The Xeva user's guide

Arvind Mer [1,2] ***and Benjamin Haibe-Kains*** [1,2,3]

[1]Princess Margaret Cancer Centre, University Health Network, Toronto Canada
[2]Department of Medical Biophysics, University of Toronto, Toronto Canada
[3]Department of Computer Science, University of Toronto, Toronto Canada

October 2, 2018

Contents

1 Introduction 2
2 Installation and Settings 2
3 Definations 2
4 Data Access. 3
5 Visualizing PDX Growth Curve 4
6 PDX Model Response 6

1 Introduction

The Xeva package provides efficient and powerfull functions for patient drived xenograft (PDX) based pharmacogenomic data analysis.

2 Installation and Settings

Xeva requires that several packages are installed. However, all dependencies are available from CRAN or Bioconductor.

```
source('http://bioconductor.org/biocLite.R')
biocLite('Xeva')
```

Load Xeva into your current workspace:

```
library(Xeva)
```

Load PDXE breast cancer dataset:

```
data(brca)
print(brca)

## Xeva-set name: PDXE.BRCA
## Creation date: Fri Sep 14 11:41:33 2018
## Number of models: 849
## Number of drugs: 22
## Moleculer dataset: RNASeq, mutation, cnv
```

3 Definations

Before we further dive into the analysis and visualization, it is important to underastand terms used in the *Xeva* package. In a **Xeva** object, the **experiment** slot stores each individual PDX/mouse data. Other then the tumore growth data (time vs. tumor volume), for each individual PDX/mouse we can have meta data such as patient's age, sex, tissue histology, passage infromation etc. All this data is stored using the class **pdxModel** and a unique id called `model.id` is given to each PDX/mouse model. We will see later how to get data for an individual *model.id*.

A PDX experiment can one of the two catagerious:

- **treatment** are the experiments in which PDX recives drug (or drug combination)
- **control** are the experiments where PDX recives no drug

To see the effect of the drug several replicate experiemts are done for control and treatment. In **Xeva** a collection of PDX *model.ids* which are orignetating form same patient are called *batch*. A *batch* have two arms: *control* and *treatment*. This is illustreate in figure 1.

A **Xeva** objects binds togather all individual experiments, batch infroamtion and moleculer data into one single class XevaSet.

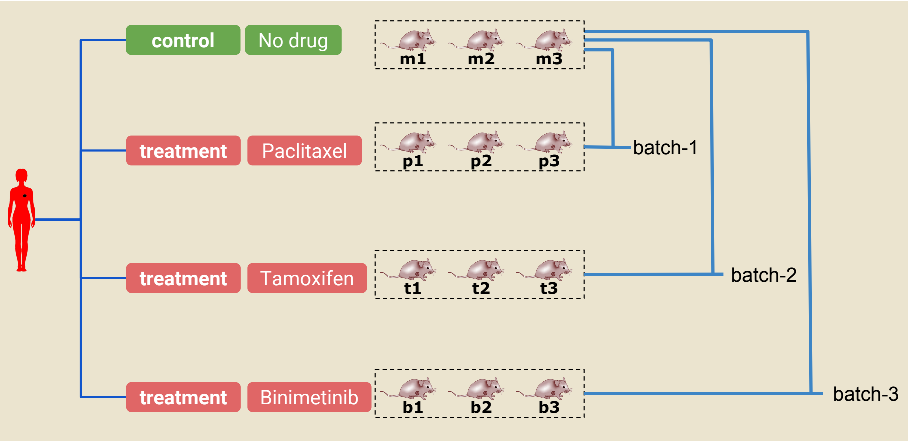

Figure 1: A PDX experiment
Text under the each PDX (e.g. m1, m2, p1 etc.) denotes *model.id* in **Xeva**. In this example three PDX are delclered as control (m1, m2 and m2). Simillerly in a treatment arm 3 PDXs are give drug paclitaxel (p1, p2 and p3). The PDXs in control arm and one of the treatment arm togather consitute a *batch*. For example control arm models (m1, m2, m3) and treatment arm moldes (t1,t2, t3) togather creates a batch called batch-2.

