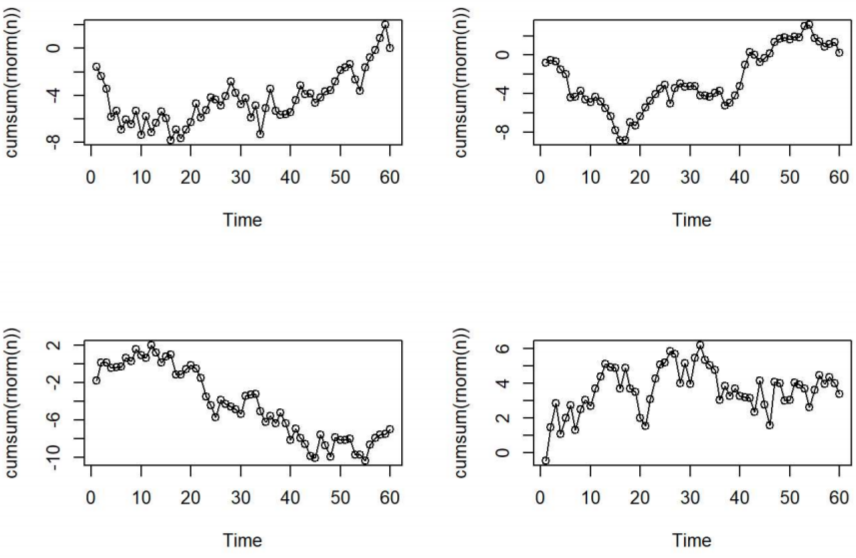
第三章：趋势

3.1 随机趋势

- 受随机扰动的影响，不随时间衰减；
- 在不同的模拟中，可能展现完全不同的趋势。

```
n=60;
opar=par(mfrow=c(2,2))
plot(cumsum(rnorm(n)),type='o',xlab='Time');
plot(cumsum(rnorm(n)),type='o',xlab='Time');
plot(cumsum(rnorm(n)),type='o',xlab='Time');
plot(cumsum(rnorm(n)),type='o',xlab='Time');
```

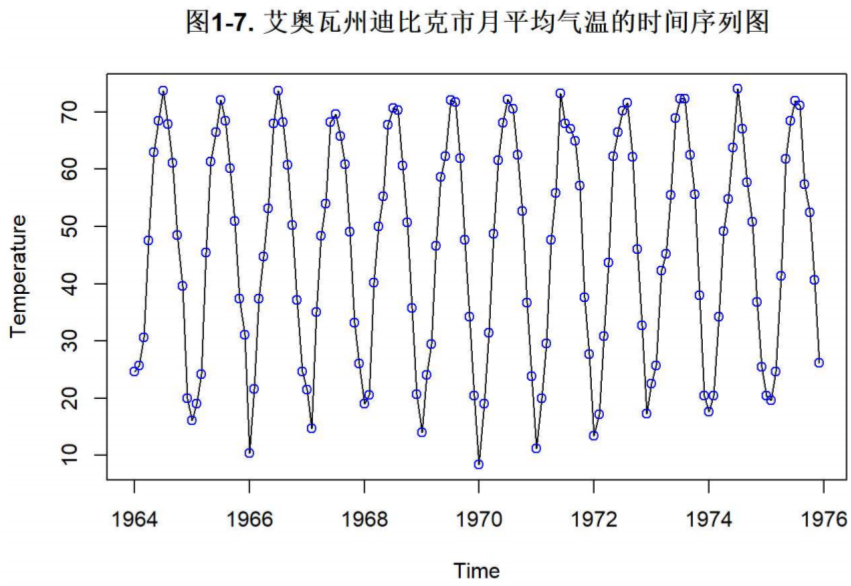



```
par(opar)
```

3.2 确定趋势

例4. 艾奥瓦州迪比克市月平均气温的时间序列图。

```
data(tempdub)
plot(tempdub, ylab='Temperature', type='l', main='图1-7. 艾奥瓦州迪比克市月平均气温的时间序列图')
points(tempdub, col='blue')
```



均值模型： $Y_t = \mu_t + X_t$，其中$E(X_t) = 0$.

