Exploring probability distributions for bivariate temporal granularities

Abstract

Recent advances in technology greatly facilitates recording and storing data at much finer temporal scales than was previously possible. As the frequency of time-oriented data increases, the number of questions about the observed variable that need to be addressed by visual representation also increases. We propose some new tools to explore this type of data, which deconstruct time in many different ways. There are several classes of time deconstructions including linear time granularities, circular time granularities and aperiodic calendar categorizations. Linear time granularities respect the linear progression of time such as hours, days, weeks and months. Circular time granularities accommodate periodicities in time such as hour of the day, and day of the week. Aperiodic calendar categorizations are neither linear nor circular, such as day of the month or public holidays.

The hierarchical structure of linear granularities creates a natural nested ordering resulting in single-order-up and multiple-order-up granularities. For example, hour of the week and second of the hour are both multiple-order-up, while hour of the day and second of the minute are single-order-up.

Visualizing data across granularities which are either single-order-up or multiple-order-up or periodic/aperiodic helps us to understand periodicities, pattern and anomalies in the data. Because of the large volume of data available, using displays of probability distributions conditional on one or more granularities is a potentially useful approach. This work provides tools for creating granularities and exploring the associated time series within the tidy workflow, so that probability distributions can be examined using the range of graphics available in (Wickham 2016).

Contents

1 Introduction

Temporal data can be available at various resolution depending on the context. Social and economic data are often collected and reported at coarser temporal scales like monthly, quarterly or annually. But with recent advancement in technology, more and more data are recorded and stored at much finer temporal scales than that was previously possible. It might be sufficient to observe energy consumption every half an hour, but energy supply needs to be monitored every minute and number of web searches requires optimization every second. As the frequency of data increases, the number of questions about the observed variable that need to be addressed also increases. For example, data collected at an hourly scale can be analyzed using coarser temporal scales like days, months or quarters. This approach requires deconstructing time in various possible ways.

A temporal granularity which results from such a deconstruction may be intuitively described as a sequence of time granules, each one consisting of a set of time instants. There are several classes of time deconstructions including linear time granularities, circular time granularities and aperiodic calendar categorizations. Linear time granularities respect the linear progression of time such as hours, days, weeks and months. Circular time granularities accommodate periodicities in time such as hour of the day, and day of the week. Aperiodic calendar categorizations are neither linear nor circular, such as day of the month or public holidays.

It is important to be able to navigate through all of these temporal granularities to have multiple perspectives on the observed data. This idea aligns with the notion of EDA (Tukey 1977) which emphasizes the use of multiple perspectives on data to help formulate hypotheses before proceeding to hypothesis testing.

The motivation for this work comes from the desire to provide methods to better understand large quantities of measurements on energy usage reported by smart meters in household across Australia, and indeed many parts of the world. Smart meters currently provide half-hourly use in kWh for each household, from the time that they were installed, some as early as 2012. Households are distributed geographically, and have different demographic properties such as the existence of solar panels, central heating or air conditioning. The behavioral patterns in households vary substantially, for example, some families use a dryer for their clothes while others hang them on a line, and some households might consist of night owls, while others are morning larks.

It is common to see aggregates of usage across households, total kWh used each half hour by state, for example, because energy companies need to understand maximum loads that they will have to plan ahead to accommodate. But studying overall energy use hides the distributions of usage at finer scales, and making it more difficult to find solutions to improve energy efficiency.

We propose that the analysis of probability distributions of smart meter data at finer or coarser scales can be benefited from the approach of Exploratory Data Analysis (EDA). EDA calls for utilizing visualization and transformation to explore data systematically. It is a process of generating hypothesis, testing them and consequently refining them through investigations.

The hierarchical structure of many granularities creates a natural nested ordering. For example, hours are nested within days, days within weeks, weeks within months, and so on. We refer to granularities which are nested within multiple levels as "multiple-order-up" granularities. For example, hour of the week and second of the hour are both multiple-order-up, while hour of the day and second of the minute are single-order-up.

This paper utilizes the nestedness of time granularities to obtain multiple-order-up granularities from single-order-up ones.

Finally, visualizing data across single/multiple order-up granularities help us to understand periodicities, pattern and anomalies in the data. Because of the large volume of data available, using displays of probability distributions conditional on one or more granularities is a potentially useful approach. However, this approach can lead to a myriad of choices all of which are not useful. Analysts are expected to iteratively visualize these choices for exploring possible patterns in the data. But too many choices might leave him bewildered.

This work provides tools for systematically exploring bivariate granularities within the tidy workflow through proper study of what can be considered a prospective graphic for exploration. Pairs of granularities are categorized as either a *harmony* or *clash*, where harmonies are pairs of granularities that aid exploratory data analysis, and clashes are pairs that are incompatible with each other for exploratory analysis. Probability distributions can be examined using the range of graphics available in the ggplot2 package.

In particular, this work provides the following tools.

- Functions to create multiple-order-up time granularities. This is an extension to the lubridate package, which allows for the creation of some calendar categorizations, usually single-order-up.
- Checks on the feasibility of creating plots or drawing inferences from two granularities together. Pairs of granularities can be categorized as either a *harmony* or *clash*, where harmonies are pairs of granularities that aid exploratory data analysis, and clashes are pairs that are incompatible with each other for exploratory analysis.

2 Formal conceptualization of calendar categorizations

Often we partition time into months, weeks, days and so on in a hierarchical manner to relate it to data. Such discrete abstractions of time can be thought of as time granularities (Aigner et al. 2011). Examples of time abstractions may also include day-of-week, time-of-day, week-of-year, day-of-month, month-of-year, working day/non-working day, etc which are useful to represent different periodicities in the data. Let us call all of these different abstractions of time as “calendar categorizations”.

These calendar categorizations can be linear, circular or aperiodic. The calendar categorizations as “linear” if they respect the linear progression of time. We call these **linear time granularities**. Examples include hours, days, weeks and months. **Circular time granularities** accommodate periodicities in time such as hour of the day, and day of the week. **Aperiodic time granularities** are neither linear nor circular, such as day of the month or public holidays.

Providing a formalism to these abstractions is important to model a time series across differently grained temporal domains.

2.1 Linear time granularities

There has been several attempts to provide the framework for formally characterizing time-granularities and identifying their structural properties, relationships and symbolic representations. One of the first attempts occur in (Bettini et al. 1998) with the help of the following definitions:

Definition: A **time domain** is a pair $(T; \leq)$ where T is a non-empty set of time instants and $\leq$ is a total order on T.

A time domain can be **discrete** (if there is unique predecessor and successor for every element except for the first and last one in the time domain), or it can be **dense** (if it is an infinite set). A time domain is assumed to be discrete for the purpose of our discussion.

Definition: A linear **granularity** is a mapping G from the integers (the index set) to subsets of the time domain such that:

(C1) if $i < j$ and $G(i)$ and $G(j)$ are non-empty, then each element of $G(i)$ is less than all elements of $G(j)$, and
(C2) if $i < k < j$ and $G(i)$ and $G(j)$ are non-empty, then $G(k)$ is non-empty.

Definition: Each non-empty subset $G(i)$ is called a **granule**, where i is one of the indexes and G is a linear granularity.

The first condition implies that the granules in a linear granularity are non-overlapping and their index order is same as time order. Figure 1 shows the implication of this condition. If we consider the bottom linear granularity (Aigner et al. 2011) as hourly and the entire horizon has T hours then it will have $\lfloor T/24 \rfloor$ days, $\lfloor s/(24 * 7) \rfloor$ weeks and so on.

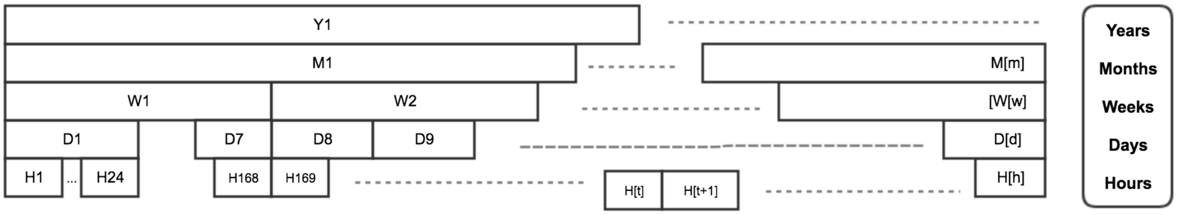


Figure 1: The time domain distributed as linear granularities

The definitions and rules for linear granularities are inadequate to reflect periodicities in time, like weekly, monthly or yearly seasonality. Hence, there is a need to define circular time granularities in a different approach.

2.2 Circular time granularities

A time domain, as defined by (Bettini et al. 1998), is essentially a mapping of row numbers (the index set) to the time index in a tsibble(Wang, Cook, and Hyndman 2019) for a given key. A linear granularity is a mapping of row numbers to subsets of the time domain. For example, if the time index is days, then a linear granularity might be weeks, months or years. What we need to add to this are additional categorizations of time that are not linear granularities and are useful to represent periodicity.

MOH:	$C_1(s) = s \mod 60$	$n_1 = \ \ 60$
MOD:	$C_2(s) = s \mod 1440$	$n_2 = 1440$
HOD:	$C_4(s) = \lfloor s/60 \rfloor \mod 24$	$n_4 = \ \ 24$
HOW:	$C_6(s) = \lfloor s/60 \rfloor \mod 24 * 7$	$n_5 = \ \ 168$
DOW:	$C_7(s) = \lfloor s/24 * 60 \rfloor \mod 7$	$n_6 = \ \ 7$

Table 1: Illustrative circular granularities with time index in minutes

We want to use modular arithmetic to define circular granularity. Hence, we start with the definition of equivalence classes and then move on to define a circular granularity.

