An Analysis of Market Factors and Investor Sentiment as Determinants of the CBOE SKEW Index

Adhiraj Chhoda

2025-05-30

Contents

```r
library(readr)
library(readxl)
library(dplyr)
library(zoo)
library(lubridate)
library(janitor)
library(purrr)
library(corrplot)
library(stargazer)
library(car)
library(lmtest)
library(sandwich)
library(ggplot2)
library(moments)

study_start_date <- as.Date("2011-02-23")
study_end_date <- as.Date("2019-10-04")

sentiment_raw_raw <- read_excel("data/sentiment.xls", col_names = FALSE, .name_repair = "minimal")
sentiment_header_rows <- sentiment_raw_raw[2:4, ]
sentiment_header_transposed <- t(sentiment_header_rows)
sentiment_header_df <- as.data.frame(sentiment_header_transposed, stringsAsFactors = FALSE)
sentiment_header_combined <- apply(sentiment_header_df, 1, function(row_vals) {
  paste(na.omit(as.character(row_vals)), collapse = " ")
})
sentiment_colnames <- janitor::make_clean_names(sentiment_header_combined)
sentiment_raw_data <- read_excel("data/sentiment.xls", skip = 4, col_names = FALSE)
colnames(sentiment_raw_data) <- sentiment_colnames

skew_raw_data <- read_csv("data/SKEW_History.csv")
vix_raw_data <- read_csv("data/VIX_History.csv")
returns_spx_levels_raw <- read_csv("data/returns.csv")

skew_index_clean <- skew_raw_data %>%
  rename_with(~"Date_chr", any_of(c("DATE", "Date"))) %>%
  mutate(Date = mdy(Date_chr)) %>%
  select(Date, SKEW) %>%
```

```
  filter(!is.na(Date) & !is.na(SKEW) & SKEW != ".") %>%
  mutate(SKEW = as.numeric(SKEW)) %>%
  filter(!is.na(SKEW)) %>%
  arrange(Date) %>%
  distinct(Date, .keep_all = TRUE)

vix_clean <- vix_raw_data %>%
  rename_with(~"Date_chr", any_of(c("DATE", "Date"))) %>%
  rename_with(~"VIX_close", any_of(c("CLOSE", "Close", "VIXCLS"))) %>%
  mutate(Date = mdy(Date_chr)) %>%
  select(Date, VIX = VIX_close) %>%
  filter(!is.na(Date) & !is.na(VIX) & VIX != ".") %>%
  mutate(VIX = as.numeric(VIX)) %>%
  filter(!is.na(VIX)) %>%
  arrange(Date) %>%
  distinct(Date, .keep_all = TRUE)

spx_levels_clean <- returns_spx_levels_raw %>%
  rename_with(~"Date_chr", any_of(c("DATE", "Date"))) %>%
  rename_with(~"Close_SPX", any_of(c("CLOSE", "Close", "Adj Close"))) %>%
  mutate(Date = mdy(Date_chr)) %>%
  select(Date, Close_SPX) %>%
  filter(!is.na(Date) & !is.na(Close_SPX)) %>%
  mutate(Close_SPX = as.numeric(Close_SPX)) %>%
  arrange(Date) %>%
  distinct(Date, .keep_all = TRUE)

sentiment_clean <- sentiment_raw_data %>%
  mutate(
    reported_date_as_numeric = suppressWarnings(as.numeric(as.character(reported_date))),
    Date_temp = case_when(
      !is.na(reported_date_as_numeric) & reported_date_as_numeric > 0 & reported_date_as_numeric < 600
      TRUE ~ suppressWarnings(ymd(reported_date))
    )
  ) %>%
  filter(!is.na(bullish) & !is.na(Date_temp)) %>%
  select(Date = Date_temp, Bullish_val = bullish) %>%
  mutate(Bullish_val = as.numeric(as.character(Bullish_val))) %>%
  filter(!is.na(Bullish_val)) %>%
  arrange(Date) %>%
  distinct(Date, .keep_all = TRUE)

indexpc_archive_raw <- read_csv("data/pc/indexpcarchive.csv", skip = 2,
                                col_types = cols(DATE = col_character(), .default = col_double()),
                                show_col_types = FALSE)
indexpc_recent_raw <- read_csv("data/pc/indexpc.csv", skip = 2,
                               col_types = cols(DATE = col_character(), .default = col_double()),
                               show_col_types = FALSE)

colnames(indexpc_archive_raw) <- c("Trade_date_chr", "Call_Volume", "Put_Volume", "Total_Volume", "PC_
colnames(indexpc_recent_raw) <- c("Trade_date_chr", "Call_Volume", "Put_Volume", "Total_Volume", "PC_I

indexpc_archive_clean <- indexpc_archive_raw %>%
```

```
  mutate(Date = mdy(Trade_date_chr)) %>%
  select(Date, PC_Ratio)

indexpc_recent_clean <- indexpc_recent_raw %>%
  mutate(Date = mdy(Trade_date_chr)) %>%
  select(Date, PC_Ratio)

combined_pc_clean <- bind_rows(indexpc_archive_clean, indexpc_recent_clean) %>%
  filter(!is.na(Date) & !is.na(PC_Ratio)) %>%
  arrange(Date) %>%
  distinct(Date, .keep_all = TRUE)

spx_metrics <- spx_levels_clean %>%
  arrange(Date) %>%
  filter(Close_SPX > 0) %>%
  mutate(
    log_return = c(NA, diff(log(Close_SPX))),
    RealizedVol = rollapplyr(log_return, width = 21, FUN = sd, fill = NA, align = "right", na.rm = TRU
    MarketReturn = rollapplyr(log_return, width = 21, FUN = sum, fill = NA, align = "right", na.rm = '
  ) %>%
  select(Date, RealizedVol, MarketReturn)

skew_daily_filtered <- skew_index_clean %>%
  filter(Date >= study_start_date & Date <= study_end_date)

vix_daily_filtered <- vix_clean %>%
  filter(Date >= study_start_date & Date <= study_end_date)

spx_metrics_filtered <- spx_metrics %>%
  filter(Date >= study_start_date & Date <= study_end_date)

pc_daily_filtered <- combined_pc_clean %>%
  filter(Date >= study_start_date & Date <= study_end_date)

actual_trading_days_df <- spx_metrics_filtered %>%
  select(Date) %>%
  distinct(Date) %>%
  arrange(Date)

sentiment_daily_processed <- actual_trading_days_df %>%
  left_join(sentiment_clean, by = "Date") %>%
  arrange(Date) %>%
  mutate(Sentiment_ffill = na.locf(Bullish_val, na.rm = FALSE, fromLast = FALSE)) %>%
  mutate(Sentiment = na.locf(Sentiment_ffill, na.rm = FALSE, fromLast = TRUE)) %>%
  select(Date, Sentiment) %>%
  filter(!is.na(Sentiment))

data_list <- list(skew_daily_filtered,
                  vix_daily_filtered,
                  spx_metrics_filtered,
                  pc_daily_filtered,
                  sentiment_daily_processed)
```

```r
master_df <- data_list %>%
  purrr::reduce(~ left_join(.x, .y, by = "Date")) %>%
  arrange(Date) %>%
  filter(Date >= study_start_date & Date <= study_end_date) %>%
  na.omit()

final_model_data <- master_df %>%
  mutate(
    VIX_sq = VIX * VIX,
    VIX_Sentiment_Interaction = VIX * Sentiment
  ) %>%
  arrange(Date) %>%
  mutate(
    VIX_lag1 = lag(VIX, 1),
    VIX_sq_lag1 = lag(VIX_sq, 1),
    RealizedVol_lag1 = lag(RealizedVol, 1),
    MarketReturn_lag1 = lag(MarketReturn, 1),
    Sentiment_lag1 = lag(Sentiment, 1),
    PC_Ratio_lag1 = lag(PC_Ratio, 1),
    VIX_Sentiment_Interaction_lag1 = lag(VIX_Sentiment_Interaction, 1)
  ) %>%
  mutate(
    VIX_lag1_centered = VIX_lag1 - mean(VIX_lag1, na.rm = TRUE),
    Sentiment_lag1_centered = Sentiment_lag1 - mean(Sentiment_lag1, na.rm = TRUE),
    VIX_Sent_Interact_centered_lag1 = VIX_lag1_centered * Sentiment_lag1_centered
  ) %>%
  select(
    Date,
    SKEW,
    VIX_lag1,
    VIX_sq_lag1,
    RealizedVol_lag1,
    MarketReturn_lag1,
    Sentiment_lag1,
    PC_Ratio_lag1,
    VIX_Sentiment_Interaction_lag1,
    VIX_Sent_Interact_centered_lag1
  ) %>%
  na.omit()
```

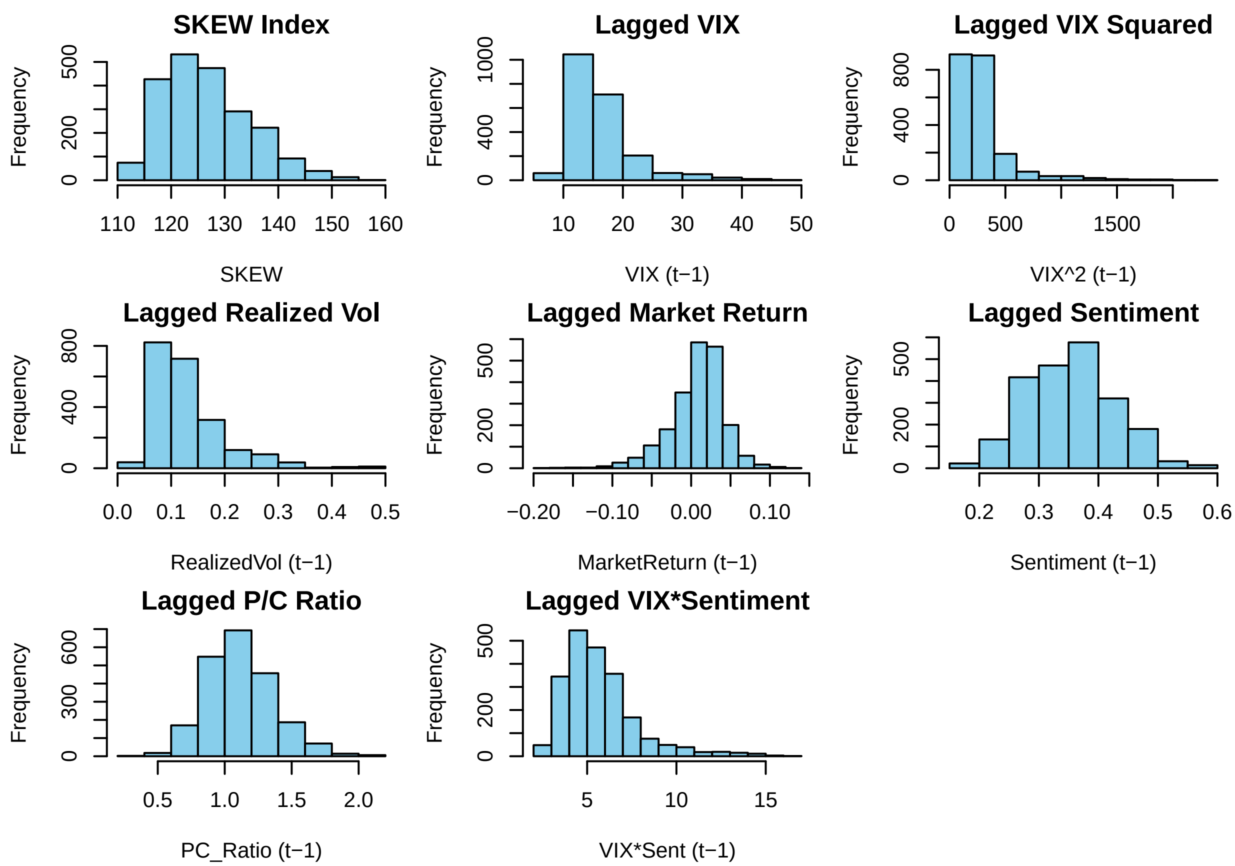
```r
summary(final_model_data[, -1])
```

```
##       SKEW          VIX_lag1      VIX_sq_lag1      RealizedVol_lag1
##  Min.   :111.3   Min.   : 9.14   Min.   :  83.54   Min.   :0.03469
##  1st Qu.:120.5   1st Qu.:12.86   1st Qu.: 165.38   1st Qu.:0.08420
##  Median :125.4   Median :14.88   Median : 221.41   Median :0.11072
##  Mean   :126.7   Mean   :16.29   Mean   : 294.38   Mean   :0.13011
##  3rd Qu.:131.7   3rd Qu.:18.11   3rd Qu.: 327.97   3rd Qu.:0.15390
##  Max.   :159.0   Max.   :48.00   Max.   :2304.00   Max.   :0.49411
##  MarketReturn_lag1   Sentiment_lag1   PC_Ratio_lag1
##  Min.   :-0.182655   Min.   :0.1775   Min.   :0.350
##  1st Qu.:-0.008998   1st Qu.:0.2975   1st Qu.:0.950
##  Median : 0.012476   Median :0.3534   Median :1.100
```

```
##  Mean   : 0.007877   Mean   :0.3531   Mean   :1.122
##  3rd Qu.: 0.029850   3rd Qu.:0.4014   3rd Qu.:1.280
##  Max.   : 0.125231   Max.   :0.5975   Max.   :2.200
##  VIX_Sentiment_Interaction_lag1 VIX_Sent_Interact_centered_lag1
##  Min.   : 2.329                 Min.   :-2.583650
##  1st Qu.: 4.267                 1st Qu.:-0.153735
##  Median : 5.287                 Median :-0.005704
##  Mean   : 5.699                 Mean   :-0.051596
##  3rd Qu.: 6.549                 3rd Qu.: 0.102383
##  Max.   :16.709                 Max.   : 1.990596
```

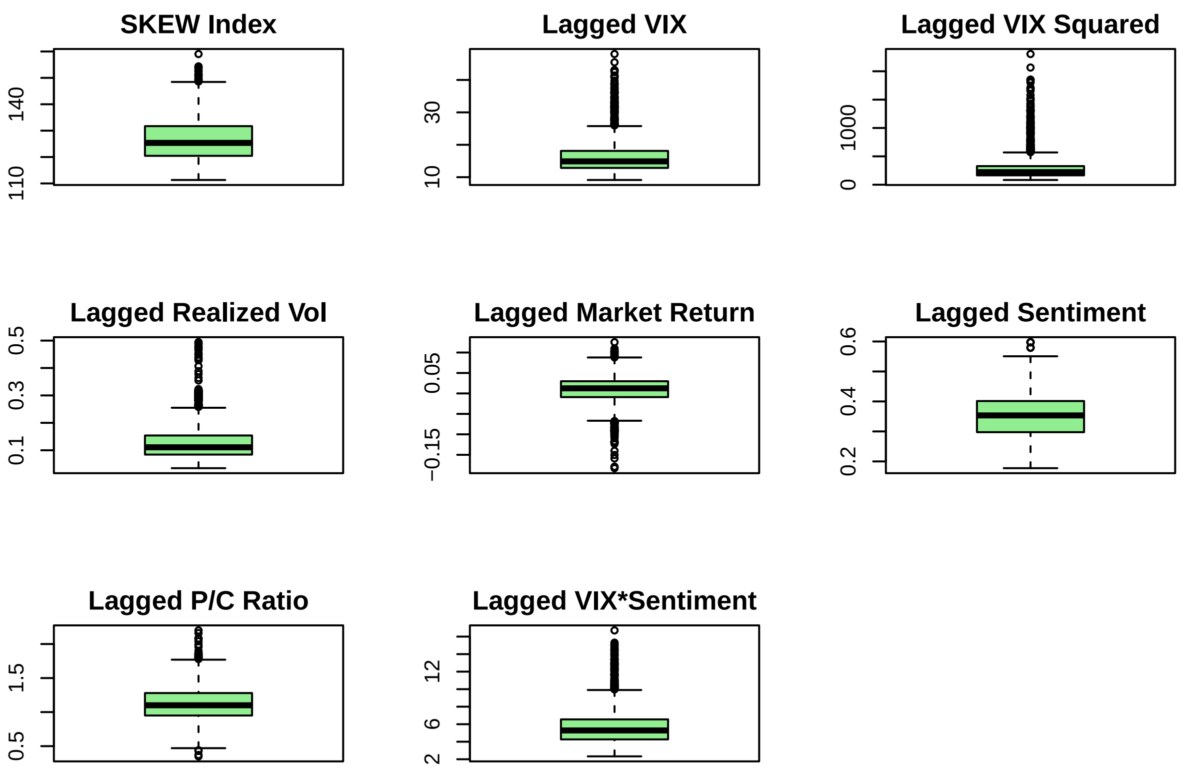
```r
#histograms

par(mfrow=c(3,3), mar=c(4,4,2,1))
hist(final_model_data$SKEW, main="SKEW Index", xlab="SKEW", col="skyblue")
hist(final_model_data$VIX_lag1, main="Lagged VIX", xlab="VIX (t-1)", col="skyblue")
hist(final_model_data$VIX_sq_lag1, main="Lagged VIX Squared", xlab="VIX^2 (t-1)", col="skyblue")
hist(final_model_data$RealizedVol_lag1, main="Lagged Realized Vol", xlab="RealizedVol (t-1)", col="sky
hist(final_model_data$MarketReturn_lag1, main="Lagged Market Return", xlab="MarketReturn (t-1)", col=
hist(final_model_data$Sentiment_lag1, main="Lagged Sentiment", xlab="Sentiment (t-1)", col="skyblue")
hist(final_model_data$PC_Ratio_lag1, main="Lagged P/C Ratio", xlab="PC_Ratio (t-1)", col="skyblue")
hist(final_model_data$VIX_Sentiment_Interaction_lag1, main="Lagged VIX*Sentiment", xlab="VIX*Sent (t-
par(mfrow=c(1,1))
```