- 随机扰动X_t是零均值的平稳序列；
- 均值过程μ_t随着时间t的变化，表现出特定的变化模式：

$$\mu_t = \begin{cases} \mu_0, & \text{常函数} \\ \beta_0 + \beta_1 t, & \text{线性函数} \\ \beta_0 + \beta_1 t + \beta_2 t^2, & \text{二次函数} \\ \cdots, & \text{多项式函数} \\ \mu_{t-T}, & \text{周期函数} \end{cases}$$

3.3 均值函数为常数

$$\hat{\mu} = \bar{Y}, \qquad E(\hat{\mu}) = \mu_0, \qquad \text{无偏估计量}$$

- X_t是平稳的时间序列，自协方差函数为γ_k, 自相关函数为ρ_k.

$$Var(\hat{\mu}) = \frac{\gamma_0}{n}[1 + 2\sum_{k=1}^{n-1}(1 - \frac{k}{n})\rho_k].$$

例1'. **白噪声序列** $\{X_t = e_t, i.i.d.\}$, 平稳的, $Var(\hat{\mu}) = \frac{\gamma_0}{n}$.

$$\gamma_k = \begin{cases} \sigma_e^2, & \text{if } k = 0 \\ 0, & \text{if } k > 0 \end{cases} \qquad\qquad \rho_k = \begin{cases} 1, & \text{if } k = 0 \\ 0, & \text{if } k > 0 \end{cases}$$

例2'. **滑动平均序列**$\{X_t = \frac{1}{2}e_t + \frac{1}{2}e_{t-1}\}$, 平稳的, $Var(\hat{\mu}) = \frac{\gamma_0}{n}[1 + \frac{n-1}{n}] = \frac{\gamma_0}{n}(\frac{2n-1}{n}) \simeq 2\frac{\gamma_0}{n}$.

$$\gamma_k = \begin{cases} 0.5 \cdot \sigma_e^2, & \text{if } k = 0 \\ 0.25 \cdot \sigma_e^2, & \text{if } k = 1 \\ 0, & \text{if } k > 1 \end{cases} \qquad\qquad \rho_k = \begin{cases} 1, & \text{if } k = 0 \\ 0.5, & \text{if } k = 1 \\ 0, & \text{if } k > 1 \end{cases}$$

滑动平均序列 $\{X_t = e_t - \frac{1}{2}e_{t-1}\}$, 平稳的, $Var(\hat{\mu}) = \frac{\gamma_0}{n}[1 - 0.8(\frac{n-1}{n})] = \frac{\gamma_0}{n}(\frac{0.2n+0.8}{n}) \simeq 0.2\frac{\gamma_0}{n}$.

$$\gamma_k = \begin{cases} 1.25 \cdot \sigma_e^2, & \text{if } k = 0 \\ -0.5 \cdot \sigma_e^2, & \text{if } k = 1 \\ 0, & \text{if } k > 1 \end{cases} \qquad\qquad \rho_k = \begin{cases} 1, & \text{if } k = 0 \\ -0.4, & \text{if } k = 1 \\ 0, & \text{if } k > 1 \end{cases}$$

例5. 一般的平稳序列，随着滞后的增加，**自相关函数迅速衰减，** $\sum_{k=0}^{+\infty}|\rho_k| < \infty$. 当$n$充分大的时候，$Var(\hat{\mu}) \simeq \frac{\gamma_0}{n}[\sum_{k=0}^{+\infty}\rho_k]$.

例如，对于所有的整数k, $\rho_k = \phi^{|k|}, \phi \in (-1, +1)$. 那么当$n$充分大的时候，$Var(\hat{\mu}) \simeq \frac{\gamma_0}{n}[1 + \frac{2\phi}{1-\phi}] = \frac{(1+\phi)\gamma_0}{(1-\phi)n}$.

