DeepTarget

Trinh Nguyen,
Sanju Sinha

May 19, 2023

Contents

1 Deep Target Overview

DeepTarget is a computational tool written in R for deep characterization of cancer drugs' mechanism of action by integrating existing large-scale genetic and drug screens. This user guide will guide users on using functions for DeepTarget's core analysis. The results generated from these analyses can allow one to make predictions about: (1) The primary target(s) of a drug. (2) Whether the drug specifically targets the wild-type (naturally occurring form) or mutated forms of the target. (3) The identification of secondary target(s) that are engaged by the drug when the primary target is not expressed.

2 Run DeepTarget

2.1 Quick start

In this working example, we will analyze using public data from depmap.org.

```
library(DeepTarget) # Load the DeepTarget package
load(onTargetM) # Load the 'onTargetM' dataset
```

2.2 Description of onTargetM

This contains a list of 5 matrices: Drug metadata, Viability scores after Drug Treatment, Gene effect score after knockout, Mutation binary matrix, Expression profile. To retrieve the most updated data, please refer to https://depmap.org/portal/download/all/.

```
sapply(OntargetM,dim)
#DrugMetadata secondary_prism avana_CRISPR mutations_mat expression_20Q4
1448            1618        17453         19350           19182
  5             392          392           392             392
# more detail about version and data link, please do:
?OntargetM
```

3 Core Analysis

The script below outlines the steps to load the necessary data, perform the core analysis, and save the results. We focused on drugs mentioned in the manuscripts for illustration.

For a given drug, the core analysis includes generating a similarity score between the viability after drug treatment and each gene knockout, computing if the drug may differentially bind to the known drug target mutant form or WT form, by calculating this similarity in cell lines with known target WT vs. mutant form, and finally, finding the secondary targets of the drug by repeating this analysis in the cell lines where the primary target is not expressed.

3.1 Data Loading and Preparation

The script loads OntargetM object and prepare the matrix for viability after drug treatment for the drugs that we are interested in and the matrix of Gene effect scores after CRISPR-KO.

```
library(DeepTarget)
data("OntargetM")
sapply(OntargetM,dim)
dir.create(’./result/’)
drug.name <- c(’atiprimod’,’AMG-232’,’pitavastatin’,
’Ro-4987655’,’alexidine’,’RGFP966’,’dabrafenib’,
’olaparib’,’CGM097’,’ibrutinib’,’palbociclib’)
idx.Drug <- which(OntargetM$DrugMetadata$name %in% drug.name)
S.Drug <- OntargetM$DrugMetadata$broad_id_trimmed [idx.Drug]
idx.DrugN <- which(row.names(OntargetM$secondary_prism) %in% S.Drug)
sec.prism.f <- OntargetM$secondary_prism[idx.DrugN, ]
KO.GES <- OntargetM$avana_CRISPR
```

3.2 Computing Similarity Between Drug Treatment and Gene Knockout

The script computes the similarity Between Drug Treatment and Gene Knockout using GetSim function.

```
List.sim <- NULL;
for (i in 1:nrow(sec.prism.f)){
DRS=as.data.frame(sec.prism.f[i,])
DRS <- t(DRS)
row.names(DRS) <- row.names(sec.prism.f)[i]
out <- GetSim(row.names(sec.prism.f)[i],DRS=DRS, GES=KO.GES)
List.sim [[length(List.sim) + 1]] <- out
}
names(List.sim) <- row.names(sec.prism.f)
saveRDS(List.sim,
file = ’Result/similarity_KO_DrugTreatment.RDS’)
```

3.3 Performing Pathway Analysis Based on Drug Meta Data

The script performs the pathway analyis based on the drug metadata using DoPWY function.

```
metadata <- OntargetM$DrugMetadata
Pwy.Enr <- DoPWY(Sim.GES.DRS=List.sim,D.M = metadata)
saveRDS(Pwy.Enr,
file = ’Result/Pwy.Enrichment.RDS’)
```

3.4 Predicting Similarity Across Known Targeted Genes and All Genes

The script predicts similarity Across Known Targeted Genes and All Genes using PredTarget and PredMaxSim functions.

```
DrugTargetSim <- PredTarget(Sim.GES.DRS=List.sim, D.M=metadata)
DrugGeneMaxSim <- PredMaxSim(Sim.GES.DRS=List.sim, D.M=metadata)
```