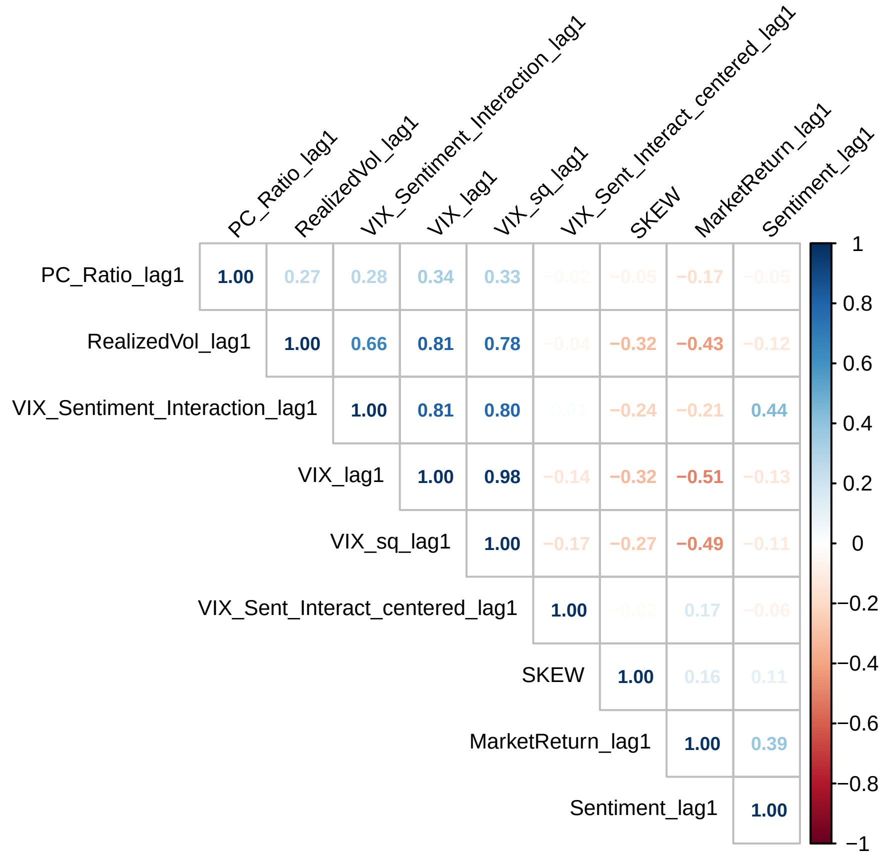
```r
#boxplots

par(mfrow=c(3,3), mar=c(4,4,2,1))
boxplot(final_model_data$SKEW, main="SKEW Index", col="lightgreen")
boxplot(final_model_data$VIX_lag1, main="Lagged VIX", col="lightgreen")
boxplot(final_model_data$VIX_sq_lag1, main="Lagged VIX Squared", col="lightgreen")
boxplot(final_model_data$RealizedVol_lag1, main="Lagged Realized Vol", col="lightgreen")
boxplot(final_model_data$MarketReturn_lag1, main="Lagged Market Return", col="lightgreen")
boxplot(final_model_data$Sentiment_lag1, main="Lagged Sentiment", col="lightgreen")
boxplot(final_model_data$PC_Ratio_lag1, main="Lagged P/C Ratio", col="lightgreen")
boxplot(final_model_data$VIX_Sentiment_Interaction_lag1, main="Lagged VIX*Sentiment", col="lightgreen
par(mfrow=c(1,1))
```




```r
#correlation plot

numeric_cols_for_corr <- final_model_data %>% select(-Date)
cor_matrix_full <- cor(numeric_cols_for_corr, use = "complete.obs")
corrplot(cor_matrix_full, method = "number", type = "upper", order = "hclust",
         tl.col = "black", tl.srt = 45, tl.cex = 0.7, number.cex = 0.6,
         addCoef.col = "black",
         cl.cex = 0.7)
title("Correlation Matrix of Model Variables", line = 3)
```




```
model1 <- lm(SKEW ~ VIX_lag1 + VIX_sq_lag1 + RealizedVol_lag1 +
                 MarketReturn_lag1 + Sentiment_lag1 + PC_Ratio_lag1 +
                 VIX_Sent_Interact_centered_lag1,
             data = final_model_data)
summary(model1)
```

Call: lm(formula = SKEW ~ VIX_lag1 + VIX_sq_lag1 + RealizedVol_lag1 + MarketReturn_lag1 + Sentiment_lag1 + PC_Ratio_lag1 + VIX_Sent_Interact_centered_lag1, data = final_model_data)

Residuals: Min 1Q Median 3Q Max -17.688 -5.342 -1.240 4.349 31.766

Coefficients: Estimate Std. Error t value Pr(>|t|)
(Intercept) 142.329818 1.750968 81.286 < 2e-16 ***VIX_lag1 -1.640704 0.148173 -11.073 < 2e-16*** VIX_sq_lag1 0.029782 0.003104 9.596 < 2e-16 ***RealizedVol_lag1 -19.189084 4.071824 -4.713 2.6e-06*** MarketReturn_lag1 -11.442986 5.844857 -1.958 0.05038 .
Sentiment_lag1 6.857745 2.392234 2.867 0.00419 ** PC_Ratio_lag1 2.183907 0.674255 3.239 0.00122 **
VIX_Sent_Interact_centered_lag1 -0.125238 0.457637 -0.274 0.78437
— Signif. codes: 0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

Residual standard error: 7.43 on 2157 degrees of freedom Multiple R-squared: 0.1603, Adjusted R-squared: 0.1575 F-statistic: 58.81 on 7 and 2157 DF, p-value: < 2.2e-16

```
my_covariate_labels_m1_centered <- c(
  "VIX (t-1)",
  "VIX$^{2}$ (t-1)",
  "Realized Vol (t-1, 21d)",
  "Market Return (t-1, 21d)",
```

```r
  "AAII Sentiment (t-1, Bullish \\%)",
  "SPX Put-Call Ratio (t-1)",
  "$\\text{VIX}_c\\ (t\\!-\\!1) \\times \\text{Sent}_c\\ (t\\!-\\!1)$"
)


stargazer(model1,
          type = "latex",
          title = "Table 1: OLS Regression for SKEW Index (Centered Interaction)",
          align = TRUE,
          dep.var.labels = "CBOE SKEW Index (Daily)",
          covariate.labels = my_covariate_labels_m1_centered,
          ci = TRUE, ci.level = 0.95, single.row = FALSE,
          omit.stat = c("ser", "rsq", "f"),
          add.lines = list(
              c("Observations", formatC(nobs(model1), format="d", big.mark=",")),
              c("R-squared", format(round(summary(model1)$r.squared, 3), nsmall = 3)),
              c("Adj. R-squared", format(round(summary(model1)$adj.r.squared, 3), nsmall = 3)),
              c("F-statistic", paste0(format(round(summary(model1)$fstatistic[["value"]],2),nsmall=2)
                                      ifelse(pf(summary(model1)$fstatistic[["value"]],
                                                summary(model1)$fstatistic[["numdf"]],
                                                summary(model1)$fstatistic[["dendf"]],
                                                lower.tail=FALSE)<0.001,"^{***}",
                                             ifelse(pf(summary(model1)$fstatistic[["value"]],
                                                       summary(model1)$fstatistic[["numdf"]],
                                                       summary(model1)$fstatistic[["dendf"]],
                                                       lower.tail=FALSE)<0.01,"^{**}",
                                                    ifelse(pf(summary(model1)$fstatistic[["value"]],
                                                              summary(model1)$fstatistic[["numdf"]],
                                                              summary(model1)$fstatistic[["dendf"]],
                                                              lower.tail=FALSE)<0.05,"^{*}", ""))),
                                      " (df = ", summary(model1)$fstatistic[["numdf"]], ", ", summary(mo
            star.cutoffs = c(0.05, 0.01, 0.001),
            notes = c("$^{*}$p$<$.05; $^{**}$p$<$.01; $^{***}$p$<$.001. OLS Standard Errors."),
            notes.align = "l",
            notes.append = FALSE,
            notes.label = "",
            header = FALSE,
            float = FALSE,
            no.space = TRUE,
            font.size = "small",
            digits = 3
    )
```

	Dependent variable:
	CBOE SKEW Index (Daily)
VIX (t-1)	-1.641^{***}
	(-1.931,-1.350)
VIX^2 (t-1)	0.030^{***}
	(0.024,0.036)
Realized Vol (t-1, 21d)	-19.189^{***}
	(-27.170,-11.208)
Market Return (t-1, 21d)	-11.443
	(-22.899,0.013)
AAII Sentiment (t-1, Bullish %)	6.858^{**}
	(2.169,11.546)
SPX Put-Call Ratio (t-1)	2.184^{**}
	(0.862,3.505)
$\text{VIX}_c\ (t-1) \times \text{Sent}_c\ (t-1)$	-0.125
	(-1.022,0.772)
Constant	142.330^{***}
	(138.898,145.762)
Observations	$2,165$
R-squared	0.160
Adj. R-squared	0.158
F-statistic	$58.81^{***}(df = 7, 2157)$
Observations	2,165
Adjusted R^2	0.158

*p<.05; **p<.01; ***p<.001. OLS Standard Errors.

```r
par(mfrow=c(2,2))
plot(model1)
```

```r
par(mfrow=c(1,1))

crPlots(model1,
      terms = ~ .,
      layout = NULL,
      ask = FALSE,
      smooth = list(smoother = loessLine, col.lines = "blue"),
      col = "black",
      pch = 19,
      cex = 0.7,
      main="Component+Residual Plots (Model 1)")
```

```r
print("--- Normality of Residuals ---")
```

[1] "— Normality of Residuals —"

```r
shapiro_test_result <- shapiro.test(residuals(model1))
print(shapiro_test_result)
```

`Shapiro-Wilk normality test`

data: residuals(model1) W = 0.96534, p-value < 2.2e-16

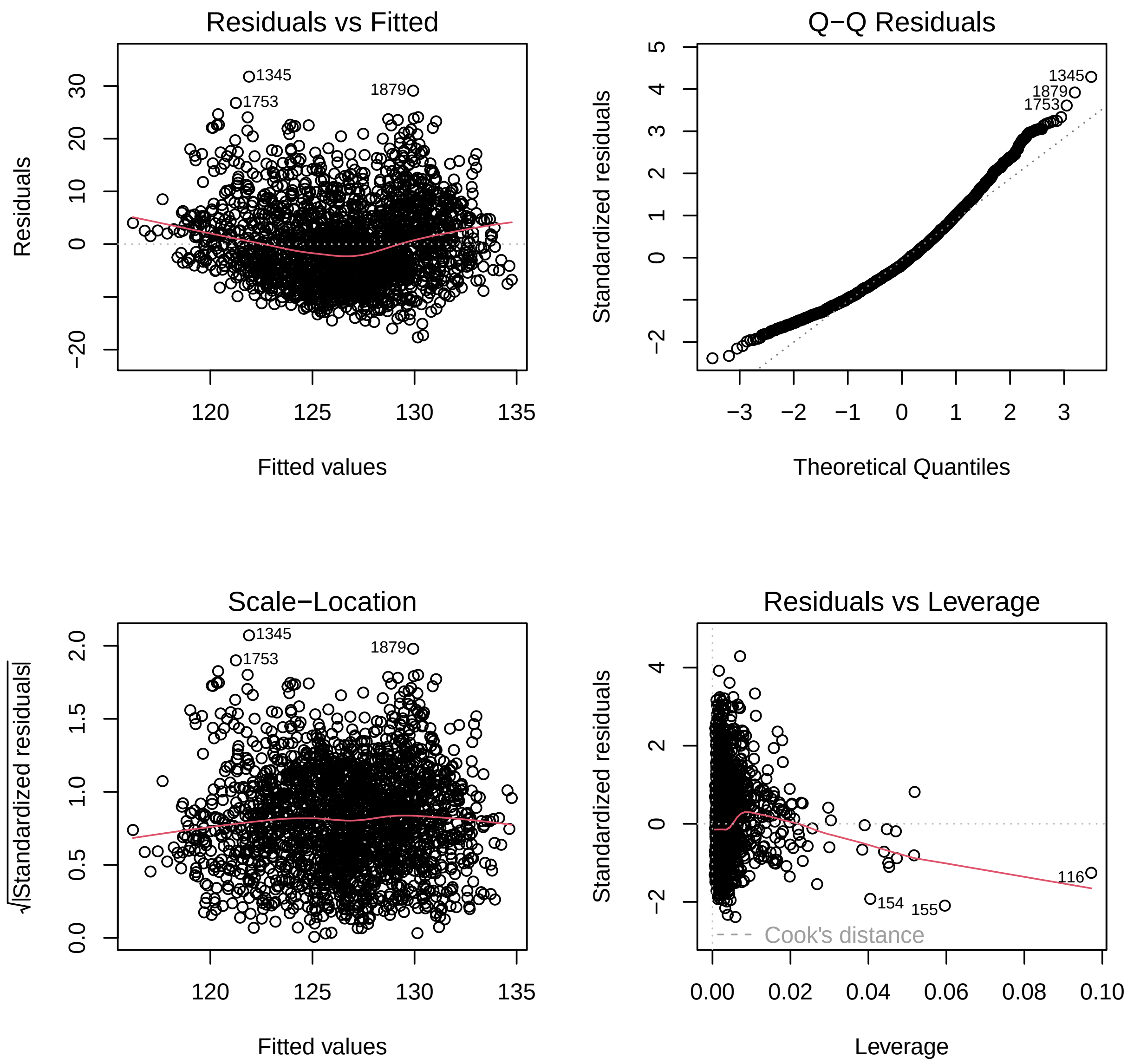


Figure 1: Regression Diagnostic Plots for OLS Model

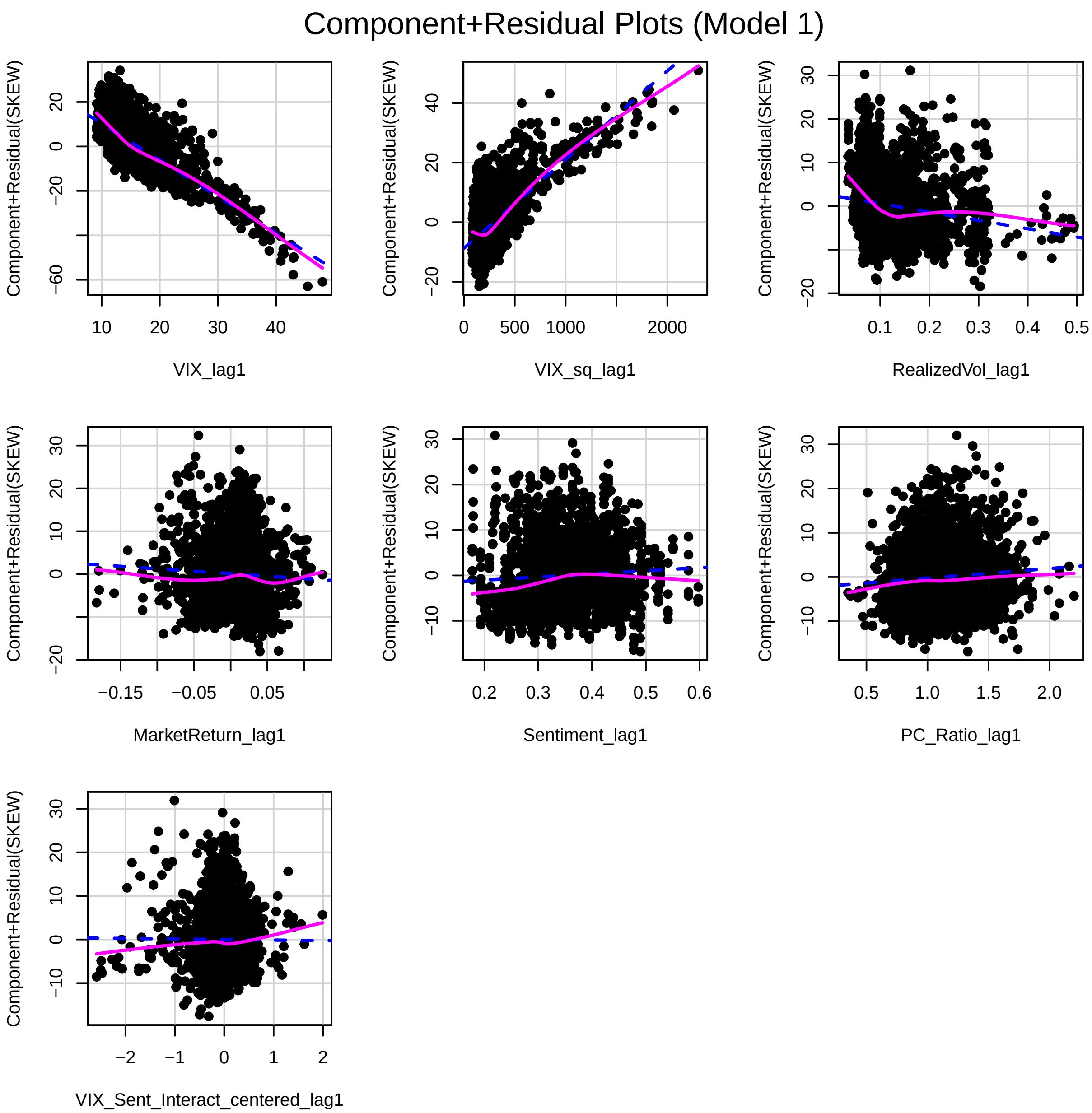


Figure 2: Regression Diagnostic Plots for OLS Model

```r
hist(residuals(model1), main="Histogram of Residuals", xlab="Residuals", breaks=50, col="lightblue")
```

