refreg: an R package for estimating conditional reference regions (id 2021-118)

Dear referees, we are very grateful for your comments and suggestions. In the lines below we answer them point-by-point:

 # REVIEWER 1

Journey Review -- refreg: an R package for estimating conditional reference regions

The article introduces an R package ('refreg') which calculates conditional multivariate reference regions (MVRs). MVRs then applied to the field of clinical practice involved in the diagnosis of diabetes or hypothyroidism. These diseases require more than one continuous test for diagnosis. The article explains the alternative methods. However, R packages (gamlss; referenceIntervals, etc.) do not meet the requirement to support multiple continuing tests simultaneously, which can lead to biased diagnoses. Also, clinical medicine has not yet been widely used due to the limited reference material available for the MVRs. So, I believe the research area of this article is a step forward for the R community in the field of clinical practice.

For me, this approach makes sense. the MVRs are an extension of the reference interval. To get the distribution, we care about the region and the shape. Regions are non-parametric. We use a multivariate kernel density estimator to estimate the region. The shape is changed by the influence of covariates. The effect of covariates is on the mean and variance-covariance matrix of the multivariate. Flexible additive predictors can estimate the effect of covariates. And the penalised spline smoother estimates the non-linear effects of continuous covariates.

In the 'Introduction' section, we could assume that the reader has a basic knowledge of mathematics and statistics, but for deeper knowledge like clinical practice, especially some proper nouns. It would be useful to explain. What this proper noun means. Why we are using this proper noun. What is the clinical logic behind it. See Appendix 1 for detailed suggestions. In summary 1) We have only said that "continuous diagnostic tests" require the use of a "reference interval". But how? What is the next step after the 'reference interval' has been calculated? 2) We use 'healthy subjects' to calculate the 'reference interval'. Why use healthy subjects? What are the implications if we do not distinguish between the types of subjects? 3) We use 'sick people' when the disease is not relevant to the patient characteristics. Why do we use 'sick patients' at this point? Why is it not feasible to use healthy patients at this point?

The 'Overview of the package refreg' section doesn't do as good a job as our abstract' section. Please do not wait until the 'refreg in practice' section to make everything specific, precise and clear. See Appendix 1 for detailed suggestions. In summary 1) We define covariates as independent variables and calculate new covariates in the derivation process. So, make a distinction each time a 'covariate' appears. 2) Avoid using words with vague outlines, such as 'response'. It can stand for any equation or a specific result. 3) Standardise the representative words. 4) 'summary()', 'forecast()', etc. are well-known functions in other packages, please cite the source. Or if we created ourselves, mention it. 5) Apart from these two main functions, no basic explanations have been written for the other functions involved in the package. This leads to these formulas appearing somewhat abruptly and confusingly in the 'refreg in practice' section.

'refreg in practice' section is good. refreg package has data, so it's convenient for readers to simulate the examples. When we have a concrete example, the explanation becomes more straightforward and clearer. Three examples, one for diabetes diagnosis (FPG, HbA1c), one for pollutant monitoring (SO2, NOx) and one for conceptual extension of multidimensional space. Each line of code is followed by a short analysis, explanation and summary. The refreg package does not have a vignette file, so perhaps a little streamlining of this section would make a good vignette.

As an R package that has been successfully released on CRAN, it bodes well that the authors have worked hard enough to check many of the formats and requirements. So overall there are no major problems. The additional checks I have made are detailed in Appendix 2. Things we don't want to include such as setwd() are not present. The only problem is with the alignment conventions between functions as suggested by 'styler' package.

In clinical practice, many medical decisions are based on continuous diagnostic tests (Hallworth, 2011), the interpretation of which requires a reference interval, i.e., one that characterizes healthy subjects' results. Reference intervals for a single test are estimated from the 2.5 and 97.5 empirical percentiles of the distribution for the healthy population; thus, 95% of healthy patients are located within the interval limits (Wright and Royston, 1999). If the test results are influenced by some patient characteristics independent of the disease (e.g., age and gender), the reference intervals for specific patient groups must be obtained. These covariate-dependent reference intervals, usually termed reference curves, are estimated using quantile regression (Koenker and Bassett Jr, 1978) or location-scale models (Cole

Suggestions:
It depends on what kind of potential readers we want. But literally, we can probably assume that our R journal readers have some more adamic background like basic mathematics and statistics. But I believe that not all R journal readers have a background in clinical practice, which suggests that some of the proper terms associated with clinical practice would be better explained in general terms.

