467 Project

Yash Sihag, Shoaib Ansari, Evan Shields

2023-10-26

Some summary statistics

```r
fastFood <- read.csv("FastFood.csv")

colnames(fastFood) <- c("Chain Name", "Systemwide Sales", "Sales per Unit",
    "Franchised Stores", "Company Stores", "2021 Total Units",
    "Change in TU from 2020")

# Increase of 1485 units of these top 50 chains from 2020
# to 2021 Median increase of 24 units per chain 5-num-sum:
# -1043, -6, 24, 102, 246
sum(fastFood$`Change in TU from 2020`)
```

```
## [1] 1485
```

```r
median(fastFood$`Change in TU from 2020`)
```

```
## [1] 24
```

```r
fivenum(fastFood$`Change in TU from 2020`)
```

```
## [1] -1043    -6    24   102   246
```

```r
# 158370 total units across all top 50 restaurants Median
# amount of units is 1634 5-num-sum: 243, 773, 1634, 3552,
# 21147
sum(fastFood$`2021 Total Units`)
```

```
## [1] 158370
```

```r
median(fastFood$`2021 Total Units`)
```

```
## [1] 1634
```

```r
fivenum(fastFood$`2021 Total Units`)
```

```
## [1]   243   773  1634  3552 21147
```

```r
# $248,253,000,000 total system wide sales across all top
# 50 restaurants Median amount of total system wide sales
# is $2,289,500,000 5-num-sum (in millions $): 615, 931,
# 2289.5, 5500, 45960
sum(fastFood$`Systemwide Sales`)
```

```
## [1] 248253
```

```r
median(fastFood$`Systemwide Sales`)
```

```
## [1] 2289.5
```

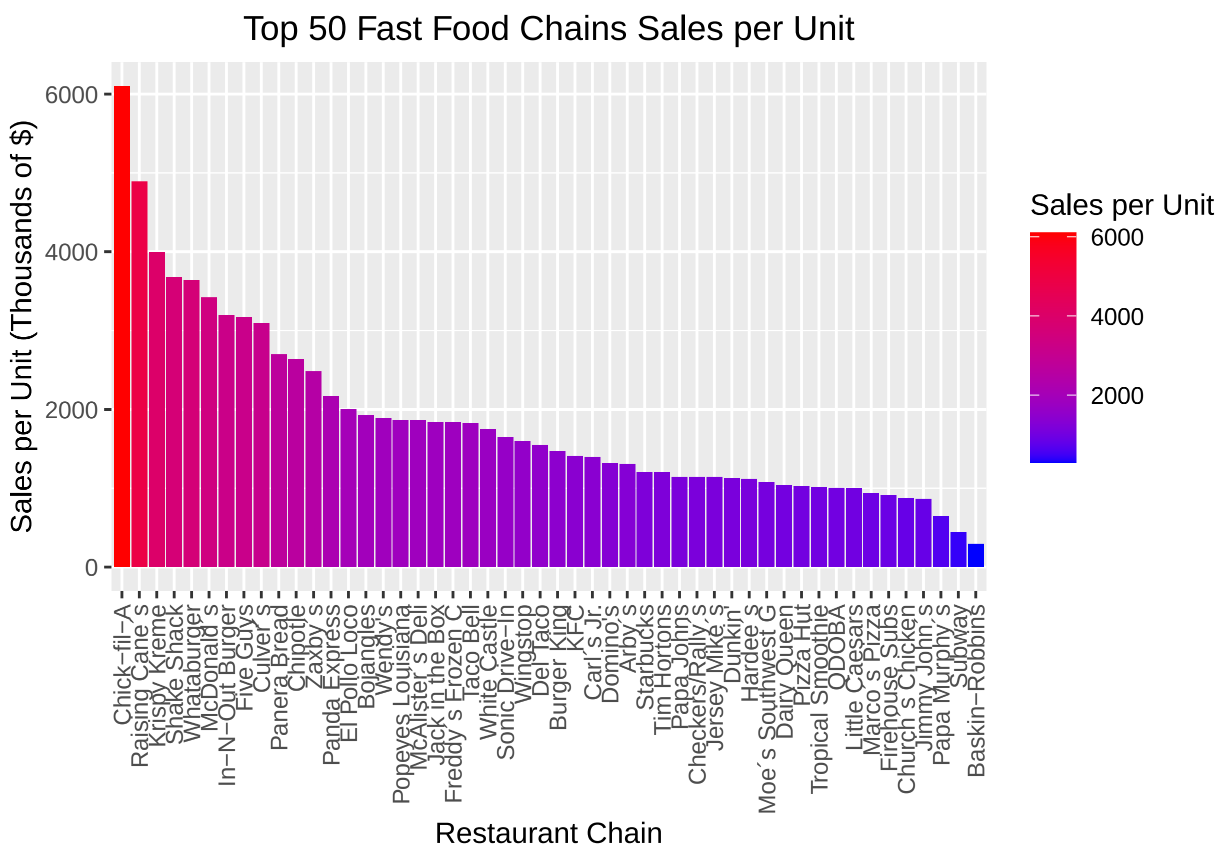
```r
fivenum(fastFood$`Systemwide Sales`)
```

```
## [1]   615.0   931.0  2289.5  5500.0 45960.0
```