Definition: Equivalence class Let $m \in N\backslash 0$. For any $a \in Z$ (set of integers), $[a]$ is defined as the equivalence class to which a belongs if $[a] = \{b \in Z | a \equiv (b \mod m)\}$.

The set of all equivalence classes of the integers for a modulus m is called the ring of integers modulo m, denoted by Z_m. Thus $Z_m = \{[0], [1], ..., [m-1]\}$. However, we often write $Z_m = \{0, 1, ..., (m-1)\}$, which is the set of integers modulo m.

Definition: A **circular granularity** C with a modular period m is defined to be a mapping from the integers Z (Index Set) to Z_m, such that $C(s) = (s \mod m)$ for $s \in Z$.

For example, suppose C is a circular granularity denoting Hour-of-Day and we have hourly data for 100 hours. The modular period $m = 24$, since each day consists of 24 hours and C is a mapping from $1, 2, \ldots, 100$ to $0, 1, 2, \ldots, 23$ such that $C(s) = s \mod 24$ for $s \in 1, 2, \ldots, 100$.

Definition: A **cycle** is defined as the progression of each circular granularity with modular period m through $\{1, 2, \ldots, (m-1), 0\}$ once.

Definition: A **circular granule** represents an equivalence class inside each cycle.

2.3 Aperiodic time granularities

Definition: An **Aperiodic circular granularity** can not be defined using modular arithmetic in a similar fashion. The modulus for these type of calendar categorizations are not constant due to unequal length of some linear granularities. For example, please refer to the table below:

HOM:	$C_3(s) = s \mod 720$ (approximately)	$n_3 = 744$
HOY:	$C_4(s) = s \mod 8760$ (except for leap years)	$n_4 = 8784$
DOM:	$C_6(s) = \lfloor s/24 \rfloor \mod 30$ (approximately)	$n_6 = 31$
DOY:	$C_7(s) = \lfloor s/24 \rfloor \mod 365$ (except for leap years)	$n_7 = 366$
WOM:	$C_8(s) = \lfloor s/168 \rfloor \mod 4$ (approximately)	$n_8 = 5$
WOY:	$C_9(s) = \lfloor s/168 \rfloor \mod 52$ (approximately)	$n_9 = 53$
MOY:	$C_{10}(s) = \lfloor s/720 \rfloor \mod 12$ (approximately)	$n_{10} = 12$

Table 2: Illustrative aperiodic circular granularities with time index in hours

Identifying repeating (periodic/aperiodic) patterns are necessary in revealing patterns and future trends of a temporal data. Often there is a need for periodicity detection to find whether and how frequent a periodic/aperiodic pattern is repeated within the series. To consider the exhaustive set of temporal regularities that might exist in the data, we can categorize the set of temporal granularities to single or multiple order up granularities.

3 Single-order-up and multiple-order-up granularities

The hierarchical structure of time creates a natural nested ordering, where hours are nested within days, days within weeks, weeks within months, and so on. We refer to granularities which are nested within multiple levels as **multiple-order-up** granularities. For example, hour of the week and second of the hour are both multiple-order-up, while hour of the day and second of the minute are **single-order-up**.

3.1 Single-order-up granularities

The day is the basic unit of time underlying all calendars. The most common way of defining days are from sunrise to sunrise (Hindu calendar, for example) or from sunset to sunset (Islamic and Hebrew calendars). Further, various calendars use different conventions to structure days into larger units: weeks, months, years and cycle of years. Any civil day is divided into 24 hours and each hour into 60 minutes and each minute into 60 seconds. There are exceptions where Greeks and Romans separate the "day" and "night" into 12 hours each. In London, for example, length of each "hour" varies from about 39 minutes in December to 83 minutes in June. The French revolutionary calendar divided each day into 10 "hours", each "hour" into 100 "minutes" and each "minute" into 100 "seconds".

Let us define the arrangement of these coarser temporal units from finer temporal units in an hierarchy table. For example, in "Mayan" calendar, one day is referred to as 1 kin and they believed that the universe is destroyed at the start of every cycle (2,880,000 days). The Mayan calendar was structured as follows:

- 1 kin = 1 day
- 1 uinal = 20 kin
- 1 tun = 18 uinal
- 1 katun = 20 tun
- 1 baktun = 20 katun

Thus, the hierarchy table for the "Mayan" calendar would look like the following:

Temporal units	Conversion factor	conversion in days	single-order-up accessors
1 kin	1 day		kin_uinal
1 uinal	20 kin	20 days	uinal_tun

Temporal units	Conversion factor	conversion in days	single-order-up accessors
1 tun	18 uinal	360 days	tun_katun
1 katun	20 tun	7200 days	katun_baktun
1 baktun	20 katun	144,000 days	1

Single-order-up granularities like kin_uinal(kin of the uinal), uinal_tun(uinal of the tun) can be computed using modular arithmatic in this periodic set up.

3.2 Computation of multiple order-up granularities

The hierarchical nature of time helps to create a framework where the single-order-up granularities can be used recursively to create multiple-order-up granularities.The idea is to incorporate and access all possible calendar categorizations to account for different periodicities in the data. The computation of multiple-order-up granularities from single-order-up granularities can differ basis the single-order-up granularities are circular or aperiodic.

3.2.1 Periodic single-order-up granularities

Definition: A **hierarchy table** has three columns:
- The first column represents the linear granularities in ascending order of temporal hierarchy.
- The second column represents the constant which relates subsequent linear granularities.
- The third column provides the accessor function for single-order up circular granularities.

Definition: **Order** is defined as the position of the linear granularities in the hierarchical table.

Suppose, z is the index set of the tsibble, x,y are two linear granularities with $order(x) < order(y)$. Also, $f_{(x,y)}$ denotes the accessor function for computing circular granularity "x_y" and $c(x,y)$ is a constant which relates x and y. It is easy to see that for $order(x+1) = order(y)$, the function is same as the single-order-up granularities.

Then, the accessor function f can be used recursively to obtain any multiple-order-up granularities as follows:

$$\begin{aligned} f_{(x,y)}(z) &= f_{(x,x+1)}(z) + c(x,x+1)(f_{(x+1,y)}(z)-1) \\ &= f_{(x,x+1)}(z) + c(x,x+1)[f_{(x+1,x+2)}(z) + c(x+1,x+2)(f_{(x+2,y)}(z)-1)-1] \\ &= f_{(x,x+1)}(z) + c(x,x+1)(f_{(x+1,x+2)}(z)-1) + c(x,x+1)c(x+1,x+2)(f_{(x+2,y)}(z)-1) \\ &= f_{(x,x+1)}(z) + c(x,x+1)(f_{(x+1,x+2)}(z)-1) + c(x,x+2)(f_{(x+2,y)}(z)-1) \\ &\vdots \\ &= \sum_{i=0}^{order(y)-order(x)-1} c(x,x+i)(f_{(x+i,x+i+1)}(z)-1) \end{aligned} \tag{1}$$

For illustration, let us consider the following hierarchy table.

linear granularities	Conversion factor	Single-order-up accessors
second	60	second_minute
minute	30	minute_hhour
hhour	2	hhour_hour
hour	24	hour_day
day	7	day_week
week	1	1

From Equation(1), we have

$$\begin{aligned} f_{(hhour,week)}(z) &= f_{(hhour,hour)}(z) + c(hhour,hour)f_{(hour,day)}(z) + c(hhour,day)f_{(day,week)}(z) \\ &= hhour\_hour(z) + 2hour\_day(z) + 2 * 24day\_week(z) \end{aligned} \tag{2}$$

3.2.2 Mixed single-order-up granularities - circular or aperiodic

Suppose we extend the hierarchy table to include the linear granularities month and quarter. Since months consists of unequal number of days, any linear granularity which is higher in order than months will also have unequal number of days. Hence, any calendar categorization which includes one linear granularity with orders at most as days and another one with orders at least as months will be aperiodic in nature. However, it is very much possible for a calendar categorization consisting of two linear granularities higher in order than month to be periodic.For example, the categorization month_quarter would be periodic since each quarter always consists of three months.

linear granularities	Conversion factor	Single-order-up accessors
second	60	second_minute
minute	30	minute_hhour
hhour	2	hhour_hour
hour	24	hour_day
day	aperiodic	day_month
month	3	month_quarter
quarter	1	1

There can be three scenarios for obtaining calendar categorization here: - calendar categorization consisting of two linear granularities whose orders are less than day
- calendar categorization consisting of two linear granularities whose orders are more than month
- calendar categorization consisting of one linear granularity with order at most day and another with order at least month

The calendar categorization resulting from the first two cases are periodic and has been handled in the earlier section.

The calendar categorization resulting from the last case are aperiodic. Examples might include day of the quarter or hour of the month. In this section, we will see how to obtain aperiodic circular granularities of these types.

$$f_{(hour,month)}(z) = f_{(hour,day)}(z) + c(hour,day)f_{(day,month)}(z) \tag{3}$$

Here, the first part of the equation is a single-order-up granularity which can be obtained using last section. The second part is not single-order-up and can not be broken down further since each month consists of different number of days. In this case, it is important for us to know which month of the year and if the year is a leap year to obtain day of the month.

