Supporting Information

Hari Rallapalli

Contents

Introduction

This document details the analyses run to produce the statistics and main figures for the tumor imaging pipeline paper. The relevant raw data in .csv format are available on request:

1. "SASRESULTS.csv" - This file contains manual and semi-automated segmentation data
2. "SMO Tumor Segmentations" - This file contains the semi-automated segmentation data from the aSmo tumors used to train the empirical and linear discriminant classifiers

3. “PLX Tumor Segmentations” - This file contains the semi-automated segmentation data from the aSmo tumors entered into the PLX5622 preclinical trial

We will start by comparing semi-automated segmentation to manual segmentation:

```
suppressMessages({
  require(ggplot2)
  require(lmerTest)
  require(tidyverse)
  require(ggpubr)
  require(ggthemes)
  require(BlandAltmanLeh)
  require(MASS)
  require(FactoMineR)
  require(multcomp)
  require(pwr)
  require(cowplot)
  require(gridExtra)
  require(caret)
})

select <- dplyr::select
```

```
# plot theme (?)


plottheme <-  theme_pubr() +
theme(
strip.background = element_blank(),
panel.grid.major = element_blank(),
panel.grid.minor = element_blank(),
panel.background = element_rect(fill = "transparent", colour = NA),
plot.background = element_rect(fill = "transparent", colour = NA),
axis.line = element_line(size = 1, color = "black"),
axis.text = element_text(face = "bold", size = "17", color =  "black"),
axis.title = element_text(face = "bold", size = "19", color =  "black"),
axis.title.x = element_text(face = "bold", size = "19", color =  "black"),
axis.title.y = element_text(face = "bold", size = "19", color =  "black"),
strip.text = element_text(face = "bold", size = "17", color =  "black"),
legend.text = element_text(face = "bold", size = "17", color =  "black"),
legend.title = element_text(face = "bold", size = "17", color =  "black"),
legend.position = "right",
legend.background = element_rect(fill = "transparent", colour = NA)
)
```

Segmentation comparison - Manual vs Semi-automated

Data import and summary of number of readers/images for method comparison

```
df <- read_csv('/media/sf_MINCVM/SMO-Tumor-Data/SASRESULTS.csv') %>%
gather(
READER,
MEASUREMENTS,
READER1,
READER2,
READER3,
READER4,
READER5,
READER6,
READER7,
READER8
) %>% arrange(READER)
```

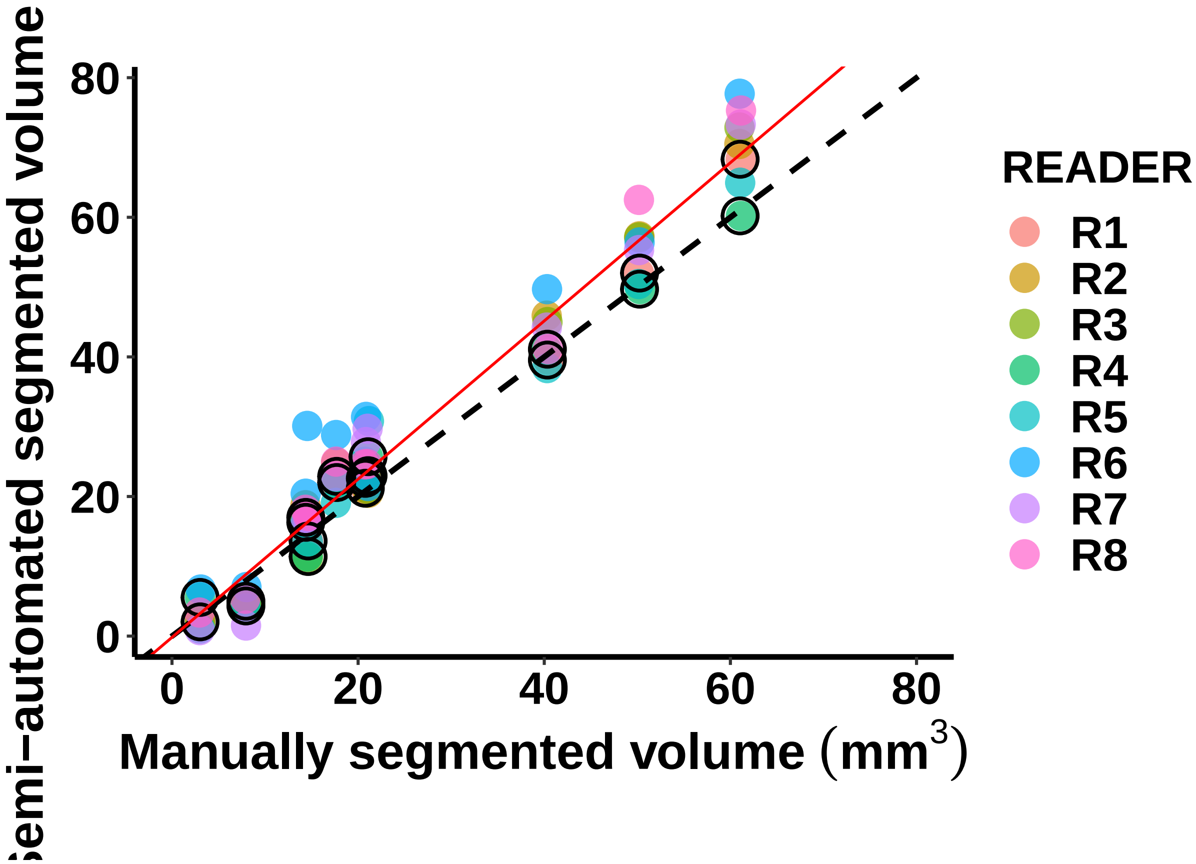
Bland Altman analysis

Manual to semi-automated segmetnation method comparisons are made by both linear comparison, and difference comparisons (Bland Altman analysis).

```
dfR1 <- df %>%
mutate (normdiffmeas = (MEASUREMENTS - MANUALVOLUME) / MANUALVOLUME) %>%
filter(READER == "READER1" | READER == "READER4")


sasplotaccuracy <- df %>% ggplot() +
aes(x = MANUALVOLUME, y = MEASUREMENTS, color = READER) +
geom_jitter(size = 5,
alpha = 0.7,
shape = 16) +
geom_abline(
intercept = 0,
slope = 1,
color = "black",
linetype = "dashed",
size = 1
) +
geom_point(
data = dfR1,
shape = 21,
size = 5.5,
stroke = 1,
color = "black"
) +
plottheme +
scale_fill_colorblind() +
coord_fixed(ratio = .75) +
ylab(bquote(bold('Semi-automated segmented volume' ~ (mm ^ 3)))) +
xlab(bquote(bold('Manually segmented volume' ~ (mm ^ 3)))) +
geom_abline(intercept = -0.18257,
slope = 1.13338,
color = "red") +
xlim(0, 80) +
scale_color_discrete(
"READER",
labels = c(
"READER1" = "R1",
"READER2" = "R2",
"READER3" = "R3",
"READER4" = "R4",
"READER5" = "R5",
"READER6" = "R6",
"READER7" = "R7",
"READER8" = "R8"
)
)


sasreg <- lm(MEASUREMENTS ~ MANUALVOLUME, data = df)

summary(sasreg)
```

```
##
## Call:
## lm(formula = MEASUREMENTS ~ MANUALVOLUME, data = df)
##
## Residuals:
##     Min      1Q  Median      3Q     Max
## -8.8057 -2.7898 -0.1717  2.1666 13.7010
##
## Coefficients:
##              Estimate Std. Error t value Pr(>|t|)
## (Intercept)  -0.18257    0.80837  -0.226    0.822
## MANUALVOLUME  1.13338    0.02615  43.349   <2e-16 ***
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Residual standard error: 4.217 on 78 degrees of freedom
## Multiple R-squared:  0.9601, Adjusted R-squared:  0.9596
## F-statistic:  1879 on 1 and 78 DF,  p-value: < 2.2e-16
```

```
sasplotaccuracy
```



```
sasblandaltmanstats <-
  bland.altman.stats(df$MEASUREMENTS, df$MANUALVOLUME)


  sasblandaltmanstats$CV =
    sasblandaltmanstats$critical.diff / 1.96 / sasblandaltmanstats$mean.diffs
```

Raw aSmo progression data

In an effort to study and parse the heterogeneity in progression of aSmo tumors, cohorts of aSmo-1 and aSmo-5 tumors were generated and studied with our pipeline.

