Skewness

```r
data.file <- file.path(data.dir, "Reproducible Finance.csv")

symbols <- c("SPY", "EFA", "IJS", "EEM", "AGG")

# Yahoo! Finance

prices <- getSymbols(symbols,
             from = "2012-12-31",
             to = "2017-12-31",
             auto.assign = T,
             warnings = F) %>%
          map(~Ad(get(.))) %>%
  reduce(merge) %>%
  `colnames<-`(symbols)
```

```
'getSymbols' currently uses auto.assign=TRUE by default, but will
use auto.assign=FALSE in 0.5-0. You will still be able to use
'loadSymbols' to automatically load data. getOption("getSymbols.env")
and getOption("getSymbols.auto.assign") will still be checked for
alternate defaults.

This message is shown once per session and may be disabled by setting
options("getSymbols.warning4.0"=FALSE). See ?getSymbols for details.
```

```r
# Verfiy Import
head(prices, 3)
```

```
                SPY      EFA      IJS      EEM      AGG
2012-12-31 124.0723 46.16656 72.91736 37.94026 92.87048
2013-01-02 127.2522 46.88106 74.91806 38.68453 92.76180
2013-01-03 126.9648 46.42639 74.81892 38.41077 92.52770
```

```r
# convert to monthly prices.

prices_monthly <- to.monthly(prices,
                             indexAt = "lastof",
                             OHLC = F)

head(prices_monthly)
```

```
                SPY      EFA      IJS      EEM      AGG
2012-12-31 124.0723 46.16656 72.91736 37.94026 92.87048
2013-01-31 130.4236 47.88786 76.81961 37.82905 92.29356
2013-02-28 132.0876 47.27079 78.07232 36.96502 92.83891
```

```
2013-03-31 137.1031 47.88786 81.27948 36.58861 92.93040
2013-04-30 139.7372 50.29119 81.37890 37.03346 93.83049
2013-05-31 143.0363 48.77287 84.86759 35.24553 91.95281
```

```r
# Convert to monthly returns, dplyr.

asset_returns_dplyr_byhand <-
  prices %>%
  to.monthly(indexAt = "lastof", OHLC = F) %>%
  # convert the index to a date
  data.frame(date = index(.)) %>%
  # now remove the index because it got converted to row names
  remove_rownames() %>%
  gather(asset, prices, - date) %>%
  group_by(asset) %>%
  mutate(returns = (log(prices) - log(lag(prices)))) %>%
  select(-prices) %>%
  spread(asset, returns) %>%
  select(date, symbols)

head(asset_returns_dplyr_byhand)
```

```
# A tibble: 6 x 6
  date           SPY     EFA      IJS      EEM       AGG
  <date>       <dbl>   <dbl>    <dbl>    <dbl>     <dbl>
1 2012-12-31 NA      NA      NA       NA       NA
2 2013-01-31  0.0499  0.0366  0.0521  -0.00294 -0.00623
3 2013-02-28  0.0127 -0.0130  0.0162  -0.0231   0.00589
4 2013-03-31  0.0373  0.0130  0.0403  -0.0102   0.000985
5 2013-04-30  0.0190  0.0490  0.00122  0.0121   0.00964
6 2013-05-31  0.0233 -0.0307  0.0420  -0.0495  -0.0202
```

```r
asset_returns_dplyr_byhand <- asset_returns_dplyr_byhand %>%
  na.omit()


