1.

Hi, my name is Ryo Nakagawara. Like the previous time I spoke at TokyoR I will again be speaking in English. Some of the slides are in Japanese but I tried to make it as simple as possible for everyone to understand.

2.

A little bit about me, I studied both Psychology and Economics and am now a junior data scientist at ACDI/VOCA, an international development NGO. I am also an editor on the R Weekly newsletter.

3.

My interests are soccer and data visualization with {ggplot2}. If you follow me on Twitter you’ll see that I combine my two interests quite often. Last year I did a talk at Tokyo.R on using R to visualize the World Cup…

--- but this time I’ll be talking about …

4.

My {bulletchartr} package!

5.

So, what exactly IS a bullet chart?

A bullet chart is basically a **variation of a bar chart** that allows you to compare one measure against other metrics such as to a **target** and/or some **qualitative measures of performance** (for example: low – medium – high or poor – satisfactory – good). These qualitative labels are displayed as **varying intensities** of a single color from dark to light.

6.

With the {bulletchartr} package, using the `bullet\_chart()` function, this is what the output looks like.

**PAUSE (2~3 seconds)**

At ACDI/VOCA we also use a modified version of the bullet chart but **first** I need to explain a bit about what we do.

7.

At ACDI/VOCA we implement a lot of **economic development projects** around the world and to evaluate our impact, we create indicators or KPIs that have yearly targets and monitor their progress over time.

8.

So, usually you would gather data gradually in a linear way throughout the course of the project, like the **orange** line on the graph. **But** some of our project activities are **not** linear.

9.

For example, the **purple** line in the graph: for projects based on output from a harvest; a lot of data can only be gathered near the end of the year.

10.

On the other hand, the **green** line, if we’re distributing funds or loans, then most of the data is gathered early in the year.

11.

Our solution to track these **non-linear activities** is by a modified bullet chart using the {bulletchartr} package.

The x-axis now represents **both** the percentage of the yearly target accomplished so far **AND** the percentage of the year that has passed. (We use percentages to normalize different indicators of varying units.)

There is a vertical line showing TODAY, which shows at what percentage of the year AND what percentage of the target we should be at right now.

The color inside the bar is green if we are near or past the TODAY line. Orange when we’re close and red when we’re very behind schedule/target.

12.

For example if TODAY is September, then we’re at 75% of the year. Ind. 4 is at 25% because value is 6 but the target is 24 which is 100%. TODAY we should be at 18 or 75%. We’re normalizing both time and data as percentage of the target on the **same scale**.

13.

We’ve created a couple of variants, such as `bullet\_chart\_symbol()` which uses different shapes to denote the value of the indicator - last week and last year.

And the `bullet\_chart\_vline()` variant, which is a simplified version that just denotes how we were doing at **this time** last year.

14.

Last but not least, there is the `bullet\_chart\_wide()` which is most similar to the regular bullet chart except it uses different widths of the bars to show the benchmarks for previous time points.

15.

Special features include an interactive version powered by `ggiraph`! (g-giraffe) You can hover over the bars to see more information about the indicator.

16.

{bulletchartr} has **many** arguments but the important ones are specifying the legend, calendar type, and showing informational text labels. For example, for Indicator 38 at the top; we collected 4 more data points compared to last week but we are 16 behind compared to the same time last year. Both are still behind what we **should** have today according to the TODAY line.

17.

The hardest part of creating this package was converting it from in-house/company only usage to generalizing it for the public and open sourcing it.

I didn’t have time to go over them but there are **a lot more** function arguments and I have tried to create easy defaults to make it easy on the user. Please see the package vignettes for more details!

Right now we only have qualitative labels set for “Low”, “Medium”, and “High” in the regular version and “Last Week”, “Last Year” in the time-comparison version but in later updates I would like to allow users to set their own labels.

Finally, I am aiming for a CRAN release by the end of the year.

18.

Thank you for listening! Check out my other stuff on Github and Twitter!