Project 1: Trending dataset

```r
library(forecast)
library(ggplot2)
```

```r
#70.19999695    71.09999847 71.69999695 72.30000305 73.09999847 72.90000153 74.40000153 75.40000153 76

# Import with scan
singapur = c(70.19999695,71.09999847,71.69999695,72.30000305,73.09999847,72.90000153,74.40000153,75.400
```

```r
singapur = ts(singapur, start = 1980)
```

```r
plot(singapur, ylab="Labour Force Participation Rate 25 - 54")
```


```r
# clearly trending dataset
# Model cannot exceed 100%
```

Potential models There are two models which can be considered:

- Linear Trend Model with holt()
- With Damping Parameter
- Without Damping Parameter
- ARIMA model

Exponential smoothing

Exponential smoothing with library(forecast):

- Simple exponential smoothing: ses()
- Holt's linear trend model: holt() + damped
- Holt-Winters seasonal method: hw()

- Automated exponential smoothing: ets()

ses() can't be used in this case because dataset is trending, Holt-Winters seasonal method can't be used because dataset is not showing a seasonal pattern

How does the Holt Linear Trend Model Work?

Estimated forecast value at t time point Yth/t= *Level value at t time point* (l_t) + *Trend value at t time point mulitpled by h* (hb_t) *(h = number of steps we want to go to future)*

Smoothing Parameters of a Holt Linear Trend Model

Alpha = Smoothing parameter for the level, if only recent data is required for the label value

Beta = tells the modelif only recent data is required for the trend

Gamma is omitted here as there is no seasonality

0 < X < 1

Closer to 0:Smooth model, older data is considered too Closer to 1: Reactive model, heavily relies on recent data

For example, if I have a beta parameter close to 0, then the slope of the trend stays constant, if it's closer to 1, it means that the slope of the trend can change on short notice or just two or three observations

Theoretical Example of holt() Function

library(forecast) holt(data, h=5); forecast legth of 5 years

```
holttrend =  holt(singapur, h=5)
# alpha = 0.6378, beta = 0.121;
# This indicates that the trend which is basically the slope of the time series plot is fairly constans
```

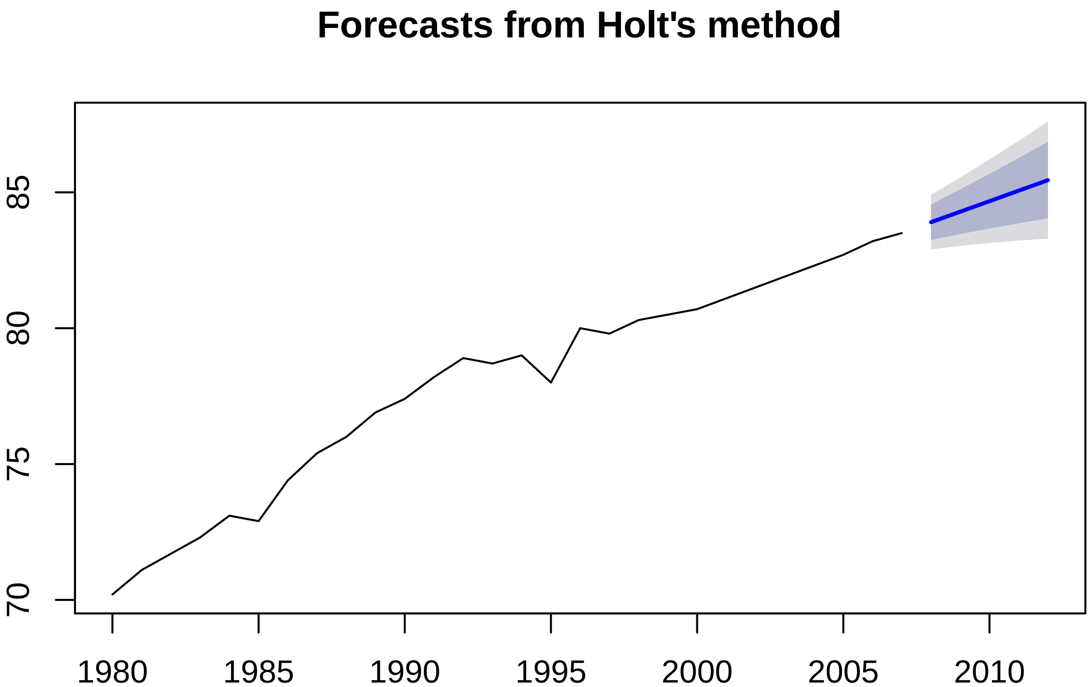
```
summary(holttrend)
```

```
##
## Forecast method: Holt's method
##
## Model Information:
## Holt's method
##
## Call:
##  holt(y = singapur, h = 5)
##
##   Smoothing parameters:
##     alpha = 0.6378
##     beta  = 0.121
##
##   Initial states:
##     l = 69.6189
##     b = 0.6666
##
##   sigma:  0.5119
##
##      AIC     AICc      BIC
## 65.79969 68.52696 72.46071
##
## Error measures:
##                       ME      RMSE       MAE        MPE      MAPE
## Training set -0.08255756 0.5118728 0.3084448 -0.1057487 0.3986115
```

```
##                     MASE        ACF1
## Training set 0.5047278 -0.09187386
##
## Forecasts:
##      Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2008       83.90072 83.24473 84.55671 82.89747 84.90397
## 2009       84.28752 83.46402 85.11103 83.02808 85.54696
## 2010       84.67432 83.66866 85.67999 83.13629 86.21235
## 2011       85.06112 83.86007 86.26218 83.22427 86.89798
## 2012       85.44793 84.03927 86.85658 83.29357 87.60228
```

```
# shows forecasted 5 years from 2008 to 2012
# show 2 confidence intervals (80 and 95%)
```

```
plot(holttrend)
```




```
# blue are forecasted 5 years
# slope of the forecasted period is nearly the same as the slope between 1998 onwards
```