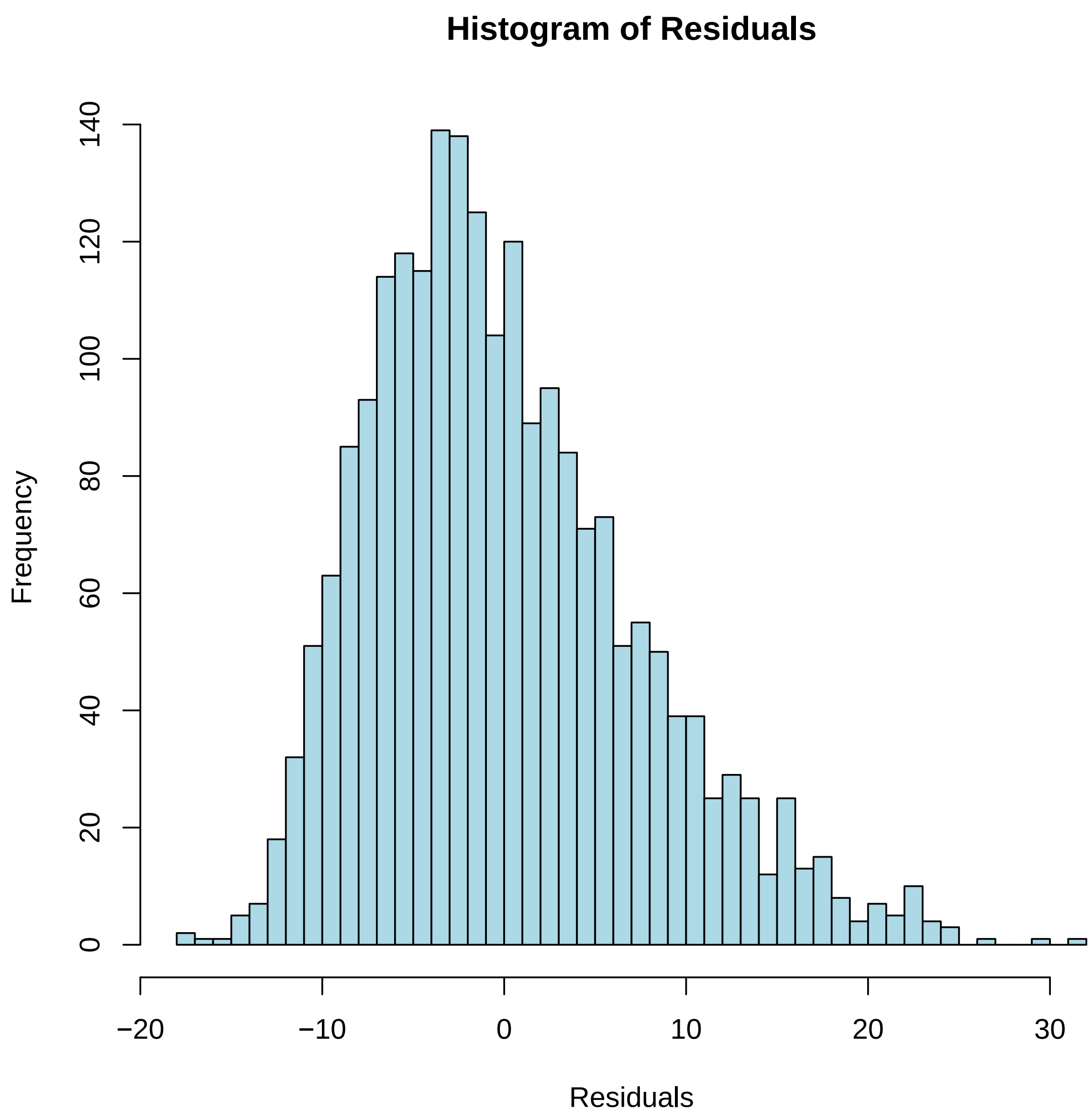


Figure 3: Regression Diagnostic Plots for OLS Model

```r
print(paste("Skewness of residuals:", skewness(residuals(model1))))
```

[1] "Skewness of residuals: 0.744755613212608"

```r
print(paste("Kurtosis of residuals (excess kurtosis):", kurtosis(residuals(model1)) - 3))
```

[1] "Kurtosis of residuals (excess kurtosis): 0.45261610468358"

```r
print("--- Homoscedasticity ---")
```

[1] "— Homoscedasticity —"

```r
bp_test_result <- bptest(model1, studentize = FALSE)
print(bp_test_result)
```

```
Breusch-Pagan test
```

data: model1 BP = 58.866, df = 7, p-value = 2.541e-10

```r
print("--- Autocorrelation of Residuals ---")
```

[1] "— Autocorrelation of Residuals —"

```r
dw_test_result <- dwtest(model1)
print(dw_test_result)
```

```
Durbin-Watson test
```

data: model1 DW = 0.27423, p-value < 2.2e-16 alternative hypothesis: true autocorrelation is greater than 0

```r
par(mfrow=c(1,2))
acf(residuals(model1), main="ACF of Residuals")
pacf(residuals(model1), main="PACF of Residuals")
```

```r
par(mfrow=c(1,1))
ljung_box_test_10 <- Box.test(residuals(model1), lag = 10, type = "Ljung-Box")
ljung_box_test_20 <- Box.test(residuals(model1), lag = 20, type = "Ljung-Box")
print("Ljung-Box test for autocorrelation (10 lags):")
```

[1] "Ljung-Box test for autocorrelation (10 lags):"

```r
print(ljung_box_test_10)
```

```
Box-Ljung test
```

data: residuals(model1) X-squared = 10035, df = 10, p-value < 2.2e-16

```r
print("Ljung-Box test for autocorrelation (20 lags):")
```

[1] "Ljung-Box test for autocorrelation (20 lags):"

```r
print(ljung_box_test_20)
```

```
Box-Ljung test
```

data: residuals(model1) X-squared = 16182, df = 20, p-value < 2.2e-16

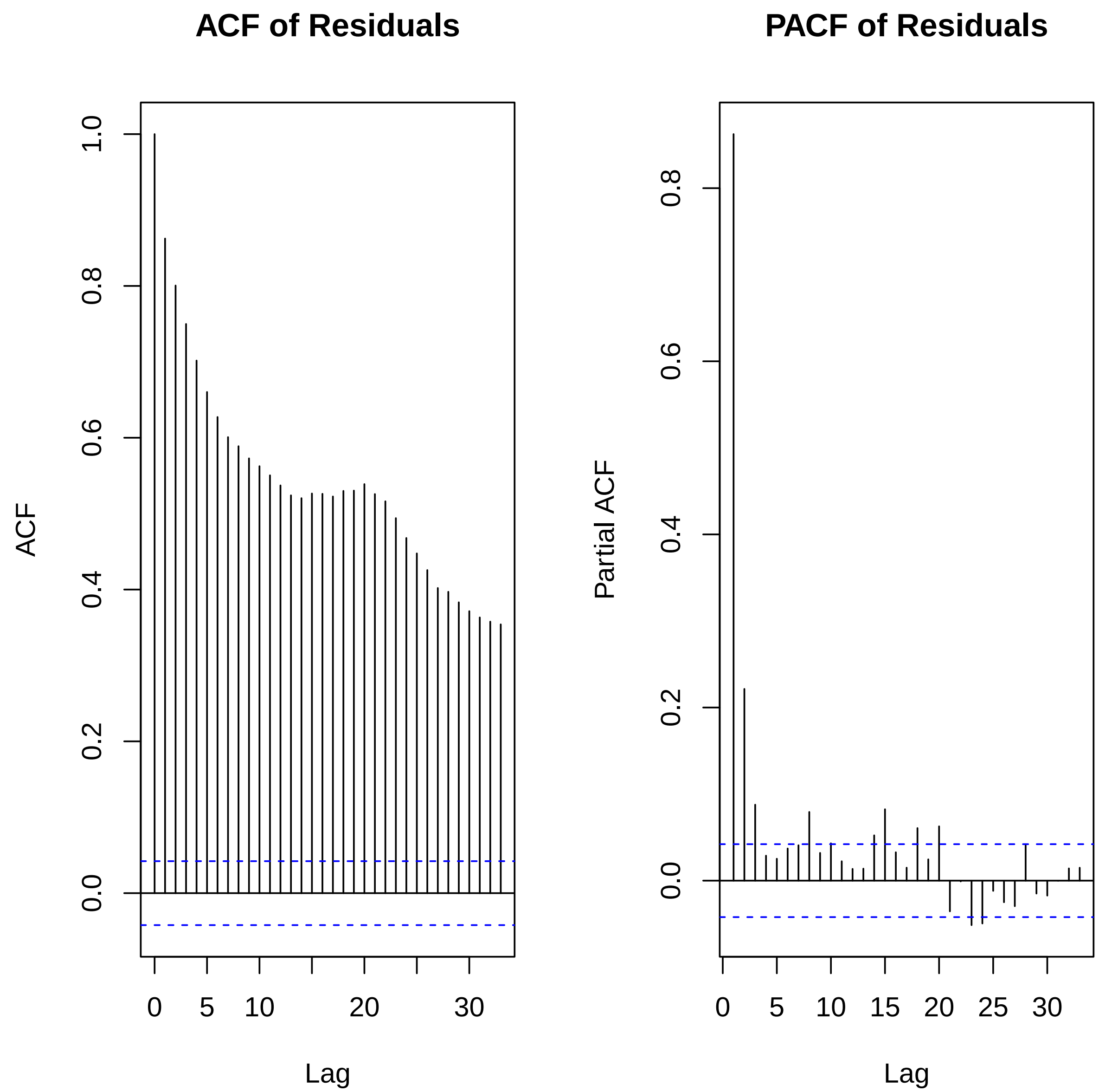


Figure 4: Regression Diagnostic Plots for OLS Model

```r
print("--- Multicollinearity ---")
```

[1] "— Multicollinearity —"

```r
vif_values <- vif(model1)
print(vif_values)
```

```
              VIX_lag1                VIX_sq_lag1
             25.064134                  21.865569
      RealizedVol_lag1           MarketReturn_lag1
              3.011822                   1.621272
        Sentiment_lag1               PC_Ratio_lag1
              1.209796                   1.134291
```

VIX_Sent_Interact_centered_lag1 1.110096

```r
print("--- Outliers and Influential Points ---")
```

[1] "— Outliers and Influential Points —"

```r
cooksd <- cooks.distance(model1)
plot(cooksd, pch="*", cex=1, main="Cook's Distance Plot", ylab="Cook's Distance")
abline(h = 4/nobs(model1), col="red", lty=2)
```

```r
influential_threshold_4_n <- 4/nobs(model1)
influential_points_4_n <- which(cooksd > influential_threshold_4_n)
print(paste("Number of points with Cook's D > 4/n:", length(influential_points_4_n)))
```

[1] "Number of points with Cook's D > 4/n: 94"

```r
if(length(influential_points_4_n) > 0 && length(influential_points_4_n) < 20) {
    print(paste("Indices of points with Cook's D > 4/n:", paste(head(influential_points_4_n, 10), col
}
influential_points_1 <- which(cooksd > 1)
print(paste("Number of points with Cook's D > 1:", length(influential_points_1)))
```

[1] "Number of points with Cook's D > 1: 0"

```r
if(length(influential_points_1) > 0) {
    print(paste("Indices of points with Cook's D > 1:", paste(influential_points_1, collapse=", ")))
}

residuals_df_m1 <- data.frame(Date = final_model_data$Date, Residuals = residuals(model1))
p_res_time_m1 <- ggplot(residuals_df_m1, aes(x = Date, y = Residuals)) +
  geom_line(color = "steelblue") + geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
  labs(title = "Model 1 Residuals vs. Time", x = "Date", y = "Residuals") + theme_minimal()
print(p_res_time_m1)
```

