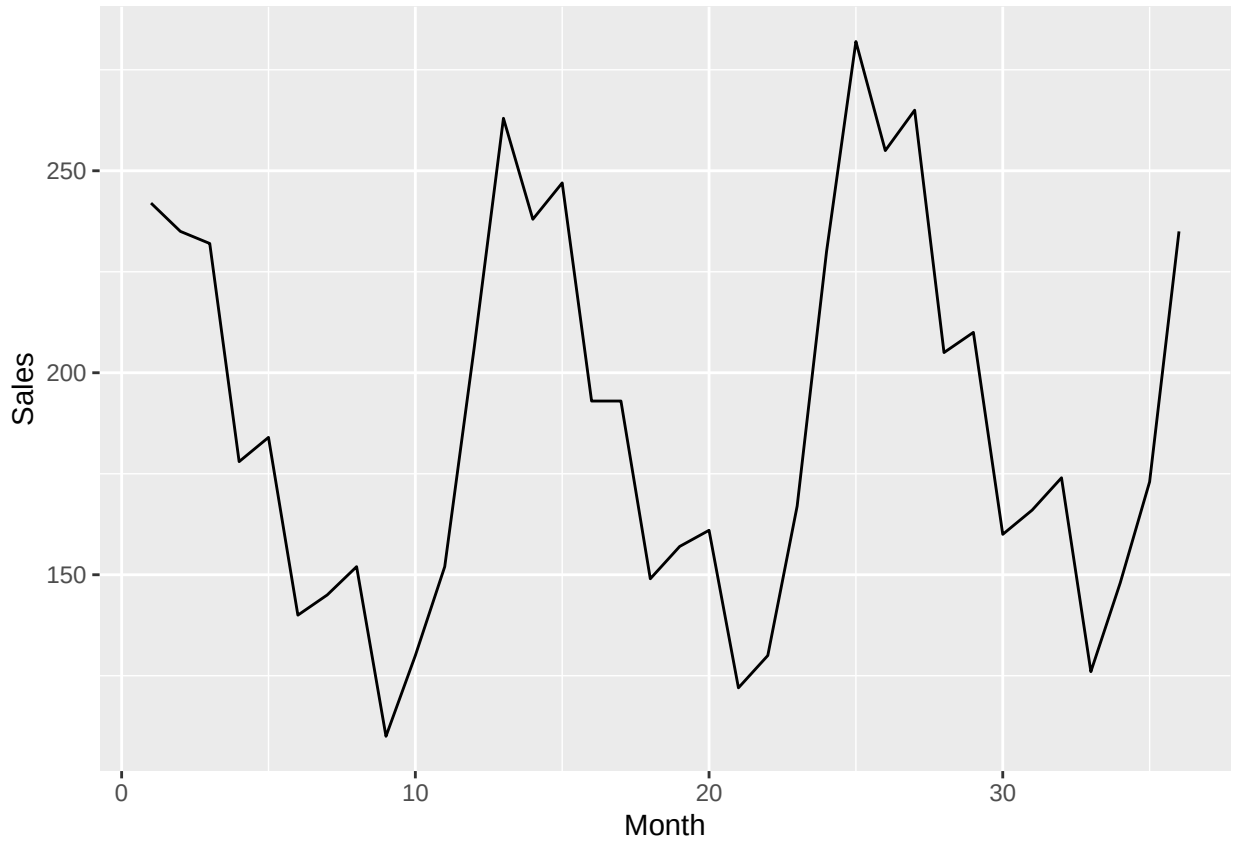
Time Series Assignment

2025-02-13



#comments

There is some seasonality in a motion of going up and down, the down shows that its rising indicating that there is a linear trend

```
vin <- vin %>% mutate(season = rep(1:12, length.out = nrow(vin)))
```

```
trend <- lm(Sales ~ Month, data = vin)
summary(trend)
```

```
##
## Call:
## lm(formula = Sales ~ Month, data = vin)
##
## Residuals:
##    Min     1Q Median     3Q    Max
## -77.75 -36.48 -11.64  44.93  94.44
```

```
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept) 187.85556   16.28847  11.533 2.68e-13 ***
## Month        -0.01171    0.76770  -0.015    0.988
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 47.85 on 34 degrees of freedom
## Multiple R-squared:  6.845e-06,  Adjusted R-squared:  -0.0294
## F-statistic: 0.0002327 on 1 and 34 DF,  p-value: 0.9879
```

```
seasons <- lm(Sales ~ Month + as.factor(season), data = vin)
summary(seasons)
```

```
##
## Call:
## lm(formula = Sales ~ Month + as.factor(season), data = vin)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.1250 -1.9583  0.1667  2.2292  7.4583
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)          249.10764    2.74567  90.727  < 2e-16 ***