The nature of this particular data contains a classic problem. The trend cannot continue indefinitely.

It is impossible for a labor force participation rate to cross the 100% mark.

This fact needs to be incorporated into the model. This can be done by adding damped argument to holt() function.

$\mathbf{0 < \varphi < 1}$

If the φ is close to 1, it is the same as a standard holt linear model

If the φ is close to 0, than the curve gets flat faily soon

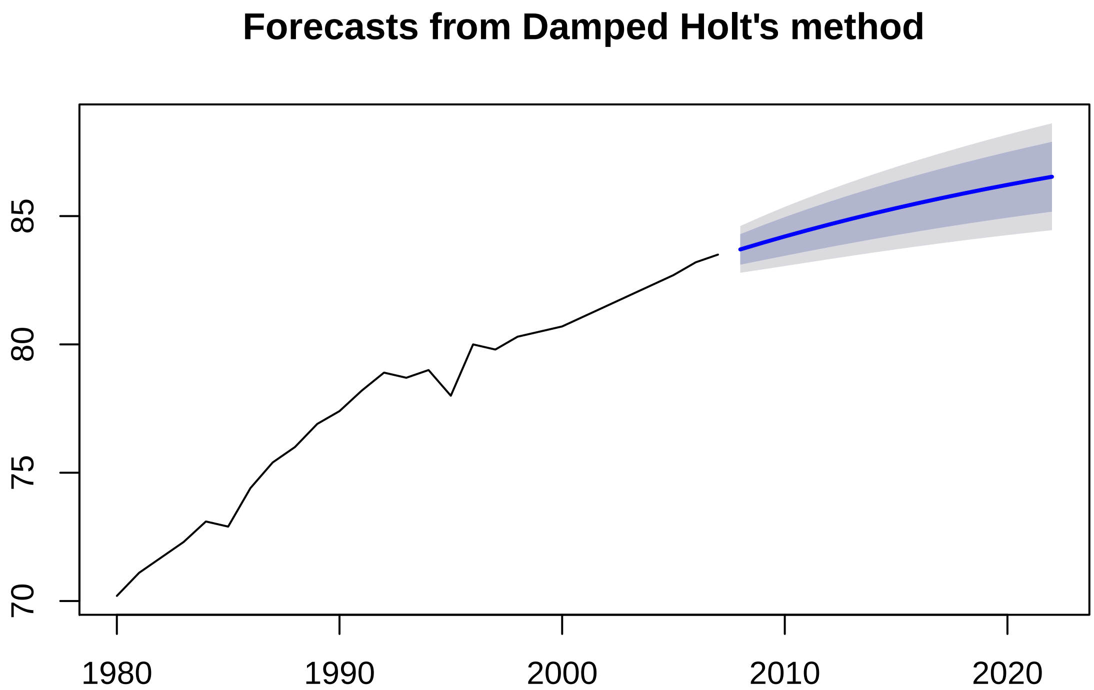
In practice, the parameter φ is set somewhere between 0.8 and 0.95

