Cosmetic Chemicals Analysis

Final Report

Project Idea

Cosmetics are a daily part of millions of lives. However, many products may contain chemicals linked to cancer, birth defects, or reproductive harm. The California Safe Cosmetics Program (CSCP) collects and reports data from manufacturers, packers, and distributors who sell cosmetics in California that contain hazardous or potentially hazardous chemicals.

Our goal, with this Datafest style project, is to use this dataset to explore patterns in chemical use, brand behavior, and public health risk in the cosmetics industry. By analyzing any visible trends, we aim to highlight which companies and categories are most associated with reportable chemicals We intend to use visualizations such as scatter plots, histograms, and box plots. We also will be using data cleaning techniques like imposition to ensure the data is usable for analysis.

(Dataset Source: https://catalog.data.gov/dataset/chemicals-in-cosmetics-d55bf (https://catalog.data.gov/dataset/chemicals-in-cosmetics-d55bf))

Project Report

First, before we complete any testing/visualization, we must load our data into a readable variable.

```
data <- read.csv("cscpopendata.csv")
head(data)
```

	CD…	ProductName	CS…	C..	Compa
	<int>	<chr>	<int>	<chr>	<
1	2	ULTRA COLOR RICH EXTRA PLUMP LIPSTICK-ALL SHADES	NA		
2	3	Glover's Medicated Shampoo	NA		
3	3	Glover's Medicated Shampoo	NA		
4	4	PRECISION GLIMMER EYE LINER-ALL SHADES �	NA		
5	5	AVON BRILLIANT SHINE LIP GLOSS-ALL SHADES �	NA		
6	6	JILLIAN DEMPSEY FOR AVON CELESTIAL EYESHADOW-ALL SHADES �	NA		

6 rows | 1-6 of 23 columns

Looking at our data, we are provided with a lot of information, most importantly, Product Name, Company Name, Brand Name, Cosmetic Category & Subcategory, Chemical Name, as well as Reported/Discontinued Date. Let’s graph out some of the data to see the different variables. For these graphs, we implemented a new r library called lubridate, which allows us to format the date in a specific way in which we can *extract* a value (e.g. Year).

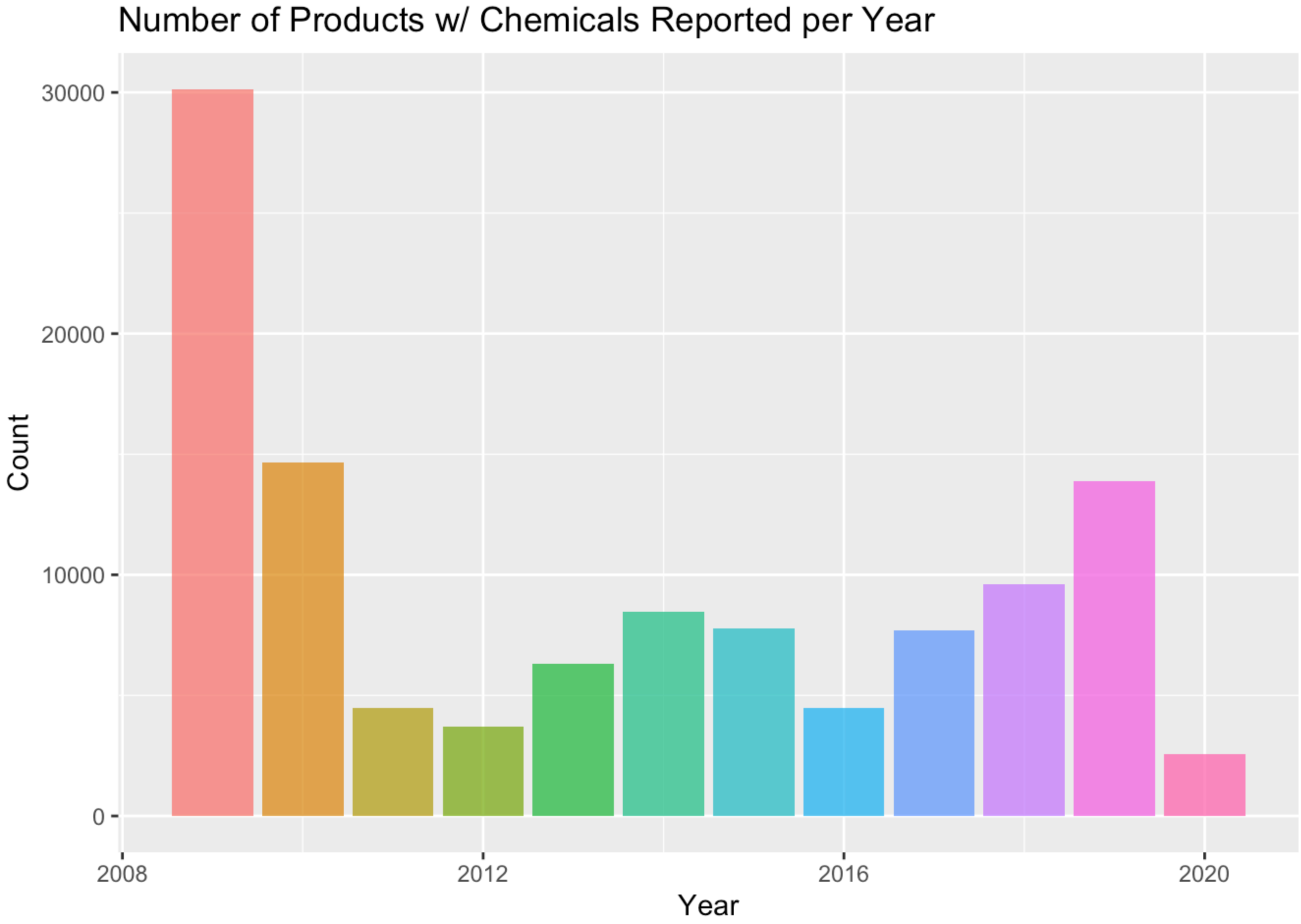
Source: https://lubridate.tidyverse.org/ (https://lubridate.tidyverse.org/)

Figure 1: plot the Number of Products Reported per Year

```
library(ggplot2)
library(tidyverse)
```

```
## ── Attaching core tidyverse packages ──────────────────────── tidyverse 2.0.0 ──
## ✔ dplyr     1.1.4     ✔ readr     2.1.5
## ✔ forcats   1.0.1     ✔ stringr   1.5.2
## ✔ lubridate 1.9.4     ✔ tibble    3.3.0
## ✔ purrr     1.1.0     ✔ tidyr     1.3.1
## ── Conflicts ────────────────────────────────────────── tidyverse_conflicts() ──
## ✖ dplyr::filter() masks stats::filter()
## ✖ dplyr::lag()    masks stats::lag()
## ℹ Use the conflicted package (<http://conflicted.r-lib.org/>) to force all conflicts
to become errors
```

```
# use the lubridate package (from tidyverse) mdy to convert the date to a usable format
data$YearReported <- year(mdy(data$InitialDateReported))

# clean the data - make sure to leave out any without a year reported OR have no chemica
l count?
cleaneddata <- data |> filter(!is.na(YearReported), ChemicalCount > 0)

# Figure 1: plot the Number of Products Reported per Year
ggplot(cleaneddata, aes(x = YearReported, fill = factor(YearReported))) +
  geom_bar(alpha = 0.75, show.legend = F) +
  labs(title = "Number of Products w/ Chemicals Reported per Year",
       x = "Year",
       y = "Count")
```



As we can see from the plot above, we start off at an extreme high with 2009 Cosmetic Chemical Reportings. After 2009, the number of reported products drops a lot and then stays pretty steady from 2010 to 2018, with small ups and downs but nothing major. There's a small bump in 2019, and then a bigger drop in 2020, which was probably affected by the disruptions during the early COVID-19 pandemic. Overall, the steady decline after 2009 is a positive trend. Even though there are small ups and downs, the levels stay much lower than the initial peak, which could point to safer formulations or better oversight.

Let's keep looking…

Since this dataset contains 114,635 observations, plotting out each chemical will be overkill and will not yield an image that can be analyzed further. For this reason, we have decided to plot the top 20 chemicals reported in the dataset and the total number of reports for each. Even so, since the 20 chemical names likely won't fit on the plotted screen, we went ahead and flipped the plot output using reorder() and coord_flip(). The newly loaded library, dplyr, allows us to easily manipulate and clean the dataset. This way, we are able to count occurrences of Chemicals and select the top entries. In addition, we decided to hide the legend, since the fill color only corresponds to chemical names and the legend is unnecessary for interpretation, making the plot cleaner.

Source: https://stackoverflow.com/questions/10868308/regular-expression-a-za-z0-9 (https://stackoverflow.com/questions/10868308/regular-expression-a-za-z0-9)