Thanks for highlighting this topic. Because we are used to some terms, we did not realize some of your views. We tried to clarify it better.

For example

1) What is a '*continuous diagnostic test'*? Just as we give an explanation of the "*reference interval*" in the *abstract* section. We could also give a quick one-sentence summary of what a "*continuous diagnostic test*" is
2) How do '*continuous diagnostic tests*' obtain their results? We have just said that a *'continuous diagnostic test'* requires a *'reference interval'*. Why does it require a *'reference interval'*? How does it require a *'reference interval'*? Is there any clinical logic behind this?

A continuous test is just a test which takes values in a continuous scale, so in order to interpret them physicians needs some limits to define diagnostic cut points. Reference interval is the most common way of defining these diagnostic cut points. Such intervals are defined as two points between which most healthy patients are contained. The logic behind is that if a value is not located inside the interval, it is quite probable that this patient does not belong to the healthy population.

3) When we get a *'reference interval'*, how does this relate to a *'continuous diagnostic test'*? How do we use the *'reference interval'* to get the results of a *'continuous diagnostic test'*? Do we look at the peaks? Should we look at the most dense areas? Or what?

I think that this sentence in the paper already clarifies this topic Reference intervals for a single test are estimated from the 2.5 and 97.5 empirical percentiles of the distribution for the healthy population; thus, 95% of healthy patients are located within the interval limits (Wright and Royston, 1999). We look at the central part of the distribution that it is likely to be the densest.

4) Why do we use *'healthy people'* to calculate the *'reference interval'*? Why is it important for us to use "*healthy people*"? What is the clinical logic behind this?

We use healthy subjects to characterize the test variability in case of no pathology. By doing so we can identify which test values are not likely to be seen for a healthy subject.

5) Why do we need "*patient groups*" to be considered when "*patient characteristics*" such as "age" and "gender" are not relevant to the disease? Is there any clinical logic behind this?

Continuous tests usually show different mean, and variance, values for different healthy patient groups. For instance, older patients show higher values of glucose even if they do not suffer from diabetes. So, it is common in clinical practice to estimate reference intervals stratified by patients' groups, for instance females/males, young/middle age/elder etc.

Based on these questions we rewrite first introduction paragraph in order to be clearer for statisticians.

In clinical practice, many medical decisions are based on continuous diagnostic tests (Hallworth, 2011) – i.e., tests that provide results along a continuous, quantitative scale. The interpretation of this continuous values by physicians requires the comparison of the obtained value with a pre-defined reference interval, so that a result could be classified as positive or negative (ie, disease present or absent) based on these comparator value. A reference interval is an interval containing most healthy subjects' results. For a single test they are usually estimated from the 2.5 and 97.5 empirical percentiles of the distribution for the healthy population; thus, 95% of healthy patients are located within the interval limits (Wright and Royston, 1999). Those patients falling outside the reference interval, are likely to have an undiagnosed disease. If the test results are influenced by some patient characteristics independent of the disease (e.g., age and gender), reference intervals for specific patient groups must be obtained. These covariate-dependent reference intervals, usually termed reference curves, are estimated using quantile regression (Koenker and Bassett Jr, 1978) or location-scale models (Cole and Green, 1992; Stasinopoulos et al., 2017). Several R packages for estimating reference intervals and reference curves already exist, including the R package ***referenceIntervals****, which comprises a collection of tools, the R package* ***gamlss*** *(Stasinopoulos et al., 2007), which provides a general tool for deriving reference curves in clinical practice (WHO, 2006), and software* ***RefCurv*** *(Winkler et al., 2019), recently proposed to facilitate clinicians' use of* ***gamlss****. However, all these packages were produced to provide reference intervals for single tests; they cannot address diseases for which diagnosis and control are based on multiple tests.*

The **refreg** package contains a set of functions for estimating a conditional reference, or uncertainty, region. Its working framework was designed so that people without a strong statistical background can use it. Indeed, only two functions need to be taken into account by the user: 1) the effects of the covariates on responses need to be estimated using the bivRegr function, a step that requires the user choose which variables may influence the region; 2) bivRegion needs to be applied to a bivRegr object so that the reference region can be estimated. Numerical and graphical summaries of `bivRegr`- and `bivRegion`-fitted objects for both objects' classes can be obtained using the `summary_boot`, `summary`, `predict`, and `plot` routines.