4 Visualization

Analysts often want to fit their data to statistical models, either to test hypotheses or predict future values. However, improper choice of models can lead to wrong predictions. One important use of visualization is exploratory data analysis, which is gaining insight into how data is distributed to inform data transformation and modeling decisions.

But with huge amount of data being available, sometimes mean, median or any one summary statistic is not enough to understand a data set. Soon enough following questions become more interesting:

- Are values clustered around mean/median or mostly around tails? In other words, what is the combined weight of tails relative to the rest of the distribution?
- Does values rise very quickly between 25th percentile and median but not as quickly between median and 75th percentile? More generally, how the variation is the data set changes across different percentiles/deciles?
- Is the tail on the left hand side longer than that on the right side? Or are they equally balanced around mean/median?

This is when displaying a probability distribution becomes a potentially useful approach.

The entire distribution can be visualized or some contextual summary statistics can be visualized to emphasize certain properties of the distribution. These properties can throw light on central tendency, skewness, kurtosis, variation of the distribution and can also be useful in detecting extreme behavior or anomalies in the data set.

4.1 Statistical distribution plots

Most commonly used techniques to display distribution of data include the histogram (Karl Pearson), which shows the prevalence of values grouped into bins and the box-and-whisker plots (Tukey 1977) which convey statistical features such as the median, quartile boundaries, hinges, whiskers and extreme outliers. The box plot is a compact distributional summary, displaying less detail than a histogram. Due to wide spread popularity and simplicity in implementation, a number of variations are proposed to the original one which provides alternate definitions of quantiles, whiskers, fences and outliers. Notched box plots (Mcgill, Tukey, and Larsen 1978, 1978) has box widths proportional to the number of points in the group and display confidence interval around medians aims to overcome some drawbacks of box plots.

The vase plot (Benjamini 1988, 1988) was a major revision from the concept of box plots where the width of box at each point is proportional to estimated density. Violin plots (Hintze and Nelson 1998, 1998) display the density for all data points and not only the box. The summary plot (Potter et al. 2010, 2010) combines a minimal box plot with glyphs representing the first five moments (mean, standard deviation, skewness, kurtosis and tailings), and a sectioned density plot crossed with a violin plot (both color and width are mapped to estimated density), and an overlay of a reference distribution. The highest density region (HDR) box plot proposed by (Hyndman 1996) displays a probability density region that contains points of relatively highest density. The probabilities for which the summarization is required can be chosen based on the requirement. These regions do not need to be contiguous and help identify multi-modality. The letter-value box plot (Hofmann, Wickham, and Kafadar 2017, 2006) was designed to adjust for number of outliers proportional to the data size and display more reliable estimates of tail. Because this display just adds extra letter values, it suffers from the same problems as the original box plot, and multimodality is almost impossible to spot(Wickham and Stryjewski, n.d.).

Moreover, much like the quartiles divide the data set equally into four equal parts, extensions might include dividing the data set even further. The deciles plots consist of 9 values that split the data set into ten parts and the percentile plot consists of 99 values that split the data set into hundred parts. A large data set is required before the extreme percentiles can be estimated with any accuracy.

Finally, a density plot which uses a kernel density estimate to show the probability density function of the variable can show the entire distribution. Also, a Ridge line plot (sometimes called Joy plot) shows the distribution of a numeric value for several groups. Distribution can be represented using histograms or density plots, all aligned to the same horizontal scale and presented with a slight overlap.

4.2 Harmony and Clashes

We investigate some combinations of circular time granularities (periodic/aperiodic) which facilitate or hinder exploratory analysis. The combinations of circular granularities which promote the exploratory analysis through visualization are referred to as **harmonies** and the ones which impede the analysis are referred to as **clashes**.

Figure 2 (a) shows the letter value plot of electricity consumption of Victoria across days of the month for few months like January, February, April and December. Letter value plots convey detailed information about tails of the distribution and outliers are unexpected observations rather than extreme observations. M, F, E, D and C represents 50%, 25%, 12.5%, 6.25% and 3.13% of the tail area respectively. Some observations that can be made from these letter-value plots include a) Right tails for most days in January lying above the median is more extended than the left tails contrary to days in April which has longer left tails. b) Days in mid January and February are characterized by high variation in consumption level, but days in January have longer right tails. c) Last five days in December (Christmas holidays) shows low variation in consumption with longer left tails implying people typically consume less electricity. We can conclude that Month-of-Year and Day-of-Month are harmonies.

Figure 2 (c) shows box plot of electricity consumption of Victoria from 2012 to 2014 across days of the year by the 1st, 15th, 29th and 31st days of the month. The box plot is a a very compact distributional summary, displaying median, quartile boundaries, hinges, whiskers and outliers. All facets do not contain data across same x-axis levels leading to difficulty in comparison across facets. Hence, Day-of-Month and Day-of-Year are clashes.

Take another example Figure 2 (d) showing violin plot across days of the month faceted by week of the month. Violin plots are a combination of box plot and density plot. Here, the first week of the month correspond to certain days of the month which are different for different week of the month. These kinds of graphics can hinder our comprehension across different weeks of the month and can be categorized as a incompatible combination to plot.

In Figure 2 (e), variations across week of the year conditional on week of the month can be observed through a ridge plot. The y-axis represents week of the year and the x-axis represents electricity consumption. Ridge plots are density plots all aligned to the same horizontal scale and presented with a slight overlap. They are designed to bring out changes in distributions over time. The data is distributed unequally for different levels of weeks of the year for different facets, which hinders comparison across facets. So, Week-of-Month and Week-of-Year are clashes.

Figure 2 (f) shows decile plots of consumption across day of the year and month of the year. Decile plots are useful in displaying distribution through deciles without much clutter. This plot, however, seem ineffective for comparison as levels of x-axis for which observed data is plotted are completely disjoint across facets. So Month-of-Year and Day-of-Year are clashes as well.

From the above examples, we can see that the choices of the circular granularities are harmonies while some are clashes. Also since some choices work and others don't, it must be the attributes of the circular granularities or their relationships which are in play in deciding if the resulting plot would be a good candidate for exploratory analysis.

In Figure 2 (c), we have empty combinations when we plot observations with "Day-of-Month" and "Day-of-Year". Here, the 1st day of the month can never correspond to 2nd, 3rd or 4th day of the year. Hence, "Day-of-Month" and "Day-of-Year" are clashes due to the way they map to calendar.

In Figure 2 (a), we will not have any empty combinations because every DoM can occur in all MoY. Here, mapping from days to months is irregular in the sense that one month can consist of 28, 29, 30 or 31 days. There is no denying that the 29th, 30th or 31st day of the month needs to be analysed with caution due to the irregular mapping, however, the first 28 days of the month will occur in all months of the year.

More generally, if we have two circular granularities C1 and C2 which has [A, B, C, D] and [X, Y, Z] categories/levels. When C1 and C2 are used as aesthetics or facets, problems arise when we encounter empty combinations. Let $S_{i,j}$ be the set of combination of the levels of C1 and C2. In this example, we have 12 such sets $S_{i,j}$ because i can take 4 values and j can take 3 values. The graphs that don't work are those where many of these 12 sets are empty. In other words, if there are levels of the x-axis which are not spanned by levels of the faceted variable or vice versa we will have structurally empty sets leading to potential ineffective graphs.

Thus, the common link that differentiates harmonies from clashes are:

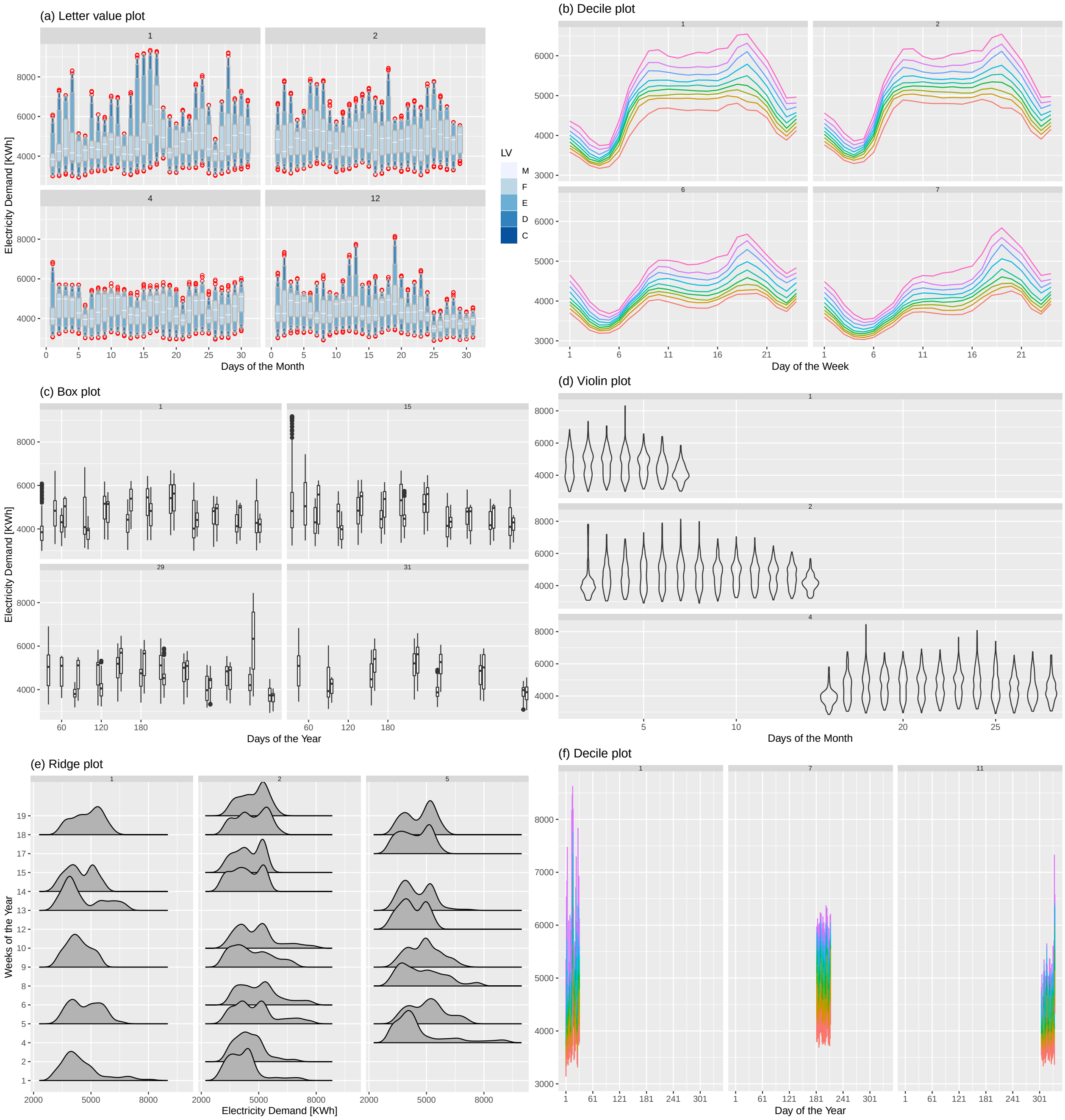


Figure 2: Various probability distribution plots of electricity consumption data of Victoria from 2012 to 2014. (a) Letter value plot by DoM and MoY, (b) Decile plot by HoD and DoW (c) Box plot by DoY and DoM, (d) Violin plot of DoM and WoM, (e) Ridge plot by WoM and WoY, (f) Decile plot by DoY and MoY. Only plots (a) and (b) show harmonised time variables.

a) There should not be any levels of the faceted variable which is empty for one or more levels of the factor plotted across x-axis.

b) There should not be any level of the factor plotted across x-axis which doesn't have values for all levels of factors plotted across facets.

The time series variable can be plotted against many time granularities to get more understanding of the underlying periodicty, however, we will restrict ourselves to see the distribution of the time series across bivariate temporal granularities. That neccessiates plotting one temporal granularity along the x-axis and the other one across facets.

Now, due to the hierarchical arrangement of the granularities, there are certain granularities which when plotted together do not give us the layout to do exploration, for example, structurally empty combinations (clashes) are not recommended to plot together. The harmonies when plotted together can help exploration. But still the question remains that which distribution plot should be chosen to bring out the best of exploratory data analysis. This is a function of which features of the distribution we are interested to look at, how much display space is available to us and also if the number of observations are enough for that distribution plot.

4.3 Advice algorithm for exploring conditional probability distributions

Recommendations for distribution plots depend on the levels(very high/high/medium/low) of the two granularities plotted. They will vary depending on which granularity is placed on the x-axis and which one across facets. Assumptions are made to ensure display is not too cluttered by the space occupied by various kinds of distribution plots. Moreover, the recommendation system ensures that there are just enough observations before choosing a distribution plot.

Levels are categorized as very high/high/medium/low each for the facet variable and the x-axis variable. Default values for these levels are based on levels of common temporal granularities like day of the month, day of a fortnight or day of a week. For example, any levels above 31 is considered as very high, any levels between 14 to 31 are taken as high and that between 7 to 14 is taken as medium and below 7 is low. 31, 14 and 7 are the levels of days-of-month, days-of-fortnight and days-of week respectively. These default values are decided based on usual cognitive power while comparing across facets and display size available to us. Let us consider case by case and see which plots are better suitable in which scenarios.

- very high facet and x-axis levels

X-axis variable is treated as a categorical variable and hence any plots which

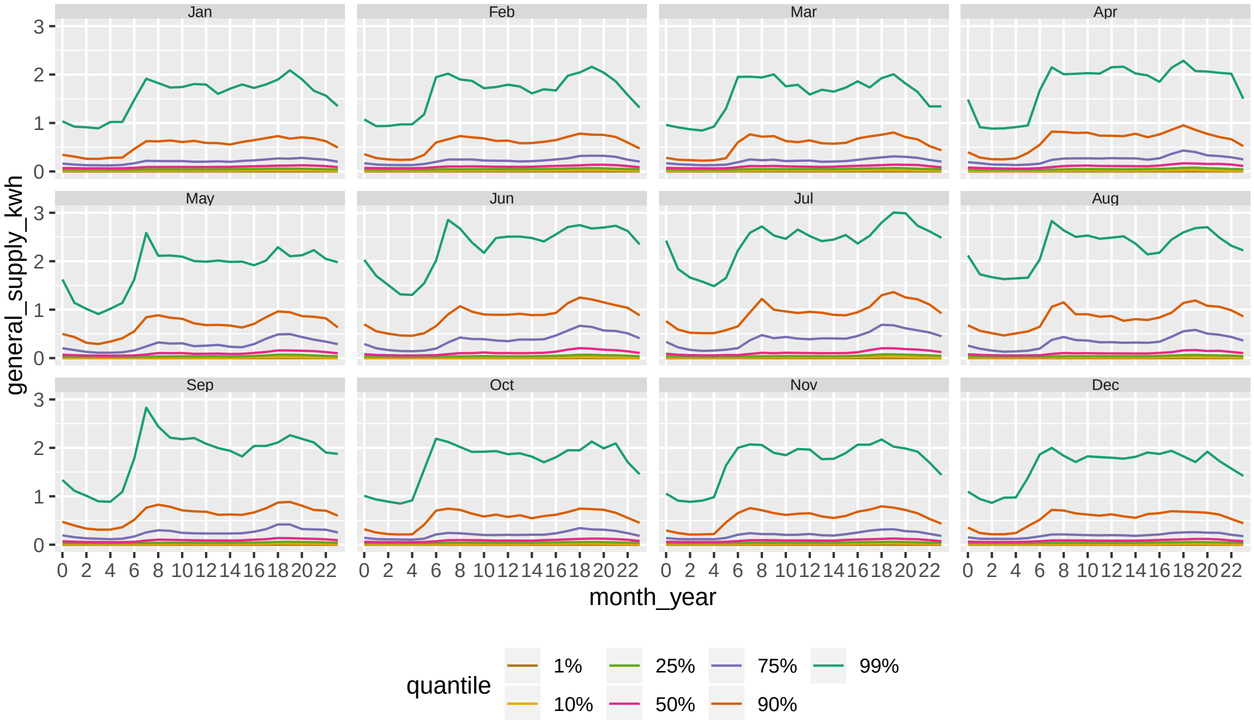
5 Case study: Analysis on smart meter data

Smart meters provide large quantities of measurements on energy usage for households across Australia, and indeed many parts of the world. Households are distributed geographically and have different demographic properties such as the existence of solar panels, central heating or air conditioning. The behavioral patterns in households vary substantially, for example, some families use a dryer for their clothes while others hang them on a line, and some households might consist of night owls, while others are morning larks.

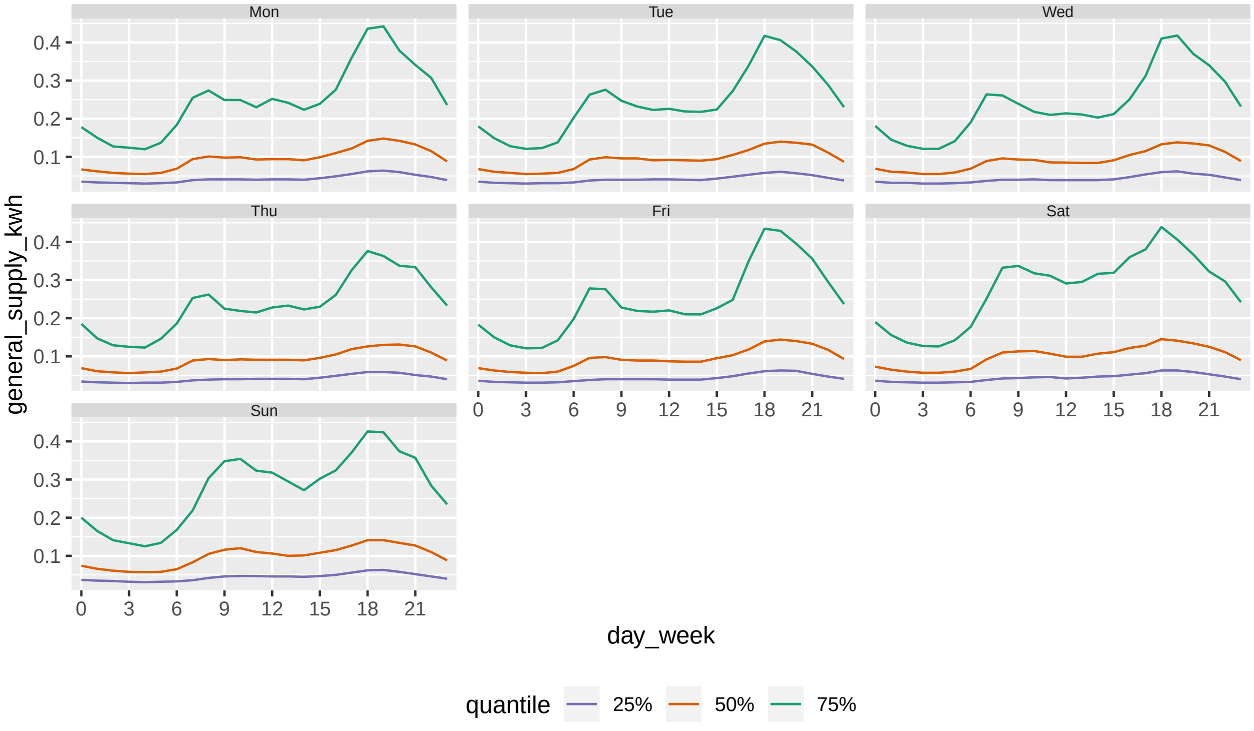
It is common to see aggregates of usage across households, total kwh used each half-hour by state, for example, because energy companies need to understand maximum loads that they will have to plan ahead to accommodate. But studying overall energy use hides the distributions of usage at finer scales, and making it more difficult to find solutions to improve energy efficiency.