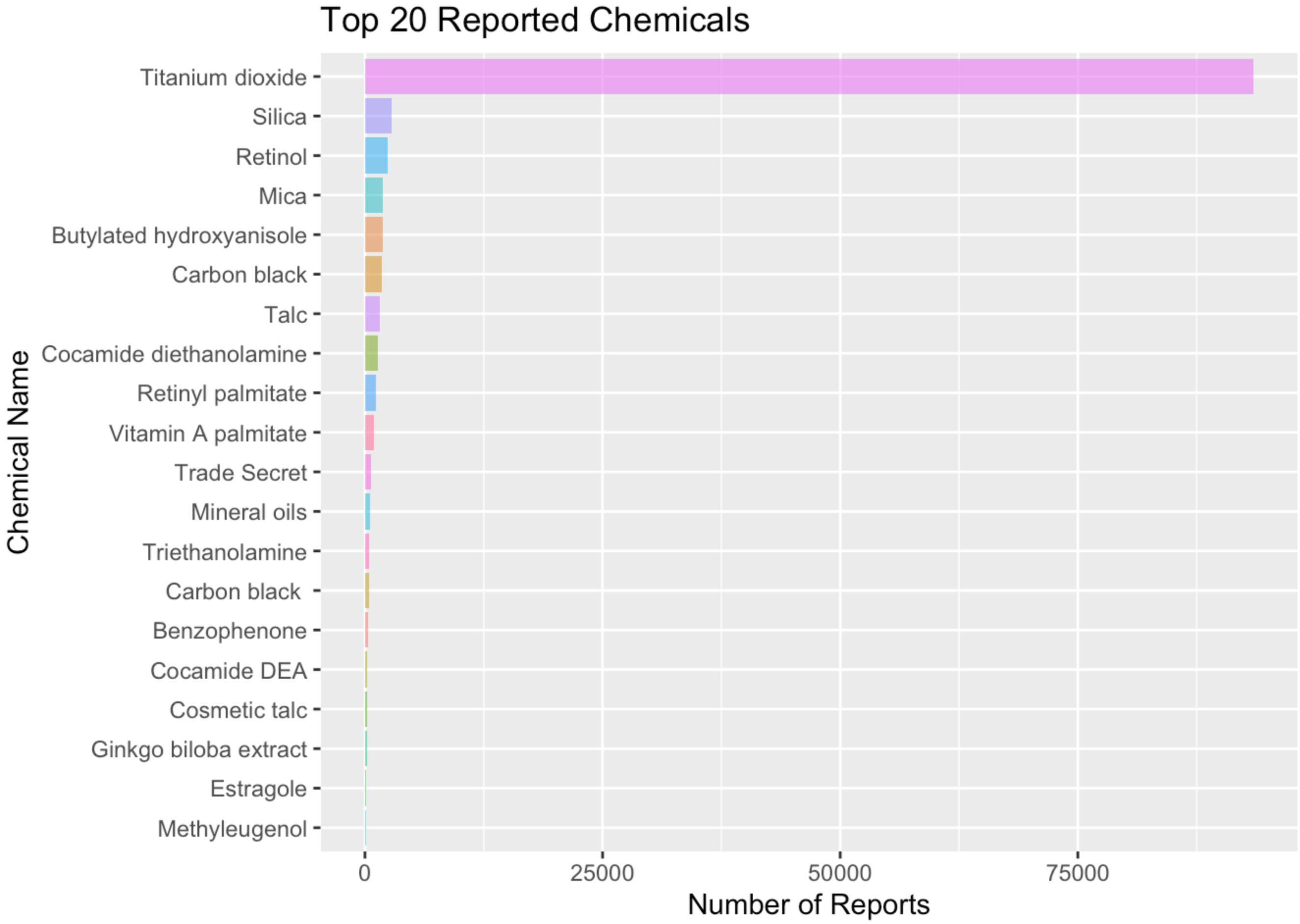
Figure 2: plot the amount of different Chemical reported

```
#1 - use mutate() from dplyr (in tidyverse) to create a new column ChemShort using Chemi
calName
#2 - use sub() from base r to find first character that is NOT a letter or number
   # and everything after that character and substitute it with "", or nothing
#3 - chain and filter to make sure all the values are entered properly to clean the data
cleaneddata2 <- data |>
  mutate(ChemShort = sub("[^A-Za-z0-9 ]+.*", "", ChemicalName)) |>
  filter(!is.na(ChemShort), ChemShort != "")



# use the count() from dplyr to filter through the new ChemShort column and
# count the occurences of each chemical, sort in descending order
chemicalCount <- count(cleaneddata2, ChemShort, sort = TRUE)

# use the slice_head() from dplyr to select the first n rows
top20 <- slice_head(chemicalCount, n = 20)
head(chemicalCount, n = 20)
```

	ChemShort <chr>	**n** <int>
1	Titanium dioxide	93480
2	Silica	2817
3	Retinol	2424
4	Mica	1919
5	Butylated hydroxyanisole	1888
6	Carbon black	1758
7	Talc	1549
8	Cocamide diethanolamine	1397
9	Retinyl palmitate	1181
10	Vitamin A palmitate	971

1-10 of 20 rows Previous **1** 2 Next

```
# plot the top 20 Most Frequent Chemical Reports
ggplot(top20, aes(x = reorder(ChemShort, n), y = n, fill = ChemShort, alpha = 0.75)) +
  geom_col(show.legend = FALSE) +
  coord_flip() +
  labs(
    title = "Top 20 Reported Chemicals",
    x = "Chemical Name",
    y = "Number of Reports"
  )
```



Based on this plot, we can see that Titanium dioxide is the most reported chemical in this dataset with 90,747 more reports than the second highest chemical reporting: Silica. We went ahead and printed out the first 20 values as well so that we can clearly define the top 20 chemical reportings in the dataset. Considering this dataset carries data from 2009-2020, we can say that Titanium dioxide is an extremely common component in the products recorded in this dataset. Other chemicals with larger case reportings are Silica, Retinol, and Mica, reflecting the long term patterns in consumer makeup composition.

Figure 3: plot the type of products that uses the top 20 chemicals

```
#get top 20 names of the chemicals
top20_names <- top20$ChemShort
top20_names
```

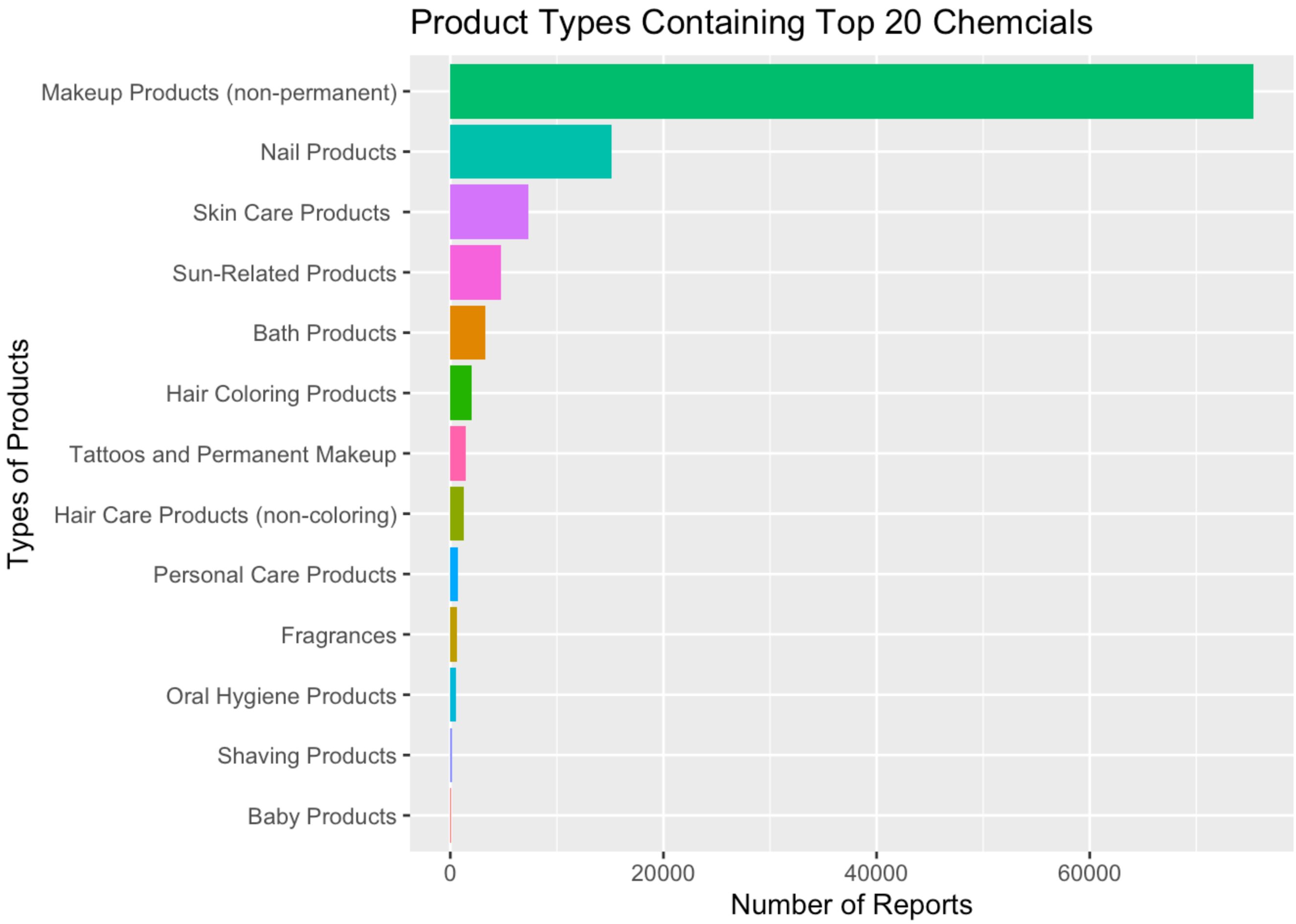
```
##  [1] "Titanium dioxide"          "Silica"
##  [3] "Retinol"                   "Mica"
##  [5] "Butylated hydroxyanisole" "Carbon black"
##  [7] "Talc"                      "Cocamide diethanolamine"
##  [9] "Retinyl palmitate"        "Vitamin A palmitate"
## [11] "Trade Secret"             "Mineral oils"
## [13] "Triethanolamine"          "Carbon black "
## [15] "Benzophenone"             "Cocamide DEA"
## [17] "Cosmetic talc"            "Ginkgo biloba extract"
## [19] "Estragole"                "Methyleugenol"
```

```
#filter to data to only contain the top 20 chemicals
data_top20 <- filter(cleaneddata2, ChemShort %in% top20_names)

#count the products types with the top 20 chemicals
productType <- count(data_top20, PrimaryCategory, sort = TRUE)
productType
```

PrimaryCategory <chr>	**n** <int>
Makeup Products (non-permanent)	75390
Nail Products	15132
Skin Care Products	7280
Sun-Related Products	4795
Bath Products	3267
Hair Coloring Products	1994
Tattoos and Permanent Makeup	1476
Hair Care Products (non-coloring)	1279
Personal Care Products	692
Fragrances	606

1-10 of 13 rows Previous **1** 2 Next

```
ggplot(productType, aes(x = reorder(PrimaryCategory, n), y = n, fill = PrimaryCategory))
+
  geom_col(show.legend = FALSE) +
  coord_flip() +
  labs(
    title = "Product Types Containing Top 20 Chemcials",
    x = "Types of Products",
    y = "Number of Reports"
  )
```



After identifying the top 20 chemicals, we created a plot to see which product categories contain these chemicals most often. Based on the plot, makeup products contain the highest number of chemicals from the top-20 list. This makes sense, as there is a wide variety of makeup products (which we will explore in the next graph). The second highest category is nail products.