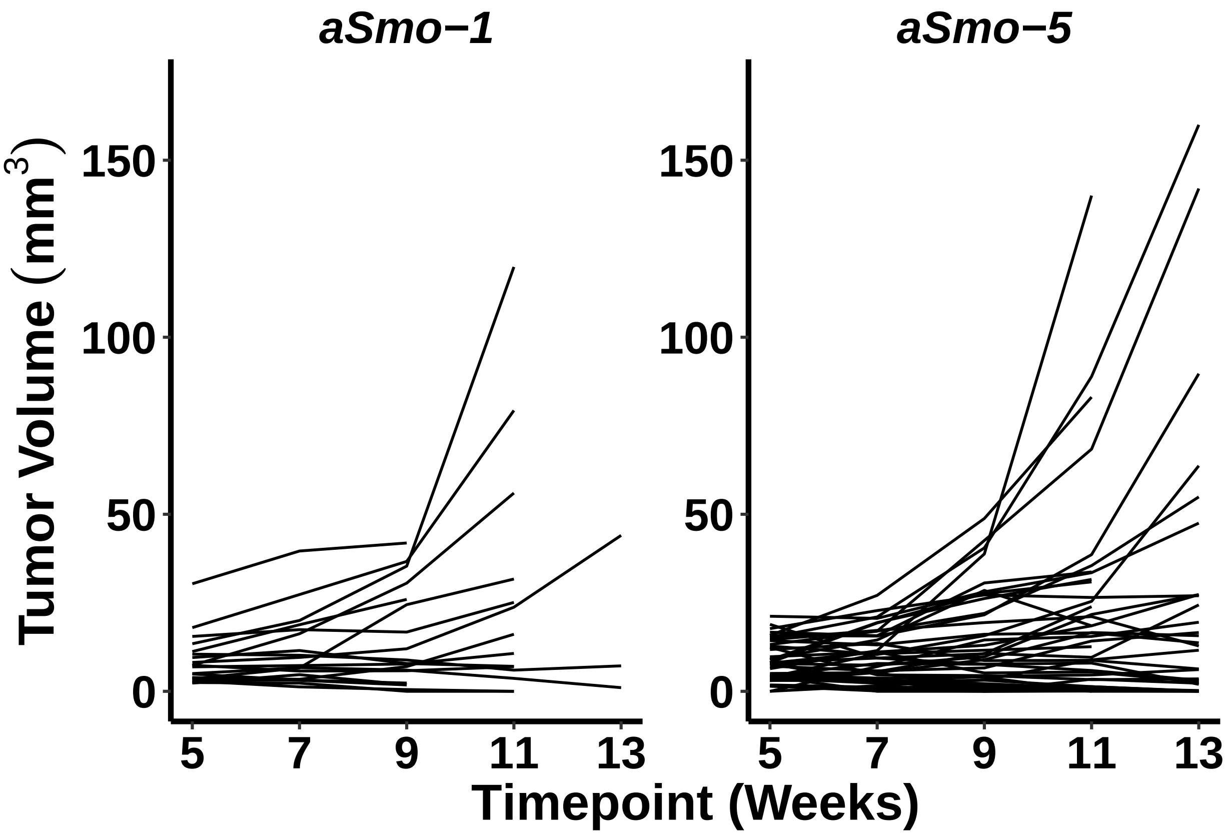
Data import and number of tumors studied in each model

```
gf <-
  read_csv('/media/sf_MINCVM/SMO-Tumor-Data/SMO Tumor Segmentations.csv') %>%
  select(MODEL, BATCH, ID, WEEK, HEMISPHERE, SEX, VOXELS, VOLUMES) %>%
  mutate(
  MOUSEID = paste(BATCH, ID),
  TUMORID = paste(MODEL, MOUSEID, HEMISPHERE),
  TIMEPOINT = parse_number(WEEK)
  ) %>%
  group_by(TUMORID) %>%
  arrange(TIMEPOINT) %>%
  mutate(VOLCHANGE = last(VOLUMES) - nth(VOLUMES, 2),
  OUTCOME = factor(ifelse(
  VOLCHANGE > 0, "Progressing", "Non-progressing"
  )))

  metcutoff = gf %>% filter(TIMEPOINT == 7, VOLUMES > 7) %>% select(TUMORID)

  gf <-
  gf %>% mutate(CLASSES = as.factor(
  ifelse(
  VOLCHANGE > 0 &
  is.element(TUMORID, metcutoff$TUMORID) == TRUE,
  'Progressing',
  ifelse(
  VOLCHANGE < 0 &
  is.element(TUMORID, metcutoff$TUMORID) == FALSE &
  TIMEPOINT > 0,
  'Non-progressing',
  'Misclassified'
  )
  )
  ))

  gf <- gf %>%
  mutate(W5VOLUMES = nth(VOLUMES, 1),
  W7VOLUMES = nth(VOLUMES, 2)) %>%
  mutate(
  VOLUMECHANGE = W7VOLUMES - W5VOLUMES,
  NORMALIZEDGROWTHRATE = VOLUMECHANGE / (2 * W5VOLUMES)
  )

  is.na(gf) <- sapply(gf, is.infinite)

  gf$INFINITYCHECK = is.na(gf$NORMALIZEDGROWTHRATE)

  gf[is.na(gf)] <- 1000

  modellabels = c("SMO1" = "aSmo-1", "SMO5" = "aSmo-5")

  gf %>% group_by(MODEL) %>%
  summarize(Number_of_tumors = n_distinct(TUMORID)) %>%
  knitr::kable()
```

MODEL	Number_of_tumors
SMO1	20
SMO5	42

Raw longitudinal progression data illustrates heterogeneity in tumor growth

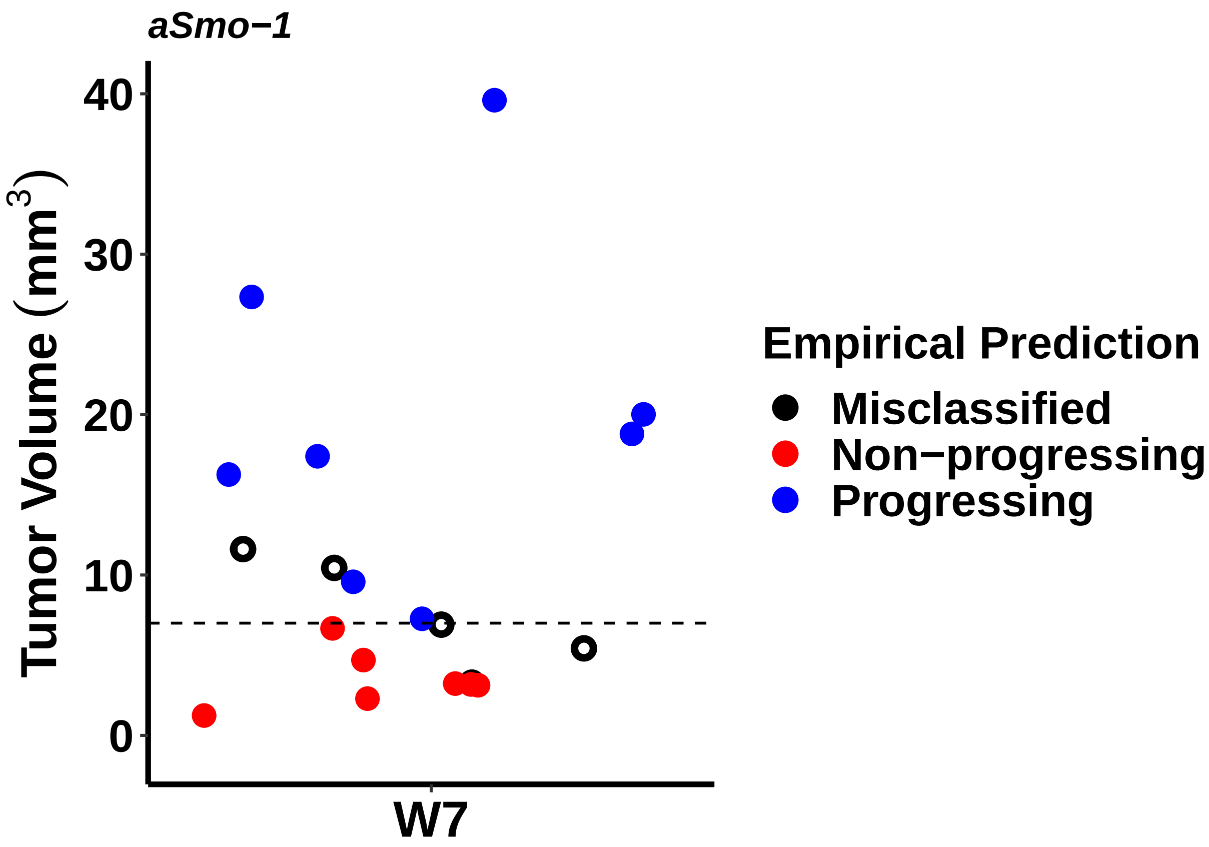
```
smorawprogressionplot <- gf %>%
  ggplot() +
  aes(x = TIMEPOINT, y = VOLUMES, group = TUMORID) +
  geom_line() +
  plottheme  +
  xlab("Timepoint (Weeks)") +
  ylab(bquote(bold('Tumor Volume' ~ (mm ^ 3)))) +
  facet_wrap( ~ MODEL,
  labeller = labeller(MODEL = modellabels),
  scales = "free") +
  theme(strip.text = element_text(face = "bold.italic")) +
  scale_x_continuous(breaks = c(5, 7, 9, 11, 13)) +
  scale_y_continuous(limits = c(0, 170))

  smorawprogressionplot
```



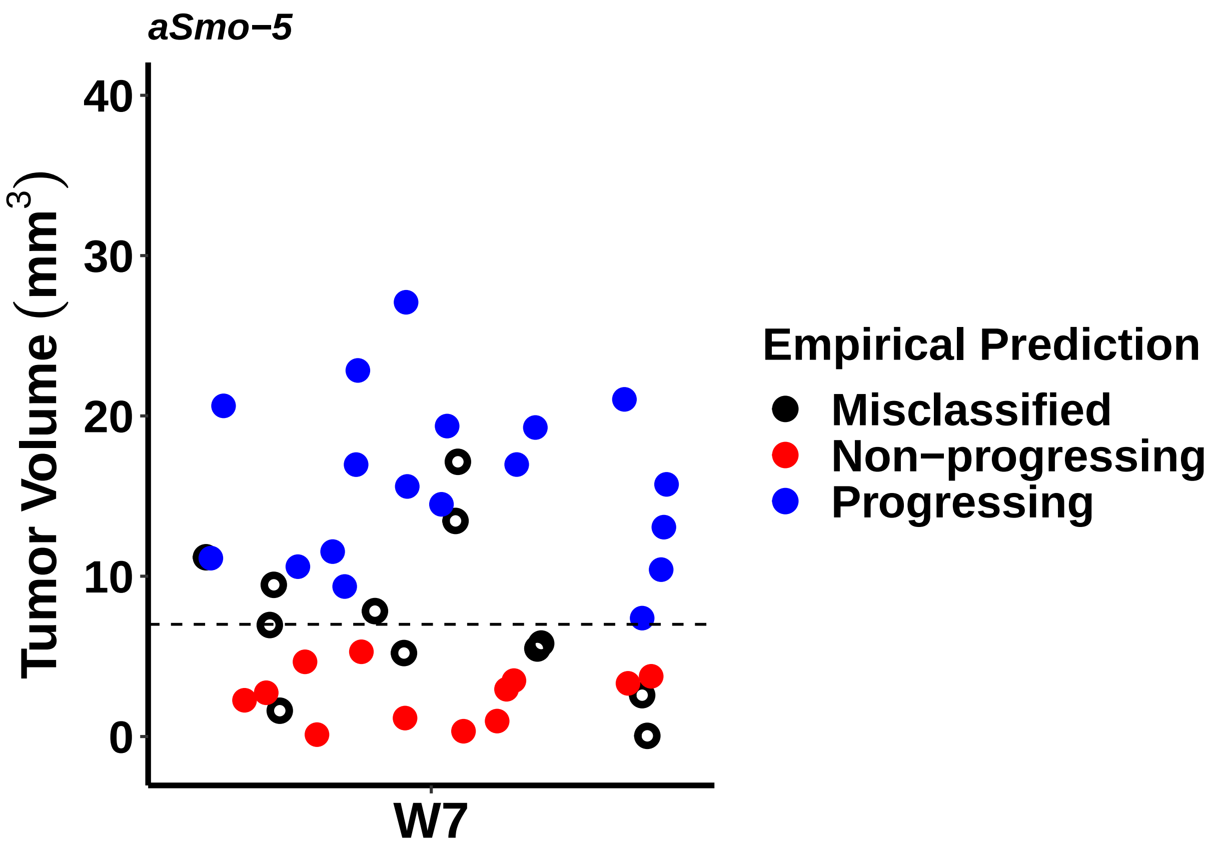
Classification of raw progression - Are tumors ‘Progressing’ or ‘Non-progressing’?

