The Tomato example: illustrating the smoothing and extraction of traits (SET) using growthPheno Version 2.x

Chris Brien

24 July, 2024

This vignette illustrates the use of the two growthPheno (Brien, 2024) wrapper functions traitSmooths and traitExtractFeatures that are key to carrying out the smoothing and extracting traits (SET) method described by Brien et al. (2020). The Tomato example, used here, is the example that Brien et al. (2020) used to illustrate the SET method. More details on the rationale for this method are available in Brien et al. (2020, Methods section).

Here, the process has been modified from that described in the paper to take advantage of the new wrapper functions and other new capabilities that have been built into in Version 2.x of growthPheno. In particular, both natural cubic smoothing splines (NCSS) and P-splines (PS) are investigated for smoothing not only the Projected Shoot Area (PSA), but also the Water Use (WU). A segmented smooth, as suggested in Brien et al. (2020), is used to allow for a discontinuity in the growth resulting from unintentional, restricted watering for three days following imaging on DAP 39.

Two different approaches are shown for smoothing the two traits:

**PSA:** For this trait, we first use traitSmooths to compare several smooths using logarithmic smoothing and then automatically choose a P-spline smooth whose lambda value is in the middle of the values for which smooths have been obtained. This is then followed by a comparison of two contending smooths. Finally, the chosen smooth is extracted and added to the data.

**WU:** A more time-efficient approach is taken with this trait. First several direct smooths are compared and stored. Then plots of two contending smooths amongst the stored smooths are compared. Finally the chosen smooth is extracted from the stored smooths.

**Initialize**

Set up characters for variable names and titles

```r
# The responses
responses <- c("PSA", paste("PSA", c("AGR", "RGR"), sep = "."))
responses.smooth <- paste0("s", responses)

# Specify time intervals of homogeneous growth dynamics
DAP.endpts <- c(18, 22, 27, 33, 39, 43, 51)
nDAP.endpts <- length(DAP.endpts)
DAP.starts <- DAP.endpts[-nDAP.endpts]
DAP.stops <- DAP.endpts[-1]
DAP.segs <- list(c(DAP.endpts[1]-1, 39),
                 c(40, DAP.endpts[nDAP.endpts]))
```
Step I: Import the longitudinal data

In this step, the aim is to produce the data.frame `longi.dat` that contains the imaging variables, covariates and factors for the experiment.

Load the pre-prepared data

```r
data(tomato.dat)
```

Copy the data to preserve the original data.frame

```r
longi.dat <- tomato.dat
```

Step II: Investigate the smoothing of the PSA and obtain growth rates

The growth rates are the Absolute Growth Rate (AGR) and the Relative Growth Rate (RGR) for the PSA, which must be calculated from the observed data by differencing consecutive observations for a plant. They will also be calculated from the smoothed traits by differencing, although `growthPheno` can also obtain growth rates using the first derivatives of the smooths.

Fit three-parameter logistic curves logistic curves to compare with spline curves

We fit a three-parameter logistic curve, using `nlme` (Pinheiro J., Bates D., and R Core Team, 2023), as an alternative to spline smoothing.

*Organize non-missing data into a grouped object*

```r
logist.dat <- na.omit(longi.dat)
logist.grp <- nlme::groupedData(PSA ~ cDAP | Snapshot.ID.Tag, 
    data = logist.dat)
```

*Fit logistics to individuals and obtain fitted values*
Compute smooths and growth rates of the PSA for a range of smoothing parameters

We begin by using the function `traitSmooth` to investigate a set of smooths for the PSA, employing all five `traitSmooth` steps of (i) Smooth, (ii) Profile plots, (iii) Median deviations plots, (iv) Choose a smooth, and (v) Chosen smooth plot. The only changes to the defaults for these five steps are to the df values that are investigated and to specify segmented smoothing. This includes allowing `traitSmooth` to choose automatically a single smooth as the chosen smooth. A segmented smooth involving two segments has also been specified, as suggested by Brien et al. (2020). The breakpoint for the segments is DAP 39, it coinciding with the start of an unintentional, three-day restriction in the watering; thus, the segments consist of DAP 18–39 and DAP 40–51. The growth rates are calculated, by default, from both the unsmoothed trait PSA and the smoothed trait sPSA by difference, rather than from the spline derivatives. Thus, the growth rate calculation for the smoothed data matches that which is obligatory for the observed data. Also, three-parameter logistic curves are fitted to the data using the R package `nlme` and growth rates calculated for it. The default layouts of the three sets of plots produced are modified using the three arguments `profile.plot.args`, `meddevn.plot.args` and `chosen.plot.args`.