Suggestions:

When we define independent variables as parameters that are often used in the calculation/deduction process (e.g. mean, variance, covariance, summation, etc.). Perhaps we would do well to make the distinction in the following paragraphs in order to avoid any misunderstanding. We are always welcome to recall the previous paragraph. And we have already done this, for example, when we write "see equation xx", I literally turn up the pages to find that corresponding equation. We can also do this for the following question! For example, in the previous ' *Statistical methodology'* section, we have defined *X = (.)* as a vector of covariates. And later, during the derivation of the method, we calculated a new covariate based on this X. Then in the '*Overview of the package refreg'* section I see the word *'covariate'* again, but we don't have anything specific to say about it. I am wondering if the specific *'covariate'* here refers to

- X itself as defined in the previous section

- or is it the new covariance calculated from X (If so, do a recall. This is something you mentioned in the previous section! A few pages back…).
- or if this is just a 'covariate' in the generic definition and doesn't mean anything specific (i.e., not relevant to the previous section)

In the model formulation **X** represents a covariates vector that it is the same for each response parameter. Then we use the term covariate to denominate any predictor variable. We changed the paper using the term predictor variable when possible.

2. It is good for authors to change their representations for the same thing in order to make the article attractive. But as we said, our technique is friendly to people with a "*not strong statistical background*". Perhaps keeping the explanation (i.e., words of interpretation) of the same thing consistent and fixed would be better for the reader reading the article for the first time. Especially if the same thing is shown in different sentences.
For example, I think that *'bivRegr function', 'bivRegr()', 'bivRegr object*', and *'bivRegr-*' all denote the same thing. So, when I first saw *'bivRegion-fitted objects'*, I wondered
- Is there another auxiliary function that calls 'bivRegion-fitted'?
- Or is this just the result of applying the '*bivRegion*' function?
- Or something else...

Thanks for this suggestion, we avoided the use of terms like 'bivRegr- 'bivRegion-fitted objects' and tried to homogenize this concept.

3. If we could be able to remove the 'vague' words, the article might become more friendly. For example, *responses'* (maybe literally it could stand for anything?)
- Do '*responses'* and '*bivRegion-fitted object'* mean the same thing?
- Do '*responses'* and '*y*' (i.e. the bivariate continuous random variable of interest defined in the section "*conditional reference region*" mean the same thing? If so, do a recall (this is something you mentioned in the previous section! A few pages back…).
- Or, if this '*responses'* does mean something else?
- Or, if this '*responses*' just means a general response to any equation? (i.e. it can stand for a variety of things, so give us a notation)

Responses and **y** mean the same thing, we use responses as a synonym of the multivariate regression response in order to highlight that we are working with more than one response variable.

4. Are *'summary_boot', 'summary', 'predict', 'plot'* from the same R package *refreg* or from some other package? As we know, there are some well-known packages that contain functions with these names. If you use them from another package, it is probably best to give credit to someone else. If these are functions that you have defined yourself, it would also be good to specify them.

Those are S3 methods for two main package functions, we clarified this properly in the paper now.

5. I can understand that we only want to highlight the two most important functions (e.g., *bivRegr()* and *bivRegion()*). But the other functions involved in the package are not even given a basic explanation. This led to a sudden appearance of other unexplained functions (e.g., *summary_boot()*) in the section *'refreg in practice'* that followed, which left me feeling somewhat abrupt and confused.

Ok, this S3 functionalities was further explained in an additional paragraph for the reader understanding.

The following R software style problems were corrected in the package code, that it will be uploaded again to CRAN.

ASCII Characters	*tools::showNonASCII()* *tools::showNonASCIIfile(file)*	refcurv.R contains Unicode characters because the names of the authors they cite are not English names.	Specify Unicode characters in the special Unicode escape "\u1234" format.

```
> tools::showNonASCIIfile("refcurv.R")
4: #' in Mart<c3><ad>nez Silva et. al (2016).
10: #' @details In the Mart<c3><ad>nez Silva et. al (2016) the non linear
 effects of the continuous
13: #' @references Mart<c3><ad>nez--Silva, I., Roca--Pardi<c3><b1>as, J.,
 & Ord<c3><b3><c3><b1>ez, C. (2016). Forecasting SO2 pollution incidents
 by means of quantile curves based on additive models. Environmetrics, 27
(3), 147--157.
```

I consulted with some colleagues from Spain, and we do not know how to include accents or ñ letter in R documentations. With the Unicode strategy I obtain the following after running `?refcurv`:

Mart\u00ednez–Silva, I., Roca–Pardi\u00f1as, J., & Ord\u00f3\u00f1ez, C. (2016). Forecasting SO2 pollution incidents by means of quantile curves based on additive models. Environmetrics, 27(3), 147–157.