Empirical classification by means of a 7 mm^3 volume threshold at W7.

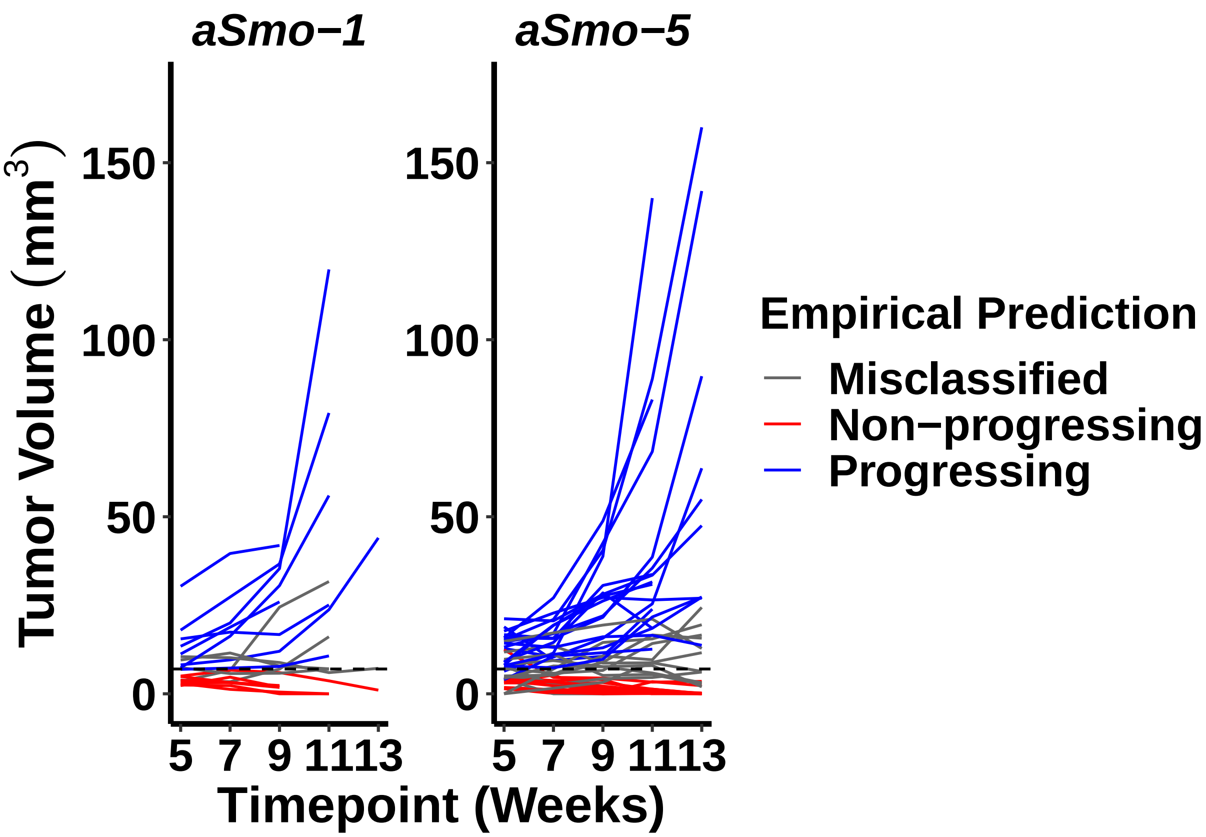
```
gfSMO1 <-
  gf %>% distinct(TUMORID, .keep_all = TRUE) %>% filter(MODEL == "SMO1")

  smo1simpleclassificationW7plot <-
  gf %>% distinct(TUMORID, .keep_all = TRUE) %>% filter(MODEL == "SMO1") %>%
  ggplot +
  aes(x = MODEL, y = W7VOLUMES, color = CLASSES) +
  geom_jitter(
  data = subset(gfSMO1, CLASSES == "Misclassified"),
  width = 0.5,
  size = 2,
  shape = 21,
  stroke = 2
  ) +
  geom_jitter(
  data = subset(gfSMO1, CLASSES != "Misclassified"),
  width = 0.5,
  size = 3,
  shape = 16,
  stroke = 2
  ) +
  plottheme +
  scale_color_manual("Empirical Prediction", values = c("black", "red", "blue")) +
  geom_hline(yintercept = 7, linetype = "dashed") +
  theme(legend.position = "right", axis.text.x = element_blank()) +
  ylab(bquote(bold('Tumor Volume' ~ (mm ^ 3)))) +
  xlab(bquote(bold('W7'))) +
  ggtitle('aSmo-1') +
  theme(plot.title = element_text(face = "bold.italic")) +
  ylim(c(-1, 40))



  smo1simpleclassificationW7plot
```



```
gfSMO5 <-
  gf %>% distinct(TUMORID, .keep_all = TRUE) %>% filter(MODEL == "SMO5")

  smo5simpleclassificationW7plot <-
  gf %>% distinct(TUMORID, .keep_all = TRUE) %>% filter(MODEL == "SMO5") %>%
  ggplot +
  aes(x = MODEL, y = W7VOLUMES, color = CLASSES) +
  geom_jitter(
  data = subset(gfSMO5, CLASSES == "Misclassified"),
  width = 0.5,
  size = 2,
  shape = 21,
  stroke = 2
  ) +
  geom_jitter(
  data = subset(gfSMO5, CLASSES != "Misclassified"),
  width = 0.5,
  size = 3,
  shape = 16,
  stroke = 2
  ) +
  plottheme +
  scale_color_manual("Empirical Prediction", values = c("black", "red", "blue")) +
  geom_hline(yintercept = 7, linetype = "dashed") +
  theme(legend.position = "right", axis.text.x = element_blank()) +
  ylab(bquote(bold('Tumor Volume' ~ (mm ^ 3)))) +
  xlab(bquote(bold('W7'))) +
  ggtitle('aSmo-5') +
  theme(plot.title = element_text(face = "bold.italic")) +
  ylim(c(-1, 40))



  smo5simpleclassificationW7plot
```



```
smosimpleclassificationplot <- gf %>%
  ggplot() +
  aes(
  x = TIMEPOINT,
  y = VOLUMES,
  group = TUMORID,
  color = CLASSES
  ) +
  geom_line() +
  plottheme +
  scale_color_manual("Empirical Prediction", values = c("#666666", "red", "blue")) +
  geom_hline(yintercept = 7, linetype = "dashed") +
  theme(legend.position = "right") +
  xlab("Timepoint (Weeks)") +
  ylab(bquote(bold('Tumor Volume' ~ (mm ^ 3)))) +
  facet_wrap( ~ MODEL,
  labeller = labeller(MODEL = modellabels),
  scales = "free") +
  theme(strip.text = element_text(face = "bold.italic")) +
  scale_x_continuous(breaks = c(5, 7, 9, 11, 13)) +
  scale_y_continuous(limits = c(0, 170))



  smosimpleclassificationplot
```



Linear discriminant analysis (LDA) based on early tumor progression features

```
reg <-
  gf %>% lda(
  OUTCOME ~ W7VOLUMES + VOLUMECHANGE + NORMALIZEDGROWTHRATE + INFINITYCHECK,
  data = .,
  na.action = "na.omit"
  )

  regCV <-
  gf %>% lda(
  OUTCOME ~ W7VOLUMES + VOLUMECHANGE + NORMALIZEDGROWTHRATE + INFINITYCHECK,
  data = .,
  na.action = "na.omit",
  CV = TRUE
  )
```

```r
reg
```

```
## Call:
## lda(OUTCOME ~ W7VOLUMES + VOLUMECHANGE + NORMALIZEDGROWTHRATE +
##     INFINITYCHECK, data = ., na.action = "na.omit")
##
## Prior probabilities of groups:
## Non-progressing     Progressing
##       0.4244604       0.5755396
##
## Group means:
##                  W7VOLUMES VOLUMECHANGE NORMALIZEDGROWTHRATE
## Non-progressing   5.367661  -0.04005085          -0.03168919
## Progressing      12.919813   2.51906250          62.61649404
##                  INFINITYCHECKTRUE
## Non-progressing             0.0000
## Progressing                 0.0625
##
## Coefficients of linear discriminants:
##                               LD1
## W7VOLUMES                0.1745186
## VOLUMECHANGE            -0.1698068
## NORMALIZEDGROWTHRATE     1.8210264
## INFINITYCHECKTRUE    -1817.5314066
```

```r
res <- table(gf$OUTCOME, regCV$class)
print('Classification Accuracy:')
```

```
## [1] "Classification Accuracy:"
```

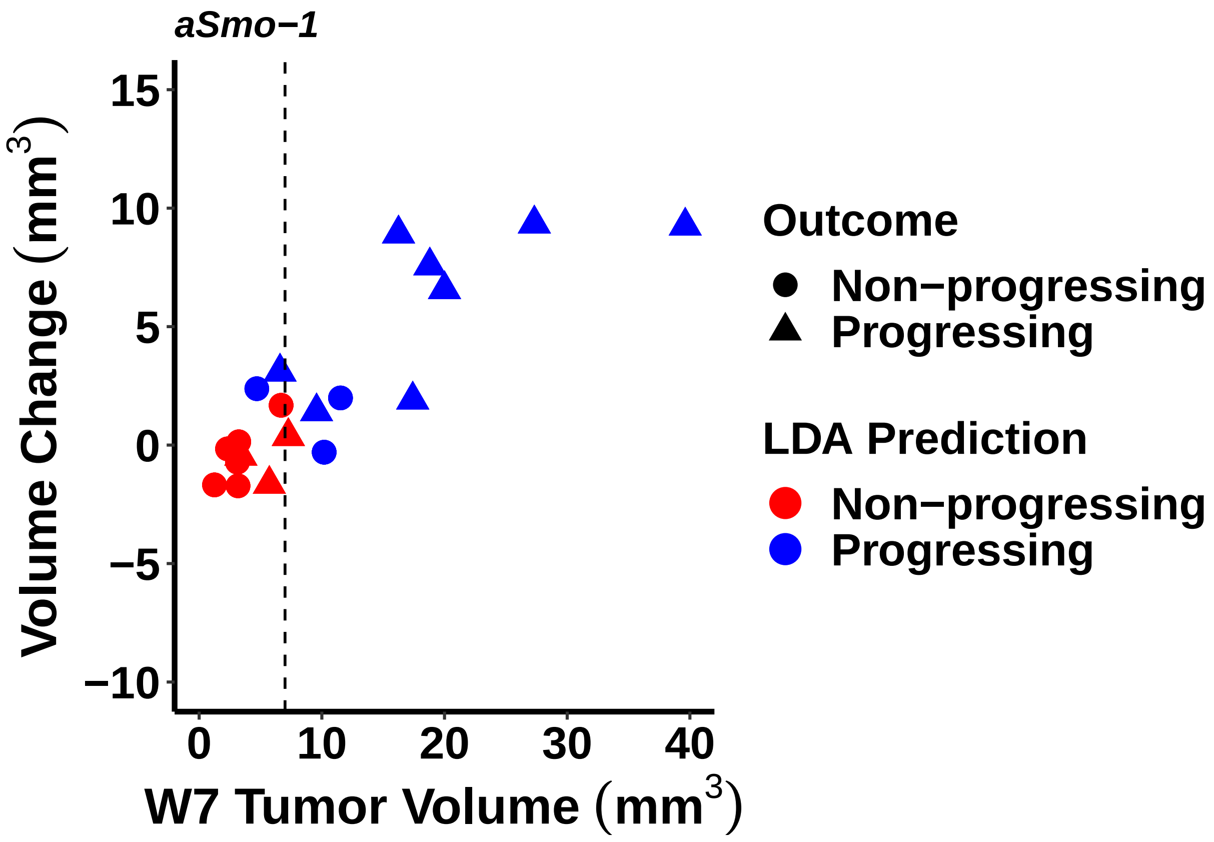
```r
diag(prop.table(res, 1))
```

```
## Non-progressing     Progressing
##       0.6440678       0.7375000
```

