BRCA 1/2 Missense Analyses

Tanya L. Hoskin, Steven N. Hart, PhD

April 20, 2017

Contents

Necessary packages

```
#library(knitr)
library("OptimalCutpoints")
library("randomForest")
library("ROCR")
library('e1071')
library('ggplot2')
library('gridExtra')
```

Read in the data sets and create subsets

```
##################################################
# Change file path as appropriate before running#
```

```
################################################

brca2 <- read.csv("./sources/brca2_final.tsv",header=T,sep="\t")
brca1 <- read.csv("./sources/brca1_final.tsv",header=T,sep="\t")

previous_reports <- rbind(brca2,brca1)

#Update 2/2/18 Found out 5 variants were incorrect
previous_reports[which(previous_reports$CAVA_AA=="E2847K"),"HDR_Classification"]=="Intermediate"
```

```
## [1] FALSE
```

```
previous_reports[which(previous_reports$CAVA_AA=="E2847K"),"hdr_del"]==0
```

```
## [1] FALSE
```

```
# E3002D did not change its classification

previous_reports[which(previous_reports$CAVA_AA=="F2562L"),"HDR_Classification"]=="Intermediate"
```

```
## [1] FALSE
```

```
previous_reports[which(previous_reports$CAVA_AA=="F2562L"),"hdr_del"]==0
```

```
## [1] FALSE
```

```
previous_reports[which(previous_reports$CAVA_AA=="G2812E"),"HDR_Classification"]=="Deleterious"
```

```
## [1] FALSE
```

```
previous_reports[which(previous_reports$CAVA_AA=="F2562L"),"hdr_del"]==1
```

```
## [1] TRUE
```

```
# R3007G did not change its classification

table(previous_reports$CAVA_GENE,previous_reports$HDR_Classification)
```

```
##
##         Deleterious Intermediate Neutral
##   BRCA2          71           21     115
##   BRCA1          61           21     166
```

```
brca2$Reported<-NULL
brca1$Reported<-NULL
#Make a version of data with no missings on models for randomForest()
vus.brca2 <- subset(brca2,!is.na(brca2$MutationassessorScore))
vus.brca1 <- subset(brca1,(!is.na(brca1$MutationassessorScore)&!is.na(brca1$a_gvgd_prior)))

#All BRCA1/2 variants without functional data
all2 <- read.csv("./sources/brca2_other.csv")
all2$Reported <- NULL
all1 <- read.csv("./sources/brca1_other.csv")
all1$Reported <- NULL
```

Code binary variables to report model performance at published cutpoints

```
stack <- rbind(brca2,brca1)
stack$fathmmrank_pred         <- ifelse(stack$FathmmConvertedRankscore>0.8133,1,0)
```

```
stack$fathmm_pred              <- ifelse(stack$FathmmScore<(1.5*-1),1,0)
stack$metalrrank_pred          <- ifelse(stack$MetalrRankscore>0.8111,1,0)
stack$metalrscore_pred         <- ifelse(stack$MetalrScore>0.5,1,0)
stack$metasvmrank_pred         <- ifelse(stack$MetasvmRankscore>0.8227,1,0)
stack$metasvmscore_pred        <- ifelse(stack$MetasvmScore>0,1,0)
stack$mutationtaster_pred      <- ifelse(stack$MutationtasterScore>0.5,1,0)
stack$mutationtasterrank_pred <- ifelse(stack$MutationtasterConvertedRankscore>0.3171,1,0)
stack$pp2hdivrank_pred         <- ifelse(stack$Polyphen2HdivRankscore>0.3428,1,0)
stack$pp2hdiv_pred             <- ifelse(stack$Polyphen2HdivScore>0.453,1,0)
stack$pp2hvarrank_pred         <- ifelse(stack$Polyphen2HvarRankscore>0.442,1,0)
stack$pp2hvar_pred             <- ifelse(stack$Polyphen2HvarScore>0.447,1,0)
stack$proveanrank_pred         <- ifelse(stack$ProveanConvertedRankscore>0.543,1,0)
stack$provean_pred             <- ifelse(stack$ProveanScore<(2.5*-1),1,0)
stack$siftrank_pred            <- ifelse(stack$SiftConvertedRankscore>0.395,1,0)
stack$sift_pred                <- ifelse(stack$SiftScore<0.05,1,0)

brca2.pub <- subset(stack,X_CHROM=="13")
brca1.pub <- subset(stack,X_CHROM=="17")
```

Table S3

Performance characteristics of commonly used software with default thresholds

```
options(digits=3)

sensspec <- function (pred,out){
  t <- table(pred,out,useNA="no")
  m2 <- margin.table(t,2)
  tp <- t[2,2]
  tn <- t[1,1]
  nd <- m2[2]
  nh <- m2[1]
  fn <- nd - tp
  fp <- nh - tn
  sens <- tp/nd
  spec <- tn/nh
  mcc <- (tp*tn-fp*fn)/sqrt((tp+fp)*(tp+fn)*(tn+fp)*(tn+fn))
  est <- cbind(tn,fn,fp,tp,sens,spec,mcc)
}

supp3names <- c("FathmmConvertedRankscore",
                "FathmmScore",
                "MetalrRankscore",
                "MetalrScore",
                "MetasvmRankscore",
                "MetasvmScore",
                "MutationtasterConvertedRankscore",
                "MutationtasterScore",
                "Polyphen2HdivRankscore",
                "Polyphen2HdivScore",
                "Polyphen2HvarRankscore",
```

```
                  "Polyphen2HvarScore",
                  "ProveanConvertedRankscore",
                  "ProveanScore",
                  "SiftConvertedRankscore",
                  "SiftScore")

supp3.names <- matrix(supp3names,nrow=length(supp3names),ncol=1)

e1.1  <- sensspec(brca1.pub$fathmmrank_pred,brca1.pub$hdr_del)
e1.2  <- sensspec(brca1.pub$fathmm_pred,brca1.pub$hdr_del)
e1.3  <- sensspec(brca1.pub$metalrrank_pred,brca1.pub$hdr_del)
e1.4  <- sensspec(brca1.pub$metalrscore_pred,brca1.pub$hdr_del)
e1.5  <- sensspec(brca1.pub$metasvmrank_pred,brca1.pub$hdr_del)
e1.6  <- sensspec(brca1.pub$metasvmscore_pred,brca1.pub$hdr_del)
e1.7  <- sensspec(brca1.pub$mutationtasterrank_pred,brca1.pub$hdr_del)
e1.8  <- sensspec(brca1.pub$mutationtaster_pred,brca1.pub$hdr_del)
e1.9  <- sensspec(brca1.pub$pp2hdivrank_pred,brca1.pub$hdr_del)
e1.10 <- sensspec(brca1.pub$pp2hdiv_pred,brca1.pub$hdr_del)
e1.11 <- sensspec(brca1.pub$pp2hvarrank_pred,brca1.pub$hdr_del)
e1.12 <- sensspec(brca1.pub$pp2hvar_pred,brca1.pub$hdr_del)
e1.13 <- sensspec(brca1.pub$proveanrank_pred,brca1.pub$hdr_del)
e1.14 <- sensspec(brca1.pub$provean_pred,brca1.pub$hdr_del)
e1.15 <- sensspec(brca1.pub$siftrank_pred,brca1.pub$hdr_del)
e1.16 <- sensspec(brca1.pub$sift_pred,brca1.pub$hdr_del)

brca1.published <- rbind(e1.1,e1.2,e1.3,e1.4,e1.5,e1.6,e1.7,e1.8,
                         e1.9,e1.10,e1.11,e1.12,e1.13,e1.14,e1.15,e1.16)

brca1.published. <- data.frame(supp3.names,brca1.published)

e2.1  <- sensspec(brca2.pub$fathmmrank_pred,brca2.pub$hdr_del)
e2.2  <- sensspec(brca2.pub$fathmm_pred,brca2.pub$hdr_del)
e2.3  <- sensspec(brca2.pub$metalrrank_pred,brca2.pub$hdr_del)
e2.4  <- sensspec(brca2.pub$metalrscore_pred,brca2.pub$hdr_del)
e2.5  <- sensspec(brca2.pub$metasvmrank_pred,brca2.pub$hdr_del)
e2.6  <- sensspec(brca2.pub$metasvmscore_pred,brca2.pub$hdr_del)
e2.7  <- sensspec(brca2.pub$mutationtasterrank_pred,brca2.pub$hdr_del)
e2.8  <- sensspec(brca2.pub$mutationtaster_pred,brca2.pub$hdr_del)
e2.9  <- sensspec(brca2.pub$pp2hdivrank_pred,brca2.pub$hdr_del)
e2.10 <- sensspec(brca2.pub$pp2hdiv_pred,brca2.pub$hdr_del)
e2.11 <- sensspec(brca2.pub$pp2hvarrank_pred,brca2.pub$hdr_del)
e2.12 <- sensspec(brca2.pub$pp2hvar_pred,brca2.pub$hdr_del)
e2.13 <- sensspec(brca2.pub$proveanrank_pred,brca2.pub$hdr_del)
e2.14 <- sensspec(brca2.pub$provean_pred,brca2.pub$hdr_del)
e2.15 <- sensspec(brca2.pub$siftrank_pred,brca2.pub$hdr_del)
e2.16 <- sensspec(brca2.pub$sift_pred,brca2.pub$hdr_del)

brca2.published <- rbind(e2.1,e2.2,e2.3,e2.4,e2.5,e2.6,e2.7,e2.8,
                         e2.9,e2.10,e2.11,e2.12,e2.13,e2.14,e2.15,e2.16)

brca2.published. <- data.frame(supp3.names,brca2.published)

supp_table3 <- rbind(brca1.published.,brca2.published.)
```

For BRCA1

For BRCA2

```
# BRCA2
brca2.published.$Gene="BRCA2"
ST3 <- rbind(ST3,brca2.published.)
### Add default thresholds
ST3$cutoff<-1
ST3$cutoff[which(ST3$supp3.names=="FathmmConvertedRankscore")] <- 0.8133
ST3$cutoff[which(ST3$supp3.names=="MetalrRankscore")] <- 0.8111
ST3$cutoff[which(ST3$supp3.names=="MetalrScore")] <- 0.5
ST3$cutoff[which(ST3$supp3.names=="MetasvmRankscore")] <- 0.8227
ST3$cutoff[which(ST3$supp3.names=="MetasvmScore")] <- 0
ST3$cutoff[which(ST3$supp3.names=="MutationtasterScore")] <- 0.5
ST3$cutoff[which(ST3$supp3.names=="MutationtasterConvertedRankscore") ]<- 0.3171
ST3$cutoff[which(ST3$supp3.names=="Polyphen2HdivRankscore")] <- 0.3428
ST3$cutoff[which(ST3$supp3.names=="Polyphen2HdivScore")] <- 0.453
ST3$cutoff[which(ST3$supp3.names=="Polyphen2HvarRankscore")] <- 0.442
ST3$cutoff[which(ST3$supp3.names=="Polyphen2HvarScore")] <- 0.447
ST3$cutoff[which(ST3$supp3.names=="ProveanConvertedRankscore")] <- 0.543
ST3$cutoff[which(ST3$supp3.names=="SiftConvertedRankscore")] <- 0.395
ST3$cutoff[which(ST3$supp3.names=="SiftScore")] <- 0.05
ST3$cutoff[which(ST3$supp3.names=="FathmmScore")] <- (1.5*-1)
ST3$cutoff[which(ST3$supp3.names=="ProveanScore")] <- (2.5*-1)



write.table(file="tables/Table_S3.tsv",ST3,row.names = F,sep="\t")
knitr::kable(ST3)
```

supp3.names	tn	fn	fp	tp	sens	spec	mcc	Gene	cutoff
FathmmConvertedRankscore	30	8	147	51	0.864	0.169	0.040	BRCA1	0.813
FathmmScore	30	8	147	51	0.864	0.169	0.040	BRCA1	-1.500
MetalrRankscore	82	1	95	58	0.983	0.463	0.405	BRCA1	0.811
MetalrScore	82	1	95	58	0.983	0.463	0.405	BRCA1	0.500
MetasvmRankscore	118	2	59	57	0.966	0.667	0.548	BRCA1	0.823
MetasvmScore	118	2	59	57	0.966	0.667	0.548	BRCA1	0.000
MutationtasterConvertedRankscore	82	0	95	59	1.000	0.463	0.421	BRCA1	0.317
MutationtasterScore	1	0	176	59	1.000	0.006	0.038	BRCA1	0.500
Polyphen2HdivRankscore	48	0	129	59	1.000	0.271	0.292	BRCA1	0.343
Polyphen2HdivScore	48	0	129	59	1.000	0.271	0.292	BRCA1	0.453
Polyphen2HvarRankscore	64	0	113	59	1.000	0.362	0.352	BRCA1	0.442
Polyphen2HvarScore	64	0	113	59	1.000	0.362	0.352	BRCA1	0.447
ProveanConvertedRankscore	95	3	82	56	0.949	0.537	0.427	BRCA1	0.543
ProveanScore	177	58	0	1	0.017	1.000	0.113	BRCA1	-2.500
SiftConvertedRankscore	50	1	127	58	0.983	0.282	0.279	BRCA1	0.395
SiftScore	50	1	127	58	0.983	0.282	0.279	BRCA1	0.050
FathmmConvertedRankscore	73	7	63	64	0.901	0.537	0.427	BRCA2	0.813
FathmmScore	73	7	63	64	0.901	0.537	0.427	BRCA2	-1.500
MetalrRankscore	49	1	87	70	0.986	0.360	0.384	BRCA2	0.811
MetalrScore	49	1	87	70	0.986	0.360	0.384	BRCA2	0.500
MetasvmRankscore	53	1	83	70	0.986	0.390	0.406	BRCA2	0.823
MetasvmScore	53	1	83	70	0.986	0.390	0.406	BRCA2	0.000
MutationtasterConvertedRankscore	32	1	104	70	0.986	0.235	0.287	BRCA2	0.317

supp3.names	tn	fn	fp	tp	sens	spec	mcc	Gene	cutoff
MutationtasterScore	1	0	135	71	1.000	0.007	0.050	BRCA2	0.500
Polyphen2HdivRankscore	16	1	119	70	0.986	0.119	0.180	BRCA2	0.343
Polyphen2HdivScore	16	1	119	70	0.986	0.119	0.180	BRCA2	0.453
Polyphen2HvarRankscore	30	1	105	70	0.986	0.222	0.277	BRCA2	0.442
Polyphen2HvarScore	30	1	105	70	0.986	0.222	0.277	BRCA2	0.447
ProveanConvertedRankscore	126	60	10	11	0.155	0.926	0.128	BRCA2	0.543
ProveanScore	126	60	10	11	0.155	0.926	0.128	BRCA2	-2.500
SiftConvertedRankscore	27	0	109	71	1.000	0.199	0.280	BRCA2	0.395
SiftScore	27	0	109	71	1.000	0.199	0.280	BRCA2	0.050

The Making of supplemental Table 4 for existing models BRCA2, Scenario 1

```
#For these models, higher scores are worse
varnames1 <-
  c("a_gvgd_prior","CaddPhred","CaddRaw","CaddRawRankscore",
              "CAROL_score","Condel_score","DannRankscore","DannScore",
              "FathmmConvertedRankscore","GerpNr","GerpRs",
              "GerpRsRankscore","LrtConvertedRankscore","MetalrRankscore",
              "MetalrScore","MetasvmRankscore","MetasvmScore","MutationassessorRankscore",
              "MutationassessorScore","MutationtasterConvertedRankscore",
              "MutationtasterScore","PERCH","PERCH_noMAF","Phastcons20wayMammalianRankscore",
              "Phastcons7wayVertebrateRankscore","Phylop20wayMammalianRankscore",
              "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HdivScore",
              "Polyphen2HvarRankscore","Polyphen2HvarScore","ProveanConvertedRankscore",
              "SiftConvertedRankscore","Siphy29wayLogoddsRankscore",
              "Vest3Rankscore","Vest3Score")

# For these models, lower scores are worse
varnames2 <- c("FathmmScore","LrtScore","ProveanScore","SiftScore")

# These models have two optimal cutpoints by SpEqualSe criterion for one or both genes
# Special handling outside loop b/c structure of optimal.cutpoints output list is different
varnames3 <- c("FathmmConvertedRankscore","FathmmScore","LrtConvertedRankscore",
               "LrtScore","ProveanScore","DannScore","CaddRaw","CaddRawRankscore")

# BRCA2 Variants
# Outcome = Deleterious vs Intermediate/Neutral
# Find optimal cutpoint for each model

fn <- matrix(NA,nrow=length(varnames1),ncol=1)
fp <- matrix(NA,nrow=length(varnames1),ncol=1)
nh <- matrix(NA,nrow=length(varnames1),ncol=1)
nd <- matrix(NA,nrow=length(varnames1),ncol=1)
tp <- matrix(NA,nrow=length(varnames1),ncol=1)
tn <- matrix(NA,nrow=length(varnames1),ncol=1)
mcc <- matrix(NA,nrow=length(varnames1),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames1),ncol=1)
auc <- matrix(NA,nrow=length(varnames1),ncol=3)
sens <- matrix(NA,nrow=length(varnames1),ncol=3)
spec <- matrix(NA,nrow=length(varnames1),ncol=3)

for (i in 1:length(varnames1)) {
```

```
v2a.i <- optimal.cutpoints(X=varnames1[i],status="hdr_del",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                     ci.fit=TRUE,data=brca2)

auc[i,] <- v2a.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v2a.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v2a.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca2a.models.dir1 <- data.frame(varnames1,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FALSE
colnames(brca2a.models.dir1) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                  "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                  "TP","TN","mcc")

}

fn <- matrix(NA,nrow=4,ncol=1)
fp <- matrix(NA,nrow=4,ncol=1)
nh <- matrix(NA,nrow=4,ncol=1)
nd <- matrix(NA,nrow=4,ncol=1)
tp <- matrix(NA,nrow=4,ncol=1)
tn <- matrix(NA,nrow=4,ncol=1)
mcc <- matrix(NA,nrow=4,ncol=1)
cutoff <- matrix(NA,nrow=4,ncol=1)
auc <- matrix(NA,nrow=4,ncol=3)
sens <- matrix(NA,nrow=4,ncol=3)
spec <- matrix(NA,nrow=4,ncol=3)

for (i in 1:length(varnames2)) {

v2a.i <- optimal.cutpoints(X=varnames2[i],status="hdr_del",direction=">",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                     ci.fit=TRUE,data=brca2)

auc[i,] <- v2a.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v2a.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v2a.i$SpEqualSe[[1]]$measures.acc$n$d
```

```
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]


cutoff[i,1] <- v2a.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                                (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca2a.models.dir2 <- data.frame(varnames2,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FALSE
colnames(brca2a.models.dir2) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                  "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                  "TP","TN","mcc")
}

brca2a.models <- rbind(brca2a.models.dir1,brca2a.models.dir2)
brca2a.models. <- brca2a.models[order(brca2a.models$model),]
brca2a.auc <- brca2a.models[,c("model","auc","auc.lcl","auc.ucl")]
brca2a.auc. <- brca2a.auc[order(brca2a.auc$model),]

ST4_BRCA2_S1 <-brca2a.models.
ST4_BRCA2_S1$Gene="BRCA2"
ST4_BRCA2_S1$Scenario=1
```