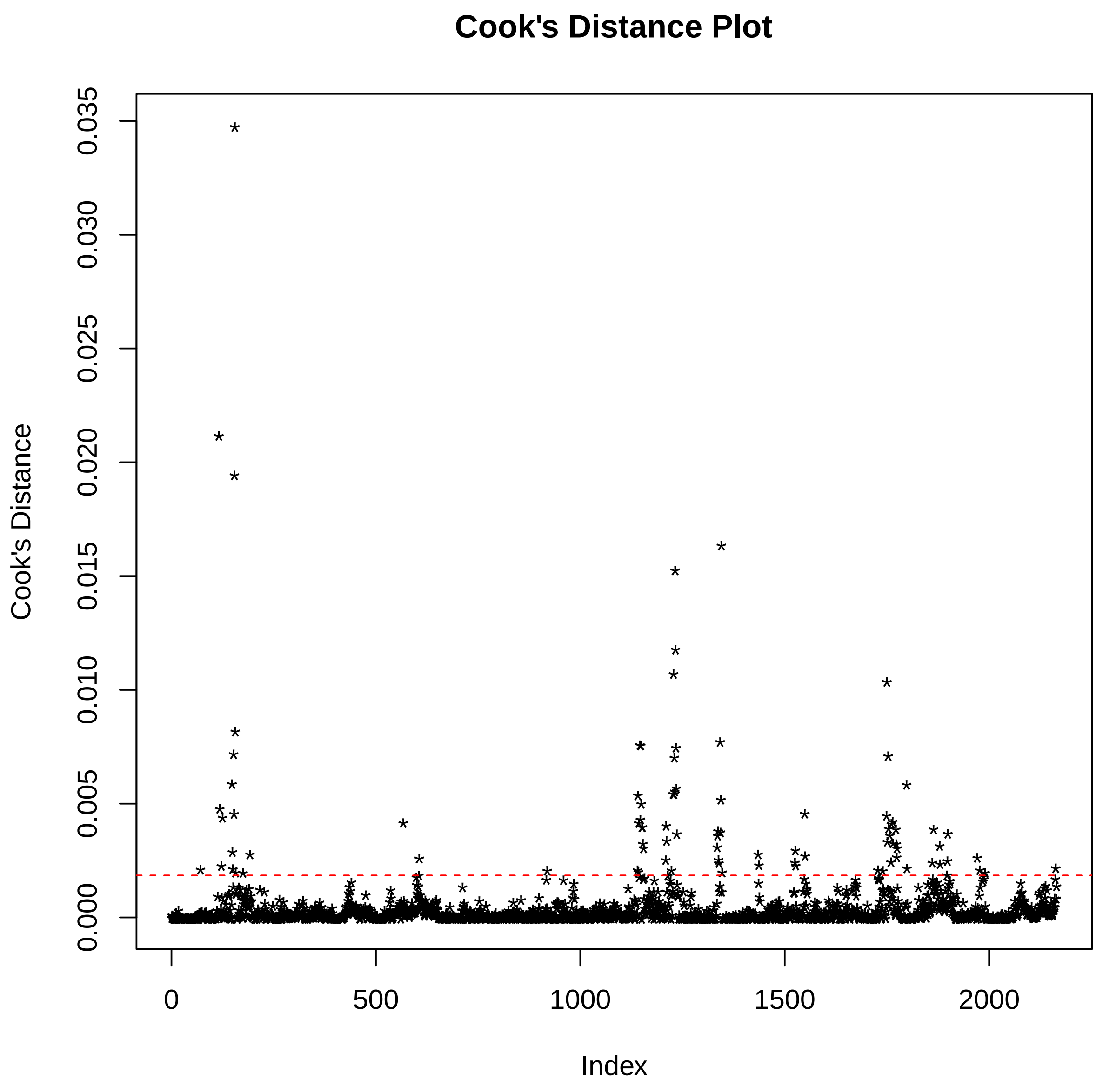


Figure 5: Regression Diagnostic Plots for OLS Model

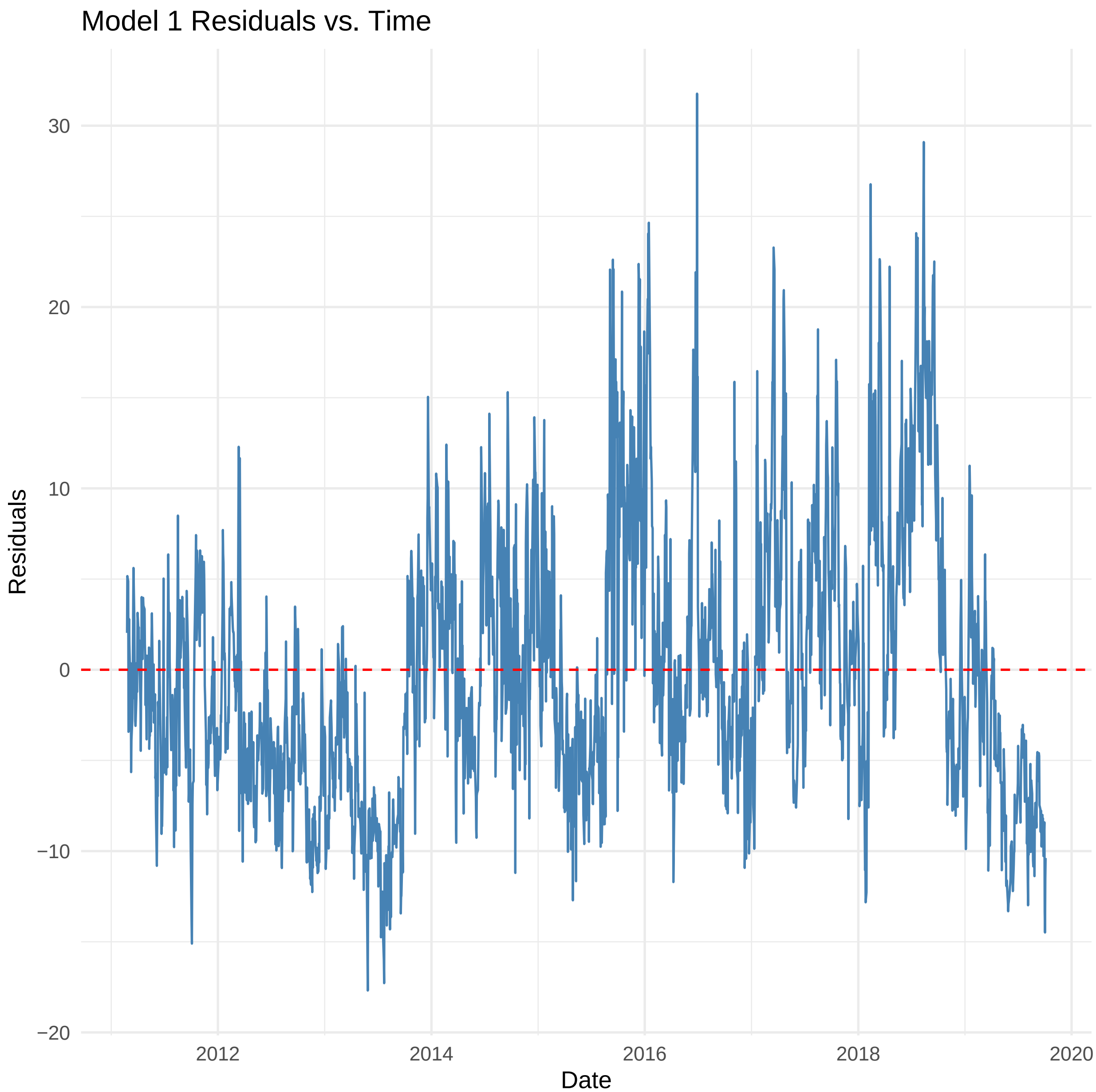


Figure 6: Regression Diagnostic Plots for OLS Model

```r
nw_lag_m1 <- floor(4*(nobs(model1)/100)^(2/9))
model1_hac_summary <- coeftest(model1, vcov. = NeweyWest(model1, lag = nw_lag_m1, prewhite = FALSE, ad

stargazer(model1,
          type = "latex",
          title = "Table 2: SKEW Index Regression with Newey-West HAC SEs (Model 1)",
          align = TRUE,
          dep.var.labels = "CBOE SKEW Index (Daily)",
          covariate.labels = my_covariate_labels_m1_centered,
          coef = list(coef(model1)),
          se = list(model1_hac_summary[, "Std. Error"]),
          t = list(model1_hac_summary[, "t value"]),
          p = list(model1_hac_summary[, "Pr(>|t|)"]),
          omit.stat = c("ser", "rsq", "f", "adj.rsq"),
          add.lines = list(
              c("Observations", formatC(nobs(model1), format="d", big.mark=",")),
              c("R-squared (OLS)", format(round(summary(model1)$r.squared, 3), nsmall = 3)),
              c("Adj. R-squared (OLS)", format(round(summary(model1)$adj.r.squared, 3), nsmall = 3)),
              c("Newey-West Lag Chosen", nw_lag_m1),
              c("Durbin-Watson Stat. (OLS)", format(round(dwtest(model1)$statistic,2),nsmall=2))
          ),
          star.cutoffs = c(0.05, 0.01, 0.001),
          notes = "$^{*}$p$<$.05; $^{**}$p$<$.01; $^{***}$p$<$.001. Newey-West HAC Standard Errors.",
          notes.align = "l", notes.append = FALSE, notes.label = "",
          header = FALSE, float = FALSE, no.space = TRUE, font.size = "small",
          digits = 3
)
```

	Dependent variable:
	CBOE SKEW Index (Daily)
VIX (t-1)	-1.641^{***}
	(0.308)
VIX^2 (t-1)	0.030^{***}
	(0.007)
Realized Vol (t-1, 21d)	-19.189^{*}
	(9.353)
Market Return (t-1, 21d)	-11.443
	(11.445)
AAII Sentiment (t-1, Bullish %)	6.858
	(5.337)
SPX Put-Call Ratio (t-1)	2.184^{*}
	(0.980)
$\text{VIX}_c\ (t-1) \times \text{Sent}_c\ (t-1)$	-0.125
	(1.034)
Constant	142.330^{***}
	(3.344)
Observations	2,165
R-squared (OLS)	0.160
Adj. R-squared (OLS)	0.158
Newey-West Lag Chosen	7
Durbin-Watson Stat. (OLS)	0.27
Observations	2,165

*p<.05; **p<.01; ***p<.001. Newey-West HAC Standard Errors.

```r
final_model_data_log <- final_model_data %>%
filter(SKEW > 0 & VIX_lag1 > 0 & RealizedVol_lag1 > 0 & PC_Ratio_lag1 > 0 & Sentiment_lag1 > 0)

final_model_data_log <- final_model_data_log %>%
  mutate(
    log_SKEW = log(SKEW),
    log_VIX_lag1 = log(VIX_lag1),
    log_RealizedVol_lag1 = log(RealizedVol_lag1),
    log_PC_Ratio_lag1 = log(PC_Ratio_lag1),
    log_Sentiment_lag1 = log(Sentiment_lag1)
  ) %>%
  mutate(
    log_VIX_lag1_centered_logmodel = log_VIX_lag1 - mean(log_VIX_lag1, na.rm=TRUE),
    log_Sentiment_lag1_centered_logmodel = log_Sentiment_lag1 - mean(log_Sentiment_lag1, na.rm=TRUE),
    logVIX_logSent_Interact_centered_lag1 = log_VIX_lag1_centered_logmodel * log_Sentiment_lag1_cente
  )

  model2_log <- lm(log_SKEW ~ log_VIX_lag1 + log_RealizedVol_lag1 +
                       MarketReturn_lag1 + log_Sentiment_lag1 + log_PC_Ratio_lag1 +
                       logVIX_logSent_Interact_centered_lag1,
                     data = final_model_data_log)

  my_covariate_labels_m2_log <- c(
  "log(VIX (t-1))",
  "log(RealizedVol (t-1))",
  "Market Return (t-1)",
  "log(Sentiment (t-1))",
  "log(P/C Ratio (t-1))",
  "$\\log(\\text{VIX}_c\\ (t{-}1)) \\times \\log(\\text{Sent}_c\\ (t{-}1))$"
)

  stargazer(model2_log,
          type = "latex",
          title = "Table 3: OLS Regression for log(SKEW) (Model 2)",
          align = TRUE,
          dep.var.labels = "log(CBOE SKEW Index)",
          covariate.labels = my_covariate_labels_m2_log,
          ci = TRUE, ci.level=0.95, single.row=FALSE,
          omit.stat=c("ser","rsq","f"),
          add.lines = list(
              c("Observations", formatC(nobs(model2_log), format="d", big.mark=",")),
              c("R-squared", format(round(summary(model2_log)$r.squared,3),nsmall=3)),
              c("Adj. R-squared", format(round(summary(model2_log)$adj.r.squared,3),nsmall=3)),
              c("F-statistic", paste0(format(round(summary(model2_log)$fstatistic[["value"]],2),nsma
                                    ifelse(pf(summary(model2_log)$fstatistic[["value"]],
                                              summary(model2_log)$fstatistic[["numdf"]],
                                              summary(model2_log)$fstatistic[["dendf"]],
                                              lower.tail=FALSE)<0.001,"^{***}",
                                           ifelse(pf(summary(model2_log)$fstatistic[["value"]],
                                                     summary(model2_log)$fstatistic[["numdf"]],
                                                     summary(model2_log)$fstatistic[["dendf"]],
                                                     lower.tail=FALSE)<0.01,"^{**}",
                                                  ifelse(pf(summary(model2_log)$fstatistic[["value"]]
```

```r
                                                    summary(model2_log)$fstatistic[["numdf"]]
                                                    summary(model2_log)$fstatistic[["dendf"]]
                                                    lower.tail=FALSE)<0.05,"^{*}", ""))),
                                  " (df = ", summary(model2_log)$fstatistic[["numdf"]], ", ", summa
          ),
          star.cutoffs = c(0.05, 0.01, 0.001),
          notes = "$^{*}$p$<$.05; $^{**}$p$<$.01; $^{***}$p$<$.001. OLS Standard Errors.",
          notes.align="l", notes.append=FALSE, notes.label="",
          header=FALSE, float=FALSE, no.space=TRUE, font.size="small",
          digits = 3
)
```

	Dependent variable:
	log(CBOE SKEW Index)
log(VIX (t-1))	-0.043^{***}
	(-0.058,-0.028)
log(RealizedVol (t-1))	-0.038^{***}
	(-0.046,-0.029)
Market Return (t-1)	-0.093^{*}
	(-0.181,-0.006)
log(Sentiment (t-1))	0.020^{**}
	(0.008,0.032)
log(P/C Ratio (t-1))	0.020^{***}
	(0.008,0.031)
log(VIX$_c$ $(t-1)$) $\times$ log(Sent$_c$ $(t-1)$)	-0.044^{*}
	(-0.086,-0.002)
Constant	4.897^{***}
	(4.837,4.957)
Observations	2,165
R-squared	0.179
Adj. R-squared	0.177
F-statistic	$78.51^{***}(df = 6, 2158)$
Observations	2,165
Adjusted R^2	0.177

*p<.05; **p<.01; ***p<.001. OLS Standard Errors.

```r
par(mfrow=c(2,2)); plot(model2_log); par(mfrow=c(1,1))
```

```r
crPlots_m2_output <- crPlots(model2_log,
                             terms= ~ .,
                             layout=NULL,
                             ask=FALSE,
                             smooth=list(smoother=loessLine,
                                         col.lines="red",
                                         spread=FALSE),
                             col="darkgray",
                             pch=1,
                             cex=0.7,
                             main="Component+Residual Plots (Model 2)")
```

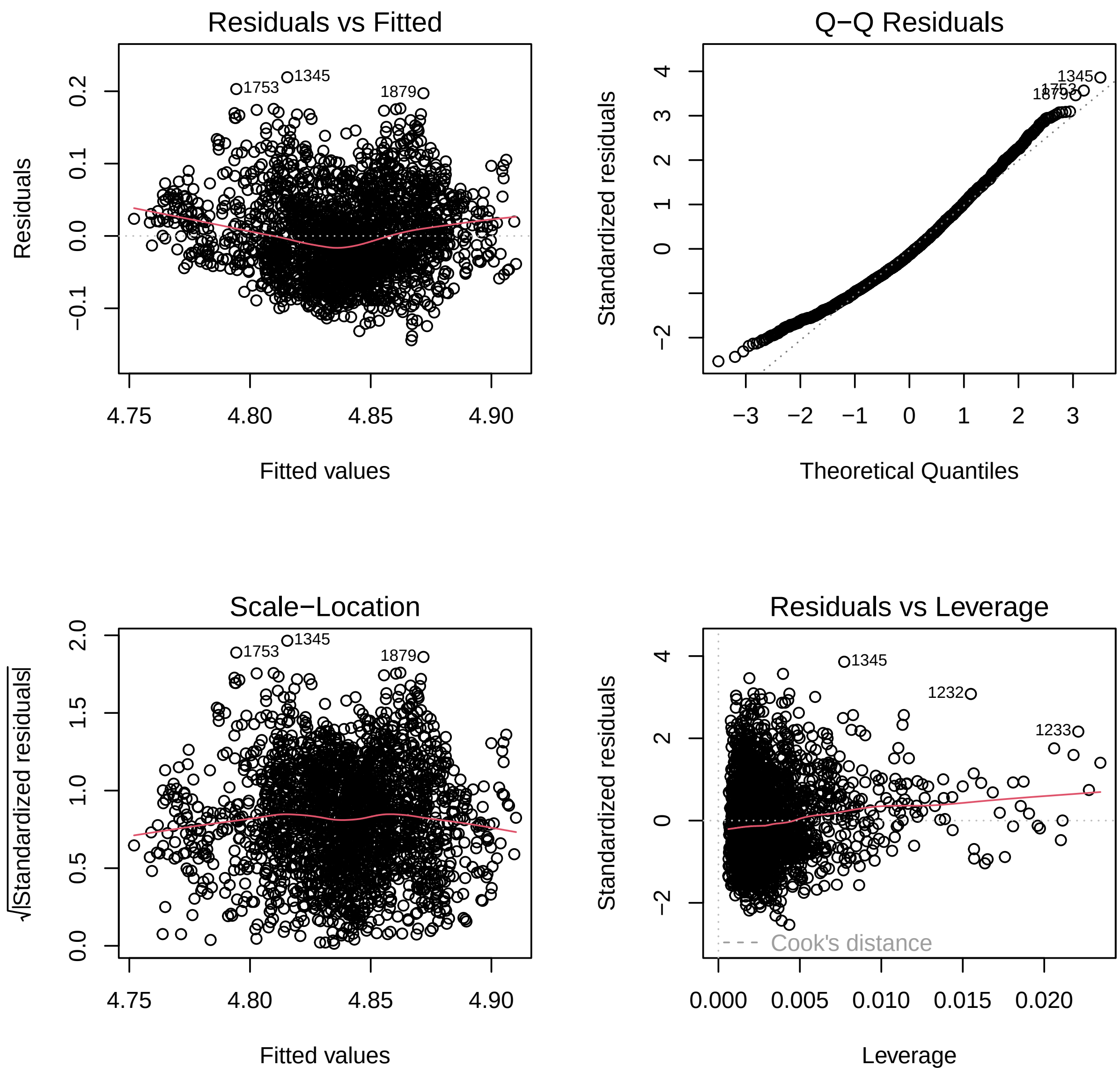


Figure 7: Regression Diagnostic Plots for Log Transformed Model

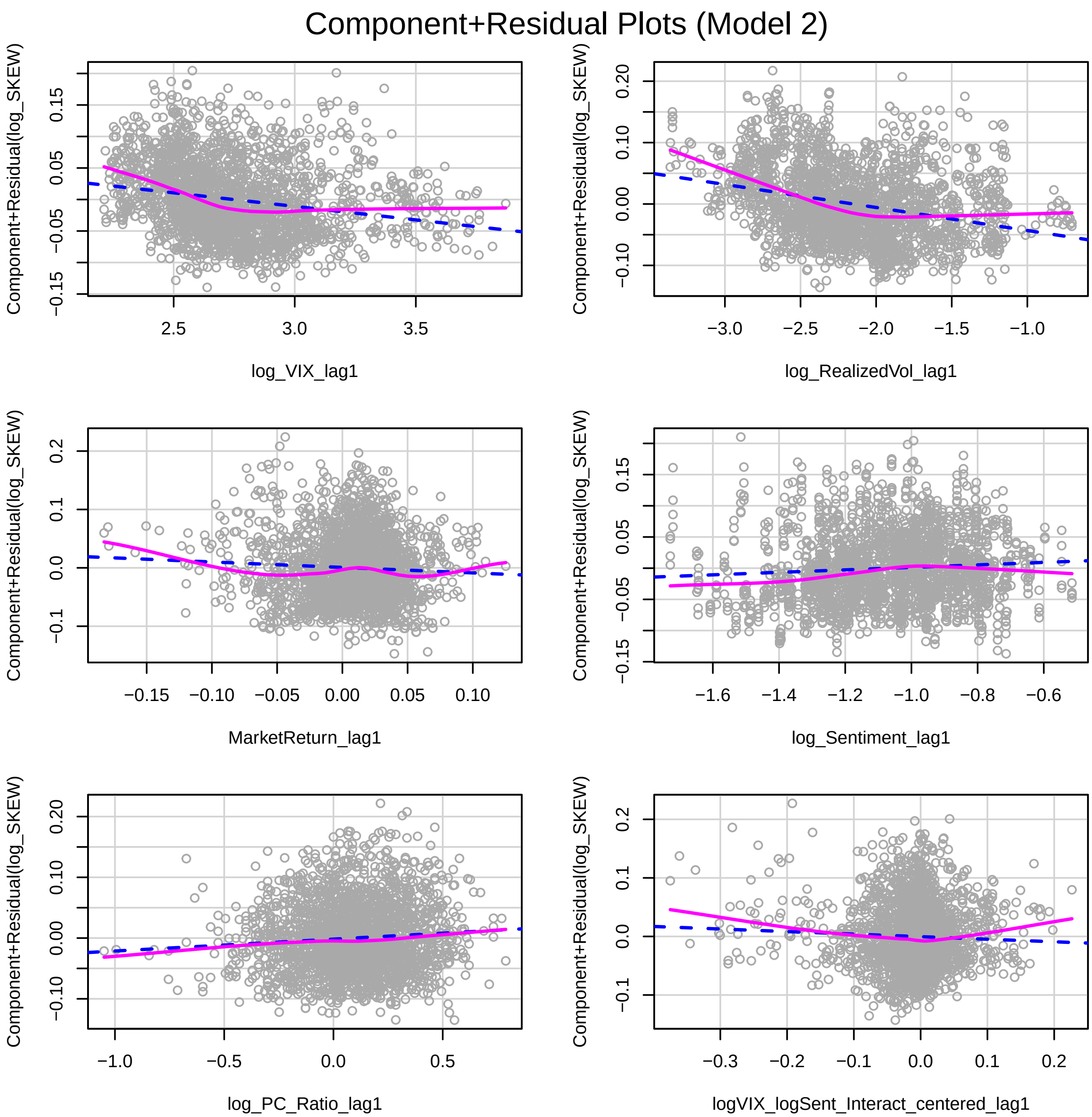


Figure 8: Regression Diagnostic Plots for Log Transformed Model

```r
shapiro_test_m2 <- shapiro.test(residuals(model2_log))
hist(residuals(model2_log), breaks=50, main="Residuals Histogram (Model 2)", col="lightcoral")
```