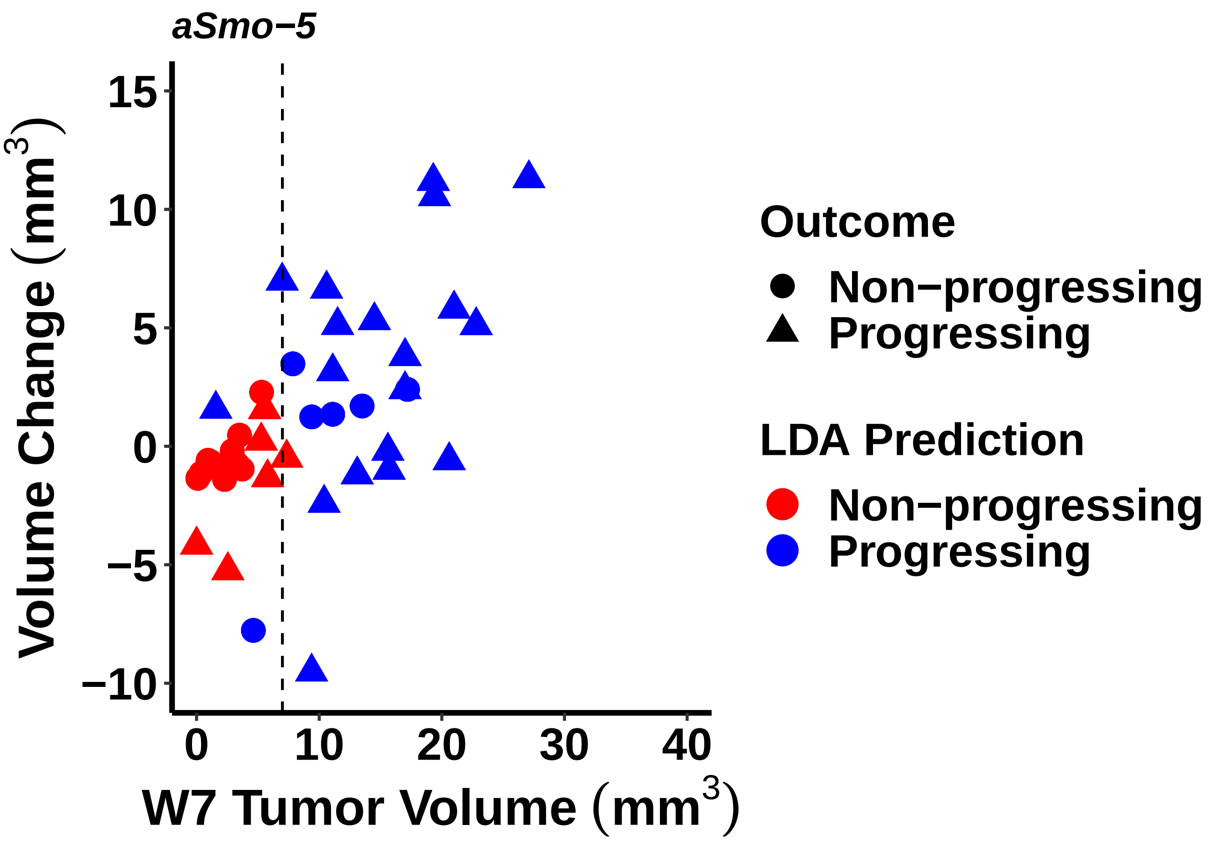
```r
preg <- predict(object = reg, newdata = gf)

gf$LDA_PREDICTION = preg$class

gf %>% group_by(LDA_PREDICTION, MODEL, OUTCOME) %>%
summarize(n = n_distinct(TUMORID)) %>%
spread(MODEL, n) %>%
knitr::kable()
```

LDA_PREDICTION	OUTCOME	SMO1	SMO5
Non-progressing	Non-progressing	6	11
Non-progressing	Progressing	3	6
Progressing	Non-progressing	3	6
Progressing	Progressing	8	19

```r
smo1regplot <- gf %>% filter(MODEL == "SMO1") %>% ggplot() +
aes(
x = W7VOLUMES,
y = VOLUMECHANGE,
shape = OUTCOME,
color = LDA_PREDICTION
) +
geom_point(size = 3, stroke = 2) +
plottheme +
theme(legend.position = "right") +
geom_vline(xintercept = 7, linetype = "dashed") +
scale_color_manual("LDA Prediction", values = c("red", "blue")) +
scale_shape("Outcome") +
guides(color = guide_legend(order = 2),
shape = guide_legend(order = 1)) +
xlab(bquote(bold('W7 Tumor Volume' ~ (mm ^ 3)))) +
ylab(bquote(bold('Volume Change' ~ (mm ^ 3)))) +
ggtitle('aSmo-1') +
theme(plot.title = element_text(face = "bold.italic")) +
coord_cartesian(xlim = c(0, 40), ylim = c(-10, 15))


smo1regplot
```



```r
smo5regplot <- gf %>% filter(MODEL == "SMO5") %>% ggplot() +
aes(
x = W7VOLUMES,
y = VOLUMECHANGE,
shape = OUTCOME,
color = LDA_PREDICTION
) +
geom_point(size = 3, stroke = 2) +
plottheme +
theme(legend.position = "right") +
geom_vline(xintercept = 7, linetype = "dashed") +
scale_color_manual("LDA Prediction", values = c("red", "blue")) +
scale_shape("Outcome") +
guides(color = guide_legend(order = 2),
shape = guide_legend(order = 1)) +
xlab(bquote(bold('W7 Tumor Volume') ~ (mm ^ 3)))) +
ylab(bquote(bold('Volume Change') ~ (mm ^ 3)))) +
ggtitle('aSmo-5') +
theme(plot.title = element_text(face = "bold.italic")) +
coord_cartesian(xlim = c(0, 40), ylim = c(-10, 15))


smo5regplot
```



Application - Preclinical trial of PLX5622

Data import and summary of tumor stratification into treatment arms

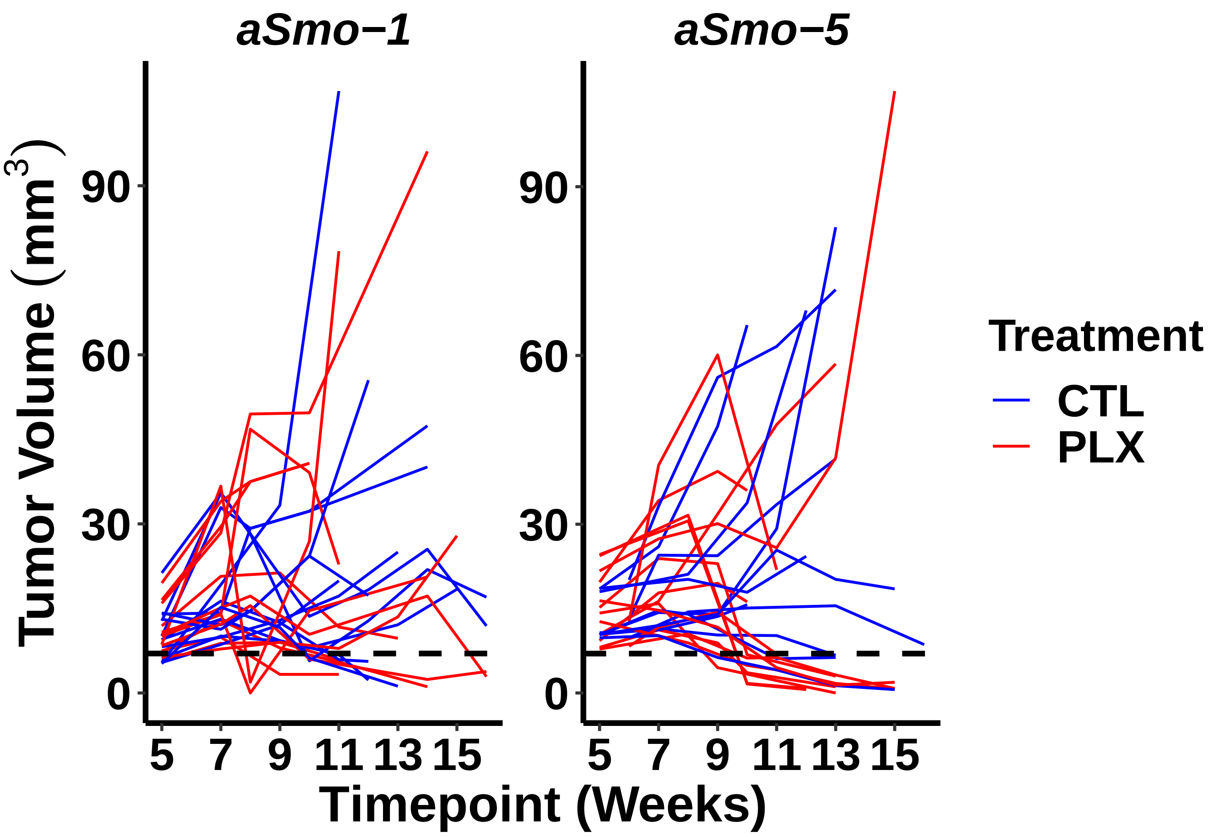
```r
plxf <-
  read_csv('/media/sf_MINCVM/SMO-Tumor-Data/PLX Tumor Segmentations.csv') %>%
  select(MODEL,
  BATCH,
  ID,
  WEEK,
  HEMISPHERE,
  SEX,
  VOXELS,
  VOLUMES,
  TREATMENT,
  TREATMENTSTART) %>%
  mutate(
  MOUSEID = paste(BATCH, ID),
  TUMORID = paste(MODEL, MOUSEID, HEMISPHERE),
  TIMEPOINT = parse_number(WEEK)
  ) %>%
  group_by(TUMORID) %>%
  arrange(TIMEPOINT) %>%
  mutate(
  VOLCHANGE = last(VOLUMES) - nth(VOLUMES, 2),
  FINALVOLUME = last(VOLUMES),
  PRETREATMENTVOLUME = nth(VOLUMES, 2),
  OUTCOME = as.factor(ifelse(
  VOLCHANGE > 0, "Progressing", "Non-progressing"
  ))
  )

  metcutoff <-
  plxf %>%  arrange(TIMEPOINT) %>% filter(TIMEPOINT == nth(TIMEPOINT, 2) &
  VOLUMES > 7) %>% select(TUMORID)

  plxf <-
  plxf %>% mutate(CLASSES = as.factor(
  ifelse(
  VOLCHANGE > 0 &
  is.element(TUMORID, metcutoff$TUMORID) == TRUE,
  'non-responding',
  ifelse(
  VOLCHANGE < 0 &
  is.element(TUMORID, metcutoff$TUMORID) == FALSE &
  TIMEPOINT > 0,
  'falsely included',
  'responded'
  )
  )
  ))
  plxf <- plxf %>%
  mutate(W5VOLUMES = nth(VOLUMES, 1),
  W7VOLUMES = nth(VOLUMES, 2)) %>%
  mutate(
  VOLUMECHANGE = W7VOLUMES - W5VOLUMES,
  NORMALIZEDGROWTHRATE = VOLUMECHANGE / (2 * W5VOLUMES)
  )

  is.na(plxf) <- sapply(plxf, is.infinite)
  plxf$INFINITYCHECK = is.na(plxf$NORMALIZEDGROWTHRATE)
  plxf[is.na(plxf)] <- 1000

  plxf %>% group_by(MODEL, TREATMENT) %>% filter(W7VOLUMES >= 7) %>%
  summarize(n = n_distinct(TUMORID)) %>%
  spread(MODEL, n) %>%
  knitr::kable()
```

