

Analysis of NSD3 Isoform Expression from TCGA-LUSC Data

David Dilworth and Dalia Barsyte-Lovejoy

2018-08-14

Objective

There are two major NSD3 isoforms expressed, long (aa 1-1437) and short (aa 1-645, differing in sequence from 620-645). Importantly, the short isoform was shown to be required for the maintenance of acute myeloid leukemia (AML) [1]. This isoform lacks a SET domain and thus methyltransferase activity. Additionally, my preliminary experiments suggest it is the short isoform that mediates NSD3 involvement in observed EMT phenotypes. It is not clear if the two isoforms are simply co-expressed or independently regulated. As a first attempt to study the differential regulation of the two isoforms, I have used the TCGA-LUSC data set to analyze relative expression levels in the context of squamous cell lung cancer (LUSC) [2].

Code & Results

This analysis will rely on three R packages; tidyverse to clean, tidy, and plot the data; ggbeeswarm to extend the functionality of ggplot to include beeswarm plots; and ggfortify for auto-plotting linear regression models. These packages are first loaded below.

```
library(tidyverse)
library(ggbeeswarm)
library(ggfortify)
```

To analyze the isoform data downloaded from Firebrowse (<http://firebrowse.org/>), we need to also download and load the correct NSD3 UCSC isoform ids used for expression analysis by Firebrowse, as they likely differ from the most current. I was able to download the correct annotations from NIH NCI Genomic Data Commons (<https://gdc.cancer.gov/about-data/data-harmonization-and-generation/gdc-reference-files>).

For reference, I've included a schematic representation of NSD3 isoforms here.

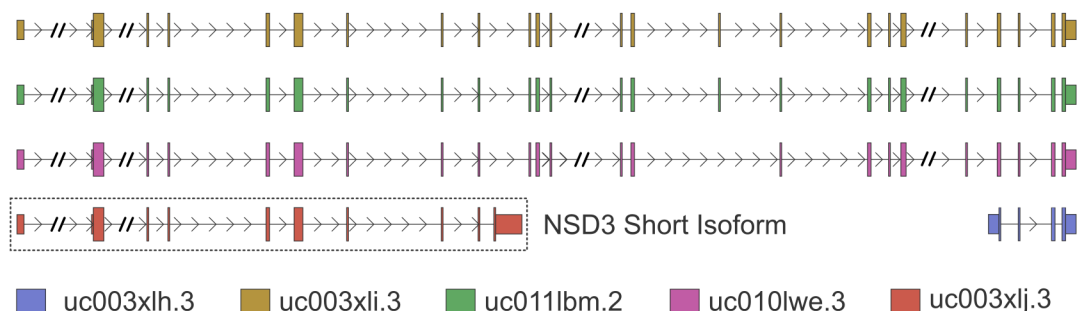


Figure 1: Schematic Representation of NSD3 Isoforms. Intron lengths have been truncated where indicated with a double-slash.

```
# The following code will filter the annotation file and pull all UCSC isoform ids
# linked to NSD3 (WHSC1L1).
```

```
NSD3.isoforms <-
  read_tsv(file = "TCGA.hg19.June2011.gaf") %>%
  filter(FeatureDBSource == "UCSCgene" &
         str_detect(Gene, "WHSC1L1") &
         CompositeType == "gene") %>%
  pull(FeatureID)
```

Next, normalized isoform expression data is loaded into R and filtered by NSD3 isoform ids identified in the previous step. I've downloaded this data from the Firebrowse website. File name: illuminahiseq_rnaseqv2-RSEM_isoforms_normalized (MD5) (http://gdac.broadinstitute.org/runs/stddata__2016_01_28/data/LUSC/20160128/gdac.broadinstitute.org_LUSC.Merge_rnaseqv2__illuminahiseq_rnaseqv2__unc_edu__Level_3__RSEM_isoforms_normalized__data.Level_3.2016012800.0.0.tar.gz)