It is noted that the plots that are produced show that the logistic would not be an adequate fit for this data, especially after DAP 42.

```r
suppressWarnings(
  longi.dat <- traitSmooth(data = tomato.dat,
    response = "PSA", response.smoothed = "sPSA",
    individuals = "Snapshot.ID.Tag", times = "DAP",
    keep.columns = c("AMF", "Zn"),
    smoothing.args = args4smoothing(df = c(4:6,12),
                        smoothing.segments = DAP.segs,
                        external.smooths = logist.dat),
    profile.plot.args =
      args4profile_plot(facet.y = "AMF",
                       colour.column = "Zn",
                       facet.labeller = labeller(AMF = labelAMF)),
    meddevn.plot.args =
      args4meddevn_plot(facet.y = "AMF",
                        facet.labeller = labeller(AMF = labelAMF)),
    chosen.plot.args =
      args4chosen_plot(facet.y = "AMF",
                        facet.labeller = labeller(AMF = labelAMF),
                        colour.column = "Zn",
                        ggplotFuncs = vline.DAP.endpts),
    mergedata = tomato.dat))
```
Compare log smoothing of PSA for NCSS with DF = 6 and PS with lambda = 1

Now compare what appear to be the best smooths for natural cubic smoothing splines (NCSS-df-6) and P-splines (PS-lambda-1) using `traitSmooth`. This is done by supplying `smoothing.args` with a list of parallel vectors, each vector being of length two. The argument `chosen.smooth.args` is set to `NULL` so that one of the smooths is not chosen for output. Again, arguments are included to control the smoothing and the layout of the profile and median-deviations plots.

Smoothing based on P-splines is chosen because it tends to smooth somewhat more than that based on NCSS splines, especially after DAP 45. Consequently, there is no need to change the values of the `chosen.splines` argument from the default values.

```r
smth.dat <- traitSmooth(data = longi.dat,
    response = "PSA", response.smoothed = "sPSA",
    individuals = "Snapshot.ID.Tag", times = "DAP",
    keep.columns = c("AMF","Zn"),
    smoothing.args =
        args4smoothing(smoothing.methods = c("log", "log"),
                       spline.types = c("N", "P"),
                       df = c(6, NA), lambdas = c(NA, 1),
                       combinations = "parallel",
                       smoothing.segments = DAP.segs),
    chosen.smooth.args = NULL,
    profile.plot.args =
        args4profile_plot(plots.by = NULL,
          ...))
```
facet.x = tune.fac, facet.y = "AMF",
facet.labeller = labeller(AMF = labelAMF),
colour.column = "AMF"),
meddevn.plot.args =
args4meddevn_plot(plots.by = NULL, plots.group = tune.fac,
facet.x = ".", facet.y = "AMF",
facet.labeller = labeller(AMF = labelAMF)))
## Warning: Removed 2 rows containing missing values or values outside the scale range ('geom_line()').

## Warning: Removed 4 rows containing missing values or values outside the scale range ('geom_point()').
## Warning: Removed 2 rows containing missing values or values outside the scale range
## (`geom_line()`).
## Removed 4 rows containing missing values or values outside the scale range
## (`geom_point()`).
Extract the chosen smooth, adding it to longi.dat

```r
longi.dat <- traitSmooth(data = smth.dat,
                           response = "PSA", response.smoothed = "sPSA",
                           individuals = "Snapshot.ID.Tag", times = "DAP",
                           keep.columns = c("AMF","Zn"),
                           smoothing.args = NULL, which.plots = "none",
                           chosen.smooth.args =
                               args4chosen_smooth(smoothing.methods = "log",
                                                  spline.types = "PS", lambdas = 1),
                           chosen.plot.args =
                               args4chosen_plot(facet.y = "AMF",
                                                facet.labeller = labeller(AMF = labelAMF),
                                                colour.column = "Zn",
                                                ggplotFuncs = vline.DAP.endpts),
                           mergedata = tomato.dat)
```

Step III: Investigate the smoothing of the WU

Explore the smooths of WU for a range of smoothing parameters

For WU, we take a slightly different approach to that taken with PSA. We first examine the fits for a range of smoothing parameters, setting the `traitSmooth` argument `chosen.smooth.args` to `NULL` so that a single
smooth is not chosen for output. We then examine the two smooths that are the main contenders and finally do plots for the smooth chosen from these two. Again, a segmented smooth involving two segments has also been specified with the breakpoint for the segments being DAP 39.