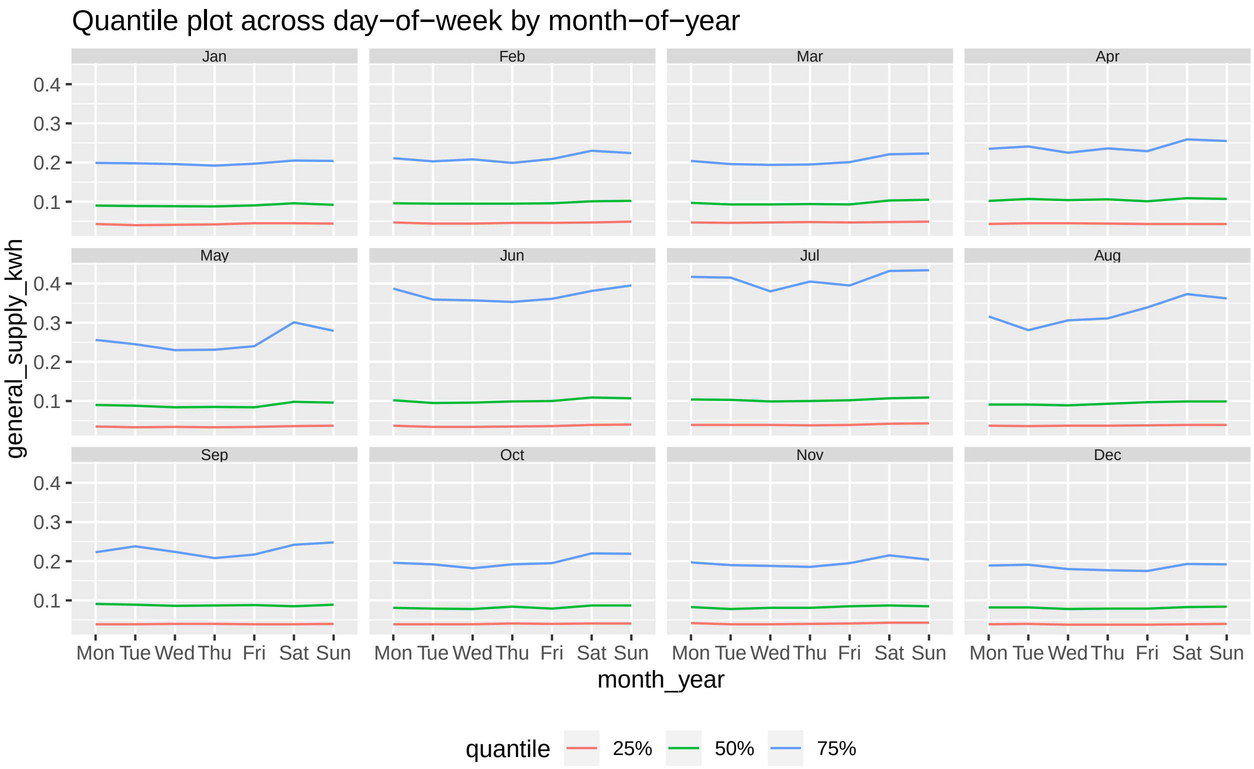
One of the customer trial (Department of the Environment and Energy 2018) conducted as part of the Smart Grid Smart City (SGSC) project (2010-2014) in Newcastle, New South Wales and some parts of Sydney provides customer wise data on half-hourly energy usage and detailed information on appliance use, climate, retail and distributor product offers, and other related factors. It would be interesting to explore the energy consumption distribution for these customers and gain more insights on their energy behavior which are otherwise lost either due to aggregation or looking only at coarser temporal units. The idea here is to show how looking at the time across different granularities together can help identify different behavioral patterns.

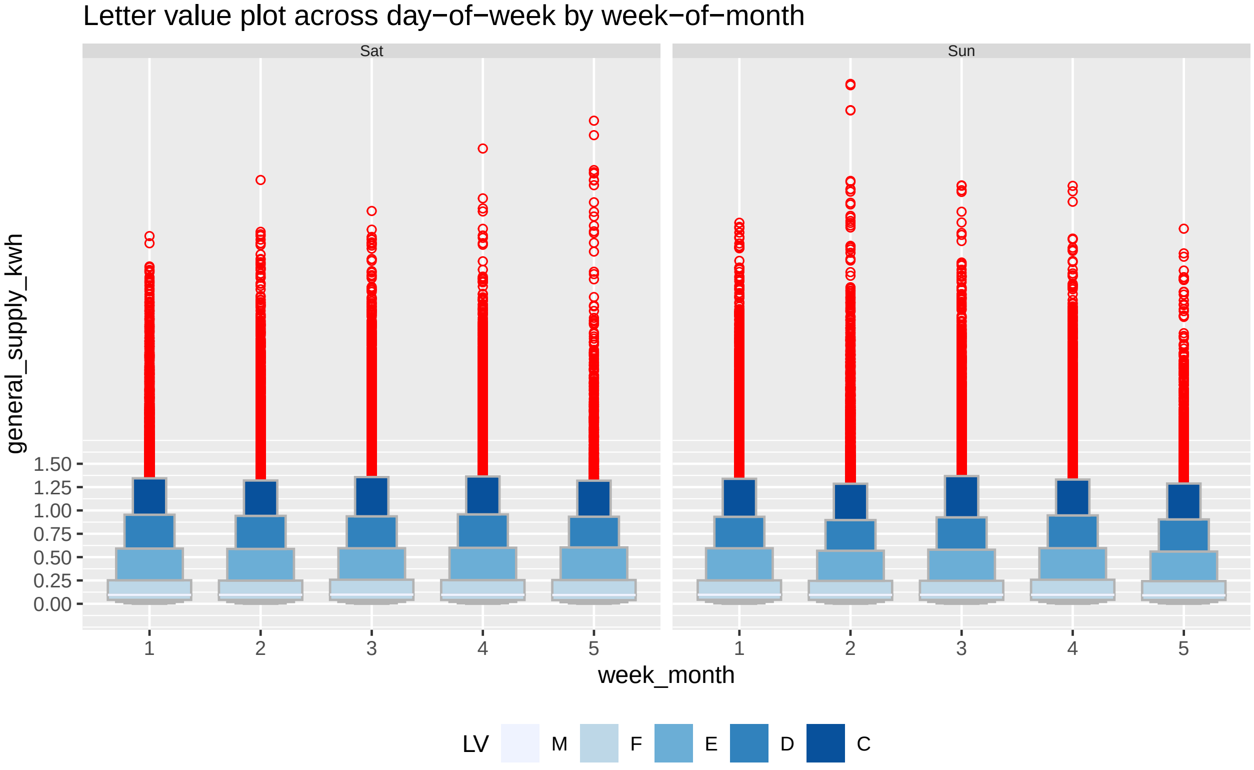
Let us see the behavior of typical and extreme behaviors of 50 households of this trial. We want to look at the distribution of energy across coarser temporal granularities and then deep dive into finer temporal granularities.



Quantile plot across hour−of−day and day−of−week







6 Case study: Analysis on cricket

Th application is not only restricted to temporal data. We provide an example of cricket to illustrate how this can be generalised in other applications. The Indian Premier League (IPL) is a professional

Table 6: T20 hierarchy table

units	convert_fct
index	1
ball	6
over	20
inning	2
match	1

Twenty20 cricket league in India contested by eight teams representing eight different cities in India. With eight teams, each team plays each other twice in a home-and-away round-robin format in the league phase. In a Twenty20 game the two teams have a single innings each, which is restricted to a maximum of 20 overs. Hence, in this format of cricket, a match will consist of 2 innings, an innings will consist of 20 overs, an over will consist of 6 balls. We have sourced the the ball by ball data for IPL 2008 from https://www.kaggle.com/littleraj30/indian-premier-league-2019-ball-by-ball. There are many interesting questions that can possibly be answered with such a dataset, however, we will explore a few and understand how the proposed approach in the paper can help answer some of the questions.