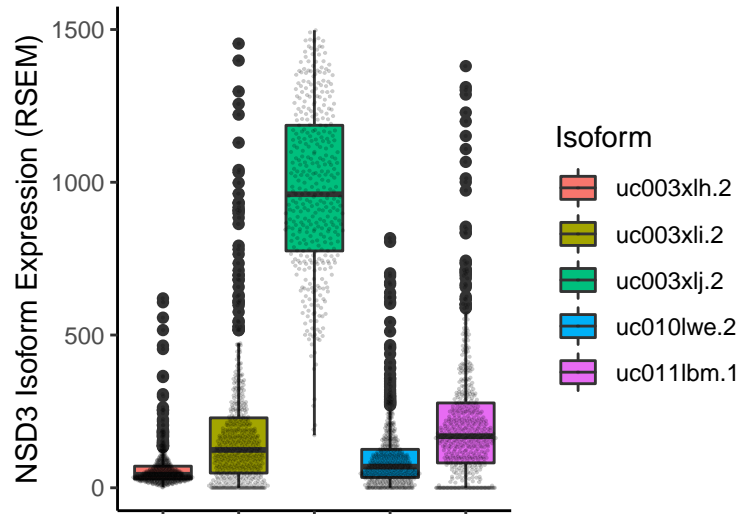
```
# Read in isoform expression data, filter for NSD3 isoforms, and prepare for plotting
# with ggplot.
```

```
iso <- read_tsv("LUSC.isoform.txt") %>%
  filter(`Hybridization REF` %in% NSD3.isoforms) %>%
  gather(Patient, RSEM, 2:553) %>%
  mutate(Isoform = `Hybridization REF`) %>%
  select(Isoform, Patient, RSEM) %>%
  mutate(RSEM = as.numeric(RSEM))
```

The first plot we will look at is a box and point plot of normalized RSEM values for each NSD3 isoform. Because of the large variance in the data, I also plotted this data with a log transformed y-axis.

```
# Boxplot of isoform expression from TCGA-LUSC. I've cut off the y-axis at 1500 RSEM
# for easier visualization, removing highly expressed NSD3 short outliers (n=174).
```

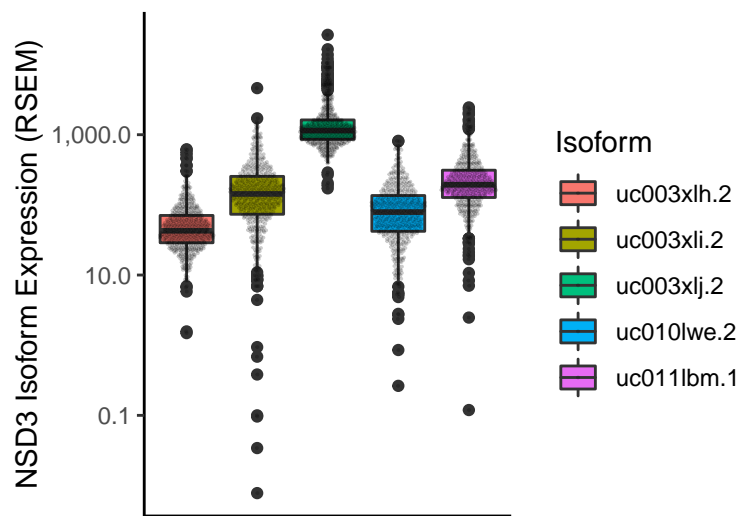
```
iso %>%
  ggplot(aes(Isoform, RSEM, fill = Isoform)) +
  geom_boxplot() +
  geom_quasirandom(pch= 21, size = 0.05, alpha =0.2) +
  scale_y_continuous("NSD3 Isoform Expression (RSEM)",
                     limits = c(0,1500)) +
  theme_classic() +
  theme(axis.text.x = element_blank()) +
  xlab("")
```



Next, the same data is plotted with a log transformed y-axis.

Boxplot with log transformed y axis to visualize all data points.

```
iso %>%
  ggplot(aes(Isoform, RSEM, fill = Isoform)) +
  geom_boxplot() +
  geom_quasirandom(pch= 21, size = 0.01, alpha =0.2) +
  scale_y_continuous("NSD3 Isoform Expression (RSEM)",
    labels = scales::comma, trans = "log10") +
  theme_classic() +
  theme(axis.text.x = element_blank()) +
  xlab("")
```