# convert from wide to long format.

asset_returns_long <-
  asset_returns_dplyr_byhand %>%
  gather(asset, returns, - date) %>%
  group_by(asset)

head(asset_returns_long)
```

```
# A tibble: 6 x 3
# Groups:   asset [1]
  date       asset returns
  <date>     <chr>   <dbl>
1 2013-01-31 SPY    0.0499
2 2013-02-28 SPY    0.0127
3 2013-03-31 SPY    0.0373
4 2013-04-30 SPY    0.0190
5 2013-05-31 SPY    0.0233
6 2013-06-30 SPY   -0.0134
```

```
# Convert to monthly returns, xts.
asset_returns_xts <-
  Return.calculate(prices_monthly,
                   method = "log") %>%
  na.omit()

head(asset_returns_xts, 3)
```

```
                  SPY         EFA        IJS          EEM           AGG
2013-01-31 0.04992319  0.03660626 0.05213324 -0.002935492 -0.0062314680
2013-02-28 0.01267820 -0.01296945 0.01617560 -0.023105348  0.0058915498
2013-03-31 0.03726787  0.01296945 0.04025802 -0.010234798  0.0009850067
```

```
# Asset Weights

w <- c(0.25,
       0.25,
       0.20,
       0.20,
       0.10)

stopifnot(sum(w) == 1)

tibble(w, symbols) %>%
  summarise(total_weights = sum(w))
```

```
# A tibble: 1 x 1
  total_weights
          <dbl>
1             1
```

```
# Portfolio returns, xts.

portfolio_returns_xts_rebalanced_monthly <-
  Return.portfolio(asset_returns_xts,
                   weights = w,
                   rebalance_on = "months") %>%
  `colnames<-`("returns")

head(portfolio_returns_xts_rebalanced_monthly)
```

```
                 returns
2013-01-31  0.0308487648
2013-02-28 -0.0008696067
2013-03-31  0.0186624756
2013-04-30  0.0206247389
2013-05-31 -0.0053528650
2013-06-30 -0.0229488564
```

```
# Portfolio returns, tidyquant.

portfolio_returns_tq_rebalanced_monthly <-
  asset_returns_long %>%
  tq_portfolio(assets_col = asset,
               returns_col = returns,
               weights = w,
               col_rename = "returns",
               rebalance_on = "months")

head(portfolio_returns_tq_rebalanced_monthly, 3)
```

```
# A tibble: 3 x 2
  date          returns