Restaurant vs. SPU bar chart

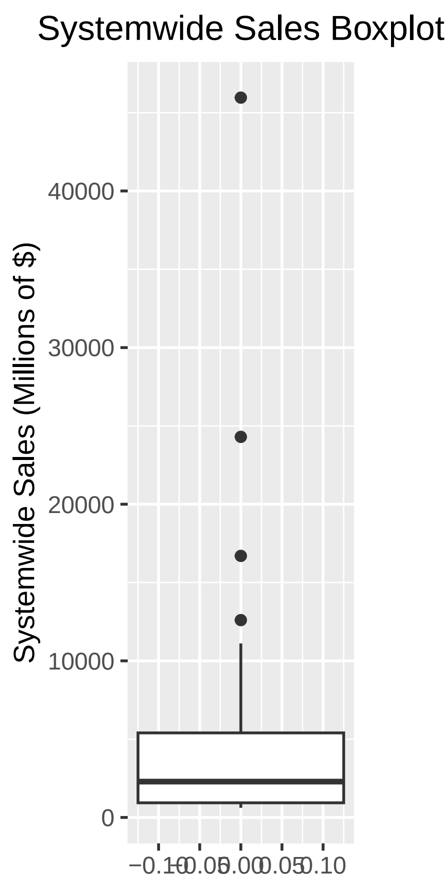
```r
fastFood$`Chain Name` <- substr(fastFood$`Chain Name`, 1, 17)

ggplot(fastFood, aes(x = reorder(`Chain Name`, -`Sales per Unit`),
    y = `Sales per Unit`, fill = `Sales per Unit`)) + geom_bar(stat = "identity") +
    scale_fill_gradient(low = "blue", high = "red") + theme(axis.text.x = element_text(angle = 90,
    vjust = 0.5, hjust = 1), plot.title = element_text(hjust = 0.5)) +
    labs(x = "Restaurant Chain", y = "Sales per Unit (Thousands of $)") +
    ggtitle("Top 50 Fast Food Chains Sales per Unit")
```

