Data 624 Project 2

5/18/2025

Prompt

This is role playing. I am your new boss. I am in charge of production at ABC Beverage and you are a team of data scientists reporting to me. My leadership has told me that new regulations are requiring us to understand our manufacturing process, the predictive factors and be able to report to them our predictive model of PH.

Please use the historical data set I am providing. Build and report the factors in BOTH a technical and non-technical report. I like to use Word and Excel. Please provide your non-technical report in a business friendly readable document and your predictions in an Excel readable format. The technical report should show clearly the models you tested and how you selected your final approach.

Approach

Data Exploration

Load and View Data

```
training <- read.csv("https://raw.githubusercontent.com/Stevee-G/Data624/refs/heads/main/Project2/Traini
testing <- read.csv("https://raw.githubusercontent.com/Stevee-G/Data624/refs/heads/main/Project2/TestDat

str(training)
```

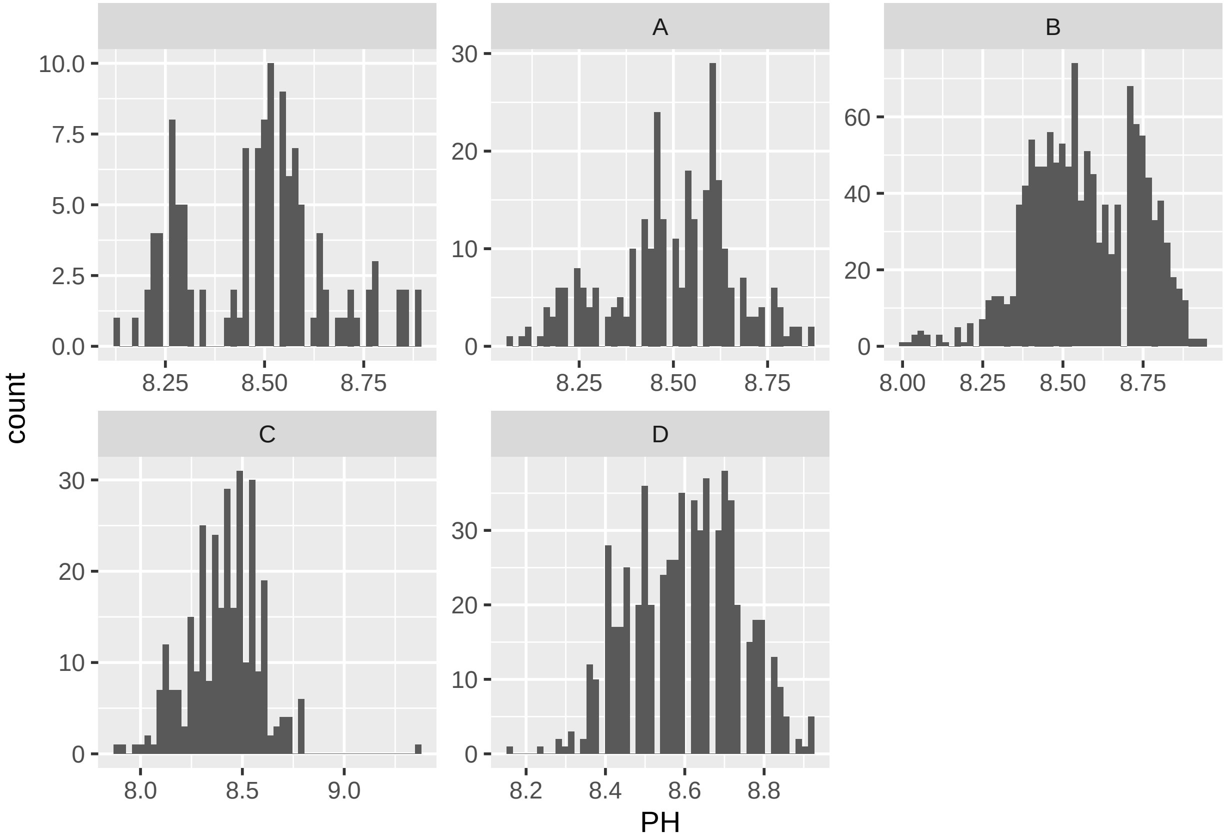
```
## 'data.frame':    2571 obs. of  33 variables:
##  $ Brand.Code       : chr  "B" "A" "B" "A" ...
##  $ Carb.Volume      : num  5.34 5.43 5.29 5.44 5.49 ...
##  $ Fill.Ounces      : num  24 24 24.1 24 24.3 ...
##  $ PC.Volume        : num  0.263 0.239 0.263 0.293 0.111 ...
##  $ Carb.Pressure    : num  68.2 68.4 70.8 63 67.2 66.6 64.2 67.6 64.2 72 ...
##  $ Carb.Temp        : num  141 140 145 133 137 ...
##  $ PSC              : num  0.104 0.124 0.09 NA 0.026 0.09 0.128 0.154 0.132 0.014 ...
##  $ PSC.Fill         : num  0.26 0.22 0.34 0.42 0.16 0.24 0.4 0.34 0.12 0.24 ...
##  $ PSC.CO2          : num  0.04 0.04 0.16 0.04 0.12 0.04 0.04 0.04 0.14 0.06 ...
##  $ Mnf.Flow         : num  -100 -100 -100 -100 -100 -100 -100 -100 -100 -100 ...
##  $ Carb.Pressure1   : num  119 122 120 115 118 ...
##  $ Fill.Pressure    : num  46 46 46 46.4 45.8 45.6 51.8 46.8 46 45.2 ...
##  $ Hyd.Pressure1    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Hyd.Pressure2    : num  NA NA NA 0 0 0 0 0 0 0 ...
##  $ Hyd.Pressure3    : num  NA NA NA 0 0 0 0 0 0 0 ...
##  $ Hyd.Pressure4    : int  118 106 82 92 92 116 124 132 90 108 ...
##  $ Filler.Level     : num  121 119 120 118 119 ...
##  $ Filler.Speed     : int  4002 3986 4020 4012 4010 4014 NA 1004 4014 4028 ...
##  $ Temperature      : num  66 67.6 67 65.6 65.6 66.2 65.8 65.2 65.4 66.6 ...
##  $ Usage.cont       : num  16.2 19.9 17.8 17.4 17.7 ...
##  $ Carb.Flow        : int  2932 3144 2914 3062 3054 2948 30 684 2902 3038 ...
##  $ Density          : num  0.88 0.92 1.58 1.54 1.54 1.52 0.84 0.84 0.9 0.9 ...
##  $ MFR              : num  725 727 735 731 723 ...
##  $ Balling          : num  1.4 1.5 3.14 3.04 3.04 ...
##  $ Pressure.Vacuum  : num  -4 -4 -3.8 -4.4 -4.4 -4.4 -4.4 -4.4 -4.4 -4.4 ...
##  $ PH               : num  8.36 8.26 8.94 8.24 8.26 8.32 8.4 8.38 8.38 8.5 ...