The function `traitSmooth` is used to produce the smooths. However, because no `chosen.smooth.args` is being specified, the function `probeSmooths` could be called directly instead. In this case, the `get.rates` and `trait.types` arguments from `probeSmooths` are set to `FALSE` and to "response" so that only the response is smoothed, without the calculation of growth rates from the smoothed response.

```r
suppressWarnings(
  smth.dat <- traitSmooth(data = longi.dat,
                           response = "WU", response.smoothed = "sWU",
                           individuals = "Snapshot.ID.Tag", times = "DAP",
                           keep.columns = c("AMF","Zn"),
                           trait.types = "response",
                           smoothing.args =
                             args4smoothing(smoothing.methods = "direct",
                                            smoothing.segments = DAP.segs),
                           chosen.smooth.args = NULL,
                           profile.plot.args =
                             args4profile_plot(plots.by = NULL,
                                               facet.y = "AMF",
                                               colour.column = "Zn",
                                               facet.labeller = labeller(AMF = labelAMF)),
                           meddevn.plot.args =
                             args4meddevn_plot(plots.by = NULL,
                                              facet.y = "AMF",
                                              facet.labeller = labeller(AMF = labelAMF))))
```

The image contains a series of graphs comparing different datasets labeled as Raw, dir-df-5, dir-df-6, dir-df-7, lambda-0.1, ir-lambda-, lambda-3.1, and lambda-1. Each graph shows the WU (Water Use) over time (DAP) with various smoothing parameters indicated. The graphs are arranged in columns and rows, with each column representing a different dataset and each row representing a different smoothing parameter.

The bottom section of the image includes tables labeled as NCSS and PS, showing median WU deviations over DAP with AMF parameter values. The tables are accompanied by line graphs illustrating the deviation trends for different smoothing parameters.
Produce plots comparing direct smoothing of WU for NCSS with DF = 6 and PS with lambda = 0.316

Now compare what appear to be the best smooths for natural cubic smoothing splines (NCSS-df-6) and for P-splines (PS-lambda-0.316). The function `traitSmooth` is used for the comparison, `probeSmooths` could be called directly instead. The PS splines with $\lambda = 0.316$ are chosen because they tend to smooth a little less than the NCSS splines, especially before DAP 26.

```r
suppressWarnings(
  traitsSmooth(data = smth.dat,
               response = "WU", response.smoothed = "sWU",
               individuals = "Snapshot.ID.Tag", times = "DAP",
               trait.types = "response",
               smoothing.args = args4smoothing(smoothing.methods = c("dir", "dir"),
                                                spline.types = c("N", "P"),
                                                df = c(6, NA), lambdas = c(NA, 0.316),
                                                smoothing.segments = DAP.segs,
                                                combinations = "parallel"),
               chosen.smooth.args = NULL,
               profile.plot.args =
               args4profile_plot(plots.by = NULL,
                                  facet.x = tune.fac, facet.y = "AMF",
                                  colour.column = "AMF",
                                  facet.labeller = labeller(AMF = labelAMF)),
               meddevn.plot.args =
               args4meddevn_plot(plots.by = NULL, plots.group = tune.fac,
                                 facet.x = ",", facet.y = "AMF",
                                 facet.labeller = labeller(AMF = labelAMF))))
```
Produce the plots for the chosen smooth and add it to longi.dat

Here `traitSmooth` is used to fit the two smooths specified in `spar.schemes` in the previous step and the `chosen.splines` argument is set for the fit using PS splines with $\lambda = 0.316$.

```r
longi.dat <- traitSmooth(data = smth.dat,
                          response = "WU", response.smoothed = "sWU",
                          individuals = "Snapshot.ID.Tag", times = "DAP",
                          trait.types = "response",
                          smoothing.args = NULL, which.plots = NULL,
                          chosen.smooth.args =
                          args4chosen_smooth(smoothing.method = "direct",
                                              spline.type = "PS",
                                              lambdas = 0.316), #tried 1 first
                          chosen.plot.args =
                          args4chosen_plot(facet.y = "AMF",
                                            facet.labeller = labelAMF(AMF = labelAMF),
                                            colour.column = "Zn",
                                            ggplotFuncs = vline.DAP.endpts),
                          mergedata = longi.dat)
```

Step IV: Identify potential outliers and remove if justified

A plant was identified as slow growing. Even though its pot had been inoculated with AMF, it had low AMF root colonization and a random mutated shoot phenotype, which could explain why its behaviour was consistent with a plant that was not inoculated with AMF. We omit the it from further analysis.
Omit responses for the outlier plant