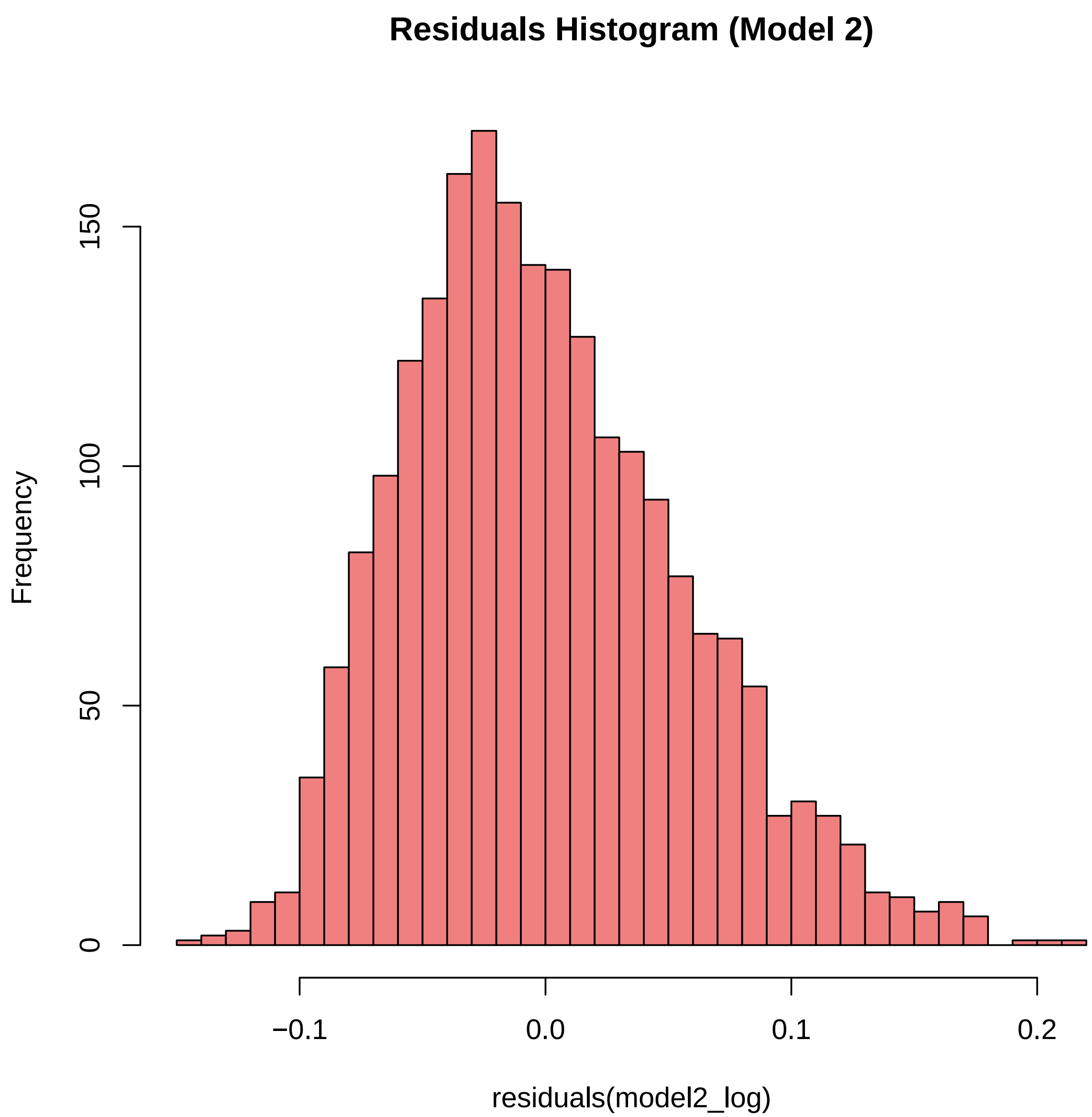


Figure 9: Regression Diagnostic Plots for Log Transformed Model

```r
skewness_m2_res <- skewness(residuals(model2_log))
kurtosis_m2_res <- kurtosis(residuals(model2_log))-3

bp_test_m2 <- bptest(model2_log, studentize=FALSE)

dw_test_m2 <- dwtest(model2_log)
par(mfrow=c(1,2)); acf(residuals(model2_log), main="ACF Residuals (Model 2)"); pacf(residuals(model2
```

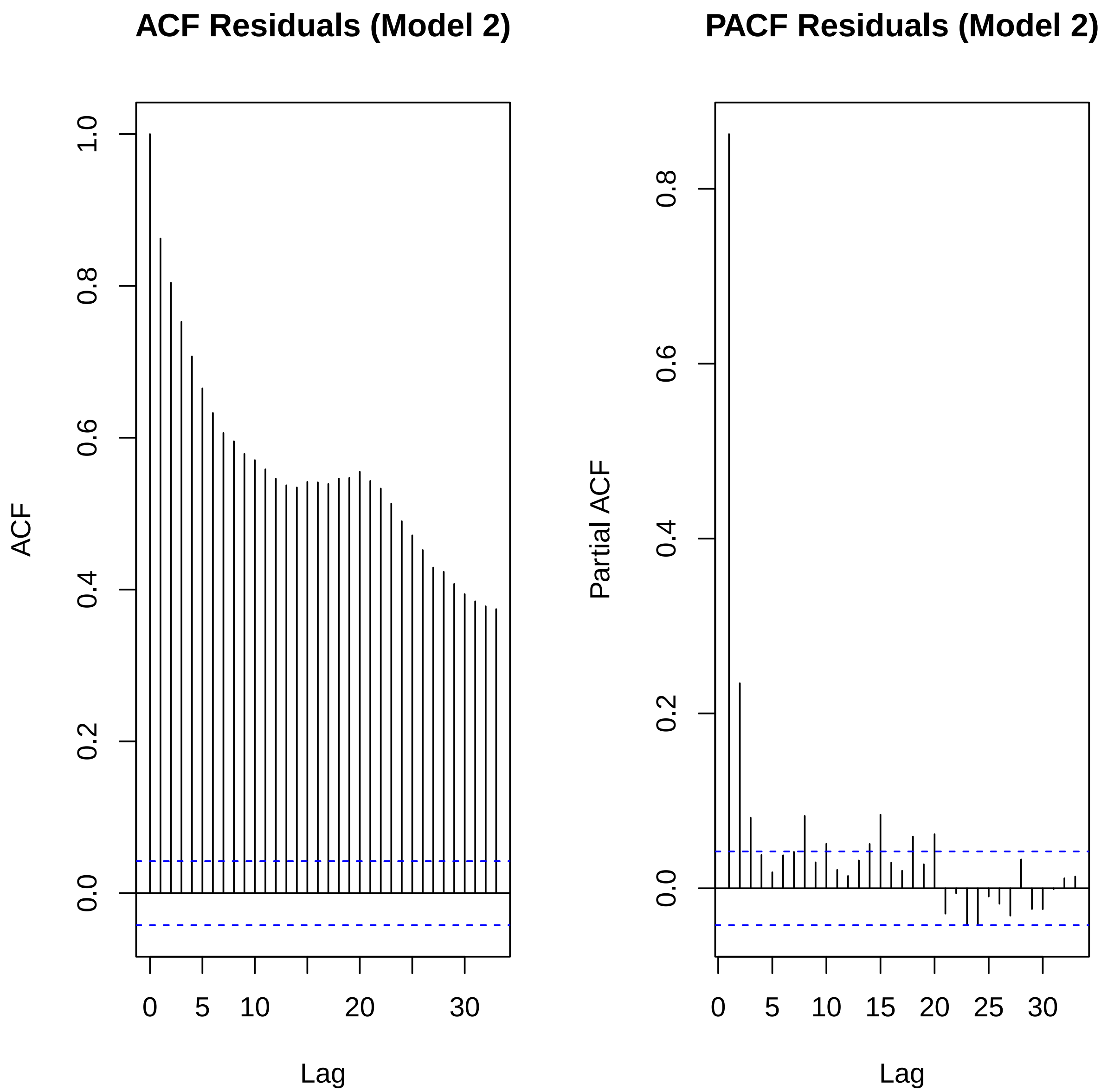


Figure 10: Regression Diagnostic Plots for Log Transformed Model

```r
ljung_box_10_m2 <- Box.test(residuals(model2_log), lag=10, type="Ljung-Box")
ljung_box_20_m2 <- Box.test(residuals(model2_log), lag=20, type="Ljung-Box")

vif_m2 <- vif(model2_log)

cooksd_m2 <- cooks.distance(model2_log)
plot(cooksd_m2, type="h", pch="*", cex=1, main="Cook's Distance Plot (Model 2)", ylab="Cook's Distan
abline(h = 4/nobs(model2_log), col="red", lty=2)
```

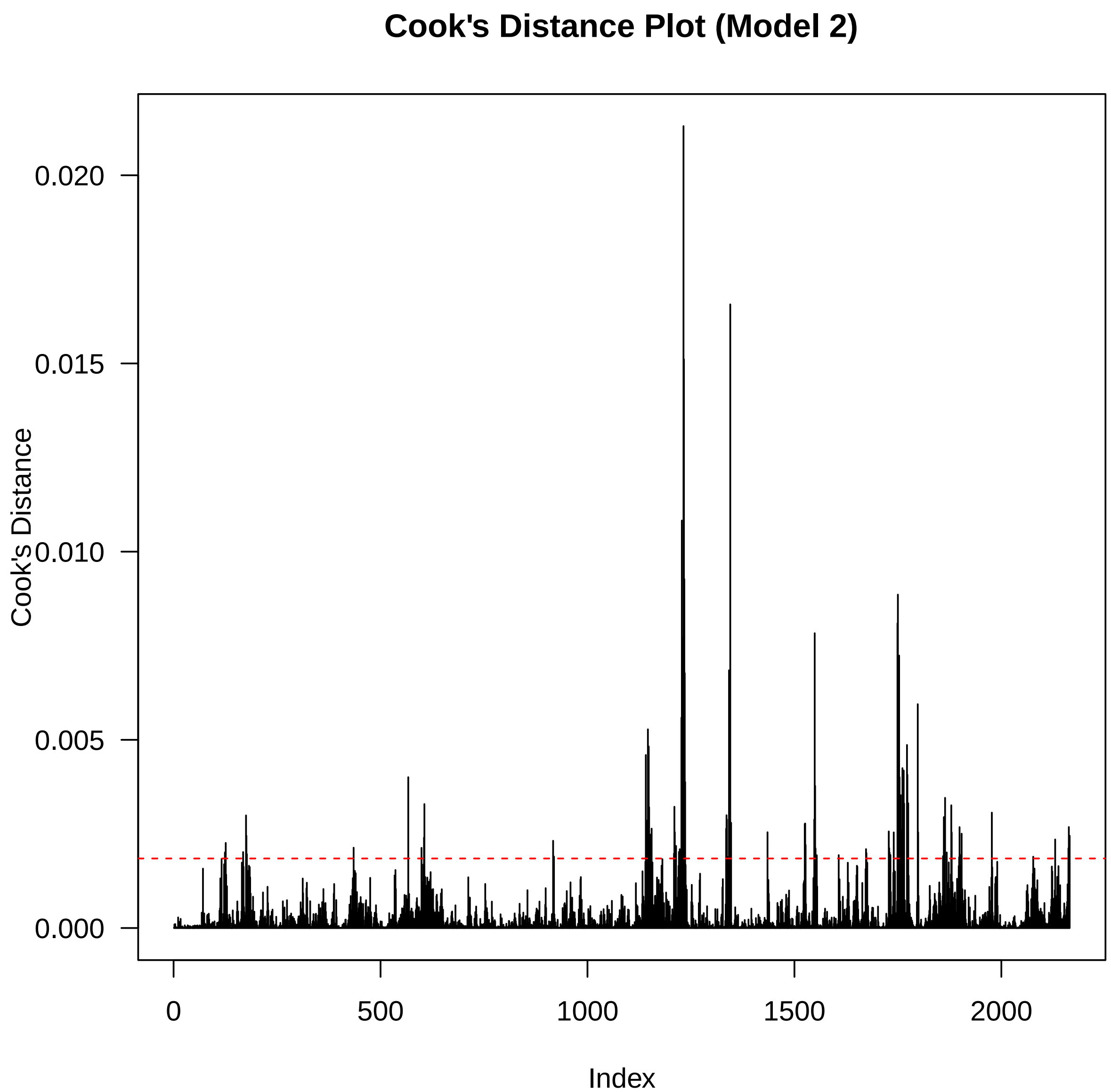


Figure 11: Regression Diagnostic Plots for Log Transformed Model

```r
influential_threshold_4_n_m2 <- 4/nobs(model2_log)
influential_points_4_n_m2 <- which(cooksd_m2 > influential_threshold_4_n_m2)

if (!is.null(model2_log$na.action) && length(model2_log$na.action) > 0 && length(final_model_data_lo
  residuals_df_m2 <- data.frame(Date = final_model_data_log$Date[-model2_log$na.action], Residuals =
} else if (nrow(final_model_data_log) == length(residuals(model2_log))) {
   residuals_df_m2 <- data.frame(Date = final_model_data_log$Date, Residuals = residuals(model2_log)
} else {
  residuals_df_m2 <- NULL
}
if(!is.null(residuals_df_m2)){
  p_res_time_m2 <- ggplot(residuals_df_m2, aes(x = Date, y = Residuals)) +
    geom_line(color = "coral") + geom_hline(yintercept = 0, linetype = "dashed", color = "red") +
    labs(title = "Model 2 (Log) Residuals vs. Time", x = "Date", y = "Residuals") + theme_minimal()
  print(p_res_time_m2)
}
```

```r
cat("\n--- Shapiro-Wilk Normality Test (Model 2) ---\n")
```

```
##
## --- Shapiro-Wilk Normality Test (Model 2) ---
```

```r
print(shapiro_test_m2)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model2_log)
## W = 0.9798, p-value < 2.2e-16
```

```r
cat("\nSkewness (Model 2 Residuals):", round(skewness_m2_res,3), "\n")
```

```
##
## Skewness (Model 2 Residuals): 0.548
```

```r
cat("Excess Kurtosis (Model 2 Residuals):", round(kurtosis_m2_res,3), "\n")
```

```
## Excess Kurtosis (Model 2 Residuals): 0.083
```

```r
cat("\n--- Breusch-Pagan Homoscedasticity Test (Model 2) ---\n")
```

```
##
## --- Breusch-Pagan Homoscedasticity Test (Model 2) ---
```

```r
print(bp_test_m2)
```

```
##
##  Breusch-Pagan test
##
## data:  model2_log
## BP = 35.793, df = 6, p-value = 3.024e-06
```