##  $ Oxygen.Filler    : num  0.022 0.026 0.024 0.03 0.03 0.024 0.066 0.046 0.064 0.022 ...
##  $ Bowl.Setpoint    : int  120 120 120 120 120 120 120 120 120 120 ...
##  $ Pressure.Setpoint: num  46.4 46.8 46.6 46 46 46 46 46 46 46 ...
##  $ Air.Pressurer    : num  143 143 142 146 146 ...
##  $ Alch.Rel         : num  6.58 6.56 7.66 7.14 7.14 7.16 6.54 6.52 6.52 6.54 ...
##  $ Carb.Rel         : num  5.32 5.3 5.84 5.42 5.44 5.44 5.38 5.34 5.34 5.34 ...
##  $ Balling.Lvl      : num  1.48 1.56 3.28 3.04 3.04 3.02 1.44 1.44 1.44 1.38 ...
```

```
str(testing)
```

```
## ’data.frame':    267 obs. of  33 variables:
##  $ Brand.Code       : chr  "D" "A" "B" "B" ...
##  $ Carb.Volume      : num  5.48 5.39 5.29 5.27 5.41 ...
##  $ Fill.Ounces      : num  24 24 23.9 23.9 24.2 ...
##  $ PC.Volume        : num  0.27 0.227 0.303 0.186 0.16 ...
##  $ Carb.Pressure    : num  65.4 63.2 66.4 64.8 69.4 73.4 65.2 67.4 66.8 72.6 ...
##  $ Carb.Temp        : num  135 135 140 139 142 ...
##  $ PSC              : num  0.236 0.042 0.068 0.004 0.04 0.078 0.088 0.076 0.246 0.146 ...
##  $ PSC.Fill         : num  0.4 0.22 0.1 0.2 0.3 0.22 0.14 0.1 0.48 0.1 ...
##  $ PSC.CO2          : num  0.04 0.08 0.02 0.02 0.06 NA 0 0.04 0.04 0.02 ...
##  $ Mnf.Flow         : num  -100 -100 -100 -100 -100 -100 -100 -100 -100 -100 ...
##  $ Carb.Pressure1   : num  117 119 120 125 115 ...
##  $ Fill.Pressure    : num  46 46.2 45.8 40 51.4 46.4 46.2 40 43.8 40.8 ...
##  $ Hyd.Pressure1    : num  0 0 0 0 0 0 0 0 0 0 ...
##  $ Hyd.Pressure2    : num  NA 0 0 0 0 0 0 0 0 0 ...
##  $ Hyd.Pressure3    : num  NA 0 0 0 0 0 0 0 0 0 ...
##  $ Hyd.Pressure4    : int  96 112 98 132 94 94 108 108 110 106 ...
##  $ Filler.Level     : num  129 120 119 120 116 ...
##  $ Filler.Speed     : int  3986 4012 4010 NA 4018 4010 4010 NA 4010 1006 ...
##  $ Temperature      : num  66 65.6 65.6 74.4 66.4 66.6 66.8 NA 65.8 66 ...
##  $ Usage.cont       : num  21.7 17.6 24.2 18.1 21.3 ...
##  $ Carb.Flow        : int  2950 2916 3056 28 3214 3064 3042 1972 2502 28 ...
##  $ Density          : num  0.88 1.5 0.9 0.74 0.88 0.84 1.48 1.6 1.52 1.48 ...
##  $ MFR              : num  728 736 735 NA 752 ...
##  $ Balling          : num  1.4 2.94 1.45 1.06 1.4 ...
##  $ Pressure.Vacuum  : num  -3.8 -4.4 -4.2 -4 -4 -3.8 -4.2 -4.4 -4.4 -4.2 ...
##  $ PH               : logi  NA NA NA NA NA NA ...
##  $ Oxygen.Filler    : num  0.022 0.03 0.046 NA 0.082 0.064 0.042 0.096 0.046 0.096 ...
##  $ Bowl.Setpoint    : int  130 120 120 120 120 120 120 120 120 120 ...
##  $ Pressure.Setpoint: num  45.2 46 46 46 50 46 46 46 46 46 ...
##  $ Air.Pressurer    : num  143 147 147 146 146 ...
##  $ Alch.Rel         : num  6.56 7.14 6.52 6.48 6.5 6.5 7.18 7.16 7.14 7.78 ...
##  $ Carb.Rel         : num  5.34 5.58 5.34 5.5 5.38 5.42 5.46 5.42 5.44 5.52 ...
##  $ Balling.Lvl      : num  1.48 3.04 1.46 1.48 1.46 1.44 3.02 3 3.1 3.12 ...
```