Damping The Holt Linear Trend Model damped = TRUE

R calculates the value of φ automatically and damps the model accordingly damped =TRUE phi = 0.8

R damps the model by the specified value of φ

```r
# example of holt model with auto generated phi
plot(holt(singapur, h=15, damped=T))
```

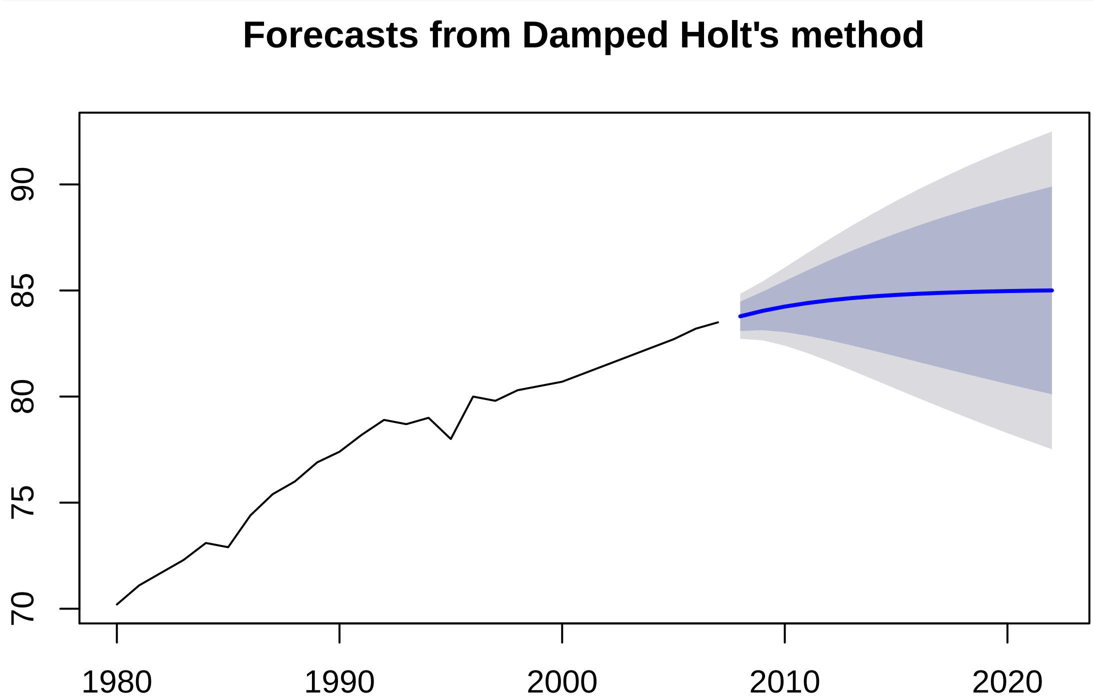



```r
# to see the generated value for phi
summary(holt(singapur, h=15, damped=T))
```

```
##
## Forecast method: Damped Holt's method
##
## Model Information:
## Damped Holt's method
##
## Call:
##  holt(y = singapur, h = 15, damped = T)
##
##   Smoothing parameters:
##     alpha = 0.5492
##     beta  = 1e-04
##     phi   = 0.9616
##
##   Initial states:
##     l = 69.5124
##     b = 0.8338
##
##   sigma:  0.4643
##
##      AIC     AICc      BIC
## 62.33509 66.33509 70.32832
##
## Error measures:
```

```
##                        ME      RMSE       MAE          MPE      MAPE
## Training set 0.000960152 0.4642828 0.3221513 -0.001788087 0.4156845
##                   MASE        ACF1
## Training set 0.5271567 -0.04817038
##
## Forecasts:
##      Point Forecast    Lo 80    Hi 80    Lo 95    Hi 95
## 2008       83.70129 83.10629 84.29629 82.79131 84.61127
## 2009       83.95895 83.28009 84.63780 82.92073 84.99716
## 2010       84.20671 83.45326 84.96016 83.05440 85.35902
## 2011       84.44496 83.62364 85.26629 83.18885 85.70107
## 2012       84.67407 83.79004 85.55809 83.32207 86.02606
## 2013       84.89438 83.95180 85.83695 83.45283 86.33592
## 2014       85.10623 84.10852 86.10394 83.58036 86.63209
## 2015       85.30994 84.25997 86.35991 83.70415 86.91573
## 2016       85.50584 84.40608 86.60560 83.82390 87.18778
## 2017       85.69421 84.54681 86.84162 83.93941 87.44902
## 2018       85.87536 84.68220 87.06852 84.05057 87.70014
## 2019       86.04955 84.81231 87.28678 84.15736 87.94173
## 2020       86.21705 84.93724 87.49685 84.25976 88.17434
## 2021       86.37812 85.05711 87.69912 84.35781 88.39842
## 2022       86.53300 85.17203 87.89398 84.45157 88.61443
```

```r
# example of holt model with manual setting of phi
plot(holt(singapur, h=15, damped=T, phi=0.8))
```