- X_t是非平稳的时间序列，自协方差函数为$\gamma_{t,s}$, 自相关函数为ρ_t, s.

$$Var(\hat{\mu}) = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}Cov(X_i, X_j) = \frac{1}{n^2}[\sum_{i=1}^{n}Var(X_i) + 2\sum_{i=2}^{n}\sum_{j=1}^{i-1}\gamma_{i,j}]$$

例3'. **随机游动序列** $\{X_t = e_1 + e_2 + \cdots + e_t\}$, ***非平稳的***, $\mu_t = t \cdot \mu, \quad \sigma_t^2 = t \cdot \sigma_e^2$.

$$\begin{aligned}\gamma_{t,s} &= Cov(e_1 + e_2 + \cdots + e_t, e_1 + e_2 + \cdots + e_s) \\ &= Var(e_1 + e_2 + \cdots + e_{min(t,s)}) \\ &= min(t,s) \cdot \sigma_e^2\end{aligned}$$

$$\rho_{t,s} = \frac{\gamma_{t,s}}{\sqrt{\gamma_{t,t}\gamma_{s,s}}} = \frac{min(t,s)\cdot\sigma_e^2}{\sqrt{t\sigma_e^2 \cdot s\sigma_e^2}} = \frac{\sqrt{min(t,s)}}{\sqrt{max(t,s)}}$$

$$\begin{aligned}Var(\hat{\mu}) &= \frac{1}{n^2}\left[\sum_{i=1}^{n} i\sigma_e^2 + 2\sum_{i=2}^{n}\sum_{j=1}^{i-1} j\sigma_e^2\right] \\ &= \frac{\sigma_e^2}{n^2}\left[\sum_{i=1}^{n} i + 2\sum_{j=1}^{n-1}\sum_{i=j+1}^{n} j\right] \\ &= \frac{\sigma_e^2}{n^2}\left[\sum_{i=1}^{n} i + 2\sum_{j=1}^{n-1}(n-j)j\right] \\ &= \frac{\sigma_e^2}{n^2}\left[(1+2n)\sum_{i=1}^{n} i - 2\sum_{j=1}^{n} j^2\right] \\ &= \frac{\sigma_e^2}{n^2}\left[(1+2n)\frac{n(n+1)}{2} - 2\frac{n(n+1)(2n+1)}{6}\right] \\ &= \sigma_e^2(2n+1)\frac{(n+1)}{6n}\end{aligned}$$

3.4 回归方法

$\{X_t\}$满足Gauss-Markov假设(零均值，等方差，不相关).

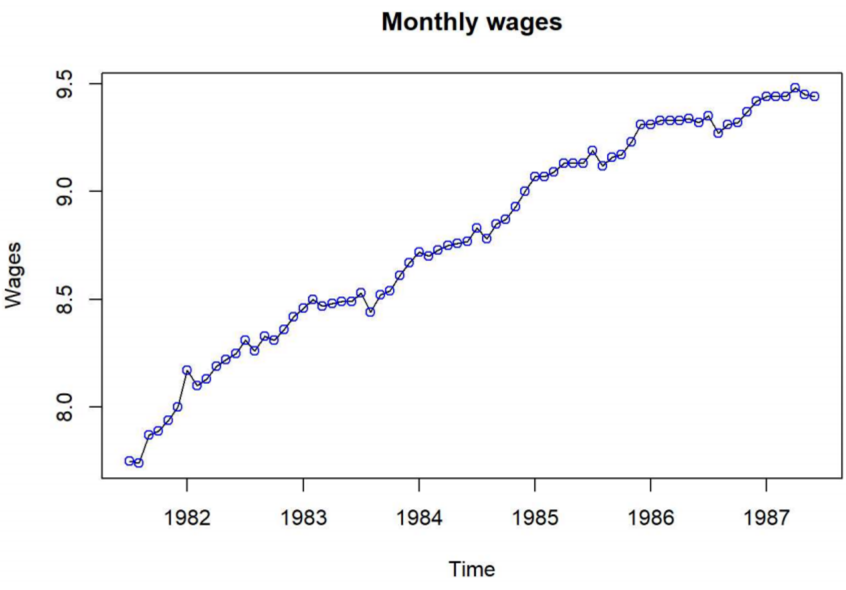
- 时间的线性趋势 $\mu_t = \beta_0 + \beta_1 t$,