Let’s Dive Deeper

Since most chemicals are found in makeup products, we can look more closely at the subcategories of makeup to identify which types contain the most chemicals.

Figure 4: plot the top 20 makeup products that uses the top 20 chemicals

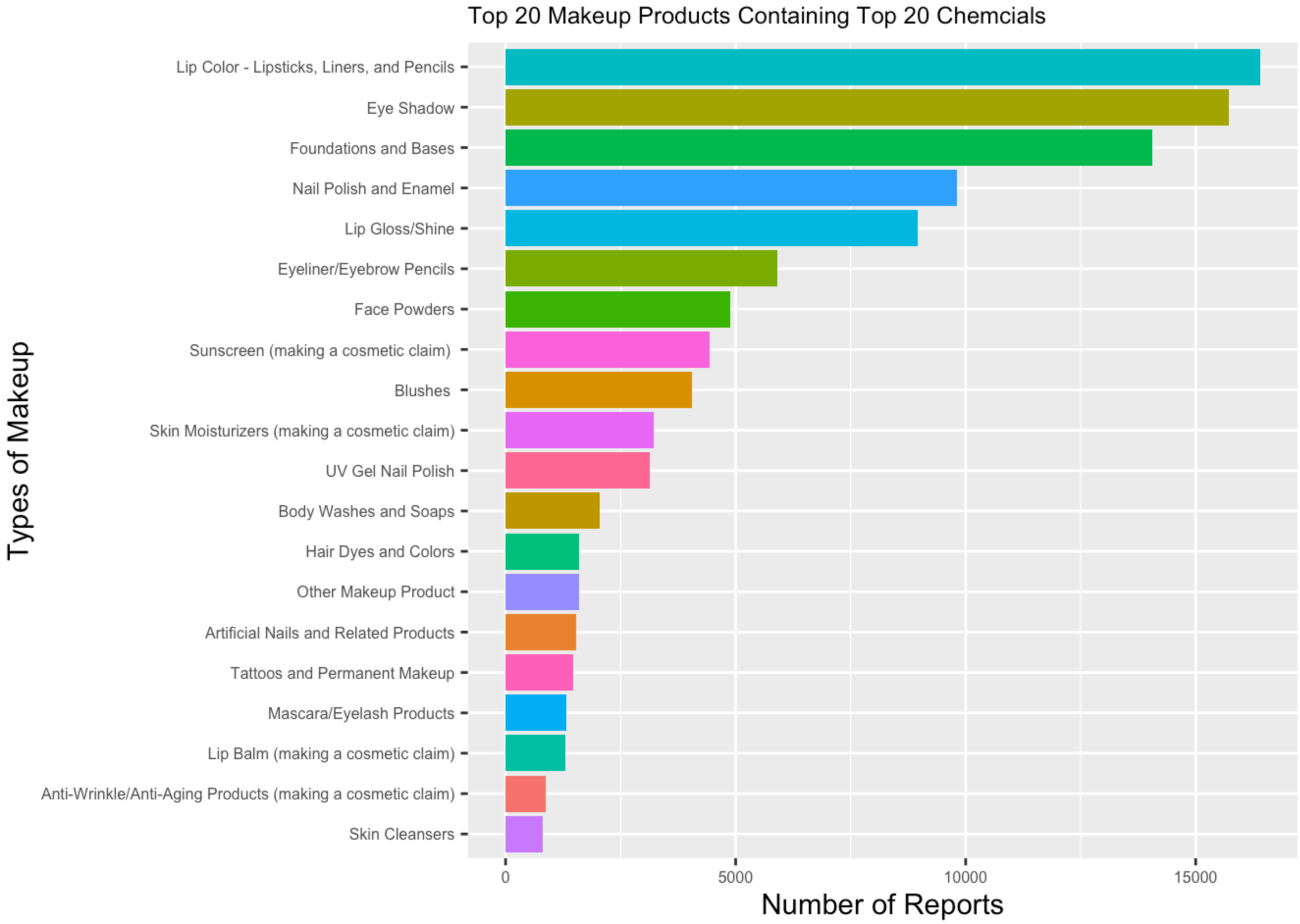
```
#count makeup types with the top 20 chemicals
makeupProduct <- count(data_top20, SubCategory, sort = TRUE)
head(makeupProduct)
```

	SubCategory <chr>	n <int>
1	Lip Color - Lipsticks, Liners, and Pencils	16402
2	Eye Shadow	15710
3	Foundations and Bases	14053

	SubCategory <chr>	n <int>
4	Nail Polish and Enamel	9821
5	Lip Gloss/Shine	8955
6	Eyeliner/Eyebrow Pencils	5914

6 rows

```
#select the first 20 rows (because 89 makeup products is too much to visualize)
top20_makeupProduct <- head(makeupProduct, n = 20)

ggplot(top20_makeupProduct, aes(x = reorder(SubCategory, n), y = n, fill = SubCategory))
+
  geom_col(show.legend = FALSE) +
  coord_flip() +
  labs(
    title = "Top 20 Makeup Products Containing Top 20 Chemcials",
    x = "Types of Makeup",
    y = "Number of Reports"
  ) +
  theme( #changed the font size because the names were too long
    axis.text.x = element_text(size = 6),
    axis.text.y = element_text(size = 6),
    plot.title = element_text(size = 9)
  )
```



Since there are 89 different types of makeup, I have only showcased the top 20 products that contain the top 20 chemicals from the list. From the plot, we see that lip products - such as lipsticks, lip liners, and lip pencils - contain the highest number of chemicals. Closely following are eye shadows, and the third most common are foundations and makeup bases.

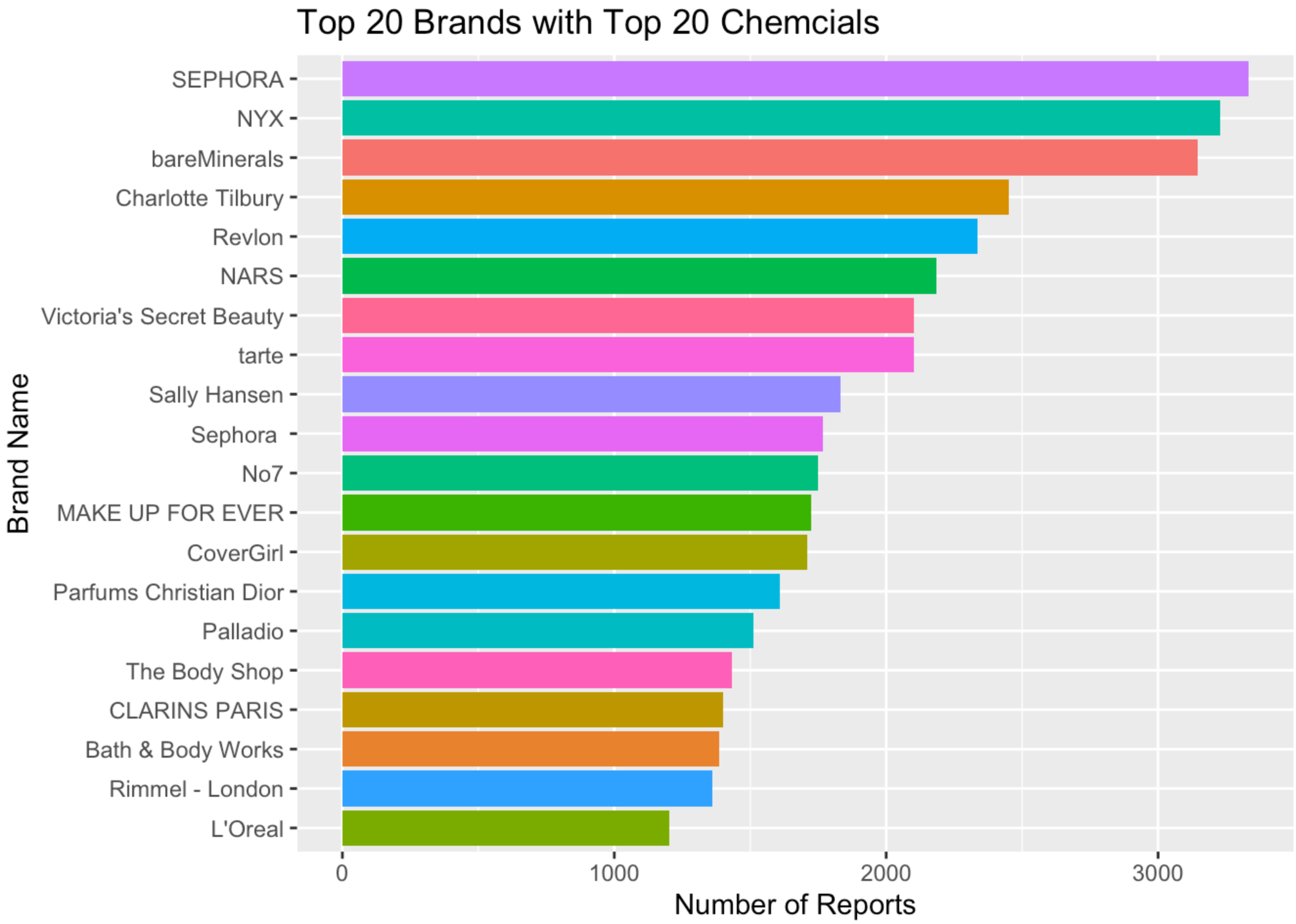
Figure 5: plot the top 20 brands that contain the top 20 chemicals

```
#count brand names with the top 20 chemicals
brands <- count(data_top20, BrandName, sort = TRUE)
head(brands)
```

	BrandName	n
	<chr>	<int>
1	SEPHORA	3333
2	NYX	3227
3	bareMinerals	3146
4	Charlotte Tilbury	2452
5	Revlon	2335
6	NARS	2185

6 rows

```
#select the first 20 rows to get 20 brands
top20brands <- head(brands, n = 20)

ggplot(top20brands, aes(x = reorder(BrandName, n), y = n, fill = BrandName)) +
  geom_col(show.legend = FALSE) +
  coord_flip() +
  labs(
    title = "Top 20 Brands with Top 20 Chemcials",
    x = "Brand Name",
    y = "Number of Reports"
  )
```



Exploring Discontinued Products

In our dataset, we can see that there are some products that were discontinued. This is indicated by a date being in the DiscontinuedDate column. In order to better understand our data, we examined the top 2 subcategories of makeup products to see if there is any analysis to be done on the proportion of discontinued products to continued products.

