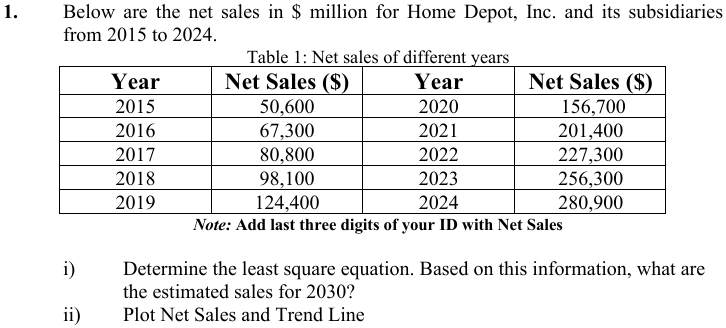
****

To determine the least square equation for the given data, follow these steps:

### 1. **Organize Data**:

Given data (Net Sales in $M, adjusted with ID last three digits, 27):

| Year |  | Net Sales () |
| --- | --- | --- |
| 2015 | 0 | 50,627 |
| 2016 | 1 | 67,327 |
| 2017 | 2 | 80,827 |
| 2018 | 3 | 98,127 |
| 2019 | 4 | 124,427 |
| 2020 | 5 | 156,727 |
| 2021 | 6 | 201,427 |
| 2022 | 7 | 227,327 |
| 2023 | 8 | 256,327 |
| 2024 | 9 | 280,927 |

### 2. **Apply the Least Squares Method**:

The least square equation is:

Where:

### 3. **Compute Summations**:

From the data:

### 4. **Calculate Constants and** :

### 5. **Final Least Squares Equation**:

Here, .

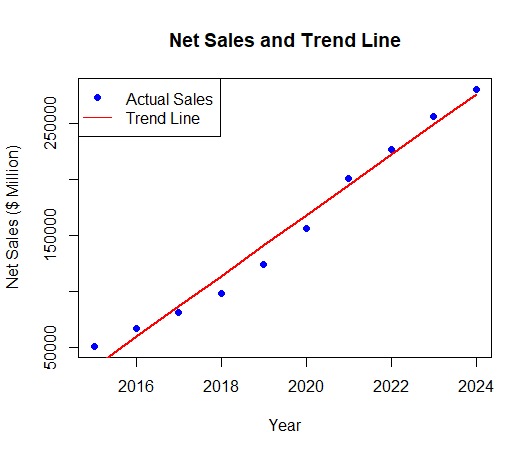
**Calculate for 2030**:

**Substitute into the equation**:

### **Estimated Sales for 2030**:

y = 32,487 + 27,093.33x

**2)plot net scale trend line**



**Graph code by R**

years <- c(2015:2024)

adjusted\_sales <- c(50627, 67327, 80827, 98127, 124427, 156727, 201427, 227327, 256327, 280927)

# Center years for regression

x <- years - 2019.5

model <- lm(adjusted\_sales ~ x)

# Prediction for the trend line

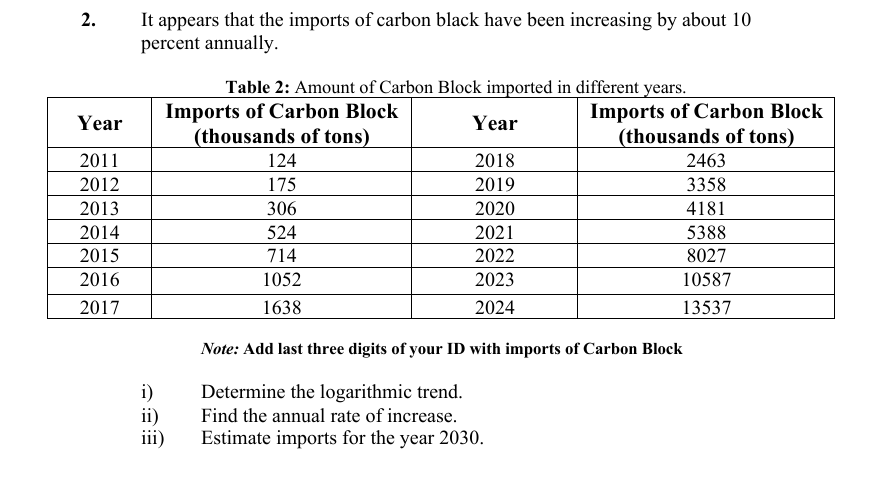
trend <- predict(model, newdata = data.frame(x = x))