  最小二乘法估计：最小化 $Q(\beta_0, \beta_1) = \sum_{t=1}^{n}[Y_t - (\beta_0 + \beta_1 t)]^2$, 令偏导数为0，求解。

$$\begin{aligned}\hat{\beta}_1 &= \frac{\sum_{t=1}^{n} t(Y_t - \bar{Y})}{\sum_{t=1}^{n}(t-\bar{t})^2} = \frac{\sum_{t=1}^{n}(t-\bar{t})(Y_t - \bar{Y})}{\sum_{t=1}^{n}(t-\bar{t})^2} \\ \hat{\beta}_0 &= \bar{Y} - \hat{\beta}_1\bar{t}\end{aligned}$$

- 时间的二次函数 $\mu_t = \beta_0 + \beta_1 t + \beta_2 t^2$

```
data(wages)
plot(wages,type='l',ylab='Wages',xlab='Time',main='Monthly wages')
points(wages,col='blue')
```

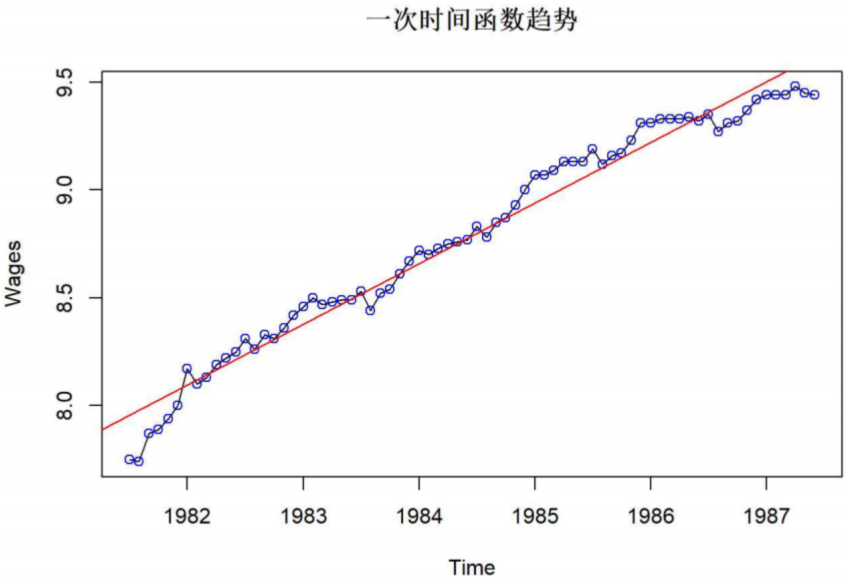


时间的一次函数拟合

```
lm_wages=lm(wages~time(wages))
summary(lm_wages)
```

```
##
## Call:
## lm(formula = wages ~ time(wages))
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -0.23828 -0.04981  0.01942  0.05845  0.13136
##
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept) -5.490e+02  1.115e+01  -49.24   <2e-16 ***
## time(wages)  2.811e-01  5.618e-03   50.03   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.08257 on 70 degrees of freedom
## Multiple R-squared:  0.9728, Adjusted R-squared:  0.9724
## F-statistic:  2503 on 1 and 70 DF,  p-value: < 2.2e-16
```

```
plot(wages,type='l',ylab='Wages',xlab='Time',main='一次时间函数趋势')
points(wages,col='blue')
abline(lm_wages,col='red')
```