BRCA1 Scenario 1

```
# BRCA1 Variants
# Outcome = Deleterious vs Intermediate/Neutral
# Find optimal cutpoint for each model
#These have one optimal cutpoint & higher is worse for BRCA1 Scenario1
varnames1. <- c("a_gvgd_prior","CaddPhred","CaddRaw","CaddRawRankscore",
                "CAROL_score","Condel_score","DannRankscore","DannScore",
                "GerpNr","GerpRs",
                "GerpRsRankscore","MetalrRankscore",
                "MetalrScore","MetasvmRankscore","MetasvmScore","MutationassessorRankscore",
                "MutationassessorScore","MutationtasterConvertedRankscore",
                "MutationtasterScore","PERCH","PERCH_noMAF","Phastcons20wayMammalianRankscore",
                "Phastcons7wayVertebrateRankscore","Phylop20wayMammalianRankscore",
                "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HdivScore",
                "Polyphen2HvarRankscore","Polyphen2HvarScore","ProveanConvertedRankscore",
                "SiftConvertedRankscore","Siphy29wayLogoddsRankscore",
                "Vest3Rankscore","Vest3Score")

#These have one optimal cutpoint & lower is worse for BRCA1 Scenario1
varnames2. <- c("SiftScore")

#These have two optimal cutpoints, pick one, direction = higher worse
varnames3a. <- c("FathmmConvertedRankscore","LrtConvertedRankscore")

#These have two optimal cutpoints, pick one, direction = lower worse
varnames3b. <- c("FathmmScore","LrtScore")
varnames3c. <- c("ProveanScore")

fn <- matrix(NA,nrow=length(varnames1.),ncol=1)
```

```
fp <- matrix(NA,nrow=length(varnames1.),ncol=1)
nh <- matrix(NA,nrow=length(varnames1.),ncol=1)
nd <- matrix(NA,nrow=length(varnames1.),ncol=1)
tp <- matrix(NA,nrow=length(varnames1.),ncol=1)
tn <- matrix(NA,nrow=length(varnames1.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames1.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames1.),ncol=1)
auc <- matrix(NA,nrow=length(varnames1.),ncol=3)
sens <- matrix(NA,nrow=length(varnames1.),ncol=3)
spec <- matrix(NA,nrow=length(varnames1.),ncol=3)

for (i in 1:length(varnames1.)) {

v1a.i <- optimal.cutpoints(X=varnames1.[i],status="hdr_del",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                     ci.fit=TRUE,data=brca1)

auc[i,] <- v1a.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1a.models.dir1 <- data.frame(varnames1.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FALS
colnames(brca1a.models.dir1) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                              "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                              "TP","TN","mcc")

}

fn <- matrix(NA,nrow=1,ncol=1)
fp <- matrix(NA,nrow=1,ncol=1)
nh <- matrix(NA,nrow=1,ncol=1)
nd <- matrix(NA,nrow=1,ncol=1)
tp <- matrix(NA,nrow=1,ncol=1)
tn <- matrix(NA,nrow=1,ncol=1)
mcc <- matrix(NA,nrow=1,ncol=1)
cutoff <- matrix(NA,nrow=1,ncol=1)
auc <- matrix(NA,nrow=1,ncol=3)
sens <- matrix(NA,nrow=1,ncol=3)
spec <- matrix(NA,nrow=1,ncol=3)

for (i in 1:length(varnames2.)) {
```

```
v1a.i <- optimal.cutpoints(X=varnames2.[i],status="hdr_del",direction=">",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                     ci.fit=TRUE,data=brca1)

auc[i,] <- v1a.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                                (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1a.models.dir2 <- data.frame(varnames2.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FALS
colnames(brca1a.models.dir2) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                  "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                  "TP","TN","mcc")

}


####

fn <- matrix(NA,nrow=length(varnames3a.),ncol=1)
fp <- matrix(NA,nrow=length(varnames3a.),ncol=1)
nh <- matrix(NA,nrow=length(varnames3a.),ncol=1)
nd <- matrix(NA,nrow=length(varnames3a.),ncol=1)
tp <- matrix(NA,nrow=length(varnames3a.),ncol=1)
tn <- matrix(NA,nrow=length(varnames3a.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames3a.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames3a.),ncol=1)
auc <- matrix(NA,nrow=length(varnames3a.),ncol=3)
sens <- matrix(NA,nrow=length(varnames3a.),ncol=3)
spec <- matrix(NA,nrow=length(varnames3a.),ncol=3)

for (i in 1:length(varnames3a.)) {

v1a.i <- optimal.cutpoints(X=varnames3a.[i],status="hdr_del",
                           tag.healthy=0,methods=c("SpEqualSe"),
                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                        ci.fit=TRUE,data=brca1)

auc[i,] <- v1a.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1,3,5)]
spec[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1,3,5)]
fn[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
```

```
fp[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$cutoff[1]

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1a.models.dir1. <- data.frame(varnames3a.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = F
colnames(brca1a.models.dir1.) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                              "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                              "TP","TN","mcc")

}


fn <- matrix(NA,nrow=length(varnames3b.),ncol=1)
fp <- matrix(NA,nrow=length(varnames3b.),ncol=1)
nh <- matrix(NA,nrow=length(varnames3b.),ncol=1)
nd <- matrix(NA,nrow=length(varnames3b.),ncol=1)
tp <- matrix(NA,nrow=length(varnames3b.),ncol=1)
tn <- matrix(NA,nrow=length(varnames3b.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames3b.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames3b.),ncol=1)
auc <- matrix(NA,nrow=length(varnames3b.),ncol=3)
sens <- matrix(NA,nrow=length(varnames3b.),ncol=3)
spec <- matrix(NA,nrow=length(varnames3b.),ncol=3)

for (i in 1:length(varnames3b.)) {

v1a.i <- optimal.cutpoints(X=varnames3b.[i],status="hdr_del",direction=">",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                    control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                    ci.fit=TRUE,data=brca1)

auc[i,] <- v1a.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(2,4,6)]
spec[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(2,4,6)]
fn[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FN[2]
fp[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FP[2]
nh[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$cutoff[2]

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                               (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))
```

```
brca1a.models.dir2. <- data.frame(varnames3b.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FA
colnames(brca1a.models.dir2.) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                   "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                   "TP","TN","mcc")

}


fn <- matrix(NA,nrow=length(varnames3c.),ncol=1)
fp <- matrix(NA,nrow=length(varnames3c.),ncol=1)
nh <- matrix(NA,nrow=length(varnames3c.),ncol=1)
nd <- matrix(NA,nrow=length(varnames3c.),ncol=1)
tp <- matrix(NA,nrow=length(varnames3c.),ncol=1)
tn <- matrix(NA,nrow=length(varnames3c.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames3c.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames3c.),ncol=1)
auc <- matrix(NA,nrow=length(varnames3c.),ncol=3)
sens <- matrix(NA,nrow=length(varnames3c.),ncol=3)
spec <- matrix(NA,nrow=length(varnames3c.),ncol=3)

for (i in 1:length(varnames3c.)) {

v1a.i <- optimal.cutpoints(X=varnames3c.[i],status="hdr_del",direction=">",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                    control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                    ci.fit=TRUE,data=brca1)

auc[i,] <- v1a.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1,3,5)]
spec[i,] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1,3,5)]
fn[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1a.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1a.i$SpEqualSe[[1]]$optimal.cutoff$cutoff[1]

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                        (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1a.models.dir2.. <- data.frame(varnames3c.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                   stringsAsFactors = FALSE)
colnames(brca1a.models.dir2..) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                    "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                    "TP","TN","mcc")

}

brca1a.models <- rbind(brca1a.models.dir1,brca1a.models.dir2,brca1a.models.dir1.,
                       brca1a.models.dir2.,brca1a.models.dir2..)
brca1a.models. <- brca1a.models[order(brca1a.models$model),]
```

```
brca1a.auc <- brca1a.models[,c("model","auc","auc.lcl","auc.ucl")]
brca1a.auc. <- brca1a.auc[order(brca1a.auc$model),]

ST4_BRCA1_S1 <- brca1a.models.
ST4_BRCA1_S1$Gene="BRCA1"
ST4_BRCA1_S1$Scenario=1
```

BRCA1, Scenario 2

```
# Scenario 2
# Outcome = Deleterious/Intermediate vs Neutral
# Find optimal cutpoint for each model
varnames1. <- c("a_gvgd_prior","CaddPhred","CaddRaw","CaddRawRankscore",
                "CAROL_score","Condel_score","DannRankscore","DannScore",
                "GerpNr","GerpRs",
                "GerpRsRankscore","MetalrRankscore",
                "MetalrScore","MetasvmRankscore","MetasvmScore","MutationassessorRankscore",
                "MutationassessorScore","MutationtasterConvertedRankscore",
                "MutationtasterScore","PERCH","PERCH_noMAF","Phastcons20wayMammalianRankscore",
                "Phastcons7wayVertebrateRankscore","Phylop20wayMammalianRankscore",
                "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HdivScore",
                "Polyphen2HvarRankscore","Polyphen2HvarScore","ProveanConvertedRankscore",
                "SiftConvertedRankscore","Siphy29wayLogoddsRankscore",
                "Vest3Rankscore","Vest3Score","FathmmConvertedRankscore","LrtConvertedRankscore")

#These have one optimal cutpoint & lower is worse for BRCA1 Scenario1
varnames2. <- c("SiftScore","FathmmScore","LrtScore","ProveanScore")

fn <- matrix(NA,nrow=length(varnames1.),ncol=1)
fp <- matrix(NA,nrow=length(varnames1.),ncol=1)
nh <- matrix(NA,nrow=length(varnames1.),ncol=1)
nd <- matrix(NA,nrow=length(varnames1.),ncol=1)
tp <- matrix(NA,nrow=length(varnames1.),ncol=1)
tn <- matrix(NA,nrow=length(varnames1.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames1.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames1.),ncol=1)
auc <- matrix(NA,nrow=length(varnames1.),ncol=3)
sens <- matrix(NA,nrow=length(varnames1.),ncol=3)
spec <- matrix(NA,nrow=length(varnames1.),ncol=3)

for (i in 1:length(varnames1.)) {

v1b.i <- optimal.cutpoints(X=varnames1.[i],status="hdr_not_neutral",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                     ci.fit=TRUE,data=brca1)

auc[i,] <- v1b.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1b.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1b.i$SpEqualSe[[1]]$measures.acc$n$d
```

```
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1b.models.dir1 <- data.frame(varnames1.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FALS
colnames(brca1b.models.dir1) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                  "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                  "TP","TN","mcc")

}

fn <- matrix(NA,nrow=length(varnames2.),ncol=1)
fp <- matrix(NA,nrow=length(varnames2.),ncol=1)
nh <- matrix(NA,nrow=length(varnames2.),ncol=1)
nd <- matrix(NA,nrow=length(varnames2.),ncol=1)
tp <- matrix(NA,nrow=length(varnames2.),ncol=1)
tn <- matrix(NA,nrow=length(varnames2.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames2.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames2.),ncol=1)
auc <- matrix(NA,nrow=length(varnames2.),ncol=3)
sens <- matrix(NA,nrow=length(varnames2.),ncol=3)
spec <- matrix(NA,nrow=length(varnames2.),ncol=3)

for (i in 1:length(varnames2.)) {

v1b.i <- optimal.cutpoints(X=varnames2.[i],status="hdr_not_neutral",direction=">",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                    control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                    ci.fit=TRUE,data=brca1)

auc[i,] <- v1b.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1b.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1b.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1b.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                               (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1b.models.dir2 <- data.frame(varnames2.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FALS
colnames(brca1b.models.dir2) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                  "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                  "TP","TN","mcc")
```

```
}

brca1b.models <- rbind(brca1b.models.dir1,brca1b.models.dir2)
brca1b.models. <- brca1b.models[order(brca1b.models$model),]
brca1b.auc <- brca1b.models[,c("model","auc","auc.lcl","auc.ucl")]
brca1b.auc. <- brca1b.auc[order(brca1b.auc$model),]


ST4_BRCA1_S2 <- brca1b.models.
ST4_BRCA1_S2$Gene="BRCA1"
ST4_BRCA1_S2$Scenario=2
```

BRCA2 Scenario 2

```
# BRCA2 Variants
# Outcome = Deleterious/Intermediate vs Neutral
# Find optimal cutpoint for each model
#These have one optimal cutpoint & higher is worse for BRCA2 Scenario2
varnames1. <- c("a_gvgd_prior","CaddPhred",
              "CAROL_score","Condel_score","DannRankscore",
              "GerpNr","GerpRs","FathmmConvertedRankscore","LrtConvertedRankscore",
              "GerpRsRankscore","MetalrRankscore",
              "MetalrScore","MetasvmRankscore","MetasvmScore","MutationassessorRankscore",
              "MutationassessorScore","MutationtasterConvertedRankscore",
              "MutationtasterScore","PERCH","PERCH_noMAF","Phastcons20wayMammalianRankscore",
              "Phastcons7wayVertebrateRankscore","Phylop20wayMammalianRankscore",
              "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HdivScore",
              "Polyphen2HvarRankscore","Polyphen2HvarScore","ProveanConvertedRankscore",
              "SiftConvertedRankscore","Siphy29wayLogoddsRankscore",
              "Vest3Rankscore","Vest3Score")

#These have one optimal cutpoint & lower is worse for BRCA2 Scenario2
varnames2. <- c("FathmmScore","LrtScore","ProveanScore","SiftScore")

#These have two optimal cutpoints, pick one, direction = higher worse
varnames3a. <- c("CaddRaw","CaddRawRankscore","DannScore")

fn <- matrix(NA,nrow=length(varnames1.),ncol=1)
fp <- matrix(NA,nrow=length(varnames1.),ncol=1)
nh <- matrix(NA,nrow=length(varnames1.),ncol=1)
nd <- matrix(NA,nrow=length(varnames1.),ncol=1)
tp <- matrix(NA,nrow=length(varnames1.),ncol=1)
tn <- matrix(NA,nrow=length(varnames1.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames1.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames1.),ncol=1)
auc <- matrix(NA,nrow=length(varnames1.),ncol=3)
sens <- matrix(NA,nrow=length(varnames1.),ncol=3)
spec <- matrix(NA,nrow=length(varnames1.),ncol=3)

for (i in 1:length(varnames1.)) {

v2b.i <- optimal.cutpoints(X=varnames1.[i],status="hdr_not_neutral",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
```

```
                                        ci.fit=TRUE,data=brca2)

auc[i,] <- v2b.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v2b.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v2b.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca2b.models.dir1 <- data.frame(varnames1.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                 stringsAsFactors = FALSE)
colnames(brca2b.models.dir1) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                  "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                  "TP","TN","mcc")

}

fn <- matrix(NA,nrow=length(varnames2.),ncol=1)
fp <- matrix(NA,nrow=length(varnames2.),ncol=1)
nh <- matrix(NA,nrow=length(varnames2.),ncol=1)
nd <- matrix(NA,nrow=length(varnames2.),ncol=1)
tp <- matrix(NA,nrow=length(varnames2.),ncol=1)
tn <- matrix(NA,nrow=length(varnames2.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames2.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames2.),ncol=1)
auc <- matrix(NA,nrow=length(varnames2.),ncol=3)
sens <- matrix(NA,nrow=length(varnames2.),ncol=3)
spec <- matrix(NA,nrow=length(varnames2.),ncol=3)

for (i in 1:length(varnames2.)) {

v2b.i <- optimal.cutpoints(X=varnames2.[i],status="hdr_not_neutral",direction=">",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                    control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                    ci.fit=TRUE,data=brca2)

auc[i,] <- v2b.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v2b.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v2b.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]
```

```
cutoff[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                    (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca2b.models.dir2 <- data.frame(varnames2.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,stringsAsFactors = FALS
colnames(brca2b.models.dir2) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                  "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                  "TP","TN","mcc")

}


####

fn <- matrix(NA,nrow=length(varnames3a.),ncol=1)
fp <- matrix(NA,nrow=length(varnames3a.),ncol=1)
nh <- matrix(NA,nrow=length(varnames3a.),ncol=1)
nd <- matrix(NA,nrow=length(varnames3a.),ncol=1)
tp <- matrix(NA,nrow=length(varnames3a.),ncol=1)
tn <- matrix(NA,nrow=length(varnames3a.),ncol=1)
mcc <- matrix(NA,nrow=length(varnames3a.),ncol=1)
cutoff <- matrix(NA,nrow=length(varnames3a.),ncol=1)
auc <- matrix(NA,nrow=length(varnames3a.),ncol=3)
sens <- matrix(NA,nrow=length(varnames3a.),ncol=3)
spec <- matrix(NA,nrow=length(varnames3a.),ncol=3)

for (i in 1:length(varnames3a.)) {

v2b.i <- optimal.cutpoints(X=varnames3a.[i],status="hdr_not_neutral",
                           tag.healthy=0,methods=c("SpEqualSe"),
                                    control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                    ci.fit=TRUE,data=brca2)

auc[i,] <- v2b.i$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$Se[c(2,4,6)]
spec[i,] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$Sp[c(2,4,6)]
fn[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$FN[2]
fp[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$FP[2]
nh[i,1] <- v2b.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v2b.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v2b.i$SpEqualSe[[1]]$optimal.cutoff$cutoff[2]

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca2b.models.dir1. <- data.frame(varnames3a.,cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                  stringsAsFactors = FALSE)
colnames(brca2b.models.dir1.) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                               "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
```

```
                              "TP","TN","mcc")

}

brca2b.models <- rbind(brca2b.models.dir1,brca2b.models.dir2,brca2b.models.dir1.)
brca2b.models. <- brca2b.models[order(brca2b.models$model),]
brca2b.auc <- brca2b.models[,c("model","auc","auc.lcl","auc.ucl")]
brca2b.auc. <- brca2b.auc[order(brca2b.auc$model),]

ST4_BRCA2_S2 <- brca2b.models.
ST4_BRCA2_S2$Gene="BRCA2"
ST4_BRCA2_S2$Scenario=2
```

Input list of individual predictive model names for use in plots

```
modelnames<-c(
"AlignGVGDPrior",
"CADDPhred",
"CADDRaw",
"CADDRawRankScore",
"CAROL",
"CONDEL",
"DannRankScore",
"DannScore",
"FathmmRankScore",
"FathmmScore",
"GerpNr",
"GerpRs",
"GerpRsRankScore",
"LRTConvertedRankScore",
"LRTScore",
"MetaLRRankScore",
"MetaLRScore",
"MetaSVMRankScore",
"MetaSVMScore",
"MutationAssessorRankScore",
"MutationAssessorScore",
"MutationTasterRankScore",
"MutationTasterScore",
"PERCH",
"PERCHnoMAF",
"Phastcons20wayMammalianRank",
"Phastcons7wayVertebrateRank",
"Phylop20wayMammalianRank",
"Phylop7wayVertebrateRank",
"Polyphen2HdivRankScore",
"Polyphen2HdivScore",
"Polyphen2HvarRankScore",
"Polyphen2HvarScore",
"PROVEANConvertedRankScore",
"PROVEANScore",
"SiftConvertedRankScore",
"SiftScore",
"Siphy29wayLogOddsRank",
```

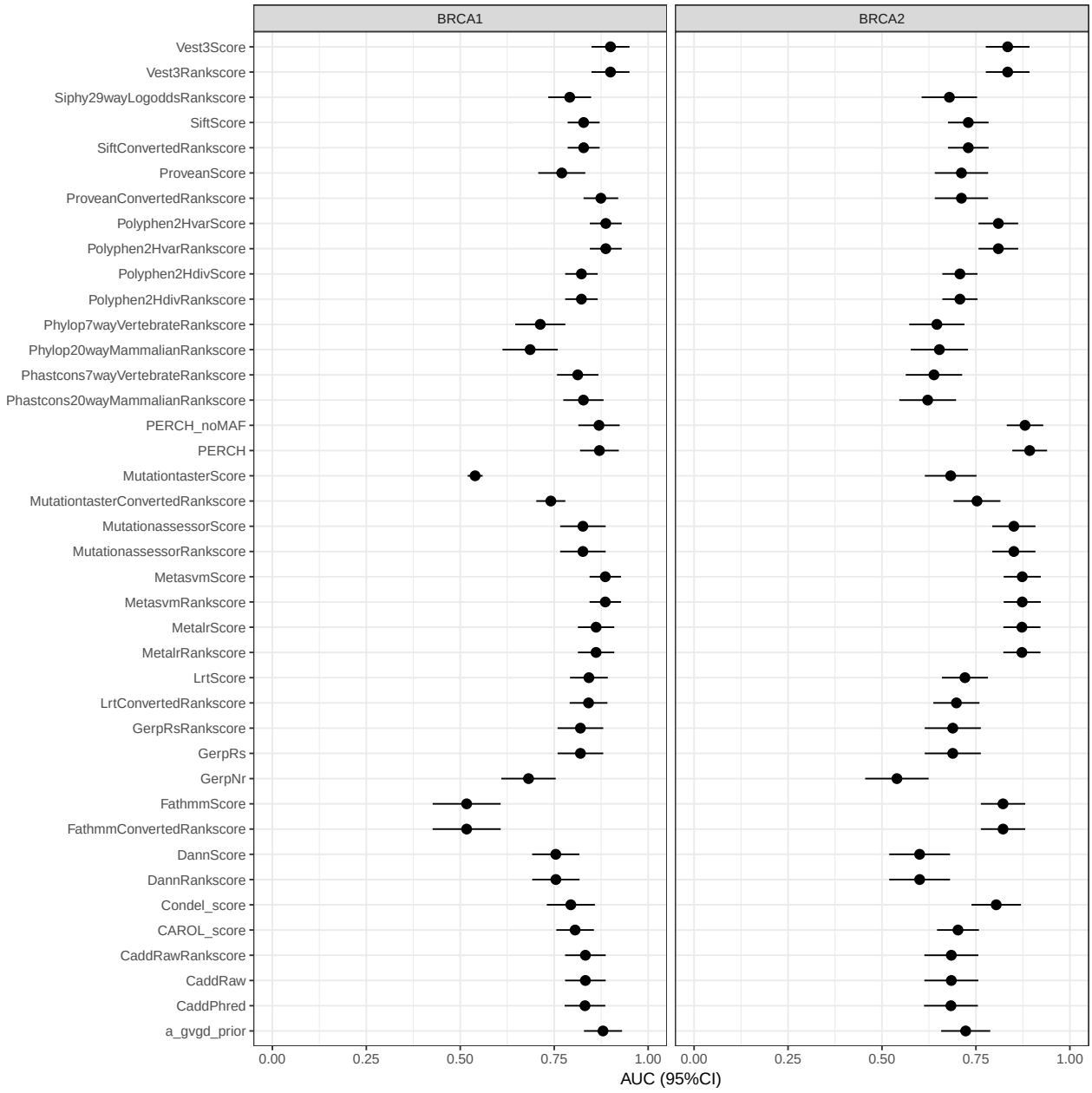
```
"Vest3RankScore",
"Vest3Score")
```

Figure 2

Area under the curve (AUC) for optimized thresholds in 40 damaging missense predictors.