The outlier plant is omitted by setting all of its responses to NA, i.e. the metadata for the plant is retained in longi.dat.

```r
# Identify the plant
omit <- with(longi.dat, Zn==90 & AMF=="+" & Block ==4)

# Identify the responses columns
NA.cols <- match("Weight.After", names(longi.dat))
responses.all <- names(longi.dat)[NA.cols]

# Set the responses for the plant to NA
longi.dat[responses.all] <- lapply(longi.dat[responses.all], function(kcol, omit)
  {kcol[omit] <- NA
   return(kcol)
  }, omit = omit)
```

Step V: Extract single-valued traits for each individual

In this step, traits that have a single-value for each plant (cart) are created from the smoothed PSA (sPSA) and the smoothed WU (sWU), along with the derived traits sPSA AGR, sPSA RGR, sWUR (smoothed Water Use Rate) and sPSA.sWUI (smoothed Water Use Index with sPSA as the numerator). The single-valued traits are based on a set of endpoints for DAP intervals. The DAP endpoints that were chosen, as described by Brien et al. (2020), are 18, 22, 27, 33, 39, 43 and 51. Corresponding to these endpoints are the time intervals DAP 18–22, DAP 22–27, DAP 27–33, DAP 33–39, DAP 39–43 and DAP 43–51. Based on these endpoints and intervals, the following single-valued traits are to be computed:

1. **single-times traits:** sPSA for each DAP
2. **growth rates for a time interval:** sPSA AGR and sPSA RGR for the six intervals.
3. **water use traits for a time interval:** sWU, sWUR and sPSA.sWUI for the six intervals.
4. **total for the overall imaging period:** sWU for DAP 18–51.
5. **maximum for the overall imaging period:** maximum of the sPSA AGR during DAP 18–51 and the DAP on which it occurred.

```r
indv.cols <- c("Snapshot.ID.Tag", "Lane", "Position", "Block", "Cart", "AMF", "Zn")
indv.dat <- subset(longi.dat, subset = DAP == DAP.endpts[1],
  select = indv.cols)
indv.dat <- traitExtractFeatures(data = longi.dat,
  starts.intvl = DAP.starts, stops.intvl = DAP.stops,
  responses4singletimes = "sPSA",
  responses4intvl.rates = "sPSA",
  water.use4intvl.traits = "sWU",
  responses4overall.total = "sWU",
  responses4overall.max = "sPSA.AGR",
  mergedata = indv.dat)
```

Finalise
```r
indv.dat <- with(indv.dat, indv.dat[order(Snapshot.ID.Tag), ])
summary(indv.dat)
```