So, we decided to include the citation without accents, and with n instead of ñ, as in many CRAN documents (see for instance some R packages of other Spanish researchers that ignore accents and the letter ñ in the references https://cran.r-project.org/web/packages/SOP/SOP.pdf, https://cran.r-project.org/web/packages/wsbackfit/wsbackfit.pdf):

Martinez–Silva, I., Roca–Pardinas, J., & Ordonez, C. (2016). Forecasting SO2 pollution incidents by means of quantile curves based on additive models. Environmetrics, 27(3), 147–157.

Thanks for discovering us this package, and check our R code, it is really helpful.

"styler" package	*styler::style_pkg()*	Out of this total of 19 R files, 17 R files had formatting issues, while 2 R files did not.	1. Try put the brackets and parentheses on a new line. 2. Alignment. Smaller formulas contained under larger formulas should be indented with spaces

```
92    return(list(trivres = YC,mean1 = modelo_m1,mean2=modelo_m2,mean3=modelo_m3,
93                var1=modelo_v1,var2=modelo_v2,var3=modelo_v3,
94                rho1=modelo_rho1,rho2=modelo_rho2,rho3=modelo_rho3))
95  }
```

```
95    return(list(
96      trivres = YC, mean1 = modelo_m1, mean2 = modelo_m2, mean3 = modelo_m3,
97      var1 = modelo_v1, var2 = modelo_v2, var3 = modelo_v3,
98      rho1 = modelo_rho1, rho2 = modelo_rho2, rho3 = modelo_rho3
99    ))
100 }
```

I run the styler function and now it seems right

```
> styler::style_pkg()
Styling  19  files:
 R/ACE.R                √
 R/aegis.R              √
 R/bivRegion.R          √
 R/bivRegr.R            √
 R/Hcov.R               √
 R/plot.bivRegion.R     √
 R/plot.bivRegr.R       √
 R/plot.refcurv.R       √
 R/plot.summary_boot.R  √
 R/plot.trivRegion.R    √
 R/pollution.R          √
 R/pollution_episode.R  √
 R/predict.bivRegion.R  √
 R/predict.bivRegr.R    √
 R/refcurv.R            √
 R/summary.bivRegion.R  √
 R/summary_boot.R       √
 R/trivRegion.R         √
 R/trivRegr.R           √
---------------------------------------
Status  Count   Legend
√       19      File unchanged.
i       0       File changed.
x       0       Styling threw an error.
---------------------------------------
```

REVIEWER 2

A. Overview

Your contribution seems meaningful to handle multiple tests for diagnosis and control (e.g. chronic disease – diabetes- and environmental science) beyond the existing packages (referenceIntervals and gamlss). More clarification in writing and illustration would enhance the adoption of the package to a broader context by R community.

B. Article

1. The first paragraph of your *Introduction* can be the beginning of the *Statistical Methodology* to illustrate how your package is different from the existing packages in terms of statistical background and user-friendly simple functions. Inclusion of a table or a figure to demonstrate a point of difference and parity can be extremely helpful to R community working at various fields who evaluate adoption of this package as an alternative to the existing packages.

As you say this paragraph could introduce the statistical methodology section. However, it was corrected following the advice of reviewer #1 in order to offer a better explanation of medical terms introducing the paper.

Regarding the table/figure idea, we do not know any package for estimating conditional reference regions, so it is not clear for us how to make this table, or figure.

2. In page 5, you stated that refreg has only two functions such as bivRegr and bivRegion, but refreg seems to have other functions such as trivRegr and trivRegion according to the Lado-Baleato_etal.R.

I wanted to highlight the two main functions, but as you said, S3 methods must be also explained in order to ease reader understanding. We extended the explanation to make this point clear, by including a new paragraph.

3. Your illustrative example, refreg in practice, may need more clear headings for better readability (e.g., Reference region for two glycemia tests conditioned by age => Case 1: glycemia tests for diabetes research, Joint prediction of two air pollutants => Case 2: Beyond the medical research – Examining SO2 and NOx pollutants with multiple tests).

Changed as suggested

4. In the pages 8-9, the lengthy example codes after a paragraph starting with “The following R output presents a subsample of the patients located outside the standardized bivariate region for $\tau = 0.95$” can be shortened or stylized better with showing R code (R> EXAMPLE CODES) only rather than showing the entire results for readability.

Ok, we just wanted to highlight how the region identifies patients of every age, and the markers values. However, as you say this can be checked by the reader in their R session.

5. Page 12's 'Conditional reference region beyond the bivariate case' part can move to after Figure 4, as the dataset you are using in the part is an extension of Case 1.