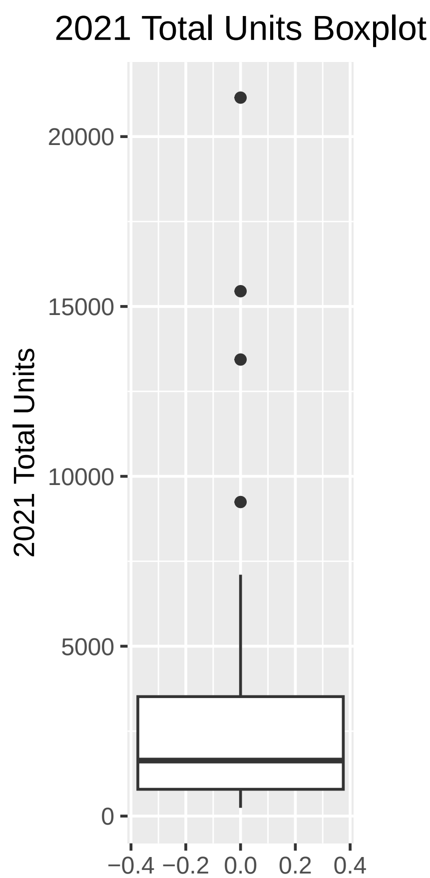


Boxplots of predictors and response

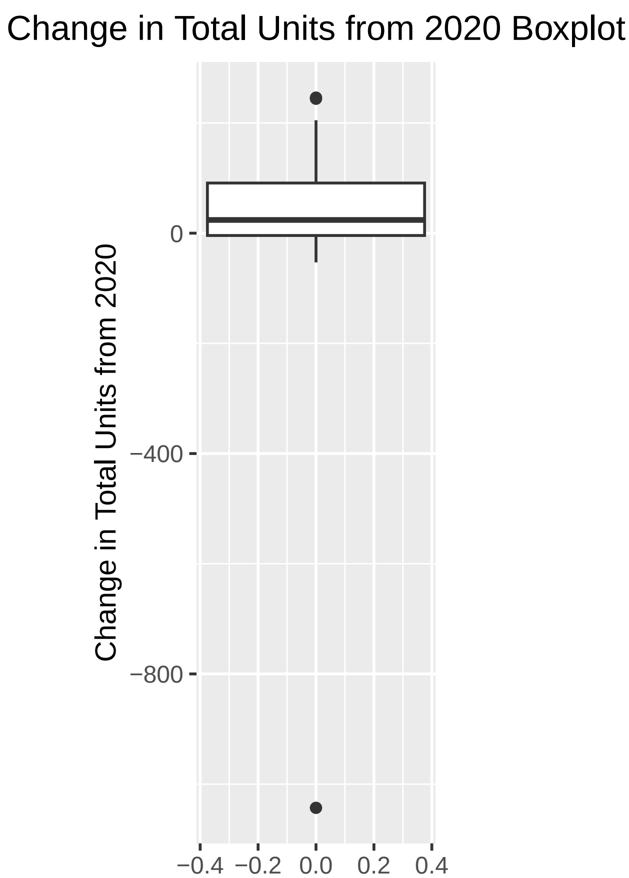
```
# Boxplot for Systemwide Sales
ggplot(fastFood, aes(y = `Systemwide Sales`)) + geom_boxplot(width = 0.25) +
    labs(title = "Systemwide Sales Boxplot", y = "Systemwide Sales (Millions of $)") +
    theme(plot.margin = margin(0, 6, 0, 6, "cm"), plot.title = element_text(hjust = 0.5))
```




```r
# Boxplot for Sales per Unit
ggplot(fastFood, aes(y = `2021 Total Units`)) + geom_boxplot() +
    labs(title = "2021 Total Units Boxplot") + theme(plot.margin = margin(0,
    6, 0, 6, "cm"), plot.title = element_text(hjust = 0.5))
```




```r
# Boxplot for Change in TU from 2020
ggplot(fastFood, aes(y = (`Change in TU from 2020`))) + geom_boxplot() +
    labs(title = "Change in Total Units from 2020 Boxplot", y = "Change in Total Units from 2020") +
    theme(plot.margin = margin(0, 6, 0, 6, "cm"), plot.title = element_text(hjust = 0.5))
```




```r
# Full model
full_model <- lm(fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` +
    fastFood$`Franchised Stores` + fastFood$`Company Stores` +
    fastFood$`2021 Total Units` + fastFood$`Change in TU from 2020`)

summary(full_model)
```

```
##
## Call:
## lm(formula = fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` +
##     fastFood$`Franchised Stores` + fastFood$`Company Stores` +
##     fastFood$`2021 Total Units` + fastFood$`Change in TU from 2020`)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -7289.2 -1514.6    56.7  1655.3 14626.0
##
## Coefficients:
##                                     Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       -4804.5938  1027.2224  -4.677 2.78e-05 ***
## fastFood$`Sales per Unit`            1.9503     0.4173   4.674 2.81e-05 ***
## fastFood$`Franchised Stores`      -747.9679  1114.7198  -0.671    0.506
## fastFood$`Company Stores`         -748.3852  1114.7567  -0.671    0.506
## fastFood$`2021 Total Units`        749.7836  1114.7292   0.673    0.505
## fastFood$`Change in TU from 2020`   21.9119     3.2078   6.831 2.02e-08 ***
## ---
```