<table>
<thead>
<tr>
<th>Snapshot.ID.Tag</th>
<th>Lane</th>
<th>Position</th>
<th>Block</th>
<th>Cart</th>
<th>AMF</th>
<th>Zn</th>
</tr>
</thead>
<tbody>
<tr>
<td>Length:32</td>
<td>6:16</td>
<td>5 : 2</td>
<td>1:8</td>
<td>1 :4</td>
<td>- :16</td>
<td>0 :8</td>
</tr>
<tr>
<td>Class :character</td>
<td>7:16</td>
<td>6 : 2</td>
<td>2:8</td>
<td>2 :4</td>
<td>15 : 10</td>
<td>8 :8</td>
</tr>
<tr>
<td>Mode :character</td>
<td>7</td>
<td>2 : 3</td>
<td>8:3</td>
<td>3 :4</td>
<td>NA's:1</td>
<td>40 :8</td>
</tr>
<tr>
<td></td>
<td>8</td>
<td>2 : 4:8</td>
<td>4 :4</td>
<td>NA's:1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>9</td>
<td>2 : 5</td>
<td>4 :4</td>
<td>NA's:1</td>
<td></td>
<td></td>
</tr>
<tr>
<td></td>
<td>10</td>
<td>2 : 6</td>
<td>4 :4</td>
<td>NA's:1</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sPSA.18</th>
<th>sPSA.22</th>
<th>sPSA.27</th>
<th>sPSA.33</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. : 2.128</td>
<td>Min. : 4.032</td>
<td>Min. : 8.37</td>
<td>Min. : 17.01</td>
</tr>
<tr>
<td>1st Qu. : 4.789</td>
<td>1st Qu. :10.501</td>
<td>1st Qu. :28.65</td>
<td>1st Qu. :63.87</td>
</tr>
<tr>
<td>Mean : 6.710</td>
<td>Mean :13.978</td>
<td>Mean :37.76</td>
<td>Mean :79.95</td>
</tr>
<tr>
<td>3rd Qu. : 8.398</td>
<td>3rd Qu. :16.807</td>
<td>3rd Qu. :47.84</td>
<td>3rd Qu. :97.53</td>
</tr>
<tr>
<td>Max. :14.100</td>
<td>Max. :27.612</td>
<td>Max. :61.20</td>
<td>Max. :129.59</td>
</tr>
<tr>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sPSA.39</th>
<th>sPSA.43</th>
<th>sPSA.51</th>
<th>sPSA.AGR.18to22</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. :34.33</td>
<td>Min. :41.16</td>
<td>Min. :71.27</td>
<td>Min. :0.3905</td>
</tr>
<tr>
<td>1st Qu. :96.46</td>
<td>1st Qu. :105.27</td>
<td>1st Qu. :122.76</td>
<td>1st Qu. :1.4727</td>
</tr>
<tr>
<td>Median :115.53</td>
<td>Median :123.55</td>
<td>Median :133.45</td>
<td>Median :1.6730</td>
</tr>
<tr>
<td>Mean :110.98</td>
<td>Mean :118.08</td>
<td>Mean :134.50</td>
<td>Mean :1.8170</td>
</tr>
<tr>
<td>3rd Qu. :133.76</td>
<td>3rd Qu. :140.45</td>
<td>3rd Qu. :154.31</td>
<td>3rd Qu. :2.3631</td>
</tr>
<tr>
<td>Max. :164.69</td>
<td>Max. :166.76</td>
<td>Max. :185.36</td>
<td>Max. :3.3781</td>
</tr>
<tr>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sPSA.RGR.18to22</th>
<th>sPSA.AGR.22to27</th>
<th>sPSA.RGR.22to27</th>
<th>sPSA.AGR.27to33</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. :0.1131</td>
<td>Min. :0.7833</td>
<td>Min. :0.1262</td>
<td>Min. :1.441</td>
</tr>
<tr>
<td>1st Qu. :0.1613</td>
<td>1st Qu. :3.6237</td>
<td>1st Qu. :0.1824</td>
<td>1st Qu. :5.793</td>
</tr>
<tr>
<td>Median :0.1827</td>
<td>Median :4.8037</td>
<td>Median :0.2005</td>
<td>Median :7.266</td>
</tr>
<tr>
<td>Mean :0.1854</td>
<td>Mean :4.7572</td>
<td>Mean :0.1961</td>
<td>Mean :7.032</td>
</tr>
<tr>
<td>3rd Qu. :0.2026</td>
<td>3rd Qu. :6.2821</td>
<td>3rd Qu. :0.2165</td>
<td>3rd Qu. :8.582</td>
</tr>
<tr>
<td>Max. :0.3192</td>
<td>Max. :8.0144</td>
<td>Max. :0.2461</td>
<td>Max. :11.397</td>
</tr>
<tr>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sPSA.RGR.27to33</th>
<th>sPSA.AGR.33to39</th>
<th>sPSA.RGR.33to39</th>
<th>sPSA.AGR.39to43</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. :0.08414</td>
<td>Min. :1.434</td>
<td>Min. :0.03775</td>
<td>Min. :-0.7949</td>
</tr>
<tr>
<td>1st Qu. :0.11848</td>
<td>1st Qu. :4.700</td>
<td>1st Qu. :0.0582</td>
<td>1st Qu. :1.4347</td>
</tr>
<tr>
<td>Median :0.12585</td>
<td>Median :5.391</td>
<td>Median :0.0582</td>
<td>Median :1.9842</td>
</tr>
<tr>
<td>Mean :0.12554</td>
<td>Mean :5.171</td>
<td>Mean :0.05843</td>
<td>Mean :1.7757</td>
</tr>
<tr>
<td>3rd Qu. :0.13267</td>
<td>3rd Qu. :5.862</td>
<td>3rd Qu. :0.06661</td>
<td>3rd Qu. :2.4714</td>
</tr>
<tr>
<td>Max. :0.16237</td>
<td>Max. :7.349</td>
<td>Max. :0.11699</td>
<td>Max. :3.1744</td>
</tr>
<tr>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sPSA.RGR.39to43</th>
<th>sPSA.AGR.43to51</th>
<th>sPSA.RGR.43to51</th>
<th>sWU.18to22</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. :0.00663</td>
<td>Min. :3.694</td>
<td>Min. :0.02885</td>
<td>Min. :79.80</td>
</tr>
<tr>
<td>1st Qu. :0.01199</td>
<td>1st Qu. :1.539</td>
<td>1st Qu. :0.01038</td>
<td>1st Qu. :85.77</td>
</tr>
<tr>
<td>Median :0.01797</td>
<td>Median :2.510</td>
<td>Median :0.02115</td>
<td>Median :96.43</td>
</tr>
<tr>
<td>Mean :0.01900</td>
<td>Mean :2.052</td>
<td>Mean :0.01831</td>
<td>Mean :93.61</td>
</tr>
<tr>
<td>3rd Qu. :0.02424</td>
<td>3rd Qu. :3.384</td>
<td>3rd Qu. :0.02619</td>
<td>3rd Qu. :100.05</td>
</tr>
<tr>
<td>Max. :0.06542</td>
<td>Max. :5.224</td>
<td>Max. :0.06864</td>
<td>Max. :104.25</td>
</tr>
<tr>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
<td>NA's :1</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>sWUR.18to22</th>
<th>sPSA.sWUI.18to22</th>
<th>sWU.22to27</th>
<th>sWUR.22to27</th>
</tr>
</thead>
<tbody>
<tr>
<td>Min. :19.95</td>
<td>Min. :0.01654</td>
<td>Min. :90.13</td>
<td>Min. :18.03</td>
</tr>
</tbody>
</table>