# Plot

plot(years, adjusted\_sales, main = "Net Sales and Trend Line", xlab = "Year", ylab = "Net Sales ($ Million)", col = "blue", pch = 19)

lines(years, trend, col = "red", lwd = 2)

legend("topleft", legend = c("Actual Sales", "Trend Line"), col = c("blue", "red"), pch = c(19, NA), lty = c(NA, 1))

2) 

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Year | Imports of Carbon (Y) | x | xx | log(Y) | x log(y) | Logarithomic Yc |  |
| 2011 | 1240027 | -13 | 169 | 6.093431 | -79.2146 | 5.190542 | 155075.2522 |
| 2012 | 1750027 | -11 | 121 | 6.243045 | -68.6735 | 5.347631 | 222654.4441 |
| 2013 | 3060027 | -9 | 81 | 6.485725 | -58.3715 | 5.50472 | 319683.5137 |
| 2014 | 5240027 | -7 | 49 | 6.719334 | -47.0353 | 5.661809 | 458996.2232 |
| 2015 | 7140027 | -5 | 25 | 6.8537 | -34.2685 | 5.818898 | 659019.0731 |
| 2016 | 10520027 | -3 | 9 | 7.022017 | -21.0661 | 5.975987 | 946208.5238 |
| 2017 | 16380027 | -1 | 1 | 7.214315 | -7.21431 | 6.133076 | 1358550.317 |
| 2018 | 24630027 | 1 | 1 | 7.391465 | 7.391465 | 6.290165 | 1950583.742 |
| 2019 | 33580027 | 3 | 9 | 7.526081 | 22.57824 | 6.447253 | 2800615.396 |
| 2020 | 41810027 | 5 | 25 | 7.62128 | 38.1064 | 6.604342 | 4021076.578 |
| 2021 | 53880027 | 7 | 49 | 7.731428 | 54.11999 | 6.761431 | 5773394.259 |
| 2022 | 80270027 | 9 | 81 | 7.904553 | 71.14098 | 6.91852 | 8289342.572 |
| 2023 | 105870027 | 11 | 121 | 8.024773 | 88.2725 | 7.075609 | 11901698.93 |
| 2024 | 135370027 | 13 | 169 | 8.131523 | 105.7098 | 7.232698 | .56 |
|  |  |  |  |  |  |  |  |