  <date>          <dbl>
1 2013-01-31  0.0308
2 2013-02-28 -0.000870
3 2013-03-31  0.0187
```

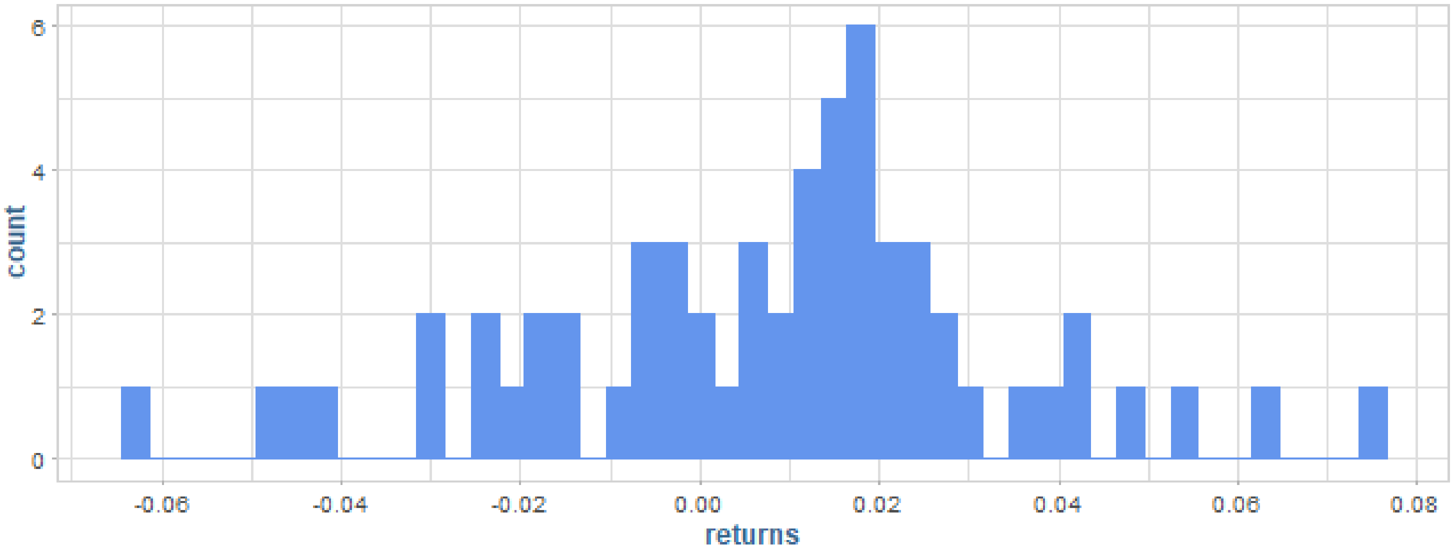
```
skew_xts <- skewness(portfolio_returns_xts_rebalanced_monthly$returns)

skew_xts
```

```
[1] -0.2296705
```

```
skew_tidy <- portfolio_returns_tq_rebalanced_monthly %>%
  summarize(skew_builtin = skewness(returns),
            skew_byhand =
              (sum((returns - mean(returns))^3)/length(returns))/
              ((sum((returns - mean(returns))^2)/length(returns)))^(3/2)) %>%
  select(skew_builtin, skew_byhand)

skew_tidy %>%
  mutate(xts = coredata(skew_xts)) %>%
  mutate_all(~ round(., 3))
```

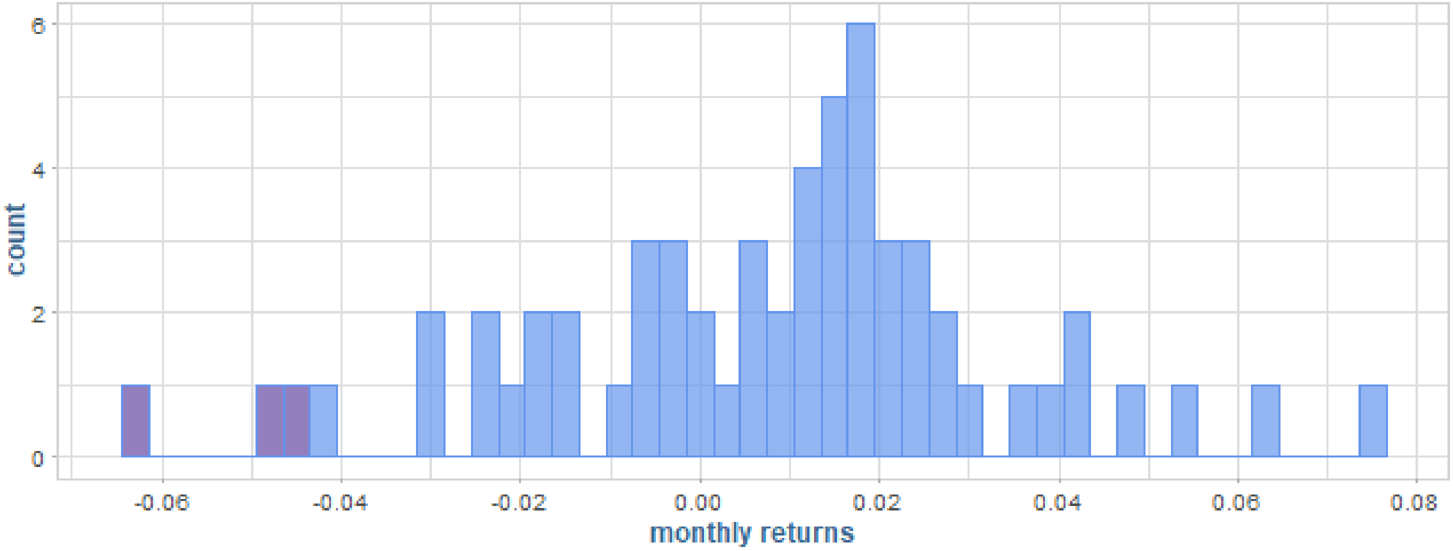
```
# A tibble: 1 x 3
  skew_builtin skew_byhand   xts
         <dbl>       <dbl> <dbl>
1        -0.23       -0.23 -0.23
```

```
portfolio_returns_tq_rebalanced_monthly %>%
  ggplot(aes(x = returns)) +
    geom_histogram(alpha = 7,
                   binwidth = .003,
                   fill = "cornflowerblue",