## Month                  1.01736    0.07554  13.467 2.14e-12 ***
## as.factor(season)2   -20.68403    3.62688  -5.703 8.30e-06 ***
## as.factor(season)3   -16.36806    3.62924  -4.510 0.000158 ***
## as.factor(season)4   -73.38542    3.63317 -20.199 3.90e-16 ***
## as.factor(season)5   -70.73611    3.63866 -19.440 8.97e-16 ***
## as.factor(season)6  -117.75347    3.64571 -32.299  < 2e-16 ***
## as.factor(season)7  -112.43750    3.65431 -30.768  < 2e-16 ***
## as.factor(season)8  -107.12153    3.66445 -29.233  < 2e-16 ***
## as.factor(season)9  -151.13889    3.67611 -41.114  < 2e-16 ***
## as.factor(season)10 -135.48958    3.68928 -36.725  < 2e-16 ***
## as.factor(season)11 -108.50694    3.70395 -29.295  < 2e-16 ***
## as.factor(season)12  -49.85764    3.72009 -13.402 2.36e-12 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.441 on 23 degrees of freedom
## Multiple R-squared:  0.9942, Adjusted R-squared:  0.9911
## F-statistic:   327 on 12 and 23 DF,  p-value: < 2.2e-16
```

```
library(modelr)
library(broom)
```

```
##
## Attaching package: 'broom'

## The following object is masked from 'package:modelr':
##
##     bootstrap
```

```r
library(janitor)
```

```
##
## Attaching package: 'janitor'

## The following objects are masked from 'package:stats':
##
##     chisq.test, fisher.test
```

```r
rmse(trend, vin)
```

```
## [1] 46.50262
```