Each dot represents the AUC, and bars are the 95% confidence intervals for each prediction.

```
################################################
#  Make forest plot of auc estimates           #
################################################
b2 <- ggplot(brca2a.auc.,aes(x=model,y=auc,ymin=auc.lcl,ymax=auc.ucl))+
  geom_pointrange()+
  ylim(0,1)+
  coord_flip()+
  theme_bw()+
  ylab("AUC (95%CI)")+
  xlab(NULL)+
  ggtitle("BRCA2") +
  scale_colour_grey()

b1 <- ggplot(brca1a.auc.,aes(x=model,y=auc,ymin=auc.lcl,ymax=auc.ucl))+
  geom_pointrange()+
  ylim(0,1)+
  coord_flip()+
  theme_bw()+
  ylab("AUC (95%CI)")+
  ggtitle("BRCA1")+
  theme(axis.text.y = element_blank())+
  xlab(NULL) + scale_colour_grey()


tmp1  <- brca1a.auc.
tmp1$Gene = "BRCA1"
tmp2  <- brca2a.auc.
tmp2$Gene = "BRCA2"
tmp<-rbind(tmp1,tmp2)

ggplot(tmp,aes(x=model,y=auc,ymin=auc.lcl,ymax=auc.ucl))+
  geom_pointrange()+
  ylim(0,1)+
  facet_grid(. ~ Gene)+
  coord_flip()+
  theme_bw()+
  ylab("AUC (95%CI)")+
  xlab(NULL)+ scale_colour_grey()
```




```
ggsave("figures/FIGURE_2.png",width=9,height=7, units="in",dpi = 600)
```

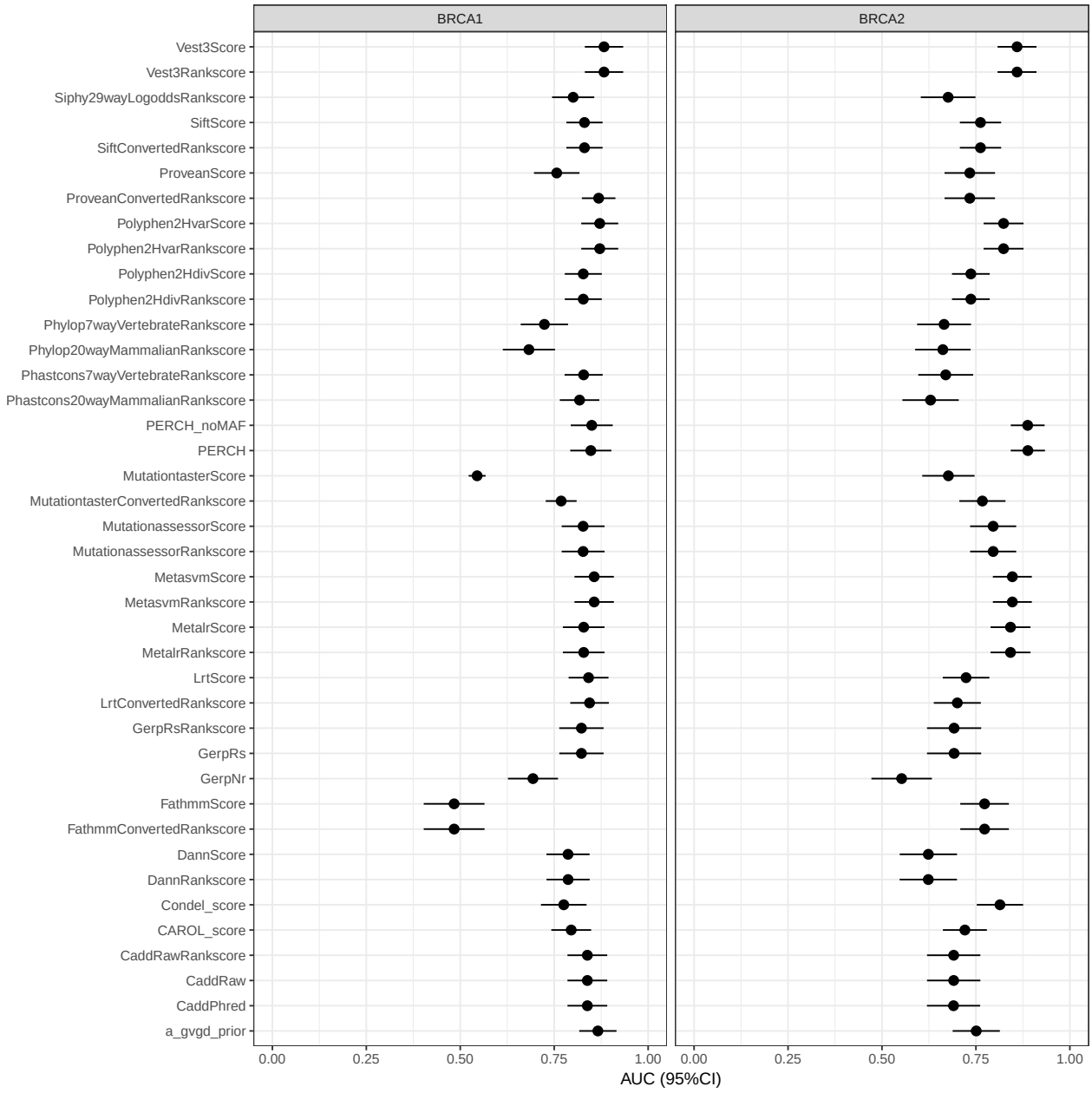
Figure S1

Area under the curve (AUC) for optimized thresholds in 40 damaging missense predictors.

Each dot represents the AUC, and bars are the 95% confidence intervals for Scenario 2.

```
##################################################
#  Make forest plot of auc estimates             #
##################################################
b2 <- ggplot(brca2b.auc.,aes(x=model,y=auc,ymin=auc.lcl,ymax=auc.ucl))+
  geom_pointrange()+
```

```
  ylim(0,1)+
  coord_flip()+
  theme_bw()+
  ylim(0,1)+ylab("AUC (95%CI)")+
  xlab(NULL)+
  ggtitle("BRCA2")

b1 <- ggplot(brca1b.auc.,aes(x=model,y=auc,ymin=auc.lcl,ymax=auc.ucl))+
  geom_pointrange()+
  ylim(0,1)+
  coord_flip()+
  theme_bw()+
  ylab("AUC (95%CI)")+
  ggtitle("BRCA1")+
  theme(axis.text.y = element_blank())+
  xlab(NULL)

tmp1  <- brca1b.auc.
tmp1$Gene = "BRCA1"
tmp2  <- brca2b.auc.
tmp2$Gene = "BRCA2"
tmp<-rbind(tmp1,tmp2)

ggplot(tmp,aes(x=model,y=auc,ymin=auc.lcl,ymax=auc.ucl))+
  geom_pointrange()+
  ylim(0,1)+
  facet_grid(. ~ Gene)+
  coord_flip()+
  theme_bw()+
  ylab("AUC (95%CI)")+
  xlab(NULL)
```




```
ggsave("figures/FIG_S1.png",width=9,height=7, units="in",dpi = 600)
```

Code for fitting Random Forests

```
##############################
# Deleterious vs Int/Neutral#
##############################

###############################
#   BRCA2                     #
###############################
```

```
class.del <- factor(vus.brca2$hdr_del)
MODELS<-vus.brca2[,c("CaddRawRankscore","DannRankscore","FathmmConvertedRankscore","GerpNr",
             "GerpRsRankscore","LrtConvertedRankscore","MetalrRankscore","MetasvmRankscore",
              "MutationassessorRankscore","MutationtasterConvertedRankscore",
               "Phastcons20wayMammalianRankscore","Phastcons7wayVertebrateRankscore",
                "Phylop20wayMammalianRankscore",
                  "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HvarRankscore",
                    "ProveanConvertedRankscore",
                    "SiftConvertedRankscore","Siphy29wayLogoddsRankscore","Vest3Rankscore",
                     "a_gvgd_prior","PERCH")]

#set.seed(559)
#tune2a <- tune.randomForest(y=class.del,x=MODELS,mtry=c(2,3,4,5,6,7,8,9,10),ntree=c(500))

#tune2b <- tuneRF(x=MODELS,y=class.del,ntreeTry=500,stepFactor=1.5,plot=T)

set.seed(559)

rf.brca2.del <- randomForest(factor(hdr_del)~
                               CaddRawRankscore+DannRankscore+FathmmConvertedRankscore+
                                   GerpNr+GerpRsRankscore+LrtConvertedRankscore+MetalrRankscore+
                                   MetasvmRankscore+
                               MutationassessorRankscore+MutationtasterConvertedRankscore+
                               Phastcons20wayMammalianRankscore+Phastcons7wayVertebrateRankscore+
                               Phylop20wayMammalianRankscore+
                               Phylop7wayVertebrateRankscore+Polyphen2HdivRankscore+
                               Polyphen2HvarRankscore+
                               ProveanConvertedRankscore+
                               SiftConvertedRankscore+Siphy29wayLogoddsRankscore+Vest3Rankscore+
                               a_gvgd_prior+PERCH,
                               importance=T,proximity=T,ntree=500,mtry=4,
                               keep.forest=T,data=vus.brca2)

imp2del <- as.data.frame(importance(rf.brca2.del,type=1))
imp2del. <- imp2del[order(imp2del$MeanDecreaseAccuracy),,drop=FALSE]

rf.brca2.pred.del <- predict(rf.brca2.del,type='prob')
vus.brca2. <- cbind(vus.brca2,round(rf.brca2.pred.del[,2],digits=3))
colnames(vus.brca2.)[76] <- "rf_prob_del"

#Add RF prediction for BRCA2 variants without functional data
brca2.del.new <- predict(rf.brca2.del,type='prob',newdata=all2)

all2. <- cbind(all2,round(brca2.del.new[,2],digits=3))
colnames(all2.)[74] <- "rf_prob_del"

##############################
#   BRCA1                    #
##############################

class.del <- factor(vus.brca1$hdr_del)
MODELS<-vus.brca1[,c("CAROL_score","Condel_score","CaddRawRankscore","DannRankscore",
                     "FathmmConvertedRankscore","GerpNr",
```

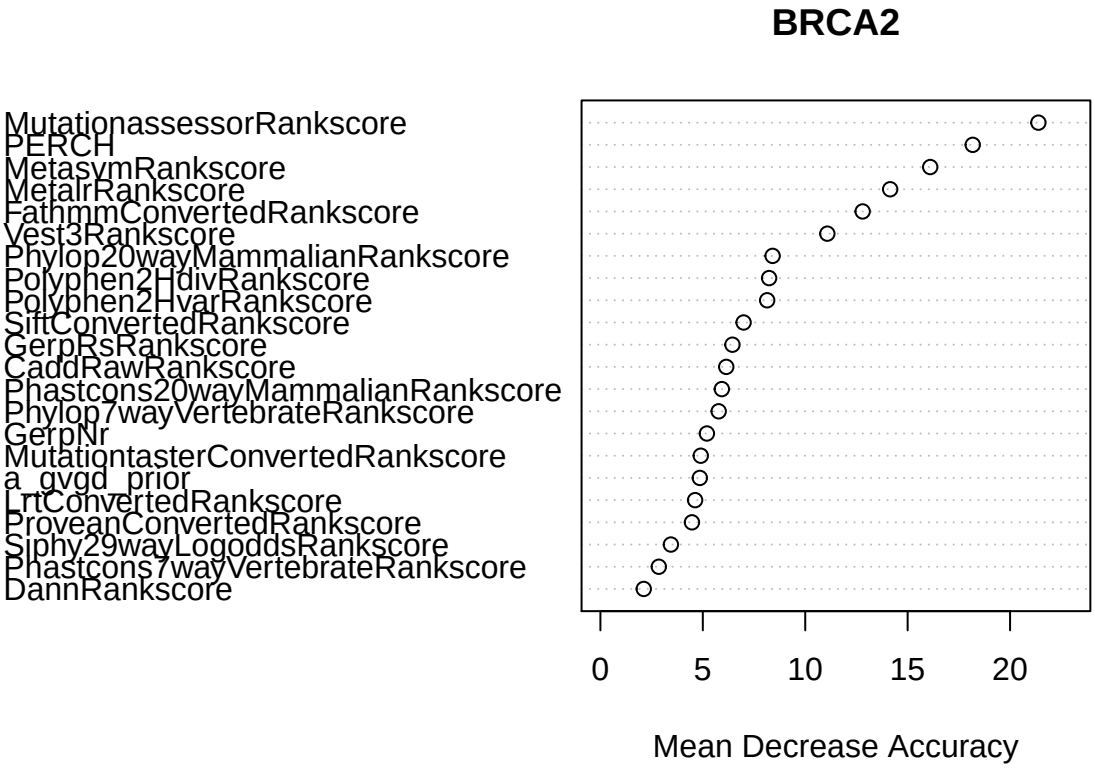
```
                      "GerpRsRankscore","LrtConvertedRankscore","MetalrRankscore",
                      "MetasvmRankscore",
                      "MutationassessorRankscore","MutationtasterConvertedRankscore",
                      "Phastcons20wayMammalianRankscore","Phastcons7wayVertebrateRankscore",
                      "Phylop20wayMammalianRankscore",
                      "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HvarRankscore",
                      "ProveanConvertedRankscore",
                      "SiftConvertedRankscore","Siphy29wayLogoddsRankscore","Vest3Rankscore",
                      "a_gvgd_prior","PERCH")]

#set.seed(559)
#tune1a <- tune.randomForest(y=class.del,x=MODELS,mtry=c(2,3,4,5,6,7,8,9,10),
#                                ntree=c(250,500,1000))

#tune1b <- tuneRF(x=MODELS,y=class.del,ntreeTry=500,stepFactor=1.5,plot=T)

set.seed(559)
rf.brca1.del <- randomForest(factor(hdr_del)~CaddRawRankscore+DannRankscore+FathmmConvertedRankscore+
                         GerpNr+GerpRsRankscore+LrtConvertedRankscore+MetalrRankscore+MetasvmRankscore+
                      MutationassessorRankscore+MutationtasterConvertedRankscore+
                      Phastcons20wayMammalianRankscore+Phastcons7wayVertebrateRankscore+
                      Phylop20wayMammalianRankscore+
                      Phylop7wayVertebrateRankscore+Polyphen2HdivRankscore+
                         Polyphen2HvarRankscore+ProveanConvertedRankscore+
                      SiftConvertedRankscore+Siphy29wayLogoddsRankscore+
                      Vest3Rankscore+a_gvgd_prior+PERCH+
                      CAROL_score+Condel_score,
                      importance=T,proximity=T,ntree=500,data=vus.brca1,
                      keep.forest=T,mtry=4)

imp1del <- as.data.frame(importance(rf.brca1.del,type=1))
imp1del. <- imp1del[order(imp1del$MeanDecreaseAccuracy),,drop=FALSE]

rf.brca1.pred.del <- predict(rf.brca1.del,type='prob')
vus.brca1. <- cbind(vus.brca1,round(rf.brca1.pred.del[,2],digits=3))
colnames(vus.brca1.)[76] <- "rf_prob_del"

#Add RF prediction for BRCA1 variants without functional data
brca1.del.new <- predict(rf.brca1.del,type='prob',newdata=all1)

all1. <- cbind(all1,round(brca1.del.new[,2],digits=3))
colnames(all1.)[74] <- "rf_prob_del"

####################################
# Del/Int vs Neutral                #
####################################

##############################
#   BRCA2                          #
##############################

class.delint <- factor(vus.brca2$hdr_not_neutral)
MODELS<-vus.brca2[,c("CaddRawRankscore","DannRankscore","FathmmConvertedRankscore","GerpNr",
```

```
             "GerpRsRankscore","LrtConvertedRankscore","MetalrRankscore","MetasvmRankscore",
              "MutationassessorRankscore","MutationtasterConvertedRankscore",
               "Phastcons20wayMammalianRankscore","Phastcons7wayVertebrateRankscore",
                 "Phylop20wayMammalianRankscore",
                  "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HvarRankscore",
                    "ProveanConvertedRankscore",
                      "SiftConvertedRankscore","Siphy29wayLogoddsRankscore","Vest3Rankscore",
                       "a_gvgd_prior","PERCH")]

#set.seed(420)
#tune2b <- tuneRF(x=MODELS,y=class.delint,ntreeTry=500,stepFactor=1.5,plot=T)

set.seed(420)
#rf.brca2.delint <- randomForest(y=class.delint,x=MODELS,importance=T,proximity=T,ntree=500,
#                  keep.forest=T,mtry=4)


rf.brca2.delint <- randomForest(factor(hdr_not_neutral)~CaddRawRankscore+DannRankscore+
                                  FathmmConvertedRankscore+
                                      GerpNr+GerpRsRankscore+LrtConvertedRankscore+MetalrRankscore+
                                      MetasvmRankscore+
                                  MutationassessorRankscore+MutationtasterConvertedRankscore+
                                  Phastcons20wayMammalianRankscore+Phastcons7wayVertebrateRankscore+
                                  Phylop20wayMammalianRankscore+
                                  Phylop7wayVertebrateRankscore+Polyphen2HdivRankscore+
                                  Polyphen2HvarRankscore+
                                  ProveanConvertedRankscore+
                                  SiftConvertedRankscore+Siphy29wayLogoddsRankscore+Vest3Rankscore+
                                  a_gvgd_prior+PERCH,
                                  importance=T,proximity=T,ntree=500,
                                  keep.forest=T,mtry=4,data=vus.brca2)

imp2delint <- as.data.frame(importance(rf.brca2.delint,type=1))
imp2delint. <- imp2delint[order(imp2delint$MeanDecreaseAccuracy),,drop=FALSE]

rf.brca2.pred.delint <- predict(rf.brca2.delint,type='prob')

vus.brca2.. <- cbind(vus.brca2.,round(rf.brca2.pred.delint[,2],digits=3))
colnames(vus.brca2..)[77] <- "rf_prob_delint"

#Add RF prediction for BRCA2 variants without functional data
brca2.delint.new <- predict(rf.brca2.delint,type='prob',newdata=all2)

all2.. <- cbind(all2.,round(brca2.delint.new[,2],digits=3))
colnames(all2..)[75] <- "rf_prob_delint"


##############################
#   BRCA1                    #
##############################


class.delint <- factor(vus.brca1$hdr_not_neutral)
```

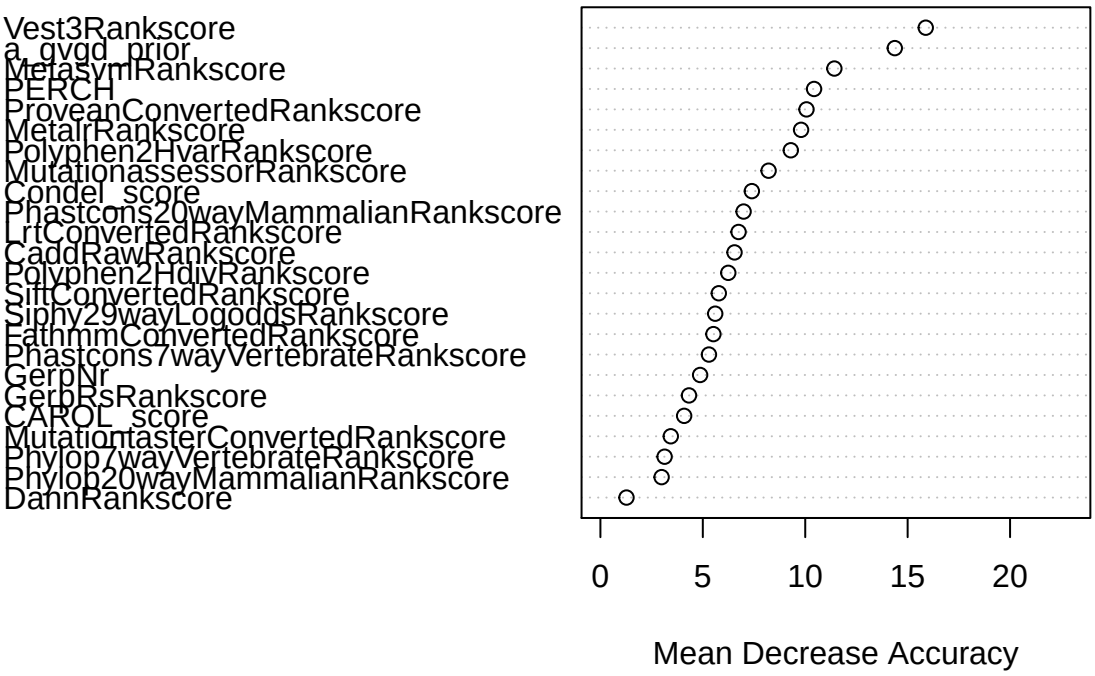
```
MODELS<-vus.brca1[,c("CAROL_score","Condel_score","CaddRawRankscore","DannRankscore",
                     "FathmmConvertedRankscore","GerpNr",
                     "GerpRsRankscore","LrtConvertedRankscore","MetalrRankscore",
                     "MetasvmRankscore",
                     "MutationassessorRankscore","MutationtasterConvertedRankscore",
                     "Phastcons20wayMammalianRankscore","Phastcons7wayVertebrateRankscore",
                     "Phylop20wayMammalianRankscore",
                     "Phylop7wayVertebrateRankscore","Polyphen2HdivRankscore","Polyphen2HvarRankscore",
                     "ProveanConvertedRankscore",
                     "SiftConvertedRankscore","Siphy29wayLogoddsRankscore","Vest3Rankscore",
                     "a_gvgd_prior","PERCH")]

#set.seed(420)
#tune1b <- tuneRF(x=MODELS,y=class.delint,ntreeTry=500,stepFactor=1.5,plot=T)

set.seed(420)

rf.brca1.delint <- randomForest(factor(hdr_not_neutral)~
                                  CaddRawRankscore+DannRankscore+
                                  FathmmConvertedRankscore+
                                      GerpNr+GerpRsRankscore+LrtConvertedRankscore+
                                      MetalrRankscore+MetasvmRankscore+
                                  MutationassessorRankscore+MutationtasterConvertedRankscore+
                                  Phastcons20wayMammalianRankscore+Phastcons7wayVertebrateRankscore+
                                  Phylop20wayMammalianRankscore+
                                  Phylop7wayVertebrateRankscore+Polyphen2HdivRankscore+
                                  Polyphen2HvarRankscore+
                                  ProveanConvertedRankscore+
                                  SiftConvertedRankscore+Siphy29wayLogoddsRankscore+Vest3Rankscore+
                                  a_gvgd_prior+PERCH+
                                  CAROL_score+
                                  Condel_score,
                                  importance=T,proximity=T,ntree=500,
                                  keep.forest=T,mtry=4,data=vus.brca1)

imp1delint <- as.data.frame(importance(rf.brca1.delint,type=1))
imp1delint. <- imp1delint[order(imp1delint$MeanDecreaseAccuracy),,drop=FALSE]

rf.brca1.pred.delint <- predict(rf.brca1.delint,type='prob')

vus.brca1.. <- cbind(vus.brca1.,round(rf.brca1.pred.delint[,2],digits=3))
colnames(vus.brca1..)[77] <- "rf_prob_delint"

# Add RF prediction for BRCA1 variants without functional data
brca1.delint.new <- predict(rf.brca1.delint,type='prob',newdata=all1)

all1.. <- cbind(all1.,round(brca1.delint.new[,2],digits=3))
colnames(all1..)[75] <- "rf_prob_delint"
```