ARIMA model

AR => Autoregressive (Seasonality, trend)

I => Integration (Differencing of the dataset)

MA => Moving Average (Movement around a constant mean)

AR is usually present where trend is in dataset, MA is where the line is flatter

As there is no seasonality, no differencing is needed (I)

ARIMA parameters: p, q, P

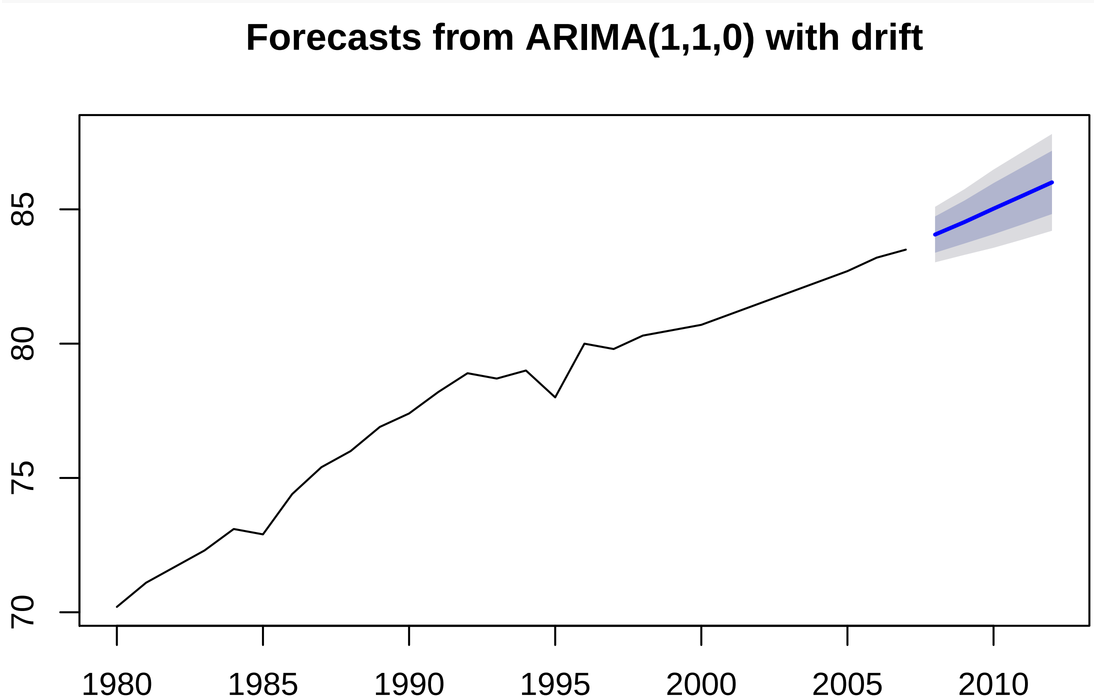
```
# Generate auto arima; dataset should be univariate time series
singapurarima = auto.arima(singapur)
```

```
# summary
summary(singapurarima)
```

```
## Series: singapur
## ARIMA(1,1,0) with drift
##
## Coefficients:
##           ar1   drift
##       -0.3690  0.4904
## s.e.   0.1763  0.0720
##
## sigma^2 estimated as 0.2779:  log likelihood=-20.05
## AIC=46.1   AICc=47.14   BIC=49.99
##
## Training set error measures:
##                       ME      RMSE       MAE        MPE      MAPE
## Training set 0.006855948 0.4981113 0.3755194 0.01821962 0.4863707
##                   MASE       ACF1
## Training set 0.6144862 0.05505323
```

```
# result is ARIMA(1,1,0) with drift
```

```
#plot arima with 5 years forecast
plot(forecast(singapurarima, h=5))
```



```r
# at this stage, the forecast is not ideal because in longer horizont, it will get over 100%
```

```r
auto.arima(singapur, stepwise=F, approximation=F)
```

```
## Series: singapur
## ARIMA(1,1,0) with drift
##
## Coefficients:
##           ar1   drift
##       -0.3690  0.4904
## s.e.   0.1763  0.0720
##
## sigma^2 estimated as 0.2779:  log likelihood=-20.05
## AIC=46.1   AICc=47.14   BIC=49.99
```