TREATMENT	SMO1	SMO5
CTL	13	12
PLX	13	14

Raw longitudinal progression stratified by tumor model and treatment arm

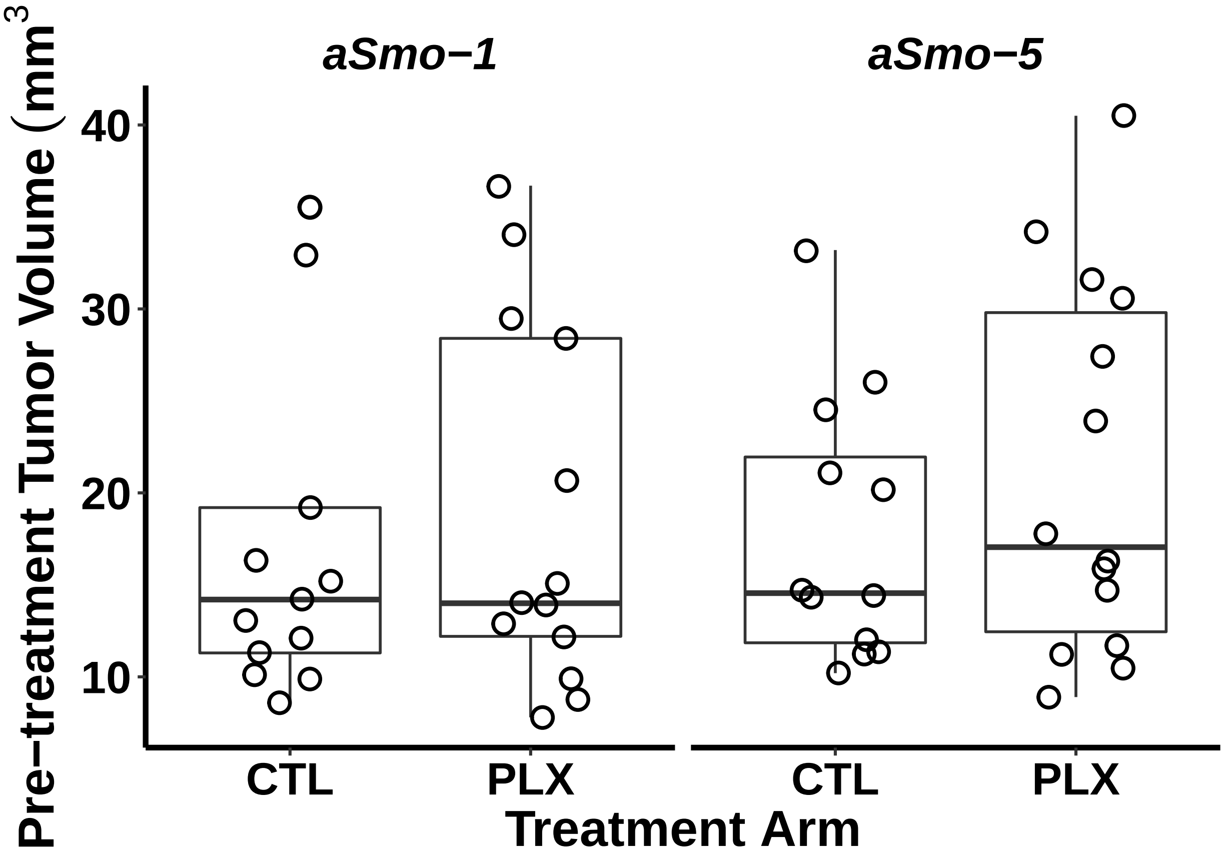
```r
plxsimpleclassificationplot.bothmodels <-
  plxf %>% filter(TREATMENT != "U" &&
  W7VOLUMES >= 7) %>% filter(CLASSES != "falsely included") %>%
  ggplot() +
  aes(
  x = TIMEPOINT,
  y = VOLUMES,
  group = TUMORID,
  color = TREATMENT
  ) +
  geom_line() +
  plottheme +
  scale_color_manual("Treatment", values = c("blue", "red")) +
  facet_wrap( ~ MODEL,
  labeller = labeller(MODEL = modellabels),
  scales = "free") +
  scale_x_continuous(breaks = c(5, 7, 9, 11, 13, 15, 17)) +
  geom_hline(yintercept = 7,
  size = 1,
  linetype = 'dashed') +
  xlab("Timepoint (Weeks)") +
  ylab(bquote(bold('Tumor Volume' ~ (mm ^ 3)))) +
  theme(strip.text = element_text(face = "bold.italic"),
  legend.position = "right")
```

```
plxsimpleclassificationplot.bothmodels
```



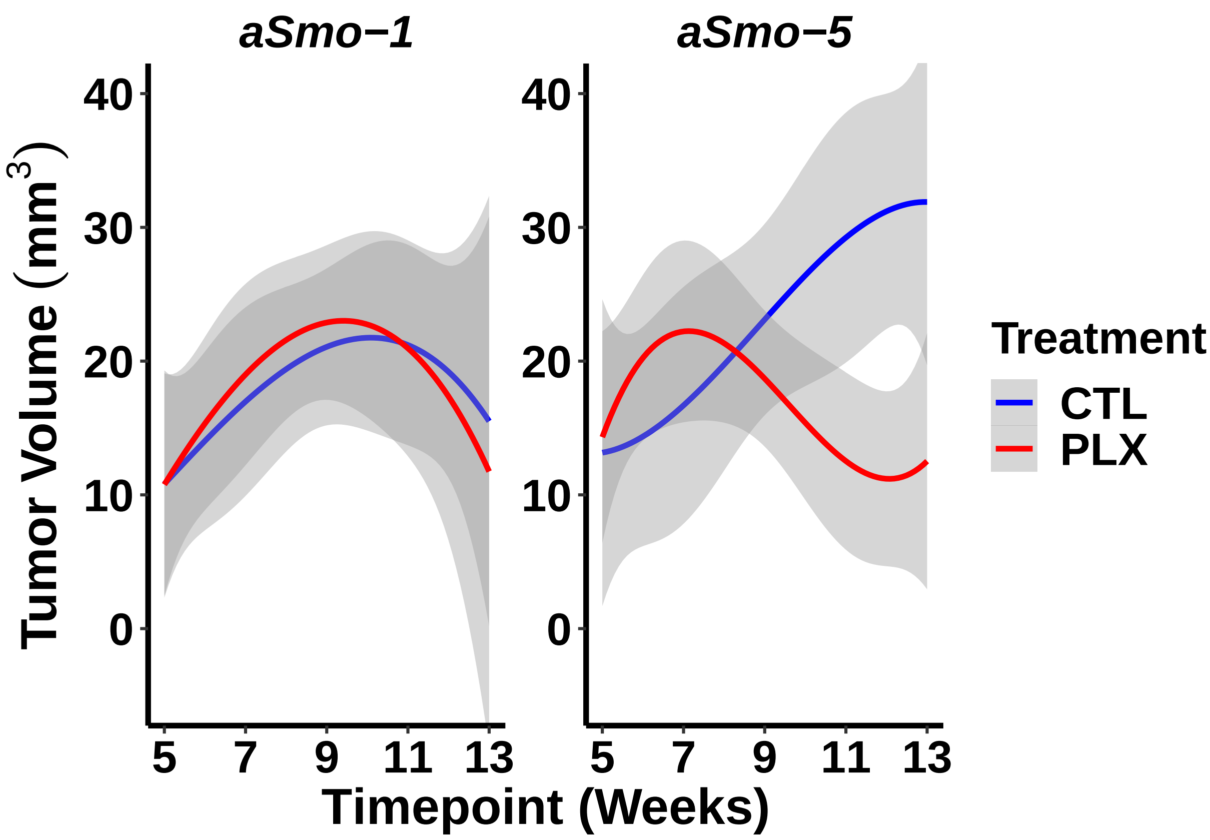
Pre-treatment tumor volumes did not differ significantly across treatment arms nor tumor models

```
plxpretreatmentvolumes <-
  plxf %>% distinct(TUMORID, .keep_all = TRUE) %>% filter(TREATMENT != "U") %>%
  filter(W7VOLUMES >= 7) %>% filter(CLASSES != "falsely included") %>%
  filter(TIMEPOINT <= 13) %>%
  ggplot() +
  aes(x = TREATMENT, y = W7VOLUMES) +
  geom_boxplot(outlier.shape = NA) +
  geom_jitter(
  shape = 21,
  stroke = 1,
  width = 0.2,
  size = 3
  ) +
  plottheme +
  scale_color_manual("Treatment", values = c("blue", "red")) +
  facet_wrap( ~ MODEL, labeller = labeller(MODEL = modellabels)) +
  xlab("Treatment Arm") +
  ylab(bquote(bold('Pre-treatment Tumor Volume' ~ (mm ^ 3)))) +
  theme(strip.text = element_text(face = "bold.italic"),
  legend.position = c(0.95, 0.5))



  plxpretreatmentvolumes
```