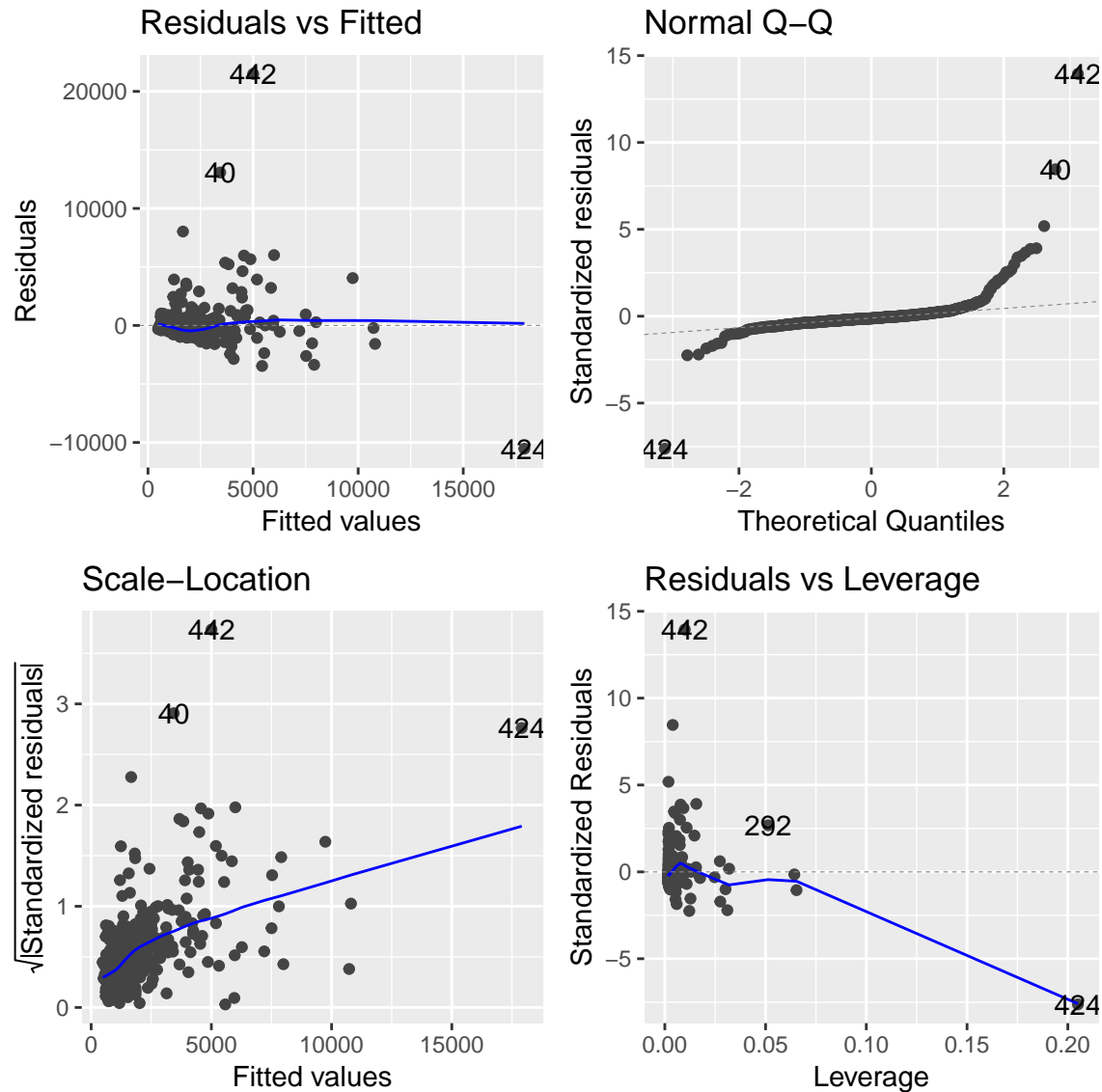


Next, I plotted the expression of short versus long isoforms to determine if there is any trend. Because there are three long isoforms, which seem to have similar expression levels, I grouped all three for the below comparison. First, linear regression is used to model the data. I found a moderate positive correlation

between long and short expression with an R-squared value of 0.492. Plotting several measures of the linear regression model shows that a linear model is generally acceptable to map the relationship with few outliers.