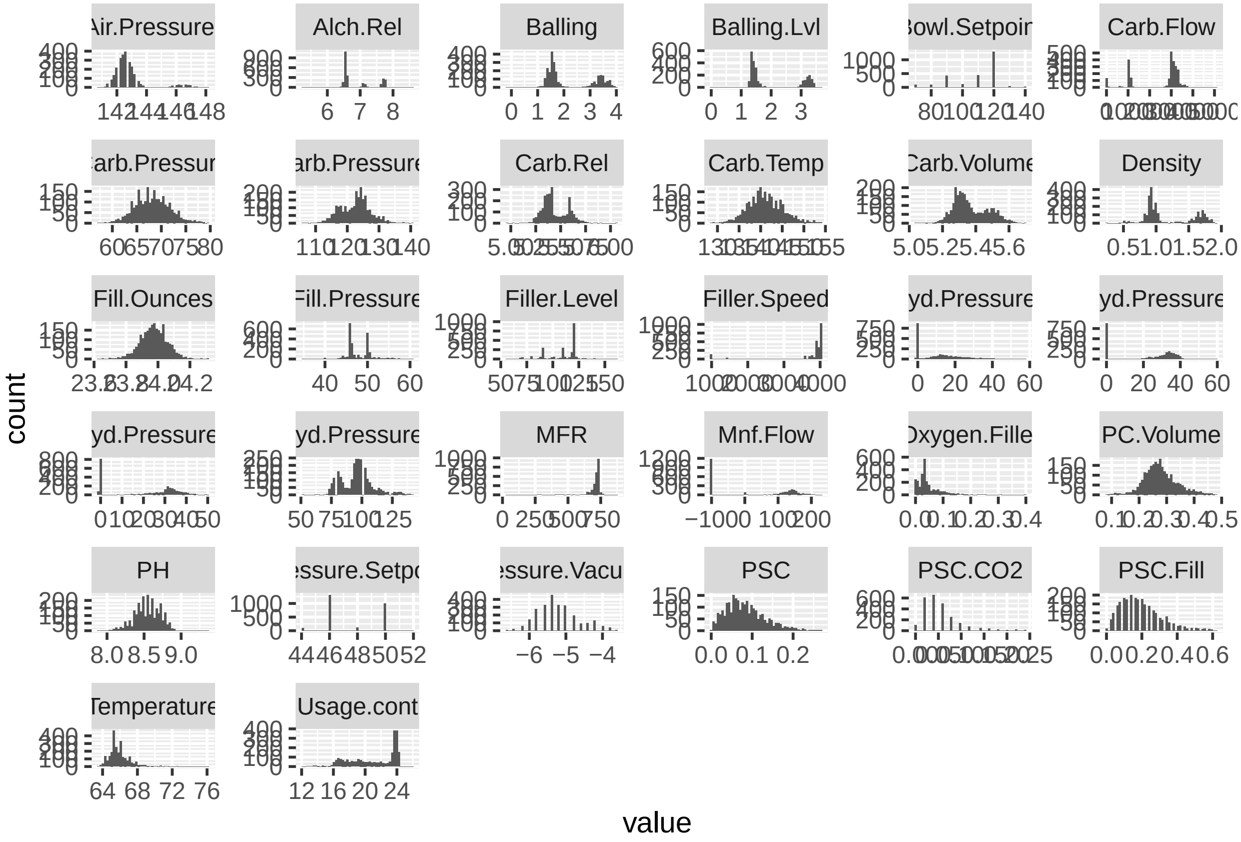
Assess PH Distributions

```
training %>%
  ggplot() +
  aes(x = PH) +
  geom_histogram(bins= 50) +
  facet_wrap(~ Brand.Code, scales = "free")
```