The dataset contains the information on match_id, inning, batting team, bowling team, over of the innings, balls of the over, total runs and many more. A hierarchy table like [] can be construed for this game format:

Each team is given a two-and-a-half-minute “strategic timeout” during each innings; one must be taken by the bowling team between the ends of the 6th and 9th overs, and one by the batting team between the ends of the 13th and 16th overs.

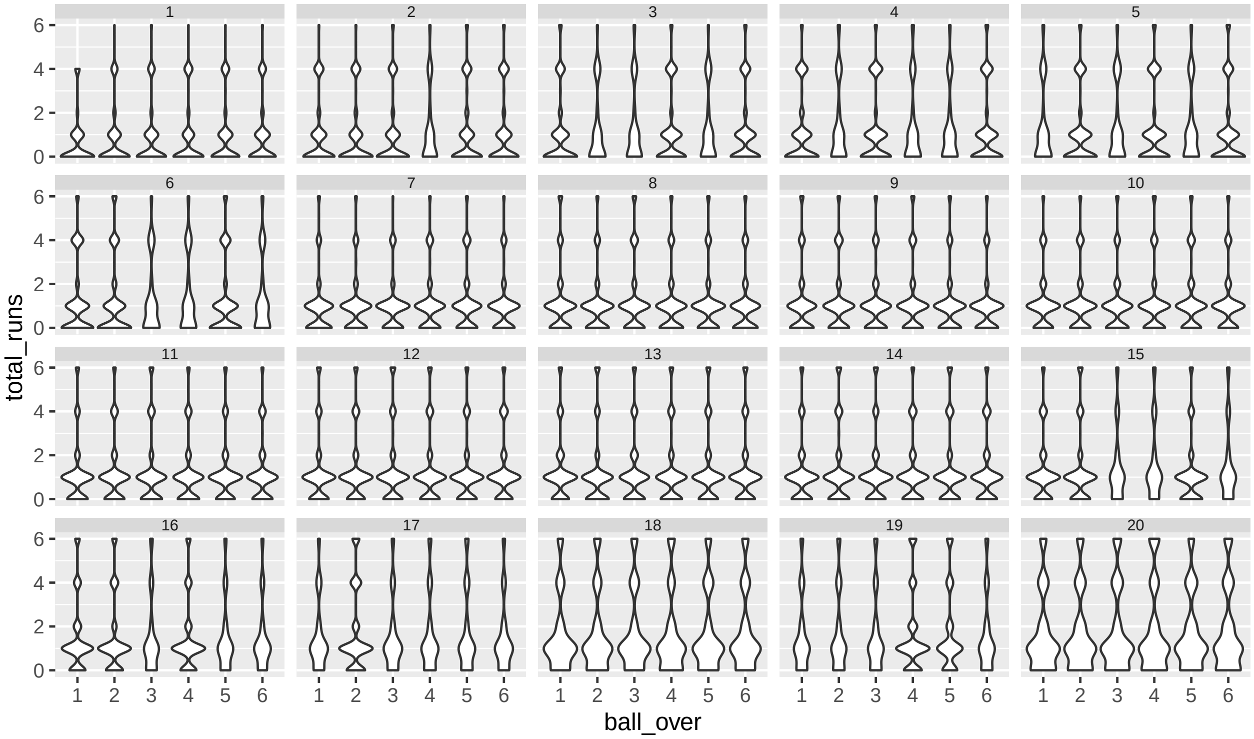
Suppose, we are interested to see how the distribution of scores vary from the start to the end of the game. Let us brainstorm some of the questions that might help us comprehend that.

a) What is the most common pattern for batting and bowling teams across balls, overs, innings and matches? Which are the teams which are typical and which are exceptions?

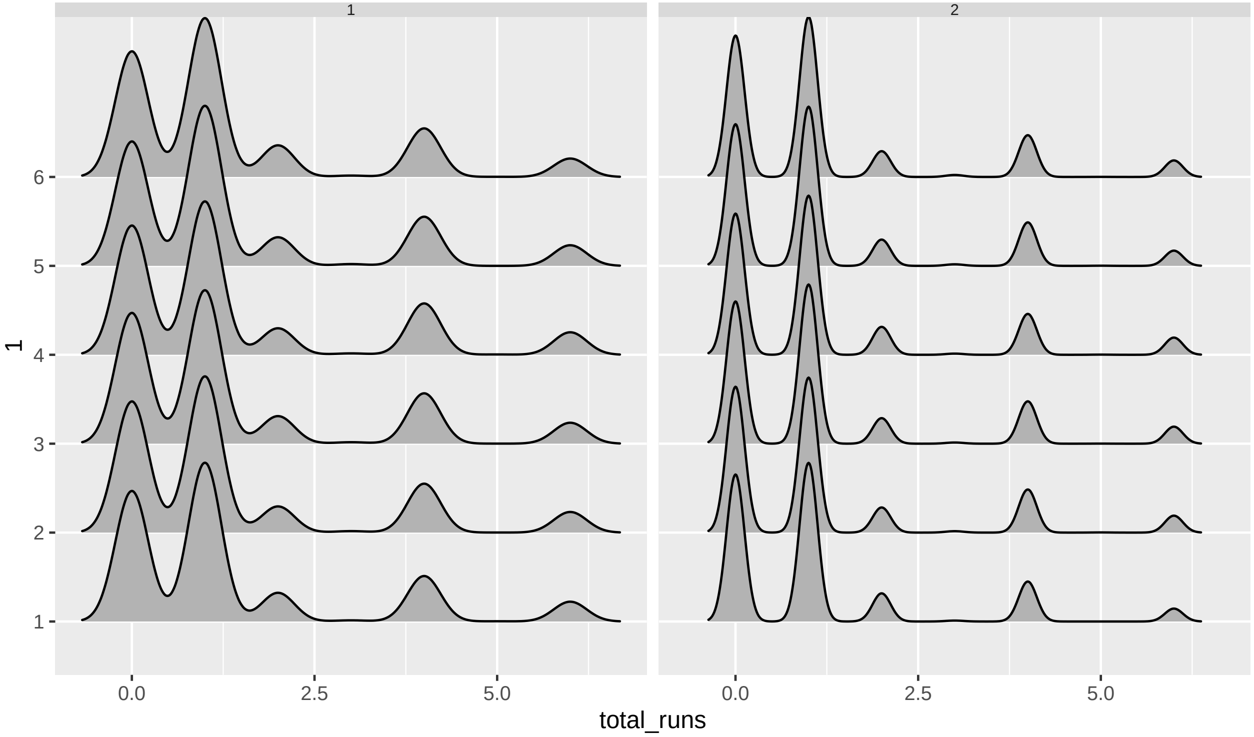
b) How the scores vary per each over each periodic granularities like ball of the over, ball of the innings, over of the inning, over of the match, inning of the match and others.

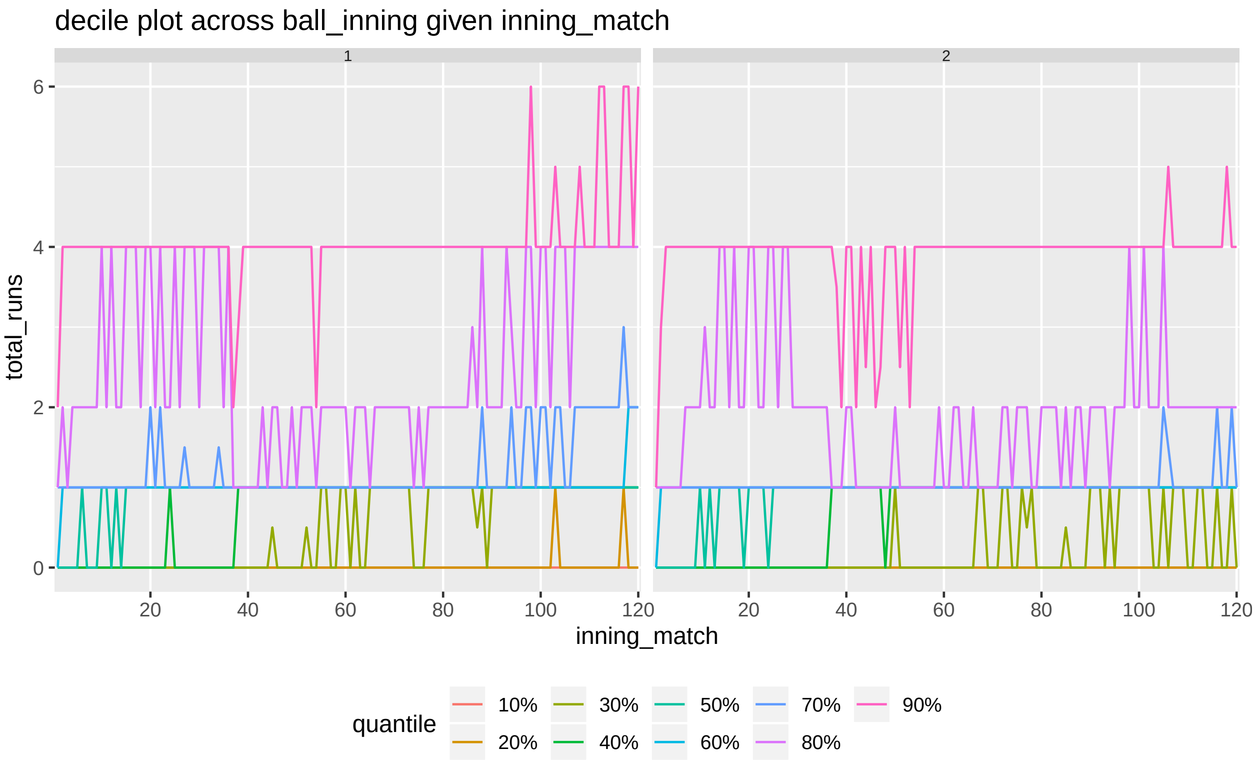
We will look at the ball by ball data for all batting teams.Since we want a periodic world, where each over consists of 6 balls and each match consists of two innings, we shall filter out the matches or overs for which that is not true.