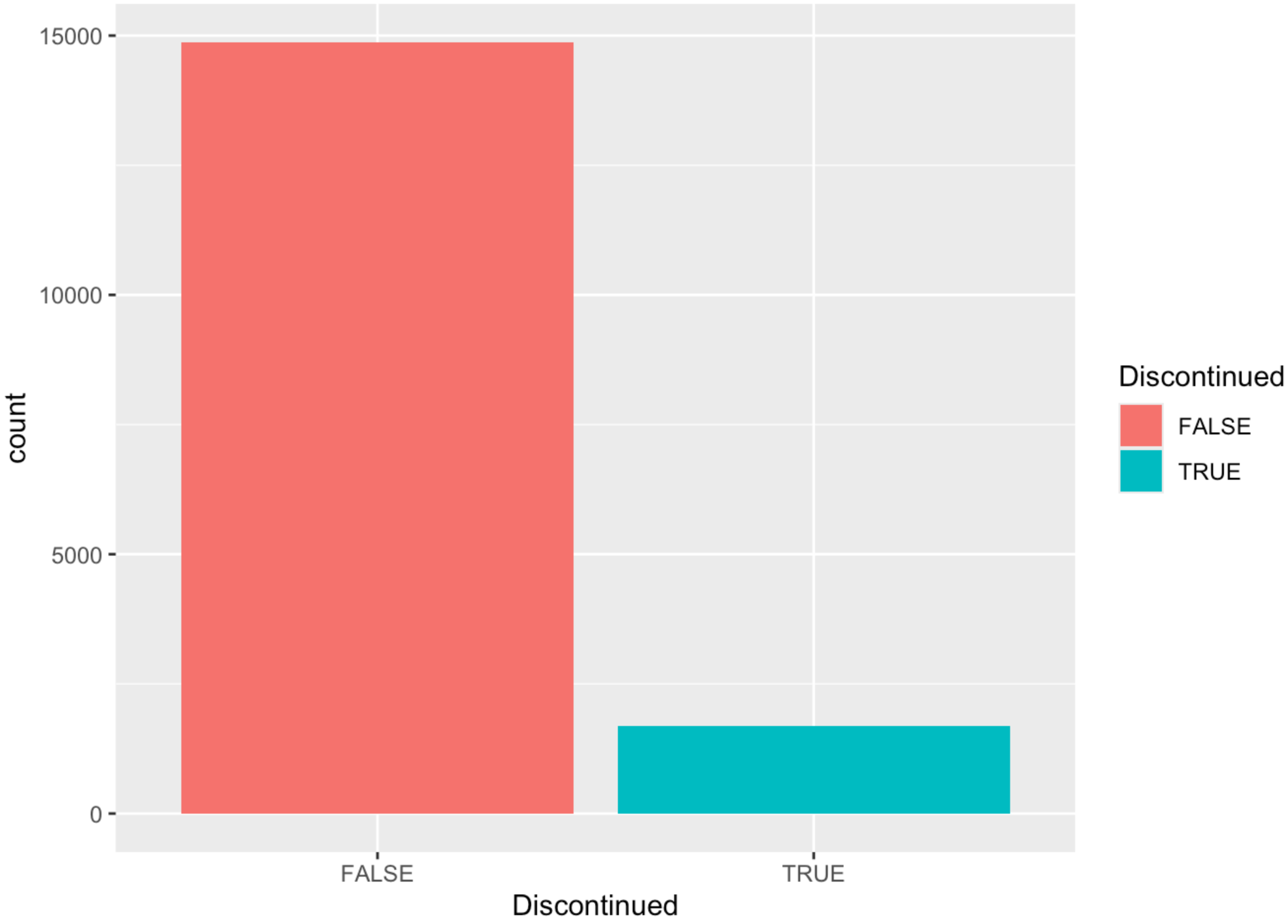
Figure 6: Examining Discontinued Products

```
# Add column for true or false if product was discontinued
data$Discontinued <- data$DiscontinuedDate != ""
head(data)
```

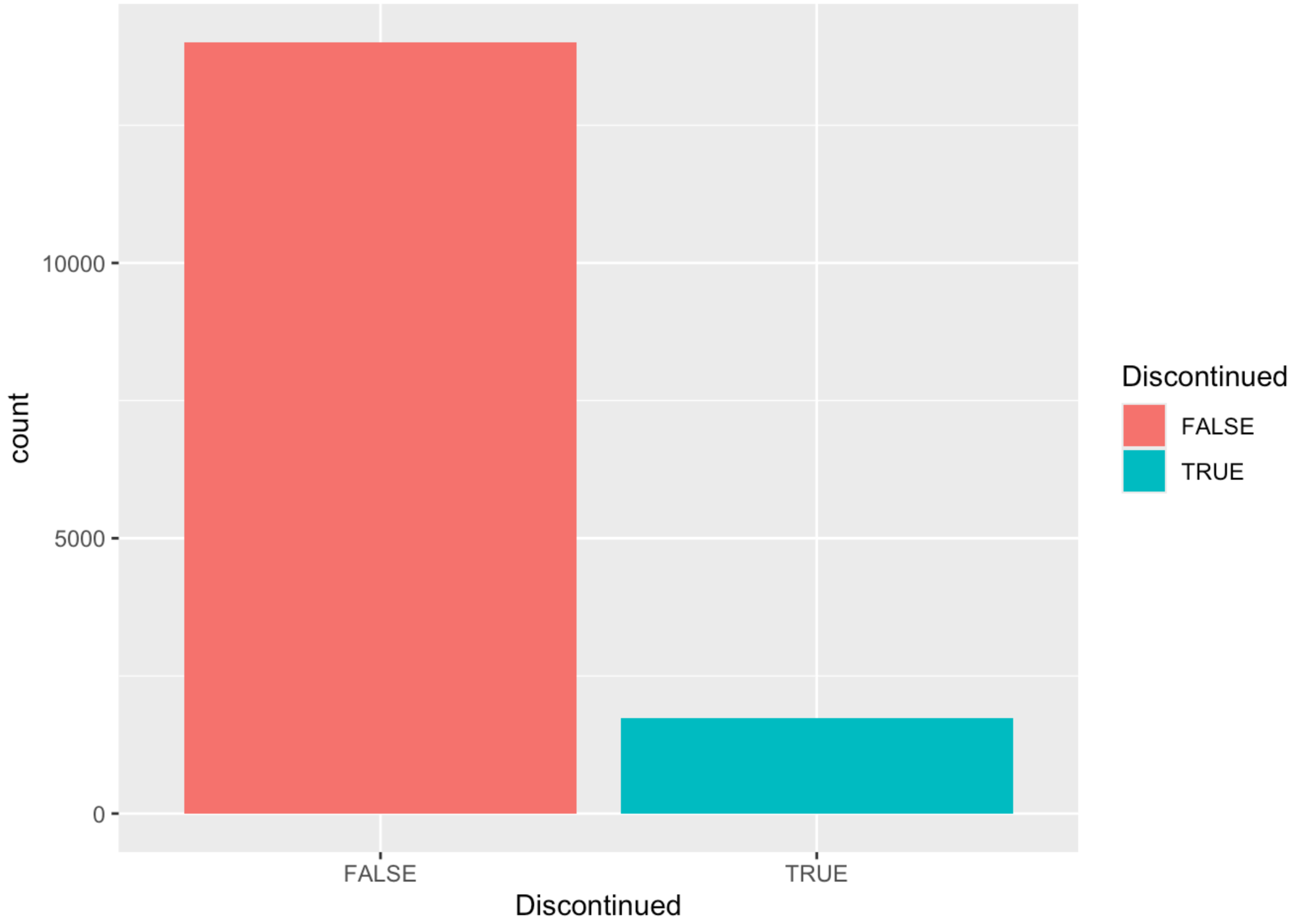
	CD...	ProductName	CS...	C..	Compa
	<int>	<chr>	<int>	<chr>	<
1	2	ULTRA COLOR RICH EXTRA PLUMP LIPSTICK-ALL SHADES	NA		
2	3	Glover's Medicated Shampoo	NA		
3	3	Glover's Medicated Shampoo	NA		
4	4	PRECISION GLIMMER EYE LINER-ALL SHADES �	NA		
5	5	AVON BRILLIANT SHINE LIP GLOSS-ALL SHADES �	NA		
6	6	JILLIAN DEMPSEY FOR AVON CELESTIAL EYESHADOW-ALL SHADES �	NA		

6 rows | 1-6 of 25 columns

```
# Examine lip and eyeshadow products (2 most common subcategories)
lip_products <- data[data$SubCategory == "Lip Color - Lipsticks, Liners, and Pencils", ]
eyeshadow_products <- data[data$SubCategory == "Eye Shadow", ]

# Plot to examine proportion of discontinued products to continued products
ggplot(lip_products) +
  geom_bar(aes(x = Discontinued, fill = Discontinued))
```



```
ggplot(eyeshadow_products) +
  geom_bar(aes(x = Discontinued, fill = Discontinued))
```



These bargraphs show that the proportion of continued products to discontinued products is relatively the same for the top 2 subcategories of makeup products and it appears that a majority of products don't get discontinued.

Now it's time to do some tests

Two Sample T-Test

Looking at our plotted visualizations above, we can see that the top two product types containing chemicals are makeup products and nail products. Let's compute a two sample t-test to determine whether the average number of chemicals reported in makeup products differs from the average number of chemicals reported in nail products.

