Baseball

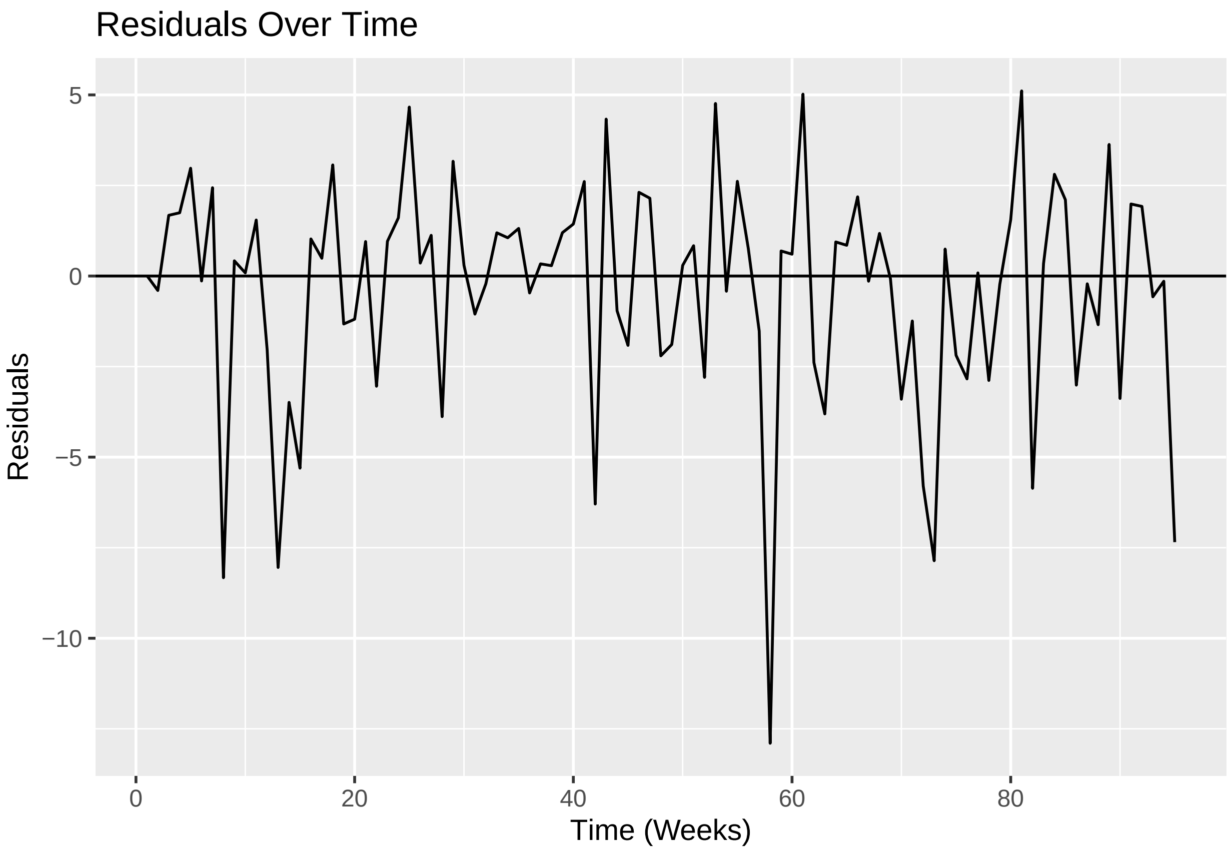
2025-04-11

```r
# Read in the data we processed/created in Python from the pybaseball library
weekly_data <- read.csv("weekly_data_for_r.csv")
```

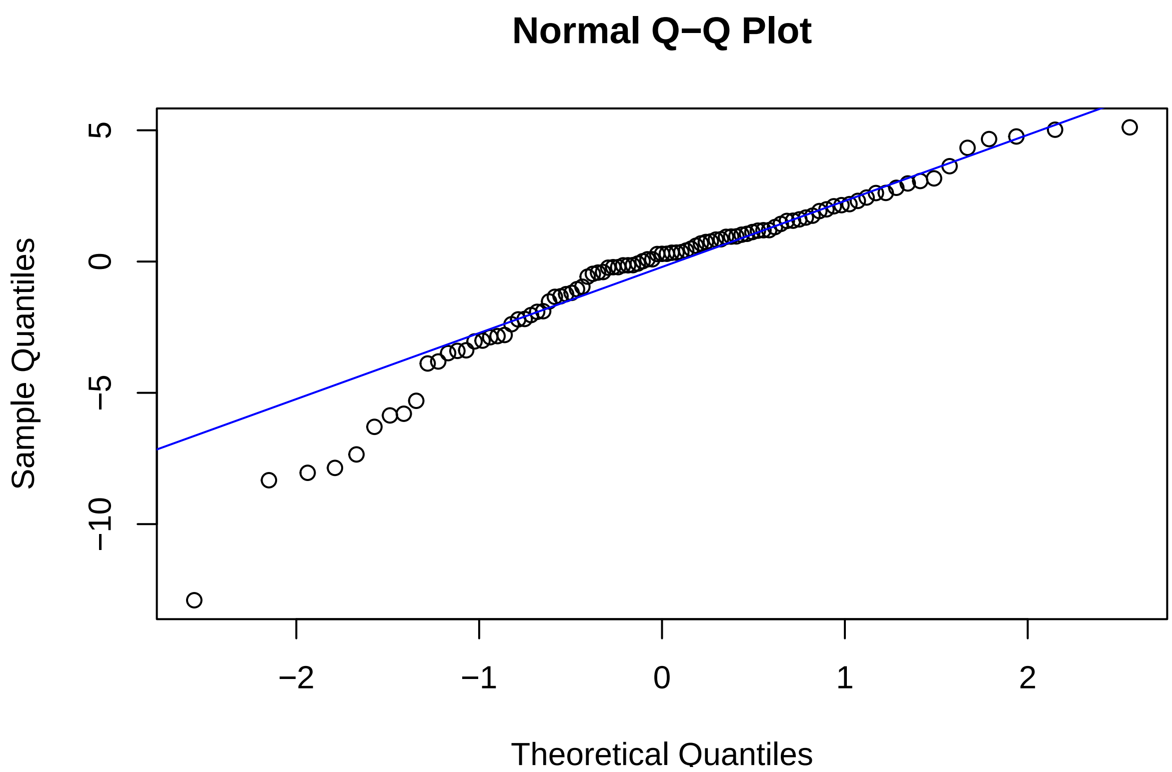
```
## Registered S3 method overwritten by 'quantmod':
##   method            from
##   as.zoo.data.frame zoo
```

```
## Baseline ARIMA(1,1,1) AIC: 491.1741
```

```r
autoplot(residuals(model_baseline)) +
  ggtitle("Residuals Over Time") +
  xlab("Time (Weeks)") + ylab("Residuals") +
  geom_hline(yintercept = 0)
```




```r
# Q-Q plot
qqnorm(residuals(model_baseline))
qqline(residuals(model_baseline), col = "blue")
```




```r
# With exog featueres
exog_vars <- c("avg_velocity", "avg_release_pos_x", "avg_spin_rate",
               "avg_pitch_number", "avg_release_extension", "rest_days", "zone_rate")

df_all <- na.omit(weekly_data[, c("K_per_9", exog_vars)])
y_all <- df_all$K_per_9
X_all <- as.matrix(df_all[, exog_vars])

model_exog <- Arima(y_all, order = c(1, 1, 1), xreg = X_all)
cat("SARIMAX AIC:", AIC(model_exog), "\n")
```

```
## SARIMAX AIC: 498.4549
```

```r
summary(model_exog)
```

```
## Series: y_all
## Regression with ARIMA(1,1,1) errors
##
## Coefficients:
##          ar1      ma1  avg_velocity  avg_release_pos_x  avg_spin_rate
##       0.0565  -1.0000        0.1796            -2.7910         0.0019
## s.e.  