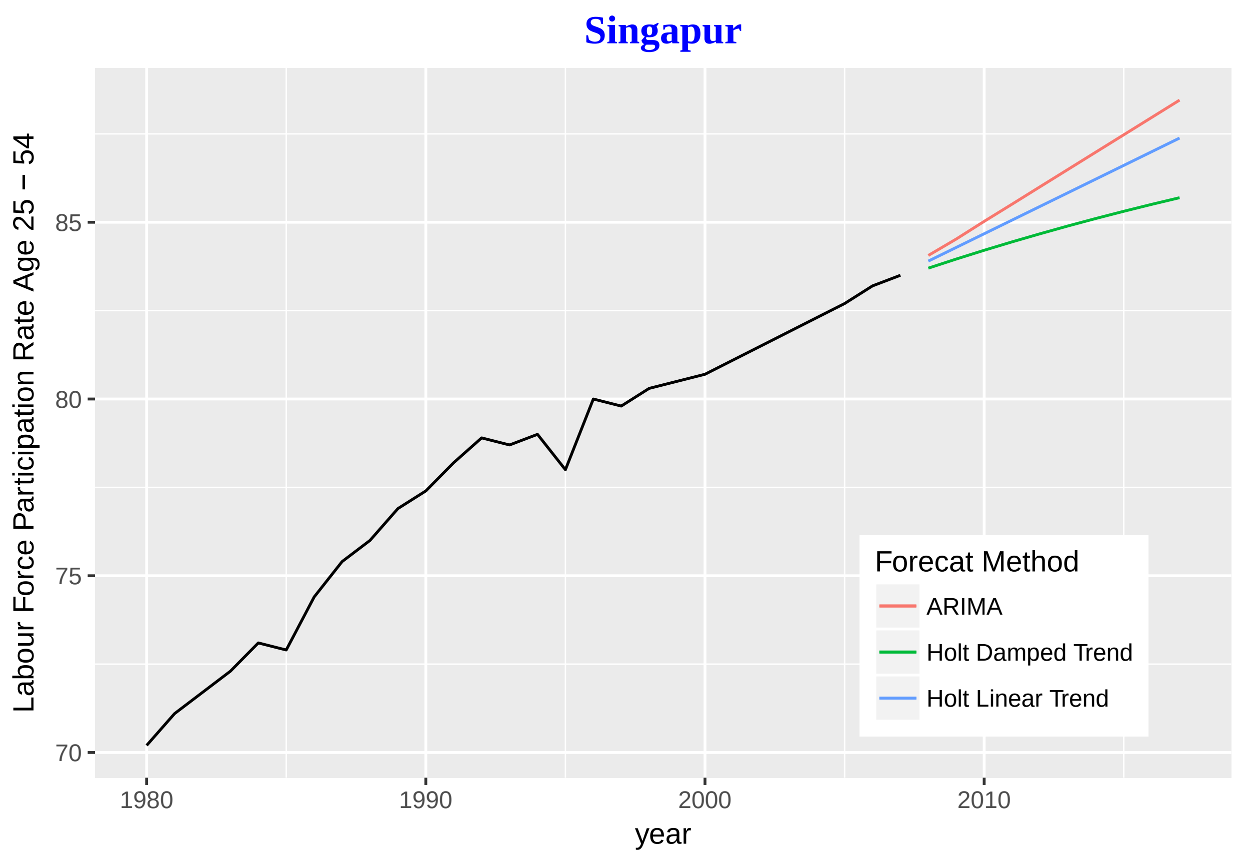
```r
# This method is same as above but takes longer to process, is more accurate and can produce different
```