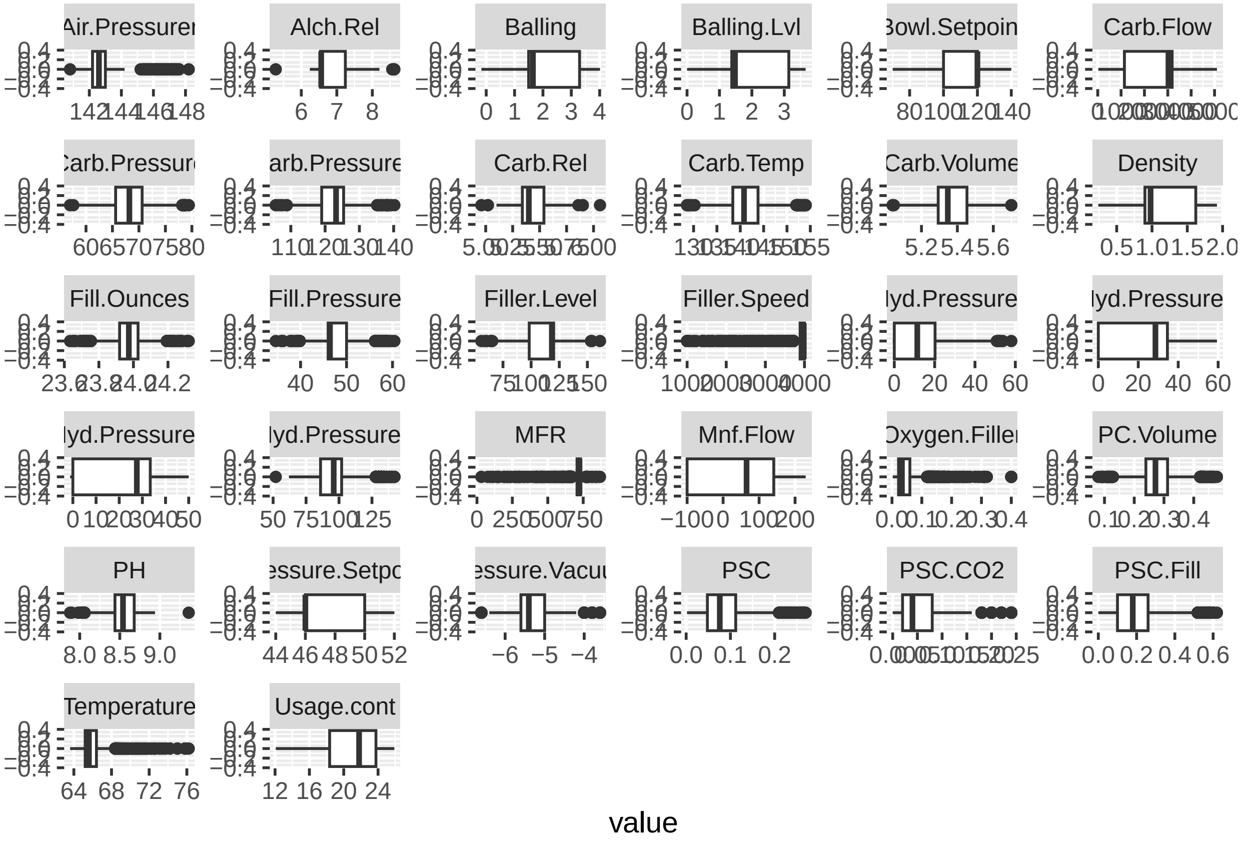


Assess Predictor Distributions, Skewness, and Relationships

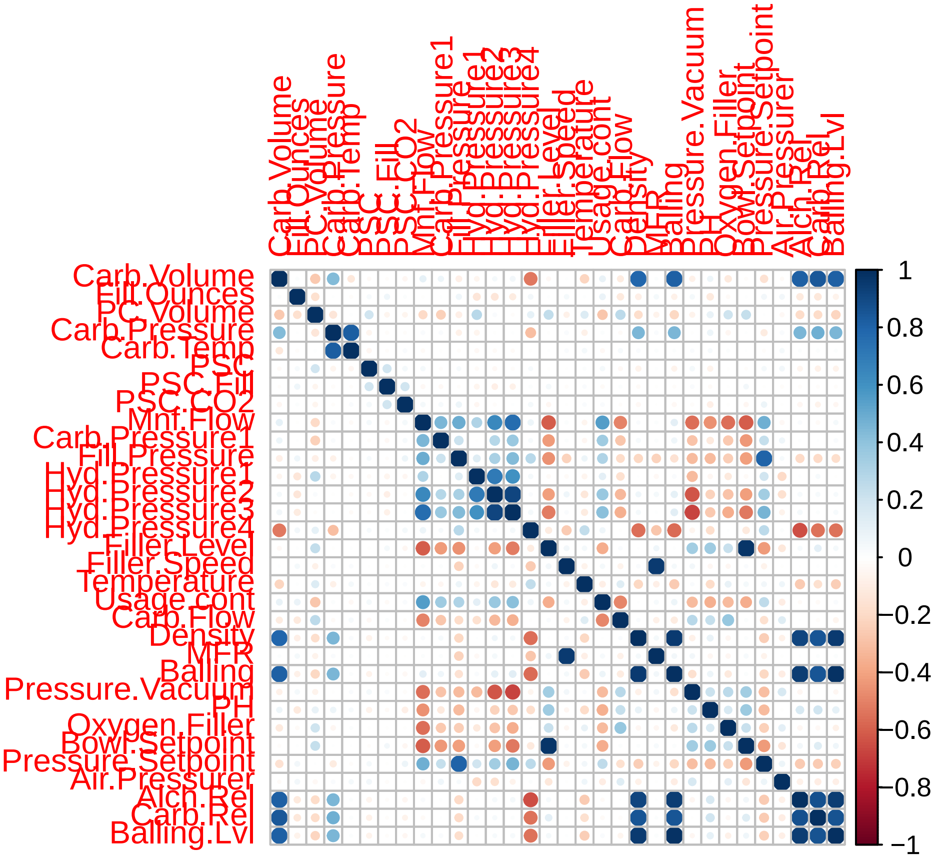
```
training %>%
  select(where(is.numeric))%>%
  gather() %>%
  filter(!is.na(value)) %>%
  ggplot(aes(value)) +
  geom_histogram(bins = 50) +
  facet_wrap(~ key, scales = "free")
```