                   color = "cornflowerblue") +
  scale_x_continuous(breaks = pretty_breaks(n = 10))
```

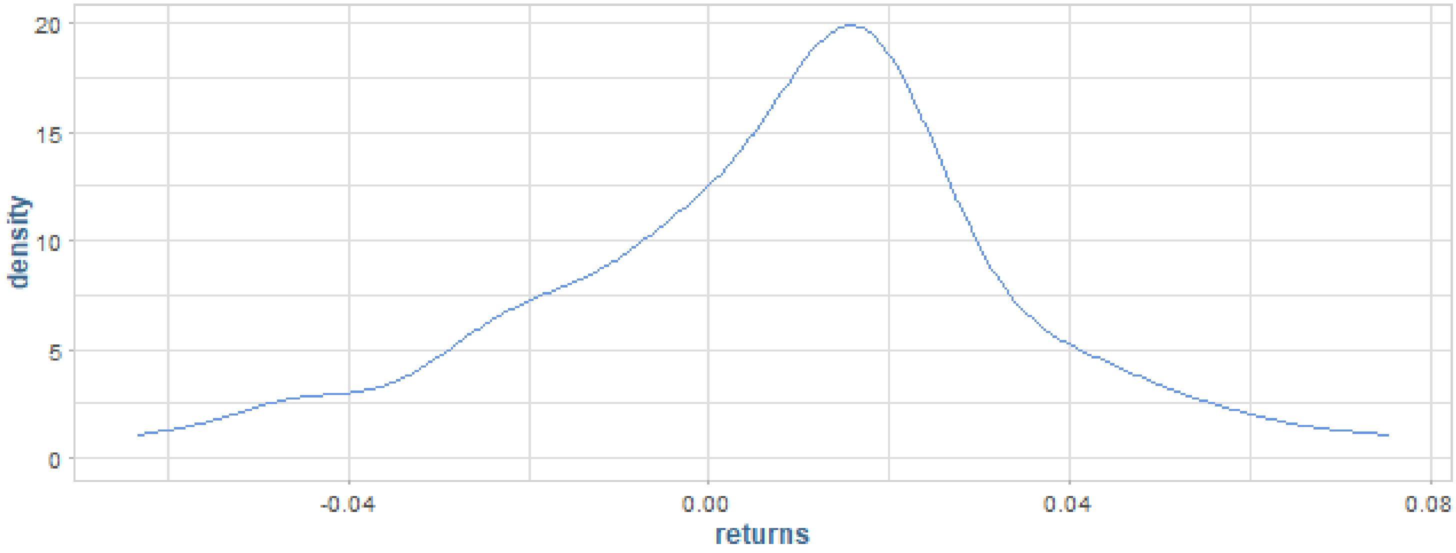


```
portfolio_returns_tq_rebalanced_monthly %>%
  mutate(hist_col_red =
           if_else(returns < (mean(returns) - 2*sd(returns)),
                   returns, as.numeric(NA)),
         if_else(returns > (mean(returns) - 2*sd(returns)),
                 returns, as.numeric(NA))) %>%
  ggplot() +
  geom_histogram(aes(x = hist_col_red),
                 alpha = .7,
                 binwidth = 0.003,
                 fill = "red",
                 color = "red") +
  geom_histogram(aes(x = returns),
                 alpha = 0.7,
                 binwidth = 0.003,
                 fill = "cornflowerblue",
                 color = "cornflowerblue") +
  scale_x_continuous(breaks = pretty_breaks(n = 10)) +
  xlab("monthly returns")
```

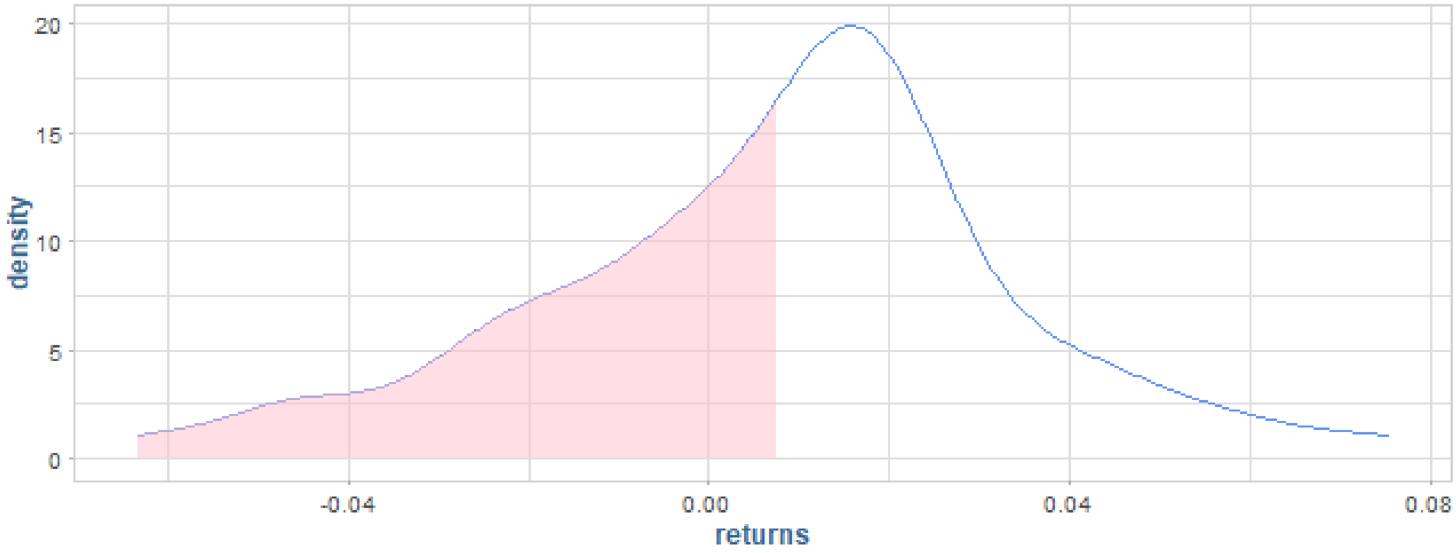
```
Warning: Removed 57 rows containing non-finite values (stat_bin).