23
## 1st Qu.:21.44 1st Qu.:0.06260 1st Qu.:102.34 1st Qu.:20.47
## Median :24.11 Median :0.07068 Median :109.55 Median :21.91
## Mean :23.40 Mean :0.07817 Mean :107.81 Mean :21.56
## 3rd Qu.:25.01 3rd Qu.:0.10147 3rd Qu.:112.68 3rd Qu.:22.54
## Max. :26.06 Max. :0.13012 Max. :125.61 Max. :25.12
## NA's :1 NA's :1 NA's :1 NA's :1
## sPSA.sWUI.22to27 sWU.27to33 sWUR.27to33 sPSA.sWUI.27to33
## Min. :0.03858 Min. :106.0 Min. :17.67 Min. :0.07756
## 1st Qu.:0.16720 1st Qu.:140.8 1st Qu.:23.46 1st Qu.:0.24544
## Median :0.22553 Median :152.7 Median :25.45 Median :0.27223
## Mean :0.21811 Mean :150.9 Mean :25.15 Mean :0.27200
## 3rd Qu.:0.27152 3rd Qu.:165.4 3rd Qu.:27.56 3rd Qu.:0.31508
## Max. :0.35963 Max. :182.4 Max. :30.41 Max. :0.40126
## NA's :1 NA's :1 NA's :1 NA's :1
## sWU.33to39 sWUR.33to39 sPSA.sWUI.33to39 sWU.39to43
## Min. :126.7 Min. :21.12 Min. :0.05969 Min. :65.15
## 1st Qu.:190.5 1st Qu.:31.75 1st Qu.:0.13273 1st Qu.:74.32
## Median :211.3 Median :35.21 Median :0.15037 Median :77.46
## Mean :204.2 Mean :34.04 Mean :0.15159 Mean :77.00
## 3rd Qu.:223.1 3rd Qu.:37.19 3rd Qu.:0.17207 3rd Qu.:80.52
## Max. :259.4 Max. :43.24 Max. :0.20415 Max. :83.88
## NA's :1 NA's :1 NA's :1 NA's :1
## sWUR.39to43 sPSA.sWUI.39to43 sWU.43to51 sWUR.43to51
## Min. :16.29 Min. :-0.04207 Min. :190.6 Min. :23.83
## 1st Qu.:18.58 1st Qu.: 0.07150 1st Qu.:230.5 1st Qu.:28.81
## Median :19.37 Median : 0.10263 Median :242.5 Median :30.32
## Mean :19.25 Mean : 0.09285 Mean :238.7 Mean :29.84
## 3rd Qu.:20.13 3rd Qu.: 0.13108 3rd Qu.:249.8 3rd Qu.:31.23
## Max. :20.97 Max. : 0.19489 Max. :268.5 Max. :33.56
## NA's :1 NA's :1 NA's :1 NA's :1
## sPSA.sWUI.43to51 sWU sPSA.AGR.max sPSA.AGR.max.DAP
## Min. :-0.13026 Min. :701.0 Min. : 3.963 Min. :12.00
## 1st Qu.: 0.04992 1st Qu.:858.5 1st Qu.: 6.150 1st Qu.:13.00
## Median : 0.08270 Median :884.0 Median : 7.744 Median :14.00
## Mean : 0.06762 Mean :874.0 Mean : 7.791 Mean :15.77
## 3rd Qu.: 0.10781 3rd Qu.:922.0 3rd Qu.: 9.148 3rd Qu.:16.00
## Max. : 0.15907 Max. :988.0 Max. :12.423 Max. :35.00
## NA's :1 NA's :1 NA's :1 NA's :1