Response to therapy is heterogeneous and tumor model-dependent

```
plxsimpleclassificationsmooth.bothmodels <-
  plxf %>% filter(TREATMENT != "U") %>% filter(W7VOLUMES >= 7) %>%
  filter(CLASSES != "falsely included") %>% filter(TIMEPOINT <= 13) %>%
  ggplot() +
  aes(x = TIMEPOINT, y = VOLUMES, color = TREATMENT) +
  # geom_smooth(formula = "y ~ poly(x,1)", span = 1) +
  geom_smooth(method = lm, formula = y ~ splines::bs(x, 3)) +
  plottheme +
  scale_color_manual("Treatment", values = c("blue", "red")) +
  facet_wrap( ~ MODEL,
  labeller = labeller(MODEL = modellabels),
  scales = "free") +
  coord_cartesian(xlim = c(5, 13), ylim = c(-5 , 40)) +
  scale_x_continuous(breaks = c(5, 7, 9, 11, 13, 15, 17)) +
  scale_y_continuous(breaks = c(-20,-10, 0, 10, 20, 30, 40)) +
  xlab("Timepoint (Weeks)") +
  ylab(bquote(bold('Tumor Volume' ~ (mm ^ 3)))) +
  theme(strip.text = element_text(face = "bold.italic"))


  plxsimpleclassificationsmooth.bothmodels
```



Effects of treatment on endpoint tumor volume reinforce this differential response

```
plxmodel <-
  plxf %>% filter(TREATMENT != "U") %>% filter(W7VOLUMES >= 7) %>%
  filter(TIMEPOINT == 5 | TIMEPOINT == 6) %>%
  lmerTest::lmer(FINALVOLUME ~ PRETREATMENTVOLUME + MODEL:TREATMENT  + (1 |
  MOUSEID) + 0,
  data = .)

  summary(plxmodel, ddf = "Kenward-Roger")
```

```
## Linear mixed model fit by REML. t-tests use Kenward-Roger's method [
## lmerModLmerTest]
## Formula:
## FINALVOLUME ~ PRETREATMENTVOLUME + MODEL:TREATMENT + (1 | MOUSEID) +
##     0
##    Data: .
```

```
##
## REML criterion at convergence: 463.8
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.2447 -0.6075 -0.2039  0.1778  2.7757
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  MOUSEID  (Intercept)  90.52    9.514
##  Residual             681.12   26.098
## Number of obs: 52, groups:  MOUSEID, 30
##
## Fixed effects:
##                         Estimate Std. Error       df t value Pr(>|t|)
## PRETREATMENTVOLUME        1.3783     0.4463  44.8093   3.088  0.00345 **
## MODELSM01:TREATMENTCTL    3.4125    11.3429  37.2509   0.301  0.76520
## MODELSM05:TREATMENTCTL    9.2684    11.5638  34.0453   0.802  0.42840
## MODELSM01:TREATMENTPLX    1.1748    11.5991  37.2238   0.101  0.91987
## MODELSM05:TREATMENTPLX -10.8190    12.2540  33.6510  -0.883  0.38356
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##                      PRETRE MODELSM01:TREATMENTC MODELSM05:TREATMENTC
## MODELSM01:TREATMENTC -0.688
## MODELSM05:TREATMENTC -0.669  0.460
## MODELSM01:TREATMENTP -0.703  0.484                0.471
## MODELSM05:TREATMENTP -0.754  0.518                0.504
##                      MODELSM01:TREATMENTP
## MODELSM01:TREATMENTC
## MODELSM05:TREATMENTC
## MODELSM01:TREATMENTP
## MODELSM05:TREATMENTP  0.530
```

Statistical power calculations for PLX5622 preclinical trial - after empirical classification

```
plxpwr <- data.frame(
  n1 = plxf %>% filter(TREATMENT == "PLX") %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  n2 = plxf %>% filter(TREATMENT == "CTL") %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  M1 = plxf %>% filter(TREATMENT == "PLX") %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% colMeans(., na.rm = TRUE),
  M2 = plxf %>% filter(TREATMENT == "CTL") %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% colMeans(., na.rm = TRUE),
  S1 = plxf %>% filter(TREATMENT == "PLX") %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% sapply(., sd, na.rm = TRUE),
  S2 = plxf %>% filter(TREATMENT == "CTL") %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% sapply(., sd, na.rm = TRUE)
  )

  plxpwr$d = (plxpwr$M1 - plxpwr$M2) / sqrt(((plxpwr$S1 ^ 2) + (plxpwr$S2 ^
  2)) / 2)


  print("Post-hoc power calculation based on calculated effect size")
```

```
## [1] "Post-hoc power calculation based on calculated effect size"
```

```
pwr.t2n.test(plxpwr$n1  ,
plxpwr$n2 ,
plxpwr$d,
sig.level = 0.05,
power = NULL)
```

```
##
##      t test power calculation
##
##              n1 = 27
##              n2 = 25
##               d = 0.2892803
##       sig.level = 0.05
##           power = 0.1756415
##     alternative = two.sided
```

```
print("a priori sample size calculation based on calculated effect size")
```

```
## [1] "a priori sample size calculation based on calculated effect size"
```

```
pwr.t.test(n = NULL,
plxpwr$d,
sig.level = 0.05,
power = 0.80)
```

```
##
##      Two-sample t test power calculation
##
##               n = 188.5505
##               d = 0.2892803
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
plxpwr <- plxpwr %>% mutate(
  n1.SM01 = plxf %>% filter(TREATMENT == "PLX") %>% filter(MODEL == "SM01") %>%
  ungroup() %>% filter(W7VOLUMES >= 7)  %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%  nrow(),
  n2.SM01 = plxf %>% filter(TREATMENT == "CTL") %>% filter(MODEL == "SM01") %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%  nrow(),
  M1.SM01 = plxf %>% filter(TREATMENT == "PLX") %>% filter(MODEL == "SM01") %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  colMeans(., na.rm = TRUE),
  M2.SM01 = plxf %>% filter(TREATMENT == "CTL") %>% filter(MODEL == "SM01") %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  colMeans(., na.rm = TRUE),
  S1.SM01 = plxf %>% filter(TREATMENT == "PLX") %>% filter(MODEL == "SM01") %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  sapply(., sd, na.rm = TRUE),
  S2.SM01 = plxf %>% filter(TREATMENT == "CTL") %>% filter(MODEL == "SM01")  %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  sapply(., sd, na.rm = TRUE)
  )

  plxpwr <-
  plxpwr %>%
  mutate(d.SM01 = (plxpwr$M1.SM01 - plxpwr$M2.SM01) / sqrt(((plxpwr$S1.SM01 ^
  2) + (plxpwr$S2.SM01 ^ 2)) / 2))


  print("Post-hoc power calculation based on calculated effect size")
```

```
## [1] "Post-hoc power calculation based on calculated effect size"
```

```
  pwr.t2n.test(
  plxpwr$n1.SM01  ,
  plxpwr$n2.SM01 ,
  plxpwr$d.SM01,
  sig.level = 0.05,
  power = NULL
  )
```

```
##
##      t test power calculation
##
##              n1 = 13
##              n2 = 13
##               d = 0.03746594
##       sig.level = 0.05
##           power = 0.05096509
##     alternative = two.sided
```

```
  print("a priori sample size calculation based on calculated effect size")
```

```
## [1] "a priori sample size calculation based on calculated effect size"
```

```
  pwr.t.test(n = NULL,
  plxpwr$d.SM01,
  sig.level = 0.05,
  power = 0.80)
```

```
##
##      Two-sample t test power calculation
##
##               n = 11184.09
##               d = 0.03746594
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
plxpwr <- plxpwr %>% mutate(
  n1.SM05 = plxf %>% filter(TREATMENT == "PLX") %>% filter(MODEL == "SM05") %>%
  ungroup() %>% filter(W7VOLUMES >= 7)  %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%  nrow(),
  n2.SM05 = plxf %>% filter(TREATMENT == "CTL") %>% filter(MODEL == "SM05") %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%  nrow(),
  M1.SM05 = plxf %>% filter(TREATMENT == "PLX") %>% filter(MODEL == "SM05") %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  colMeans(., na.rm = TRUE),
  M2.SM05 = plxf %>% filter(TREATMENT == "CTL") %>% filter(MODEL == "SM05") %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  colMeans(., na.rm = TRUE),
  S1.SM05 = plxf %>% filter(TREATMENT == "PLX") %>% filter(MODEL == "SM05")  %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  sapply(., sd, na.rm = TRUE),
  S2.SM05 = plxf %>% filter(TREATMENT == "CTL") %>% filter(MODEL == "SM05")  %>%
  ungroup() %>% filter(W7VOLUMES >= 7) %>%
  distinct(TUMORID, .keep_all = TRUE) %>% select(FINALVOLUME) %>%
  sapply(., sd, na.rm = TRUE)
  )

  plxpwr <-
  plxpwr %>%
  mutate(d.SM05 = (plxpwr$M1.SM05 - plxpwr$M2.SM05) / sqrt(((plxpwr$S1.SM05 ^
  2) + (plxpwr$S2.SM05 ^ 2)) / 2))


  print("Post-hoc power calculation based on calculated effect size")
```

```
## [1] "Post-hoc power calculation based on calculated effect size"
```

```
  pwr.t2n.test(
  plxpwr$n1.SM05  ,
  plxpwr$n2.SM05 ,
  plxpwr$d.SM05,
  sig.level = 0.05,
  power = NULL
  )
```

```
##
##      t test power calculation
##
##              n1 = 14
##              n2 = 12
##               d = 0.5239419
##       sig.level = 0.05
##           power = 0.2485027
##     alternative = two.sided
```

```
  print("a priori sample size calculation based on calculated effect size")
```

```
## [1] "a priori sample size calculation based on calculated effect size"
```

```
  pwr.t.test(n = NULL,
  plxpwr$d.SM05,
  sig.level = 0.05,
  power = 0.80)
```

