

# Forecasting: principles and practice

Rob J Hyndman

2.5 Seasonal ARIMA models

# Outline

**1** Seasonal ARIMA models

**2** Lab session 17

**3** ARIMA vs ETS

**4** Lab session 18

# Seasonal ARIMA models

ARIMA	$\underbrace{(p, d, q)}$	$\underbrace{(P, D, Q)_m}$
	↑	↑
	Non-seasonal part of the model	Seasonal part of of the model

where  $m$  = number of observations per year.

# Seasonal ARIMA models

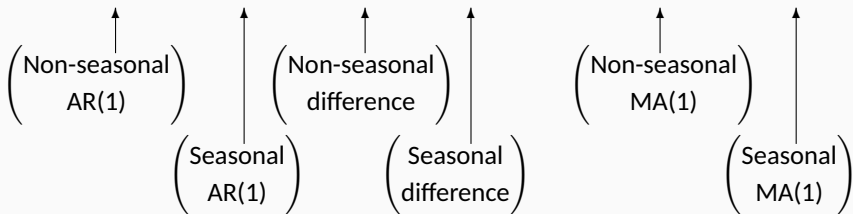
E.g.,  $\text{ARIMA}(1, 1, 1)(1, 1, 1)_4$  model (without constant)

# Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)<sub>4</sub> model (without constant)  
 $(1 - \phi_1 B)(1 - \Phi_1 B^4)(1 - B)(1 - B^4)y_t = (1 + \theta_1 B)(1 + \Theta_1 B^4)\varepsilon_t.$

# Seasonal ARIMA models

E.g.,  $\text{ARIMA}(1, 1, 1)(1, 1, 1)_4$  model (without constant)  
 $(1 - \phi_1 B)(1 - \Phi_1 B^4)(1 - B)(1 - B^4)y_t = (1 + \theta_1 B)(1 + \Theta_1 B^4)\varepsilon_t.$



# Seasonal ARIMA models

E.g., ARIMA(1, 1, 1)(1, 1, 1)<sub>4</sub> model (without constant)  
 $(1 - \phi_1 B)(1 - \Phi_1 B^4)(1 - B)(1 - B^4)y_t = (1 + \theta_1 B)(1 + \Theta_1 B^4)\varepsilon_t.$

All the factors can be multiplied out and the general model written as follows:

$$\begin{aligned} y_t = & (1 + \phi_1)y_{t-1} - \phi_1 y_{t-2} + (1 + \Phi_1)y_{t-4} \\ & - (1 + \phi_1 + \Phi_1 + \phi_1 \Phi_1)y_{t-5} + (\phi_1 + \phi_1 \Phi_1)y_{t-6} \\ & - \Phi_1 y_{t-8} + (\Phi_1 + \phi_1 \Phi_1)y_{t-9} - \phi_1 \Phi_1 y_{t-10} \\ & + \varepsilon_t + \theta_1 \varepsilon_{t-1} + \Theta_1 \varepsilon_{t-4} + \theta_1 \Theta_1 \varepsilon_{t-5}. \end{aligned}$$

# Common ARIMA models

The US Census Bureau uses the following models most often:

ARIMA(0,1,1)(0,1,1)<sub>m</sub> with log transformation

ARIMA(0,1,2)(0,1,1)<sub>m</sub> with log transformation

ARIMA(2,1,0)(0,1,1)<sub>m</sub> with log transformation

ARIMA(0,2,2)(0,1,1)<sub>m</sub> with log transformation

ARIMA(2,1,2)(0,1,1)<sub>m</sub> with no transformation



# Understanding ARIMA models

## Long-term forecasts

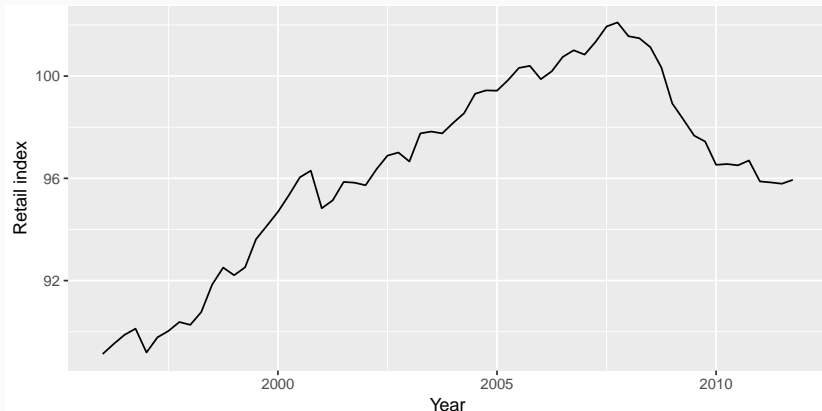
zero	$c = 0, d + D = 0$	
non-zero constant	$c = 0, d + D = 1$	$c \neq 0, d + D = 0$
linear	$c = 0, d + D = 2$	$c \neq 0, d + D = 1$
quadratic	$c = 0, d + D = 3$	$c \neq 0, d + D = 2$

## Forecast variance and $d + D$

- The higher the value of  $d + D$ , the more rapidly the prediction intervals increase in size.
- For  $d + D = 0$ , the long-term forecast standard deviation will go to the standard deviation of the historical data.

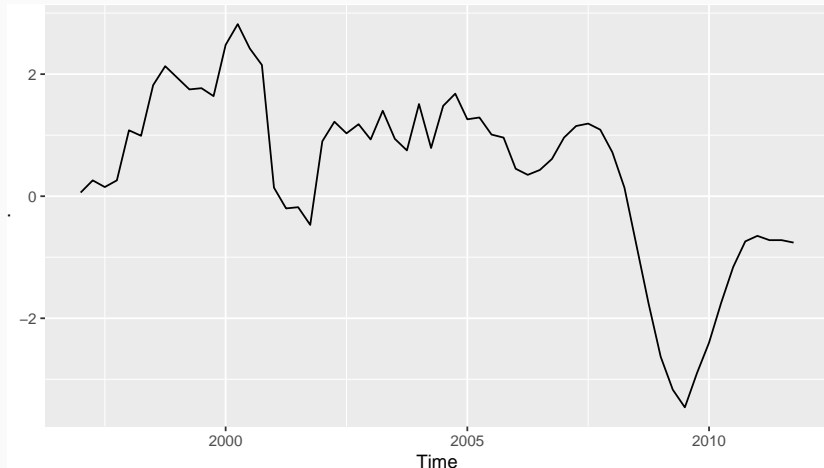
# European quarterly retail trade

```
autoplot(euretail) +  
  xlab("Year") + ylab("Retail index")
```



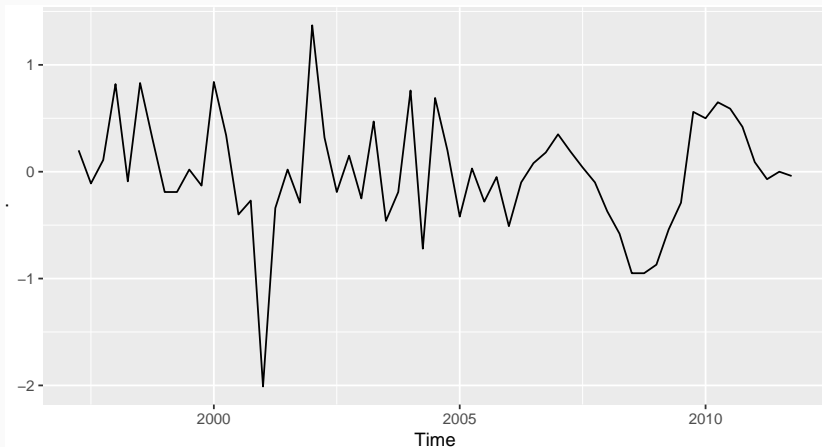
# European quarterly retail trade

```
euretail %>% diff(lag=4) %>% autoplot()
```



# European quarterly retail trade

```
euretail %>% diff(lag=4) %>% diff() %>%  
autoplot()
```



# European quarterly retail trade

```
(fit <- auto.arima(euretail))
```

```
## Series: euretail
## ARIMA(1,1,2)(0,1,1)[4]
##
## Coefficients:
##          ar1      ma1      ma2      sma1
##          0.736   -0.466   0.216   -0.843
## s.e.    0.224    0.199   0.210    0.188
##
## sigma^2 estimated as 0.159:  log likelihood=-29.62
## AIC=69.24   AICc=70.38   BIC=79.63
```