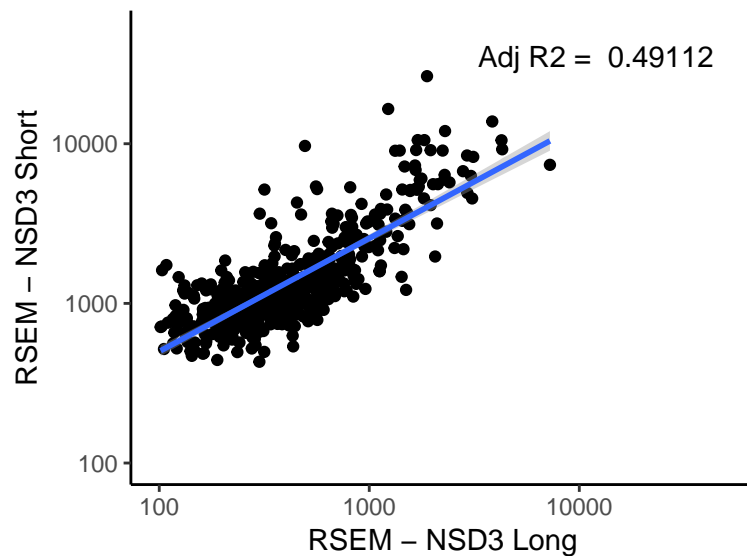
```
# Perfrom linear regression analysis on Short vs Long isoform expression analysis.  
# Long isoform expression values have been grouped.
```

```
fit <- iso %>% spread(Isoform, RSEM) %>%  
  mutate(Long = uc003xli.2 + uc010lwe.2 + uc011lbm.1) %>%  
  lm(uc003xlj.2~Long, .)  
  
summary(fit)  
  
##  
## Call:  
## lm(formula = uc003xlj.2 ~ Long, data = .)  
##  
## Residuals:  
##      Min       1Q   Median       3Q      Max   
## -10533.2   -454.0   -191.3    123.5   21497.7   
##  
## Coefficients:  
##              Estimate Std. Error t value Pr(>|t|)      
## (Intercept)  475.6156    87.2158   5.453 7.48e-08 ***  
## Long         2.4051     0.1042  23.082 < 2e-16 ***  
## ---  
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1  
##  
## Residual standard error: 1549 on 550 degrees of freedom  
## Multiple R-squared:  0.492, Adjusted R-squared:  0.4911   
## F-statistic: 532.8 on 1 and 550 DF,  p-value: < 2.2e-16  
  
autoplot(fit)
```



Scatter plot including R-squared value.

```
iso %>% spread(Isoform, RSEM) %>%
  mutate(Long = uc003xli.2 + uc010lwe.2 + uc011lbm.1) %>%
  ggplot(aes(Long, uc003xlj.2)) +
  geom_point() +
  geom_smooth(method='lm', formula=y~x) +
  annotate("text", label = paste("Adj R2 = ", signif(summary(fit)$adj.r.squared, 5)),
         x = 12000, y = 35000) +
  scale_x_log10("RSEM - NSD3 Long", limits = c(100, 50000)) +
  scale_y_log10("RSEM - NSD3 Short", limits = c(100, 50000)) +
  theme_classic()
```



After visualizing the plot it appears that the data may actually have a curvilinear shape. As long isoform expression increases, expression of the short isoform lags before also trending upward. Therefore, I decided to test if a polynomial regression model may fit the data better.