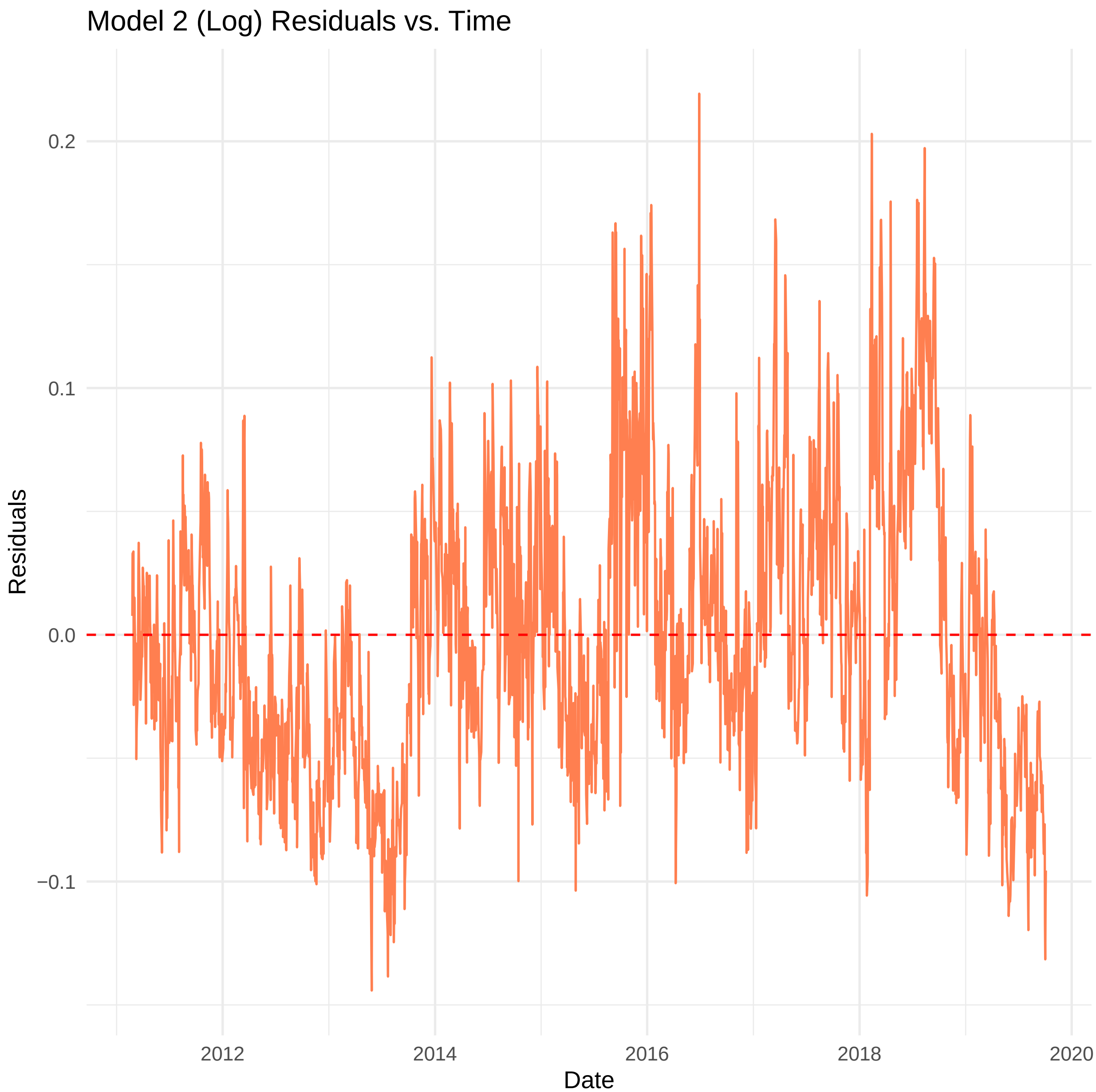


Figure 12: Regression Diagnostic Plots for Log Transformed Model

```r
cat("\n--- Durbin-Watson Autocorrelation Test (Model 2) ---\n")
```

```
##
## --- Durbin-Watson Autocorrelation Test (Model 2) ---
```

```r
print(dw_test_m2)
```

```
##
##  Durbin-Watson test
##
## data:  model2_log
## DW = 0.27364, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
cat("\n--- Ljung-Box Autocorrelation Test (10 lags, Model 2) ---\n")
```

```
##
## --- Ljung-Box Autocorrelation Test (10 lags, Model 2) ---
```

```r
print(ljung_box_10_m2)
```

```
##
##  Box-Ljung test
##
## data:  residuals(model2_log)
## X-squared = 10167, df = 10, p-value < 2.2e-16
```

```r
cat("\n--- Ljung-Box Autocorrelation Test (20 lags, Model 2) ---\n")
```

```
##
## --- Ljung-Box Autocorrelation Test (20 lags, Model 2) ---
```

```r
print(ljung_box_20_m2)
```

```
##
##  Box-Ljung test
##
## data:  residuals(model2_log)
## X-squared = 16642, df = 20, p-value < 2.2e-16
```

```r
cat("\n--- Variance Inflation Factors (Model 2) ---\n")
```

```
##
## --- Variance Inflation Factors (Model 2) ---
```

```r
print(vif_m2)
```

```
##                           log_VIX_lag1                  log_RealizedVol_lag1
##                               3.222413                              2.736576
##                      MarketReturn_lag1                    log_Sentiment_lag1
##                               1.590752                              1.189602
##                      log_PC_Ratio_lag1 logVIX_logSent_Interact_centered_lag1
##                               1.118945                              1.039206
```

```
    cat("\n--- Influential Points (Model 2) ---\n")
```

```
##
## --- Influential Points (Model 2) ---
```

```
    cat("Number of points with Cook's D > 4/n:", length(influential_points_4_n_m2), "\n")
```

```
## Number of points with Cook's D > 4/n: 100
```

```
    if(length(influential_points_4_n_m2) > 0 && length(influential_points_4_n_m2) < 20) {
        cat("Indices of points with Cook's D > 4/n (first 10):", paste(head(influential_points_4_n_m2, 
    }
    influential_points_1_m2 <- which(cooksd_m2 > 1)
    cat("Number of points with Cook's D > 1:", length(influential_points_1_m2), "\n")
```

```
## Number of points with Cook's D > 1: 0
```

```
    if(length(influential_points_1_m2) > 0) {
        cat("Indices of points with Cook's D > 1:", paste(influential_points_1_m2, collapse=", "), "\n")
    }
```

```
  nw_lag_m2 <- floor(4*(nobs(model2_log)/100)^(2/9))
  model2_log_hac_summary <- coeftest(model2_log, vcov. = NeweyWest(model2_log, lag = nw_lag_m2, prewhite
  print(model2_log_hac_summary)
```

t test of coefficients:

```
                                    Estimate Std. Error t value  Pr(>|t|)
```

(Intercept) 4.8966239 0.0672105 72.8551 < 2.2e-16 log_VIX_lag1 -0.0429481 0.0157489 -2.7271 0.0064419 log_RealizedVol_lag1 -0.0375033 0.0101894 -3.6806 0.0002384 MarketReturn_lag1 -0.0934942 0.0941889 -0.9926 0.3210044 log_Sentiment_lag1 0.0199118 0.0143564 1.3870 0.1655955 log_PC_Ratio_lag1 0.0195837 0.0080337 2.4377 0.0148620 logVIX_logSent_Interact_centered_lag1 -0.0436371 0.0496226 -0.8794 0.3792928

(Intercept) ***log_VIX_lag1*** *log_RealizedVol_lag1* ** MarketReturn_lag1
log_Sentiment_lag1
log_PC_Ratio_lag1 *
logVIX_logSent_Interact_centered_lag1
— Signif. codes: 0 '***' ***0.001*** '**' *0.01* '*' 0.05 '.' 0.1 ' ' 1

```r
colnames(model2_log_hac_summary) <- c("Estimate", "Std. Error", "t value", "Pr(>|t|)")


stargazer(model2_log,
        type = "latex",
        title = "Table 4: OLSLog(SKEW) Index Regression with Newey-West HAC SEs (Model 2)",
        align = TRUE,
        dep.var.labels = "log(CBOE SKEW Index)",
        covariate.labels = my_covariate_labels_m2_log,
        ci = TRUE, ci.level = 0.95, single.row = FALSE,
        omit.stat = c("ser", "rsq", "f"),
        add.lines = list(
            c("Observations", formatC(nobs(model2_log), format="d", big.mark=",")),
            c("R-squared", format(round(summary(model2_log)$r.squared, 3), nsmall = 3)),
            c("Adj. R-squared", format(round(summary(model2_log)$adj.r.squared, 3), nsmall = 3)),
            c("F-statistic", paste0(
                format(round(summary(model2_log)$fstatistic[["value"]], 2), nsmall=2),
                ifelse(pf(summary(model2_log)$fstatistic[["value"]],
                          summary(model2_log)$fstatistic[["numdf"]], summary(model2_log)$fstatistic[[
                       ifelse(pf(summary(model2_log)$fstatistic[["value"]], summary(model2_log)$fstati
                              ifelse(pf(summary(model2_log)$fstatistic[["value"]], summary(model2_log
                " (df = ", summary(model2_log)$fstatistic[["numdf"]], ", ", summary(model2_log)$fstati
        ),
        star.cutoffs = c(0.05, 0.01, 0.001),
        notes = c("$^{*}$p$<$.05; $^{**}$p$<$.01; $^{***}$p$<$.001. Newey-West HAC Standard Errors.")
        notes.align = "l",
        notes.append = FALSE,
        header = FALSE,
        float = FALSE,
        no.space = TRUE,
        font.size = "small",
        digits = 3)
```

	Dependent variable:
	log(CBOE SKEW Index)
log(VIX (t-1))	-0.043^{***}
	(-0.058,-0.028)
log(RealizedVol (t-1))	-0.038^{***}
	(-0.046,-0.029)
Market Return (t-1)	-0.093^{*}
	(-0.181,-0.006)
log(Sentiment (t-1))	0.020^{**}
	(0.008,0.032)
log(P/C Ratio (t-1))	0.020^{***}
	(0.008,0.031)
log(VIX$_c$ $(t-1)$) $\times$ log(Sent$_c$ $(t-1)$)	-0.044^{*}
	(-0.086,-0.002)
Constant	4.897^{***}
	(4.837,4.957)
Observations	$2,165$
R-squared	0.179
Adj. R-squared	0.177
F-statistic	$78.51 * * * (df = 6, 2158)$
Observations	2,165
Adjusted R^2	0.177
Note:	*p<.05; **p<.01; ***p<.001. Newey-West HAC Standard Errors.

```r
model3_log <- lm(log_SKEW ~ log_VIX_lag1 + log_RealizedVol_lag1 + log_PC_Ratio_lag1,
                 data = final_model_data_log)

summary(model3_log)
```

```
##
## Call:
## lm(formula = log_SKEW ~ log_VIX_lag1 + log_RealizedVol_lag1 +
##     log_PC_Ratio_lag1, data = final_model_data_log)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.138506 -0.041790 -0.006619  0.037302  0.220947
##
## Coefficients:
##                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)           4.852184   0.028124 172.526  < 2e-16 ***
## log_VIX_lag1         -0.036167   0.007364  -4.911 9.73e-07 ***
## log_RealizedVol_lag1 -0.039505   0.004373  -9.033  < 2e-16 ***
## log_PC_Ratio_lag1     0.019308   0.005683   3.398 0.000692 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.05721 on 2161 degrees of freedom
## Multiple R-squared:  0.1729, Adjusted R-squared:  0.1718
## F-statistic: 150.6 on 3 and 2161 DF,  p-value: < 2.2e-16
```

```r
par(mfrow=c(2,2)); plot(model3_log); par(mfrow=c(1,1))
```

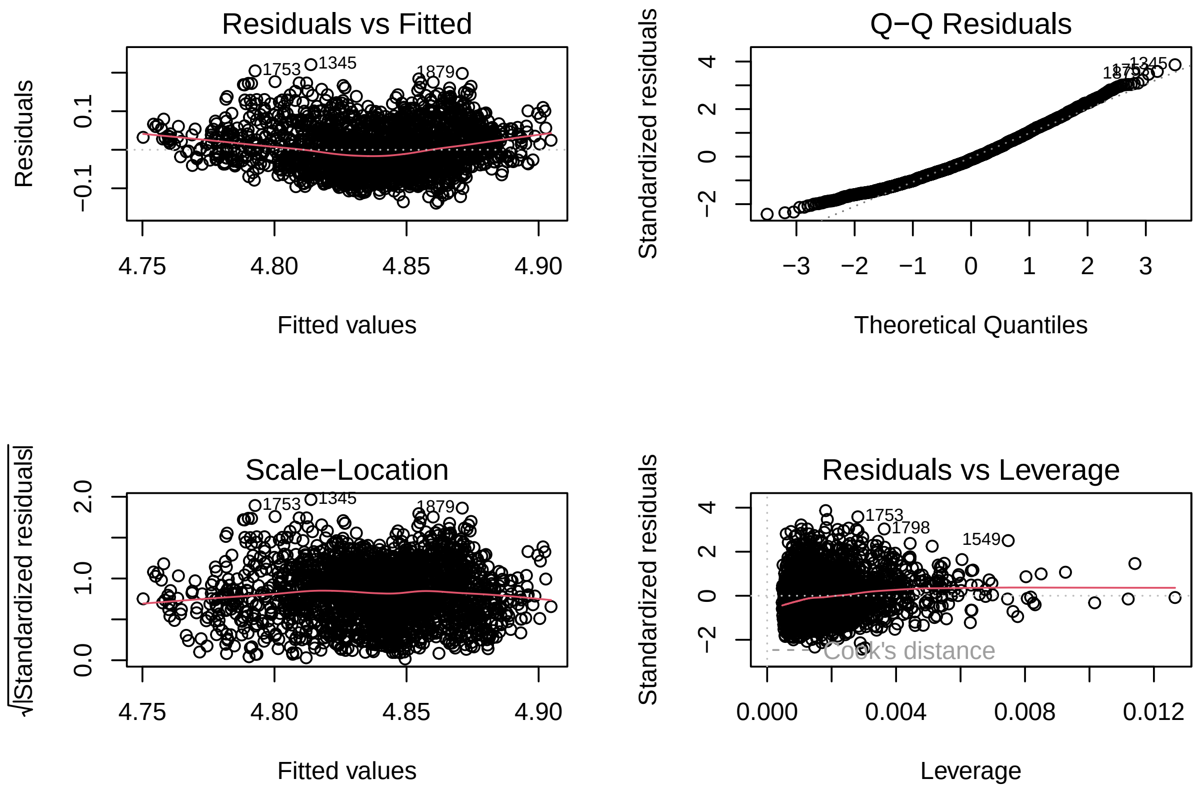


Figure 13: Standard Diagnostic Plots for Model 3 (OLS)

```r
crPlots(model3_log, terms = ~ ., layout=NULL, ask=FALSE, smooth=list(smoother=loessLine, col.lines="p
```

```r
cat("\n-- Normality of Residuals --\n")
```

```
##
## -- Normality of Residuals --
```

```r
hist(residuals(model3_log), breaks=50, main="Histogram of Residuals (Model 3 OLS)", col="salmon")
```

```r
shapiro_test_m3 <- shapiro.test(residuals(model3_log))
print(shapiro_test_m3)
```

```
##
##  Shapiro-Wilk normality test
##
## data:  residuals(model3_log)
## W = 0.97888, p-value < 2.2e-16
```

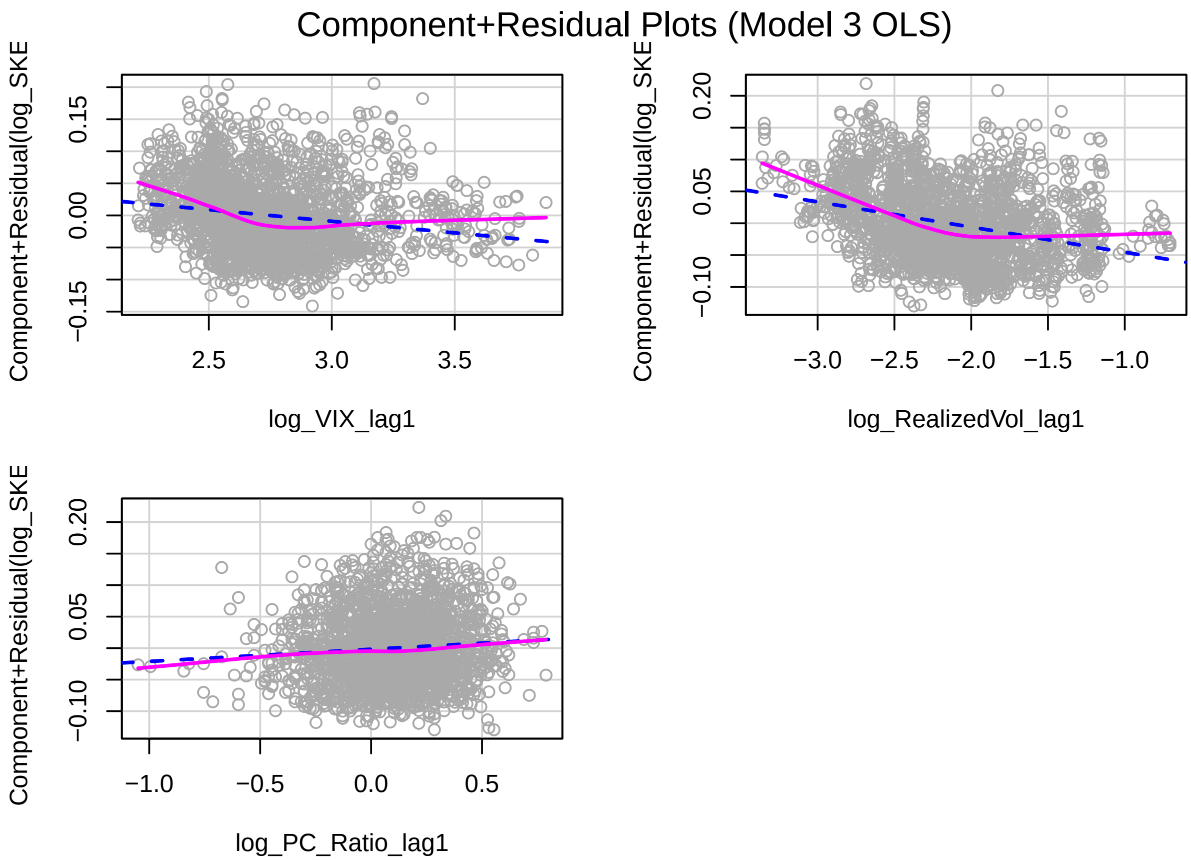


Figure 14: Standard Diagnostic Plots for Model 3 (OLS)