**head(indv.dat)**

```r
## Snapshot.ID.Tag Lane Position Block Cart AMF Zn sPSA.18 sPSA.22 sPSA.27
## 1 061472 6 5 1 1 - 0 9.856841 21.132127 61.20433
## 2 061473 6 6 1 2 + 10 8.219937 15.732854 39.75138
## 3 061474 6 7 1 3 - 90 2.469923 4.032111 10.07049
## 4 061475 6 8 1 4 + 40 8.971075 14.864706 31.21562
## 5 061476 6 9 1 5 + 90 4.823554 9.198190 27.09603
## 6 061477 6 10 1 6 - 40 4.998369 11.434154 33.88250
## sPSA.33 sPSA.39 sPSA.43 sPSA.51 sPSA.AGR.18to22 sPSA.RGR.18to22
## 1 129.58879 164.69352 166.75700 171.47291 2.8188215 0.1906572
## 2 87.87222 123.11477 131.05159 159.65092 1.8782293 0.1622972
## 3 65.05030 99.72473 107.64422 131.06986 1.4734077 0.1262460
## 4 62.69652 94.52888 105.67301 127.43397 1.0936589 0.1613739
```
<table>
<thead>
<tr>
<th></th>
<th>sPSA.AGR.22to27</th>
<th>sPSA.RGR.22to27</th>
<th>sPSA.AGR.27to33</th>
<th>sPSA.RGR.27to33</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>8.014441</td>
<td>0.2126847</td>
<td>11.397410</td>
<td>0.1250247</td>
</tr>
<tr>
<td>2</td>
<td>4.803705</td>
<td>0.1853787</td>
<td>8.020140</td>
<td>0.1322065</td>
</tr>
<tr>
<td>3</td>
<td>1.207676</td>
<td>0.1830638</td>
<td>2.473389</td>
<td>0.1509488</td>
</tr>
<tr>
<td>4</td>
<td>3.270184</td>
<td>0.1483858</td>
<td>5.639112</td>
<td>0.123737</td>
</tr>
<tr>
<td>5</td>
<td>3.579568</td>
<td>0.2160761</td>
<td>5.933415</td>
<td>0.1398198</td>
</tr>
<tr>
<td>6</td>
<td>4.486707</td>
<td>0.2172588</td>
<td>9.313008</td>
<td>0.1627345</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>sPSA.AGR.33to39</th>
<th>sPSA.RGR.33to39</th>
<th>sPSA.AGR.39to43</th>
<th>sPSA.RGR.39to43</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>5.850789</td>
<td>0.03995334</td>
<td>0.5158698</td>
<td>0.003112841</td>
</tr>
<tr>
<td>2</td>
<td>5.873758</td>
<td>0.05620555</td>
<td>1.9840258</td>
<td>0.015618520</td>
</tr>
<tr>
<td>3</td>
<td>3.561867</td>
<td>0.10324189</td>
<td>3.0271466</td>
<td>0.058100365</td>
</tr>
<tr>
<td>4</td>
<td>5.779072</td>
<td>0.07120882</td>
<td>1.9874220</td>
<td>0.019174584</td>
</tr>
<tr>
<td>5</td>
<td>5.305394</td>
<td>0.06843325</td>
<td>2.7860332</td>
<td>0.027861036</td>
</tr>
<tr>
<td>6</td>
<td>7.340184</td>
<td>0.06653549</td>
<td>2.4429507</td>
<td>0.017622072</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>sPSA.AGR.43to51</th>
<th>sPSA.RGR.43to51</th>
<th>sWU.18to22</th>
<th>sWUR.18to22</th>
<th>sPSA.sWUI.18to22</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0.5894883</td>
<td>0.003485951</td>
<td>97.91084</td>
<td>24.47771</td>
<td>0.11515871</td>
</tr>
<tr>
<td>2</td>
<td>3.5749165</td>
<td>0.024674829</td>
<td>97.85921</td>
<td>24.46480</td>
<td>0.07677277</td>
</tr>
<tr>
<td>3</td>
<td>2.4468849</td>
<td>0.036139220</td>
<td>94.46701</td>
<td>23.61675</td>
<td>0.01653687</td>
</tr>
<tr>
<td>4</td>
<td>2.9242928</td>
<td>0.024577301</td>
<td>101.82429</td>
<td>25.46607</td>
<td>0.05788041</td>
</tr>
<tr>
<td>5</td>
<td>2.7201203</td>
<td>0.023406106</td>
<td>96.41753</td>
<td>24.10438</td>
<td>0.04537179</td>
</tr>
<tr>
<td>6</td>
<td>5.2239236</td>
<td>0.031934903</td>
<td>98.41988</td>
<td>24.60497</td>
<td>0.06539112</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>sWU.22to27</th>
<th>sPSA.RGR.22to27</th>
<th>sWU.27to33</th>