3.5 Computing the Interaction

The script computes the interaction between the drug and knockout (KO) gene expression in terms of both mutant vs non-mutant and lower vs higher expression.

```
d.mt <- OntargetM$mutations_mat
d.expr <- OntargetM$expression_20Q4
out.MutantTarget <- NULL;
out.LowexpTarget <- NULL;
for (i in 1:nrow(sec.prism.f)){
    DRS=as.data.frame(sec.prism.f[i,])
    DRS <- t(DRS)
    row.names(DRS) <- row.names(sec.prism.f)[i]
    ## for mutant
    Out.M <- DoInteractMutant(Predtargets=DrugTargetSim[i,],
                              Mutant=d.mt,DRS=DRS,GES=KO.GES)
    TargetMutSpecificity <- data.frame(MaxTgt_Inter_Mut_strength=
   sapply(Out.M, function(x) x[1]),
   MaxTgt_Inter_Mut_Pval=sapply(Out.M, function(x) x[2]))
    out.MutantTarget <- rbind (out.MutantTarget,TargetMutSpecificity)
    ## for expression.
    Out.Expr <- DoInteractExp(DrugTargetSim[i,],d.expr,
                              DRS=DRS,GES=KO.GES,CutOff = 2)
    TargetExpSpecificity <- data.frame(
    MaxTgt_Inter_Exp_strength=
    sapply(Out.Expr, function(x) x[1]),
    MaxTgt_Inter_Exp_Pval=
    sapply(Out.Expr, function(x) x[2])
    )
    out.LowexpTarget <- rbind ( out.LowexpTarget,TargetExpSpecificity)
}
```

3.6 Interaction Assessment

This part of the script assesses whether the interaction is true or false based on a certain cut-off, and computes the p-value from a linear model.

```
Whether_interaction_Ex_based= ifelse(
    out.LowexpTarget$MaxTgt_Inter_Exp_strength <0
    & out.LowexpTarget$MaxTgt_Inter_Exp_Pval <0.2,TRUE,FALSE)
predicted_resistance_mut= ifelse(
    out.MutantTarget$MaxTgt_Inter_Mut_Pval<0.1,TRUE,FALSE)
```

3.7 Preparation for Output

Lastly, the script gathers the results into a final data frame and writes it to a CSV file.

```
Pred.d <- NULL;
Pred.d <- cbind(DrugTargetSim,
                  DrugGeneMaxSim,out.MutantTarget,
                  predicted_resistance_mut)
mutant.C <- sapply(Pred.d[,3],function(x)errHandle(sum(d.mt[x,] ==1)))
Pred.d$mutant.C <- mutant.C
Low.Exp.C = sapply(Pred.d[,3],
function(x)errHandle(sum(d.expr[x,] < 2)))
Pred.d <- cbind(Pred.d, out.LowexpTarget,
                  Whether_interaction_Ex_based,Low.Exp)
FileN <- "./Result/Prediction_sim_KO_DrugTreatment_CoreAnalysis.csv"
write.csv (Pred.d, FileN)
```

3.8 Identifying Drugs with low primary target expressing cell lines

For the drugs where the primary target is not expressed in at least five cell lines, we will identify their secondary target below. This section identifies primary target genes that are not expressed in at least 5 cell lines.

```
Low.i <- which(Pred.d$Low.Exp.C >5)
Pred.d.f <- Pred.d[ Low.i,]
Low.Exp.G = sapply(Pred.d.f[,3],
function(x) errHandle(names(which(d.expr[x,]<2))))
identical (names(Low.Exp.G),Pred.d.f[,3] )
sim.LowExp <- NULL;
sec.prism.f.f <- sec.prism.f[Low.i,]
identical (row.names(sec.prism.f.f) ,Pred.d.f[,1])
```

3.9 Calculating Drug KO Similarities in cell lines with low primary target expression

The following script performs calculations to determine DKS score in cell lines with low primary target expression. This DKS score is called Secondary DKS Score and denotes the secondary target probability.

```
for (i in 1:nrow(Pred.d.f)){
    DRS.L= sec.prism.f.f[i,Low.Exp.G[[unlist(Pred.d.f[i,3])]]]
    DRS.L <- t(as.data.frame(DRS.L))
    row.names(DRS.L) <- Pred.d.f[i,1]
    out <- GetSim(Pred.d.f[i,1],DRS=DRS.L, GES=KO.GES)
    sim.LowExp [[length(sim.LowExp) + 1]] <- out
}
names(sim.LowExp) <-Pred.d.f[,1]
```

3.10 Saving the Results

Finally, the results are saved to a file for later use.

```
saveRDS(sim.LowExp,
        file = ’Result/similarity_KO_LowExp_DrugTreatment.RDS’)
```

4 Application

For this section, we will use the information obtained from Core analysis.

