**Lab 2**

**You are provided with the weekly historical sales revenue from 45 Walmart stores (real data):**

|  |  |
| --- | --- |
| **Variable** | **Definition** |
| Weekly\_Sales | Weekly sales for a given store ($) |
| CPI | Consumer Price Index *(Google it!)* |
| Size | Size of the store |
| IsHoliday | Whether the week is a special holiday week |
| Temperature | Average temperature of that region |
| Fuel\_Price | Cost of fuel in that region |
| Unemployment | Unemployment rate in that region |
| Store | Store ID |
| Date | Start date of the particular week |

This lab will provide you with the opportunity to work on a business case using real data. Start with [the template](https://drive.google.com/file/d/1ul9Q_oN1IGEPysIhBSlDwiicX6SaUMJf/view?usp=sharing), load the following libraries in the order: *tidyverse*, *tidymodels*, *plotly*, *skimr,* ***lubridate, car***. Remember, you load the libraries in the first chunk: *library(‘library\_name’)*

After loading the libraries, load the [**walmartSales.csv**](https://drive.google.com/open?id=1Z8-uG76Q3NWPZWJo3xu1YJuRdkirGlis) in the **data** folder as *dfw*. Then, explore the dataset using ***any of the following functions*** *as you see fit* to understand the data first:

head(), str(), glimpse(), nrow(), dim(), summary(), and skim() -> Use wide screen!

1. Create a regression model using Weekly\_Sales as the DV (Dependent Variable, outcome variable), and CPI as the IV (Independent Variable, feature, predictor, explanatory variable). *[If you don’t remember how to run and interpret a linear model in R, see the appendix]*

*fitCPI <- lm(*...fill in here...*)  
summary(fitCPI)*

What is the coefficient of CPI, and what does it mean in plain English?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Based on the output, how good is the model explaining the variance in Weekly\_Sales? Why? Given the fact, do you think your interpretation of the coefficient of CPI is still useful? Why?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. For Store 10, create a scatter plot of the relationship between CPI and Weekly\_Sales. Add a regression line to this plot. What do you observe? Does it align with your interpretation in Q1? Now, try it for Store 11, Store 12, and Store 13. What do you think is going on here?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

plot <- *dfw %>%*

*filter(*...fill in here...*) %>%*

*ggplot(*...fill in here...*) +   => Set Weekly\_Sales as y!*

*geom\_point() +*

*geom\_smooth(method=lm)*

*plot             => For the static plot*

*ggplotly(plot)    => For the dynamic plot*

1. Now, filter for the year 2012 instead of a store (so, you’ll plot data from all stores in a year). For this, you will need to (install and) load the lubridate library. Check the cheat sheet for lubridate [here](https://github.com/rstudio/cheatsheets/raw/master/lubridate.pdf). *[ Start by copying/pasting your code from Q2 into a new chunk and reuse ]*

What do you observe? Why do you think there are almost vertical clusters of observations?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Now, create a plot of sales in Store 1 in the year 2010. Did you know that you can use multiple arguments in one filter function as follows: *filter(argument\_1, argument\_2,…)?*

Compared to the earlier plots, do you notice a difference in the range of CPI? Why is it so?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Build another regression model but this time include both CPI and Size as independent variables and call it *fitCPISize*. Compare this model with the model you built in Q1.

Which model is better at explaining Weekly Sales? Why? **Hint:** Use *anova()* **as well**.

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Has the estimated coefficient for CPI changed? If so, why do you think it has changed?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

1. Let’s build a full model now and call it *fitFull*. This time, include all the variables in the dataset **(EXCEPT Store AND Date)** and report your observations. You can **also** use *anova()* to compare the reduced model in Q5 with the full model you have just built in this question.
2. The output of Q7 shows that temperature is positively associated with weekly sales. However, is that relationship really linear? Test it out by adding a squared transformation of temperature into the model using the following *I(Temperature^2)* and call it *fitFullTemp*

What is the coefficient of the squared term? Is it statistically significant? What does it mean? Based on this, what would you do differently if you were managing Walmart’s promotions?

\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Let’s visualize the relationship of temperature with sales to understand what it means:

dfw %>% ggplot(aes(...fill in here...)) +

geom\_smooth(...fill in here...**, formula = y ~ x + I(x^2)**)

*This is to visualize the shape of the relationship between temperature and sales. Note that the values shown in this plot are not accurate (but the curved shape is) because the model defined in the ggplot() above includes only Temperature and its squared form as the two independent variables.*

1. In a true predictive analytics exercise, we need to split the dataset, train the model using the training dataset and make predictions using the test set. Now, let’s do it the predictive way.   *[ Do* ***not*** *hesitate to copy and paste* ***your own code*** *from above, change, and reuse here ]*
2. Set the seed to **333** [ Always set the seed and split your data in the same chunk! ]
3. Randomly sample 80% of the data for training, and assign the difference to the test set

*dfwTrain <- dfw %>% sample\_frac(...fill in here...)*

*dfwTest <-* ***dplyr::setdiff****(dfw, dfwTrain) =>Make sure dplyr is there!*

1. Run the model from **Q8 using only the training set** now, and store the model as *fitOrg    [ By the way, as a prep for more advanced analysis work, try running* tidy(fitOrg)*Can you imagine the benefits of being able to convert a regression output into a tibble? ]*
2. Create a new copy of the test data frame *dfwTest* by adding the predicted values as a new column. Name this new dataframe as *resultsOrg*

resultsOrg <- dfwTest %>%

              mutate(predictedSales = predict(fitOrg, dfwTest))

Before you press on, check out *resultsOrg*, the new data frame you have just created.