Before doing so, however, we must remember to clean our data. This includes removing any categories with missing information such as chemical names, chemical counts, or products with no (0) chemicals reported (probably due to unreported info). This way, our test will stay accurate to the reported values. For this particular test, our null hypothesis will be that the average chemicals reported between makeup and nail products are equal.

Source: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/t.test (https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/t.test)

```
t_testdata <- data |>
  subset(!is.na(PrimaryCategory)) |>     # subset rows without missing categories
  subset(!is.na(ChemicalCount)) |>       # subset rows without a missing chemical count
  subset(ChemicalCount > 0)              # some rows have data that says chemical count
= 0, remove them

# create a new variable to store the categories that we want to extract
categories <- c("Makeup Products (non-permanent)", "Nail Products")

# extract all values within the listed categories
t_testdata <- t_testdata |> subset(PrimaryCategory %in% categories)

# column value conversion
t_testdata$PrimaryCategory <- factor(
  # select the column that we are targeting
  t_testdata$PrimaryCategory,
  # specify what should be included and in what order
  levels = categories,
  # provide new labels
  labels = c("Makeup", "Nail")
)

# conduct the two sample t-test
t_test_result <- t.test(ChemicalCount ~ PrimaryCategory,
                        data = t_testdata,
                        alternative = "two.sided")

t_test_result
```

```
## 
##  Welch Two Sample t-test
## 
## data:  ChemicalCount by PrimaryCategory
## t = 3.8171, df = 23854, p-value = 0.0001354
## alternative hypothesis: true difference in means between group Makeup and group Nail
is not equal to 0
## 95 percent confidence interval:
##  0.009496352 0.029542773
## sample estimates:
## mean in group Makeup   mean in group Nail 
##              1.32220              1.30268
```

Based on the test results above, we can see that there is a statistically significant difference between the mean number of chemicals in Makeup products as opposed to Nail Products. The mean chemical count in Makeup was higher at 1.32220 than for Nail Products at 1.30268. With a p-value of 0.0001354, which is less than 0.05, the data provides strong evidence to reject the null hypothesis that the mean chemical counts in makeup and nail products are equal.

Chi-Square Goodness-of-Fit Test

Looking back at our earlier plots, we can tell that makeup products make up a large portion of the reported items in the dataset. To dig deeper, let’s take a look at the different types of makeup products to see whether they are all reported at similar rates, or if some subcategories show up much more often than others. We used a chi-square goodness-of-fit test to check whether the top ten makeup SUB-categories are equally represented.

Source: https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/Chisquare (https://www.rdocumentation.org/packages/stats/versions/3.6.2/topics/Chisquare)

```
# count the observations per makeup subcategory
subcategory_count <- table(data_top20$SubCategory)

# subcategories sorted by frequency
subcategory_sorted <- sort(subcategory_count, decreasing = TRUE)

# split to find the top 10
top10_subcategory_count <- head(subcategory_sorted, 10)
top10_subcategory_count
```

```
## 
##  Lip Color - Lipsticks, Liners, and Pencils
##                                       16402
##                                  Eye Shadow
##                                       15710
##                       Foundations and Bases
##                                       14053
##                      Nail Polish and Enamel
##                                        9821
##                             Lip Gloss/Shine
##                                        8955
##                    Eyeliner/Eyebrow Pencils
##                                        5914
##                                Face Powders
##                                        4891
##        Sunscreen (making a cosmetic claim) 
##                                        4443
##                                    Blushes 
##                                        4058
## Skin Moisturizers (making a cosmetic claim)
##                                        3220
```

```
# list values needed to complete testing
observed <- top10_subcategory_count
n <- length(observed)
expected <- sum(observed) / n

# complete the test
chi_sq_test <- sum((observed - expected)^2 / expected)
chi_sq_test
```

```
## [1] 26339.79
```

```
# to find p-value, calculate df (degree of freedom)
df <- n - 1
# use pre-built pchisq function to find the p-value
p <- 1 - pchisq(chi_sq_test, df)
p
```

```
## [1] 0
```

The chi-square goodness-of-fit test determined that there is a chi-square value of 26,340, which means that the observed counts are largely different from what is expected if all subcategories were reported equally. This means that some makeup types appear much more often than others in our dataset. With a very small p-value, so small that the computer cannot complete the calculation, we have strong evidence that the makeup subcategories are not equally represented in the dataset, and that certain products therefore have a larger effect on the reports.

Correlation Testing

In the first plot, we displayed how many products were reported each year. Instead of focusing on the number of products, this time, we want to examine whether the number of chemicals in products has changed over time. More specifically, we are testing to see if “newer” reported products tend to have more or less chemicals than the older reported ones. This could shed some light on potential shifts in manufacturing practices, safety regulations, or consumer preferences. To do so, we are going to use a correlation test to measure the relationship between year and chemical count.

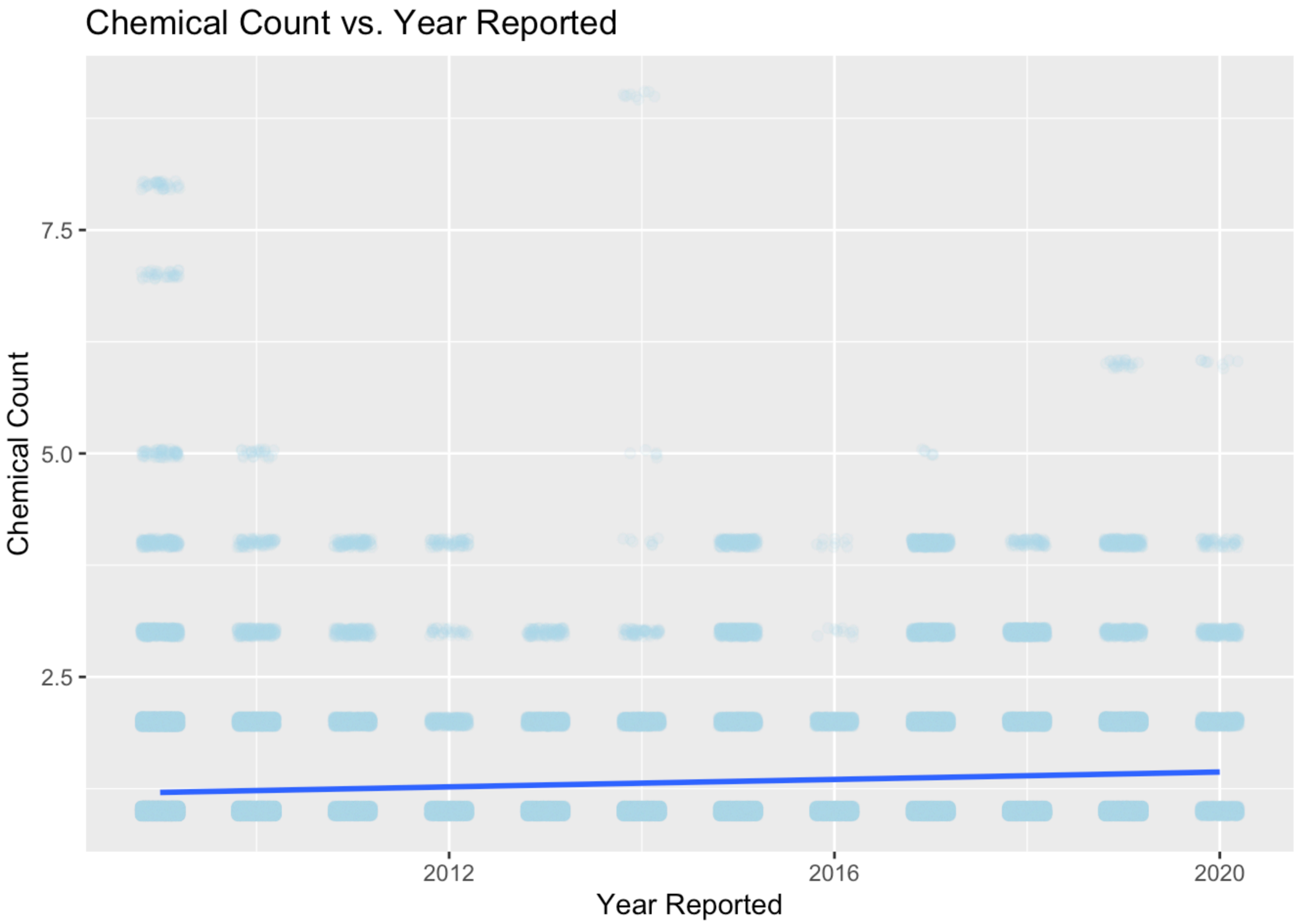
```
# clean all the data
corrtest_data <- data |>
  filter(!is.na(YearReported)) |>
  filter(!is.na(ChemicalCount)) |>
  filter(ChemicalCount > 0)

# find values to complete correlation test
x <- corrtest_data$YearReported
y <- corrtest_data$ChemicalCount

# find the covariance
co_variance <- sum((x - mean(x)) * (y - mean(y)))
# find the standard deviation
std_dev <- sqrt(sum((x - mean(x))^2) * sum((y - mean(y))^2))
# find the correlation co-efficient: r = covariance/stddev
r <- co_variance / std_dev
r
```

```
## [1] 0.1272165
```

Figure 7: Correlation Test Graph

```
ggplot(corrtest_data, aes(x = YearReported, y = ChemicalCount)) +
  geom_jitter(width = 0.2, height = 0.05, alpha = 0.1, color = "lightblue") +
  geom_smooth(method = "lm", se = FALSE) +
  labs(
    title = "Chemical Count vs. Year Reported",
    x = "Year Reported",
    y = "Chemical Count"
  )
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



The results of our correlation test, 0.1272165, reveals a weak, but positive relationship between year and chemical count. This means that products reported in more recent years tend to have slightly more chemicals than older ones. The effect is pretty small, though, so it's more of a subtle trend than a big change. I also plotted the data with a jitter since most chemical reportings are the same amount.

Linear Regression Analysis

As stated before, the first figure shows us the variation of products reported with chemicals each year from 2009 to 2020. We can further examine this pattern by performing a linear regression analysis on the data to determine whether or not reporting frequencies of products with chemicals in them changes over time.