# European quarterly retail trade

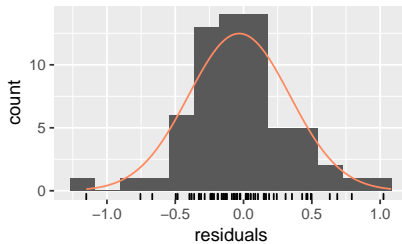
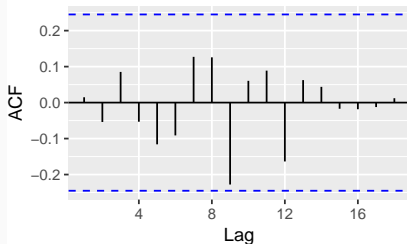
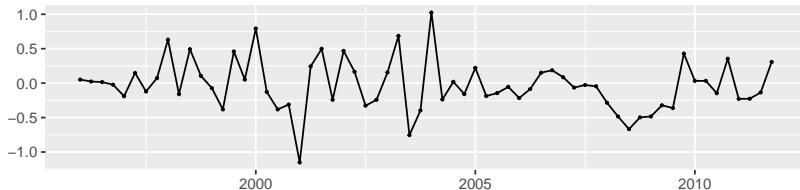
```
(fit <- auto.arima(euretail, stepwise=TRUE,  
  approximation=FALSE))
```

```
## Series: euretail  
## ARIMA(1,1,2)(0,1,1)[4]  
##  
## Coefficients:  
##          ar1      ma1      ma2      sma1  
##      0.736  -0.466   0.216  -0.843  
## s.e.  0.224    0.199   0.210   0.188  
##  
## sigma^2 estimated as 0.159:  log likelihood=-29.62  
## AIC=69.24   AICc=70.38   BIC=79.63
```

# European quarterly retail trade

```
checkresiduals(fit, test=FALSE)
```

Residuals from ARIMA(1,1,2)(0,1,1)[4]



# European quarterly retail trade

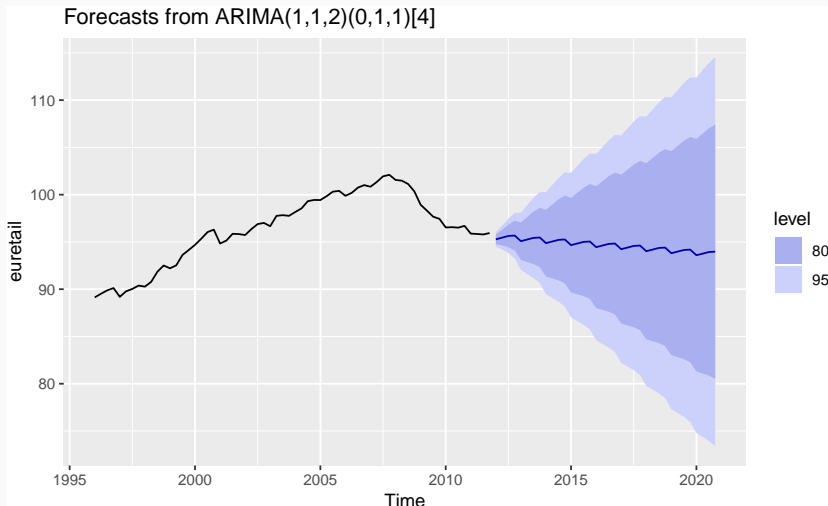
```
checkresiduals(fit, plot=FALSE)
```

```
##  
##  Ljung-Box test  
##  
## data:  Residuals from ARIMA(1,1,2)(0,1,1)[4]  
## Q* = 4.9, df = 4, p-value = 0.3  
##  
## Model df: 4.    Total lags used: 8
```