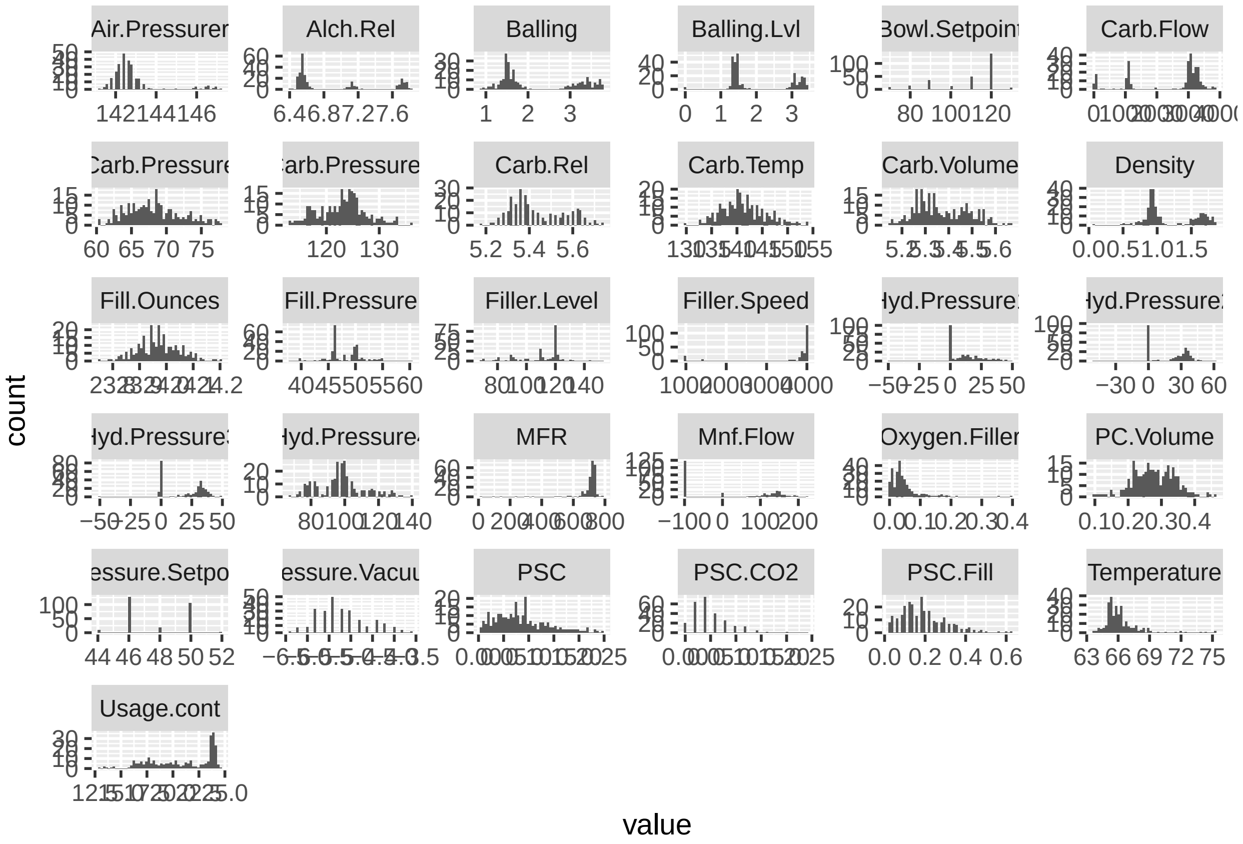
```
training %>%
  select(where(is.numeric))%>%
  gather() %>%
  filter(!is.na(value)) %>%
  ggplot(aes(value)) +
  geom_boxplot() +
  facet_wrap(~key, scales = "free")
```



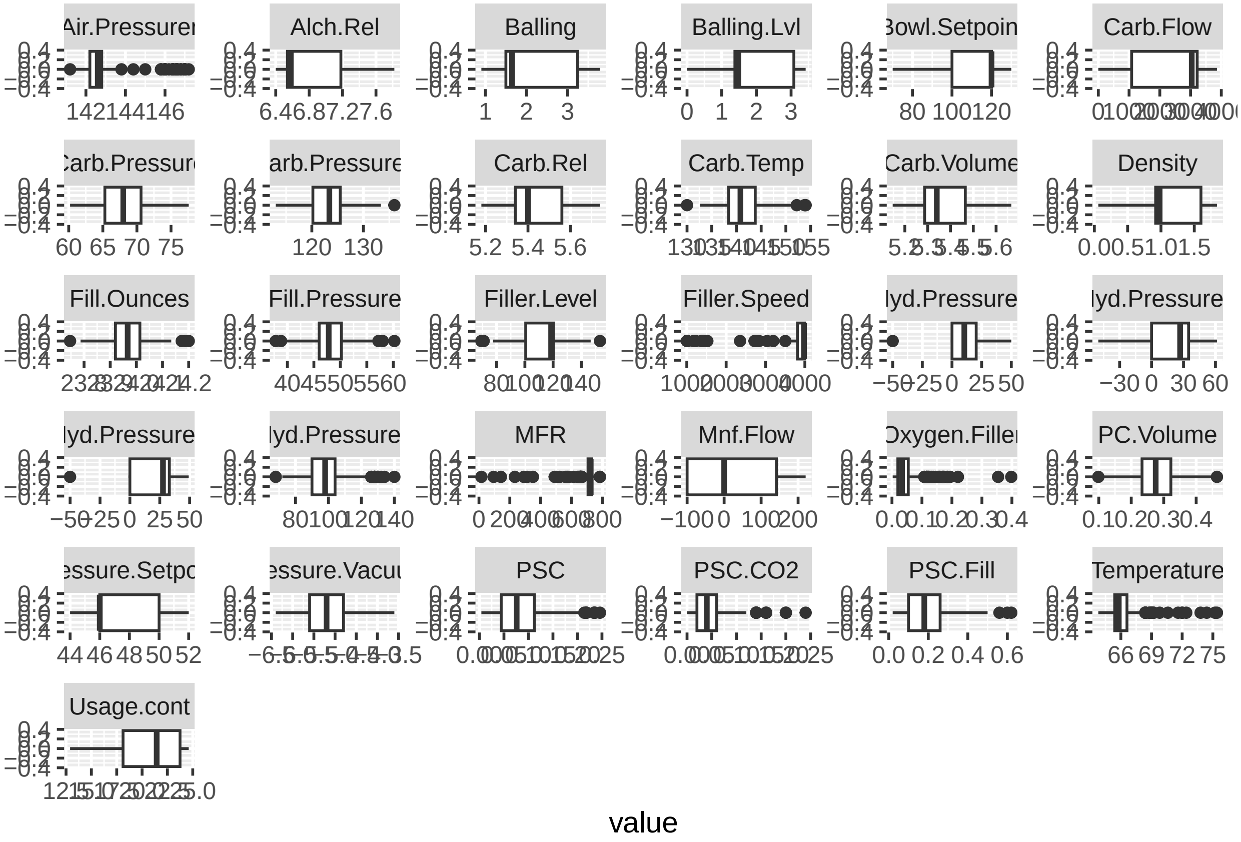
```r
training_cor <- cor(training %>%
                      select(where(is.numeric)),
                    use = "complete.obs")
corrplot(training_cor)
```