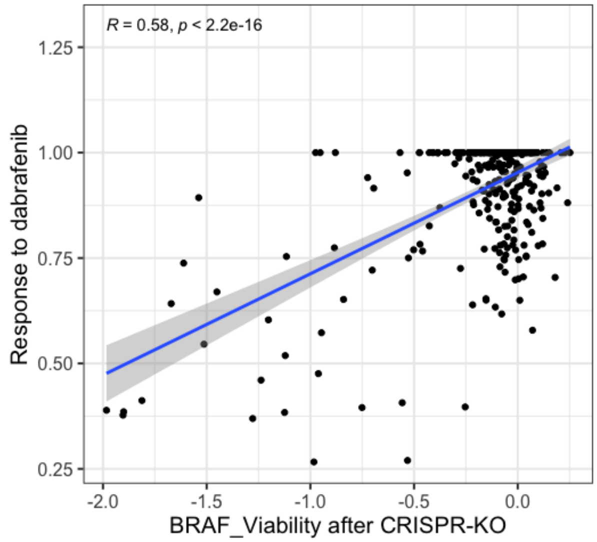
4.1 Finding a drug’s primary target(s)

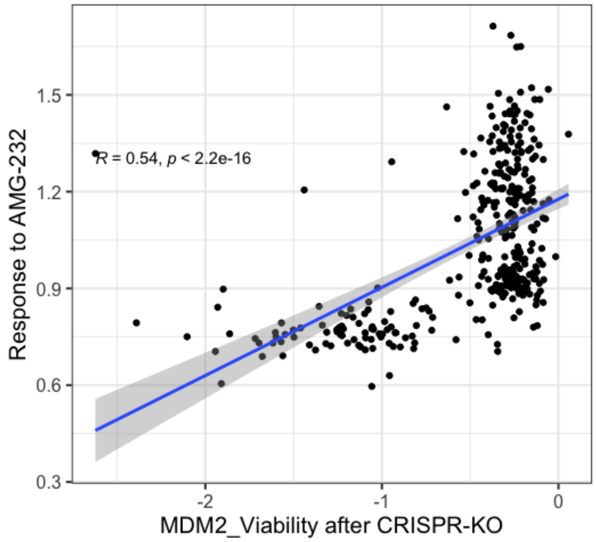
The script below generates the correlation plots for primary target BRAF and MDM2 for Dabrafenib and AMG-232 respectively.

```
Drug_P_targets <- Pred.d$MaxTargetName
DeepTarget_P <- Pred.d$BestTargetGene
idx <- which (Drug_P_targets==DeepTarget_P)
Pred.d[idx,c(2:6)]
drugName MaxTargetName   Maxcorr    KnownTargetCorrP    KnownTargetCorrFDR
dabrafenib     BRAF      0.2555770     1.736879e-06        3.031375e-02
dabrafenib     BRAF      0.5758545     1.736879e-06        3.031375e-02
palbociclib    CDK6      0.2778016     3.605860e-06        5.283088e-03
AMG-232        MDM2      0.5362421     6.787674e-28        5.923264e-24
CGM097         MDM2      0.4952637     5.347038e-23        9.332186e-19

> Pred.d[idx,c(2,8:11)]
```

```
drugName        BestTargetGene BestTargetCorr BestTargetCorrP BestTargetCorrFDR
dabrafenib       BRAF            0.2555770     1.736879e-06      3.031375e-02
dabrafenib       BRAF            0.5758545     1.736879e-06      3.031375e-02
palbociclib      CDK6            0.2778016     3.605860e-06      5.283088e-03
AMG-232          MDM2           0.5362421      6.787674e-28      5.923264e-24
CGM097           MDM2            0.4952637     5.347038e-23      9.332186e-19
##
DOI = 'dabrafenib'
GOI = 'BRAF'
## due to duplicated assay prepare data.
identical(Pred.d$DrugID, row.names(sec.prism.f))
DRS=as.data.frame(sec.prism.f[5,])
DRS <- t(DRS)
row.names(DRS) <- row.names(sec.prism.f)[5]
plotCor(DN=DOI,GN=GOI,Pred=Pred.d[5,],DRS= DRS,GES= KO.GES,plot=TRUE);
dev.off()
## This drug is unique in this Prediction data object.
DOI = 'AMG-232'
GOI = 'MDM2'
identical ( Pred.d$DrugID, row.names(sec.prism.f))
plotCor(DN=DOI,GN=GOI,Pred=Pred.d,DRS=sec.prism.f,
GES= KO.GES,plot=TRUE);
dev.off()
```