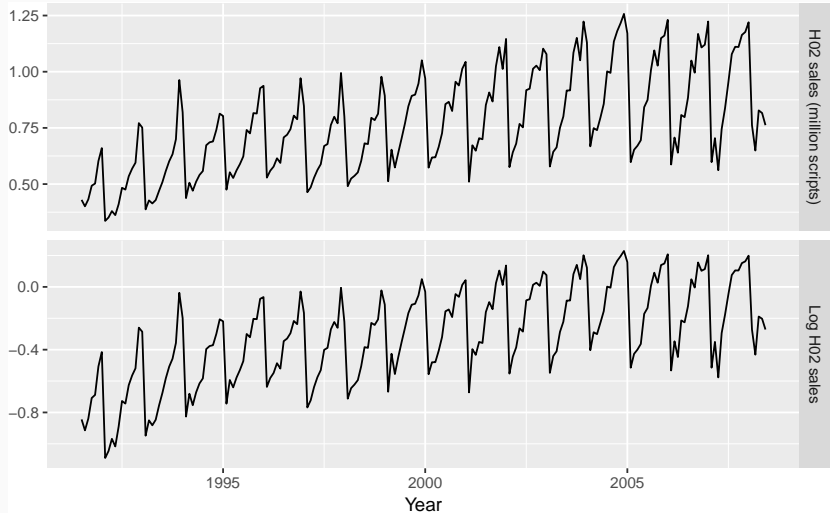


# European quarterly retail trade

```
forecast(fit, h=36) %>% autoplot()
```

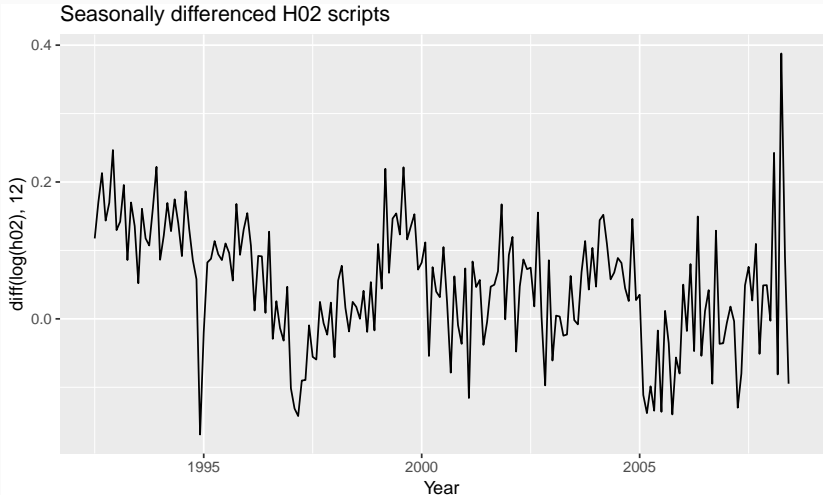


# Corticosteroid drug sales



# Corticosteroid drug sales

```
autoplot(diff(log(h02),12), xlab="Year",  
main="Seasonally differenced H02 scripts")
```



# Corticosteroid drug sales

```
(fit <- auto.arima(h02, lambda=0, max.order=9,  
  stepwise=FALSE, approximation=FALSE))
```

```
## Series: h02
```

```
## ARIMA(4,1,1)(2,1,2)[12]
```

```
## Box Cox transformation: lambda= 0
```

```
##
```

```
## Coefficients:
```

```
##          ar1      ar2      ar3      ar4      ma1      sar1
```

```
##          -0.042  0.210  0.202  -0.227  -0.742  0.621
```

```
## s.e.      0.217  0.181  0.114   0.081   0.207  0.242
```

```
##          sar2      sma1      sma2
```

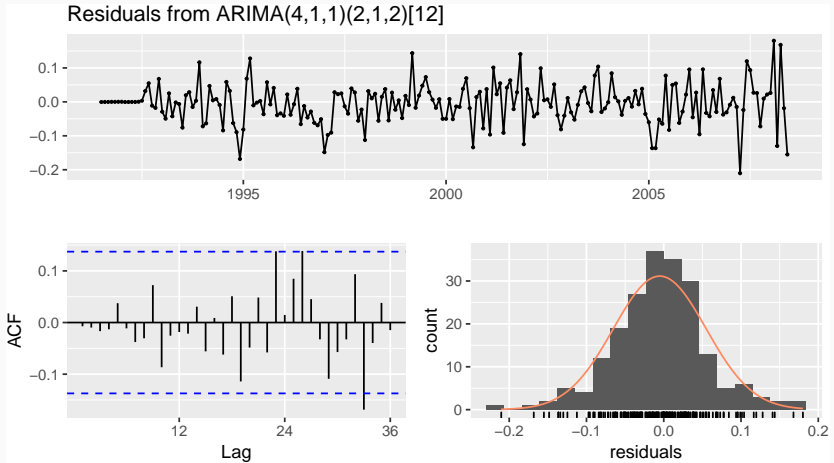
```
##          -0.383  -1.202  0.496
```

```
## s.e.      0.118   0.249  0.214
```

```
##
```