Here, N=14; ∑logy= 86.9626823; ∑xlogy = 71.47543718; ∑x2 = 910

We know,

Log(a) = = = 6.211620164

**Log(b)=** = = = 0.078544436

**Trend equation:**

logY = log(a) + log(b)x

**For the year 2023, x= 25**

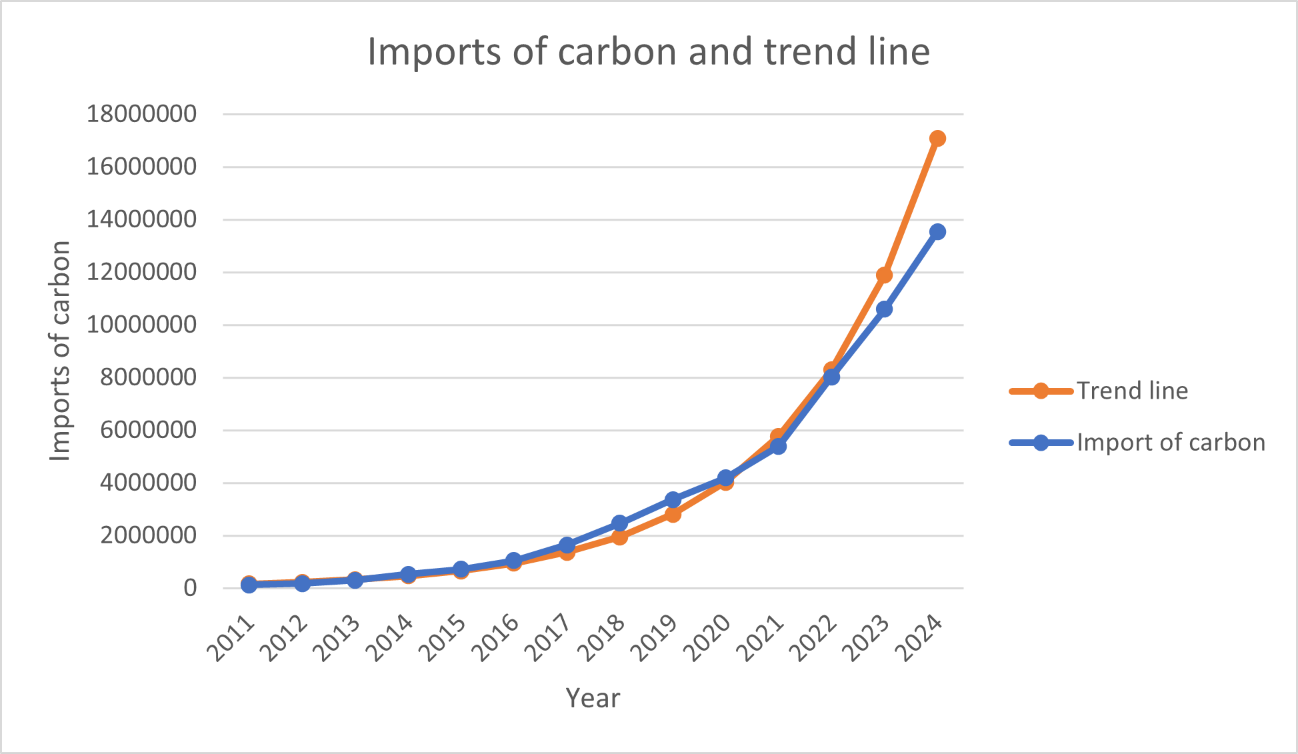
logY = = 6.211620164 + 0.078544436 \* 25 = 8.175231