```
# Clean data to get only products that have a year reported and have chemicals
chem_data <- data |>
  filter(!is.na(YearReported)) |>
  filter(!is.na(ChemicalCount)) |>
  filter(ChemicalCount > 0)

# Data for linear regression - year reported and counts of products reported per
# year
linreg_data <- chem_data |>
  group_by(YearReported) |>
  summarize(report_freq = n())
head(linreg_data)
```

YearReported	report_freq
<dbl>	<int>
2009	30135
2010	14645
2011	4497
2012	3722
2013	6318
2014	8461

6 rows

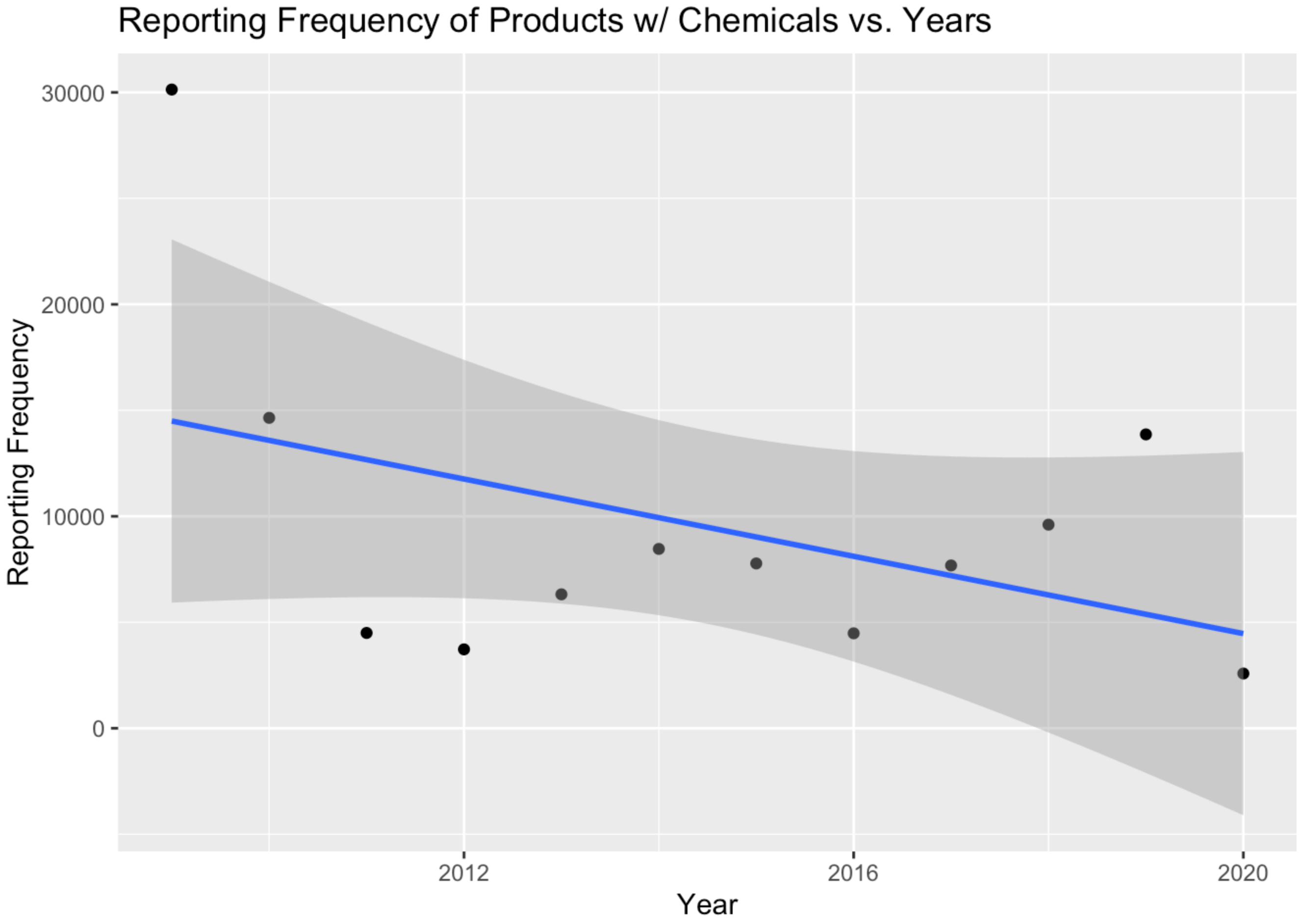
```
# Create linear regression model
lin_model <- lm(report_freq ~ YearReported, data = linreg_data)

# Get summary of model
summary(lin_model)
```

```
## 
## Call:
## lm(formula = report_freq ~ YearReported, data = linreg_data)
## 
## Residuals:
##    Min     1Q Median     3Q    Max 
##  -8175  -3858  -1362   1625  15640 
## 
## Coefficients:
##               Estimate Std. Error t value Pr(>|t|)
## (Intercept)  1846169.2  1192897.3   1.548    0.153
## YearReported    -911.7      592.2  -1.540    0.155
## 
## Residual standard error: 7081 on 10 degrees of freedom
## Multiple R-squared:  0.1916, Adjusted R-squared:  0.1108 
## F-statistic: 2.371 on 1 and 10 DF,  p-value: 0.1547
```

Figure 8: Linear Regression Analysis

```
# Plot frequency of products reported yearly against year
ggplot(linreg_data, aes(x = YearReported, y = report_freq)) +
  geom_point() +
  geom_smooth(method = "lm") +
  xlab("Year") +
  ylab("Reporting Frequency") +
  ggtitle("Reporting Frequency of Products w/ Chemicals vs. Years")
```

```
## `geom_smooth()` using formula = 'y ~ x'
```



The results of this linear regression analysis reveal that reporting frequencies over time do not closely follow a linear trend and therefore using Year Reported to predict the frequency of products with chemicals being produced is not a great indicator. The adjusted R^2 value being 0.1108 reflects the fact that the Year Reported of a products does not explain much of the variability in the frequency of products with chemicals being reported each year.

One-Sample T-Test

Examining Figure 3, we can see that makeup products is the most common category for products to have reported chemicals in them. However, this does not tell us how many chemicals are typically reported in makeup products. Therefore, we can run a one-sample t-test on products that fall under the makeup products category to determine whether or not the mean chemical count in makeup products is similar to the mean chemical count in the entire dataset, which is our large sample.

```
# Determine mean chemical count for entire dataset
chem_mean <- mean(data$ChemicalCount, na.rm = T)
chem_mean
```

```
## [1] 1.288359
```

```
# Clean Data
single_t_data <- data |>
  filter(!is.na(YearReported)) |>
  filter(!is.na(ChemicalCount))

# Take sample data - makeup products
makeup_data <- single_t_data[single_t_data$PrimaryCategory == "Makeup Products (non-perm
anent)", ]

# Perform t-test
one_t_result <- t.test(x = makeup_data$ChemicalCount, mu = chem_mean)

# Print summary of t-test
one_t_result
```

```
##
##  One Sample t-test
##
## data:  makeup_data$ChemicalCount
## t = 13.597, df = 75826, p-value < 2.2e-16
## alternative hypothesis: true mean is not equal to 1.288359
## 95 percent confidence interval:
##  1.315262 1.324325
## sample estimates:
## mean of x
##  1.319794
```

The results of this one sample t-test tell us that the average number of chemicals reported in makeup products is not equal to the mean number of chemicals in all the products of the dataset. The average of the dataset is about 1.288 chemicals per product. However, the mean number of checmicals in makeup products is about 1.30, with a 95% confidence interval of about 1.315 to 1.324. Additionally the p-value of the test is 2.2e-16, which is much smaller than 0.05. This indicates that the difference between the average number of chemicals in makeup products and the number of chemicals in all the products in the dataset is not due to chance.

Logistic Regression

Another question we can examine with this data is whether or not the presence of chemicals in a product is a good indicator of whether or not that product will be discontinued. While we saw that many products do not get discontinued in Figure 6, we did not examine the influence of chemicals in products on their discontinued status. To do this, we can perform a logistic regression.

```
# Load in data for logistic regression
logreg_data <- data

# Determine whether or not product has reported chemicals
logreg_data$HasChemical <- ifelse(data$ChemicalCount > 0, TRUE, FALSE)

# Determine whether or not product has reported discontinued date
logreg_data$Discontinued <- ifelse(data$DiscontinuedDate == "", FALSE, TRUE)

# Make logistic model
log_model <- glm(Discontinued ~ HasChemical,
                 data = logreg_data,
                 family = binomial)

# Examine summary of logistic regression model
summary(log_model)
```

```
##
## Call:
## glm(formula = Discontinued ~ HasChemical, family = binomial,
##     data = logreg_data)
##
## Coefficients:
##                 Estimate Std. Error z value Pr(>|z|)
## (Intercept)     -1.11866    0.07874  -14.21   <2e-16 ***
## HasChemicalTRUE -0.95498    0.07930  -12.04   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## (Dispersion parameter for binomial family taken to be 1)
##
##     Null deviance: 80734  on 114634  degrees of freedom
## Residual deviance: 80612  on 114633  degrees of freedom
## AIC: 80616
##
## Number of Fisher Scoring iterations: 4
```

```
# Convert log-odds to odds
discontinued_no_chem_odds <- exp(coef(log_model)["(Intercept)"])

# Convert odds to probability
discontinued_no_chem_prob <- discontinued_no_chem_odds / (1 + discontinued_no_chem_odds)
str_c("Probability of product without chemicals being discontinued: ", round(discontinue
d_no_chem_prob, 3))
```

```
## [1] "Probability of product without chemicals being discontinued: 0.246"
```

```
# Compute probability of products with chemicals being discontinued
joint_odds <- coef(log_model)["(Intercept)"] + coef(log_model)["HasChemicalTRUE"]

prob_chem <- exp(joint_odds) / (1 + exp(joint_odds))

str_c("Probability of product with chemicals being discontinued: ", round(prob_chem, 3))
```