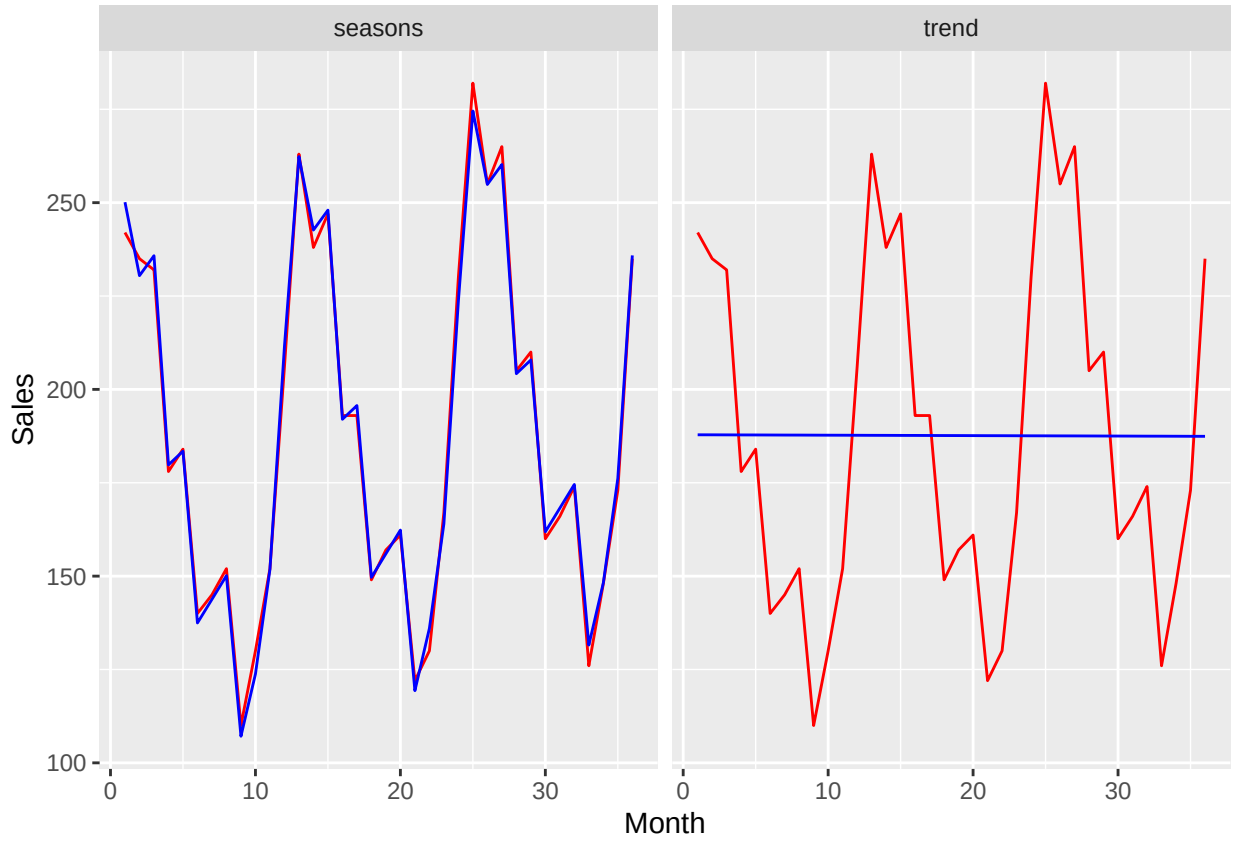
```r
rmse(seasons, vin)
```

```
## [1] 3.549746
```

```r
trend_p <- augment(trend, vin) %>% mutate(model = "trend")
seasons_p <- augment(seasons, vin) %>% mutate(model = "seasons")
```

```r
compare <- rbind(trend_p, seasons_p)
```

```r
compare %>%
  ggplot()+
  geom_line(aes(x=Month,y=Sales), color = "red")+
  geom_line(aes(x=Month,y=.fitted), color = "blue")+
  facet_wrap(~model)
```




```
forecasts <- data.frame(Month = c(37:48), season = c(1:12)) %>%
  add_predictions(seasons)

print(forecasts)
```

```
##    Month season     pred
## 1     37      1 286.7500
## 2     38      2 267.0833
## 3     39      3 272.4167
## 4     40      4 216.4167
## 5     41      5 220.0833
## 6     42      6 174.0833
## 7     43      7 180.4167
## 8     44      8 186.7500
## 9     45      9 143.7500
## 10    46     10 160.4167
## 11    47     11 188.4167
## 12    48     12 248.0833
```

#comments

the sales start high on January and start declining the most during month 6, maybe summer time. the sales get back up around month 8 little by little. also similar to the other 3 years but this time with more sales starting January

```r
actual_jan_sales <- 295  # Given actual value in thousands
forecast_jan_sales <- forecasts %>% filter(Month == 37) %>% pull(pred)
```

```r
forecast_error <- (forecast_jan_sales - actual_jan_sales) * 1000
print(paste("Forecast Error for January:", forecast_error))
```

```
## [1] "Forecast Error for January: -8249.99999999983"
```

#comments

Our model is an accurate model closer to the years before if we wanted to get closer to our predictions and minimize the forecast error if we got more data from certain seasons promotions or if we used arima models, i also think if we had more information on the sales and wether they were offering new products or price changes.