Changed as suggested

C. Package – Code improvement

There is a list of errors that I have encountered when I ran the Lado-Baleato_etal.R with RStudio Cloud. If you can address the following errors, R community may be easier to adopt your package without a huge learning curve.

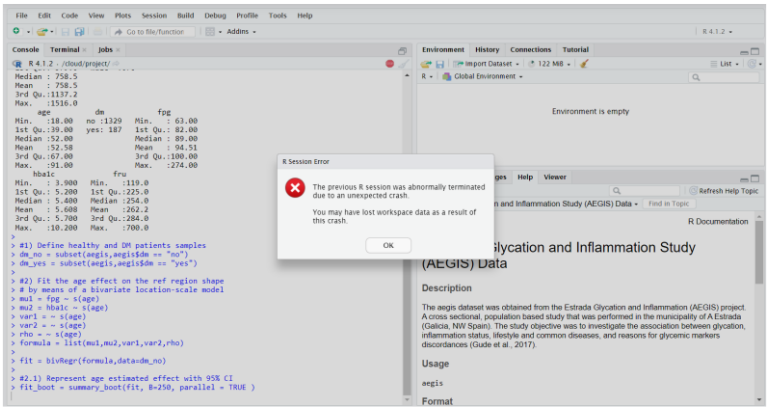
1. Loading Refreg library does not seem smooth like the following warning messages.

```
> library(refreg)
Warning messages:
1: In fun(libname, pkgname) : couldn't connect to display ":0"
2: In rgl.init(initValue, onlyNULL) : RGL: unable to open X11 display
3: 'rgl.init' failed, running with 'rgl.useNULL = TRUE'
```

The package was updated in order to avoid some format issues, and non ASCII characters. It did not show this error in our computers.

2. fit_boot = summary_boot(fit, B=250, parallel = TRUE) has kept not running well due to "unexpected crush."

This function demands a higher computational effort using foreach library, we tried the code in our computers and this error did not happen to us.



3. For the second illustration, I have encountered the following messages.

```
> fit = trivRegr(formula,data=dm_no)
Error in is.data.frame(data) : object 'dm_no' not found
> #2) Trivariate region estimation and representation
> region = trivRegion(fit,tau=0.95)
Error in rep(1, n) : invalid 'times' argument
> plot(region,planes = F,size=5, col="red", incol = "grey", xlab="FPG, mg/dl",
+ ylab="HbA1c, %", zlab="Fru, mg/dL")
There were 24 warnings (use warnings() to see them)
> plot(region,planes = T,size=5,col="red",incol = "grey", xlab="FPG, mg/dl",
```

```
+ ylab="HbA1c, %", zlab="Fru, mg/dL")
There were 24 warnings (use warnings() to see them)
> plot(region,cond=T,newdata=data.frame(age=c(20,70)), xlab="FPG, mg/dl",
+ ylab="HbA1c, %", zlab="Fru, mg/dL", legend=T)
Error in eval(predvars, data, env) : object 'Nox_0' not found
In addition: Warning message:
In predict.gam(object$fit$mu1, newdata) :
not all required variables have been supplied in newdata!
```

Thanks for highlighting this error, we are assuming that the dataset was loaded from the previous example, but if someone wants to reproduce only the three-dimension case, it will obtain this error. We update the code as

```
R> dm_no = subset(aegis,aegis$dm==0)
R> mu1 = fpg ~ s(age)
R> mu2 = hba1c ~ s(age)
R> mu3 = fru ~ s(age)
R> var1 = ~ s(age)
R> var2 = ~ s(age)
R> var3 = ~ s(age)
R> rho12 = ~ s(age)
R> rho13 = ~ s(age)
R> rho23 = ~ s(age)
R> formula = list(mu1,mu2,mu3,var1,var2,var3,rho12,rho13,rho23)
R> fit = trivRegr(formula,data=dm_no)
R> region = trivRegion(fit,tau=0.95)
R> plot(region,planes = F,size=5, col="red", incol = "grey", xlab="FPG, mg/dl",
ylab="HbA1c, %", zlab="Fru, mg/dL")
R> plot(region,planes = T,size=5,col="red",incol = "grey", xlab="FPG, mg/dl",
ylab="HbA1c, %", zlab="Fru, mg/dL")
R> plot(region,cond=T,newdata=data.frame(age=c(20,70)), xlab="FPG, mg/dl",
ylab="HbA1c, %", zlab="Fru, mg/dL", legend=T)
```