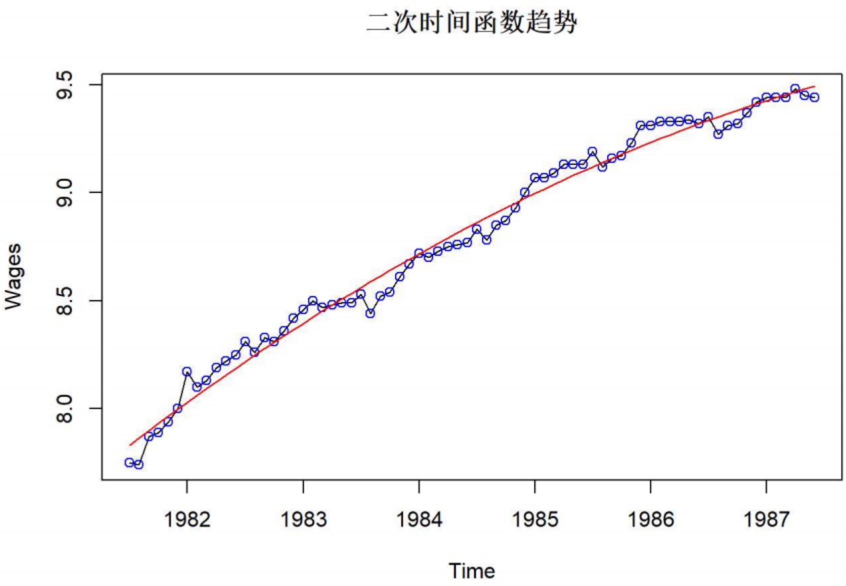
最小二乘估计的斜率$\hat{\beta}_1 =$ 0.2811和截距 $\hat{\beta}_0 =$ -549 . 图表3-2展示了随机游动并叠加了最小二乘回归趋势线。

二次时间函数拟合

```
lm2_wages=lm(wages~time(wages)+I(time(wages)^2))
summary(lm2_wages)
```

```
##
## Call:
## lm(formula = wages ~ time(wages) + I(time(wages)^2))
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.148318 -0.041440  0.001563  0.050089  0.139839
##
## Coefficients:
##                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)      -8.495e+04  1.019e+04  -8.336 4.87e-12 ***
## time(wages)       8.534e+01  1.027e+01   8.309 5.44e-12 ***
## I(time(wages)^2) -2.143e-02  2.588e-03  -8.282 6.10e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05889 on 69 degrees of freedom
## Multiple R-squared:  0.9864, Adjusted R-squared:  0.986
## F-statistic:  2494 on 2 and 69 DF,  p-value: < 2.2e-16
```

```
xfit <- time(wages)
yfit <- fitted(lm2_wages)
plot(wages,type='l',ylab='Wages',xlab='Time',main='二次时间函数趋势')
points(wages,col='blue')
lines(as.vector(xfit),as.vector(yfit),col='red')
```



- 周期性或季节性趋势

```
data(tempdub)
month.=season(tempdub)
model2=lm(tempdub~month.-1) # -1 removes the intercept term
summary(model2)
```

```
##
## Call:
## lm(formula = tempdub ~ month. - 1)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2750 -2.2479  0.1125  1.8896  9.8250
##
## Coefficients:
##                  Estimate Std. Error t value Pr(>|t|)
## month.January      16.608      0.987   16.83   <2e-16 ***
## month.February     20.650      0.987   20.92   <2e-16 ***
## month.March        32.475      0.987   32.90   <2e-16 ***
## month.April        46.525      0.987   47.14   <2e-16 ***
## month.May          58.092      0.987   58.86   <2e-16 ***
## month.June         67.500      0.987   68.39   <2e-16 ***
## month.July         71.717      0.987   72.66   <2e-16 ***
## month.August       69.333      0.987   70.25   <2e-16 ***
## month.September    61.025      0.987   61.83   <2e-16 ***
## month.October      50.975      0.987   51.65   <2e-16 ***
## month.November     36.650      0.987   37.13   <2e-16 ***
## month.December     23.642      0.987   23.95   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.419 on 132 degrees of freedom
## Multiple R-squared:  0.9957, Adjusted R-squared:  0.9953
## F-statistic:  2569 on 12 and 132 DF,  p-value: < 2.2e-16
```