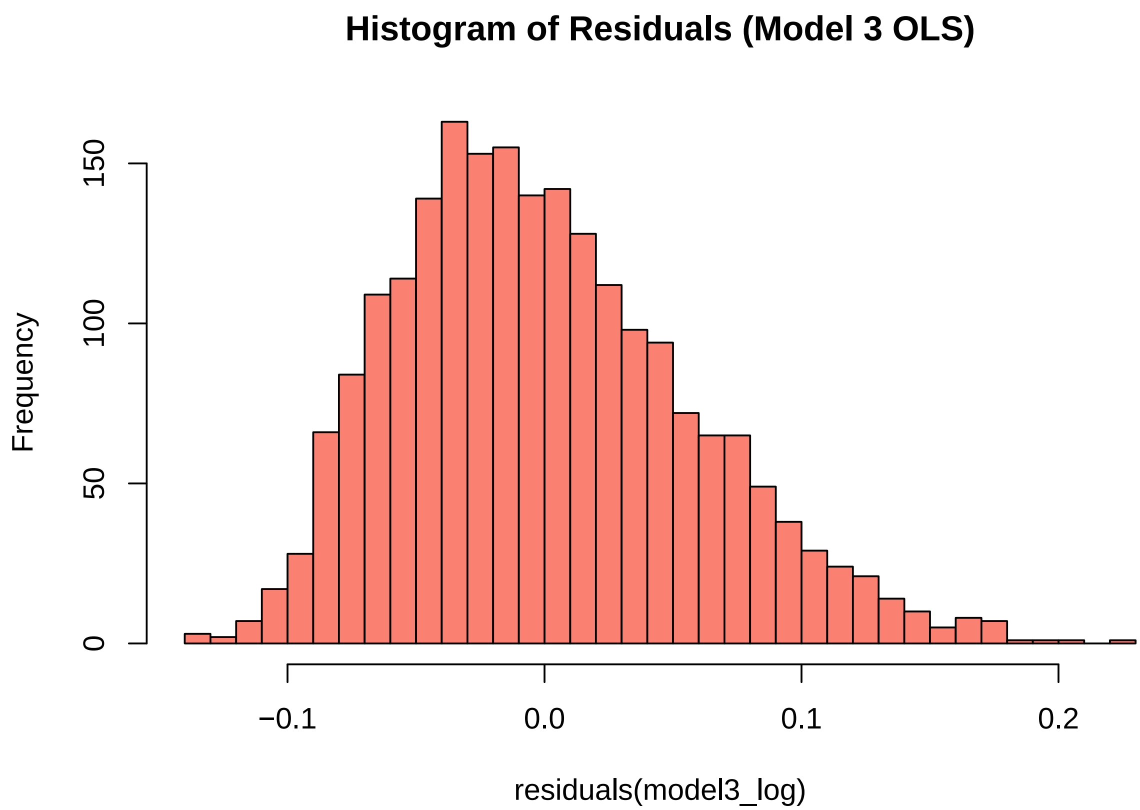


Figure 15: Standard Diagnostic Plots for Model 3 (OLS)

```r
cat("Skewness (Model 3 OLS Residuals):", skewness(residuals(model3_log)), "\n")
```

```
## Skewness (Model 3 OLS Residuals): 0.5525604
```

```r
cat("Excess Kurtosis (Model 3 OLS Residuals):", kurtosis(residuals(model3_log)) - 3, "\n")
```

```
## Excess Kurtosis (Model 3 OLS Residuals): 0.0569077
```

```r
cat("\n-- Homoscedasticity (Breusch-Pagan Test) --\n")
```

```
##
## -- Homoscedasticity (Breusch-Pagan Test) --
```

```r
bp_test_m3 <- bptest(model3_log, studentize=FALSE)
print(bp_test_m3)
```

```
##
##  Breusch-Pagan test
##
## data:  model3_log
## BP = 6.7456, df = 3, p-value = 0.08046
```

```r
cat("\n-- Independence of Residuals (Durbin-Watson & Ljung-Box) --\n")
```

```
##
## -- Independence of Residuals (Durbin-Watson & Ljung-Box) --
```

```r
dw_test_m3 <- dwtest(model3_log)
print(dw_test_m3)
```

```
##
##  Durbin-Watson test
##
## data:  model3_log
## DW = 0.27002, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0
```

```r
par(mfrow=c(1,2)); acf(residuals(model3_log), main="ACF Residuals (Model 3 OLS)"); pacf(residuals(mode
```

```r
ljung_box_10_m3 <- Box.test(residuals(model3_log), lag=10, type="Ljung-Box")
print(ljung_box_10_m3)
```

```
##
##  Box-Ljung test
##
## data:  residuals(model3_log)
## X-squared = 10340, df = 10, p-value < 2.2e-16
```

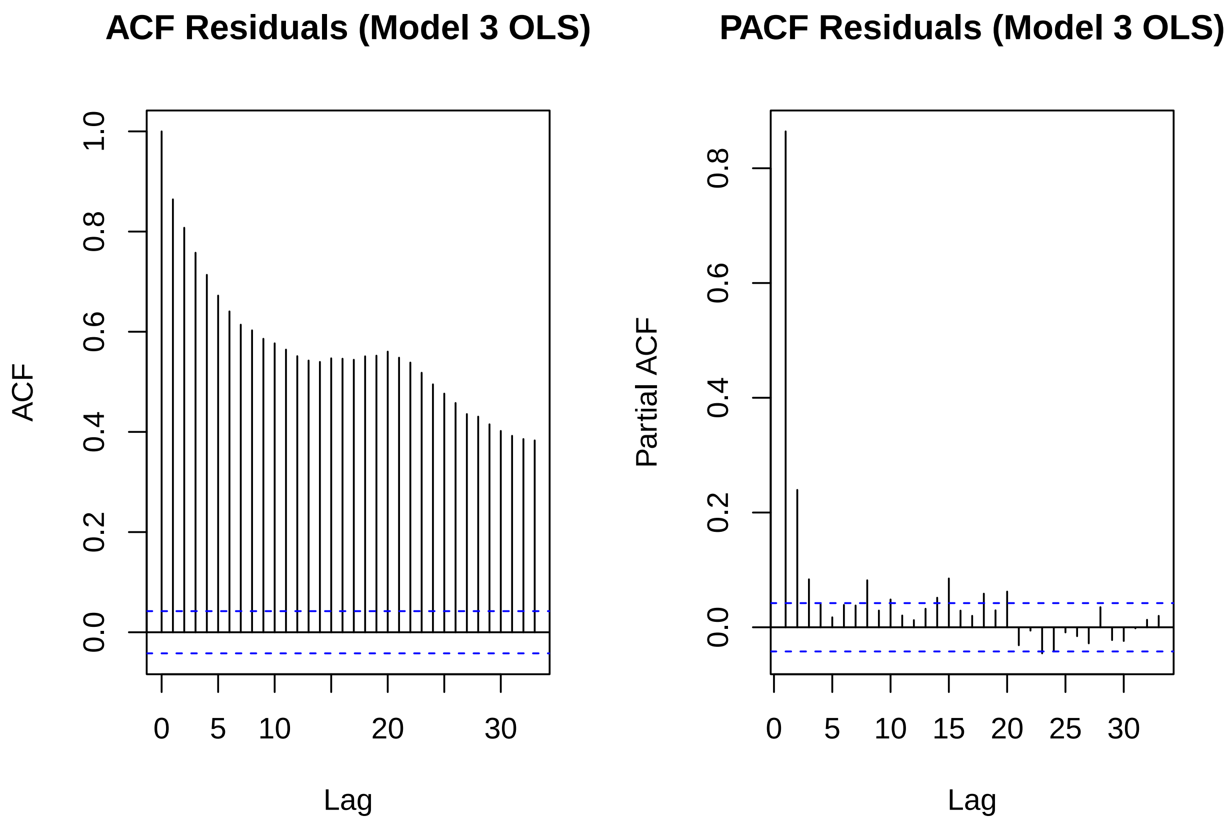


Figure 16: Standard Diagnostic Plots for Model 3 (OLS)

```r
cat("\n-- Multicollinearity (VIFs) --\n")
```

```
##
## -- Multicollinearity (VIFs) --
```

```r
vif_m3 <- vif(model3_log)
print(vif_m3)
```

```
##         log_VIX_lag1 log_RealizedVol_lag1     log_PC_Ratio_lag1
##             2.843912             2.693415              1.118418
```

```r
cat("\n-- Influential Points (Cook's Distance) --\n")
```

```
##
## -- Influential Points (Cook's Distance) --
```

```r
cooksd_m3 <- cooks.distance(model3_log)
plot(cooksd_m3, type="h", main="Cook's Distance Plot (Model 3 OLS)", ylab="Cook's Distance")
abline(h = 4/nobs(model3_log), col="darkred", lty=2)
```

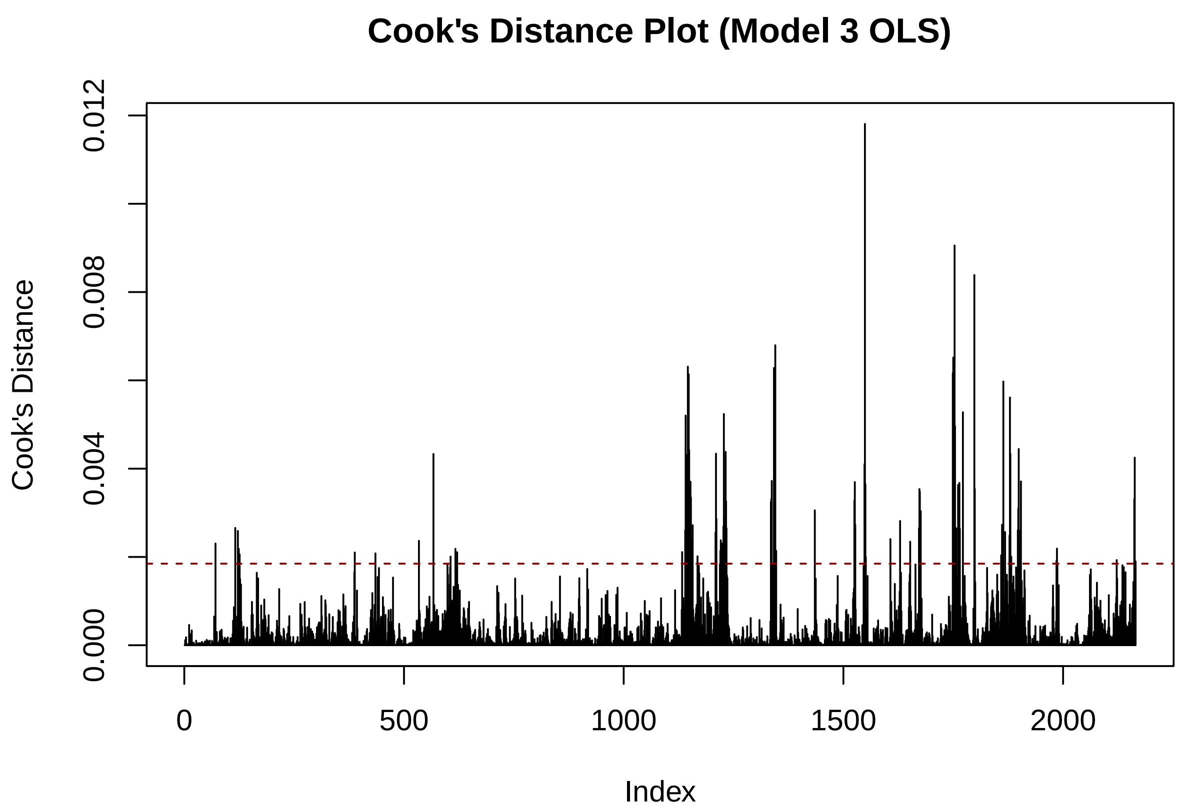


Figure 17: Standard Diagnostic Plots for Model 3 (OLS)

```r
cat("\n--- Model 3 with HAC Standard Errors ---\n")
```

```
##
## --- Model 3 with HAC Standard Errors ---
```

```r
nw_lag_m3 <- floor(4*(nobs(model3_log)/100)^(2/9))
model3_log_hac_summary <- coeftest(model3_log, vcov. = NeweyWest(model3_log, lag = nw_lag_m3, prewhite
print(model3_log_hac_summary)
```

```
##
## t test of coefficients:
##
##                        Estimate Std. Error t value  Pr(>|t|)
## (Intercept)           4.8521842  0.0591626 82.0144 < 2.2e-16 ***
## log_VIX_lag1         -0.0361670  0.0148889 -2.4291   0.01522 *
## log_RealizedVol_lag1 -0.0395053  0.0101164 -3.9051 9.709e-05 ***
## log_PC_Ratio_lag1     0.0193084  0.0080644  2.3943   0.01674 *
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
```

```r
cat("\n--- Stargazer Table for Model 3 (HAC SEs) ---\n")
```

```
##
## --- Stargazer Table for Model 3 (HAC SEs) ---
```

```r
my_covariate_labels_m3_log <- c(
  "log(VIX (t-1))",
  "log(RealizedVol (t-1))",
  "log(P/C Ratio (t-1))"
)

nw_lag_m3_for_table <- floor(4*(nobs(model3_log)/100)^(2/9))
model3_log_hac_summary <- coeftest(model3_log, vcov. = NeweyWest(model3_log, lag = nw_lag_m3_for_table
dw_test_m3 <- dwtest(model3_log)



my_covariate_labels_m3_log <- c(
  "log(VIX (t-1))",
  "log(RealizedVol (t-1))",
  "log(P/C Ratio (t-1))"
)
```

```r
stargazer(model3_log,
          type = "latex",
          title = "Table 5: Parsimonious Log(SKEW) Index Regression with Newey-West HAC SEs (Model 3)",
          align = TRUE,
          dep.var.labels = "log(CBOE SKEW Index)",
          covariate.labels = my_covariate_labels_m3_log,
          coef = list(coef(model3_log)),
          se = list(model3_log_hac_summary[, "Std. Error"]),
```

```r
          t = list(model3_log_hac_summary[, "t value"]),
          p = list(model3_log_hac_summary[, "Pr(>|t|)"]),
          ci = TRUE,
          omit.stat = c("ser", "f", "bic", "aic", "ll"),
          add.lines = list(
            c("Observations", formatC(nobs(model3_log), format="d", big.mark=",")),
            c("R-squared (OLS)", format(round(summary(model3_log)$r.squared, 3), nsmall = 3)),
            c("Adj. R-squared (OLS)", format(round(summary(model3_log)$adj.r.squared, 3), nsmall = 3)),
            c("Newey-West Lag Chosen", nw_lag_m3_for_table),
            c("Durbin-Watson Stat. (OLS)", format(round(dw_test_m3$statistic, 2), nsmall=2))
          ),
          star.cutoffs = c(0.05, 0.01, 0.001),
          notes = "$^{*}$p$<$.05; $^{**}$p$<$.01; $^{***}$p$<$.001. Newey-West HAC Standard Errors.",
          notes.align = "l",
          notes.append = FALSE,
          header = FALSE,
          float = FALSE,
          no.space = TRUE,
          font.size = "small",
          digits = 3)
```