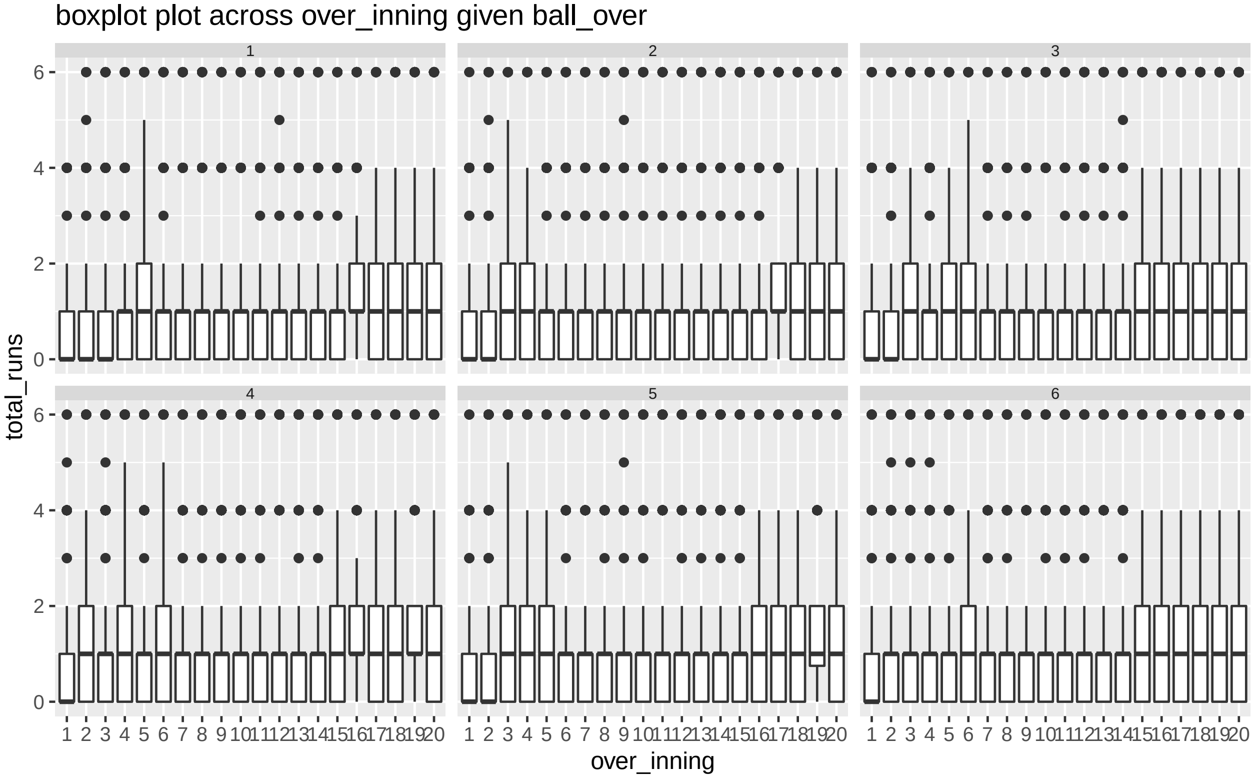
violin plot across ball_over given over_inning



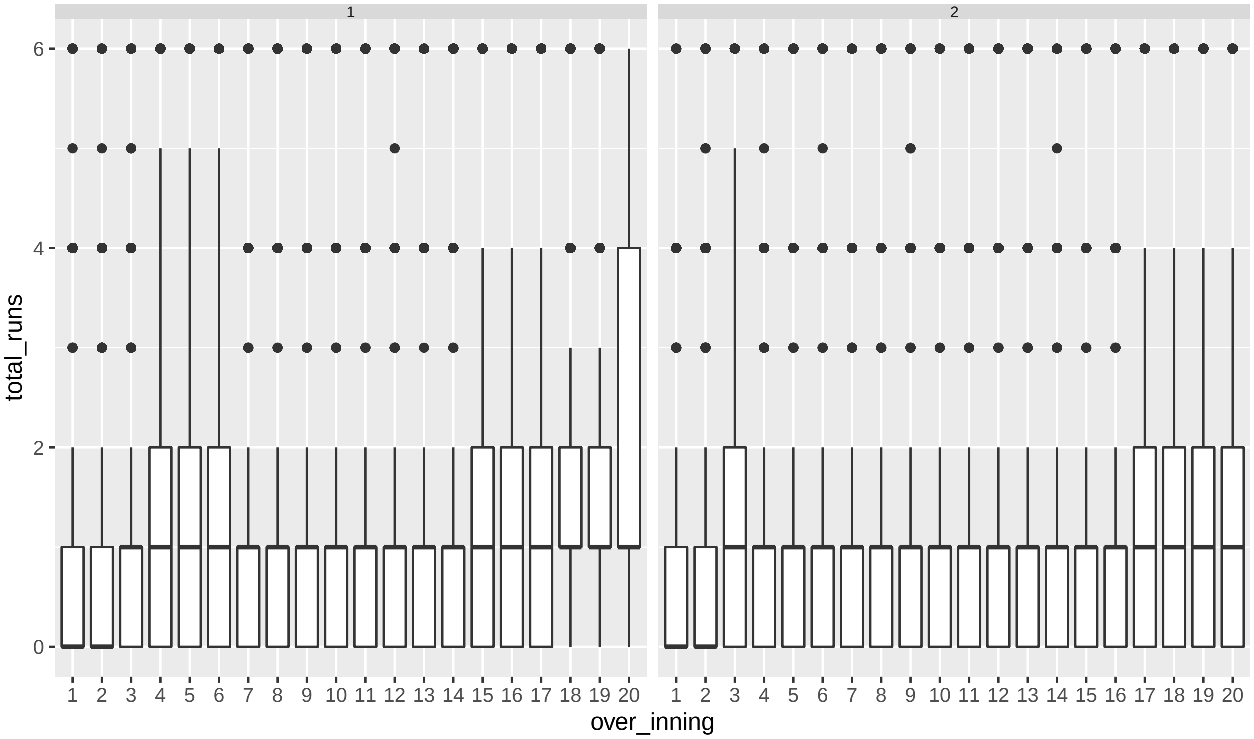
ridge plot across ball_over given inning_match



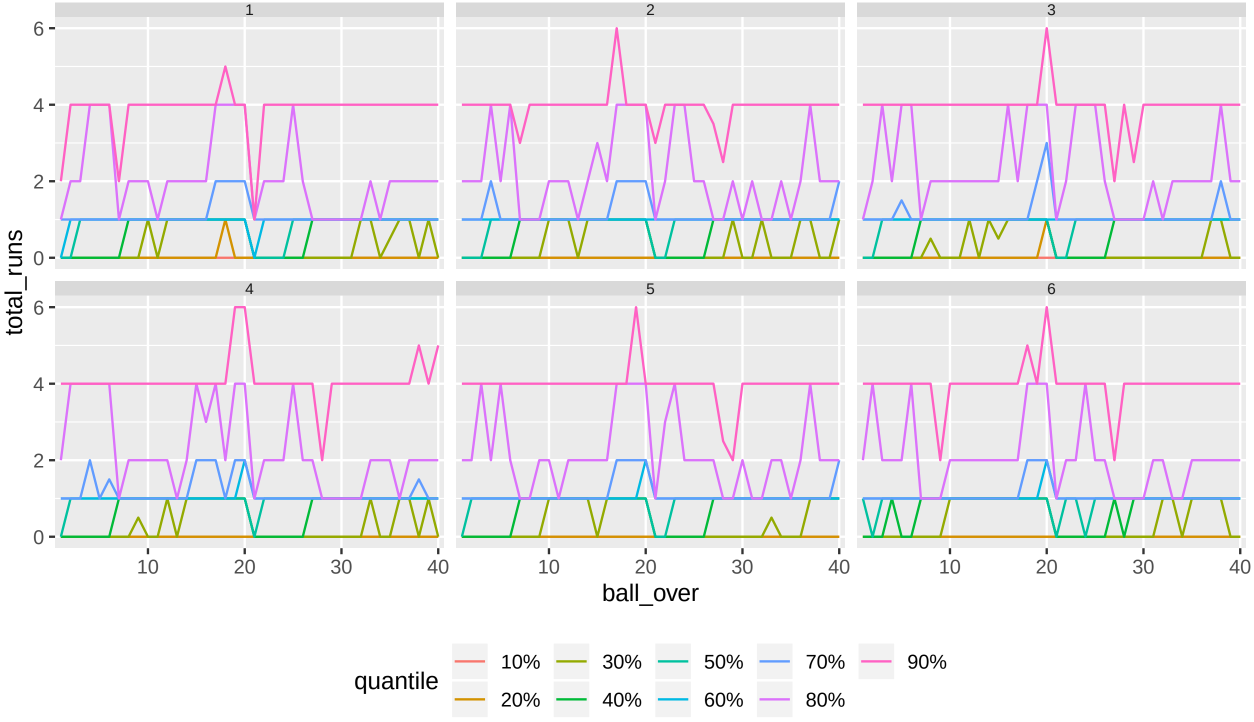

decile plot across ball_inning given inning_match
1
2
total_runs
0
2
4
6
20
40
60
80
100
120
inning_match
quantile
10%
20%
30%
40%
50%
60%
70%
80%
90%


boxplot plot across over_inning given ball_over
1
2
3
4
5
6
total_runs
0
2
4
6
1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20
over_inning