4 Data Access

As mentioned earlier **Xeva** stores data for each individual PDX model. We can retrive meta information about PDX such as number of models and tissue type see the infromation about models using:

```
brca.mod <- modelInfo(brca)
dim(brca.mod)

## [1] 849   5

brca.mod[1:4, ]

##                 model.id tissue   tissue.name patient.id        drug
## X.1004.BG98 X.1004.BG98   BRCA Breast Cancer     X-1004      BGJ398
## X.1004.biib X.1004.biib   BRCA Breast Cancer     X-1004 binimetinib
## X.1004.BK20 X.1004.BK20   BRCA Breast Cancer     X-1004      BKM120
## X.1004.BY19 X.1004.BY19   BRCA Breast Cancer     X-1004      BYL719
```

The output shows that the *brca* dataset contatins 849 PDX models. We can see time vs. tumore volume data for a model as:

```
model.data <- getExperiment(brca, model.id = "X.1004.BG98")
head(model.data)

##      model.id drug.join.name time volume body.weight volume.normal
## 1 X.1004.BG98         BGJ398    0  199.7        28.2     0.0000000
## 2 X.1004.BG98         BGJ398    2  181.9        28.0    -0.0891337
## 3 X.1004.BG98         BGJ398    5  172.7        28.4    -0.1352028
## 4 X.1004.BG98         BGJ398    9  129.6        27.2    -0.3510265
## 5 X.1004.BG98         BGJ398   12   91.3        26.7    -0.5428142
## 6 X.1004.BG98         BGJ398   16  117.1        26.2    -0.4136204
```

Simillerly for **batch** we can obtin all predefined batch names as:

```
batch.name <- batchInfo(brca)
batch.name[1:4]

## [1] "X-1004.BGJ398"      "X-1004.binimetinib" "X-1004.BKM120"
## [4] "X-1004.BYL719"
```

The information about a **batch** can be shown as :

```
batchInfo(brca, batch = "X-1004.binimetinib")

## $`X-1004.binimetinib`
## name = X-1004.binimetinib
## control = X.1004.uned
## treatment = X.1004.biib
```

Here for the batch named *X-1004.binimetinib* we can see that control sample is *X.1004.uned* and treatment sample is *X.1004.biib*.

5 Visualizing PDX Growth Curve

Xeva provides function to plot time vs. tumor volum data for individual models and also for a batch. Data can be plotted by using the name of the batch:

```
plotPDX(brca, batch = "X-4567.BKM120")
```

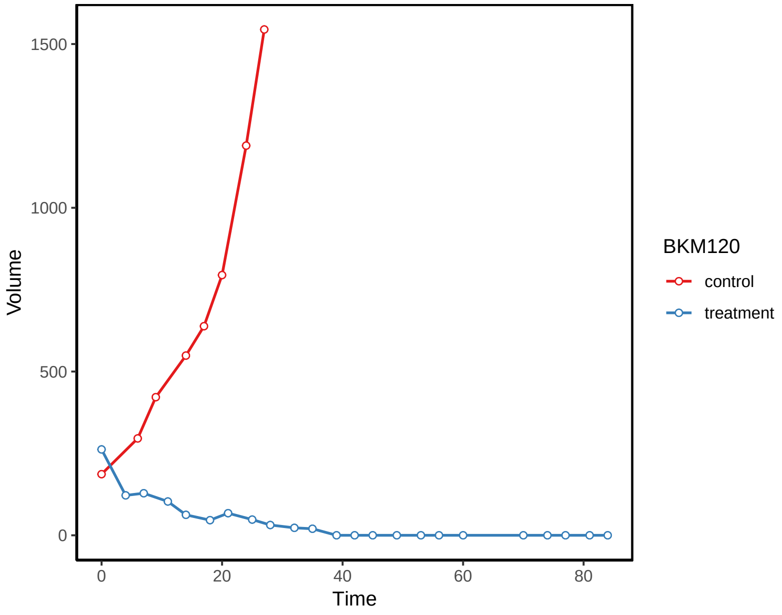

Figure 2: Tumor growth curve for control and treated PDX

Different aspects of this visualization can be chaged. For example we can plot normalized volume and change colors of lines:

```
plotPDX(brca, batch = "X-4567.BKM120", vol.normal = T, control.col = "#a6611a",
        treatment.col = "#018571", major.line.size = 1, max.time = 40)
```