Coparison Plot

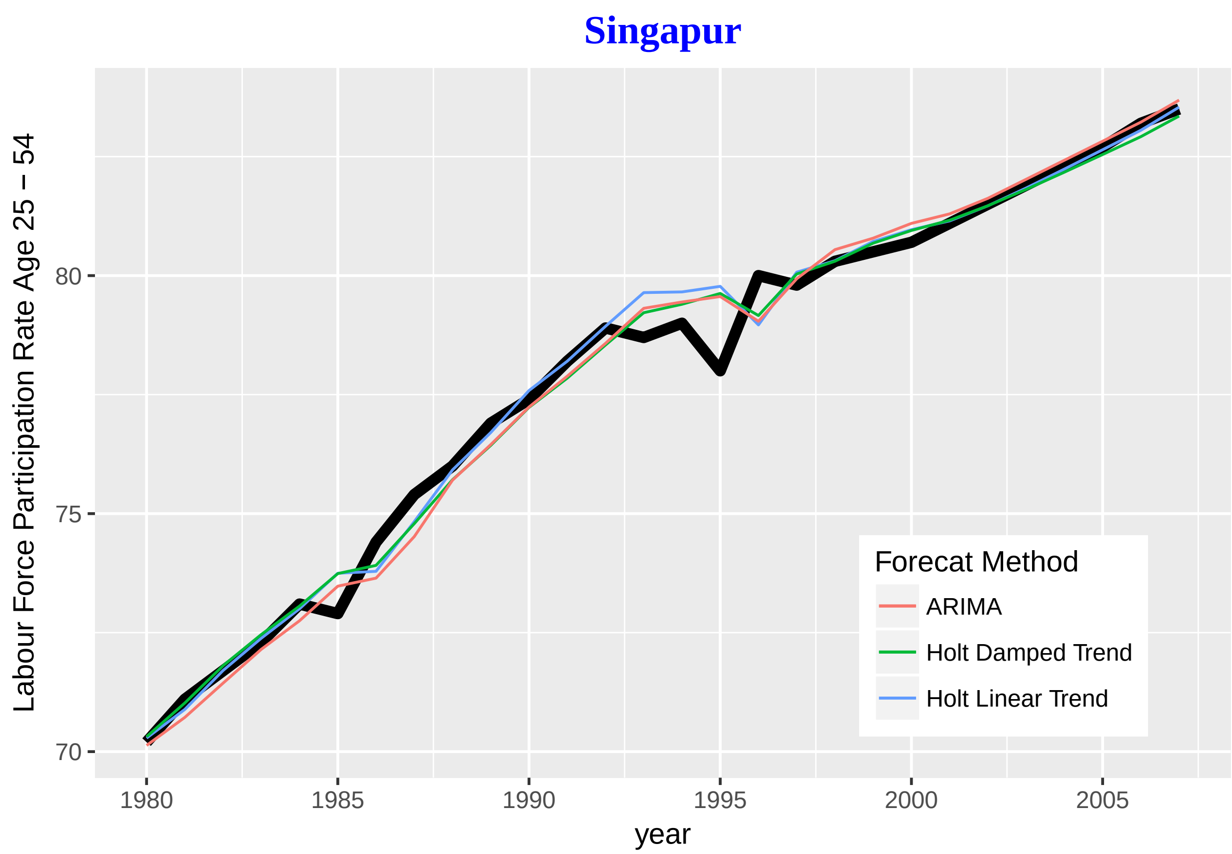
Prepare my three forecast models which I want to plot:

```r
holttrend = holt(singapur, h=10)
holtdamped = holt(singapur, h=10, damped = T)
arimafore = forecast(auto.arima(singapur), h=10)
```

```r
autoplot(singapur)+
  forecast:: autolayer(holttrend$mean, series = "Holt Linear Trend")+
  forecast:: autolayer(holtdamped$mean, series = "Holt Damped Trend")+
  forecast:: autolayer(arimafore$mean, series = "ARIMA")+
  xlab("year") + ylab("Labour Force Participation Rate Age 25 - 54")+
  guides(colour = guide_legend(title="Forecat Method"))+
  theme(legend.position = c(0.8,0.2))+
  ggtitle("Singapur")+
  theme(plot.title=element_text(family="Times", hjust=0.5, color="blue", face="bold", size=15))
```




```r
autoplot(singapur)+ geom_line(size=2)+
  forecast:: autolayer(holttrend$fitted, series = "Holt Linear Trend")+
  forecast:: autolayer(holtdamped$fitted, series = "Holt Damped Trend")+
  forecast:: autolayer(arimafore$fitted, series = "ARIMA")+
  xlab("year") + ylab("Labour Force Participation Rate Age 25 - 54")+
  guides(colour = guide_legend(title="Forecat Method"))+
  theme(legend.position = c(0.8,0.2))+
  ggtitle("Singapur")+
  theme(plot.title=element_text(family="Times", hjust=0.5, color="blue", face="bold", size=15))
```




```
# We can see that blue line adjust slowlier to changes
```