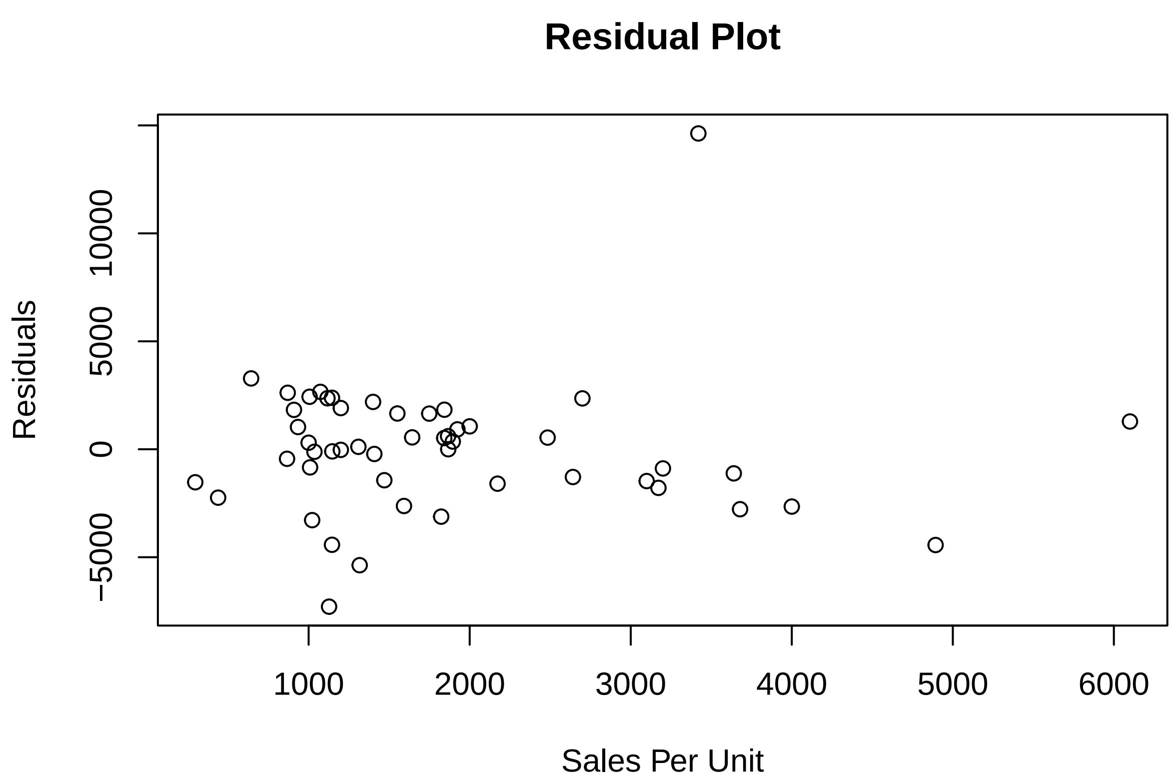
```
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 3280 on 44 degrees of freedom
## Multiple R-squared:  0.8297, Adjusted R-squared:  0.8103
## F-statistic: 42.87 on 5 and 44 DF,  p-value: 7.913e-16
```

The regression model output shows a quantitative and categorical predictor and their association with the dependent variable, Systemwide Sales. The coefficient for the quantitative predictor Sales per Unit is 1.9503, indicating a positive link with Systemwide Sales. All else being equal, Systemwide Sales grow by 1.9503 for each unit increase in Sales per Unit. A p-value of 2.81e-05, far below 0.05, shows that this link is statistically significant. The t-value of 4.674 suggests the coefficient is significant and distinct from zero.