```



```
portfolio_density_plot <-
  portfolio_returns_tq_rebalanced_monthly %>%
  ggplot(aes(x = returns)) +
  stat_density(geom = "line",
               alpha = 1,
               color = "cornflowerblue")

portfolio_density_plot
```

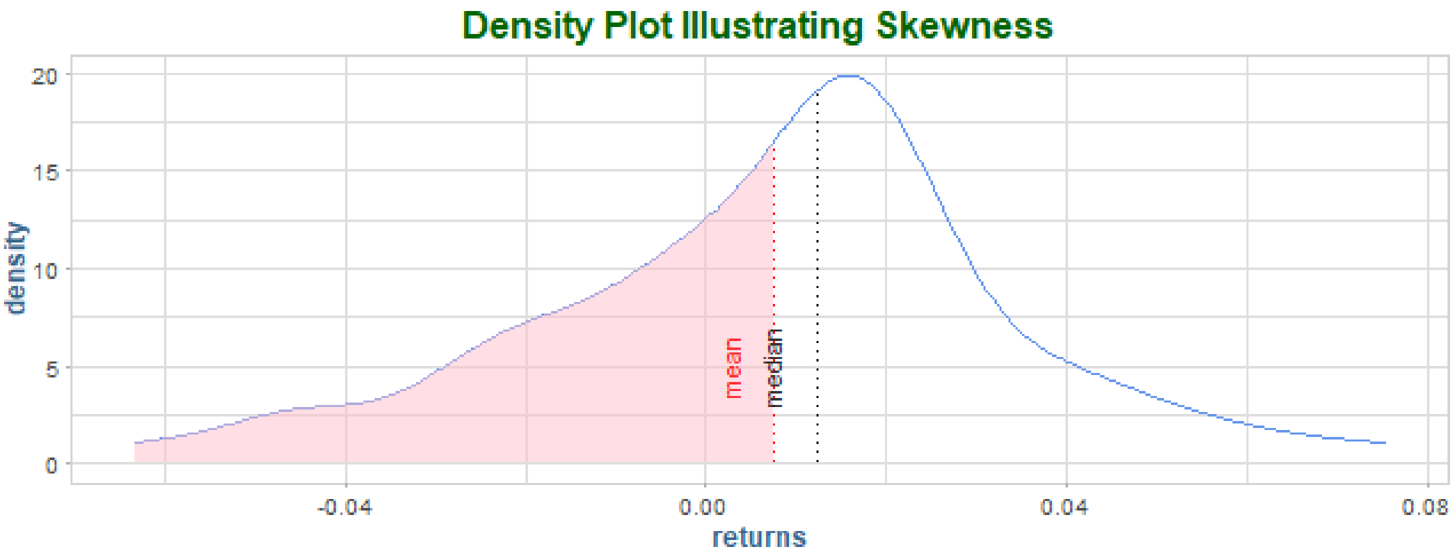


```
shaded_area_data <-
  ggplot_build(portfolio_density_plot)$data[[1]] %>%
  filter(x < mean(portfolio_returns_tq_rebalanced_monthly$returns))

portfolio_density_plot_shaded <-
  portfolio_density_plot +
  geom_area(data = shaded_area_data,
            aes(x = x, y = y),
            fill = "pink",
            alpha = 0.5)

portfolio_density_plot_shaded
```

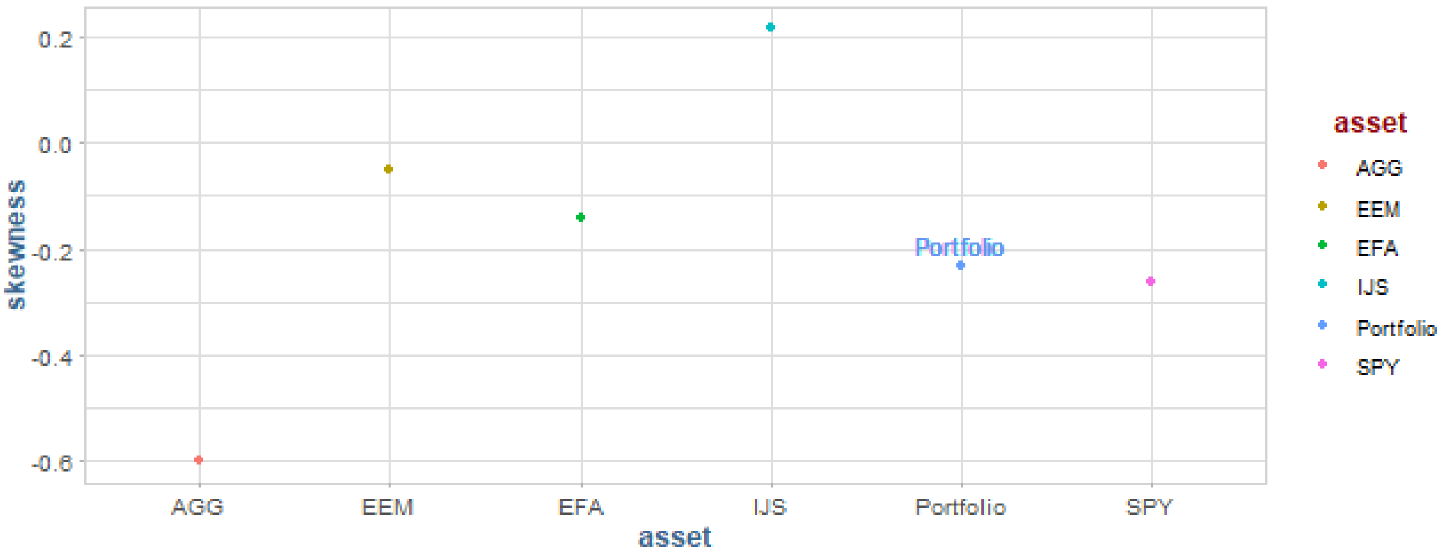


```
median <- median(portfolio_returns_tq_rebalanced_monthly$returns)

mean <- mean(portfolio_returns_tq_rebalanced_monthly$returns)
```

```
median_line_data <-
  ggplot_build(portfolio_density_plot)$data[[1]] %>%
  filter(x <= median)

portfolio_density_plot_shaded +
  geom_segment(data = shaded_area_data,
               aes(x = mean, y = 0,
                   xend = mean,
                   yend = density),
               color = "red",
               linetype = "dotted") +
  annotate(geom = "text",
           x = mean,
           y = 5,
           label = "mean",
           color = "red",
           fontface = "plain",
           angle = 90,
           alpha = 0.8,
           vjust = -1.75) +
  geom_segment(data = median_line_data,
               aes(x = median,
                   y = 0,