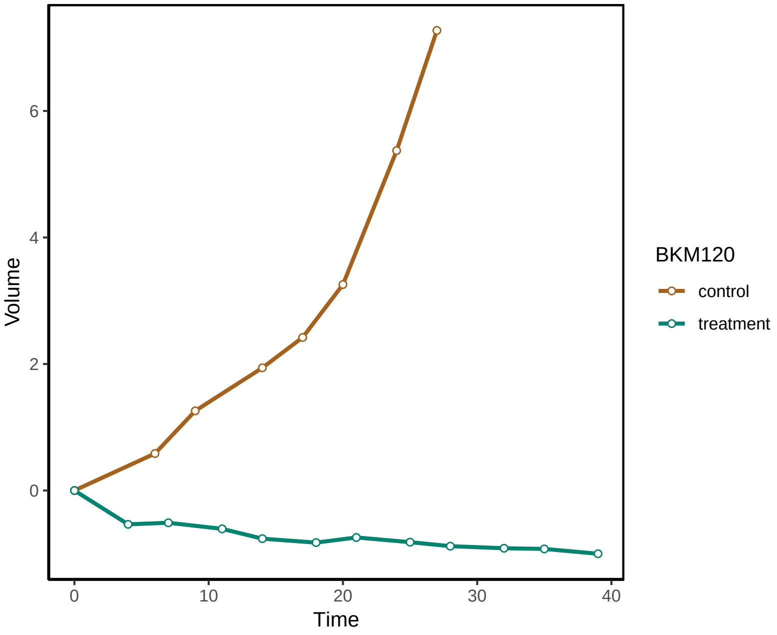

Figure 3: Tumor growth curve for control and treated PDX
Here volume is normalised and plots are trunketed at 40 days

Data can also be visualised at patient level by specifying patient id.

```
plotPDX(brca, patient.id="X-3078", drug="paclitaxel",control.name = "untreated")
```

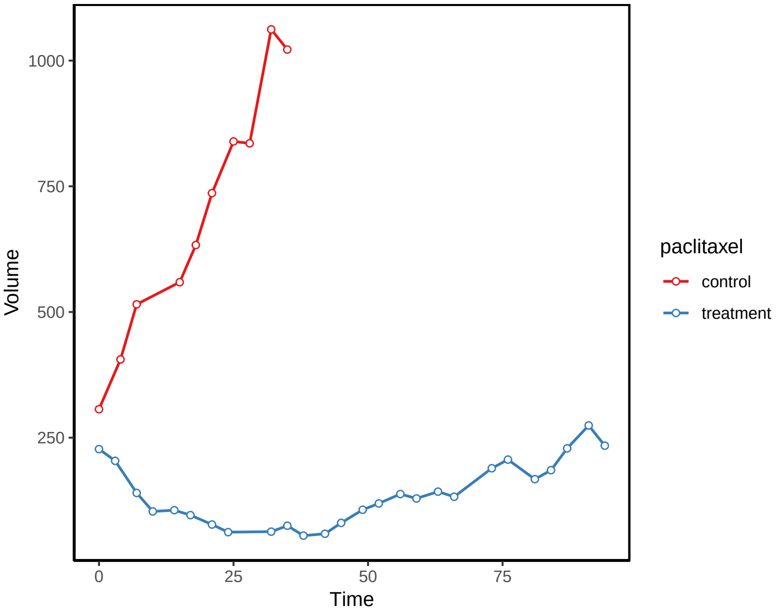

Figure 4: Tumor growth curve for control and treated PDX generated using patient id and drug

6 PDX Model Response

Xeva can effectively summarize the PDX response data. Here we summarize the **mRECIST** values for models in the dataset:

```
brca.mr <- summarizeResponse(brca, response.measure = "mRECIST")
brca.mr[1:5, 1:4]

##                   X-1004 X-1008 X-1286 X-1298
## BGJ398                PR     SD     PD     SD
## binimetinib           PD     SD     SD     PD
## BKM120                SD     SD     SD     PR
## BYL719                SD     PR     SD     PD
## BYL719 + LEE011       PD     SD     SD     PD
```

These **mRECIST** values can be visualized as:

```
plotmRECIST(brca.mr, control.name="untreated", row_fontsize=13, col_fontsize=12)
```