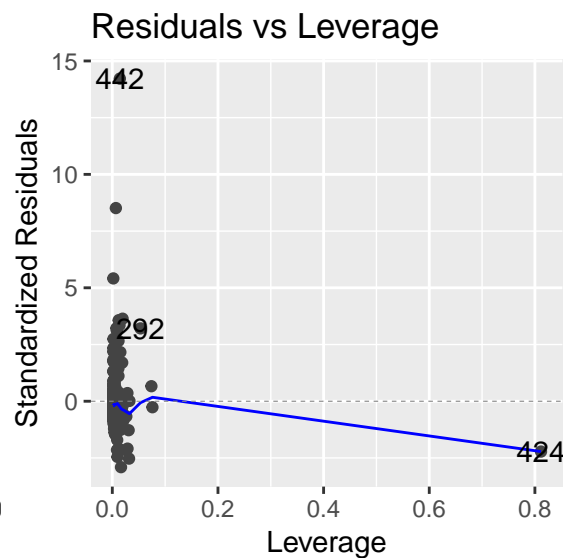
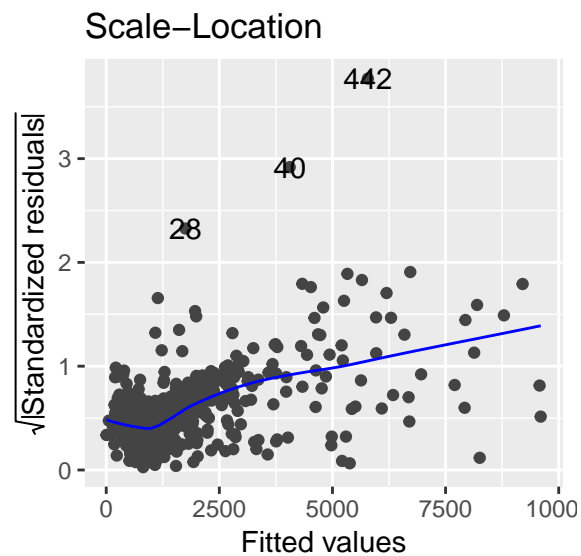
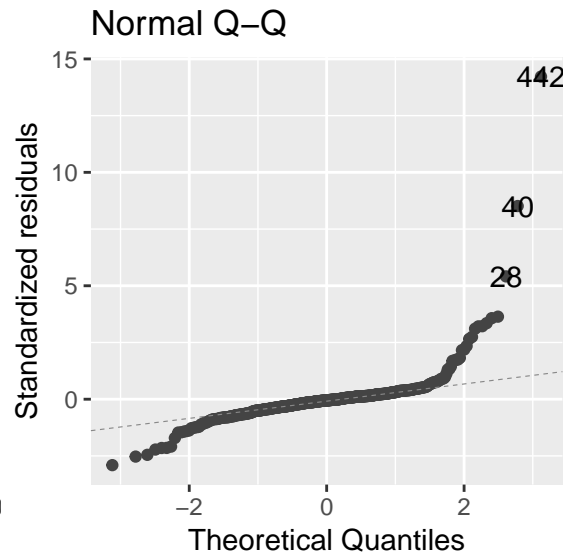
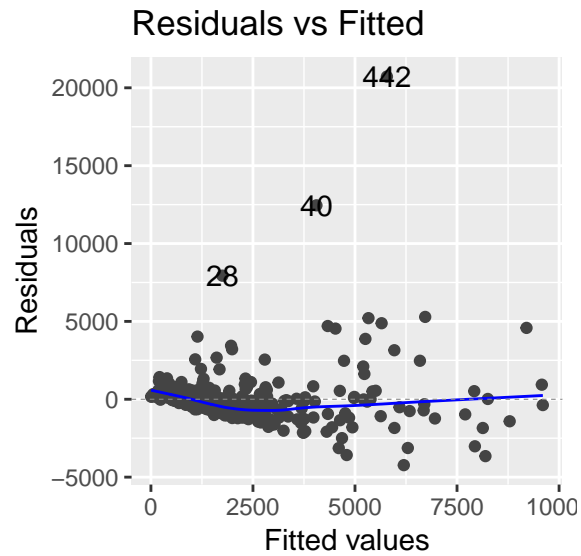
Modeling the data with a polynomial function

```
fit2 <- iso %>% spread(Isoform, RSEM) %>%
  mutate(Long = uc003xli.2 + uc010lwe.2 + uc011lbm.1) %>%
  lm(uc003xlj.2~poly(Long,2), .)

summary(fit2)

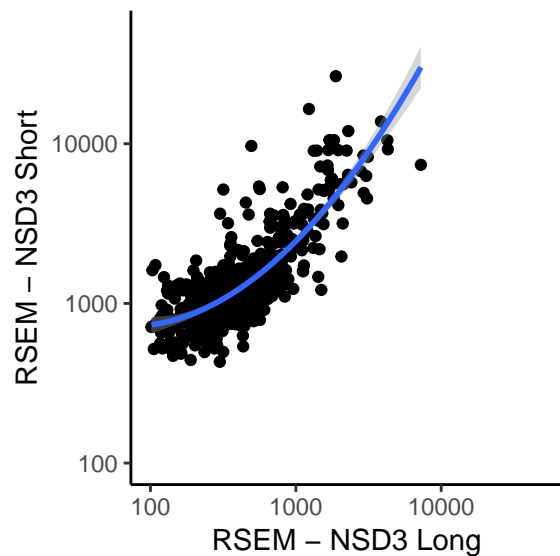
##
## Call:
## lm(formula = uc003xlj.2 ~ poly(Long, 2), data = .)
##
## Residuals:
##      Min       1Q   Median       3Q      Max
## -4230.6  -501.9   -72.2    248.0  20721.5
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)    1793.58     62.46  28.714 < 2e-16 ***
## poly(Long, 2)1  35751.32    1467.55  24.361 < 2e-16 ***
## poly(Long, 2)2 -11709.64    1467.55  -7.979 8.65e-15 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 1468 on 549 degrees of freedom
## Multiple R-squared:  0.5448, Adjusted R-squared:  0.5432
## F-statistic: 328.6 on 2 and 549 DF, p-value: < 2.2e-16

autoplot(fit2)
```