Random Forest Variable Importance Plot (Scenario 1) - Not used

```
dotchart(imp2del.[,1],labels=rownames(imp2del.),xlim=c(0,23),xlab='Mean Decrease Accuracy',main='BRCA2'
```



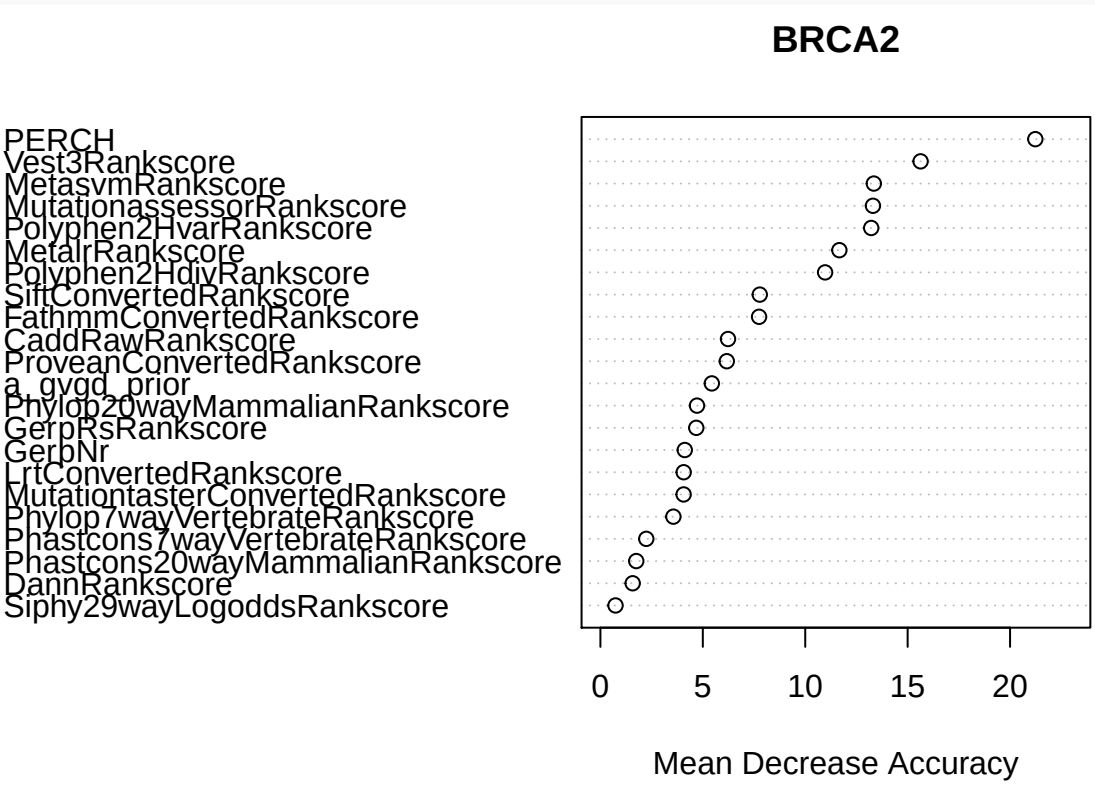

```
dotchart(imp1del.[,1],labels=rownames(imp1del.),xlim=c(0,23),xlab='Mean Decrease Accuracy',main='BRCA1')
```



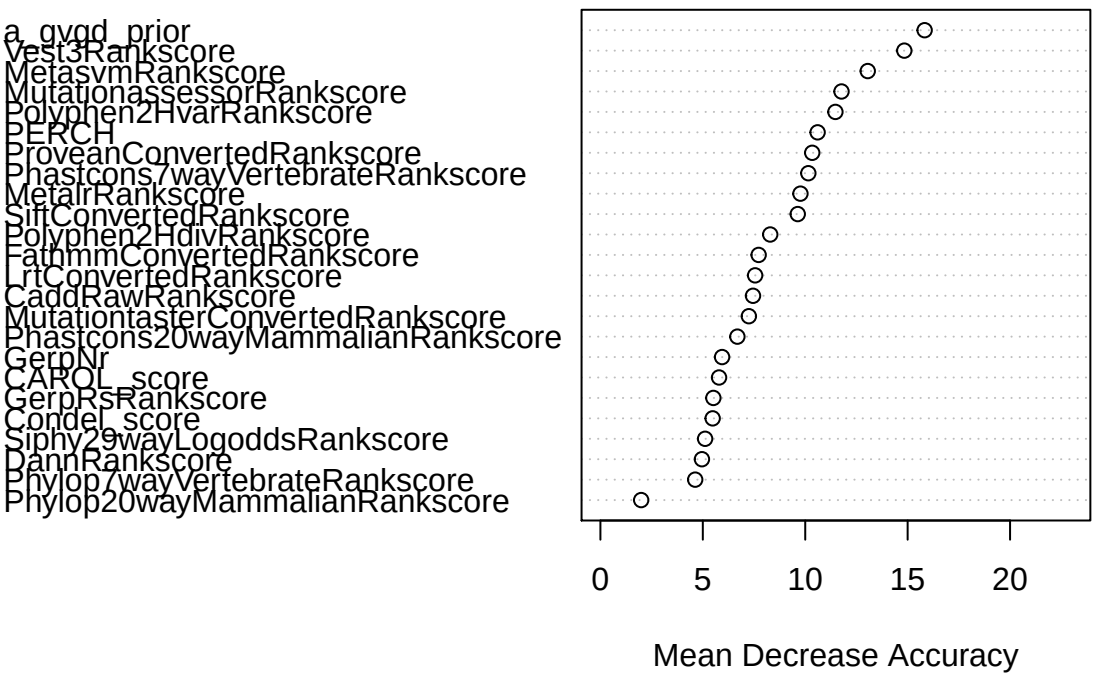

```
par(xpd=T)
```

Random Forest Variable Importance Plot (Scenario 2)

```
dotchart(imp2delint.[,1],labels=rownames(imp2delint.),xlim=c(0,23),xlab='Mean Decrease Accuracy',main='
```




```
dotchart(imp1delint.[,1],labels=rownames(imp1delint.),xlim=c(0,23),xlab='Mean Decrease Accuracy',main='
```



```
par(xpd=T)
```

Code for fitting NVM

```
brca2$perch_del <- ifelse(brca2$PERCH>=0.295957,1,0)
brca2$metasvmrank_del <- ifelse(brca2$MetasvmRankscore>=0.93181,1,0)
brca2$metalrrank_del <- ifelse(brca2$MetalrRankscore>=0.92107,1,0)
brca2$mutationassessor_del <- ifelse(brca2$MutationassessorRankscore>=0.84368,1,0)
brca2$vest3rank_del <- ifelse(brca2$Vest3Rankscore>=0.79963,1,0)
brca2$pp2hvarrank_del <- ifelse(brca2$Polyphen2HvarRankscore>=0.97092,1,0)
brca2$fathmm_del <- ifelse(brca2$FathmmConvertedRankscore>=0.8581,1,0)
brca2$mutationtaster_del <- ifelse(brca2$MutationtasterConvertedRankscore>=0.58814,1,0)
brca2$sift_del <- ifelse(brca2$SiftConvertedRankscore>=0.9122,1,0)
brca2$agvgd_del <- ifelse(brca2$a_gvgd_prior>=0.81,1,0)
brca2$lrt_del <- ifelse(brca2$LrtScore<=0,1,0)
brca2$provean_del <- ifelse(brca2$ProveanConvertedRankscore>=0.42352,1,0)
brca2$pp2hdiv_del <- ifelse(brca2$Polyphen2HdivRankscore>=0.89865,1,0)
brca2$gerprs_del <- ifelse(brca2$GerpRsRankscore>=0.7396,1,0)
brca2$cadd_del <- ifelse(brca2$CaddPhred>=29.4,1,0)
brca2$siphy_del <- ifelse(brca2$Siphy29wayLogoddsRankscore>=0.78299,1,0)
brca2$phylopmammal_del <- ifelse(brca2$Phylop20wayMammalianRankscore>=0.63346,1,0)
brca2$phylopvertebrate_del <- ifelse(brca2$Phylop7wayVertebrateRankscore>=0.76621,1,0)
brca2$phastconsvertebrate_del <- ifelse(brca2$Phastcons7wayVertebrateRankscore>=0.61045,1,0)
brca2$phastconsmammal_del <- ifelse(brca2$Phastcons20wayMammalianRankscore>=0.60234,1,0)
brca2$dann_del <- ifelse(brca2$DannRankscore>=0.84566,1,0)
brca2$gerpnr_del <- ifelse(brca2$GerpNr>=5.49,1,0)

brca2$sum_del_1 = with(brca2, perch_del)
brca2$sum_del_2 = with(brca2, perch_del+metasvmrank_del)
brca2$sum_del_3 = with(brca2, perch_del+metasvmrank_del+metalrrank_del)
brca2$sum_del_4 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del)
brca2$sum_del_5 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+
                         pp2hvarrank_del)
brca2$sum_del_6 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del)
brca2$sum_del_7 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del+pp2hdiv_del)
brca2$sum_del_8 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del+pp2hdiv_del+fathmm_del)
brca2$sum_del_9 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del)
brca2$sum_del_10 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del)
brca2$sum_del_11 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del)
brca2$sum_del_12 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del)
brca2$sum_del_13 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                  mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
```

```
                    lrt_del+agvgd_del+provean_del)
brca2$sum_del_14 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del)
brca2$sum_del_15 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del)
brca2$sum_del_16 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del+siphy_del)
brca2$sum_del_17 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del+siphy_del+phastconsvertebrate_del
brca2$sum_del_18 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del+siphy_del+phastconsvertebrate_del
                     phylopmammal_del)
brca2$sum_del_19 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del+siphy_del+phastconsvertebrate_del
                     phylopmammal_del+phylopvertebrate_del)
brca2$sum_del_20 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del+siphy_del+phastconsvertebrate_del
                     phylopmammal_del+phylopvertebrate_del+gerpnr_del)
brca2$sum_del_21 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del+siphy_del+phastconsvertebrate_del
                     phylopmammal_del+phylopvertebrate_del+gerpnr_del+phastconsmammal_del)
brca2$sum_del_22 = with(brca2, perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                   mutationassessor_del+pp2hdiv_del+fathmm_del+sift_del+mutationtaster_del+
                    lrt_del+agvgd_del+provean_del+cadd_del+gerprs_del+siphy_del+phastconsvertebrate_del
                     phylopmammal_del+phylopvertebrate_del+gerpnr_del+phastconsmammal_del+
                       dann_del)

brca2$perch_nn <- ifelse(brca2$PERCH>=0.232796,1,0)
brca2$metasvmrank_nn <- ifelse(brca2$MetasvmRankscore>=0.92233,1,0)
brca2$metalrrank_nn <- ifelse(brca2$MetalrRankscore>=0.91514,1,0)
brca2$mutationassessor_nn <- ifelse(brca2$MutationassessorRankscore>=0.83541,1,0)
brca2$vest3rank_nn <- ifelse(brca2$Vest3Rankscore>=0.7831,1,0)
brca2$fathmm_nn <- ifelse(brca2$FathmmConvertedRankscore>=0.84569,1,0)
brca2$pp2hvarrank_nn <- ifelse(brca2$Polyphen2HvarRankscore>=0.97092,1,0)
brca2$mutationtaster_nn <- ifelse(brca2$MutationtasterConvertedRankscore>=0.58814,1,0)
brca2$sift_nn <- ifelse(brca2$SiftConvertedRankscore>=0.91219,1,0)
brca2$agvgd_nn <- ifelse(brca2$a_gvgd_prior>=0.81,1,0)
brca2$lrt_nn <- ifelse(brca2$LrtScore<=0,1,0)
brca2$provean_nn <- ifelse(brca2$ProveanConvertedRankscore>=0.4197,1,0)
brca2$pp2hdiv_nn <- ifelse(brca2$Polyphen2HdivRankscore>=0.89865,1,0)
brca2$gerprs_nn <- ifelse(brca2$GerpRsRankscore>=0.71253,1,0)
brca2$cadd_nn <- ifelse(brca2$CaddRawRankscore>=0.86323,1,0)
brca2$siphy_nn <- ifelse(brca2$Siphy29wayLogoddsRankscore>=0.75958,1,0)
brca2$phylopmammal_nn <- ifelse(brca2$Phylop20wayMammalianRankscore>=0.61215,1,0)
brca2$phylopvertebrate_nn <- ifelse(brca2$Phylop7wayVertebrateRankscore>=0.76621,1,0)
```

```
brca2$phastconsvertebrate_nn <- ifelse(brca2$Phastcons7wayVertebrateRankscore>=0.59096,1,0)
brca2$phastconsmammal_nn <- ifelse(brca2$Phastcons20wayMammalianRankscore>=0.60234,1,0)
brca2$dann_nn <- ifelse(brca2$DannRankscore>=0.8433,1,0)
brca2$gerpnr_nn <- ifelse(brca2$GerpNr>=5.49,1,0)

brca2$sum_nn_1 = with(brca2, perch_nn)
brca2$sum_nn_2 = with(brca2, perch_nn+vest3rank_nn)
brca2$sum_nn_3 = with(brca2, perch_nn+vest3rank_nn+metalrrank_nn)
brca2$sum_nn_4 = with(brca2, perch_nn+vest3rank_nn+metalrrank_nn+pp2hvarrank_nn)
brca2$sum_nn_5 = with(brca2, perch_nn+vest3rank_nn+metalrrank_nn+pp2hvarrank_nn+metasvmrank_nn)
brca2$sum_nn_6 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn)
brca2$sum_nn_7 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn)
brca2$sum_nn_8 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn)
brca2$sum_nn_9 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn)
brca2$sum_nn_10 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn)
brca2$sum_nn_11 = with(brca2,perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn)
brca2$sum_nn_12 = with(brca2,perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn)
brca2$sum_nn_13 = with(brca2,perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn+provean_nn)
brca2$sum_nn_14 = with(brca2,perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn)
brca2$sum_nn_15 = with(brca2,perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn)
brca2$sum_nn_16 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn+
                 siphy_nn)
brca2$sum_nn_17 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn+
                 siphy_nn+gerprs_nn)
brca2$sum_nn_18 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn+
                 siphy_nn+gerprs_nn+phylopmammal_nn)
brca2$sum_nn_19 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                 agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn+
                 siphy_nn+gerprs_nn+phylopmammal_nn+phylopvertebrate_nn)
brca2$sum_nn_20 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                 pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
```

```
                agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn+
                siphy_nn+gerprs_nn+phylopmammal_nn+phylopvertebrate_nn+gerpnr_nn)
brca2$sum_nn_21 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn+
                siphy_nn+gerprs_nn+phylopmammal_nn+phylopvertebrate_nn+gerpnr_nn+
                dann_nn)
brca2$sum_nn_22 = with(brca2, perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                pp2hdiv_nn+sift_nn+mutationassessor_nn+mutationtaster_nn+lrt_nn+
                agvgd_nn+fathmm_nn+provean_nn+phastconsvertebrate_nn+cadd_nn+
                siphy_nn+gerprs_nn+phylopmammal_nn+phylopvertebrate_nn+gerpnr_nn+
                dann_nn+ phastconsmammal_nn)

#BRCA1


brca1$perch_del <- ifelse(brca1$PERCH_noMAF>=0.206316,1,0)
brca1$metasvmrank_del <- ifelse(brca1$MetasvmRankscore>=0.9083,1,0)
brca1$metalrrank_del <- ifelse(brca1$MetalrRankscore>=0.90291,1,0)
brca1$mutationassessor_del <- ifelse(brca1$MutationassessorRankscore>=0.54432,1,0)
brca1$vest3rank_del <- ifelse(brca1$Vest3Rankscore>=0.85546,1,0)
brca1$fathmm_del <- ifelse(brca1$FathmmConvertedRankscore>=0.88158,1,0)
brca1$pp2hvarrank_del <- ifelse(brca1$Polyphen2HvarRankscore>=0.91584,1,0)
brca1$mutationtaster_del <- ifelse(brca1$MutationtasterConvertedRankscore>=0.81033,1,0)
brca1$sift_del <- ifelse(brca1$SiftConvertedRankscore>=0.91219,1,0)
brca1$agvgd_del <- ifelse(brca1$a_gvgd_prior>=0.29,1,0)
brca1$lrt_del <- ifelse(brca1$LrtScore<=0.000926,1,0)
brca1$provean_del <- ifelse(brca1$ProveanConvertedRankscore>=0.69906,1,0)
brca1$pp2hdiv_del <- ifelse(brca1$Polyphen2HdivRankscore>=0.89865,1,0)
brca1$gerprs_del <- ifelse(brca1$GerpRsRankscore>=0.67646,1,0)
brca1$cadd_del <- ifelse(brca1$CaddRawRankscore>=0.66888,1,0)
brca1$siphy_del <- ifelse(brca1$Siphy29wayLogoddsRankscore>=0.49967,1,0)
brca1$phylopmammal_del <- ifelse(brca1$Phylop20wayMammalianRankscore>=0.48891,1,0)
brca1$phylopvertebrate_del <- ifelse(brca1$Phylop7wayVertebrateRankscore>=0.60462,1,0)
brca1$phastconsvertebrate_del <- ifelse(brca1$Phastcons7wayVertebrateRankscore>=0.54028,1,0)
brca1$phastconsmammal_del <- ifelse(brca1$Phastcons20wayMammalianRankscore>=0.51549,1,0)
brca1$dann_del <- ifelse(brca1$DannRankscore>=0.69278,1,0)
brca1$gerpnr_del <- ifelse(brca1$GerpNr>=5.22,1,0)
brca1$carol_del <- ifelse(brca1$CAROL_score>=1,1,0)
brca1$condel_del <- ifelse(brca1$Condel_score>=0.689,1,0)

brca1$sum_del_1 = with(brca1, vest3rank_del)
brca1$sum_del_2 = with(brca1, vest3rank_del+perch_del)
brca1$sum_del_3 = with(brca1, vest3rank_del+perch_del+agvgd_del)
brca1$sum_del_4 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del)
brca1$sum_del_5 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del)
brca1$sum_del_6 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                sift_del)
brca1$sum_del_7 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                sift_del+pp2hdiv_del)
brca1$sum_del_8 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                sift_del+pp2hdiv_del+gerprs_del)
brca1$sum_del_9 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
```

```
                  sift_del+pp2hdiv_del+gerprs_del+lrt_del)
brca1$sum_del_10 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                  sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del)
brca1$sum_del_11 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                  sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del)
brca1$sum_del_12 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                  sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del)
brca1$sum_del_13 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                  sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del)
brca1$sum_del_14 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del)
brca1$sum_del_15 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del)
brca1$sum_del_16 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del)
brca1$sum_del_17 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+
                   cadd_del)
brca1$sum_del_18 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+cadd_de
                   phastconsmammal_del)
brca1$sum_del_19 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+cadd_de
                   phastconsmammal_del+dann_del)
brca1$sum_del_20 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+cadd_de
                   phastconsmammal_del+dann_del+condel_del)
brca1$sum_del_21 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+cadd_de
                   phastconsmammal_del+dann_del+condel_del+phylopvertebrate_del)
brca1$sum_del_22 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+cadd_de
                   phastconsmammal_del+dann_del+condel_del+phylopvertebrate_del+gerpnr_del)
brca1$sum_del_23 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+cadd_de
                   phastconsmammal_del+dann_del+condel_del+phylopvertebrate_del+gerpnr_del+
                   phylopmammal_del)
brca1$sum_del_24 = with(brca1, vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                   sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                   metalrrank_del+phastconsvertebrate_del+mutationtaster_del+carol_del+siphy_del+cadd_de
                   phastconsmammal_del+dann_del+condel_del+phylopvertebrate_del+gerpnr_del+
```

```
                  phylopmammal_del+fathmm_del)

brca1$perch_nn <- ifelse(brca1$PERCH_noMAF>=0.140337,1,0)
brca1$metasvmrank_nn <- ifelse(brca1$MetasvmRankscore>=0.8874,1,0)
brca1$metalrrank_nn <- ifelse(brca1$MetalrRankscore>=0.89743,1,0)
brca1$mutationassessor_nn <- ifelse(brca1$MutationassessorRankscore>=0.44691,1,0)
brca1$vest3rank_nn <- ifelse(brca1$Vest3Rankscore>=0.80056,1,0)
brca1$fathmm_nn <- ifelse(brca1$FathmmConvertedRankscore>=0.88313,1,0)
brca1$pp2hvarrank_nn <- ifelse(brca1$Polyphen2HvarRankscore>=0.85043,1,0)
brca1$mutationtaster_nn <- ifelse(brca1$MutationtasterConvertedRankscore>=0.81033,1,0)
brca1$sift_nn <- ifelse(brca1$SiftConvertedRankscore>=0.91219,1,0)
brca1$agvgd_nn <- ifelse(brca1$a_gvgd_prior>=0.29,1,0)
brca1$lrt_nn <- ifelse(brca1$LrtConvertedRankscore>=0.38483,1,0)
brca1$provean_nn <- ifelse(brca1$ProveanConvertedRankscore>=0.65469,1,0)
brca1$pp2hdiv_nn <- ifelse(brca1$Polyphen2HdivRankscore>=0.89865,1,0)
brca1$gerprs_nn <- ifelse(brca1$GerpRsRankscore>=0.61659,1,0)
brca1$cadd_nn <- ifelse(brca1$CaddRawRankscore>=0.64548,1,0)
brca1$siphy_nn <- ifelse(brca1$Siphy29wayLogoddsRankscore>=0.48478,1,0)
brca1$phylopmammal_nn <- ifelse(brca1$Phylop20wayMammalianRankscore>=0.48811,1,0)
brca1$phylopvertebrate_nn <- ifelse(brca1$Phylop7wayVertebrateRankscore>=0.60462,1,0)
brca1$phastconsvertebrate_nn <- ifelse(brca1$Phastcons7wayVertebrateRankscore>=0.46971,1,0)
brca1$phastconsmammal_nn <- ifelse(brca1$Phastcons20wayMammalianRankscore>=0.47504,1,0)
brca1$dann_nn <- ifelse(brca1$DannRankscore>=0.67191,1,0)
brca1$gerpnr_nn <- ifelse(brca1$GerpNr>=5.21,1,0)
brca1$carol_nn <- ifelse(brca1$CAROL_score>=1,1,0)
brca1$condel_nn <- ifelse(brca1$Condel_score>=0.674,1,0)

brca1$sum_nn_1 = with(brca1, vest3rank_nn)
brca1$sum_nn_2 = with(brca1, vest3rank_nn+agvgd_nn)
brca1$sum_nn_3 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn)
brca1$sum_nn_4 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn)
brca1$sum_nn_5 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn)
brca1$sum_nn_6 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn)
brca1$sum_nn_7 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn)
brca1$sum_nn_8 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn)
brca1$sum_nn_9 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn)
brca1$sum_nn_10 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                   pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn)
brca1$sum_nn_11 = with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                   pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn)
brca1$sum_nn_12= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn)
brca1$sum_nn_13= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn)
brca1$sum_nn_14= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn)
brca1$sum_nn_15= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
```

```
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn)
brca1$sum_nn_16= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn)
brca1$sum_nn_17= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn)
brca1$sum_nn_18= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn+carol_nn)
brca1$sum_nn_19= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn+carol_nn+dann_nn)
brca1$sum_nn_20= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn+carol_nn+dann_nn+condel_nn)
brca1$sum_nn_21= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn+carol_nn+dann_nn+condel_nn+phylopvertebrate_nn)
brca1$sum_nn_22= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn+carol_nn+dann_nn+condel_nn+phylopvertebrate_nn+
                  phylopmammal_nn)
brca1$sum_nn_23= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn+carol_nn+dann_nn+condel_nn+phylopvertebrate_nn+
                  phylopmammal_nn+gerpnr_nn)
brca1$sum_nn_24= with(brca1, vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn+mutationtaster_nn+provean_nn+
                  cadd_nn+phastconsmammal_nn+metalrrank_nn+siphy_nn+mutationassessor_nn+
                  phastconsvertebrate_nn+carol_nn+dann_nn+condel_nn+phylopvertebrate_nn+
                  phylopmammal_nn+gerpnr_nn+fathmm_nn)
```