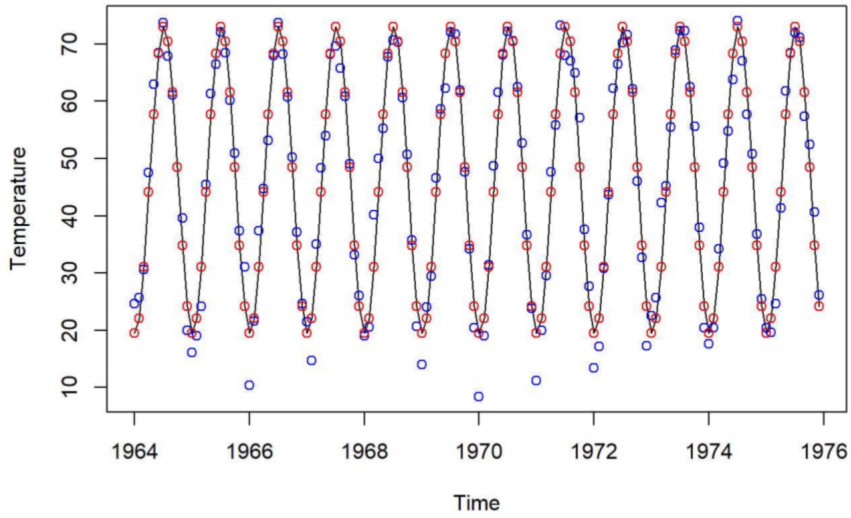
```
model3=lm(tempdub~month.) # intercept is automatically included so one month (Jan) is dropped
summary(model3)
```

```
##
## Call:
## lm(formula = tempdub ~ month.)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.2750 -2.2479  0.1125  1.8896  9.8250
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)       16.608      0.987  16.828  < 2e-16 ***
## month.February     4.042      1.396   2.896  0.00443 **
## month.March       15.867      1.396  11.368  < 2e-16 ***
## month.April       29.917      1.396  21.434  < 2e-16 ***
## month.May         41.483      1.396  29.721  < 2e-16 ***
## month.June        50.892      1.396  36.461  < 2e-16 ***
## month.July        55.108      1.396  39.482  < 2e-16 ***
## month.August      52.725      1.396  37.775  < 2e-16 ***
## month.September   44.417      1.396  31.822  < 2e-16 ***
## month.October     34.367      1.396  24.622  < 2e-16 ***
## month.November    20.042      1.396  14.359  < 2e-16 ***
## month.December     7.033      1.396   5.039 1.51e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.419 on 132 degrees of freedom
## Multiple R-squared:  0.9712, Adjusted R-squared:  0.9688
## F-statistic: 405.1 on 11 and 132 DF,  p-value: < 2.2e-16
```

```
# first creates the first pair of harmonic functions and then fit the model
har.=harmonic(tempdub,1)##a matrix consisting of \cos(2k ?? t), \sin(2k ?? t), k=1,2,...,m, excluding any zero functions.
model4=lm(tempdub~har.)
summary(model4)
```

```
##
## Call:
## lm(formula = tempdub ~ har.)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -11.1580  -2.2756  -0.1457   2.3754  11.2671
##
## Coefficients:
##                 Estimate Std. Error t value Pr(>|t|)
## (Intercept)      46.2660     0.3088 149.816  < 2e-16 ***
## har.cos(2*pi*t) -26.7079     0.4367 -61.154  < 2e-16 ***
## har.sin(2*pi*t)  -2.1697     0.4367  -4.968 1.93e-06 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3.706 on 141 degrees of freedom
## Multiple R-squared:  0.9639, Adjusted R-squared:  0.9634
## F-statistic:  1882 on 2 and 141 DF,  p-value: < 2.2e-16
```

```
tempFit4 <- ts(fitted(model4),freq=12,start=c(1964,1))
plot(tempFit4,ylab='Temperature',type='l',
ylim=range(c(fitted(model4),tempdub))) # the ylim option ensures that the
# y axis has a range that fits the raw data and the fitted values
points(tempdub, col='blue')
points(tempFit4,col='red')
```


Temperature
Time
10
20
30
40
50
60
70
1964
1966
1968
1970
1972
1974
1976