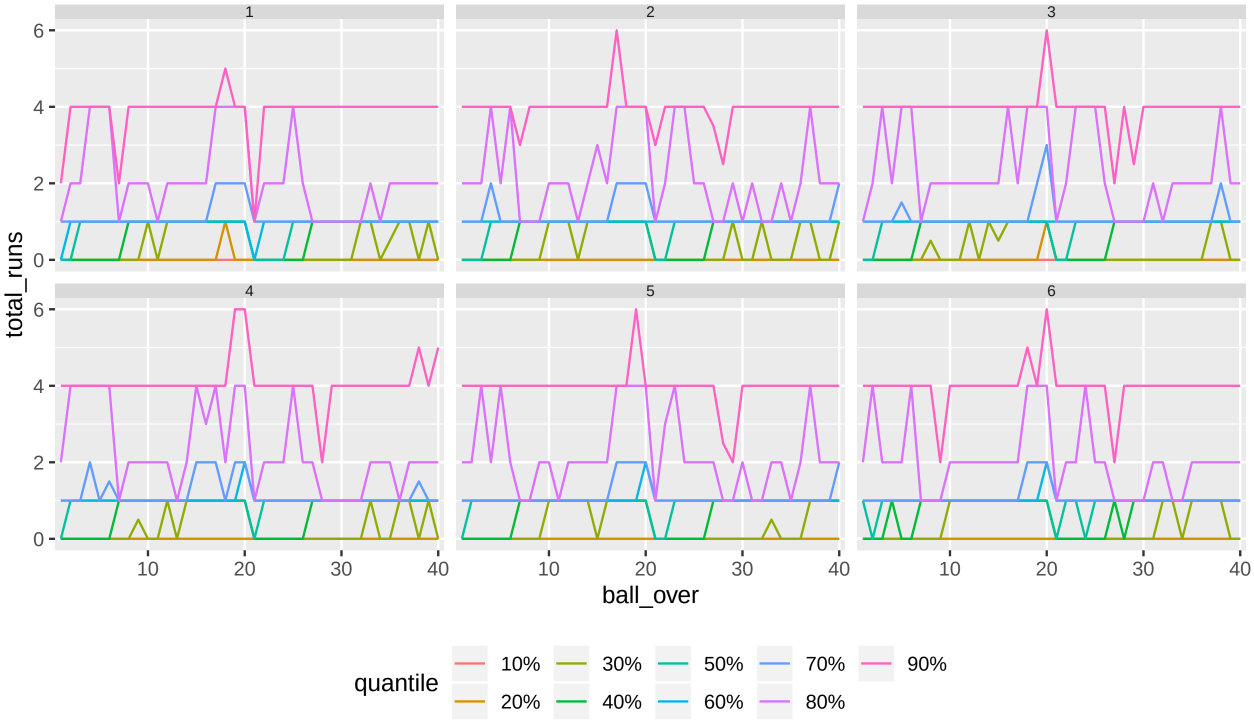
boxplot plot across over_inning given inning_match



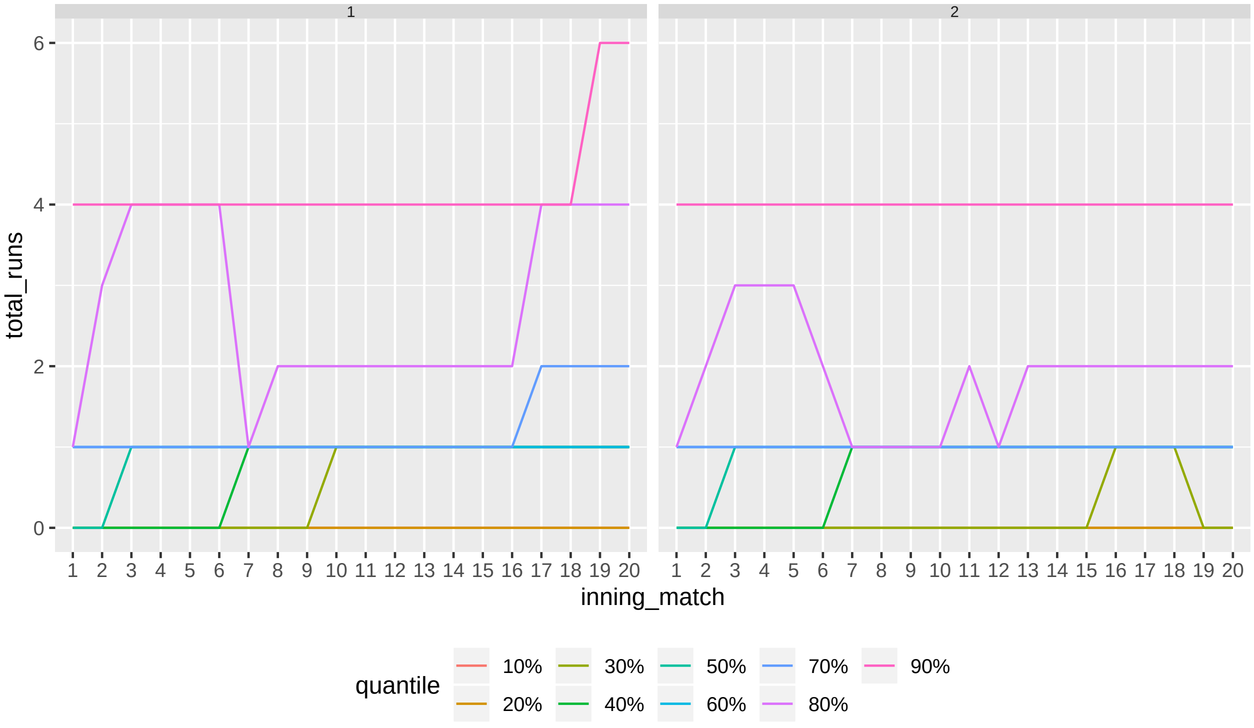
decile plot across over_match given ball_over



decile plot across over_match given ball_over



quantile plot across over_inning given inning_match



The scores/ball increases from 1st to 120th ball in the 90th percentile. In the last 2 overs, players are more vulnerable to get more scores, which is evident from the fact that in 90 percent of cases, their scores vary between 4 and 6. Till around 50th ball, players play safe and may get no runs per ball in 25% of the times, after which they more likely to get one score per ball.

7 Discussion

Acknowledgements

8 Bibliography

Aigner, Wolfgang, Silvia Miksch, Heidrun Schumann, and Christian Tominski. 2011. *Visualization of Time-Oriented Data.* Springer Science & Business Media.

Benjamini, Yoav. 1988. “Opening the Box of a Boxplot.” *Am. Stat.* 42 (4). Taylor & Francis: 257–62.

Bettini, Claudio, Curtis E Dyreson, William S Evans, Richard T Snodgrass, and X Sean Wang. 1998. “A Glossary of Time Granularity Concepts.” In *Temporal Databases: Research and Practice*, edited by Opher Etzion, Sushil Jajodia, and Suryanarayana Sripada, 406–13. Berlin, Heidelberg: Springer Berlin Heidelberg.

Department of the Environment and Energy. 2018. *Smart-Grid Smart-City Customer Trial Data.* Australian Government, Department of the Environment; Energy: Department of the Environment; Energy, Australia. https://data.gov.au/dataset/4e21dea3-9b87-4610-94c7-15a8a77907ef.

Hintze, Jerry L, and Ray D Nelson. 1998. “Violin Plots: A Box Plot-Density Trace Synergism.” *Am. Stat.* 52 (2). [American Statistical Association, Taylor & Francis, Ltd.]: 181–84.

Hofmann, Heike, Hadley Wickham, and Karen Kafadar. 2017. “Letter-Value Plots: Boxplots for Large Data.” *J. Comput. Graph. Stat.* 26 (3). Taylor & Francis: 469–77.

Hyndman, Rob J. 1996. “Computing and Graphing Highest Density Regions.” *Am. Stat.* 50 (2). [American Statistical Association, Taylor & Francis, Ltd.]: 120–26.

Mcgill, Robert, John W Tukey, and Wayne A Larsen. 1978. “Variations of Box Plots.” *Am. Stat.* 32 (1). Taylor & Francis: 12–16.

Potter, K, J Kniss, R Riesenfeld, and C R Johnson. 2010. “Visualizing Summary Statistics and Uncertainty.” *Comput. Graph. Forum* 29 (3): 823–32.

Tukey, John W. 1977. *Exploratory Data Analysis.* Vol. 2. Reading, Mass.

Wang, Earo, Dianne Cook, and Rob J Hyndman. 2019. “A New Tidy Data Structure to Support Exploration and Modeling of Temporal Data,” January.

Wickham, Hadley. 2016. *Ggplot2: Elegant Graphics for Data Analysis.* Springer-Verlag New York. http://ggplot2.org.

Wickham, Hadley, and Lisa Stryjewski. n.d. “40 Years of Boxplots.”