Calculate MCC at optimal cutpoint for each possible number of votes in NVM

```
brca2.votes.train <- subset(brca2,set_=='train')
brca1.votes.train <- subset(brca1,set_=='train')
brca2.votes.test <- subset(brca2,set_=='test')
brca1.votes.test <- subset(brca1,set_=='test')

# BRCA2 - Deleterious vs Intermediate/Neutral (Scenario 1)
fn <- matrix(nrow=22,ncol=1)
fp <- matrix(nrow=22,ncol=1)
nh <- matrix(nrow=22,ncol=1)
nd <- matrix(nrow=22,ncol=1)
tp <- matrix(nrow=22,ncol=1)
```

```
tn <- matrix(nrow=22,ncol=1)
mcc <- matrix(nrow=22,ncol=1)
cutoff <- matrix(nrow=22,ncol=1)

x <- brca2.votes.train[,c("sum_del_1","sum_del_2","sum_del_3","sum_del_4","sum_del_5",
                          "sum_del_6","sum_del_7","sum_del_8","sum_del_9","sum_del_10",
                          "sum_del_11","sum_del_12","sum_del_13","sum_del_14","sum_del_15",
                          "sum_del_16","sum_del_17","sum_del_18","sum_del_19","sum_del_20",
                          "sum_del_21","sum_del_22","hdr_del")]

for (i in 1:22)
{

x$use <- x[,i]
v2a.sum.i <- optimal.cutpoints(X="use",status="hdr_del",
                            tag.healthy=0,methods=c("SpEqualSe"),valueSe=0.90,
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=x)

fn[i,1] <- v2a.sum.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v2a.sum.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v2a.sum.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v2a.sum.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v2a.sum.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                                (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

votes <- as.matrix(rep(1:22),nrow=22,ncol=1)
nvm.mcc.2a <- data.frame(votes,cutoff,mcc)

}

#BRCA1 - Deleterious vs Intermediate/Neutral (Scenario 1)

fn <- matrix(nrow=24,ncol=1)
fp <- matrix(nrow=24,ncol=1)
nh <- matrix(nrow=24,ncol=1)
nd <- matrix(nrow=24,ncol=1)
tp <- matrix(nrow=24,ncol=1)
tn <- matrix(nrow=24,ncol=1)
mcc <- matrix(nrow=24,ncol=1)
cutoff <- matrix(nrow=24,ncol=1)

x <- brca1.votes.train[,c("sum_del_1","sum_del_2","sum_del_3","sum_del_4","sum_del_5",
                          "sum_del_6","sum_del_7","sum_del_8","sum_del_9","sum_del_10",
                          "sum_del_11","sum_del_12","sum_del_13","sum_del_14","sum_del_15",
                          "sum_del_16","sum_del_17","sum_del_18","sum_del_19","sum_del_20",
                          "sum_del_21","sum_del_22","sum_del_23","sum_del_24","hdr_del")]
```

```
for (i in 1:24){

x$use <- x[,i]
v1a.sum.i <- optimal.cutpoints(X="use",status="hdr_del",
                           tag.healthy=0,methods=c("SpEqualSe"),valueSe=0.90,
                                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                     ci.fit=TRUE,data=x)

fn[i,1] <- v1a.sum.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v1a.sum.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1a.sum.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1a.sum.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1a.sum.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                                (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

votes <- as.matrix(rep(1:24),nrow=24,ncol=1)
nvm.mcc.1a <- data.frame(votes,cutoff,mcc)

}

#BRCA2 - Deleterious/Intermediate vs Neutral (Scenario 2)

fn <- matrix(nrow=22,ncol=1)
fp <- matrix(nrow=22,ncol=1)
nh <- matrix(nrow=22,ncol=1)
nd <- matrix(nrow=22,ncol=1)
tp <- matrix(nrow=22,ncol=1)
tn <- matrix(nrow=22,ncol=1)
mcc <- matrix(nrow=22,ncol=1)
cutoff <- matrix(nrow=22,ncol=1)

x <- brca2.votes.train[,c("sum_nn_1","sum_nn_2","sum_nn_3","sum_nn_4","sum_nn_5",
                          "sum_nn_6","sum_nn_7","sum_nn_8","sum_nn_9","sum_nn_10",
                          "sum_nn_11","sum_nn_12","sum_nn_13","sum_nn_14","sum_nn_15",
                          "sum_nn_16","sum_nn_17","sum_nn_18","sum_nn_19","sum_nn_20",
                          "sum_nn_21","sum_nn_22","hdr_not_neutral")]

for (i in 1:22){

x$use <- x[,i]
v2b.sum.i <- optimal.cutpoints(X="use",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("SpEqualSe"),valueSe=0.90,
                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                     ci.fit=TRUE,data=x)

fn[i,1] <- v2b.sum.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v2b.sum.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v2b.sum.i$SpEqualSe[[1]]$measures.acc$n$h
```

```
nd[i,1] <- v2b.sum.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v2b.sum.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                                                (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

votes <- as.matrix(rep(1:22),nrow=22,ncol=1)
nvm.mcc.2b <- data.frame(votes,cutoff,mcc)

}


#BRCA1 - Deleterious/Intermediate vs Neutral (Scenario 2)

fn <- matrix(nrow=24,ncol=1)
fp <- matrix(nrow=24,ncol=1)
nh <- matrix(nrow=24,ncol=1)
nd <- matrix(nrow=24,ncol=1)
tp <- matrix(nrow=24,ncol=1)
tn <- matrix(nrow=24,ncol=1)
mcc <- matrix(nrow=24,ncol=1)
cutoff <- matrix(nrow=24,ncol=1)

x <- brca1.votes.train[,c("sum_nn_1","sum_nn_2","sum_nn_3","sum_nn_4","sum_nn_5",
                          "sum_nn_6","sum_nn_7","sum_nn_8","sum_nn_9","sum_nn_10",
                          "sum_nn_11","sum_nn_12","sum_nn_13","sum_nn_14","sum_nn_15",
                          "sum_nn_16","sum_nn_17","sum_nn_18","sum_nn_19","sum_nn_20",
                          "sum_nn_21","sum_nn_22","sum_nn_23","sum_nn_24","hdr_not_neutral")]


for (i in 1:24){

x$use <- x[,i]
v1b.sum.i <- optimal.cutpoints(X="use",status="hdr_not_neutral",
                               tag.healthy=0,methods=c("SpEqualSe"),valueSe=0.90,
                     control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                     ci.fit=TRUE,data=x)


fn[i,1] <- v1b.sum.i$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- v1b.sum.i$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- v1b.sum.i$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- v1b.sum.i$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- v1b.sum.i$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
                              (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))
```

```
votes <- as.matrix(rep(1:24),nrow=24,ncol=1)
nvm.mcc.1b <- data.frame(votes,cutoff,mcc)

}
```

Figure S2

NVM - MCC vs number of models voting plots

BRCA2, Scenario 1

```
FIG_S2_A<-NULL
FIG_S2_A$Votes <- nvm.mcc.2a$votes
FIG_S2_A$MCC   <- nvm.mcc.2a$mcc
FIG_S2_A$Gene  <- "BRCA2"
FIG_S2_A$Scenario <- 1
```

BRCA1, Scenario 1

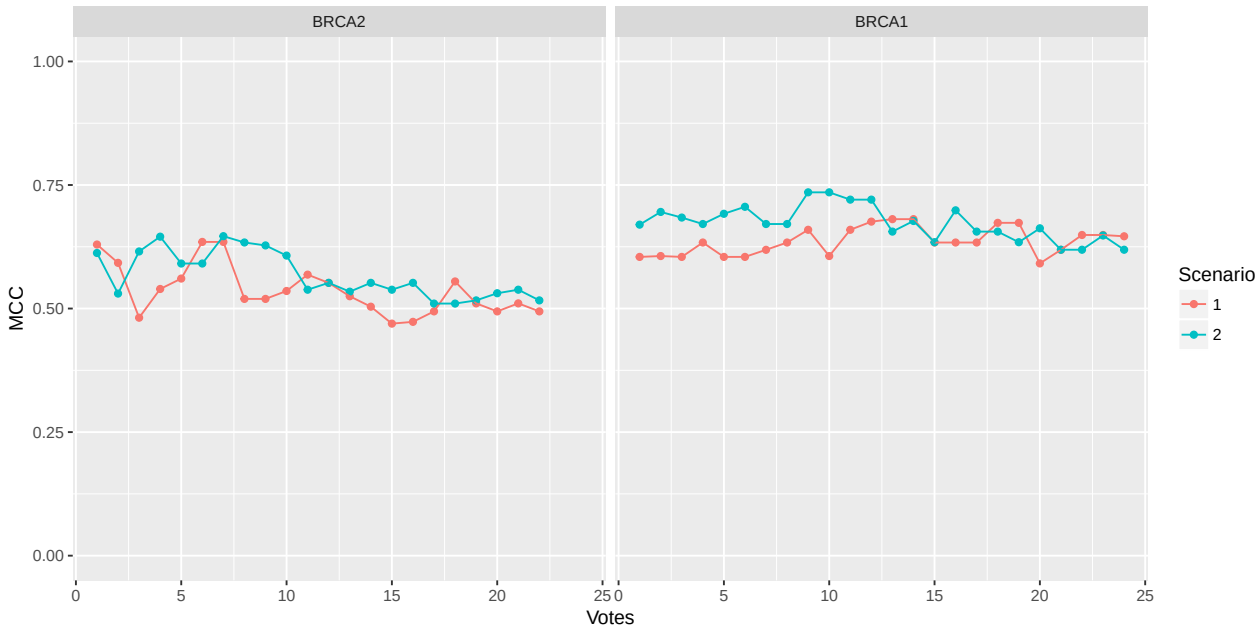
```
FIG_S2_B<-NULL
FIG_S2_B$Votes <- nvm.mcc.1a$votes
FIG_S2_B$MCC   <- nvm.mcc.1a$mcc
FIG_S2_B$Gene  <- "BRCA1"
FIG_S2_B$Scenario <- 1
```

BRCA2, Scenario 2

```
FIG_S2_C<-NULL
FIG_S2_C$Votes <- nvm.mcc.2b$votes
FIG_S2_C$MCC   <- nvm.mcc.2b$mcc
FIG_S2_C$Gene  <- "BRCA2"
FIG_S2_C$Scenario <- 2
```

BRCA1, Scenario 2

```
FIG_S2_D<-NULL
FIG_S2_D$Votes <- nvm.mcc.1b$votes
FIG_S2_D$MCC   <- nvm.mcc.1b$mcc
FIG_S2_D$Gene  <- "BRCA1"
FIG_S2_D$Scenario <- 2


FIG_S2 <- rbind(data.frame(FIG_S2_A),data.frame(FIG_S2_B),data.frame(FIG_S2_C),data.frame(FIG_S2_D))
FIG_S2$Scenario<-as.factor(FIG_S2$Scenario)
ggplot(FIG_S2,aes(x=Votes,y=MCC,colour=Scenario))+geom_point()+geom_line()+ facet_grid(. ~ Gene)+ylim(0
```




```
ggsave("figures/FIG_S2.png",width=9,height=7, units="in",dpi = 600)
```

ROC analysis new models Add model names for new models

```
modelnames2<-c(
"AlignGVGDPrior",
"CADDPhred",
"CADDRaw",
"CADDRawRankScore",
"CAROL",
"CONDEL",
"DannRankScore",
"DannScore",
"FathmmRankScore",
"FathmmScore",
"GerpNr",
"GerpRs",
"GerpRsRankScore",
"LRTConvertedRankScore",
"LRTScore",
"MetaLRRankScore",
"MetaLRScore",
"MetaSVMRankScore",
"MetaSVMScore",
"MutationAssessorRankScore",
"MutationAssessorScore",
"MutationTasterRankScore",
"MutationTasterScore",
"PERCH",
"PERCHnoMAF",
"Phastcons20wayMammalianRank",
"Phastcons7wayVertebrateRank",
"Phylop20wayMammalianRank",
"Phylop7wayVertebrateRank",
```

```
"Polyphen2HdivRankScore",
"Polyphen2HdivScore",
"Polyphen2HvarRankScore",
"Polyphen2HvarScore",
"PROVEANConvertedRankScore",
"PROVEANScore",
"SiftConvertedRankScore",
"SiftScore",
"Siphy29wayLogOddsRank",
"Vest3RankScore",
"Vest3Score",
"NewRFModel",
"NewVoteModel-Training",
"NewVoteModel-Validation",
"NewVoteModel-Both")
```

Determine optimal cutpoints for RF BRCA2: Deleterious/Intermediate vs Neutral (Scenario1)

```
#BRCA1
#Deleterious vs Intermediate/Neutral (Scenario 1)

v2a.41 <- optimal.cutpoints(X="rf_prob_del",status="hdr_del",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=vus.brca2..)


v2a.42 <- optimal.cutpoints(X="sum_del_6",status="hdr_del",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=brca2.votes.train)

v2a.43 <- optimal.cutpoints(X="sum_del_6",status="hdr_del",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=brca2.votes.test)

v2a.44 <- optimal.cutpoints(X="sum_del_6",status="hdr_del",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=brca2)


# Extract performance statistics for new models

list.new <- list(v2a.41, v2a.42, v2a.43, v2a.44)
names(list.new) <- c("v2a.41","v2a.42","v2a.43","v2a.44")

fn <- matrix(NA,nrow=4,ncol=1)
fp <- matrix(NA,nrow=4,ncol=1)
nh <- matrix(NA,nrow=4,ncol=1)
nd <- matrix(NA,nrow=4,ncol=1)
tp <- matrix(NA,nrow=4,ncol=1)
tn <- matrix(NA,nrow=4,ncol=1)
```

```
mcc <- matrix(NA,nrow=4,ncol=1)
cutoff <- matrix(NA,nrow=4,ncol=1)
auc <- matrix(NA,nrow=4,ncol=3)
sens <- matrix(NA,nrow=4,ncol=3)
spec <- matrix(NA,nrow=4,ncol=3)

for (i in 1:length(list.new)) {

auc[i,] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca2a.new.models <- data.frame(modelnames2[41:44],cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                stringsAsFactors = FALSE)
colnames(brca2a.new.models) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                 "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                 "TP","TN","mcc")

}

#kable(brca2a.new.models, caption='Performance of New Models for BRCA2, Scenario 1')
brca2a.new.models$Gene="BRCA2"
brca2a.new.models$Scenario=1


ST4 <-brca2a.new.models
```

BRCA2: Deleterious/Intermediate vs Neutral (Scenario2)

```
v2b.41 <- optimal.cutpoints(X="rf_prob_delint",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=vus.brca2..)

v2b.42 <- optimal.cutpoints(X="sum_nn_7",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=brca2.votes.train)

v2b.43 <- optimal.cutpoints(X="sum_nn_7",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=brca2.votes.test)
```

```
v2b.44 <- optimal.cutpoints(X="sum_nn_7",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=brca2)


# Extract performance statistics for new models

list.new <- list(v2b.41, v2b.42, v2b.43, v2b.44)
names(list.new) <- c("v2b.41","v2b.42","v2b.43","v2b.44")

fn <- matrix(NA,nrow=4,ncol=1)
fp <- matrix(NA,nrow=4,ncol=1)
nh <- matrix(NA,nrow=4,ncol=1)
nd <- matrix(NA,nrow=4,ncol=1)
tp <- matrix(NA,nrow=4,ncol=1)
tn <- matrix(NA,nrow=4,ncol=1)
mcc <- matrix(NA,nrow=4,ncol=1)
cutoff <- matrix(NA,nrow=4,ncol=1)
auc <- matrix(NA,nrow=4,ncol=3)
sens <- matrix(NA,nrow=4,ncol=3)
spec <- matrix(NA,nrow=4,ncol=3)

for (i in 1:length(list.new)) {

auc[i,] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca2b.new.models <- data.frame(modelnames2[41:44],cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                stringsAsFactors = FALSE)
colnames(brca2b.new.models) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                 "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                 "TP","TN","mcc")

}

#kable(brca2b.new.models, caption='Performance of New Models for BRCA2, Scenario 2')
brca2b.new.models$Gene="BRCA2"
brca2b.new.models$Scenario=2
ST4<-rbind(brca2b.new.models,ST4)
```

BRCA1: Deleterious vs Intermediate/Neutral (Scenario 1)

```
v1a.41 <- optimal.cutpoints(X="rf_prob_del",status="hdr_del",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=vus.brca1..)

v1a.42 <- optimal.cutpoints(X="sum_del_13",status="hdr_del",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                            ci.fit=TRUE,data=brca1.votes.train)

v1a.43 <- optimal.cutpoints(X="sum_del_13",status="hdr_del",
                            tag.healthy=0,methods=c("MinValueSp"),
                                      ci.fit=TRUE,data=brca1.votes.test)

v1a.44 <- optimal.cutpoints(X="sum_del_13",status="hdr_del",
                            tag.healthy=0,methods=c("MinValueSp"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull",valueSp=0.89
                            ci.fit=TRUE,data=brca1)

# Extract performance statistics for new models

list.new <- list(v1a.41, v1a.42, v1a.43, v1a.44)
names(list.new) <- c("v1a.41","v1a.42","v1a.43","v1a.44")

fn <- matrix(NA,nrow=4,ncol=1)
fp <- matrix(NA,nrow=4,ncol=1)
nh <- matrix(NA,nrow=4,ncol=1)
nd <- matrix(NA,nrow=4,ncol=1)
tp <- matrix(NA,nrow=4,ncol=1)
tn <- matrix(NA,nrow=4,ncol=1)
mcc <- matrix(NA,nrow=4,ncol=1)
cutoff <- matrix(NA,nrow=4,ncol=1)
auc <- matrix(NA,nrow=4,ncol=3)
sens <- matrix(NA,nrow=4,ncol=3)
spec <- matrix(NA,nrow=4,ncol=3)

for (i in 1:1) {

auc[i,] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Se[c(1,3,5)]
spec[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1,3,5)]
fn[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$cutoff[1]

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))
```

```
brca1a.new.models.1 <- data.frame(cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                  stringsAsFactors = FALSE)

}

for (i in 2:2) {

auc[i,] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1a.new.models.2 <- data.frame(cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                  stringsAsFactors = FALSE)

}

for (i in 3:4) {

auc[i,] <- list.new[[i]]$MinValueSp[[1]]$measures.acc$AUC
sens[i,] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- list.new[[i]]$MinValueSp[[1]]$measures.acc$n$h
nd[i,1] <- list.new[[i]]$MinValueSp[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1a.new.models.3 <- data.frame(cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                  stringsAsFactors = FALSE)
}


brca1a.new.models <- cbind(modelnames2[41:44],brca1a.new.models.3)
colnames(brca1a.new.models) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                 "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                 "TP","TN","mcc")
```

```
#kable(brca1a.new.models, caption='Performance of New Models for BRCA1, Scenario 1')
brca1a.new.models$Gene="BRCA1"
brca1a.new.models$Scenario=1
ST4<-rbind(brca1a.new.models,ST4)
```