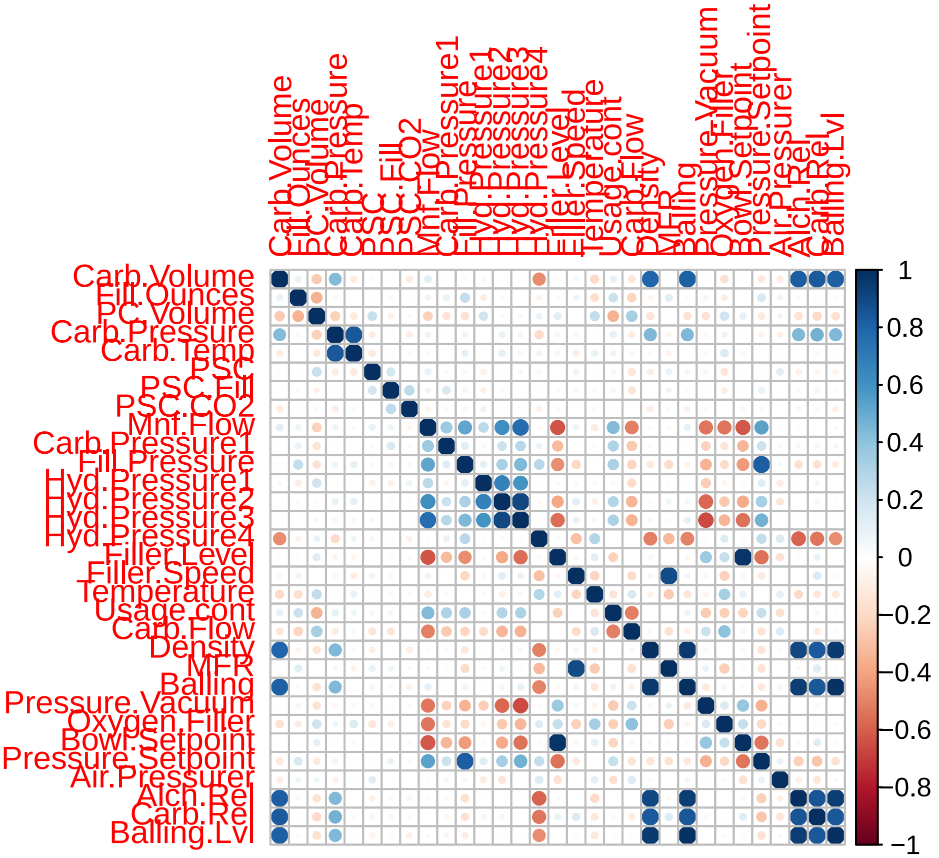
```
testing %>%
  select(where(is.numeric)) %>%
  gather() %>%
  filter(!is.na(value)) %>%
  ggplot(aes(value)) +
  geom_histogram(bins = 50) +
  facet_wrap(~ key, scales = "free")
```



```r
testing %>%
  select(where(is.numeric)) %>%
  gather() %>%
  filter(!is.na(value)) %>%
  ggplot(aes(value)) +
  geom_boxplot() +
  facet_wrap(~key, scales = "free")
```



```r
testing_cor <- cor(testing %>%
                      select(where(is.numeric)),
                    use = "complete.obs")
corrplot(testing_cor)
```



Data Preparation

Address Missing Data

```
miss_data <- data.frame(
  Feature = names(training),
  count_missing = colSums(is.na(training)),
  percent_miss = colMeans(is.na(training)) * 100
)
print(miss_data)
```

```
##                             Feature count_missing percent_miss
## Brand.Code               Brand.Code             0   0.00000000
## Carb.Volume             Carb.Volume            10   0.38895371
## Fill.Ounces             Fill.Ounces            38   1.47802412
## PC.Volume                 PC.Volume            39   1.51691949
## Carb.Pressure         Carb.Pressure            27   1.05017503
## Carb.Temp                 Carb.Temp            26   1.01127966
## PSC                             PSC            33   1.28354726
## PSC.Fill                   PSC.Fill            23   0.89459354
## PSC.CO2                     PSC.CO2            39   1.51691949
## Mnf.Flow                   Mnf.Flow             2   0.07779074
## Carb.Pressure1       Carb.Pressure1            32   1.24465189
## Fill.Pressure         Fill.Pressure            22   0.85569817
## Hyd.Pressure1         Hyd.Pressure1            11   0.42784909
## Hyd.Pressure2         Hyd.Pressure2            15   0.58343057
## Hyd.Pressure3         Hyd.Pressure3            15   0.58343057
## Hyd.Pressure4         Hyd.Pressure4            30   1.16686114
## Filler.Level           Filler.Level            20   0.77790743
## Filler.Speed           Filler.Speed            57   2.21703617
## Temperature             Temperature            14   0.54453520
## Usage.cont               Usage.cont             5   0.19447686
## Carb.Flow                 Carb.Flow             2   0.07779074
## Density                     Density             1   0.03889537
## MFR                             MFR           212   8.24581875
## Balling                     Balling             1   0.03889537
## Pressure.Vacuum     Pressure.Vacuum             0   0.00000000
## PH                               PH             4   0.15558149
## Oxygen.Filler         Oxygen.Filler            12   0.46674446
## Bowl.Setpoint         Bowl.Setpoint             2   0.07779074
## Pressure.Setpoint Pressure.Setpoint            12   0.46674446
## Air.Pressurer         Air.Pressurer             0   0.00000000
## Alch.Rel                   Alch.Rel             9   0.35005834
## Carb.Rel                   Carb.Rel            10   0.38895371
## Balling.Lvl             Balling.Lvl             1   0.03889537
```

```r
miss_scale <- 0.3
training <- training[, colMeans(is.na(training)) <= miss_scale]

training$PH_mean <- training$PH
training$PH_mean[is.na(training$PH_mean)] <- mean(training$PH, na.rm = TRUE)
training$PH_median <- training$PH
training$PH_median[is.na(training$PH_median)] <- median(training$PH, na.rm = TRUE)

scaled_data <- scale(training[sapply(training, is.numeric)])
scaled_data <- as.data.frame(scaled_data)

colnames(scaled_data) <- colnames(training)[sapply(training, is.numeric)]

if (sum(complete.cases(scaled_data)) > 5) {
  num_data <- knnImputation(scaled_data, k = 5)
} else {
  stop("impution not be completed, not enough cases")
}

num_data <- as.data.frame(num_data)
X <- num_data[ , !names(num_data) %in% c("PH")]
y <- num_data$PH

set.seed(123)

trainIndex <- createDataPartition(y, p = 0.8, list = FALSE)
X_train <- X[trainIndex, ]
X_test <- X[-trainIndex, ]
y_train <- y[trainIndex]
y_test <- y[-trainIndex]
```

