

```

#Machine Vibration Trend Analysis Project
#load necessary packages
library(tidyverse)

#check working directory and change it if needed to csv file location
getwd()

#COLLECT AND WRANGLE DATA
#read csv files into R
Motor_1A_velocity_rms<- read_csv("Motor_1A_velocity_rms_dataset.csv")

#check the tibble structure to ensure proper import
str(Motor_1A_velocity_rms)

#convert the timestamp column to mdy_hms format using lubridate
Motor_1A_velocity_rms$timestamp <- mdy_hms(Motor_1A_velocity_rms$timestamp)

#add column to tibble with date (extracted from timestamp) for grouping in qcc
Motor_1A_velocity_rms$date <- as.Date(Motor_1A_velocity_rms$timestamp)

#Inspect tibble for new column titled "date"
str(Motor_1A_velocity_rms$date)

#add columns for year, month, day of week to facilitate aggregations
Motor_1A_velocity_rms$year <- format(as.Date(Motor_1A_velocity_rms$date),"%Y")
Motor_1A_velocity_rms$month_num <- format(as.Date(Motor_1A_velocity_rms$date),"%m")
Motor_1A_velocity_rms$month_abbrev <- format(as.Date(Motor_1A_velocity_rms$date),"%b")
Motor_1A_velocity_rms$week <- week(Motor_1A_velocity_rms$date)
Motor_1A_velocity_rms$day_of_week <- format(as.Date(Motor_1A_velocity_rms$date),"%A")

#put days_of_week in proper order of weekdays, starting with sunday
Motor_1A_velocity_rms$day_of_week <- ordered(Motor_1A_velocity_rms$day_of_week, levels
=
      c("Sunday", "Monday", "Tuesday",
        "Wednesday", "Thursday", "Friday",
        "Saturday"))

#DESCRIPTIVE ANALYSIS
#calculate summary statistics for dataset, ungrouped by time period
summary(Motor_1A_velocity_rms)

#use attach function to permit calling columns by name without tibble reference
attach(Motor_1A_velocity_rms)

#calculate the mean and quantiles (0, 25, 50, 75, 1) for each month and week.
#Note since multiple variables are being calculated, we use reframe instead of aggregate
motor_1V_max_summary_stats_monthly <- Motor_1A_velocity_rms %>%
  group_by(month_num) %>%
  reframe(max = max(`Motor 1V (inch/sec)`),
          mean = mean(`Motor 1V (inch/sec)`),
          min = min(`Motor 1V (inch/sec)`),

```

```

    quant_25th = quantile(`Motor 1V (inch/sec)`, 0.25),
    quant_75th = quantile(`Motor 1V (inch/sec)`, 0.75)
  )
print(motor_1V_max_summary_stats_monthly)

motor_1V_max_summary_stats_weekly <- Motor_1A_velocity_rms %>%
  group_by(week) %>%
  reframe(max = max(`Motor 1V (inch/sec)`),
    mean = mean(`Motor 1V (inch/sec)`),
    min = min(`Motor 1V (inch/sec)`),
    quant_25th = quantile(`Motor 1V (inch/sec)`, 0.25),
    quant_75th = quantile(`Motor 1V (inch/sec)`, 0.75)
  )
print(motor_1V_max_summary_stats_weekly)

#Plotting (line, scatter, box and whisker)

#line plot showing mean and max values for each month
line_plot_motor_1V_monthly_stats <- motor_1V_max_summary_stats_monthly %>%
  ggplot(aes(x = month_num)) + geom_point(aes(y = mean, color = "mean")) +
  geom_line(aes(y = mean, color = "mean", group = 1)) +
  geom_point(aes(y = max, color = "max")) +
  geom_line(aes(y = max, color = "max", group = 1)) +
  labs(title = "Monthly Maximum and Mean Velocity", x = "Month", color = "") +
  theme(legend.position = "bottom")

print(line_plot_motor_1V_monthly_stats)

#line plot showing mean and max values for each week
line_plot_motor_1V_weekly_stats <- motor_1V_max_summary_stats_weekly %>%
  ggplot(aes(x = week)) + geom_point(aes(y = mean, color = "mean")) +
  geom_line(aes(y = mean, color = "mean", group = 1)) +
  geom_point(aes(y = max, color = "max")) +
  geom_line(aes(y = max, color = "max", group = 1)) +
  labs(title = "Weekly Maximum and Mean Velocity", x = "Month", color = "") +
  scale_x_continuous(n.breaks = 10) + theme(legend.position = "bottom")

print(line_plot_motor_1V_weekly_stats)

#Scatter plot showing all points on each day of a given month (set in the filter command)
scatter_plot_motor_1V_custom_dates <- Motor_1A_velocity_rms %>%
  filter(month_num == "01") %>%
  ggplot(aes(x = date, y = `Motor 1V (inch/sec)`) + geom_point(position = "dodge")
scatter_plot_motor_1V_custom_dates

#box plots for all months in the data set
box_plot_motor_1V_allmonths <- Motor_1A_velocity_rms %>%
  ggplot(aes(x = date, y = `Motor 1V (inch/sec)`, fill = month_num)) +
  geom_boxplot() + facet_wrap(~month_num, scale = "free") +
  labs(title = "Box Plots: All Months", x = "Month") +
  theme(legend.position = "none")

```

```
box_plot_motor_1V_allmonths
```

```
#box plot for a specific date range in the data set
box_plot_motor_1V_custom_dates <- Motor_1A_velocity_rms %>%
  filter(between(date, as.Date("2024-09-01"), as.Date("2024-09-07"))) %>%
  ggplot(aes(x = date, y = `Motor 1V (inch/sec)`)) + geom_boxplot() +
  labs(title = "Box Plot", x = "Date") +
  theme(legend.position = "none")
box_plot_motor_1V_custom_dates
```

```
#Plotting quality control charts
#load qcc package
library(qcc)
```

```
#format dataset using qcc.groups to produce xbar charts with qcc function
##all months
motor_1V_bymonth <- qcc.groups(`Motor 1V (inch/sec)`, month_num)
##All weeks
motor_1V_byweek <- qcc.groups(`Motor 1V (inch/sec)`, week)
```

```
##Syntax to select specific month(s) data for qcc.groups to pass to qcc charting
```

```
#Single month selection
motor_1V_Jan <-filter(Motor_1A_velocity_rms, month_abbrev == "Jan")
motor_1V_Jan_qccgroups <- qcc.groups(motor_1V_Jan$`Motor 1V (inch/sec)`,
motor_1V_Jan$week)
#multiple month selection
motor_1V_Jan_to_Mar <-filter(Motor_1A_velocity_rms, month_abbrev == "Jan" |
                             month_abbrev == "Feb" |
                             month_abbrev == "Mar")
motor_1V_Jan_to_Mar_qccgroups <- qcc.groups(motor_1V_Jan_to_Mar$`Motor 1V (inch/sec)`,
motor_1V_Jan_to_Mar$week)
```

```
#Create xbar charts using qcc package from qcc.groups functions above
qcc(motor_1V_bymonth, type = "xbar")
```

```
qcc(motor_1V_byweek, type = "xbar")
```

```
qcc(motor_1V_Jan_qccgroups, type = "xbar")
```

```
qcc(motor_1V_Jan_to_Mar_qccgroups, type = "xbar")
```

```
#EXPORT data frame file
Motor_1A_velocity_rms_R_processed <- Motor_1A_velocity_rms
write.csv(Motor_1A_velocity_rms_R_processed, file =
"Motor_1A_velocity_rms_R_processed.csv")
```