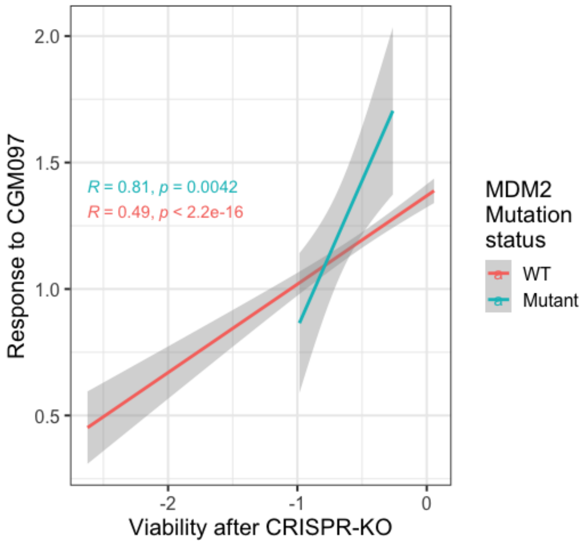


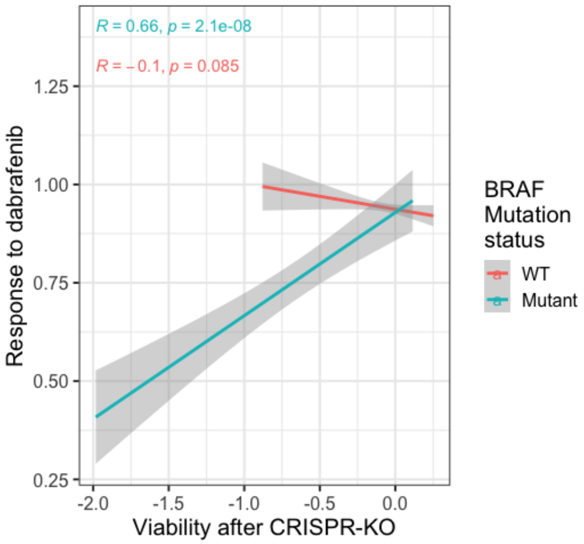
4.2 Predicting whether the drug specifically targets the wild-type or mutated target forms

The script below shows whether the drugs, CGM097 and Dabrafenib, target the wild-type or mutated target forms of MDM2 and BRAF respectively from the Prediction object and then generate the plots for visualization.

```
Pred.d.f <- Pred.d[,c(2,3,12:15)]
Pred.d.f.f <- Pred.d.f[which (Pred.d.f$predicted_resistance_mut==TRUE), ]

drugName MaxTargetName strength strength_Pval predicted_re_mut mutant.C
dabrafenib    BRAF    0.3914397 1.176856e-02          TRUE       65
dabrafenib    BRAF    0.3282302 1.578261e-09          TRUE       65
CGM097        MDM2     0.8120986  2.063320e-02         TRUE      12
## plotting second and third rows.
which.mut <- which (Pred.d$predicted_resistance_mut==TRUE);
## dabrafenib
cr.i <- which.mut[2]
DOI =Pred.d[cr.i,2]
GOI =Pred.d[cr.i,3]
DRS=as.data.frame(sec.prism.f[cr.i,])
DRS <- t(DRS)
row.names(DRS) <- row.names(sec.prism.f)[cr.i]
head(DRS)
out <- DMB(DN=DOI,GN=GOI,Pred=Pred.d[cr.i,],
Mutant=d.mt,DRS= DRS,GES= KO.GES,plot=TRUE)
print (out)
dev.off();
## for drug CGM097
cr.i <- which.mut[3]
DOI=Pred.d[cr.i,2]
GOI=Pred.d[cr.i,3]
DRS=as.data.frame(sec.prism.f[cr.i,])
DRS <- t(DRS)
row.names(DRS) <- row.names(sec.prism.f)[cr.i]
head(DRS)
out <- DMB(DN=DOI,GN=GOI,Pred=Pred.d[cr.i,],
Mutant=d.mt,DRS= DRS,GES= KO.GES,plot=TRUE)
print (out)
dev.off();
###
```