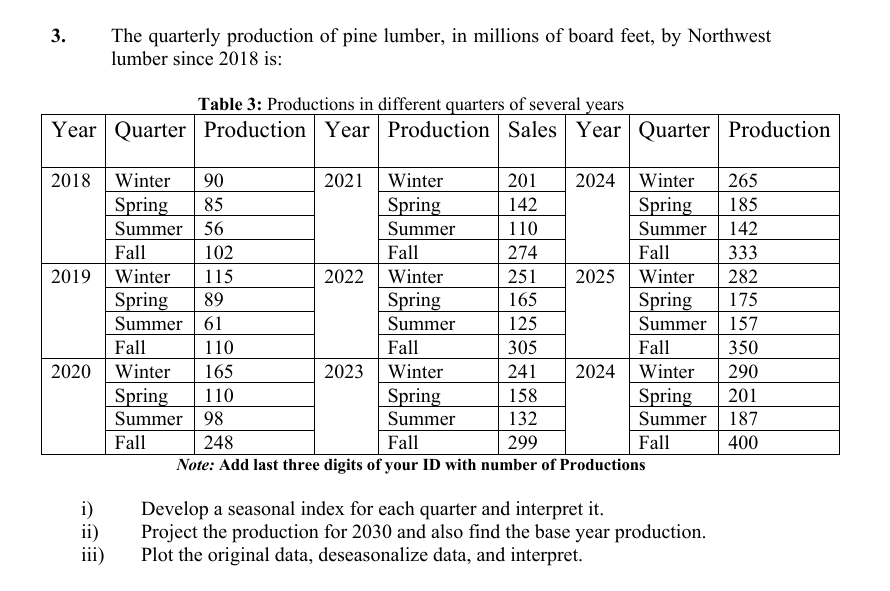
Y2030 = 149703193

**Annual rate of increase** = (eb -1) x 100%

= (e0.0785 -1) x 100%

= 8.1663 %



****

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Year | Winter(production) | Spring(production) | Summer(production) | Fall(production) | Mean |
| 2018 | 90027 | 85027 | 56027 | 102027 | 83277 |
| 2019 | 115027 | 89027 | 61027 | 110027 | 93777 |
| 2020 | 165027 | 110027 | 98027 | 248027 | 155277 |
| 2021 | 201027 | 142027 | 110027 | 274027 | 181777 |
| 2022 | 251027 | 165027 | 125027 | 305027 | 211527 |
| 2023 | 241027 | 158027 | 132027 | 299027 | 207527 |
| 2024 | 265027 | 185027 | 142027 | 333027 | 231277 |
| 2025 | 282027 | 175027 | 157027 | 350027 | 241027 |
| 2026 | 290027 | 201027 | 187027 | 400027 | 269527 |
|  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Seasonal Index calculation:Divide seasonal value of each year with the mean of each year.Then we get, | | | | | | |
|
| Year | Winter(production) | Spring(production) | Summer(production) | Fall(production) |  |  |
| 2018 | 1.081054793 | 1.021014206 | 0.672778798 | 1.225152203 |  |  |
| 2019 | 1.226601405 | 0.949347921 | 0.650767246 | 1.173283428 |  |  |
| 2020 | 1.062791012 | 0.708585302 | 0.631304057 | 1.597319629 |  |  |
| 2021 | 1.105898986 | 0.78132547 | 0.605285597 | 1.507489946 |  |  |
| 2022 | 1.18673739 | 0.780169907 | 0.591068752 | 1.44202395 |  |  |
| 2023 | 1.161424778 | 0.76147682 | 0.636191917 | 1.440906484 |  |  |
| 2024 | 1.145928908 | 0.800023349 | 0.614099111 | 1.439948633 |  |  |
| 2025 | 1.170105424 | 0.726171757 | 0.651491327 | 1.452231493 |  |  |
| 2026 | 1.07605917 | 0.745851065 | 0.693908217 | 1.484181548 |  |  |
|  |  |  |  |  |  |  |

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Overall Seasonal Index | | | | | | | |
| Seasonal Index | Winter | Spring | Summer | Fall |  |  |  |
| SI | 1.135177985 | 0.808218422 | 0.638543891 | 1.418059701 |  |  |  |
| Sum of SI | 4 |  |  |  |  |  |  |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| DESEASONISED DATA | | | | |  |  |
| Year | Winter(production) | Spring(production) | Summer(production) | Fall(production) | Sum | Sl no |
| 2018 | 79306.50626 | 105202.9967 | 87741.81503 | 71948.3107 | 344199.629 | 1 |
| 2019 | 101329.4844 | 110152.1539 | 95572.13033 | 77589.82212 | 384643.591 | 2 |
| 2020 | 145375.4408 | 136135.2291 | 153516.4635 | 174905.8941 | 609933.028 | 3 |
| 2021 | 177088.5294 | 175728.4865 | 172309.2202 | 193240.8062 | 718367.042 | 4 |
| 2022 | 221134.4857 | 204186.1402 | 195800.1661 | 215101.663 | 836222.455 | 5 |
| 2023 | 212325.2945 | 195525.1151 | 206762.6075 | 210870.5294 | 825483.547 | 6 |
| 2024 | 233467.3535 | 228931.9261 | 222423.2381 | 234846.953 | 919669.471 | 7 |
| 2025 | 248442.9787 | 216559.0331 | 245914.184 | 246835.1647 | 957751.361 | 8 |
| 2026 | 255490.3317 | 248728.5548 | 292896.0758 | 282094.6111 | 1079209.57 | 9 |

Production in 2030:

For winter y = 22625x – 5E+07; for x = 2030 we get production = 35,928,750

For spring, y = 18312x-4E+07; for x = 2030 we get production =

For summer,y = 24066x-5E+07; for x = 2030 we get production =

For fall, y = 24764 X - 4E+07; for x=2030 we get production =