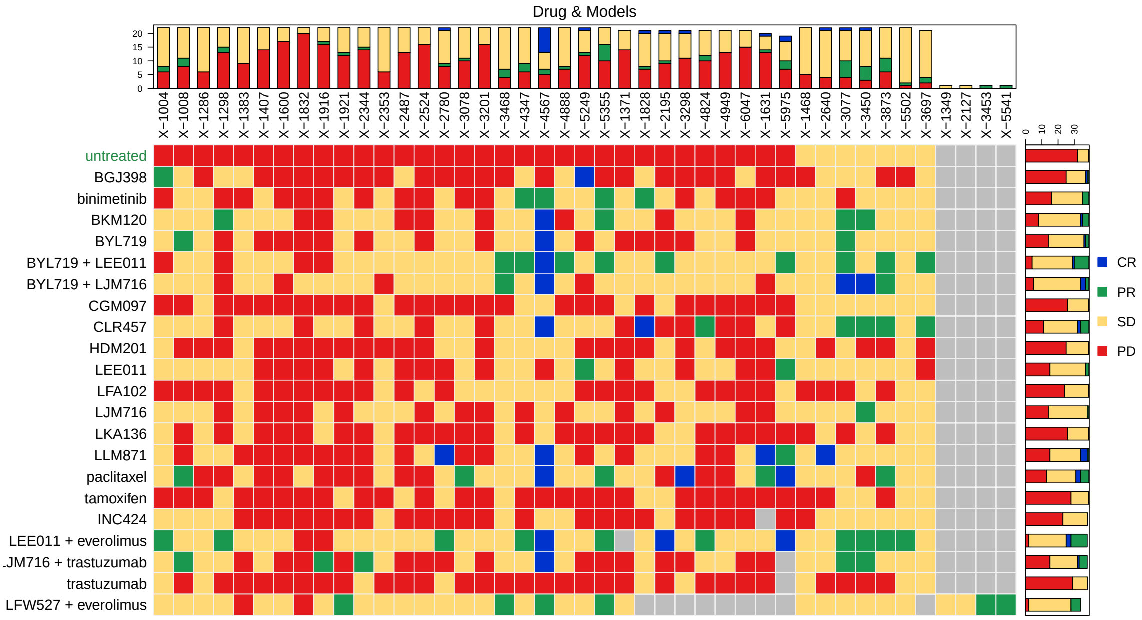

Figure 5: mRECIST plot for PDXE breast cancer data

Waterfall plots are also commonly used to visulize the PDX response data. Xeva provides function to visualize and color waterfall plots.

```
waterfall(brca, drug="binimetinib", res.measure="best.average.response")
```

It is usefull to color the bars of waterfall plot by genomic properties. Here we create waterfallplot for drug BYL719 and color it by mutation in CDK13 gene. First we extract the genomic data for models:

```
mut <- summarizeMolecularProfiles(brca,drug = "BYL719", mDataType="mutation")

## Loading required package:  Biobase
```