Plotting the data with a polynomial regression model

```
iso %>% spread(Isoform, RSEM) %>%
  mutate(Long = uc003xli.2 + uc010lwe.2 + uc011lbm.1) %>%
  ggplot(aes(Long, uc003xlj.2)) +
  geom_point() +
  geom_smooth(method='lm', formula=y~poly(x,2)) +
  scale_x_log10("RSEM - NSD3 Long", limits = c(100, 50000)) +
  scale_y_log10("RSEM - NSD3 Short", limits = c(100, 50000)) +
  theme_classic()
```

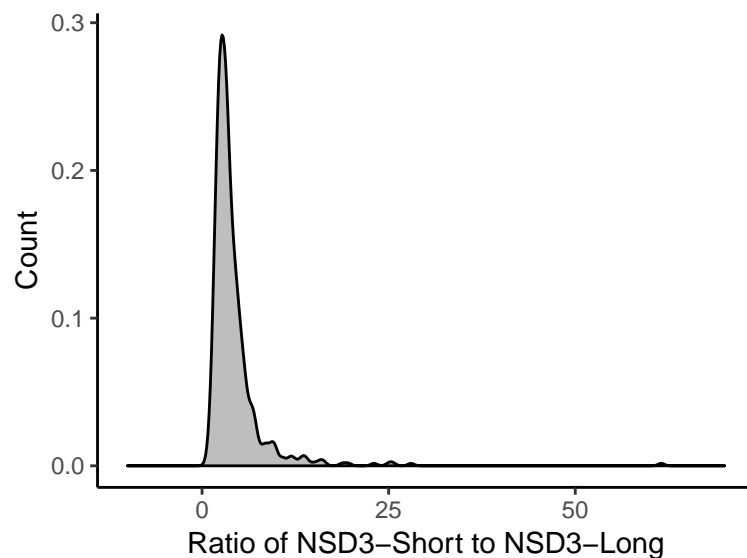


When comparing the polynomial vs linear regression model, while visually it appears to fit the data better, it did not significantly reduce the standard error, 1468 vs 1549 respectively, and the p-value remained the same.

Lastly, as an additional visualization I've plotted the distribution of the ratio between long and short NSD3 expression, which we can see is relatively uniform with a positive peak, indicating a similar ratio across samples with higher short form expression relative to the long isoforms.

Distribution of Long vs Short Isoform Ratio

```
iso %>% spread(Isoform, RSEM) %>%
  mutate(Long = uc003xli.2 + uc010lwe.2 + uc011lbm.1) %>%
  mutate(Ratio = uc003xlj.2 / Long) %>%
  ggplot(aes(Ratio)) +
  geom_density(fill = "grey") +
  theme_classic() +
  xlab("Ratio of NSD3-Short to NSD3-Long") +
  ylab("Count") +
  scale_x_continuous(limits= c(-10, 70))
```



Observations

Here I observed that expression of long NSD3 isoforms vs short isoform is moderately correlated in the context of the TCGA-LUSC data-set. This observation suggests that the isoforms are likely co-regulated, as one increases so does the other. The analysis will be helpful in framing shared and separate functions of the two isoforms.

Acknowledgement

The results shown here are in whole or part based upon data generated by the TCGA Research Network: <http://cancergenome.nih.gov>

References

1. Shen C, Ipsaro JJ, Shi J, et al. NSD3-short is an adaptor protein that couples BRD4 to the CHD8 chromatin remodeler. *Molecular cell*. 2015;60(6):847-859. doi:10.1016/j.molcel.2015.10.033.
2. Weinstein JN, Collisson EA, Mills GB, et al. The Cancer Genome Atlas Pan-Cancer Analysis Project. *Nature genetics*. 2013;45(10):1113-1120. doi:10.1038/ng.2764.

ExpID-028