

# A Model for Energy-Saving in an IoT Smarthome accounting for End-User Convenience

*Alistair Francis Bowman Grevis-James*

*November 2019*

## **Abstract**

The preface pretty much says it all.  
Second paragraph of abstract starts here.

# Contents

<b>1 Introduction</b>	<b>3</b>
1.1 Background . . . . .	3
1.1.1 Humankind, Technology & Development . . . . .	3
1.1.2 The Rise and Rise of the World Wide Web . . . . .	3
1.1.3 Forging a New Technological Paradigm . . . . .	4
1.1.4 The Bigger Picture . . . . .	4
1.1.5 Optimising Global Energy Usage . . . . .	6
1.1.6 The IoT Smart Home & Service Oriented Computing . . . . .	7
1.2 Aim . . . . .	7
1.2.1 Research Questions . . . . .	7
1.2.2 Research Contributions . . . . .	7
<b>2 Preliminaries</b>	<b>8</b>
2.1 Related Work . . . . .	8
2.1.1 Ubiquity of the World Wide Web and Implications for Energy Consumption . . . . .	8
2.1.2 Smart Metering . . . . .	8
2.1.3 Electricity Demand Response, Smart Electricity Grids and Smart Home Data . . . . .	8
2.1.4 Predictive Modelling and Forecasting . . . . .	12
2.1.5 Smart Home and Energy Efficiency . . . . .	14
2.1.6 IoT and End User Convenience . . . . .	14
2.1.7 Heading??? . . . . .	15
2.1.8 Summary . . . . .	17
2.2 The Data Set . . . . .	17
<b>3 Data Preprocessing &amp; Visualisation</b>	<b>18</b>
3.1 Importing & Preprocessing the Activities Meta Data . . . . .	18
3.2 Importing & Preprocessing the Sensor Meta Data . . . . .	18
3.3 Importing & Preprocessing the Activities Data . . . . .	19
3.4 Importing, Visualizing and Preprocessing the SubActivities Data . . . . .	21
3.4.1 Sub Activity Preprocessing - Duplicate sub activities . . . . .	23
3.4.2 SubActivity Visualisation and Outlier Inspection . . . . .	23
3.4.3 SubActivity Cleansing - Outlier Removal . . . . .	26
<b>4 Jewelry + foyer closet + cereal + containers + lamp</b>	<b>26</b>
<b>References</b>	<b>39</b>

# 1 Introduction

## 1.1 Background

### 1.1.1 Humankind, Technology & Development

Since the inception of the first home computers in the late 1970s [1], modern society has become dependent on, and - indeed - inexorably bound to digital technology. The rapid and widespread adoption of computational technology has led to the fastest rate of societal and economic development our species has ever experienced, as exemplified in Figure 1 and Figure 2, below. One of the most salient manifestations of progress has been the widespread availability and adoption of Information and Communications Technology (**ICT**), including the rise of the global network of networks known as the Internet.

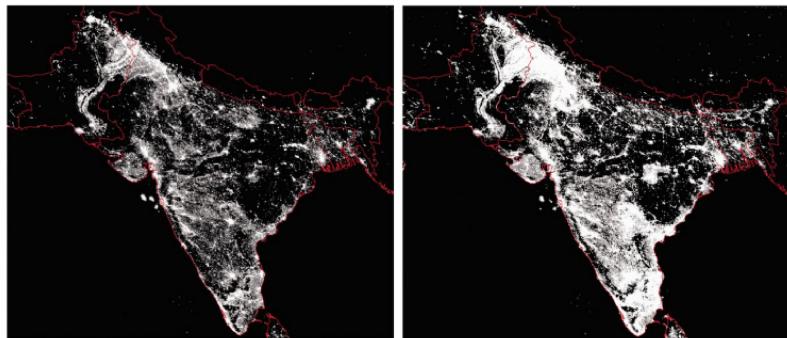


Figure 1: Satellite images of South Asia by night. Left (South Asia in 1994) Right (South Asia in 2010). Images are taken from Maxim Pinkovskiy and Xavier Sala-i-Martin (2016) - Lights, Camera ... Income! Illuminating the National Accounts-Household Surveys Debate. The Quarterly Journal of Economics.