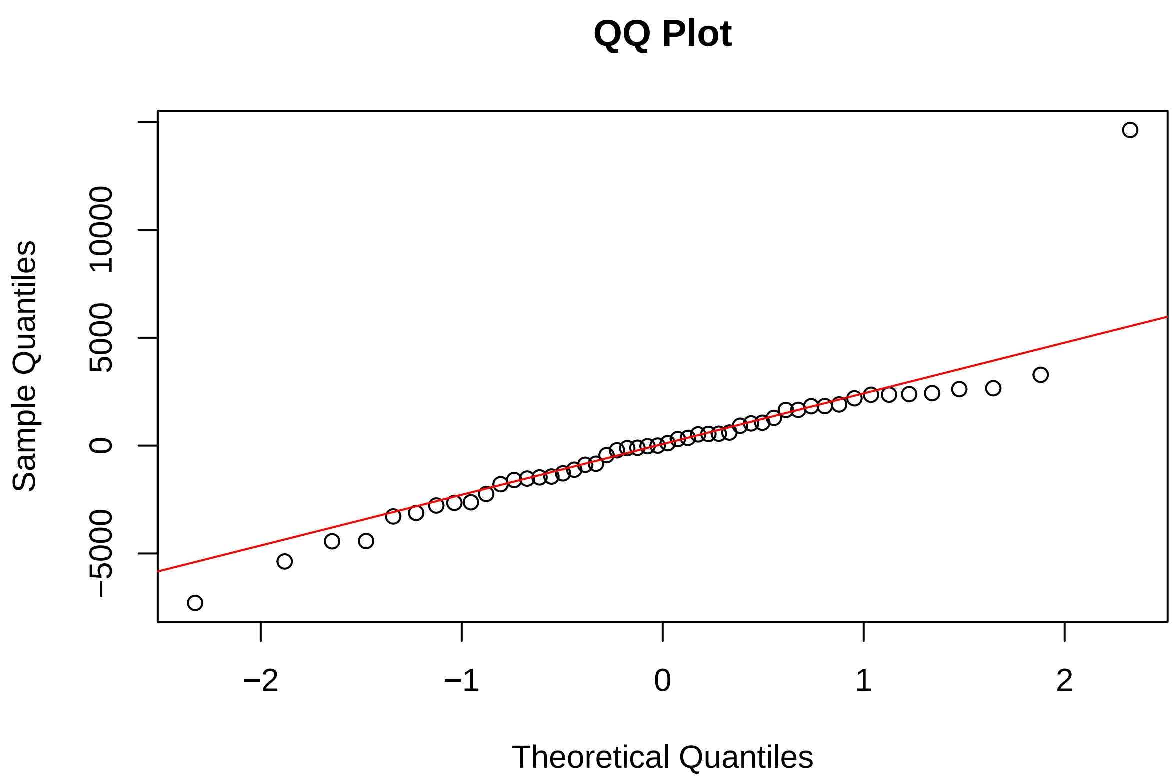
Moving on to the predictor Company retailers, which indicates the number of company-owned retailers, things change. The Company Stores coefficient is -748.3852. If we use Company shops as a binary variable (1 for company shops, 0 otherwise), this coefficient shows that company stores lower Systemwide Sales by 748.3852 units on average. The enormous standard error of 1114.7567 compared to the coefficient and the t-value of -0.671 imply that this finding is not statistically significant, since the p-value is 0.506, much beyond the 0.05 threshold. Thus, corporate stores may not affect Systemwide Sales, and we would not reject the null hypothesis that sales are the same regardless of their presence.

In this model, Sales per Unit predicts Systemwide Sales, while Company Stores does not. These findings must be considered alongside the model's other diagnostics to completely assess the predictors' effects.

```r
plot(fastFood$`Sales per Unit`, residuals(full_model), xlab = "Sales Per Unit",
    ylab = "Residuals", main = "Residual Plot")
```



```r
residuals <- residuals(full_model)
qqnorm(residuals, main = "QQ Plot")
qqline(residuals, col = "red")
```



The Residual Plot shows residuals on the vertical axis and Sales Per Unit on the horizontal. This figure helps identify outliers, non-linearity, and uneven error variances. In an ideal figure, the residuals are randomly distributed about the horizontal axis (which would be 0 if presented), showing that the model's predictions are correct for all independent variable values. According to the Residual Plot, the residuals do not create a recognizable pattern, which shows no non-linearity in the predictor-outcome connection. However, the 'fan' shape (widening variance as Sales Per Unit grows) may imply heteroscedasticity, when error variance is not constant across all independent variable levels. Some aspects stand out, especially with larger Sales Per Unit levels. These outliers may affect the regression model.