Address Degenerate Variables

Assessing Models

Decision Tree Model

```r
dt_model <- train(X_train, y_train, method = "rpart")
dt_pred <- predict(dt_model, X_test)
```

Linear Regression Model

```r
linear_model <- train(X_train, y_train, method = "lm")
linear_pred <- predict(linear_model, X_test)
```

Neural Network Model

```r
nn_model <- train(X_train, y_train, method = "nnet", linout = TRUE, trace = FALSE, maxit = 500)
nn_pred <- predict(nn_model, X_test)
```

Random Forest Model

```r
rf_model <- train(X_train, y_train, method = "rf", ntree = 100)
rf_pred <- predict(rf_model, X_test)
```

Support Vector Machine (SVM) Model

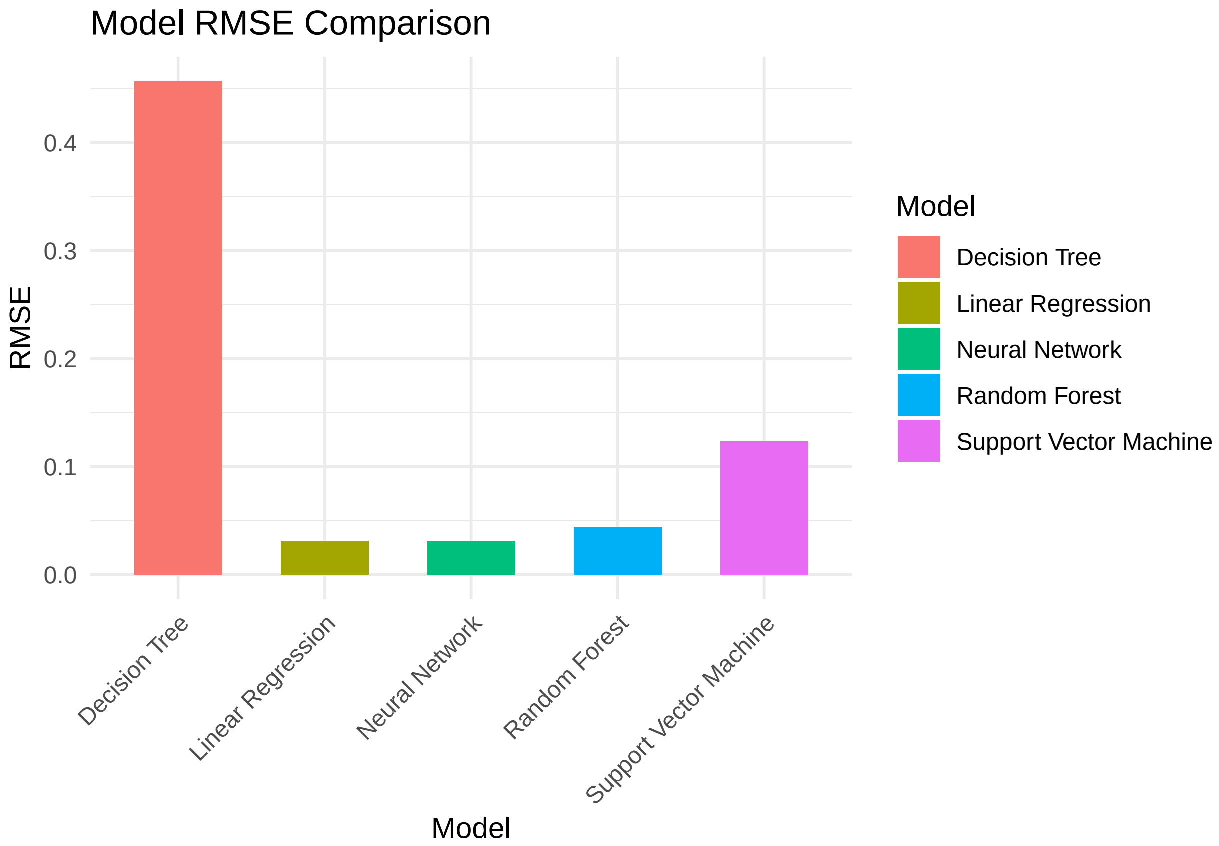
```r
svm_model <- train(X_train, y_train, method = "svmRadial")
svm_pred <- predict(svm_model, X_test)
```