# Corticosteroid drug sales

`checkresiduals(fit)`



# Corticosteroid drug sales

```
##  
##  Ljung-Box test  
##  
## data:  Residuals from ARIMA(4,1,1)(2,1,2)[12]  
## Q* = 16, df = 15, p-value = 0.4  
##  
## Model df: 9.    Total lags used: 24
```

# Corticosteroid drug sales

Training data: July 1991 to June 2006

Test data: July 2006–June 2008

```
getrmse <- function(x,h,...)
{
  train.end <- time(x)[length(x)-h]
  test.start <- time(x)[length(x)-h+1]
  train <- window(x,end=train.end)
  test <- window(x,start=test.start)
  fit <- Arima(train,...)
  fc <- forecast(fit,h=h)
  return(accuracy(fc,test)[2,"RMSE"])
}

getrmse(h02,h=24,order=c(3,0,0),seasonal=c(2,1,0),lambda=0)
getrmse(h02,h=24,order=c(3,0,1),seasonal=c(2,1,0),lambda=0)
getrmse(h02,h=24,order=c(3,0,2),seasonal=c(2,1,0),lambda=0)
getrmse(h02,h=24,order=c(3,0,1),seasonal=c(1,1,0),lambda=0)
```

# Corticosteroid drug sales

Model	RMSE
ARIMA(4,1,2)(2,1,2)[12]	0.0614
ARIMA(4,1,1)(2,1,2)[12]	0.0615
ARIMA(3,0,1)(0,1,2)[12]	0.0622
ARIMA(3,0,1)(1,1,1)[12]	0.0630
ARIMA(2,1,4)(0,1,1)[12]	0.0632
ARIMA(2,1,3)(0,1,1)[12]	0.0634
ARIMA(4,1,2)(1,1,2)[12]	0.0634
ARIMA(3,1,2)(2,1,2)[12]	0.0636
ARIMA(3,0,3)(0,1,1)[12]	0.0639
ARIMA(2,1,5)(0,1,1)[12]	0.0640
ARIMA(3,0,1)(0,1,1)[12]	0.0644
ARIMA(3,0,2)(0,1,1)[12]	0.0644
ARIMA(3,0,2)(2,1,0)[12]	0.0645

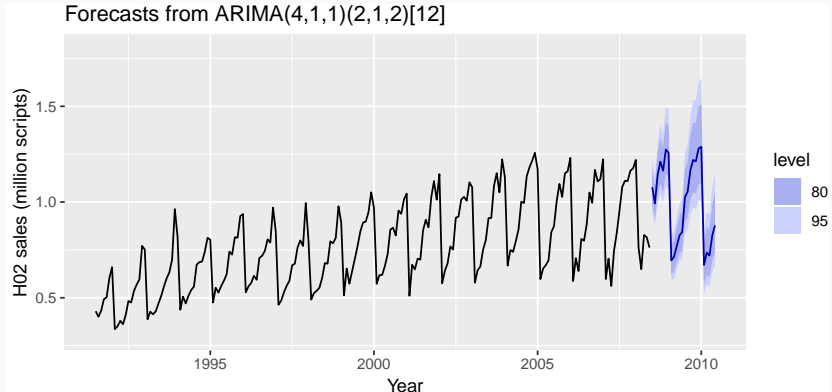


# Corticosteroid drug sales

- Models with lowest AICc values tend to give slightly better results than the other models.
- AICc comparisons must have the same orders of differencing. But RMSE test set comparisons can involve any models.
- Use the best model available, even if it does not pass all tests.