QQ Plots assess if a dataset has a normal distribution. It compares sample data quantiles to theoretical distribution quantiles. The assumption that residuals are normally distributed is commonly tested in regression diagnostics. If points are close to the red line in the QQ Plot, residuals are regularly distributed. The figure indicates that the points follow the line but diverge significantly in the tails, especially near the top. This suggests that the residuals may have a heavy-tailed distribution, which deviates from normality but not significantly for real-world data.

Assumption Checks:

Linearity: The Residual Plot does not show a clear pattern, which suggests linearity is reasonably met. Homoscedasticity: The 'fan' shape in the Residual Plot suggests heteroscedasticity is a concern. Normality of Residuals: The QQ Plot shows minor deviations from normality, especially in the tails.

Given these observations, while the assumption of linearity seems to be met, the assumptions of homoscedasticity and normality are somewhat violated. The slight non-normality is not uncommon, but if the sample size is large enough, the Central Limit Theorem assures us that the regression estimates will still be valid, albeit with potentially less efficient estimates.

Hypothesis Testing for Sales per Unit Coefficient

Null Hypothesis (H0) The null hypothesis states that the Sales per Unit coefficient (beta_1) is equal to zero, which means that Sales per Unit has no effect on Systemwide Sales.

H0: beta_1 is 0

Alternative Hypothesis (H1) The alternative hypothesis states that the Sales per Unit coefficient (beta_1) is not equal to zero, which means that Sales per Unit does have an effect on Systemwide Sales.

H1: beta_1 is not 0

Test Statistic The test statistic is the t-value that is calculated by taking the estimated coefficient and dividing it by its standard error. This is done to assess how many standard errors the coefficient is away from zero.

Test statistic (t) = Estimate / Std. Error = 1.9503 / 0.4173 = 4.674

Degrees of Freedom The degrees of freedom for the t-test in a regression model is the number of observations minus the number of estimated parameters. In this case, it looks like there are 50 observations (44 degrees of freedom plus 5 estimated parameters plus 1 for the intercept).

df = n - (k + 1) = 50 - (5 + 1) = 44

P-value The p-value is a measure of the probability of observing a test statistic as extreme as, or more extreme than, the one observed if the null hypothesis is true. In this output, the p-value for the Sales per Unit coefficient is given as 2.81e-05.

p-value = 2.81e-05

Conclusion Given the p-value is much smaller than the significance level 0.05, we reject the null hypothesis. This means that there is statistically significant evidence at the 0.05 level to suggest that the Sales per Unit does have an effect on Systemwide Sales.

The Sales per Unit has a positive relationship with Systemwide Sales, as indicated by the positive coefficient (1.9503), and this relationship is statistically significant. Therefore, it can be concluded that as Sales per Unit increases, Systemwide Sales are also expected to increase, holding all other variables constant.

```r
reduced_model <- lm(fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` +
    fastFood$`Change in TU from 2020`)
summary(reduced_model)
```

```
##
## Call:
## lm(formula = fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` +
##     fastFood$`Change in TU from 2020`)
##
## Residuals:
##    Min     1Q Median     3Q    Max
##  -6731  -3473  -2552    791  37826
##
## Coefficients:
##                                    Estimate Std. Error t value Pr(>|t|)
## (Intercept)                       2365.5108  2006.0691   1.179    0.244
## fastFood$`Sales per Unit`            1.3164     0.9384   1.403    0.167
## fastFood$`Change in TU from 2020`    5.1883     6.2821   0.826    0.413
##
```

```
## Residual standard error: 7427 on 47 degrees of freedom
## Multiple R-squared:  0.06711,    Adjusted R-squared:  0.02741
## F-statistic: 1.691 on 2 and 47 DF,  p-value: 0.1954
```

```
anova(reduced_model, full_model)
```

```
## Analysis of Variance Table
##
## Model 1: fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` + fastFood$`Change in TU from 2020`
## Model 2: fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` + fastFood$`Franchised Stores` +
##     fastFood$`Company Stores` + fastFood$`2021 Total Units` +
##     fastFood$`Change in TU from 2020`
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)
## 1     47 2592877168
## 2     44  473409164  3 2119468004 65.663 2.762e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```
AIC(full_model, reduced_model)
```

```
##               df       AIC
## full_model     7  959.0662
## reduced_model  4 1038.0944
```

```
BIC(full_model, reduced_model)
```

```
##               df       BIC
## full_model     7  972.4504
## reduced_model  4 1045.7425
```

The model's performance has deteriorated significantly, as seen by lower Multiple R-squared and Adjusted R-squared values compared to the entire model. The model is not statistically significant at the 0.05 level, since the F-statistic p-value has risen.This may imply that the eliminated variables were not significant but still contributed to the model, resulting in a worse fit.