Model Performance Evaluation and Visualization

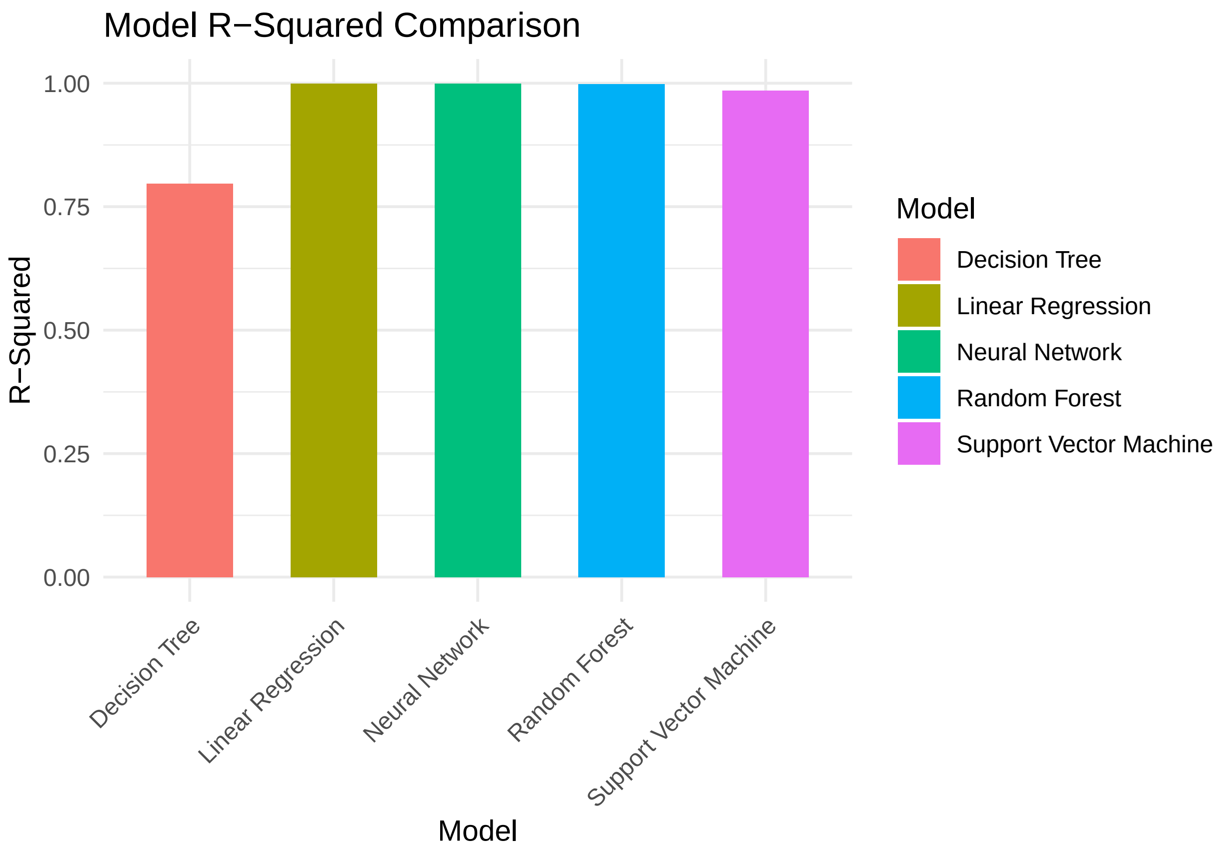
```r
model_results <- data.frame(
  Model = c("Linear Regression", "Decision Tree", "Random Forest", "Support Vector Machine", "Neural Ne
  RMSE = c(postResample(linear_pred, y_test)[1],
           postResample(dt_pred, y_test)[1],
           postResample(rf_pred, y_test)[1],
           postResample(svm_pred, y_test)[1],
           postResample(nn_pred, y_test)[1]),
  Rsquared = c(postResample(linear_pred, y_test)[2],
               postResample(dt_pred, y_test)[2],
               postResample(rf_pred, y_test)[2],
               postResample(svm_pred, y_test)[2],
                postResample(nn_pred, y_test)[2])
)
print(model_results)
```

```
##                    Model       RMSE  Rsquared
## 1      Linear Regression 0.03105924 0.9990577
## 2          Decision Tree 0.45647364 0.7962813
## 3          Random Forest 0.04431565 0.9981571
## 4 Support Vector Machine 0.12374884 0.9855253
## 5         Neural Network 0.03126260 0.9990451
```

```
plot1 <- ggplot(model_results, aes(x=Model, y=RMSE, fill=Model)) +
  geom_bar(stat="identity", width=0.6) +
  labs(title="Model RMSE Comparison", y="RMSE", x="Model") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
plot2 <- ggplot(model_results, aes(x=Model, y=Rsquared, fill=Model)) +
  geom_bar(stat="identity", width=0.6) +
  labs(title="Model R-Squared Comparison", y="R-Squared", x="Model") +
  theme_minimal() +
  theme(axis.text.x = element_text(angle = 45, hjust = 1))
print(plot1)
```



```
print(plot2)
```


Model R−Squared Comparison
R−Squared
1.00
0.75
0.50
0.25
0.00
Decision Tree
Linear Regression
Neural Network
Random Forest
Support Vector Machine
Model
Model
Decision Tree
Linear Regression
Neural Network
Random Forest
Support Vector Machine