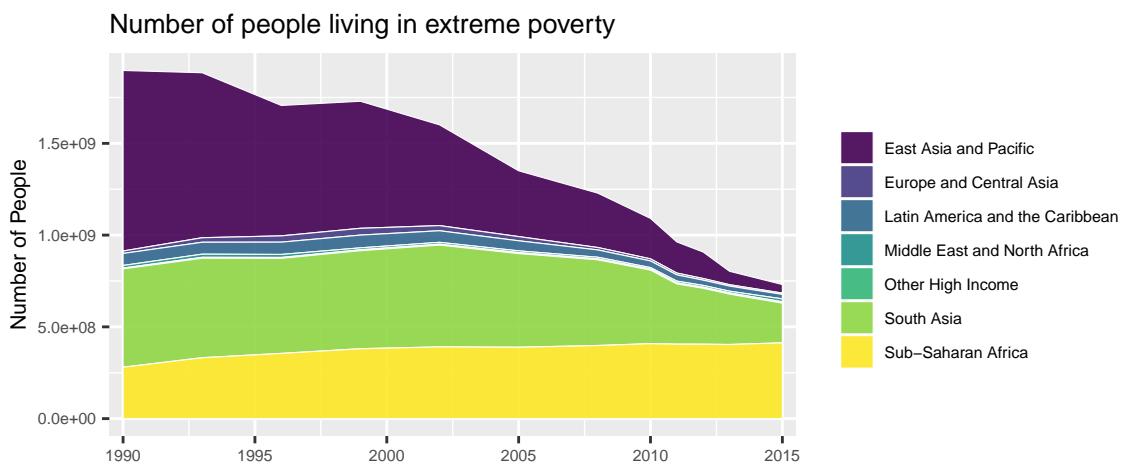


Figure 2: The number of people living in extreme poverty between the years of 1990 till 2015, segmented by region. Source: The World Bank

### 1.1.2 The Rise and Rise of the World Wide Web

According to the International Telecommunication Unions (**ITU**) 2015 figures, Internet penetration has grown from just over 400 million users (6 per cent of global population) in 2000 to 3.2 billion users in 2015 (43 per cent of global population). This includes around 2 billion users from developing countries [2]. ICTs bring a broad range of benefits and are recognised as a key to eradicating poverty and unemployment. They enable and facilitate the building of a people-centred, inclusive and development-oriented Information Society, where everyone can create, access, utilise and share information and knowledge. This

enables individuals, communities and peoples to achieve their full potential in promoting their sustainable development and improving their quality of life [3]. In addition to a rapidly growing internet user-base both in the developing and developed world, the nature of internet usage has fundamentally changed. Once the purview of academics, engineers and computer scientists sending tiny packets of information back and forth, there are now some 2.5 quintillion bytes of data created each day by all manner of users, industries and sensors, to name a few [4].

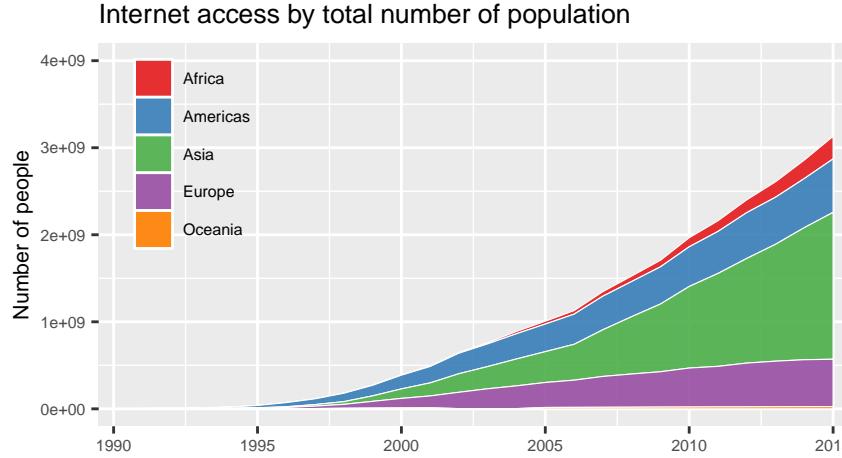


Figure 3: Internet access by total number of population, Source: The World Bank, World Development Indicators

### 1.1.3 Forging a New Technological Paradigm

Such rapid and widespread internet adoption has created a seemingly insatiable demand for exponentially greater computational power and digital storage capacity. This has led to a new and ubiquitous technological paradigm: Cloud Computing. Cloud Computing is succinctly defined as; the practice of using a network of remote servers hosted on the Internet to store, manage, and process data, rather than a local server or a personal computer [5].

As with the initial rise and widespread implementation of the Internet, Cloud Computing itself acts as a facilitator for new technologies. One such example being the Internet of Things (**IoT**) paradigm. The IoT can be surmised as the extension of the Internet and the Web into the physical realm, by means of the widespread deployment of spatially distributed devices with embedded identification, sensing and/or actuation capabilities, allowing objects to be sensed or controlled remotely through the internet [6], [7], [8]. IoT thus represents a convergence of real-world objects and digital objects into a unified cyber-physical system.

### 1.1.4 The Bigger Picture

Considering the bigger picture of societal benefit, as seen in figure 2, from 1990 through 2015 the number of people living in extreme poverty has dropped by more than half. As evidenced by Figure 3, over the same time period, the percentage of people with access to the internet has moved from around 1 per cent to an average of around 50 per cent (this trend is now moving exponentially as a function of time). And, figure 4 shows that over the last 100 years, the human population has experienced unprecedented growth from 1 billion individuals to in excess of 7 billion. Mankind have thus simultaneously increased our population, increased of technological development and decreased poverty (to name but a few, ‘key performance indicators’).