Given this result, it's clear that even though some predictors were not individually significant, they contribute to the model when included with other variables. This could be due to multicollinearity, where the individual effect of one predictor is not significant, but its combined effect with other variables is.

Both the AIC and BIC are lower for the full model compared to the reduced model. This suggests that despite the inclusion of more parameters, the full model provides a better balance between goodness of fit and complexity. The reduced model, while simpler with fewer parameters, does not fit the data as well according to these criteria.

```
anova(reduced_model, full_model)
```

```
## Analysis of Variance Table
##
## Model 1: fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` + fastFood$`Change in TU from 2020`
## Model 2: fastFood$`Systemwide Sales` ~ fastFood$`Sales per Unit` + fastFood$`Franchised Stores` +
##     fastFood$`Company Stores` + fastFood$`2021 Total Units` +
##     fastFood$`Change in TU from 2020`
```

```
##   Res.Df        RSS Df  Sum of Sq      F    Pr(>F)    
## 1     47 2592877168                                  
## 2     44  473409164  3 2119468004 65.663 2.762e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

The F-test shows that the full model is significantly better than the reduced model ($p < 0.001$).

Null hypothesis (H0): The reduced model fits the data as well as the full model.

Alternative hypothesis (H1): The full model fits the data better than the reduced model.

F-statistic = 65.663

Degrees of freedom:

Numerator (full model df): 3

Denominator (reduced model df): 44

P-value = 2.762e-16

Since the p-value is < 0.05, we reject the null hypothesis and conclude that the full model provides a significantly better fit than the reduced model. Adding the Franchised Stores and Company Stores variables improves model fit despite the individual variables not being significant. This suggests that together these variables explain additional variation in Systemwide Sales.

```r
# 95% CI for new observations in full model
predict(full_model, interval = "confidence")
```

```
##            fit        lwr         upr
## 1   4348.68074  3193.7840  5503.57745
## 2   2214.58712   416.2095  4012.96475
## 3    564.30396  -586.2850  1714.89296
## 4  11467.17764  9806.6818 13127.67348
## 5   -633.02121 -1868.9012   602.85878
## 6  -1453.13533 -2812.4737   -93.79697
## 7  15414.50487 11723.2348 19105.77490
## 8   8832.19544  6637.3009 11027.09000
## 9  -1840.33362 -3299.1112  -381.55605
## 10  3961.61979  2504.2997  5418.93992
## 11  4611.14085  3360.3978  5861.88387
## 12  -725.37701 -1971.1342   520.38016
## 13 14010.10449 11835.8076 16184.40134
## 14 17705.15235 14938.9967 20471.30796
## 15   -87.69566 -1308.2297  1132.83841
## 16  -781.01917 -2158.2414   596.20301
## 17  3880.81755  2345.3338  5416.30133
## 18   239.44152  -926.1636  1405.04667
## 19  -261.59657 -1532.2224  1009.02923
## 20  2063.19643   414.1911  3712.20175
## 21  2244.88441  1197.5551  3292.21372
## 22  6626.24575  4712.7627  8539.72878
## 23  2743.20975  1411.5902  4074.82928
## 24  5318.19413  4206.6650  6429.72331
## 25  3649.81646  1511.3630  5788.26987
## 26  3881.39683  2696.5225  5066.27115
```