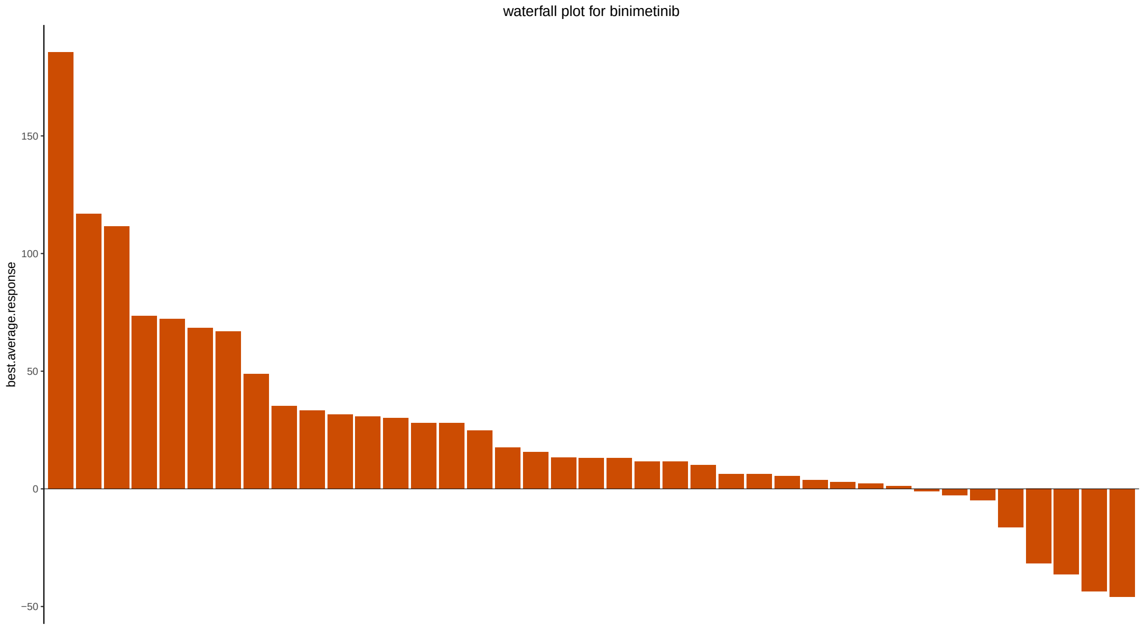

Figure 6: waterfall plot for binimetinib drug response in PDXs

```
## Loading required package:  BiocGenerics


## Loading required package:  parallel


##
## Attaching package:  'BiocGenerics'


## The following objects are masked from 'package:parallel':
##
##    clusterApply, clusterApplyLB, clusterCall, clusterEvalQ,
##    clusterExport, clusterMap, parApply, parCapply, parLapply,
##    parLapplyLB, parRapply, parSapply, parSapplyLB


## The following objects are masked from 'package:stats':
##
##    IQR, mad, sd, var, xtabs


## The following objects are masked from 'package:base':
##
##    anyDuplicated, append, as.data.frame, cbind, colMeans, colnames,
##    colSums, do.call, duplicated, eval, evalq, Filter, Find, get,
##    grep, grepl, intersect, is.unsorted, lapply, lengths, Map,
##    mapply, match, mget, order, paste, pmax, pmax.int, pmin,
##    pmin.int, Position, rank, rbind, Reduce, rowMeans, rownames,
##    rowSums, sapply, setdiff, sort, table, tapply, union, unique,
##    unsplit, which, which.max, which.min
## Welcome to Bioconductor
##
##    Vignettes contain introductory material; view with
##    'browseVignettes()'.  To cite Bioconductor, see
##    'citation("Biobase")', and for packages 'citation("pkgname")'.
```

```
model.type <- Biobase::exprs(mut)["CDK13", ]
model.type[grepl("Mut", model.type)] <- "mutation"
model.type[model.type!="mutation"] <- "wild type"
model.color <- list("mutation"="#fb8072", "wild type"="#80b1d3")
waterfall(brca, drug="BYL719", res.measure="best.average.response",
          model.id=names(model.type), model.type= model.type,
          type.color = model.color)
```

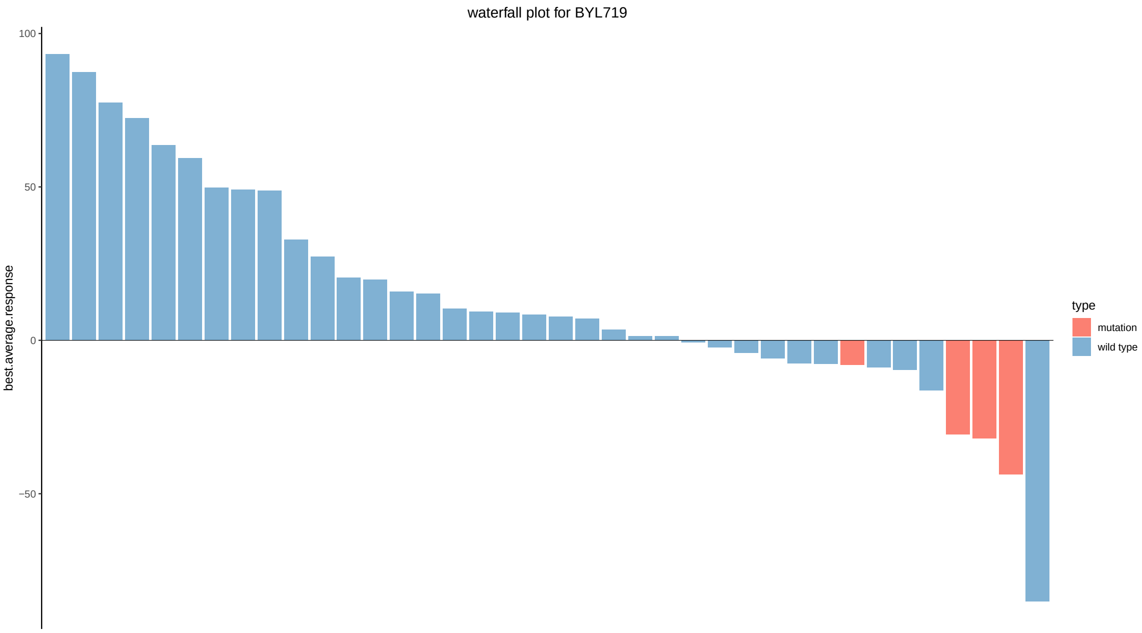

Figure 7: waterfall plot for binimetinib drug response in PDXs