```
## [1] "Probability of product with chemicals being discontinued: 0.112"
```

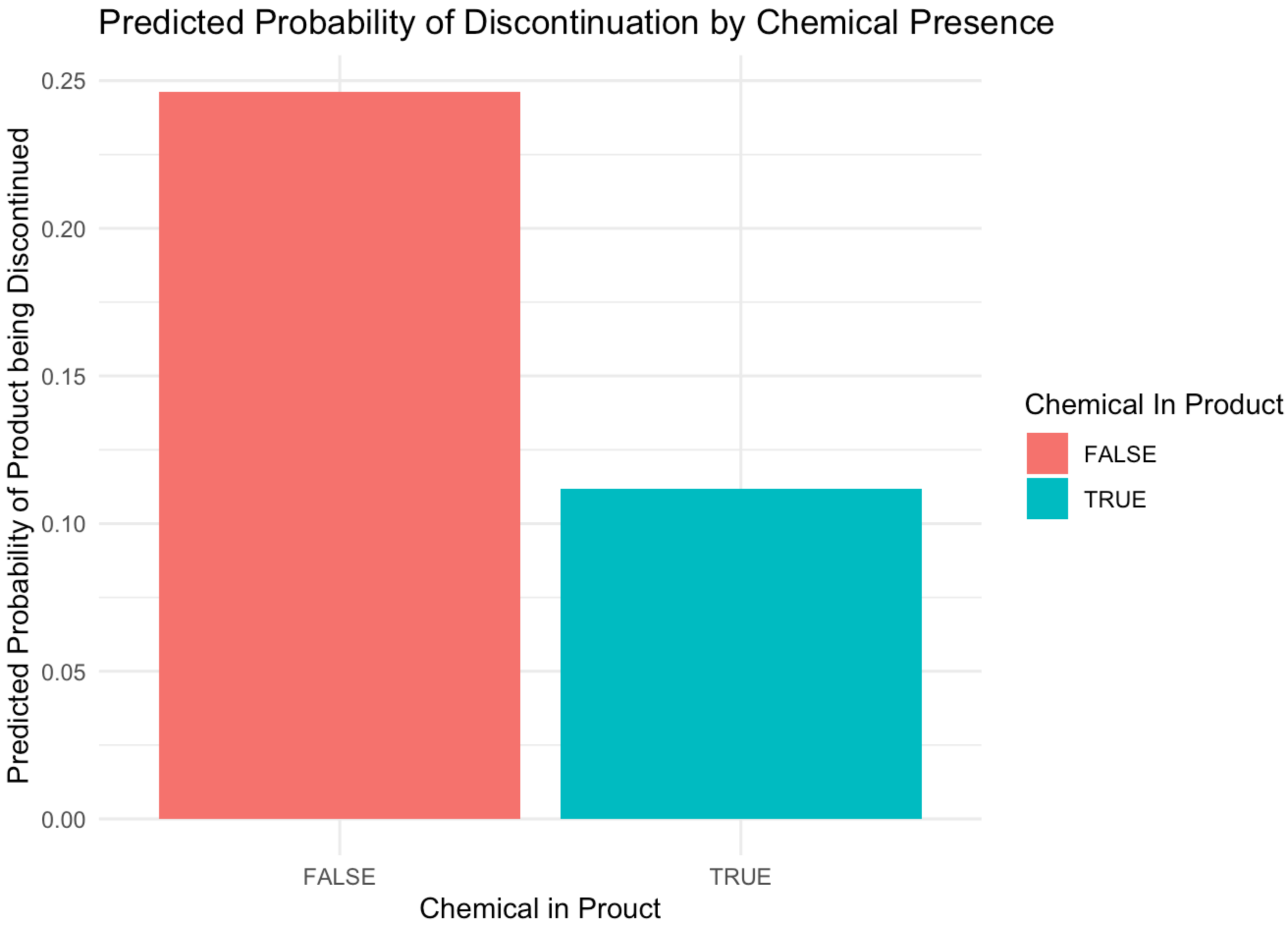
Figure 9: Probability of Discontinuation of Products Based on Chemical Presence

```
# Plot predicted probabilities from logistic regression model
# Prediction dataset
prediction_df <- data.frame(
  HasChemical = c(FALSE, TRUE)
)

# Add predicted probabilities
prediction_df$pred_prob <- predict(log_model, newdata = prediction_df, type = "respons
e")
head(prediction_df)
```

	HasChemical <lgl>	pred_prob <dbl>
1	FALSE	0.2462601
2	TRUE	0.1116854

2 rows

```
# Plot bar graph showing predicted probability of discontinuation
ggplot(prediction_df, aes(x = HasChemical, y = pred_prob, fill = HasChemical)) +
  geom_col() +
  xlab("Chemical in Prouct") +
  ylab("Predicted Probability of Product being Discontinued") +
  ggtitle("Predicted Probability of Discontinuation by Chemical Presence") +
  scale_fill_discrete(name = "Chemical In Product") +
  theme_minimal()
```



The results of this logistic regression indicate that products without chemicals have about a 24.6% chance of being discontinued. Through some calculations, we see that products with chemicals have about a 11.2% chance of being discontinued. Additionally, the p-value for both coefficients are extremely small, being <2e-16. This means that the presence of chemicals in products is statistically significant, so it is an indicator that products with chemicals are less likely to be discontinued. This finding is interesting to us because we thought that products without chemicals would be less likely to be discontinued.

Source: https://stats.oarc.ucla.edu/r/dae/logit-regression/ (https://stats.oarc.ucla.edu/r/dae/logit-regression/)

Test for Independence (Chi-Square)

Using the Chi-Square Test of Independence, we want to see whether certain brands focus on specific product categories.

Source: https://statsandr.com/blog/chi-square-test-of-independence-in-r/ (https://statsandr.com/blog/chi-square-test-of-independence-in-r/)

```
#count number of observations for Brand Name
brand_Name <- count(data, BrandName, sort = TRUE)

#identify the top 10
top10_Brand <- head(brand_Name, n= 10)
#get the Brand Names
top10_BrandName <- top10_Brand$BrandName

#filter the data to only include the top 10 Brand Names
new_data <- filter(data, BrandName %in% top10_BrandName)
head(new_data)
```

	CD...	ProductName	CS...	C..	Co
	<int>	<chr>	<int>	<chr>	
1	46	Colorstay 12 Hour Eye Shadow Quad- Copper Spice-champagne 99	NA		
2	57	Colorstay 12 Hour Eye Shadow Quad- peach 98	NA		
3	58	Colorstay 12 Hour Eye Shadow Quad- copper 97	NA		
4	59	Colorstay 12 Hour Eye Shadow Quad- bronze 96	NA		
5	60	Colorstay 12 Hour Eye Shadow Quad- Coffee bean-iced mocha 95	NA		
6	61	Colorstay 12 Hour Eye Shadow Quad- dark brown 97	NA		

6 rows | 1-6 of 25 columns

```
#chi-square test
test <- chisq.test(table(new_data$BrandName, new_data$PrimaryCategory))
```

```
## Warning in chisq.test(table(new_data$BrandName, new_data$PrimaryCategory)):
## Chi-squared approximation may be incorrect
```

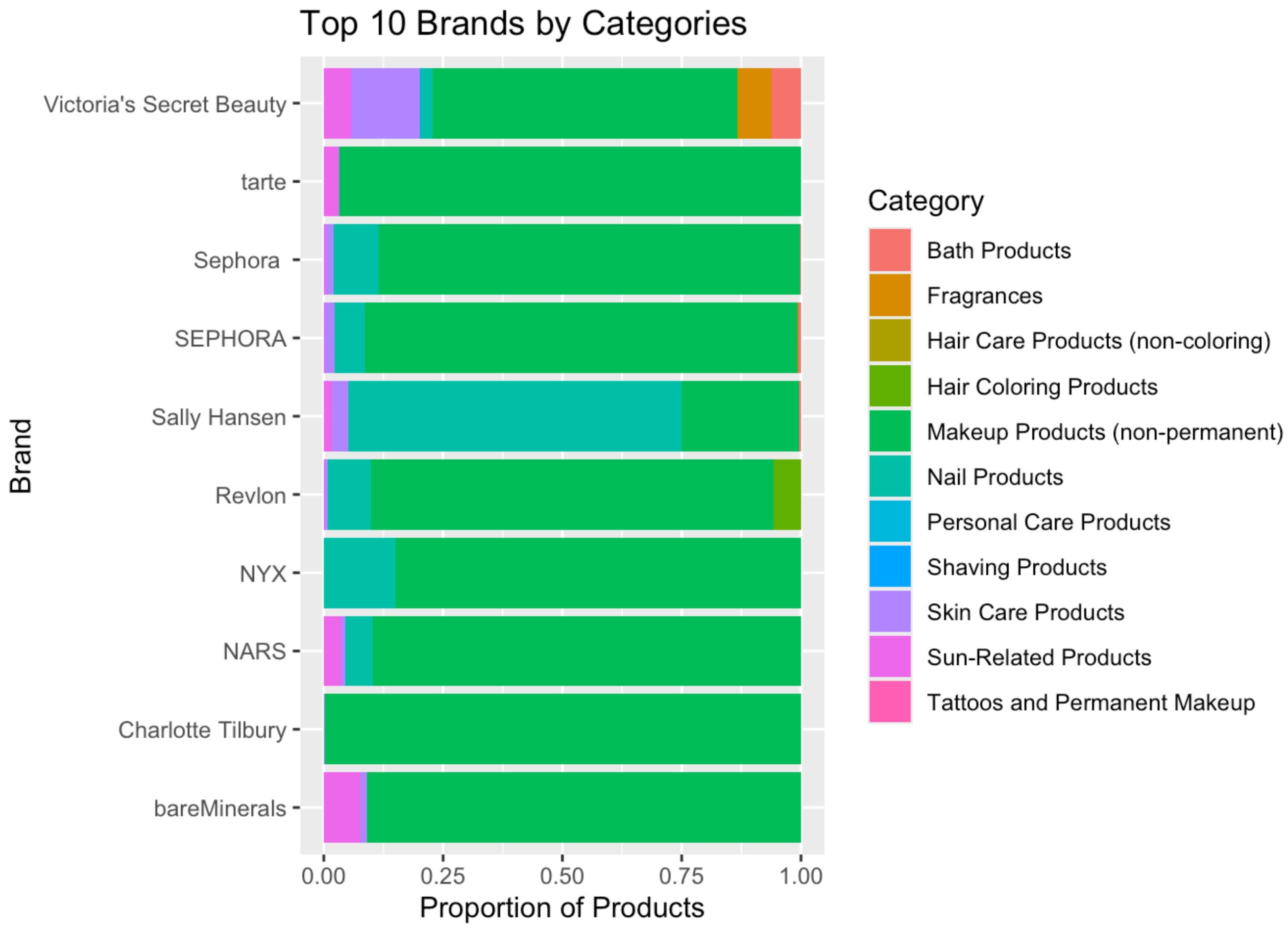
```
test
```