BRCA1 - Deleterious/Intermediate vs Neutral (Scenario 2)

```
#BRCA1 - Deleterious/Intermediate vs Neutral (Scenario 2)
v1b.41 <- optimal.cutpoints(X="rf_prob_delint",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=vus.brca1..)

v1b.42 <- optimal.cutpoints(X="sum_nn_9",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("SpEqualSe"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull"),
                                      ci.fit=TRUE,data=brca1.votes.train)

v1b.43 <- optimal.cutpoints(X="sum_nn_9",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("MinValueSp"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull",valueSp=0.9)
                                      ci.fit=TRUE,data=brca1.votes.test)

v1b.44 <- optimal.cutpoints(X="sum_nn_9",status="hdr_not_neutral",
                            tag.healthy=0,methods=c("MinValueSp"),
                                      control = control.cutpoints(ci.SeSp = "AgrestiCoull",valueSp=0.85
                                      ci.fit=TRUE,data=brca1)

list.new <- list(v1b.41, v1b.42, v1b.43, v1b.44)
names(list.new) <- c("v1b.41","v1b.42","v1b.43","v1b.44")

fn <- matrix(NA,nrow=4,ncol=1)
fp <- matrix(NA,nrow=4,ncol=1)
nh <- matrix(NA,nrow=4,ncol=1)
nd <- matrix(NA,nrow=4,ncol=1)
tp <- matrix(NA,nrow=4,ncol=1)
tn <- matrix(NA,nrow=4,ncol=1)
mcc <- matrix(NA,nrow=4,ncol=1)
cutoff <- matrix(NA,nrow=4,ncol=1)
auc <- matrix(NA,nrow=4,ncol=3)
sens <- matrix(NA,nrow=4,ncol=3)
spec <- matrix(NA,nrow=4,ncol=3)

for (i in 1:2) {

auc[i,] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$AUC
sens[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$h
nd[i,1] <- list.new[[i]]$SpEqualSe[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]
```

```
cutoff[i,1] <- list.new[[i]]$SpEqualSe[[1]]$optimal.cutoff$cutoff[1]

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1a.new.models.1 <- data.frame(cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                  stringsAsFactors = FALSE)
}


for (i in 3:4) {

auc[i,] <- list.new[[i]]$MinValueSp[[1]]$measures.acc$AUC
sens[i,] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$Se[c(1:3)]
spec[i,] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$Sp[c(1:3)]
fn[i,1] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$FN[1]
fp[i,1] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$FP[1]
nh[i,1] <- list.new[[i]]$MinValueSp[[1]]$measures.acc$n$h
nd[i,1] <- list.new[[i]]$MinValueSp[[1]]$measures.acc$n$d
tp[i,1] <- nd[i,] - fn[i,]
tn[i,1] <- nh[i,] - fp[i,]

cutoff[i,1] <- list.new[[i]]$MinValueSp[[1]]$optimal.cutoff$cutoff

mcc[i,1] <- (tp[i,]*tn[i,]-fp[i,]*fn[i,])/sqrt((tp[i,]+fp[i,])*
            (tp[i,]+fn[i,])*(tn[i,]+fp[i,])*(tn[i,]+fn[i,]))

brca1b.new.models.2 <- data.frame(cutoff,sens,spec,auc,fn,fp,tp,tn,mcc,
                                  stringsAsFactors = FALSE)
}


brca1b.new.models <- cbind(modelnames2[41:44],brca1b.new.models.2)
colnames(brca1b.new.models) <- c("model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                                 "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                                 "TP","TN","mcc")

#kable(brca1b.new.models, caption='Performance of New Models for BRCA1, Scenario 2')
brca1b.new.models$Gene="BRCA1"
brca1b.new.models$Scenario=2
ST4<-rbind(brca1b.new.models,ST4)
```

Appending new methods to Table S4 and sorting by MCC

```
ST4 <- rbind(ST4_BRCA1_S1,ST4_BRCA1_S2,ST4_BRCA2_S1,ST4_BRCA2_S2,ST4)
ST4 <- ST4[order(ST4$mcc,decreasing = T),]
```

Table 2

Performance of in silico prediction models with optimized thresholds for classification of BRCA1 and BRCA2 missense variants (Subset of table S4)

```
brca1_idx = which(ST4$model != "NewVoteModel-Training" &
                    ST4$model != "NewVoteModel-Both" &
                    ST4$Gene=="BRCA1" &
                    ST4$Scenario ==1)[1:12]
brca2_idx = which(ST4$model != "NewVoteModel-Training" &
                    ST4$model != "NewVoteModel-Both" &
                    ST4$Gene=="BRCA2" &
                    ST4$Scenario ==1)[1:12]
tmp_table <-ST4[c(brca1_idx,brca2_idx),]
tmp_table = tmp_table[which(tmp_table$Scenario==1),]
tmp_table$AUC <-paste(round(tmp_table$auc,2)," (",round(tmp_table$auc.lcl,3),"-",round(tmp_table$auc.uc
tmp_table$FN_FP_TP_TN<-paste(tmp_table$FN,tmp_table$FP,tmp_table$TP,tmp_table$TN,sep=" / ")
tmp_table$ORDER=1
tmp_table$cutpoint<-round(tmp_table$cutpoint,2)
tmp_table$ORDER[which(tmp_table$Gene=="BRCA2")]=0
TABLE_2 = unique(tmp_table[order(tmp_table$ORDER, tmp_table$mcc,decreasing=T),c("Gene","model","cutpoin
knitr::kable(TABLE_2,row.names = F)
```

Gene	model	cutpoint	AUC	FN_FP_TP_TN	mcc
BRCA1	NewVoteModel-Validation	9.00	0.94 (0.897-0.983)	5 / 8 / 25 / 79	0.719
BRCA1	Vest3Rankscore	0.86	0.9 (0.849-0.95)	8 / 24 / 51 / 153	0.678
BRCA1	Vest3Score	0.87	0.9 (0.849-0.95)	8 / 24 / 51 / 153	0.678
BRCA1	NewRFModel	0.30	0.92 (0.879-0.96)	8 / 26 / 51 / 151	0.663
BRCA1	a_gvgd_prior	0.29	0.88 (0.829-0.931)	7 / 37 / 54 / 150	0.614
BRCA1	PERCH_noMAF	0.21	0.87 (0.814-0.924)	10 / 31 / 51 / 156	0.614
BRCA1	PERCH	0.24	0.87 (0.819-0.922)	10 / 32 / 51 / 155	0.607
BRCA1	Polyphen2HvarRankscore	0.92	0.89 (0.845-0.93)	11 / 30 / 48 / 147	0.593
BRCA1	Polyphen2HvarScore	1.00	0.89 (0.845-0.93)	11 / 30 / 48 / 147	0.593
BRCA1	MetasvmRankscore	0.91	0.89 (0.844-0.928)	11 / 34 / 48 / 143	0.565
BRCA1	MetasvmScore	0.52	0.89 (0.844-0.928)	11 / 34 / 48 / 143	0.565
BRCA1	SiftConvertedRankscore	0.91	0.83 (0.786-0.871)	5 / 52 / 54 / 125	0.541
BRCA2	NewVoteModel-Validation	4.00	0.89 (0.826-0.963)	6 / 9 / 29 / 59	0.683
BRCA2	PERCH	0.30	0.89 (0.847-0.939)	11 / 21 / 60 / 115	0.672
BRCA2	PERCH_noMAF	0.27	0.88 (0.832-0.929)	12 / 23 / 59 / 113	0.642
BRCA2	NewRFModel	0.37	0.9 (0.843-0.947)	12 / 24 / 59 / 111	0.633
BRCA2	MetasvmRankscore	0.93	0.87 (0.824-0.923)	15 / 29 / 56 / 107	0.555
BRCA2	MetasvmScore	0.70	0.87 (0.824-0.923)	15 / 29 / 56 / 107	0.555
BRCA2	MetalrRankscore	0.92	0.87 (0.823-0.922)	16 / 30 / 55 / 106	0.535
BRCA2	MetalrScore	0.77	0.87 (0.823-0.922)	16 / 30 / 55 / 106	0.535
BRCA2	Vest3Rankscore	0.80	0.83 (0.776-0.893)	16 / 30 / 55 / 106	0.535
BRCA2	Vest3Score	0.81	0.83 (0.776-0.893)	16 / 30 / 55 / 106	0.535
BRCA2	Polyphen2HvarRankscore	0.97	0.81 (0.757-0.862)	15 / 35 / 56 / 100	0.507
BRCA2	Polyphen2HvarScore	1.00	0.81 (0.757-0.862)	15 / 35 / 56 / 100	0.507

```
write.table(TABLE_2,file="tables/Table_2.tsv",row.names = F,col.names = T,sep="\t")
```

Figure S3

Receiver operating characteristic (ROC) curves for the NVM model applied to the entire data set using rules for BRCA1 and BRCA2 genes.

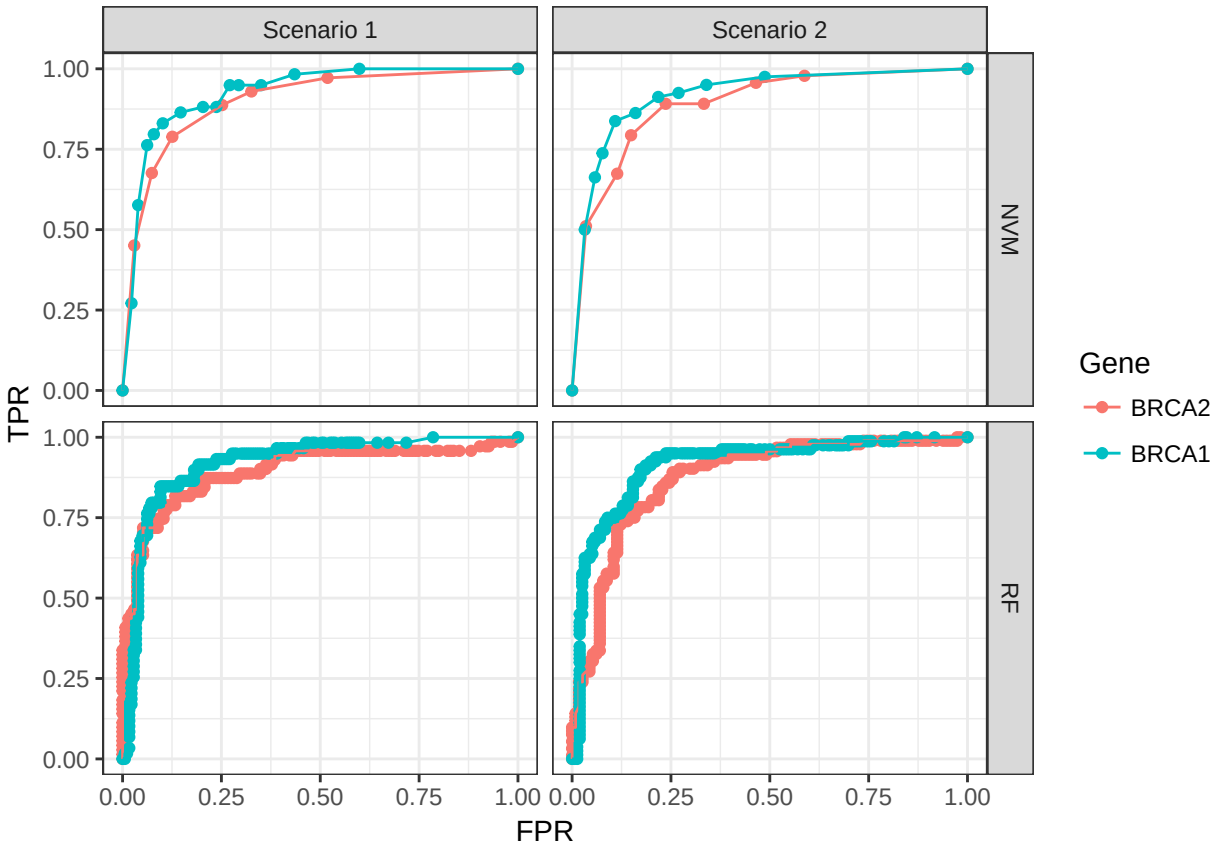
Assemble stats for NVM

```
# Numbers for NVM ROC figure
roc2a.nvm <- prediction(brca2$sum_del_6,brca2$hdr_del)
perf2a.nvm <- performance(roc2a.nvm,"tpr","fpr")
x_val <- unlist(perf2a.nvm@x.values)
y_val <- unlist(perf2a.nvm@y.values)
gene  <- rep("BRCA2",length(x_val))
Scenario  <- rep("Scenario 1",length(x_val))
tmp<- data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario)

roc2b.nvm <- prediction(brca2$sum_nn_7,brca2$hdr_not_neutral)
perf2b.nvm <- performance(roc2b.nvm,"tpr","fpr")
x_val <- unlist(perf2b.nvm@x.values)
y_val <- unlist(perf2b.nvm@y.values)
gene  <- rep("BRCA2",length(x_val))
Scenario  <- rep("Scenario 2",length(x_val))
tmp<- rbind(tmp,data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario))

roc1a.nvm <- prediction(brca1$sum_del_13,brca1$hdr_del)
perf1a.nvm <- performance(roc1a.nvm,"tpr","fpr")
x_val <- unlist(perf1a.nvm@x.values)
y_val <- unlist(perf1a.nvm@y.values)
gene  <- rep("BRCA1",length(x_val))
Scenario  <- rep("Scenario 1",length(x_val))
tmp<- rbind(tmp,data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario))

roc1b.nvm <- prediction(brca1$sum_nn_9,brca1$hdr_not_neutral)
perf1b.nvm <- performance(roc1b.nvm,"tpr","fpr")
x_val <- unlist(perf1b.nvm@x.values)
y_val <- unlist(perf1b.nvm@y.values)
gene  <- rep("BRCA1",length(x_val))
Scenario  <- rep("Scenario 2",length(x_val))
tmp<- rbind(tmp,data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario))
tmp$Model <-"NVM"
```

Assemble stats for RF

```
# Numbers for RF ROC figure
roc2a.rf <- prediction(vus.brca2..$rf_prob_del,vus.brca2..$hdr_del)
perf2a.rf<- performance(roc2a.rf,"tpr","fpr")
x_val <- unlist(perf2a.rf@x.values)
y_val <- unlist(perf2a.rf@y.values)
gene  <- rep("BRCA2",length(x_val))
Scenario  <- rep("Scenario 1",length(x_val))
tmp2<- data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario)
```

```
roc2b.rf <- prediction(vus.brca2..$rf_prob_delint,vus.brca2..$hdr_not_neutral)
perf2b.rf <- performance(roc2b.rf,"tpr","fpr")
x_val <- unlist(perf2b.rf@x.values)
y_val <- unlist(perf2b.rf@y.values)
gene  <- rep("BRCA2",length(x_val))
Scenario  <- rep("Scenario 2",length(x_val))
tmp2<- rbind(tmp2,data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario))

roc1a.rf <- prediction(vus.brca1..$rf_prob_del,vus.brca1..$hdr_del)
perf1a.rf <- performance(roc1a.rf,"tpr","fpr")
x_val <- unlist(perf1a.rf@x.values)
y_val <- unlist(perf1a.rf@y.values)
gene  <- rep("BRCA1",length(x_val))
Scenario  <- rep("Scenario 1",length(x_val))
tmp2<- rbind(tmp2,data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario))


roc1b.rf <- prediction(vus.brca1..$rf_prob_delint,vus.brca1..$hdr_not_neutral)
perf1b.rf <- performance(roc1b.rf,"tpr","fpr")
x_val <- unlist(perf1b.rf@x.values)
y_val <- unlist(perf1b.rf@y.values)
gene  <- rep("BRCA1",length(x_val))
Scenario  <- rep("Scenario 2",length(x_val))
tmp2<- rbind(tmp2,data.frame(FPR=x_val,TPR=y_val,Gene=gene,Scenario))
tmp2$Model <- "RF"

tmp_SF3 <- rbind(tmp,tmp2)
ggplot(tmp_SF3,aes(x=FPR,y=TPR,colour=Gene))+geom_point()+geom_line()+facet_grid(Model ~ Scenario)+theme
```




```
ggsave("figures/FIG_S3.png",height = 9,width = 5,units="in",dpi = 600)
```

Table S4

Sensitivity, Specificity, AUC, and MCC for each gene and each Scenario.

```
brca2a.models.$Gene="BRCA2"
brca2a.models.$Scenario="1"
brca2a.output <- rbind(brca2a.models., brca2a.new.models)

brca2a.output. <- cbind(rep(2,44),rep(1,44),brca2a.output)
colnames(brca2a.output.) <- c("Gene","Scenario","model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                              "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                              "TP","TN","mcc")
brca2a.output.$Gene="BRCA2"

brca2b.models.$Gene="BRCA2"
brca2b.models.$Scenario="2"
brca2b.output <- rbind(brca2b.models., brca2b.new.models)
brca2b.output. <- cbind(rep(2,44),rep(2,44),brca2b.output)
colnames(brca2b.output.) <- c("Gene","Scenario","model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                              "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                              "TP","TN","mcc")
brca2b.output.$Gene="BRCA2"
```

```
brca1a.models.$Gene="BRCA1"
brca1a.models.$Scenario="1"
brca1a.output <- rbind(brca1a.models., brca1a.new.models)
brca1a.output. <- cbind(rep(1,44),rep(1,44),brca1a.output)
colnames(brca1a.output.) <- c("Gene","Scenario","model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                              "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                              "TP","TN","mcc")
brca1a.output.$Gene="BRCA1"

brca1b.models.$Gene="BRCA1"
brca1b.models.$Scenario="2"
brca1b.output <- rbind(brca1b.models., brca1b.new.models)
brca1b.output. <- cbind(rep(1,44),rep(2,44),brca1b.output)
colnames(brca1b.output.) <- c("Gene","Scenario","model","cutpoint","sens","sens.lcl","sens.ucl","spec",
                              "spec.lcl","spec.ucl","auc","auc.lcl","auc.ucl","FN","FP",
                              "TP","TN","mcc")
brca1b.output.$Gene="BRCA1"

ST4 <- rbind(brca2a.new.models,brca2b.new.models,brca1a.new.models,brca1b.new.models,ST4)
```

```
knitr::kable(ST4)
```

	model	cutpoint	sens	sens.lcl	sens.ucl	spec	spec.lcl	spec.ucl	auc
1	NewRFModel	0.371	0.831	0.727	0.901	0.822	0.749	0.878	0.895
2	NewVoteModel-Training	4.000	0.750	0.589	0.862	0.881	0.782	0.938	0.904
3	NewVoteModel-Validation	4.000	0.829	0.673	0.919	0.868	0.767	0.929	0.894
4	NewVoteModel-Both	4.000	0.789	0.680	0.868	0.874	0.808	0.920	0.900
5	NewRFModel	0.437	0.804	0.712	0.873	0.798	0.715	0.862	0.873
6	NewVoteModel-Training	5.000	0.804	0.668	0.893	0.842	0.726	0.915	0.890
7	NewVoteModel-Validation	5.000	0.783	0.644	0.877	0.860	0.747	0.927	0.887
8	NewVoteModel-Both	5.000	0.793	0.700	0.864	0.851	0.774	0.905	0.890
9	NewRFModel	0.298	0.864	0.755	0.930	0.853	0.793	0.898	0.920
10	NewVoteModel-Training	9.000	0.828	0.655	0.924	0.889	0.807	0.939	0.913
11	NewVoteModel-Validation	9.000	0.833	0.653	0.944	0.908	0.827	0.959	0.940
12	NewVoteModel-Both	9.000	0.831	0.715	0.905	0.898	0.845	0.935	0.926
13	NewRFModel	0.378	0.850	0.756	0.912	0.846	0.781	0.894	0.915
14	NewVoteModel-Training	6.000	0.900	0.769	0.960	0.861	0.768	0.920	0.911
15	NewVoteModel-Validation	6.000	0.775	0.625	0.877	0.922	0.840	0.964	0.923
16	NewVoteModel-Both	6.000	0.838	0.742	0.903	0.891	0.832	0.931	0.918
162	NewVoteModel-Training	6.000	0.900	0.769	0.960	0.861	0.768	0.920	0.911
164	NewVoteModel-Both	6.000	0.838	0.742	0.903	0.891	0.832	0.931	0.918
167	NewVoteModel-Validation	9.000	0.833	0.653	0.944	0.908	0.827	0.959	0.940
163	NewVoteModel-Validation	6.000	0.775	0.625	0.877	0.922	0.840	0.964	0.923
168	NewVoteModel-Both	9.000	0.831	0.715	0.905	0.898	0.845	0.935	0.926
175	NewVoteModel-Validation	4.000	0.829	0.673	0.919	0.868	0.767	0.929	0.894
166	NewVoteModel-Training	9.000	0.828	0.655	0.924	0.889	0.807	0.939	0.913
33	Vest3Rankscore	0.855	0.864	0.755	0.930	0.864	0.806	0.907	0.900
34	Vest3Score	0.868	0.864	0.755	0.930	0.864	0.806	0.907	0.900
161	NewRFModel	0.378	0.850	0.756	0.912	0.846	0.781	0.894	0.915
222	PERCH	0.296	0.845	0.743	0.911	0.846	0.775	0.897	0.893
193	PERCH	0.233	0.837	0.748	0.899	0.835	0.756	0.892	0.888
165	NewRFModel	0.298	0.864	0.755	0.930	0.853	0.793	0.898	0.920
176	NewVoteModel-Both	4.000	0.789	0.680	0.868	0.874	0.808	0.920	0.900
331	Vest3Rankscore	0.801	0.838	0.742	0.903	0.840	0.774	0.889	0.883