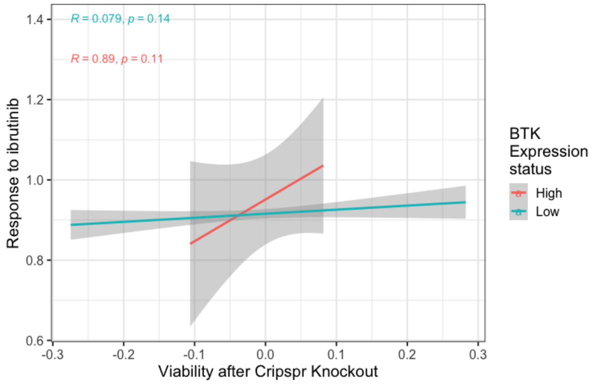
4.3 Predicting secondary target(s) that mediate its response when the primary target is not expressed

As an example, let's look at the drug 'ibrutinib' from the Prediction object. Ibrutinib is a well-known covalent target of BTK. Please note that for this drug, there are two assays. We selected one of them for illustration.

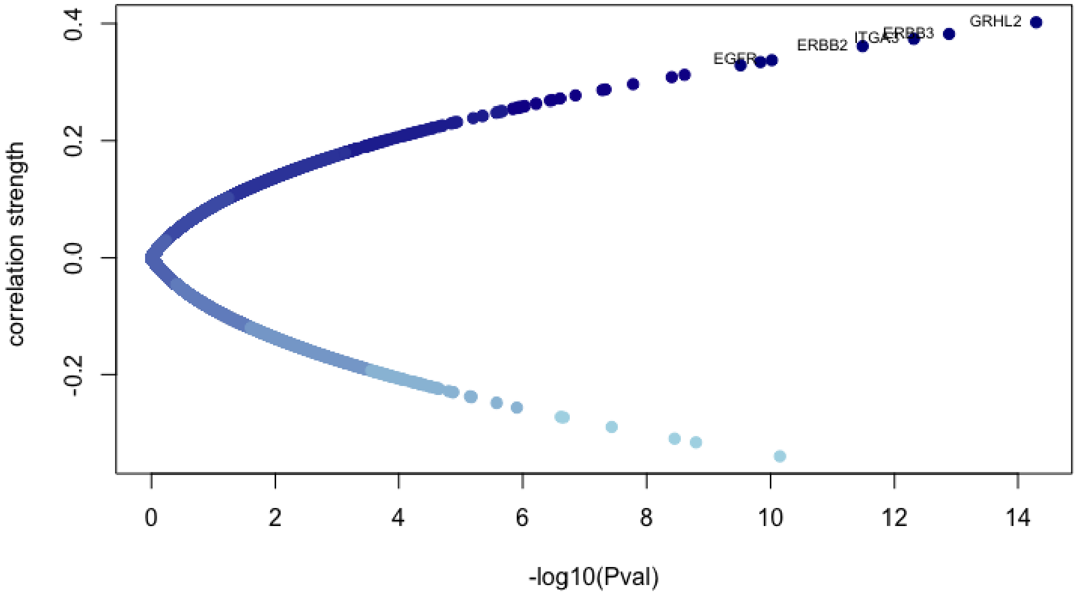
```
## This drug has two assays.
# The index is 14 in the orders of the interesting drugs.
identical (Pred.d$DrugID, row.names(sec.prism.f))
DRS=as.data.frame(sec.prism.f[14,])
DRS <- t(DRS)
row.names(DRS) <- row.names(sec.prism.f)[14]
####
DOI="ibrutinib"
GOI="BTK"
out <- DTR (DN=DOI,GN=GOI,Pred=Pred.d[14,], Exp=d.expr
             ,DRS= DRS,GES=KO.GES,CutOff= 2)
print(out)
dev.off()
```

