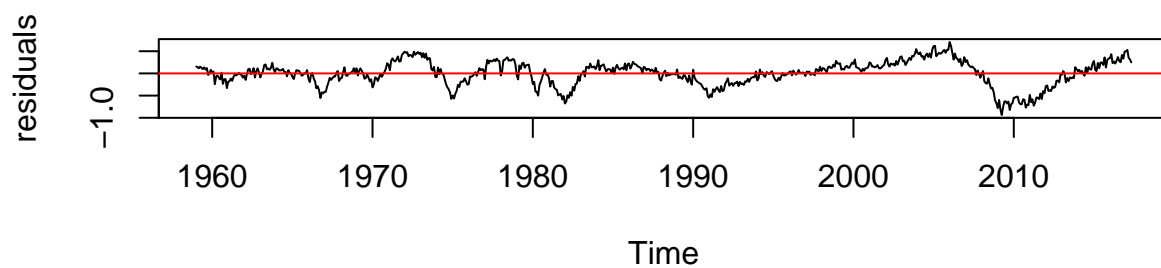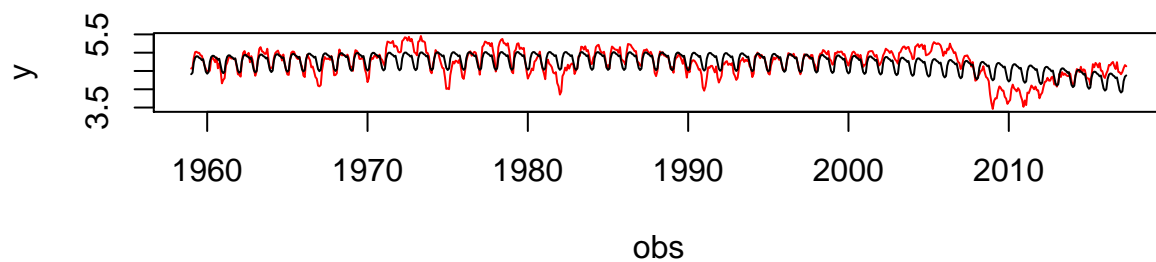
```r
plotmod(m2)
```



```r
plotmod(m3)
```

```
plotmod(m4)
```



```
sum((fitted(m3)-fitted(m4))^2)
```

```
## [1] 0.08851392
```

The residuals of an AR(2) model, found by choosing the opitmal lag-structure according to the BIC, look fairly evenly distributed around 0 making them closer to a white noise process. This indicates that an AR(2) takes out most of the systematic information in the de-trended, seasonally adjusted data. The fitted values closely resemble the actual data.

```
armod <- auto.arima(resid(m3), ic ="bic")
summary(armod)
```

```
## Series: resid(m3)
## ARIMA(2,0,0) with zero mean
##
## Coefficients:
##          ar1     ar2
##       0.6939  0.2620
## s.e.  0.0364  0.0364
##
## sigma^2 estimated as 0.008206:  log likelihood=688.61
## AIC=-1371.22   AICc=-1371.19   BIC=-1357.56
##
## Training set error measures:
##                         ME       RMSE        MAE      MPE      MAPE
## Training set 0.0002409101 0.09045513 0.07060001 70.39446 205.2784
##                   MASE       ACF1
## Training set 0.9510421 -0.0249441
```

```
plotmod(armod)
```

```
abline(h=0, col = "red")
```