	model	cutpoint	sens	sens.lcl	sens.ucl	spec	spec.lcl	spec.ucl	auc
341	Vest3Score	0.812	0.838	0.742	0.903	0.840	0.774	0.889	0.883
110	a_gvgd_prior	0.290	0.817	0.720	0.886	0.855	0.794	0.901	0.866
203	PERCH_noMAF	0.192	0.826	0.736	0.890	0.826	0.747	0.885	0.888
170	NewVoteModel-Training	5.000	0.804	0.668	0.893	0.842	0.726	0.915	0.890
172	NewVoteModel-Both	5.000	0.793	0.700	0.864	0.851	0.774	0.905	0.890
171	NewVoteModel-Validation	5.000	0.783	0.644	0.877	0.860	0.747	0.927	0.887
232	PERCH_noMAF	0.272	0.831	0.727	0.901	0.831	0.759	0.885	0.881
211	PERCH_noMAF	0.140	0.829	0.734	0.895	0.831	0.767	0.881	0.850
174	NewVoteModel-Training	4.000	0.750	0.589	0.862	0.881	0.782	0.938	0.904
173	NewRFModel	0.371	0.831	0.727	0.901	0.822	0.749	0.878	0.895
201	PERCH	0.190	0.829	0.734	0.895	0.825	0.760	0.876	0.847
116	a_gvgd_prior	0.290	0.885	0.782	0.943	0.802	0.739	0.853	0.880
21	PERCH_noMAF	0.206	0.836	0.724	0.908	0.834	0.774	0.881	0.869
20	PERCH	0.240	0.836	0.724	0.908	0.829	0.768	0.876	0.870
169	NewRFModel	0.437	0.804	0.712	0.873	0.798	0.715	0.862	0.873
311	SiftConvertedRankscore	0.912	0.862	0.770	0.921	0.763	0.690	0.823	0.831
371	SiftScore	0.000	0.862	0.770	0.921	0.763	0.690	0.823	0.831
28	Polyphen2HvarRankscore	0.916	0.814	0.696	0.893	0.831	0.768	0.879	0.887
29	Polyphen2HvarScore	0.999	0.814	0.696	0.893	0.831	0.768	0.879	0.887
323	Vest3Rankscore	0.783	0.793	0.700	0.864	0.800	0.718	0.863	0.859
333	Vest3Score	0.793	0.793	0.700	0.864	0.800	0.718	0.863	0.859
141	MetasvmRankscore	0.887	0.800	0.700	0.873	0.801	0.732	0.856	0.856
151	MetasvmScore	0.376	0.800	0.700	0.873	0.801	0.732	0.856	0.856
261	Polyphen2HdivRankscore	0.899	0.850	0.756	0.912	0.756	0.683	0.817	0.827
271	Polyphen2HdivScore	1.000	0.850	0.756	0.912	0.756	0.683	0.817	0.827
281	Polyphen2HvarRankscore	0.850	0.788	0.686	0.863	0.808	0.739	0.862	0.871
291	Polyphen2HvarScore	0.997	0.788	0.686	0.863	0.808	0.739	0.862	0.871
101	GerpRs	4.830	0.788	0.686	0.863	0.801	0.732	0.856	0.823
111	GerpRsRankscore	0.617	0.788	0.686	0.863	0.801	0.732	0.856	0.823
144	MetasvmRankscore	0.908	0.814	0.696	0.893	0.808	0.744	0.859	0.886
154	MetasvmScore	0.520	0.814	0.696	0.893	0.808	0.744	0.859	0.886
1611	MetasvmRankscore	0.932	0.789	0.680	0.868	0.787	0.711	0.847	0.873
178	MetasvmScore	0.700	0.789	0.680	0.868	0.787	0.711	0.847	0.873
361	LrtConvertedRankscore	0.385	0.788	0.686	0.863	0.788	0.718	0.845	0.844
391	LrtScore	0.002	0.788	0.686	0.863	0.788	0.718	0.845	0.841
115	MetalrRankscore	0.915	0.772	0.676	0.846	0.774	0.689	0.841	0.842
123	MetalrScore	0.751	0.772	0.676	0.846	0.774	0.689	0.841	0.842
31	SiftConvertedRankscore	0.912	0.915	0.816	0.963	0.706	0.635	0.768	0.828
35	SiftScore	0.000	0.915	0.816	0.963	0.706	0.635	0.768	0.828
142	MetalrRankscore	0.921	0.775	0.665	0.856	0.779	0.703	0.841	0.872
152	MetalrScore	0.768	0.775	0.665	0.856	0.779	0.703	0.841	0.872
352	Vest3Rankscore	0.800	0.775	0.665	0.856	0.779	0.703	0.841	0.834
362	Vest3Score	0.811	0.775	0.665	0.856	0.779	0.703	0.841	0.834
26	Polyphen2HdivRankscore	0.899	0.898	0.795	0.953	0.701	0.629	0.763	0.822
27	Polyphen2HdivScore	1.000	0.898	0.795	0.953	0.701	0.629	0.763	0.822
104	GerpRs	5.060	0.797	0.677	0.880	0.780	0.713	0.834	0.820
117	GerpRsRankscore	0.676	0.797	0.677	0.880	0.780	0.713	0.834	0.820
37	LrtConvertedRankscore	0.410	0.797	0.677	0.880	0.780	0.713	0.834	0.841
39	LrtScore	0.001	0.797	0.677	0.880	0.780	0.713	0.834	0.842
273	Polyphen2HvarRankscore	0.971	0.728	0.630	0.809	0.789	0.706	0.854	0.824
283	Polyphen2HvarScore	1.000	0.728	0.630	0.809	0.789	0.706	0.854	0.824
181	MutationtasterConvertedRankscore	0.810	0.975	0.913	0.993	0.551	0.473	0.627	0.769

	model	cutpoint	sens	sens.lcl	sens.ucl	spec	spec.lcl	spec.ucl	auc
133	MetasvmRankscore	0.922	0.761	0.664	0.836	0.757	0.671	0.826	0.847
143	MetasvmScore	0.625	0.761	0.664	0.836	0.757	0.671	0.826	0.847
301	ProveanConvertedRankscore	0.655	0.762	0.659	0.842	0.769	0.697	0.828	0.868
302	Polyphen2HvarRankscore	0.971	0.789	0.680	0.868	0.741	0.661	0.807	0.810
313	Polyphen2HvarScore	1.000	0.789	0.680	0.868	0.741	0.661	0.807	0.810
210	CaddPhred	24.800	0.762	0.659	0.842	0.763	0.690	0.823	0.838
310	CaddRaw	4.810	0.762	0.659	0.842	0.763	0.690	0.823	0.838
41	CaddRawRankscore	0.645	0.762	0.659	0.842	0.763	0.690	0.823	0.838
30	ProveanConvertedRankscore	0.699	0.780	0.659	0.866	0.780	0.713	0.834	0.874
253	Polyphen2HdivRankscore	0.899	0.957	0.893	0.983	0.509	0.418	0.599	0.737
263	Polyphen2HdivScore	1.000	0.957	0.893	0.983	0.509	0.418	0.599	0.737
221	Phastcons20wayMammalianRankscore	0.475	0.762	0.659	0.842	0.744	0.670	0.806	0.818
43	Condel_score	0.856	0.747	0.641	0.830	0.737	0.643	0.814	0.814
121	MetalrRankscore	0.897	0.750	0.645	0.832	0.750	0.677	0.811	0.828
131	MetalrScore	0.702	0.750	0.645	0.832	0.750	0.677	0.811	0.828
303	SiftConvertedRankscore	0.912	0.848	0.761	0.907	0.626	0.535	0.709	0.762
373	SiftScore	0.000	0.848	0.761	0.907	0.626	0.535	0.709	0.762
124	MetalrRankscore	0.903	0.763	0.640	0.853	0.763	0.695	0.819	0.861
134	MetalrScore	0.717	0.763	0.640	0.853	0.763	0.695	0.819	0.861
321	Siphy29wayLogoddsRankscore	0.485	0.738	0.632	0.821	0.750	0.677	0.811	0.800
1613	MutationassessorRankscore	0.544	0.763	0.640	0.853	0.757	0.689	0.814	0.826
17	MutationassessorScore	1.700	0.763	0.640	0.853	0.757	0.689	0.814	0.826
182	MutationassessorRankscore	0.844	0.732	0.619	0.821	0.748	0.669	0.814	0.851
192	MutationassessorScore	2.695	0.732	0.619	0.821	0.748	0.669	0.814	0.851
23	Phastcons7wayVertebrateRankscore	0.540	0.763	0.640	0.853	0.751	0.683	0.809	0.813
1610	MutationassessorRankscore	0.447	0.738	0.632	0.821	0.737	0.663	0.800	0.827
177	MutationassessorScore	1.390	0.738	0.632	0.821	0.737	0.663	0.800	0.827
231	Phastcons7wayVertebrateRankscore	0.470	0.738	0.632	0.821	0.731	0.656	0.794	0.828
62	Condel_score	0.859	0.746	0.627	0.837	0.713	0.625	0.788	0.804
51	CAROL_score	1.000	0.585	0.477	0.686	0.837	0.774	0.886	0.795
282	Polyphen2HdivRankscore	0.899	0.972	0.903	0.992	0.444	0.363	0.529	0.707
292	Polyphen2HdivScore	1.000	0.972	0.903	0.992	0.444	0.363	0.529	0.707
314	CAROL_score	1.000	0.886	0.797	0.939	0.525	0.428	0.621	0.721
71	DannRankscore	0.672	0.725	0.619	0.811	0.724	0.649	0.788	0.787
81	DannScore	0.995	0.725	0.619	0.811	0.724	0.649	0.788	0.787
153	MutationassessorRankscore	0.835	0.739	0.641	0.818	0.693	0.603	0.770	0.796
1612	MutationassessorScore	2.650	0.739	0.641	0.818	0.693	0.603	0.770	0.796
18	MutationtasterConvertedRankscore	0.810	0.983	0.910	0.997	0.492	0.419	0.565	0.741
92	FathmmConvertedRankscore	0.858	0.718	0.605	0.810	0.721	0.640	0.789	0.822
372	FathmmScore	-2.050	0.718	0.605	0.810	0.721	0.640	0.789	0.822
54	CAROL_score	1.000	0.639	0.514	0.748	0.807	0.745	0.858	0.806
32	Siphy29wayLogoddsRankscore	0.500	0.746	0.622	0.839	0.723	0.653	0.784	0.791
216	CaddPhred	25.200	0.729	0.604	0.826	0.734	0.665	0.794	0.832
316	CaddRaw	4.973	0.729	0.604	0.826	0.729	0.659	0.789	0.833
44	CaddRawRankscore	0.669	0.729	0.604	0.826	0.729	0.659	0.789	0.833
22	Phastcons20wayMammalianRankscore	0.515	0.729	0.604	0.826	0.729	0.659	0.789	0.828
179	MutationtasterConvertedRankscore	0.588	0.739	0.641	0.818	0.670	0.579	0.749	0.767
332	SiftConvertedRankscore	0.912	0.859	0.760	0.922	0.559	0.475	0.640	0.730
402	SiftScore	0.000	0.859	0.760	0.922	0.559	0.475	0.640	0.730
202	MutationtasterConvertedRankscore	0.588	0.789	0.680	0.868	0.632	0.549	0.709	0.753
52	CAROL_score	1.000	0.905	0.807	0.956	0.478	0.389	0.569	0.702
353	LrtScore	0.000	0.728	0.630	0.809	0.652	0.561	0.733	0.724

	model	cutpoint	sens	sens.lcl	sens.ucl	spec	spec.lcl	spec.ucl	auc
382	LrtScore	0.000	0.775	0.665	0.856	0.618	0.534	0.695	0.721
114	a_gvgd_prior	0.810	0.598	0.496	0.692	0.754	0.668	0.824	0.751
83	FathmmConvertedRankscore	0.846	0.674	0.573	0.761	0.678	0.588	0.757	0.773
343	FathmmScore	-1.890	0.674	0.573	0.761	0.678	0.588	0.757	0.773
40	ProveanScore	-0.900	0.695	0.569	0.797	0.701	0.629	0.763	0.770
132	LrtConvertedRankscore	0.843	0.775	0.665	0.856	0.574	0.490	0.654	0.698
401	ProveanScore	-0.830	0.675	0.566	0.768	0.673	0.596	0.742	0.757
93	LrtConvertedRankscore	0.843	0.728	0.630	0.809	0.600	0.509	0.685	0.700
293	ProveanConvertedRankscore	0.420	0.674	0.573	0.761	0.652	0.561	0.733	0.734
363	ProveanScore	-1.780	0.674	0.573	0.761	0.652	0.561	0.733	0.734
61	Condel_score	0.674	0.671	0.563	0.763	0.669	0.594	0.736	0.775
112	a_gvgd_prior	0.810	0.620	0.503	0.724	0.711	0.630	0.781	0.723
74	DannRankscore	0.693	0.678	0.551	0.783	0.678	0.606	0.742	0.754
84	DannScore	0.995	0.678	0.551	0.783	0.678	0.606	0.742	0.754
64	Condel_score	0.689	0.672	0.547	0.777	0.674	0.604	0.737	0.794
322	ProveanConvertedRankscore	0.424	0.662	0.546	0.761	0.654	0.571	0.729	0.711
392	ProveanScore	-1.800	0.662	0.546	0.761	0.654	0.571	0.729	0.711
183	MutationtasterScore	1.000	0.652	0.551	0.742	0.643	0.553	0.725	0.677
223	Phastcons7wayVertebrateRankscore	0.591	0.652	0.551	0.742	0.643	0.553	0.725	0.670
213	MutationtasterScore	1.000	0.690	0.575	0.786	0.618	0.534	0.695	0.683
383	CaddRaw	6.252	0.630	0.528	0.722	0.635	0.544	0.717	0.691
393	CaddRawRankscore	0.863	0.630	0.528	0.722	0.635	0.544	0.717	0.691
315	Siphy29wayLogoddsRankscore	0.760	0.620	0.517	0.712	0.643	0.553	0.725	0.676
212	CaddPhred	29.400	0.634	0.518	0.736	0.640	0.556	0.716	0.684
113	GerpRs	5.280	0.606	0.489	0.711	0.662	0.579	0.736	0.688
122	GerpRsRankscore	0.740	0.606	0.489	0.711	0.662	0.579	0.736	0.688
214	CaddPhred	28.900	0.630	0.528	0.722	0.626	0.535	0.709	0.690
312	CaddRaw	6.295	0.634	0.518	0.736	0.632	0.549	0.709	0.685
42	CaddRawRankscore	0.869	0.634	0.518	0.736	0.632	0.549	0.709	0.685
73	GerpRs	5.190	0.641	0.539	0.732	0.609	0.517	0.693	0.692
103	GerpRsRankscore	0.713	0.641	0.539	0.732	0.609	0.517	0.693	0.692
233	Phylop20wayMammalianRankscore	0.612	0.630	0.528	0.722	0.617	0.526	0.701	0.662
342	Siphy29wayLogoddsRankscore	0.783	0.620	0.503	0.724	0.618	0.534	0.695	0.679
251	Phylop7wayVertebrateRankscore	0.605	0.538	0.429	0.643	0.692	0.616	0.759	0.724
252	Phastcons7wayVertebrateRankscore	0.610	0.606	0.489	0.711	0.625	0.541	0.702	0.638
243	Phylop7wayVertebrateRankscore	0.766	0.500	0.400	0.600	0.704	0.615	0.780	0.665
63	GerpNr	5.490	0.609	0.507	0.702	0.591	0.500	0.677	0.552
241	Phylop20wayMammalianRankscore	0.488	0.600	0.490	0.700	0.609	0.531	0.682	0.683
91	GerpNr	5.210	0.625	0.515	0.723	0.577	0.498	0.652	0.694
262	Phylop20wayMammalianRankscore	0.633	0.577	0.462	0.685	0.618	0.534	0.695	0.653
25	Phylop7wayVertebrateRankscore	0.605	0.542	0.417	0.663	0.667	0.594	0.732	0.713
191	MutationtasterScore	1.000	1.000	0.954	1.000	0.090	0.054	0.145	0.545
403	DannScore	0.998	0.587	0.485	0.682	0.591	0.500	0.677	0.623
94	GerpNr	5.220	0.576	0.449	0.694	0.621	0.548	0.690	0.682
53	DannRankscore	0.843	0.587	0.485	0.682	0.583	0.491	0.669	0.623
272	Phylop7wayVertebrateRankscore	0.766	0.493	0.380	0.607	0.669	0.586	0.743	0.646
102	GerpNr	5.490	0.606	0.489	0.711	0.559	0.475	0.640	0.540
19	MutationtasterScore	1.000	1.000	0.939	1.000	0.079	0.048	0.128	0.540
242	Phastcons20wayMammalianRankscore	0.602	0.577	0.462	0.685	0.566	0.482	0.647	0.622
24	Phylop20wayMammalianRankscore	0.488	0.576	0.449	0.694	0.576	0.503	0.647	0.686
215	Phastcons20wayMammalianRankscore	0.602	0.554	0.453	0.652	0.574	0.483	0.660	0.629
72	DannRankscore	0.846	0.563	0.448	0.673	0.559	0.475	0.640	0.600

	model	cutpoint	sens	sens.lcl	sens.ucl	spec	spec.lcl	spec.ucl	auc
82	DannScore	0.998	0.563	0.448	0.673	0.559	0.475	0.640	0.600
36	FathmmConvertedRankscore	0.882	0.475	0.353	0.600	0.452	0.380	0.526	0.517
38	FathmmScore	-2.360	0.475	0.353	0.600	0.452	0.380	0.526	0.517
351	FathmmConvertedRankscore	0.883	0.425	0.323	0.534	0.442	0.367	0.521	0.484
381	FathmmScore	-2.380	0.425	0.323	0.534	0.442	0.367	0.521	0.484

```
#write.csv(tableS4,file='TableS4.csv')
```

Output dataset with predictions for variants with functional data

```
# Merge random forest prediction into dataset with vote model prediciton

brca2.rf.prob <- vus.brca2..[,c("CAVA_CSN","rf_prob_del","rf_prob_delint")]
brca2. <- merge(brca2,brca2.rf.prob, by="CAVA_CSN", all=TRUE)

brca1.rf.prob <- vus.brca1..[,c("CAVA_CSN","rf_prob_del","rf_prob_delint")]
brca1. <- merge(brca1,brca1.rf.prob, by="CAVA_CSN", all=TRUE)

#BRCA2, dichotomomize random forest probability
brca2.$rf_pred_del <- ifelse(brca2.$rf_prob_del>=0.371,1,0)
brca2.$rf_pred_delint <- ifelse(brca2.$rf_prob_delint>=0.437,1,0)

#BRCA2, dichotomize vote model sum
brca2.$nvm_pred_del <- ifelse(brca2.$sum_del_6>=4,1,0)
brca2.$nvm_pred_delint <- ifelse(brca2.$sum_nn_7>=5,1,0)


#BRCA1, dichotomomize random forest probability
brca1.$rf_pred_del <- ifelse(brca1.$rf_prob_del>=0.298,1,0)
brca1.$rf_pred_delint <- ifelse(brca1.$rf_prob_delint>=0.378,1,0)

#BRCA1, dichotomize vote model sum
brca1.$nvm_pred_del <- ifelse(brca1.$sum_del_13>=9,1,0)
brca1.$nvm_pred_delint <- ifelse(brca1.$sum_nn_9>=6,1,0)

#write.csv(brca2.,file="brca2_with_function.csv")
#write.csv(brca1.,file="brca1_with_function.csv")
```

Table S2

All variants, including those with functional data used in this study

```
#BRCA2, Add NVM

all2..$perch_del <- ifelse(all2..$PERCH>=0.295957,1,0)
all2..$metasvmrank_del <- ifelse(all2..$MetasvmRankscore>=0.93181,1,0)
all2..$metalrrank_del <- ifelse(all2..$MetalrRankscore>=0.92107,1,0)
all2..$mutationassessor_del <- ifelse(all2..$MutationassessorRankscore>=0.84368,1,0)
all2..$vest3rank_del <- ifelse(all2..$Vest3Rankscore>=0.79963,1,0)
all2..$pp2hvarrank_del <- ifelse(all2..$Polyphen2HvarRankscore>=0.97092,1,0)
```