0.1148   0.0347        0.3952             3.0098         0.0054
##       avg_pitch_number  avg_release_extension  rest_days  zone_rate
##                 0.0543                -0.7006    -0.1819    -9.3879
## s.e.            1.5177                 3.3222     0.1079     8.0650
##
## sigma^2 = 10.03:  log likelihood = -239.23
## AIC=498.45   AICc=501.11   BIC=523.89
##
## Training set error measures:
##                          ME      RMSE       MAE   MPE MAPE      MASE         ACF1
```

```
## Training set -0.07209907 2.995522 2.187366 -Inf  Inf 0.6944949 -0.009612914
```

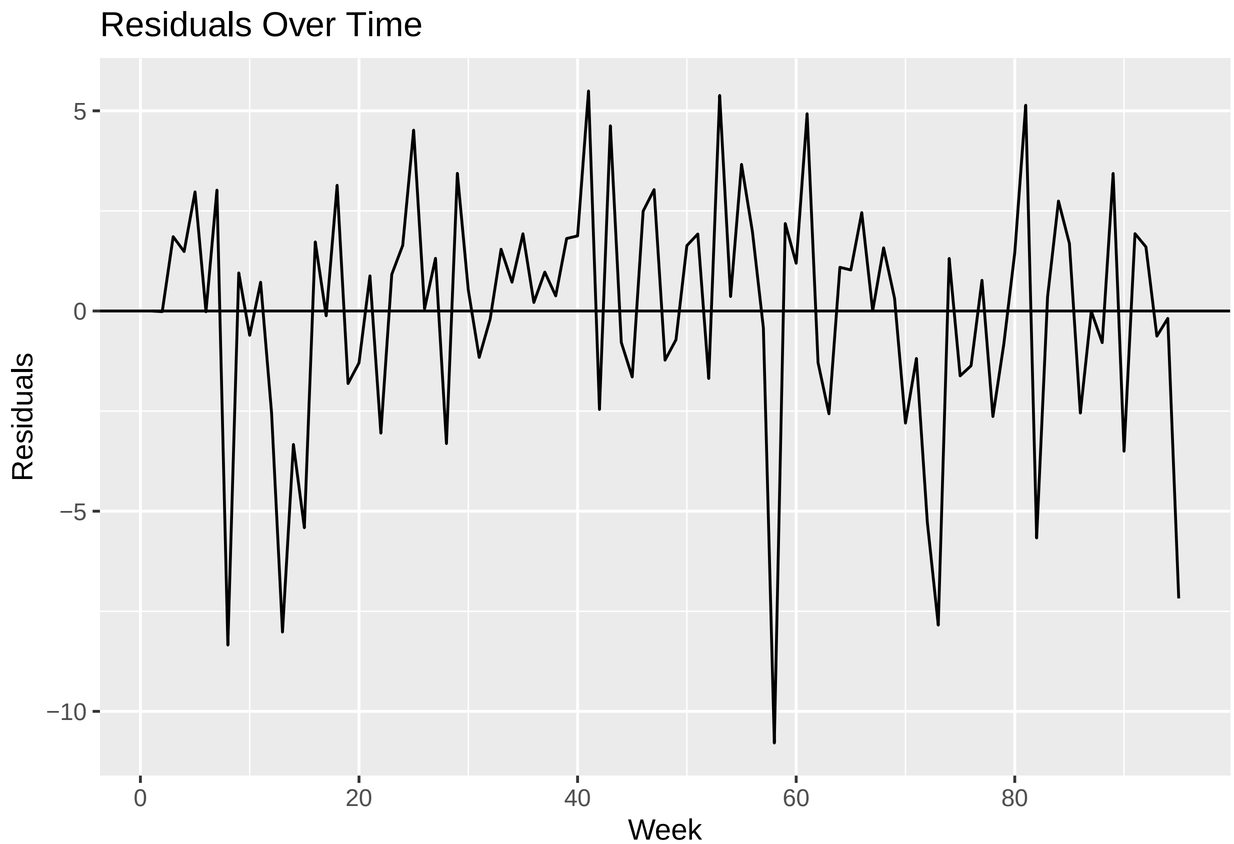
```r
# With the promising features

exog_specific <- c("avg_release_pos_x", "rest_days")

df_sel <- na.omit(weekly_data[, c("K_per_9", exog_specific)])
y_sel <- df_sel$K_per_9
X_sel <- as.matrix(df_sel[, exog_specific])

model_sel <- Arima(y_sel, order = c(1, 1, 1), xreg = X_sel)
cat("SARIMAX AIC (release_pos_x + rest_days):", AIC(model_sel), "\n")
```

```
## SARIMAX AIC (release_pos_x + rest_days): 490.591
```

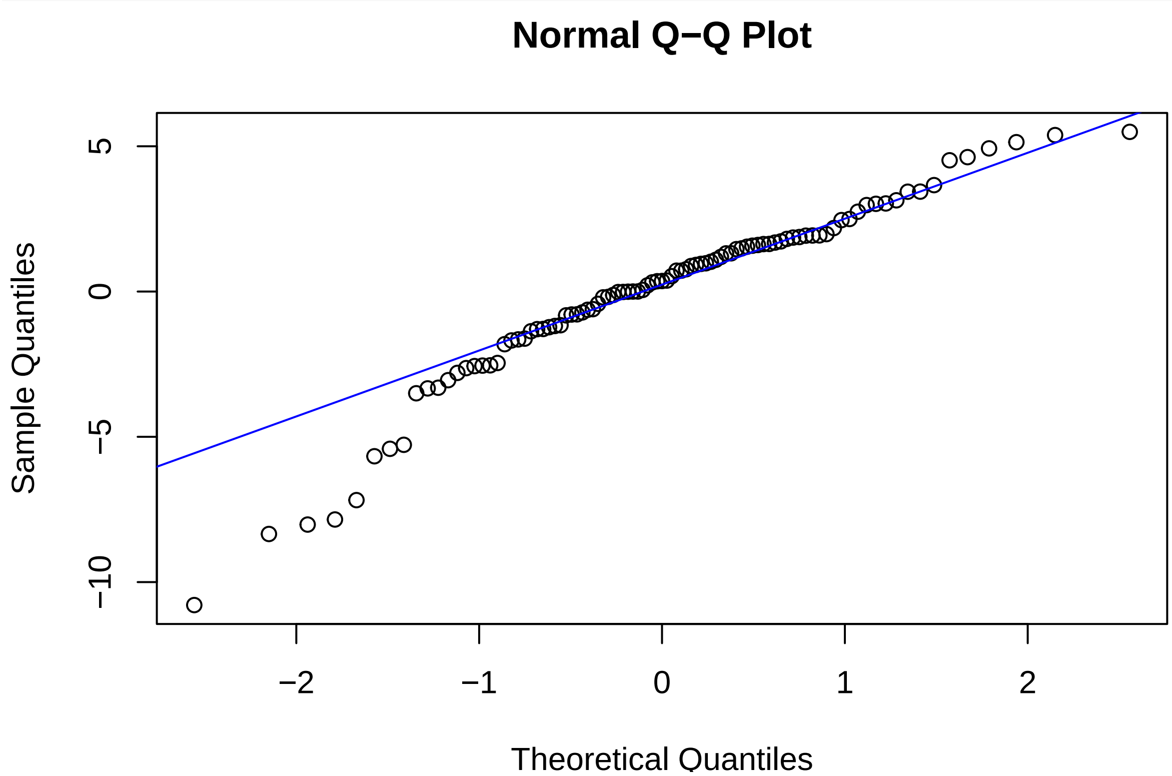
```r
summary(model_sel)
```

```
## Series: y_sel
## Regression with ARIMA(1,1,1) errors
##
## Coefficients:
##          ar1      ma1  avg_release_pos_x  rest_days
##       0.0951  -1.0000            -3.7181    -0.1846
## s.e.  0.1084   0.0465             1.5782     0.1108
##
## sigma^2 = 9.697:  log