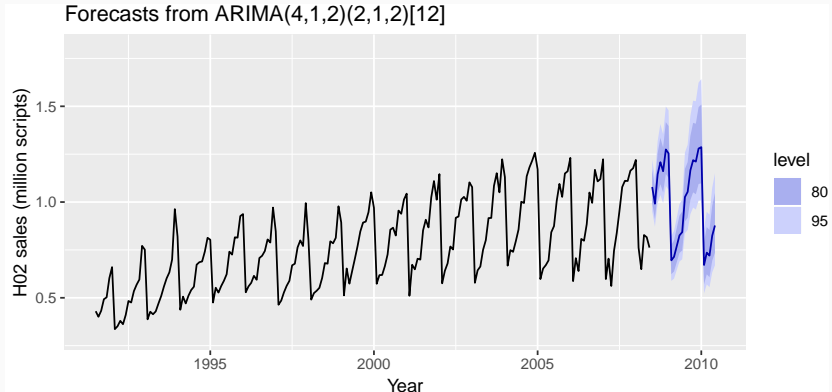
# Corticosteroid drug sales

```
fit <- Arima(h02, order=c(4,1,1), seasonal=c(2,1,2),  
  lambda=0)  
autoplot(forecast(fit)) + xlab("Year") +  
  ylab("H02 sales (million scripts)") + ylim(0.3,1.8)
```



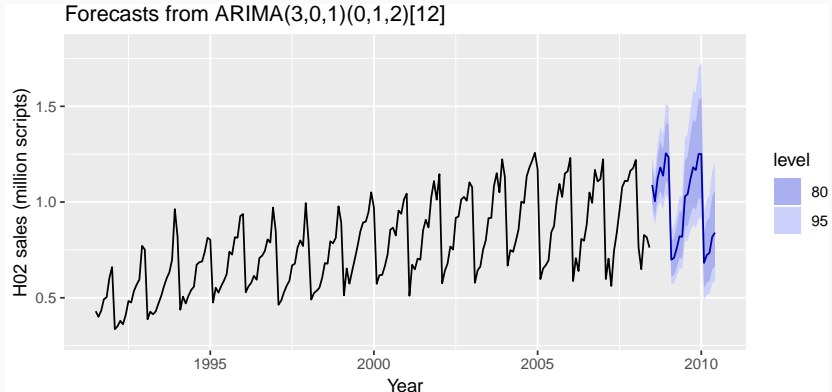
# Corticosteroid drug sales

```
fit <- Arima(h02, order=c(4,1,2), seasonal=c(2,1,2),  
  lambda=0)  
autoplot(forecast(fit)) + xlab("Year") +  
  ylab("H02 sales (million scripts)") + ylim(0.3,1.8)
```



# Corticosteroid drug sales

```
fit <- Arima(h02, order=c(3,0,1), seasonal=c(0,1,2),  
  lambda=0)  
autoplot(forecast(fit)) + xlab("Year") +  
  ylab("H02 sales (million scripts)") + ylim(0.3,1.8)
```



# Outline

**1** Seasonal ARIMA models

**2** Lab session 17

**3** ARIMA vs ETS

**4** Lab session 18

# Lab Session 17

# Outline

**1** Seasonal ARIMA models

**2** Lab session 17

**3** ARIMA vs ETS

**4** Lab session 18

# ARIMA vs ETS

- Myth that ARIMA models are more general than exponential smoothing.
- Linear exponential smoothing models all special cases of ARIMA models.
- Non-linear exponential smoothing models have no equivalent ARIMA counterparts.
- Many ARIMA models have no exponential smoothing counterparts.
- ETS models all non-stationary. Models with seasonality or non-damped trend (or both) have two unit roots; all other models have one unit root.



# Equivalences

ETS model	ARIMA model	Parameters
ETS(A,N,N)	ARIMA(0,1,1)	$\theta_1 = \alpha - 1$
ETS(A,A,N)	ARIMA(0,2,2)	$\theta_1 = \alpha + \beta - 2$ $\theta_2 = 1 - \alpha$
ETS(A,A,N)	ARIMA(1,1,2)	$\phi_1 = \phi$ $\theta_1 = \alpha + \phi\beta - 1 - \phi$ $\theta_2 = (1 - \alpha)\phi$
ETS(A,N,A)	ARIMA(0,0,m)(0,1,0) <sub>m</sub>	
ETS(A,A,A)	ARIMA(0,1,m + 1)(0,1,0) <sub>m</sub>	
ETS(A,A,A)	ARIMA(1,0,m + 1)(0,1,0) <sub>m</sub>	

# Outline

**1** Seasonal ARIMA models

**2** Lab session 17

**3** ARIMA vs ETS

**4** Lab session 18

# Lab Session 18