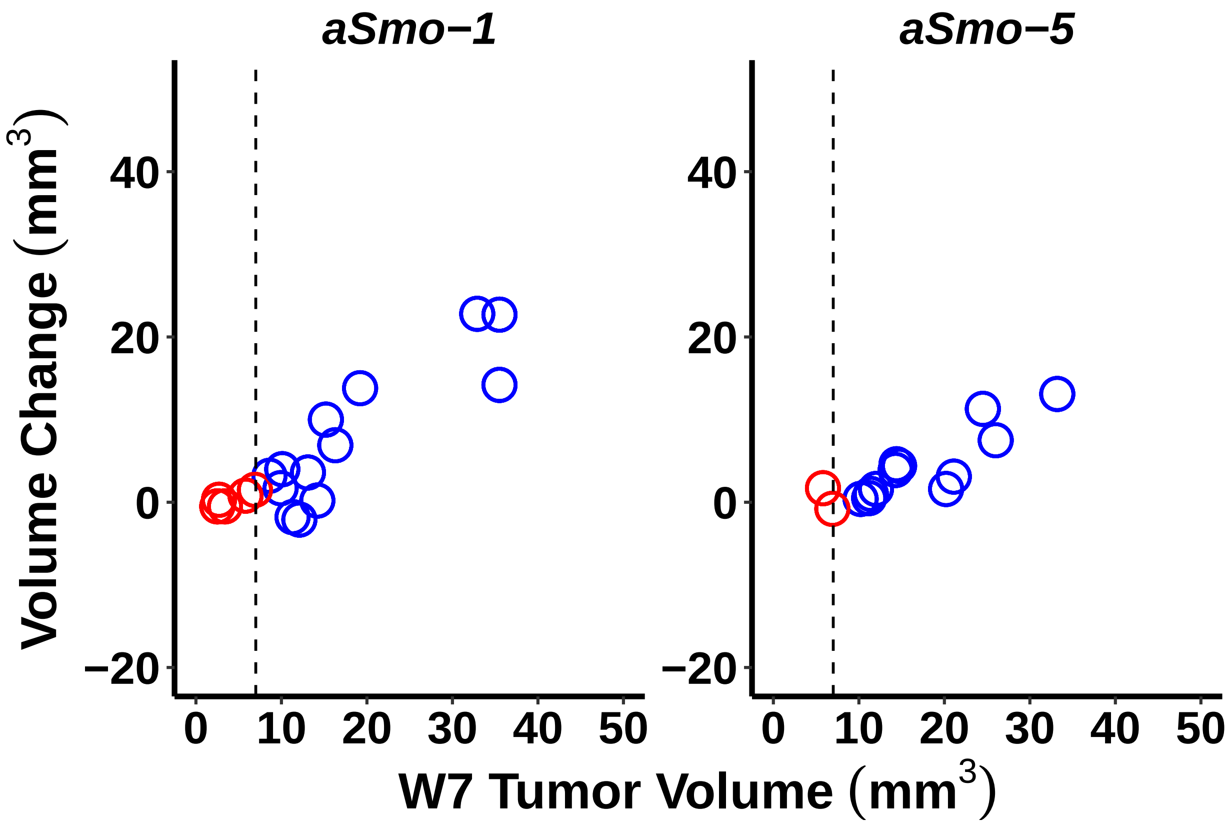
```
##
##      Two-sample t test power calculation
##
##               n = 58.15957
##               d = 0.5239419
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

Reclassification of PLX5622 preclinical trial data using trained LDA classifier

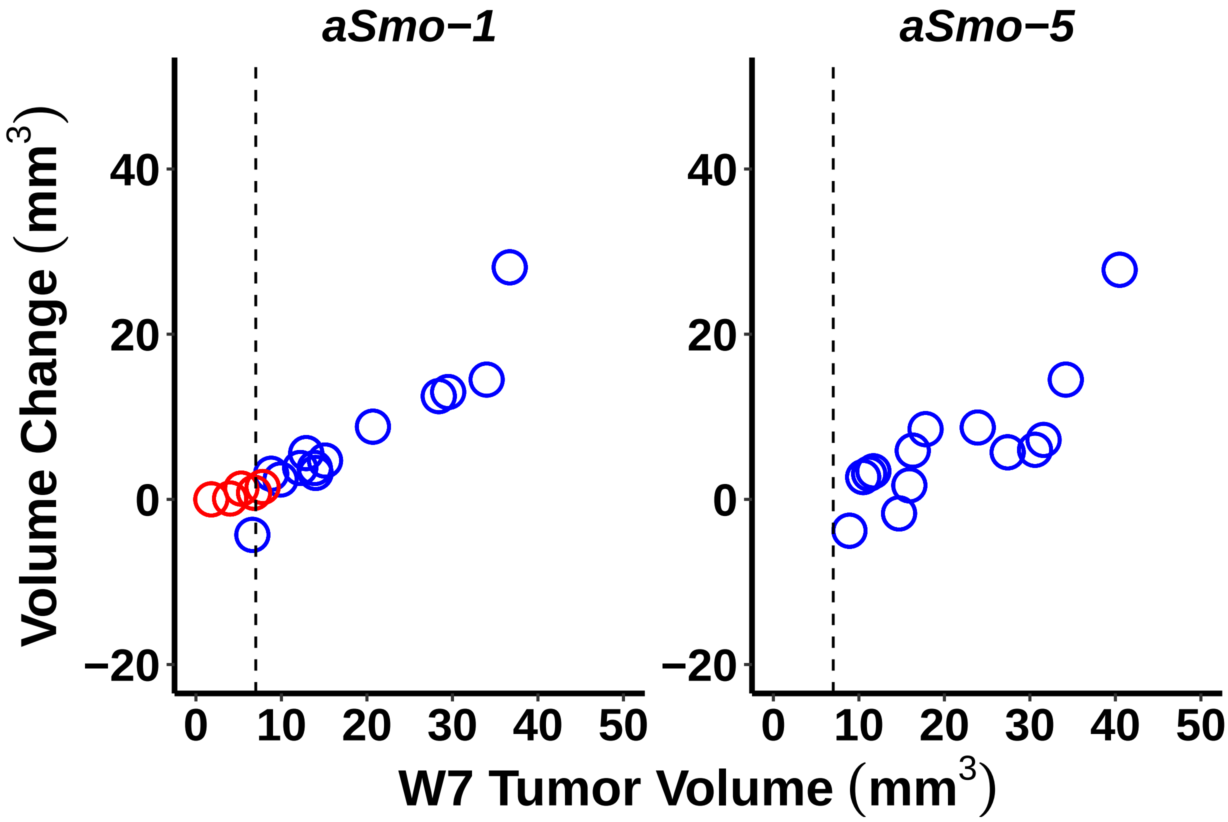
LDA criteria produce similar results to empirical classification

```
pplxreg <- predict(object = reg, newdata = plxf)

plxf$LDA_PREDICTION = pplxreg$class
```

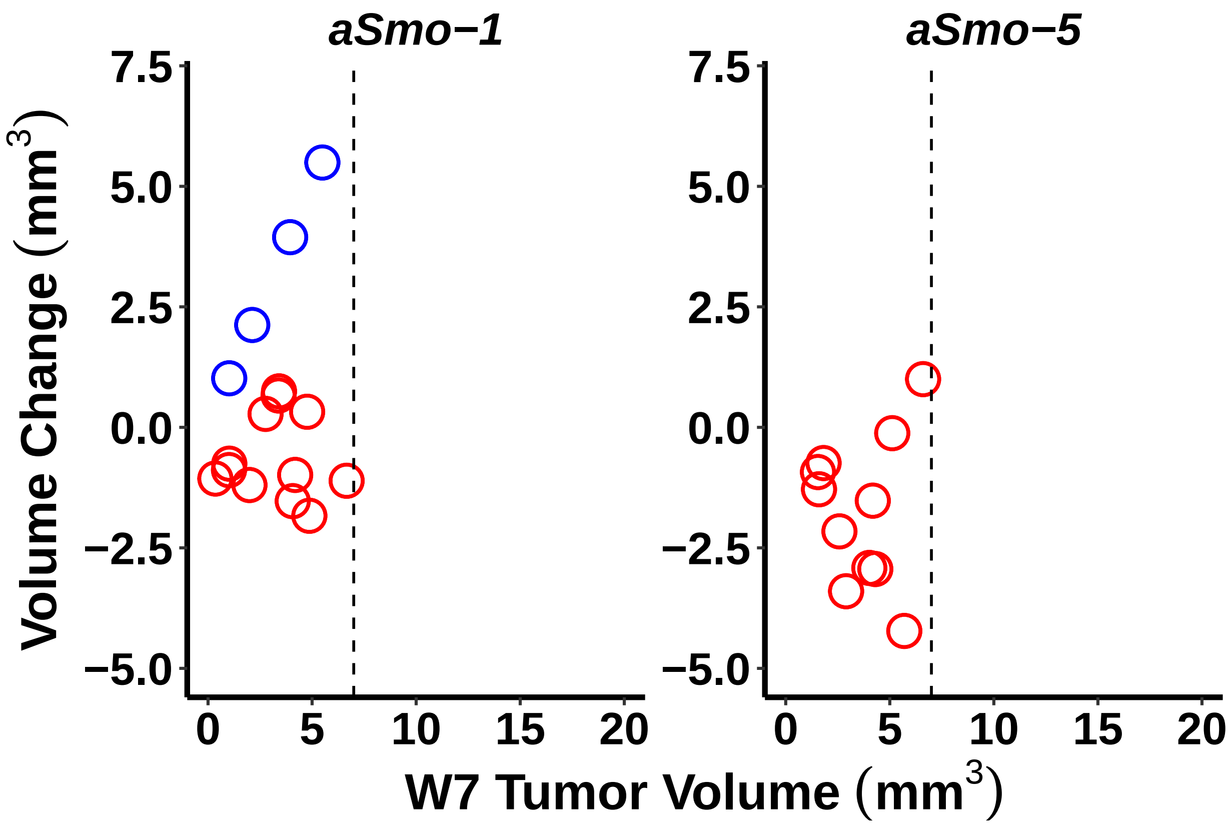
```
plxregplot.CTL <- plxf %>% filter(TREATMENT == "CTL") %>%
  ggplot() +
  aes(x = W7VOLUMES, y = VOLUMECHANGE, color = LDA_PREDICTION) +
  geom_point(size = 5,
  shape = 21,
  stroke = 1) +
  plottheme +
  theme(legend.position = "none") +
  scale_color_manual("LDA Prediction", values = c("red", "blue")) +
  scale_shape("Outcome") +
  guides(color = guide_legend(order = 2),
  shape = guide_legend(order = 1)) +
  xlab(bquote(bold('W7 Tumor Volume' ~ (mm ^ 3)))) +
  ylab(bquote(bold('Volume Change' ~ (mm ^ 3)))) +
  facet_wrap( ~ MODEL,
  labeller = labeller(MODEL = modellabels),
  scales = "free") +
  scale_x_continuous(limits = c(0, 50)) +
  scale_y_continuous(limits = c(-20, 50)) +
  theme(strip.text = element_text(face = "bold.italic")) +
  geom_vline(xintercept = 7, linetype = "dashed")