	Dependent variable:
	log(CBOE SKEW Index)
log(VIX (t-1))	-0.036^{*}
	(-0.065,-0.007)
log(RealizedVol (t-1))	-0.040^{***}
	(-0.059,-0.020)
log(P/C Ratio (t-1))	0.019^{*}
	(0.004,0.035)
Constant	4.852^{***}
	(4.736,4.968)
Observations	2, 165
R-squared (OLS)	0.173
Adj. R-squared (OLS)	0.172
Newey-West Lag Chosen	7
Durbin-Watson Stat. (OLS)	0.27
Observations	2,165
R^2	0.173
Adjusted R^2	0.172
Note:	*p<.05; **p<.01; ***p<.001. Newey-West HAC Standard Errors.

```r
calculate_rmse <- function(actual, predicted) {
  sqrt(mean((actual - predicted)^2, na.rm = TRUE))
}

calculate_rsquared_oos <- function(actual, predicted) {
  tss <- sum((actual - mean(actual, na.rm = TRUE))^2, na.rm = TRUE)
  rss <- sum((actual - predicted)^2, na.rm = TRUE)
  if (tss == 0) return(NA)
  return(1 - (rss / tss))
}

cat("\n--- Model 1 (Levels): Rolling Origin k-Fold Cross-Validation ---\n")
```

```
##
## --- Model 1 (Levels): Rolling Origin k-Fold Cross-Validation ---
```

```
data_for_cv_m1 <- final_model_data
k_folds_m1 <- 5
n_obs_m1 <- nrow(data_for_cv_m1)
initial_train_percent_m1 <- 0.70
initial_train_window_m1 <- floor(initial_train_percent_m1 * n_obs_m1)
remaining_obs_m1 <- n_obs_m1 - initial_train_window_m1
fold_size_m1 <- floor(remaining_obs_m1 / k_folds_m1)

rmse_scores_m1 <- numeric(k_folds_m1)
rsquared_oos_scores_m1 <- numeric(k_folds_m1)

model1_formula_cv <- SKEW ~ VIX_lag1 + VIX_sq_lag1 + RealizedVol_lag1 +
                            MarketReturn_lag1 + Sentiment_lag1 + PC_Ratio_lag1 +
                            VIX_Sent_Interact_centered_lag1

if (fold_size_m1 > 0) {
  for (i in 1:k_folds_m1) {
    train_end_idx <- initial_train_window_m1 + (i - 1) * fold_size_m1
    test_start_idx <- train_end_idx + 1
    if (i < k_folds_m1) {
      test_end_idx <- train_end_idx + fold_size_m1
    } else {
      test_end_idx <- n_obs_m1
    }

    if (test_start_idx > n_obs_m1) {
      rmse_scores_m1[i] <- NA
      rsquared_oos_scores_m1[i] <- NA
      cat(paste("Fold", i, "(Model 1): No test data remaining. Skipping.\n"))
      next
    }

    current_train_data_m1 <- data_for_cv_m1[1:train_end_idx, ]
    current_test_data_m1  <- data_for_cv_m1[test_start_idx:test_end_idx, ]

    if (nrow(current_test_data_m1) == 0) {
        rmse_scores_m1[i] <- NA
        rsquared_oos_scores_m1[i] <- NA
        cat(paste("Fold", i, "(Model 1): Test data has 0 rows. Skipping.\n"))
        next
    }

    model_fit_cv_m1 <- lm(model1_formula_cv, data = current_train_data_m1)
    predictions_cv_m1 <- predict(model_fit_cv_m1, newdata = current_test_data_m1)
    actuals_cv_m1 <- current_test_data_m1$SKEW

    rmse_scores_m1[i] <- calculate_rmse(actuals_cv_m1, predictions_cv_m1)
    rsquared_oos_scores_m1[i] <- calculate_rsquared_oos(actuals_cv_m1, predictions_cv_m1)
```

```r
    cat(paste("Fold", i, "RMSE (Model 1, levels):", round(rmse_scores_m1[i], 4),
              "| OOS R-squared (Model 1, levels):", round(rsquared_oos_scores_m1[i], 4),
              "| Train Size:", nrow(current_train_data_m1),
              "| Test Size:", nrow(current_test_data_m1), "\n"))
  }

  mean_cv_rmse_m1 <- mean(rmse_scores_m1, na.rm = TRUE)
  sd_cv_rmse_m1 <- sd(rmse_scores_m1, na.rm = TRUE)
  mean_cv_rsquared_oos_m1 <- mean(rsquared_oos_scores_m1, na.rm = TRUE)

  cat(paste("\nAverage Cross-Validated RMSE (Model 1, levels):", round(mean_cv_rmse_m1, 4), "\n"))
  cat(paste("Std Dev of Cross-Validated RMSE (Model 1, levels):", round(sd_cv_rmse_m1, 4), "\n"))
  cat(paste("Average Out-of-Sample R-squared (Model 1, levels):", round(mean_cv_rsquared_oos_m1, 4),

} else {
  cat("Fold size for Model 1 CV is 0. Increase k_folds or adjust initial_train_percent_m1.\n")
}
```

```
## Fold 1 RMSE (Model 1, levels): 10.4261 | OOS R-squared (Model 1, levels): -1.6195 | Train Size: 1515
## Fold 2 RMSE (Model 1, levels): 8.9033 | OOS R-squared (Model 1, levels): -0.8249 | Train Size: 1645
## Fold 3 RMSE (Model 1, levels): 13.1251 | OOS R-squared (Model 1, levels): -1.1375 | Train Size: 1775
## Fold 4 RMSE (Model 1, levels): 5.6604 | OOS R-squared (Model 1, levels): 0.3514 | Train Size: 1905 |
## Fold 5 RMSE (Model 1, levels): 8.903 | OOS R-squared (Model 1, levels): -2.5805 | Train Size: 2035 |
##
## Average Cross-Validated RMSE (Model 1, levels): 9.4036
## Std Dev of Cross-Validated RMSE (Model 1, levels): 2.7114
## Average Out-of-Sample R-squared (Model 1, levels): -1.1622
```

```r
cat("\n\n--- Model 2 (Log-Log): Rolling Origin k-Fold Cross-Validation ---\n")
```

```
##
##
## --- Model 2 (Log-Log): Rolling Origin k-Fold Cross-Validation ---
```

```r
data_for_cv_m2 <- final_model_data_log
k_folds_m2 <- 5
n_obs_m2 <- nrow(data_for_cv_m2)
initial_train_percent_m2 <- 0.70
initial_train_window_m2 <- floor(initial_train_percent_m2 * n_obs_m2)
remaining_obs_m2 <- n_obs_m2 - initial_train_window_m2
fold_size_m2 <- floor(remaining_obs_m2 / k_folds_m2)

rmse_scores_m2 <- numeric(k_folds_m2)
rsquared_oos_scores_m2 <- numeric(k_folds_m2)

model2_formula_cv <- log_SKEW ~ log_VIX_lag1 + log_RealizedVol_lag1 +
                               MarketReturn_lag1 + log_Sentiment_lag1 + log_PC_Ratio_lag1 +
                               logVIX_logSent_Interact_centered_lag1

if (fold_size_m2 > 0) {
  for (i in 1:k_folds_m2) {
    train_end_idx <- initial_train_window_m2 + (i - 1) * fold_size_m2
```

```
      test_start_idx <- train_end_idx + 1
      if (i < k_folds_m2) {
        test_end_idx <- train_end_idx + fold_size_m2
      } else {
        test_end_idx <- n_obs_m2
      }

      if (test_start_idx > n_obs_m2) {
        rmse_scores_m2[i] <- NA
        rsquared_oos_scores_m2[i] <- NA
        cat(paste("Fold", i, "(Model 2): No test data remaining. Skipping.\n"))
        next
      }

      current_train_data_m2 <- data_for_cv_m2[1:train_end_idx, ]
      current_test_data_m2  <- data_for_cv_m2[test_start_idx:test_end_idx, ]

      if (nrow(current_test_data_m2) == 0) {
          rmse_scores_m2[i] <- NA
          rsquared_oos_scores_m2[i] <- NA
          cat(paste("Fold", i, "(Model 2): Test data has 0 rows. Skipping.\n"))
          next
      }

      model_fit_cv_m2 <- lm(model2_formula_cv, data = current_train_data_m2)
      predictions_cv_m2 <- predict(model_fit_cv_m2, newdata = current_test_data_m2)
      actuals_cv_m2 <- current_test_data_m2$log_SKEW

      rmse_scores_m2[i] <- calculate_rmse(actuals_cv_m2, predictions_cv_m2)
      rsquared_oos_scores_m2[i] <- calculate_rsquared_oos(actuals_cv_m2, predictions_cv_m2)

      cat(paste("Fold", i, "RMSE (Model 2, log-scale):", round(rmse_scores_m2[i], 4),
                "| OOS R-squared (Model 2, log-scale):", round(rsquared_oos_scores_m2[i], 4),
                "| Train Size:", nrow(current_train_data_m2),
                "| Test Size:", nrow(current_test_data_m2), "\n"))
    }

    mean_cv_rmse_m2 <- mean(rmse_scores_m2, na.rm = TRUE)
    sd_cv_rmse_m2 <- sd(rmse_scores_m2, na.rm = TRUE)
    mean_cv_rsquared_oos_m2 <- mean(rsquared_oos_scores_m2, na.rm = TRUE)

    cat(paste("\nAverage Cross-Validated RMSE (Model 2, log-scale):", round(mean_cv_rmse_m2, 4), "\n"))
    cat(paste("Std Dev of Cross-Validated RMSE (Model 2, log-scale):", round(sd_cv_rmse_m2, 4), "\n"))
    cat(paste("Average Out-of-Sample R-squared (Model 2, log-scale):", round(mean_cv_rsquared_oos_m2, 4

  } else {
    cat("Fold size for Model 2 CV is 0. Increase k_folds or adjust initial_train_percent_m2.\n")
  }
```

```
## Fold 1 RMSE (Model 2, log-scale): 0.0755 | OOS R-squared (Model 2, log-scale): -1.5476 | Train Size:
## Fold 2 RMSE (Model 2, log-scale): 0.0673 | OOS R-squared (Model 2, log-scale): -0.8579 | Train Size:
## Fold 3 RMSE (Model 2, log-scale): 0.0965 | OOS R-squared (Model 2, log-scale): -1.1368 | Train Size:
## Fold 4 RMSE (Model 2, log-scale): 0.0448 | OOS R-squared (Model 2, log-scale): 0.3502 | Train Size:
```

```
## Fold 5 RMSE (Model 2, log-scale): 0.0732 | OOS R-squared (Model 2, log-scale): -2.4683 | Train Size:
##
## Average Cross-Validated RMSE (Model 2, log-scale): 0.0715
## Std Dev of Cross-Validated RMSE (Model 2, log-scale): 0.0185
## Average Out-of-Sample R-squared (Model 2, log-scale): -1.1321
```

```r
  cat("\n\n--- Model 3 (Parsimonious Log-Log): Rolling Origin k-Fold Cross-Validation ---\n")
```

```
##
##
## --- Model 3 (Parsimonious Log-Log): Rolling Origin k-Fold Cross-Validation ---
```

```r
data_for_cv_m3 <- final_model_data_log
k_folds_m3 <- 5
n_obs_m3 <- nrow(data_for_cv_m3)
initial_train_percent_m3 <- 0.70
initial_train_window_m3 <- floor(initial_train_percent_m3 * n_obs_m3)
remaining_obs_m3 <- n_obs_m3 - initial_train_window_m3
fold_size_m3 <- floor(remaining_obs_m3 / k_folds_m3)

rmse_scores_m3 <- numeric(k_folds_m3)
rsquared_oos_scores_m3 <- numeric(k_folds_m3)

model3_formula_cv <- log_SKEW ~ log_VIX_lag1 + log_RealizedVol_lag1 + log_PC_Ratio_lag1

if (fold_size_m3 > 0) {
  for (i in 1:k_folds_m3) {
    train_end_idx <- initial_train_window_m3 + (i - 1) * fold_size_m3
    test_start_idx <- train_end_idx + 1
    if (i < k_folds_m3) {
      test_end_idx <- train_end_idx + fold_size_m3
    } else {
      test_end_idx <- n_obs_m3
    }

    if (test_start_idx > n_obs_m3) {
      rmse_scores_m3[i] <- NA
      rsquared_oos_scores_m3[i] <- NA
      cat(paste("Fold", i, "(Model 3): No test data remaining. Skipping.\n"))
      next
    }

    current_train_data_m3 <- data_for_cv_m3[1:train_end_idx, ]
    current_test_data_m3  <- data_for_cv_m3[test_start_idx:test_end_idx, ]

    if (nrow(current_test_data_m3) == 0) {
        rmse_scores_m3[i] <- NA
        rsquared_oos_scores_m3[i] <- NA
        cat(paste("Fold", i, "(Model 3): Test data has 0 rows. Skipping.\n"))
        next
    }

    model_fit_cv_m3 <- lm(model3_formula_cv, data = current_train_data_m3)
```

```r
    predictions_cv_m3 <- predict(model_fit_cv_m3, newdata = current_test_data_m3)
    actuals_cv_m3 <- current_test_data_m3$log_SKEW

    rmse_scores_m3[i] <- calculate_rmse(actuals_cv_m3, predictions_cv_m3)
    rsquared_oos_scores_m3[i] <- calculate_rsquared_oos(actuals_cv_m3, predictions_cv_m3)

    cat(paste("Fold", i, "RMSE (Model 3, log-scale):", round(rmse_scores_m3[i], 4),
              "| OOS R-squared (Model 3, log-scale):", round(rsquared_oos_scores_m3[i], 4),
              "| Train Size:", nrow(current_train_data_m3),
              "| Test Size:", nrow(current_test_data_m3), "\n"))
  }

  mean_cv_rmse_m3 <- mean(rmse_scores_m3, na.rm = TRUE)
  sd_cv_rmse_m3 <- sd(rmse_scores_m3, na.rm = TRUE)
  mean_cv_rsquared_oos_m3 <- mean(rsquared_oos_scores_m3, na.rm = TRUE)

  cat(paste("\nAverage Cross-Validated RMSE (Model 3, log-scale):", round(mean_cv_rmse_m3, 4), "\n"))
  cat(paste("Std Dev of Cross-Validated RMSE (Model 3, log-scale):", round(sd_cv_rmse_m3, 4), "\n"))
  cat(paste("Average Out-of-Sample R-squared (Model 3, log-scale):", round(mean_cv_rsquared_oos_m3, 4

} else {
  cat("Fold size for Model 3 CV is 0. Increase k_folds or adjust initial_train_percent_m3.\n")
}
```

```
## Fold 1 RMSE (Model 3, log-scale): 0.0711 | OOS R-squared (Model 3, log-scale): -1.2558 | Train Size:
## Fold 2 RMSE (Model 3, log-scale): 0.066 | OOS R-squared (Model 3, log-scale): -0.7907 | Train Size:
## Fold 3 RMSE (Model 3, log-scale): 0.0966 | OOS R-squared (Model 3, log-scale): -1.1415 | Train Size:
## Fold 4 RMSE (Model 3, log-scale): 0.043 | OOS R-squared (Model 3, log-scale): 0.4018 | Train Size: 1
## Fold 5 RMSE (Model 3, log-scale): 0.0733 | OOS R-squared (Model 3, log-scale): -2.4761 | Train Size:
##
## Average Cross-Validated RMSE (Model 3, log-scale): 0.07
## Std Dev of Cross-Validated RMSE (Model 3, log-scale): 0.0191
## Average Out-of-Sample R-squared (Model 3, log-scale): -1.0524
```

```
##
## --- Model 2 (Log-Log): Sub-Period Robustness Analysis ---
```

```
##
## Sub-Period 1: Observations 1 to 1082 ( 1082 obs )
```

```
## Sub-Period 2: Observations 1083 to 2165 ( 1083 obs )
```

```
##
## -- Model 2 on Sub-Period 1 --
```

```
##
## Call:
## lm(formula = model2_formula_cv, data = sub_period1_data_m2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.129191 -0.031506 -0.002225  0.029560  0.126171
```

```
##
## Coefficients:
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           4.912728   0.033950 144.704  < 2e-16 ***
## log_VIX_lag1                         -0.028920   0.008383  -3.450 0.000583 ***
## log_RealizedVol_lag1                 -0.027683   0.005281  -5.242 1.91e-07 ***
## MarketReturn_lag1                    -0.084283   0.046034  -1.831 0.067394 .
## log_Sentiment_lag1                    0.070461   0.007543   9.341  < 2e-16 ***
## log_PC_Ratio_lag1                     0.005253   0.005923   0.887 0.375381
## logVIX_logSent_Interact_centered_lag1 -0.013666   0.026231  -0.521 0.602484
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.04355 on 1075 degrees of freedom
## Multiple R-squared:  0.2186, Adjusted R-squared:  0.2142
## F-statistic: 50.13 on 6 and 1075 DF,  p-value: < 2.2e-16

##
## Diagnostics for Model 2 - Sub-Period 1:

##
##  Durbin-Watson test
##
## data:  model2_sub1
## DW = 0.38653, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0

##
## Model 2 Sub-Period 1 - HAC Corrected Coefficients:

##
## t test of coefficients:
##
##                                        Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                           4.9127282  0.0677006 72.5655 < 2.2e-16
## log_VIX_lag1                         -0.0289196  0.0153007 -1.8901   0.05902
## log_RealizedVol_lag1                 -0.0276827  0.0109236 -2.5342   0.01141
## MarketReturn_lag1                    -0.0842826  0.0927562 -0.9086   0.36374
## log_Sentiment_lag1                    0.0704613  0.0135872  5.1859 2.568e-07
## log_PC_Ratio_lag1                     0.0052529  0.0071850  0.7311   0.46488
## logVIX_logSent_Interact_centered_lag1 -0.0136659  0.0427994 -0.3193   0.74956
##
## (Intercept)                          ***
## log_VIX_lag1                         .
## log_RealizedVol_lag1                 *
## MarketReturn_lag1
## log_Sentiment_lag1                   ***
## log_PC_Ratio_lag1
## logVIX_logSent_Interact_centered_lag1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
```

```
##
## -- Model 2 on Sub-Period 2 --

##
## Call:
## lm(formula = model2_formula_cv, data = sub_period2_data_m2)
##
## Residuals:
##       Min        1Q    Median        3Q       Max
## -0.153747 -0.046241 -0.006356  0.043721  0.200586
##
## Coefficients:
##                                       Estimate Std. Error t value Pr(>|t|)
## (Intercept)                           4.806434   0.049071  97.948  < 2e-16 ***
## log_VIX_lag1                         -0.006470   0.013773  -0.470  0.63861
## log_RealizedVol_lag1                 -0.047682   0.006434  -7.410 2.54e-13 ***
## MarketReturn_lag1                    -0.045889   0.073495  -0.624  0.53251
## log_Sentiment_lag1                    0.031396   0.009844   3.189  0.00147 **
## log_PC_Ratio_lag1                     0.013296   0.009349   1.422  0.15526
## logVIX_logSent_Interact_centered_lag1 -0.020020   0.031944  -0.627  0.53097
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 0.06243 on 1076 degrees of freedom
## Multiple R-squared:  0.1521, Adjusted R-squared:  0.1473
## F-statistic: 32.16 on 6 and 1076 DF,  p-value: < 2.2e-16

##
## Diagnostics for Model 2 - Sub-Period 2:

##
##  Durbin-Watson test
##
## data:  model2_sub2
## DW = 0.26333, p-value < 2.2e-16
## alternative hypothesis: true autocorrelation is greater than 0

##
## Model 2 Sub-Period 2 - HAC Corrected Coefficients:

##
## t test of coefficients:
##
##                                        Estimate Std. Error t value  Pr(>|t|)
## (Intercept)                           4.8064336  0.0986019 48.7458 < 2.2e-16
## log_VIX_lag1                         -0.0064703  0.0271462 -0.2384 0.8116536
## log_RealizedVol_lag1                 -0.0476819  0.0131905 -3.6149 0.0003144
## MarketReturn_lag1                    -0.0458893  0.1446394 -0.3173 0.7511024
## log_Sentiment_lag1                    0.0313961  0.0227598  1.3795 0.1680415
## log_PC_Ratio_lag1                     0.0132956  0.0128333  1.0360 0.3004263
## logVIX_logSent_Interact_centered_lag1 -0.0200201  0.0738700 -0.2710 0.7864297
##
## (Intercept)                           ***
```

```
## log_VIX_lag1
## log_RealizedVol_lag1                  ***
## MarketReturn_lag1
## log_Sentiment_lag1
## log_PC_Ratio_lag1
## logVIX_logSent_Interact_centered_lag1
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1

##
##
## --- Summary Comparison (HAC Corrected Estimates) ---

## Variable                              | Full Period (Model 2b) | Sub-Period 1          | Sub-Period 2

## --------------------------------------|------------------------|----------------------|-----------------

## (Intercept)                           | Est:   4.8966 (p: 0.000) | Est:   4.9127 (p: 0.000) | Est:   4.8
## log_VIX_lag1                          | Est:  -0.0429 (p: 0.006) | Est:  -0.0289 (p: 0.059) | Est:  -0.0
## log_RealizedVol_lag1                  | Est:  -0.0375 (p: 0.000) | Est:  -0.0277 (p: 0.011) | Est:  -0.0
## MarketReturn_lag1                     | Est:  -0.0935 (p: 0.321) | Est:  -0.0843 (p: 0.364) | Est:  -0.0
## log_Sentiment_lag1                    | Est:   0.0199 (p: 0.166) | Est:   0.0705 (p: 0.000) | Est:   0.0
## log_PC_Ratio_lag1                     | Est:   0.0196 (p: 0.015) | Est:   0.0053 (p: 0.465) | Est:   0.0
## logVIX_logSent_Interact_centered_lag1 | Est:  -0.0436 (p: 0.379) | Est:  -0.0137 (p: 0.750) | Est:

##
## Adj. R-sq (Full OLS Model 2a): 0.177

##
## Adj. R-sq (Sub-Period 1 OLS): 0.214

##
## Adj. R-sq (Sub-Period 2 OLS): 0.147
```