1. Calculate the performance measures by calling the  *rmse(resultsOrg, truth=..., estimate=...)* and *mae(...)* functions and inputting the values stored in *resultsOrg*

What do they mean? Are they interpretable? If so, how do you make sense of them?

**Hint:** Instead of running the rmse() and mae() functions separately every time, you can:

performance <- metric\_set(rmse, mae) *=> You need to run this only once!*

performance(*...fill...*, truth=*...fill...*, estimate=*...fill...*)

1. Now, add the variable Date to create a new model *fitOrgDate*, repeat the process in (c)-(d)-(e) to create a new results table *resultsOrgDate*, and calculate the performance of the new model using the new predictions. Has the model improved? Why or why not?

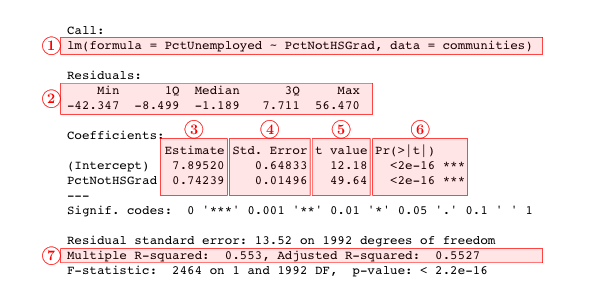
Now, take the same question from an explanatory perspective. Would you keep Date?

1. Remove Date and go back to your original model *fitOrg* from Q9c. This time, remove Unemployment and build a new model *fitOrgNoUn*. As you did in (f), make predictions using the test set and calculate performance. Has the model improved? Why or why not? Is your conclusion about Unemployment the same for both to predict and explain?
2. The finale has to be sweet, right? Instead of using sales, create a log-transformed version, set the seed, split the data, run the model *fitLog*, make predictions, calculate performance.
   1. Have the coefficient estimates and variance explained in DV improved? Compare the model output and performance of *fitLog* with that of  *fitOrg* from Q9c, and discuss.
   2. Check and compare the diagnostics from *fitLog* with those from *fitOrg*, and discuss.

**Bonus question:** Instead of predicting sales, you may also want to create a new dependent variable by dividing the Weekly Sales by store Size (“Sales per square foot” -makes sense if you focus on the utilization of store space, for example). Call it *fitSalesSqFoot*. For this exercise, like in Q10, create a variable, set the seed, split the data, make predictions, calculate performance. What do you think is going on here? Discuss. In addition, in this model, you may want to try removing the variable Size, because your DV is a function of it now. Explore the differences.

**Appendix: How to run a linear regression in R and read the regression output**

The output from summary() may seem overwhelming at first, so let’s break it down one item at a time:



|  |  |  |
| --- | --- | --- |
| **#** | **Item** | **Description** |
| **1** | formula | The formula describes the relationship between the dependent and independent variables. |
| **2** | residuals | The differences between the observed values and the predicted values are called residuals (errors). |
| **3** | coefficients | The coefficients for all the independent variables and the intercept. Using the coefficients we can write down the relationship between the dependent and the independent variables as:  PctUnemployed = 7.90 + ( 0.74 \* PctNotHSGrad )  This tells us that each unit increase in the variable PctNotHSGrad is associated with the increase of the variable PctUnemployed by 0.74, on average. |
| **4** | standard error | The standard error estimates the standard deviation of the sampling distribution of the coefficients in the model. Think of the standard error as a measure of precision for the estimated coefficients. |
| **5** | t-statistic | The t-statistic is obtained by dividing the coefficients by the standard error. |
| **6** | p-value | The p-value for each of the coefficients in the model. Recall that according to the null hypothesis, the value of the coefficient of interest is zero. The p-value tells us whether we can reject the null hypothesis or not. |
| **7** | R2 and adj-R2 | R-squared and adjusted R-squared tell us how much of the variance in our model is accounted for by the independent variables. The adjusted R2 is always smaller than R2 as it takes into account the number of independent variables (and penalized accordingly). |

**Source for the annotated output:** Blumenau and Ali of University College London (2019)

**Important note:** The data used in this lab is from multiple stores and has a time component. Ideally, you would want to take into account this fact. One simple way of doing so is to include *Store* and *Date* in your models as categorical variables (other, possibly more robust ways include panel data regression models). However, because the sample size is small in this dataset, adding so many categorical variables would reduce the power of your analysis. Try adding them if you like, and you will see a warning: “prediction from a rank-deficient fit may be misleading.” The models are informative for educational purposes, but the results may require some caution.