```
##
##  Pearson's Chi-squared test
##
## data:  table(new_data$BrandName, new_data$PrimaryCategory)
## X-squared = 14823, df = 90, p-value < 2.2e-16
```

Based on the p-value, we reject the null hypothesis. This means that there is a significant association between brand and product category. In other words, certain brands tend to specialize in specific types of products. To visualize this, we graphed the top 10 brands by category below.

Figure 10: Top 10 Brands by Categories

```
ggplot(new_data, aes(x = BrandName, fill = PrimaryCategory)) +
  geom_bar(position = "fill") +
  coord_flip() +
  labs(
    title = "Top 10 Brands by Categories",
    x = "Brand",
    y = "Proportion of Products",
    fill = "Category"
  )
```



Based on this graph we found that brands such as Sephora primarily focus on makeup products. As shown in Figure 3, makeup products contain many of the most frequently reported chemicals. This helps explain why in Figure 5, Sephora appears among the top brands associated with the top 20 chemicals. This does not necessarily mean Sephora uses more chemicals per product. Instead, it reflects that makeup-focused brands naturally appear more often in chemical reports because makeup is the category in which these chemicals are most common.

ANOVA

Using ANOVA, we want to test whether the mean ChemicalCount differs across PrimaryCategory groups.

Source: https://www.geeksforgeeks.org/maths/anova-formula/ (https://www.geeksforgeeks.org/maths/anova-formula/)

```r
#clean data
clean_data <- filter(data, !is.na(ChemicalCount), ChemicalCount > 0)

#find overall mean of Chemical Count
overall_mean <- mean(clean_data$ChemicalCount)


#get a summary of the data
summary <- clean_data |>
  group_by(PrimaryCategory) |>
  summarize(
    n_obs = n(),
    avg_chemCounts = mean(ChemicalCount)
  )
summary
```

PrimaryCategory <chr>	**n_obs** <int>	**avg_chemCounts** <dbl>
Baby Products	46	1.043478
Bath Products	3268	1.134639
Fragrances	591	1.169205
Hair Care Products (non-coloring)	1507	1.522230
Hair Coloring Products	2055	1.032117
Makeup Products (non-permanent)	75689	1.322200
Nail Products	15333	1.302680
Oral Hygiene Products	513	1.038986
Personal Care Products	665	1.153383
Shaving Products	217	1.248848

1-10 of 13 rows Previous **1** 2 Next

```
#calculating F-Value by following the steps from website above
#compute sum of squares (SS)
SSB <- sum(summary$n_obs*(summary$avg_chemCounts-overall_mean)^2)
SST <- sum((clean_data$ChemicalCount - overall_mean)^2)
SSE <- SST - SSB

#calculate df
k <- 13 #There are 13 categories
N <- nrow(clean_data)

df1 <- k - 1
df2 <- N- k

#calculate mean squares (MS)
MSB <- SSB/df1
MSE <- SSE/df2

#F-statistic
F_value <- MSB/MSE
F_value
```

```
## [1] 120.7663
```

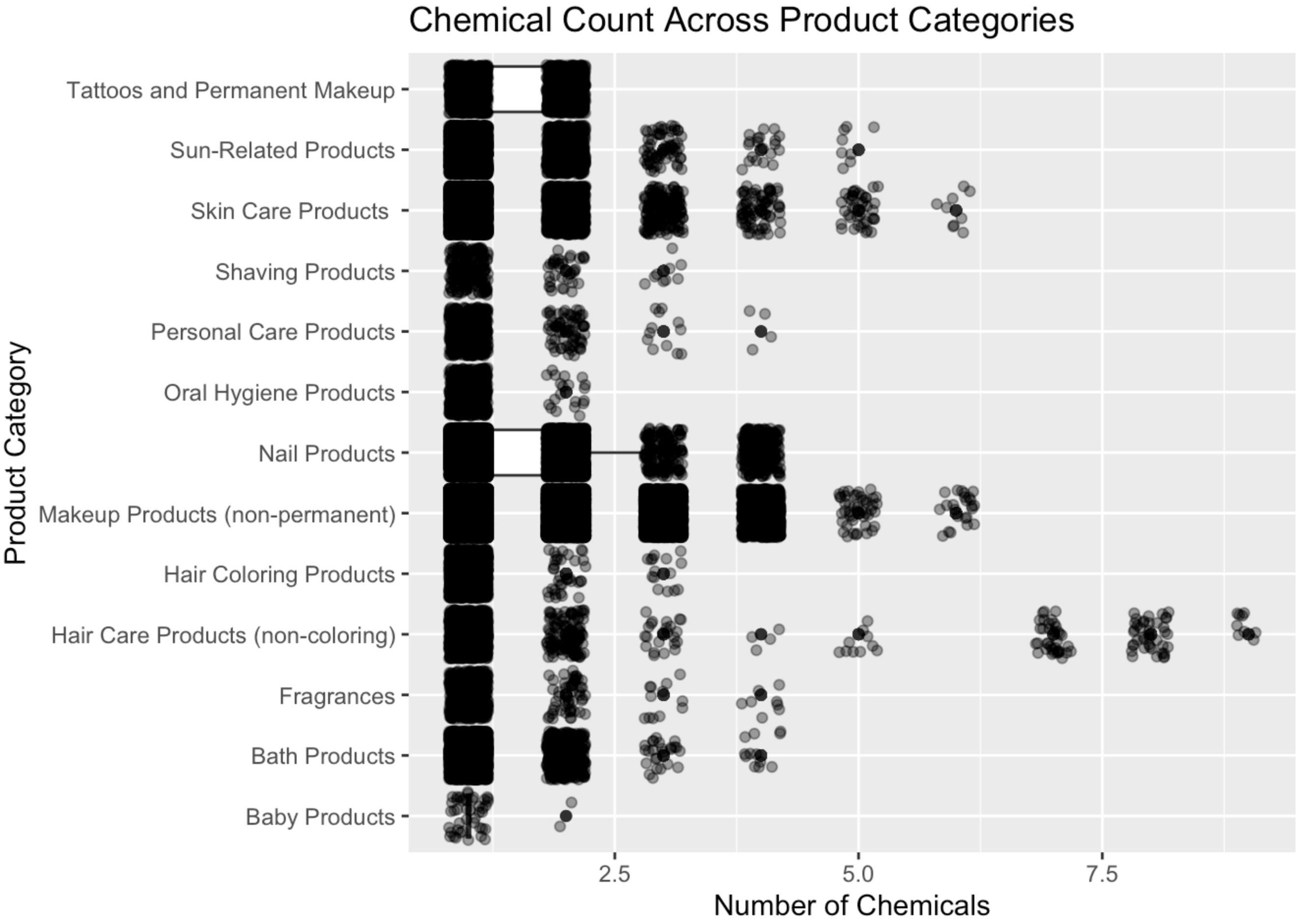
```
#p-value
p <- 1 - pf(F_value, df1, df2)
p
```

```
## [1] 0
```

Since the p-value is extremely small, we reject the null hypothesis that all categories have the same average chemical count. This suggests that certain categories, such as makeup or hair coloring products, tend to contain more chemicals on average compared to others. We can also use a boxplot to visualize this.

Figure 11: Chemical Count Across Product Categories

```
ggplot(clean_data, aes(x = PrimaryCategory, y = ChemicalCount)) +
  geom_boxplot(show.legend = FALSE) +
  geom_jitter(height = 0.2, alpha = 0.4) +
  coord_flip() +
  labs(
    title = "Chemical Count Across Product Categories",
    x = "Product Category",
    y = "Number of Chemicals"
  )
```



As we can see, the number of chemicals varies across product categories. While many categories have products containing only one chemical, other products such as Hair Coloring Products, Makeup Products, and Skin Care Products show wider distributions with higher chemical counts.

2 Prop Z-Test

We want to see if discontinued products are more likely to contain titanium dioxide than non discontinued products.

Source: https://r-coder.com/prop-test-r/ (https://r-coder.com/prop-test-r/)

```
#create a vector that checks if product contains titanium
containsTD <- data$ChemicalName == "Titanium dioxide"
#create a vector that checks if product is discontinued
discontinued <- data$DiscontinuedDate != ""

# discontinued
disc_with_TD <- sum(containsTD & discontinued)
disc_total <- sum(discontinued)

# not discontinued
nondisc_with_TD <- sum(containsTD & !discontinued)
nondisc_total <- sum(!discontinued)

#prop test
prop_test <- prop.test(
  x = c(disc_with_TD, nondisc_with_TD),
  n = c(disc_total, nondisc_total),
  correct = F
)

prop_test
```

```
## 
##  2-sample test for equality of proportions without continuity correction
## 
## data:  c(disc_with_TD, nondisc_with_TD) out of c(disc_total, nondisc_total)
## X-squared = 34.713, df = 1, p-value = 3.821e-09
## alternative hypothesis: two.sided
## 95 percent confidence interval:
##  -0.02868254 -0.01401068
## sample estimates:
##    prop 1    prop 2 
## 0.7965170 0.8178636
```

The two-proportion test indicates that the proportion of products containing Titanium Dioxide differs statistically significantly between discontinued and non-discontinued products. Around, 79.65% of discontinued products contained Titanium Dioxide, compared to 81.79% of non-discontinued products. Although the difference is statistically significant, it is relatively small, as both groups show a high presence of Titanium Dioxide. This supports our earlier observation in Figure 6 that most products containing chemicals are not discontinued. This suggests that the presence of reported chemicals, such as Titanium Dioxide, does not strongly predict whether a product will be discontinued.