                   xend = median,
                   yend = density),
               color = "black",
               linetype = "dotted") +
  annotate(geom = "text",
           x = median,
           y = 5,
           label = "median",
           color = "black",
           fontface = "plain",
           angle = 90,
           alpha = 0.8,
           vjust = -1.75) +
  ggtitle("Density Plot Illustrating Skewness")
```



```
asset_returns_long %>%
  summarize(skew_assets = skewness(returns)) %>%
  add_row(asset = "Portfolio",
          skew_assets = skew_tidy$skew_byhand) %>%
  ggplot(aes(x = asset,
             y = skew_assets,
             color = asset)) +
  geom_point() +
  geom_text(
    aes(x = "Portfolio",
        y =
          skew_tidy$skew_builtin + 0.04),
        label = "Portfolio",
    color = "cornflowerblue") +
  labs(y = "skewness")
```



```
window <- 24

rolling_skew_xts <-
  rollapply(portfolio_returns_xts_rebalanced_monthly,
            FUN = skewness,
            width = window) %>%
  na.omit()
```

```
skew_roll_24 <-
  rollify(skewness, window = window)

roll_skew_tibbletime <-
  portfolio_returns_tq_rebalanced_monthly %>%
  as_tbl_time(index = date) %>%
  mutate(skew = skew_roll_24(returns)) %>%
  select(-returns) %>%
  na.omit()
```

```
rolling_skew_tq <-
  portfolio_returns_tq_rebalanced_monthly %>%
  tq_mutate(select = returns,
            mutate_fun = rollapply,
            width = window,
            FUN = skewness,
            col_rename = 'tq') %>%