likelihood = -240.3
## AIC=490.59   AICc=491.27   BIC=503.31
##
## Training set error measures:
##                       ME     RMSE      MAE  MPE MAPE      MASE        ACF1
## Training set -0.02621395 3.030981 2.224133 -Inf  Inf 0.7061685 -0.01500854
```

```r
autoplot(residuals(model_sel)) +
  ggtitle("Residuals Over Time") +
  xlab("Week") + ylab("Residuals") +
  geom_hline(yintercept = 0)
```




```
# Q-Q plot
qqnorm(residuals(model_sel))
qqline(residuals(model_sel), col = "blue")
```



```r
pred <- fitted(model_sel)

comparison_df <- data.frame(
  Actual_K_per_9 = round(y_sel, 2),
  Predicted_K_per_9 = round(pred, 2)
)

print(head(comparison_df, 10))
```

```
##    Actual_K_per_9 Predicted_K_per_9
## 1           12.60             12.60
## 2           12.06             12.08
## 3           14.29             12.44
## 4           15.00             13.51
## 5           16.78             13.81
## 6           14.14             14.16
## 7           16.71             13.70
## 8            6.00             14.34
## 9           13.50             12.55
## 10          13.50             14.11
```

```r
sd(y_sel)
```

```
## [1] 3.194137
```

```r
sd(pred)
```

```
## [1] 1.024033
```

```r
library("Metrics")
```

```
##
## Attaching package: 'Metrics'

## The following object is masked from 'package:forecast':
##
##     accuracy
```

```r
exog_vars <- c("avg_release_pos_x", "rest_days")