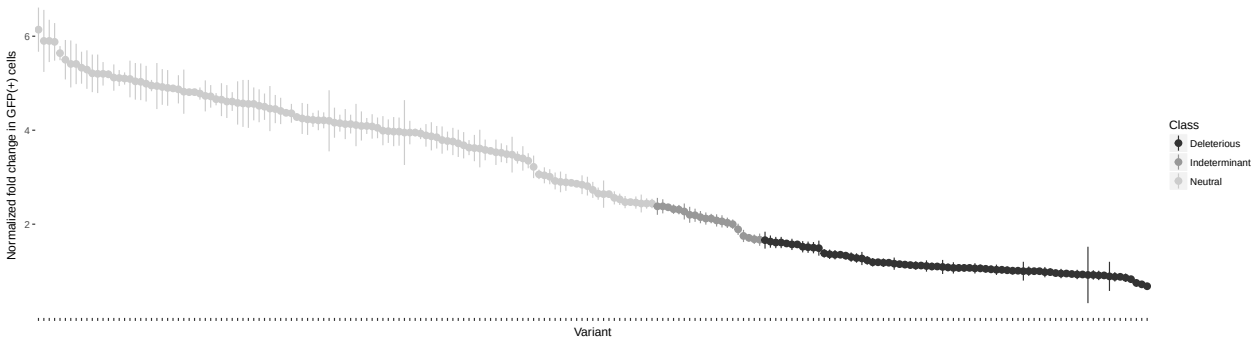
```
all2..$perch_nn <- ifelse(all2..$PERCH>=0.232796,1,0)
all2..$metasvmrank_nn <- ifelse(all2..$MetasvmRankscore>=0.92233,1,0)
all2..$metalrrank_nn <- ifelse(all2..$MetalrRankscore>=0.91514,1,0)
all2..$pp2hdiv_nn <- ifelse(all2..$Polyphen2HdivRankscore>=0.89865,1,0)
all2..$vest3rank_nn <- ifelse(all2..$Vest3Rankscore>=0.7831,1,0)
all2..$pp2hvarrank_nn <- ifelse(all2..$Polyphen2HvarRankscore>=0.97092,1,0)
all2..$sift_nn <- ifelse(all2..$SiftConvertedRankscore>=0.91219,1,0)

all2..$sum_del_6 = with(all2.., perch_del+metasvmrank_del+metalrrank_del+vest3rank_del+pp2hvarrank_del+
                    mutationassessor_del)

all2..$sum_nn_7 = with(all2.., perch_nn+vest3rank_nn+metasvmrank_nn+metalrrank_nn+pp2hvarrank_nn+
                   pp2hdiv_nn+sift_nn)


all2..$nvm_pred_del <- ifelse(all2..$sum_del_6>=4,1,0)
all2..$nvm_pred_delint <- ifelse(all2..$sum_nn_7>=5,1,0)

#BRCA2, dichotomomize random forest probability

all2..$rf_pred_del <- ifelse(all2..$rf_prob_del>=0.371,1,0)
all2..$rf_pred_delint <- ifelse(all2..$rf_prob_delint>=0.437,1,0)

#write.csv(all2..,file='all_vus_brca2.csv')

all1..$perch_del <- ifelse(all1..$PERCH_noMAF>=0.206316,1,0)
all1..$metasvmrank_del <- ifelse(all1..$MetasvmRankscore>=0.9083,1,0)
all1..$metalrrank_del <- ifelse(all1..$MetalrRankscore>=0.90291,1,0)
all1..$mutationassessor_del <- ifelse(all1..$MutationassessorRankscore>=0.54432,1,0)
all1..$vest3rank_del <- ifelse(all1..$Vest3Rankscore>=0.85546,1,0)
all1..$pp2hvarrank_del <- ifelse(all1..$Polyphen2HvarRankscore>=0.91584,1,0)
all1..$sift_del <- ifelse(all1..$SiftConvertedRankscore>=0.91219,1,0)
all1..$agvgd_del <- ifelse(all1..$a_gvgd_prior>=0.29,1,0)
all1..$lrt_del <- ifelse(all1..$LrtScore<=0.000926,1,0)
all1..$provean_del <- ifelse(all1..$ProveanConvertedRankscore>=0.69906,1,0)
all1..$pp2hdiv_del <- ifelse(all1..$Polyphen2HdivRankscore>=0.89865,1,0)
all1..$gerprs_del <- ifelse(all1..$GerpRsRankscore>=0.67646,1,0)
all1..$phastconsvertebrate_del <- ifelse(all1..$Phastcons7wayVertebrateRankscore>=0.54028,1,0)

all1..$vest3rank_nn <- ifelse(all1..$Vest3Rankscore>=0.80056,1,0)
all1..$agvgd_nn <- ifelse(all1..$a_gvgd_prior>=0.29,1,0)
all1..$perch_nn <- ifelse(all1..$PERCH_noMAF>=0.140337,1,0)
all1..$sift_nn <- ifelse(all1..$SiftConvertedRankscore>=0.91219,1,0)
all1..$metasvmrank_nn <- ifelse(all1..$MetasvmRankscore>=0.8874,1,0)
all1..$pp2hdiv_nn <- ifelse(all1..$Polyphen2HdivRankscore>=0.89865,1,0)
all1..$pp2hvarrank_nn <- ifelse(all1..$Polyphen2HvarRankscore>=0.85043,1,0)
all1..$gerprs_nn <- ifelse(all1..$GerpRsRankscore>=0.61659,1,0)
all1..$lrt_nn <- ifelse(all1..$LrtConvertedRankscore>=0.38483,1,0)


all1..$sum_del_13 = with(all1.., vest3rank_del+perch_del+agvgd_del+pp2hvarrank_del+metasvmrank_del+
                     sift_del+pp2hdiv_del+gerprs_del+lrt_del+provean_del+mutationassessor_del+
                     metalrrank_del+phastconsvertebrate_del)
```

```
all1..$sum_nn_9 = with(all1.., vest3rank_nn+agvgd_nn+perch_nn+sift_nn+metasvmrank_nn+pp2hdiv_nn+
                  pp2hvarrank_nn+gerprs_nn+lrt_nn)

all1..$nvm_pred_del <- ifelse(all1..$sum_del_13>=9,1,0)
all1..$nvm_pred_delint <- ifelse(all1..$sum_nn_9>=6,1,0)

#BRCA1, dichotomomize random forest probability

all1..$rf_pred_del <- ifelse(all1.$rf_prob_del>=0.298,1,0)
all1..$rf_pred_delint <- ifelse(all1..$rf_prob_delint>=0.378,1,0)

write.table(all1..,file='tables/Table_S2.tsv',row.names = F,sep="\t")
```

Figure 1

HDR activity of 207 BRCA2 missense variants

```
functional_data <-read.csv("./sources/functional_activity_scores.tsv",header=T,sep="\t")

lev<- (functional_data$Variant[order(functional_data$Fold,decreasing = T)])
functional_data$Variant<- factor(functional_data$Variant,lev)
functional_data<-functional_data[order(functional_data$Fold,decreasing = T),]

#Manually add lines
line1 <- which(functional_data$Fold<1.66)
line1 = functional_data$Variant[line1[1]]
line2 <- which(functional_data$Fold>2.41)
line2 <- functional_data$Variant[tail(line2,n=1)]

#Set status
functional_data$Class<-"Deleterious"
functional_data$Class[which(functional_data$Fold>1.66)]<-"Indeterminant"
functional_data$Class[which(functional_data$Fold>2.41)]<-"Neutral"

ggplot(functional_data,aes(x=Variant,y=Fold, ymin=MIN, ymax=MAX,order=Fold,colour=Class))+
  geom_pointrange()+
  theme(axis.text.x = element_blank(),panel.background = element_rect(fill="white"))+
  ylab("Normalized fold change in GFP(+) cells")+ scale_colour_grey()
```

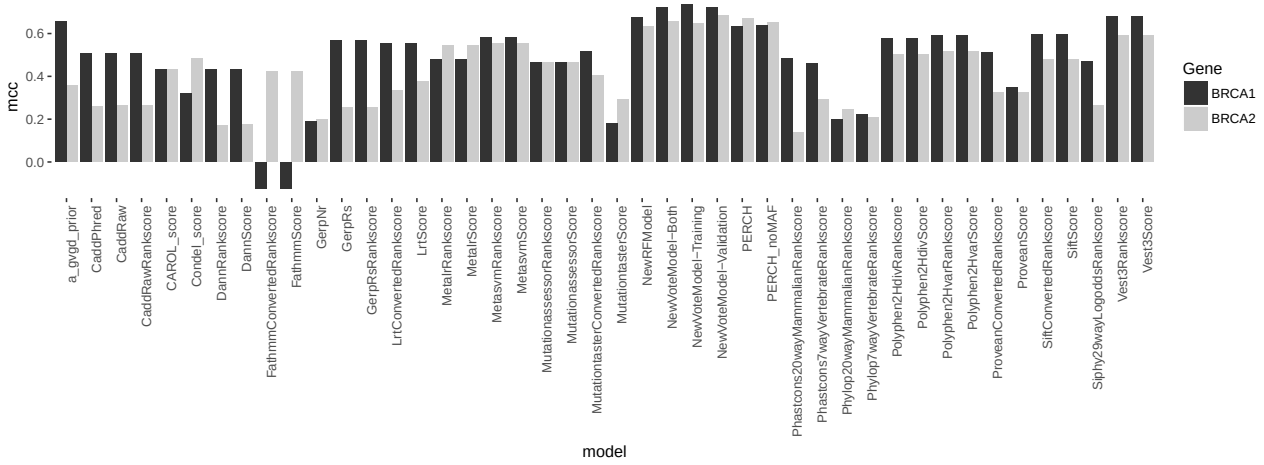


```
ggsave("figures/FIGURE_1.png",width=9,height=7, units="in",dpi = 600)
```

Figure 3

MCC values for optimized thresholds

```
ggplot(ST4,aes(x=model,y=mcc,fill=Gene))+ scale_colour_grey()+scale_fill_grey()+
  geom_bar(stat="identity",position="dodge")+
  theme(axis.text.x = element_text(angle = 90, hjust = 1),panel.background = element_rect(fill="white"))
```




```
ggsave("figures/FIGURE_3.png",width=9,height=7, units="in",dpi = 600)
```

Figure 5

Estimates of the proportion of pathogenic missense mutations along each gene.

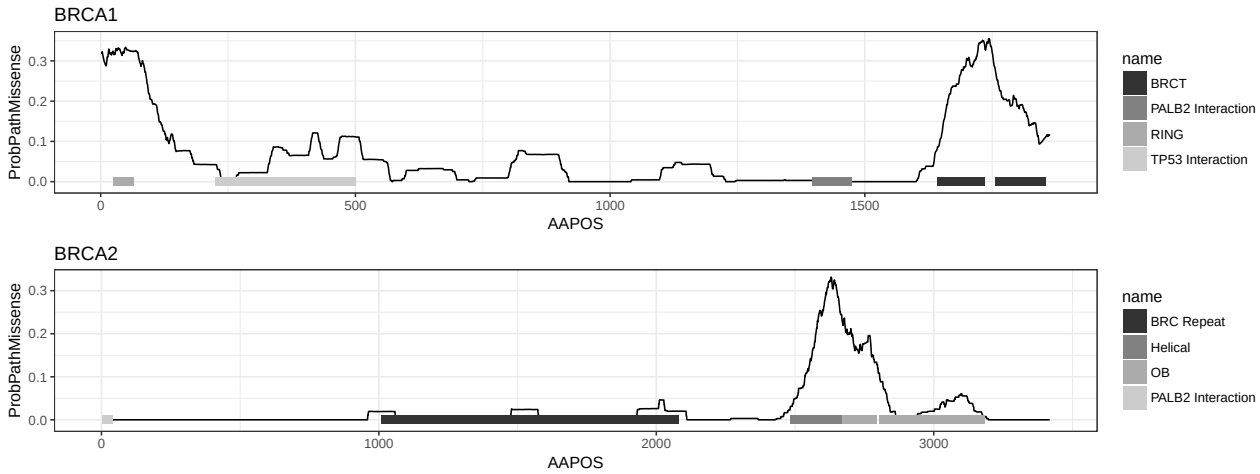
The x-axis is the amino acid position, and the y-axis is the probability of a missense mutation being damaging from the NVM model. The lines were smoothed using a 50 amino acid sliding window.

```
BRCA1_All_missense <- all1..[which(all1..$CAVA_GENE=="BRCA1"),c("AAPOS","nvm_pred_del")]
BRCA2_All_missense <- all2..[which(all2..$CAVA_GENE=="BRCA2"),c("AAPOS","nvm_pred_del")]

# reorder BRCA1
BRCA1_All_missense <- BRCA1_All_missense[order(BRCA1_All_missense$AAPOS,decreasing = F),]

#Make curve
RES<-NULL
window=50

tmp <- BRCA1_All_missense
for (i in 1:length(tmp$AAPOS)){
  pos<-tmp$AAPOS[i]
  if(pos-window < 0){winMin = 0}else{winMin = pos-window}
  winMax<-pos+window
  events<-length(which(tmp$AAPOS>winMin & tmp$AAPOS<winMax))
```

```
  pevents<-length(which(tmp$AAPOS>winMin & tmp$AAPOS<winMax & tmp$nvm_pred_del==1))
  ratio<-pevents/events
  NUM<-c(pos,ratio)
  RES<-rbind(RES,NUM)
}

RES<-data.frame(RES)
names(RES)<-c("AAPOS","ProbPathMissense")


Domains = NULL
Domains = data.frame(
  name=c("RING","TP53 Interaction","PALB2 Interaction","BRCT","BRCT"),
  min_x_pos=c(24,224,1397,1642,1756),
  max_x_pos=c(65,500,1474,1736,1855),
  min_y_pos=c(-0.01,-0.01,-0.01,-0.01,-0.01),
  max_y_pos=c(0.01,0.01,0.01,0.01,0.01)
  )

b1 <- ggplot(data=RES,aes(x=AAPOS,y=ProbPathMissense))+
  geom_line()+
  ggtitle("BRCA1")+
  geom_rect(inherit.aes = FALSE,data=Domains,
            aes(xmin=min_x_pos,xmax=max_x_pos,ymin=min_y_pos,ymax=max_y_pos,fill=name))+
  theme_bw()+scale_fill_grey()
```

```
#Make curve
RES<-NULL
window=50

tmp <- BRCA2_All_missense
for (i in 1:length(tmp$AAPOS)){
  pos<-tmp$AAPOS[i]
  if(pos-window < 0){winMin = 0}else{winMin = pos-window}
  winMax<-pos+window
  events<-length(which(tmp$AAPOS>winMin & tmp$AAPOS<winMax))
  pevents<-length(which(tmp$AAPOS>winMin & tmp$AAPOS<winMax & tmp$nvm_pred_del==1))
  ratio<-pevents/events
  NUM<-c(pos,ratio)
  RES<-rbind(RES,NUM)
}

RES<-data.frame(RES)
names(RES)<-c("AAPOS","ProbPathMissense")


Domains = NULL
Domains = data.frame(
  name=c("PALB2 Interaction","BRC Repeat","Helical","OB","OB","OB"),
  min_x_pos=c(1, 1009,2481,2670,2804,3057 ),
  max_x_pos=c(39,2083,2667,2793,3054,3186 ),
  min_y_pos=c(-0.01,-0.01,-0.01,-0.01,-0.01,-0.01),
  max_y_pos=c(0.01,0.01,0.01,0.01,0.01,0.01)
  )
```

```
b2<-ggplot(data=RES,aes(x=AAPOS,y=ProbPathMissense))+
  geom_line()+
  ggtitle("BRCA2")+
  geom_rect(inherit.aes = FALSE,data=Domains,
            aes(xmin=min_x_pos,xmax=max_x_pos,ymin=min_y_pos,ymax=max_y_pos,fill=name))+
  theme_bw()+scale_fill_grey()


FIGURE_5 = grid.arrange(b1,b2)
```




```
ggsave("figures/FIGURE_5.png",width=9,height=7, units="in",plot = FIGURE_5,dpi = 600)
#grid.arrange(b1,b2)
```

Table S5

Summary of vote model rules for functional class predictions.

BRCA1 - Scenario 1

```
row="PERCH_noMAF>=0.206316"
row=rbind(row,"MetasvmRankscore >= 0.9083")
row=rbind(row,"MetalrRankscore >= 0.90291")
row=rbind(row,"MutationassessorRankscore >= 0.54432")
row=rbind(row,"Vest3Rankscore >= 0.85546")
row=rbind(row,"FathmmConvertedRankscore >= 0.88158")
row=rbind(row,"Polyphen2HvarRankscore >= 0.91584")
row=rbind(row,"MutationtasterConvertedRankscore >= 0.81033")
row=rbind(row,"SiftConvertedRankscore >= 0.91219")
df = data.frame(row)
names(df)<-"BRCA1 - Scenario 1"
knitr::kable(df,row.names = FALSE)
```

BRCA1 - Scenario 1
PERCH_noMAF>=0.206316
MetasvmRankscore >= 0.9083
MetalrRankscore >= 0.90291
MutationassessorRankscore >= 0.54432

BRCA1 - Scenario 1
Vest3Rankscore >= 0.85546
FathmmConvertedRankscore >= 0.88158
Polyphen2HvarRankscore >= 0.91584
MutationtasterConvertedRankscore >= 0.81033
SiftConvertedRankscore >= 0.91219

```
row="PERCH_noMAF >= 0.140337"
row=rbind(row,"MetasvmRankscore >= 0.8874")
row=rbind(row,"MetalrRankscore >= 0.89743")
row=rbind(row,"MutationassessorRankscore >= 0.44691")
row=rbind(row,"Vest3Rankscore >= 0.80056")
row=rbind(row,"FathmmConvertedRankscore >=0.88313")
df = data.frame(row)
names(df)<-"BRCA1 - Scenario 2"
knitr::kable(df,row.names = FALSE)
```

BRCA1 - Scenario 2
PERCH_noMAF >= 0.140337
MetasvmRankscore >= 0.8874
MetalrRankscore >= 0.89743
MutationassessorRankscore >= 0.44691
Vest3Rankscore >= 0.80056
FathmmConvertedRankscore >=0.88313

```
row="PERCH>=0.295957"
row=rbind(row,"MetasvmRankscore >= 0.93181")
row=rbind(row,"MetalrRankscore >= 0.92107")
row=rbind(row,"MutationassessorRankscore >= 0.84368")
row=rbind(row,"Vest3Rankscore >= 0.79963")
row=rbind(row,"Polyphen2HvarRankscore >= 0.97092")
df = data.frame(row)
names(df)<-"BRCA1 - Scenario 1"
knitr::kable(df,row.names = FALSE)
```

BRCA1 - Scenario 1
PERCH>=0.295957
MetasvmRankscore >= 0.93181
MetalrRankscore >= 0.92107
MutationassessorRankscore >= 0.84368
Vest3Rankscore >= 0.79963
Polyphen2HvarRankscore >= 0.97092

```
row="PERCH >= 0.232796"
row=rbind(row,"MetasvmRankscore >= 0.92233")
row=rbind(row,"MetalrRankscore >= 0.91514")
row=rbind(row,"MutationassessorRankscore >= 0.83541")
row=rbind(row,"Vest3Rankscore >= 0.7831")
df = data.frame(row)
names(df)<-"BRCA2 - Scenario 2"
```

```
knitr::kable(df,row.names = FALSE)
```

BRCA2 - Scenario 2
PERCH >= 0.232796
MetasvmRankscore >= 0.92233
MetalrRankscore >= 0.91514
MutationassessorRankscore >= 0.83541
Vest3Rankscore >= 0.7831

Session info

```
sessionInfo()
```

```
## R version 3.2.3 (2015-12-10)
## Platform: x86_64-pc-linux-gnu (64-bit)
## Running under: CentOS release 6.9 (Final)
##
## locale:
##  [1] LC_CTYPE=en_US.UTF-8       LC_NUMERIC=C
##  [3] LC_TIME=en_US.UTF-8        LC_COLLATE=en_US.UTF-8
##  [5] LC_MONETARY=en_US.UTF-8    LC_MESSAGES=en_US.UTF-8
##  [7] LC_PAPER=en_US.UTF-8       LC_NAME=C
##  [9] LC_ADDRESS=C               LC_TELEPHONE=C
## [11] LC_MEASUREMENT=en_US.UTF-8 LC_IDENTIFICATION=C
##
## attached base packages:
## [1] stats     graphics  grDevices utils     datasets  methods   base
##
## other attached packages:
## [1] gridExtra_2.3          ggplot2_2.2.1          e1071_1.6-7
## [4] ROCR_1.0-7             gplots_3.0.1           randomForest_4.6-12
## [7] OptimalCutpoints_1.1-3
##
## loaded via a namespace (and not attached):
##  [1] Rcpp_0.12.13       knitr_1.17         magrittr_1.5
##  [4] munsell_0.4.3      lattice_0.20-34    colorspace_1.2-7
##  [7] rlang_0.1.2        highr_0.6          plyr_1.8.4
## [10] stringr_1.2.0      caTools_1.17.1     tools_3.2.3
## [13] grid_3.2.3         gtable_0.2.0       KernSmooth_2.23-15
## [16] DBI_0.4-1          htmltools_0.3.5    gtools_3.5.0
## [19] class_7.3-14       lazyeval_0.2.0     yaml_2.1.14
## [22] rprojroot_1.2      digest_0.6.12      tibble_1.3.4
## [25] Matrix_1.2-7.1     reshape2_1.4.2     bitops_1.0-6
## [28] evaluate_0.10.1    rmarkdown_1.6      labeling_0.3
## [31] gdata_2.17.0       stringi_1.1.5      scales_0.5.0
## [34] backports_1.0.3
```