  na.omit()
```

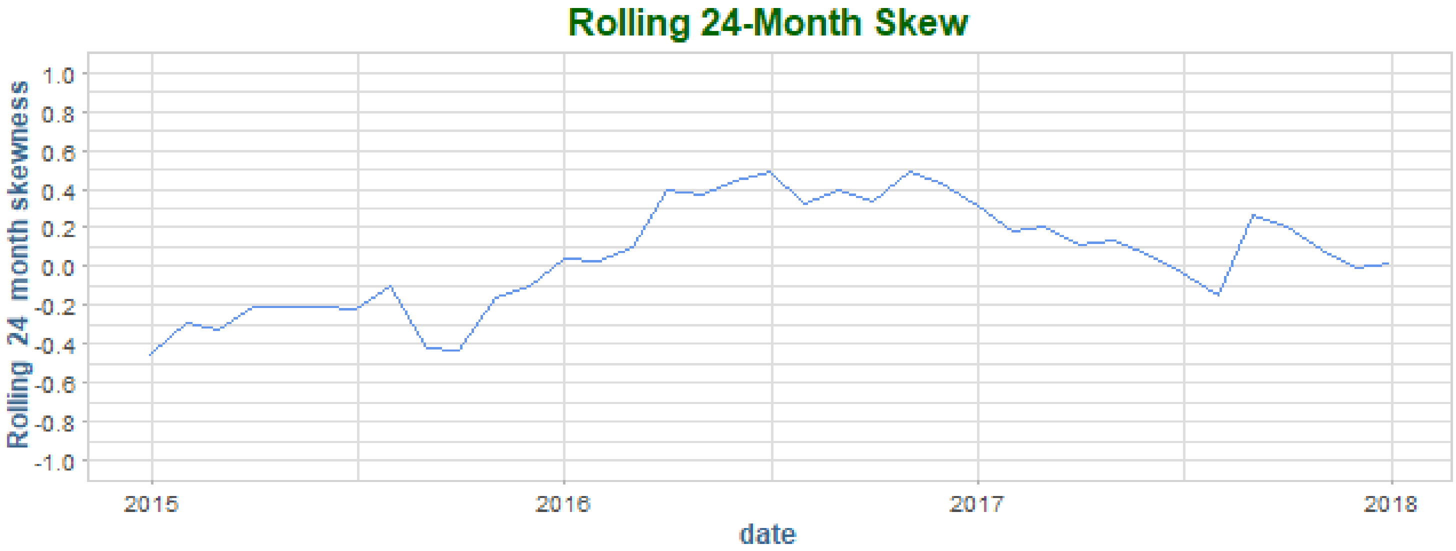
```
rolling_skew_tq %>%
  select(-returns) %>%
  mutate(xts = coredata(rolling_skew_xts),
         tbltime = roll_skew_tibbletime$skew) %>%
  mutate_if(is.numeric, ~ round(., 3)) %>%
  tail(3)
```

```
# A tibble: 3 x 4
  date           tq xts[,"returns"] tbltime
  <date>      <dbl>           <dbl>   <dbl>
1 2017-10-31  0.085           0.085   0.085
2 2017-11-30 -0.002          -0.002  -0.002
3 2017-12-31  0.018           0.018   0.018
```

```
highchart(type = "stock") %>%
  hc_title(text = "Rolling 24-Month Skewness") %>%
  hc_add_series(rolling_skew_xts,
              name = "Rolling Skewness",
              color = "cornflowerblue") %>%
  hc_yAxis(title = list(text = "skewness"),
           opposite = F,
           max = 1,
           min = -1) %>%
  hc_navigator(enabled = F) %>%
  hc_scrollbar(enabled = F) %>%
  hc_add_theme(hc_theme_flat()) %>%
  hc_exporting(enabled = F)
```

```
PhantomJS not found. You can install it with webshot::install_phantomjs(). If it is installed,
```

```
rolling_skew_tq %>%
  ggplot(aes(x = date, y = tq)) +
  geom_line(color = "cornflowerblue") +
  ggtitle("Rolling 24-Month Skew") +
  ylab(paste("Rolling ", window, " month skewness", sep = " ")) +
  scale_y_continuous(limits = c(-1, 1),
                     breaks = pretty_breaks(n = 8))
```


Rolling 24-Month Skew
Rolling 24 month skewness
1.0
0.8
0.6
0.4
0.2
0.0
-0.2
-0.4
-0.6
-0.8
-1.0
2015
2016
2017
2018
date