<th>sWUR.27to33</th>
<th>sPSA.sWUI.27to33</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>111.4264</td>
<td>22.28527</td>
<td>0.35962943</td>
<td>174.3139</td>
<td>29.08232</td>
</tr>
<tr>
<td>2</td>
<td>105.6890</td>
<td>21.13780</td>
<td>0.22725657</td>
<td>151.6969</td>
<td>25.28228</td>
</tr>
<tr>
<td>3</td>
<td>90.1329</td>
<td>18.02658</td>
<td>0.06699416</td>
<td>106.0449</td>
<td>17.67415</td>
</tr>
<tr>
<td>4</td>
<td>107.0495</td>
<td>21.40991</td>
<td>0.15274160</td>
<td>142.7822</td>
<td>23.79703</td>
</tr>
<tr>
<td>5</td>
<td>103.1972</td>
<td>20.63943</td>
<td>0.17343342</td>
<td>134.7183</td>
<td>22.45304</td>
</tr>
<tr>
<td>6</td>
<td>109.6825</td>
<td>21.93651</td>
<td>0.20466657</td>
<td>154.0212</td>
<td>25.67021</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>sPSA.AGR.max</th>
<th>sPSA.AGR.max.DAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12.42279</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8.415909</td>
<td></td>
</tr>
<tr>
<td>3</td>
<td>4.444479</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>6.198353</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>6.100730</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>10.09072</td>
<td>16</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>sPSA.AGR.39to43</th>
<th>sPSA.sWUI.39to43</th>
<th>sWU.39to43</th>
<th>sWUR.39to43</th>
<th>sPSA.sWUI.39to43</th>
<th>sWU.43to51</th>
<th>sPSA.sWUI.43to51</th>
<th>sWU.43to51</th>
<th>sPSA.sWUI.43to51</th>
<th>sWU</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20.22151</td>
<td>0.02551094</td>
<td>234.1140</td>
<td>29.26424</td>
<td>0.02014364</td>
<td>936</td>
<td>19.92687</td>
<td>0.09957441</td>
<td>240.2925</td>
<td>30.03657</td>
</tr>
<tr>
<td>2</td>
<td>17.44816</td>
<td>0.17349372</td>
<td>203.2074</td>
<td>25.40092</td>
<td>0.09633057</td>
<td>706</td>
<td>19.36545</td>
<td>0.10262720</td>
<td>242.5382</td>
<td>30.31727</td>
</tr>
<tr>
<td>3</td>
<td>20.67819</td>
<td>0.13473290</td>
<td>249.2872</td>
<td>31.16090</td>
<td>0.08729273</td>
<td>855</td>
<td>20.06866</td>
<td>0.12172963</td>
<td>262.7254</td>
<td>32.84067</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th>sPSA.AGR.max</th>
<th>sPSA.AGR.max.DAP</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>12.42279</td>
<td></td>
</tr>
<tr>
<td>2</td>
<td>8.415909</td>
<td>15</td>
</tr>
<tr>
<td>3</td>
<td>4.444479</td>
<td>23</td>
</tr>
<tr>
<td>4</td>
<td>6.198353</td>
<td>17</td>
</tr>
<tr>
<td>5</td>
<td>6.100730</td>
<td>14</td>
</tr>
<tr>
<td>6</td>
<td>10.09072</td>
<td>16</td>
</tr>
</tbody>
</table>
Step VI: Save to files

Save data files as csv, Excel and rda files

```r
save(longi.dat, file="longi.dat.rda")
write.csv(longi.dat, "longi.dat.csv", row.names = F)
WriteXLS("longi.dat", ExcelFileNames = "longi.dat.xlsx", SheetNames = "longi.dat",
        row.names = FALSE, BoldHeaderRow = TRUE, AdjWidth = TRUE, FreezeRow = 1)
save(indv.dat, file="indv.dat.rda")
write.csv(indv.dat, "indv.dat.csv", row.names = F)
WriteXLS("indv.dat", ExcelFileNames = "indv.dat.xlsx", SheetNames = "indv.dat",
        row.names = FALSE, BoldHeaderRow = TRUE, AdjWidth = TRUE, FreezeRow = 1)
```

Save the workspace image

```r
save.image("Tomato.RData")
```

Reference