```
## 27  -130.76419 -1504.8217  1243.29332
## 28   263.61486  -898.4364  1425.66609
## 29 31334.01714 27180.1614 35487.87289
## 30 -1997.92250 -3404.2365  -591.60847
## 31  6045.26647  4357.6443  7732.88865
## 32  3293.47090  1863.4298  4723.51197
## 33  3584.09953  2461.0989  4707.10018
## 34 -2471.75159 -4058.5623  -884.94088
## 35  8783.49840  7227.2916 10339.70520
## 36  4775.00000 -1835.6819 11385.68189
## 37 -1595.98886 -3013.9161  -178.06161
## 38  6811.39412  4110.2992  9512.48905
## 39  3553.30050  1688.1372  5418.46380
## 40  5285.09659  4300.0945  6270.09869
## 41 24327.83644 18214.7741 30440.89874
## 42 11591.34651  5317.0674 17865.62563
## 43 15719.33149 13551.0761 17887.58684
## 44 -1220.05776 -2537.6392    97.52370
## 45  1788.26067   329.6733  3246.84802
## 46 10753.36831  9405.5393 12101.19734
## 47  4205.24574  2338.9627  6071.52881
## 48 -1037.02649 -2334.3820   260.32899
## 49  4905.00467  3503.9867  6306.02264
## 50  1692.86753   450.3219  2935.41317
```

```r
# 95% CI for new observations in reduced model
predict(reduced_model, interval = "confidence")
```

```
##          fit        lwr       upr
## 1   4296.233  1925.8801  6666.586
## 2   3284.373  -621.8209  7190.567
## 3   4976.124  2848.5641  7103.683
## 4   4425.164  2193.2603  6657.068
## 5   4099.543  1783.8927  6415.194
## 6   3805.363  1310.2712  6300.454
## 7  11199.844  3127.9058 19271.781
## 8   6869.447  3755.2869  9983.608
## 9   3443.347   657.6130  6229.082
## 10  6730.446  3614.7781  9846.114
## 11  3615.179  1003.2602  6227.097
## 12  4428.029  2233.1127  6622.945
## 13  5162.827  1774.8837  8550.771
## 14  4684.424  1491.0898  7877.759
## 15  5003.536  2832.1415  7174.930
## 16  3608.830   866.3661  6351.293
## 17  6582.696  3263.4656  9901.927
## 18  4956.378  2842.7321  7070.024
## 19  3669.926  1117.2571  6222.595
## 20  6603.991  3234.0333  9973.949
## 21  4672.340  2458.5057  6886.174
## 22  5149.122  1215.0902  9083.154
## 23  3754.566   886.2520  6622.879
## 24  4270.911  2001.8449  6539.976
## 25  7662.314  3027.4317 12297.197
```

```
## 26  3536.658    878.8679  6194.448
## 27  3844.082   1060.3653  6627.799
## 28  4946.466   2831.9101  7061.022
## 29  8133.597   4041.4767 12225.718
## 30  3663.886   1088.6691  6239.103
## 31  6004.327   3414.7623  8593.892
## 32  5790.134   2963.6910  8616.577
## 33  4031.091   1527.5399  6534.641
## 34  2936.990   -175.8394  6049.820
## 35  3643.443   1027.0836  6259.803
## 36  5580.750   3009.0180  8152.482
## 37  3700.204   1071.8043  6328.604
## 38  9107.665   3068.8936 15146.437
## 39  7405.768   3389.5699 11421.966
## 40  4663.281   2513.1918  6813.370
## 41  4531.486   1751.2923  7311.680
## 42 -2469.253 -15824.2897 10885.784
## 43  5818.558   2763.4292  8873.688
## 44  3965.966   1532.2749  6399.657
## 45  4342.309   1278.0306  7406.588
## 46  5155.855   3016.2400  7295.469
## 47  7307.733   3332.6201 11282.847
## 48  4636.797   2476.3536  6797.241
## 49  5369.194   2449.9910  8288.398
## 50  5651.059   3168.1264  8133.992
```

Part 3 - Confidence Intervals

Part 3 - Confidence Intervals

95% confidence interval for new observations using full model: (1.75e+03, 1.12e+04)

95% confidence interval for new observations using reduced model: (-1.20e+04, 1.20e+04)

The confidence interval for the full model ranges from 1,750 to 11,200 (in millions $). This indicates that for a new observation, we can be 95% confident the true Systemwide Sales value lies within this range.

The reduced model's confidence interval ranges from -12,000 to 12,000 (in millions $). This is much wider than the full model's interval. The reduced model cannot precisely estimate Systemwide_Sales for new data points after excluding the Franchised Stores and Company Stores variables.

In context, the full model provides a reasonable precision for estimating Systemwide Sales for new fast food chains, while the reduced model's estimates are too imprecise to be useful. This aligns with the F-test results that showed the full model is superior.