We find above that Ibrutinib's response is only correlated with the BTK gene in cell lines where BTK is expressed and not in cell lines where BTK is not expressed. Next, let's look at the correlation between the BTK gene KO and this drug response in no BTK cell lines to predict the secondary targets for this drug.

```
### Let's look at the object similarity_KO_LowExp_DrugTreatment.RDS
in.f <- 'Result/similarity_KO_LowExp_DrugTreatment.RDS'
sim.LowExp <- readRDS(file = in.f)
```



```
sim.LowExp.Strength=sapply(sim.LowExp, function(x) x[,2])
dim(sim.LowExp.Strength)
sim.LowExp.Pval=sapply(sim.LowExp, function(x) x[,1])
head(sim.LowExp.Pval)
## Let's take a look at ibrutinib
plotSim(dx=sim.LowExp.Pval[,8],dy=sim.LowExp.Strength[,8],
        clr=colorRampPalette(c("lightblue",'darkblue')), plot=TRUE)
dev.off();
```

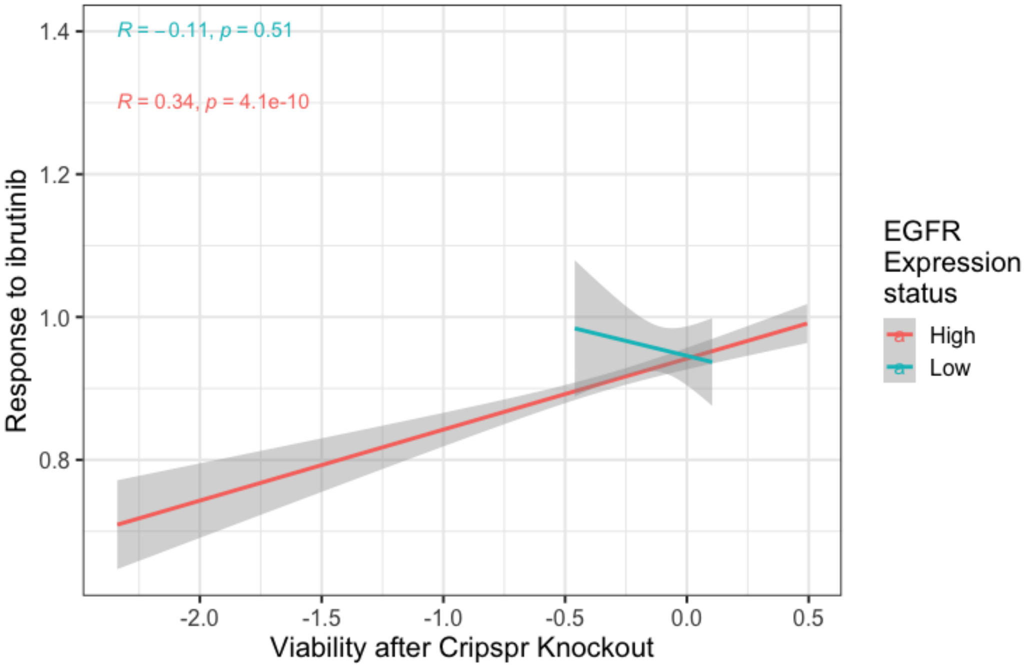


The above ranks all the genes based on their secondary DKS Scores.

4.4 Conclusion

We observe below that Ibrutinib response is strongly correlated with EGFR knockout (KO) in cell lines where Bruton's tyrosine kinase (BTK) is not expressed. However, this correlation is not observed in cell lines where BTK is expressed. Among the top predicted genes, there's more layer of evidence that Ibrutinib binds physically to EGFR. Thus, let's focus further on EGFR.

```
DOI="ibrutinib"
GOI="EGFR"
out <- DTR(DN=DOI,GN=GOI,Pred=Pred.d[14,], Exp=d.expr,
                DRS= DRS,GES=KO.GES,CutOff= 2)
print (out)
dev.off()
```



5 Getting Support

If you have any questions about this package or figures, please contact sanju@terpmail.umd.edu, and tinh.nguyen@nih.gov