df <- na.omit(weekly_data[, c("K_per_9", exog_vars)])

n <- nrow(df)
train_size <- floor(0.8 * n)
train <- df[1:train_size, ]
test <- df[(train_size + 1):n, ]

y_train <- train$K_per_9
xreg_train <- as.matrix(train[, exog_vars])

xreg_test <- as.matrix(test[, exog_vars])
y_test <- test$K_per_9

model <- Arima(y_train, order = c(1, 1, 1), xreg = xreg_train)

forecast_test <- forecast(model, xreg = xreg_test, h = nrow(test))
```

```r
cat("Train AIC:", AIC(model), "\n")
```

```
## Train AIC: 394.4264
```

```r
pred_test <- forecast_test$mean
rmse_value <- rmse(y_test, pred_test)
mae_value <- mae(y_test, pred_test)

cat("Test RMSE:", round(rmse_value, 2), "\n")
```

```
## Test RMSE: 2.98
```

```r
cat("Test MAE:", round(mae_value, 2), "\n")
```

```
## Test MAE: 2.3
```

```r
comparison <- data.frame(
  Actual_K_per_9 = round(y_test, 2),
  Predicted_K_per_9 = round(pred_test, 2)
)
print(head(comparison, 10))
```

```
##    Actual_K_per_9 Predicted_K_per_9
## 77          10.80             10.11
## 78           7.94             10.84
## 79          10.29             11.64
## 80          12.15             11.17
## 81          15.88             11.00
## 82           5.40             10.97
## 83          10.80             11.30
## 84          13.50             11.19
## 85          13.09             11.45
## 86           8.10             10.82
```