  plxregplot.CTL
```



```
plxregplot.PLX <- plxf %>% filter(TREATMENT == "PLX") %>%
  ggplot() +
  aes(x = W7VOLUMES, y = VOLUMECHANGE, color = LDA_PREDICTION) +
  geom_point(size = 5,
  shape = 21,
  stroke = 1) +
  plottheme +
  theme(legend.position = "none") +
  scale_color_manual("LDA Prediction", values = c("red", "blue")) +
  scale_shape("Outcome") +
  guides(color = guide_legend(order = 2),
  shape = guide_legend(order = 1)) +
  xlab(bquote(bold('W7 Tumor Volume' ~ (mm ^ 3)))) +
  ylab(bquote(bold('Volume Change' ~ (mm ^ 3)))) +
  facet_wrap( ~ MODEL,
  labeller = labeller(MODEL = modellabels),
  scales = "free") +
  scale_x_continuous(limits = c(0, 50)) +
  scale_y_continuous(limits = c(-20, 50)) +
  theme(strip.text = element_text(face = "bold.italic")) +
  geom_vline(xintercept = 7, linetype = "dashed")

  plxregplot.PLX
```



```
plxregplot.U <-
  plxf %>% filter(TREATMENT == "U") %>%
  filter(W7VOLUMES != 0, VOLUMECHANGE != 0) %>%
  ggplot() +
  aes(x = W7VOLUMES, y = VOLUMECHANGE, color = LDA_PREDICTION) +
  geom_point(size = 5,
  shape = 21,
  stroke = 1) +
  plottheme +
  theme(legend.position = "none") +
  scale_color_manual("LDA Prediction", values = c("red", "blue")) +
  scale_shape("Outcome") +
  guides(color = guide_legend(order = 2),
  shape = guide_legend(order = 1)) +
  xlab(bquote(bold('W7 Tumor Volume' ~ (mm ^ 3)))) +
  ylab(bquote(bold('Volume Change' ~ (mm ^ 3)))) +
  facet_wrap( ~ MODEL,
  labeller = labeller(MODEL = modellabels),
  scales = "free") +
  scale_x_continuous(limits = c(0, 20)) +
  scale_y_continuous(limits = c(-5, 7)) +
  theme(strip.text = element_text(face = "bold.italic")) +
  geom_vline(xintercept = 7, linetype = "dashed")

  plxregplot.U
```



Effects of treatment on endpoint tumor volume after LDA reclassification

```r
plxldamodel <-
  plxf %>% filter(TREATMENT != "U") %>%
  filter(CLASSES != "falsely included") %>% filter(LDA_PREDICTION !=
  "Non-progressing") %>% filter(TIMEPOINT == 5 |
  TIMEPOINT == 6) %>% lmerTest::lmer(FINALVOLUME ~ W7VOLUMES + MODEL:TREATMENT + (1 |
  MOUSEID) + 0, data = .)

  plxldamodel %>% summary(., ddf = "Kenward-Roger")
```

```
## Linear mixed model fit by REML. t-tests use Kenward-Roger's method [
## lmerModLmerTest]
## Formula: FINALVOLUME ~ W7VOLUMES + MODEL:TREATMENT + (1 | MOUSEID) + 0
##    Data: .
##
## REML criterion at convergence: 463.4
##
## Scaled residuals:
##     Min      1Q  Median      3Q     Max
## -1.2516 -0.6056 -0.1824  0.1160  2.7621
##
## Random effects:
##  Groups   Name        Variance Std.Dev.
##  MOUSEID  (Intercept) 100.7    10.04
##  Residual             666.8    25.82
## Number of obs: 52, groups:  MOUSEID, 29
##
## Fixed effects:
##                          Estimate Std. Error       df t value Pr(>|t|)
## W7VOLUMES                  1.4284     0.4432  45.4212   3.223  0.00235 **
## MODELSM01:TREATMENTCTL     2.5058    11.3055  37.7510   0.222  0.82579
## MODELSM05:TREATMENTCTL     8.3323    11.5359  34.6293   0.722  0.47497
## MODELSM01:TREATMENTPLX    -1.1603    11.5989  34.0899  -0.100  0.92090
## MODELSM05:TREATMENTPLX   -11.8751    12.2216  34.3714  -0.972  0.33801
## ---
## Signif. codes:  0 '***' 0.001 '**' 0.01 '*' 0.05 '.' 0.1 ' ' 1
##
## Correlation of Fixed Effects:
##                        W7VOLU MODELSM01:TREATMENTC MODELSM05:TREATMENTC
## MODELSM01:TREATMENTC -0.684
## MODELSM05:TREATMENTC -0.665  0.455
## MODELSM01:TREATMENTP -0.696  0.476                0.463
## MODELSM05:TREATMENTP -0.750  0.513                0.499
##                        MODELSM01:TREATMENTP
## MODELSM01:TREATMENTC
## MODELSM05:TREATMENTC
## MODELSM01:TREATMENTP
## MODELSM05:TREATMENTP  0.522
```

Statistical power calculations after LDA reclassification

```
plxldapwr <- data.frame(
  n1 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  n2 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  M1 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>%
  colMeans(., na.rm = TRUE),
  M2 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>%
  colMeans(., na.rm = TRUE),
  S1 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>%
  sapply(., sd, na.rm = TRUE),
  S2 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>%
  sapply(., sd, na.rm = TRUE)
  )

  plxldapwr$d = (plxldapwr$M1 - plxldapwr$M2) / sqrt(((plxldapwr$S1 ^ 2) +
  (plxldapwr$S2 ^ 2)) / 2)

  print("Post-hoc power calculation based on calculated effect size after LDA")
```

```
## [1] "Post-hoc power calculation based on calculated effect size after LDA"
```

```
  pwr.t2n.test(plxldapwr$n1   ,
  plxldapwr$n2 ,
  plxldapwr$d,
  sig.level = 0.05,
  power = )
```

```
##
##      t test power calculation
##
##              n1 = 27
##              n2 = 25
##               d = 0.3140286
##       sig.level = 0.05
##           power = 0.1986766
##     alternative = two.sided
```

```r
  print("a priori sample size calculation based on calculated effect size after LDA")
```

```
## [1] "a priori sample size calculation based on calculated effect size after LDA"
```

```r
  pwr.t.test(n = NULL,
  plxldapwr$d,
  sig.level = 0.05,
  power = 0.80)
```

```
##
##      Two-sample t test power calculation
##
##               n = 160.1496
##               d = 0.3140286
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```r
plxldapwr <- plxldapwr %>% mutate(
  n1.SM01 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM01") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  n2.SM01 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM01") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  M1.SM01 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM01") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% colMeans(., na.rm = TRUE),
  M2.SM01 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM01") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% colMeans(., na.rm = TRUE),
  S1.SM01 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM01")  %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% sapply(., sd, na.rm = TRUE),
  S2.SM01 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM01")  %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% sapply(., sd, na.rm = TRUE)
  )

  plxldapwr <-
  plxldapwr %>% mutate(d.SM01 = (plxldapwr$M1.SM01 - plxldapwr$M2.SM01) /
  sqrt(((plxldapwr$S1.SM01 ^ 2) + (plxldapwr$S2.SM01 ^ 2)
  ) / 2))


  print("Post-hoc power calculation based on calculated effect size")
```

```
## [1] "Post-hoc power calculation based on calculated effect size"
```

```
  pwr.t2n.test(
  plxldapwr$n1.SM01  ,
  plxldapwr$n2.SM01 ,
  plxldapwr$d.SM01,
  sig.level = 0.05,
  power = NULL
  )
```

```
##
##      t test power calculation
##
##              n1 = 13
##              n2 = 13
##               d = 0.08953191
##       sig.level = 0.05
##           power = 0.05552609
##     alternative = two.sided
```

```
  print("a priori sample size calculation based on calculated effect size")
```

```
## [1] "a priori sample size calculation based on calculated effect size"
```

```
  pwr.t.test(n = NULL,
  plxldapwr$d.SM01,
  sig.level = 0.05,
  power = 0.80)
```

```
##
##      Two-sample t test power calculation
##
##               n = 1959.268
##               d = 0.08953191
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```

```
plxldapwr <- plxldapwr %>% mutate(
  n1.SM05 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM05") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  n2.SM05 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM05") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% nrow(),
  M1.SM05 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM05") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% colMeans(., na.rm = TRUE),
  M2.SM05 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM05") %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% colMeans(., na.rm = TRUE),
  S1.SM05 = plxf %>%
  filter(TREATMENT == "PLX", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM05")  %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% sapply(., sd, na.rm = TRUE),
  S2.SM05 = plxf %>%
  filter(TREATMENT == "CTL", LDA_PREDICTION != "Non-progressing") %>%
  filter(MODEL == "SM05")  %>% distinct(TUMORID, .keep_all = TRUE) %>%
  ungroup() %>% select(FINALVOLUME) %>% sapply(., sd, na.rm = TRUE)
  )

  plxldapwr <-
  plxldapwr %>% mutate(d.SM05 = (plxldapwr$M1.SM05 - plxldapwr$M2.SM05) /
  sqrt(((plxldapwr$S1.SM05 ^ 2) + (plxldapwr$S2.SM05 ^ 2)
  ) / 2))


  print("Post-hoc power calculation based on calculated effect size")
```

```
## [1] "Post-hoc power calculation based on calculated effect size"
```

```
pwr.t2n.test(
plxldapwr$n1.SM05  ,
plxldapwr$n2.SM05 ,
plxldapwr$d.SM05,
sig.level = 0.05,
power = NULL
)
```

```
##
##      t test power calculation
##
##              n1 = 14
##              n2 = 12
##               d = 0.5239419
##       sig.level = 0.05
##           power = 0.2485027
##     alternative = two.sided
```

```r
print("a priori sample size calculation based on calculated effect size")
```

```
## [1] "a priori sample size calculation based on calculated effect size"
```

```r
pwr.t.test(n = NULL,
plxldapwr$d.SM05,
sig.level = 0.05,
power = 0.80)
```

```
##
##      Two-sample t test power calculation
##
##               n = 58.15957
##               d = 0.5239419
##       sig.level = 0.05
##           power = 0.8
##     alternative = two.sided
##
## NOTE: n is number in *each* group
```