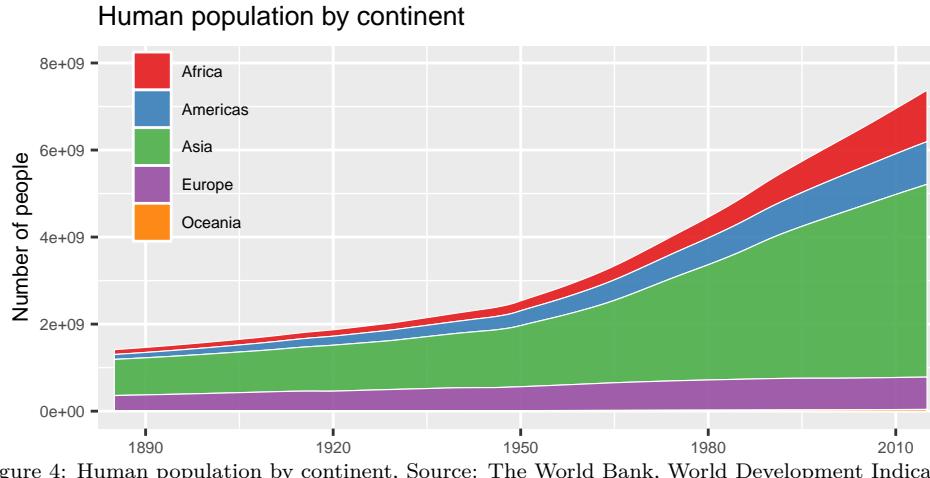


Figure 4: Human population by continent, Source: The World Bank, World Development Indicators

The success of modern human society is not without consequence. All of the benefits our society has enjoyed from the development, production and deployment of technology, has required vast amounts of energy. This energy has, since the industrial revolution, primarily been derived from the burning of fossil fuels, as illustrated by Figure 5.

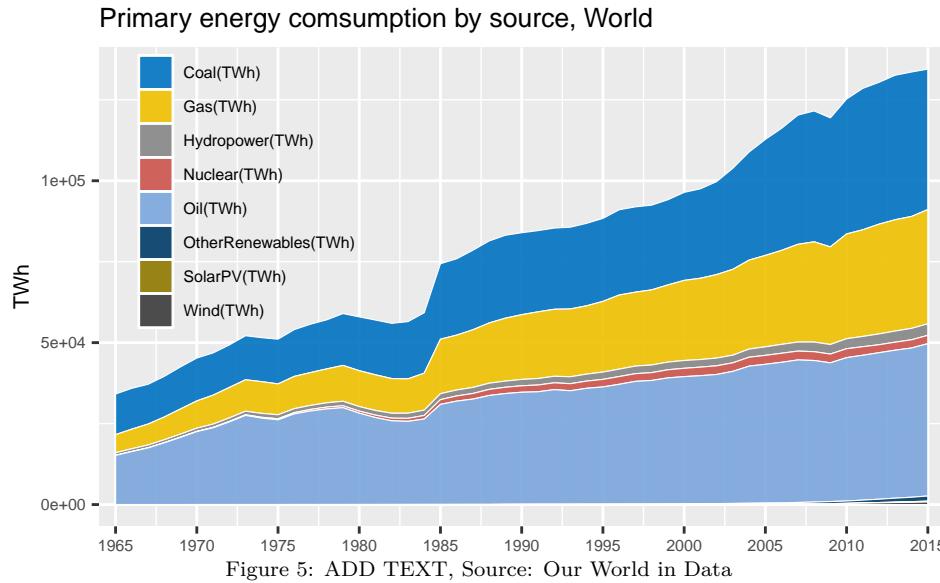


Figure 5: ADD TEXT, Source: Our World in Data

The International Panel on Climate Change (IPCC) finds that Human activities are estimated to have caused approximately 1.0°C of global warming above pre-industrial levels, with a likely range of 0.8°C to 1.2°C. Global warming is likely to reach 1.5°C between 2030 and 2052 if it continues to increase at the current rate [9].

Climate change poses an existential threat to modern human civilisation, with warming of between 1.5°C and 2°C predicted to cause increases in mean temperature in most land and ocean regions, hot extremes in most inhabited regions and heavy precipitation changes in some regions. Additionally, increases in ocean temperature as well as associated increases in ocean acidity and decreases in ocean oxygen levels are projected to reduce risks to marine biodiversity, fisheries, and ecosystems, and their functions and services to humans. Taken together, these effects will lead to risks of the health, livelihoods, food security, water supply, human security, and economic growth of mankind [9].

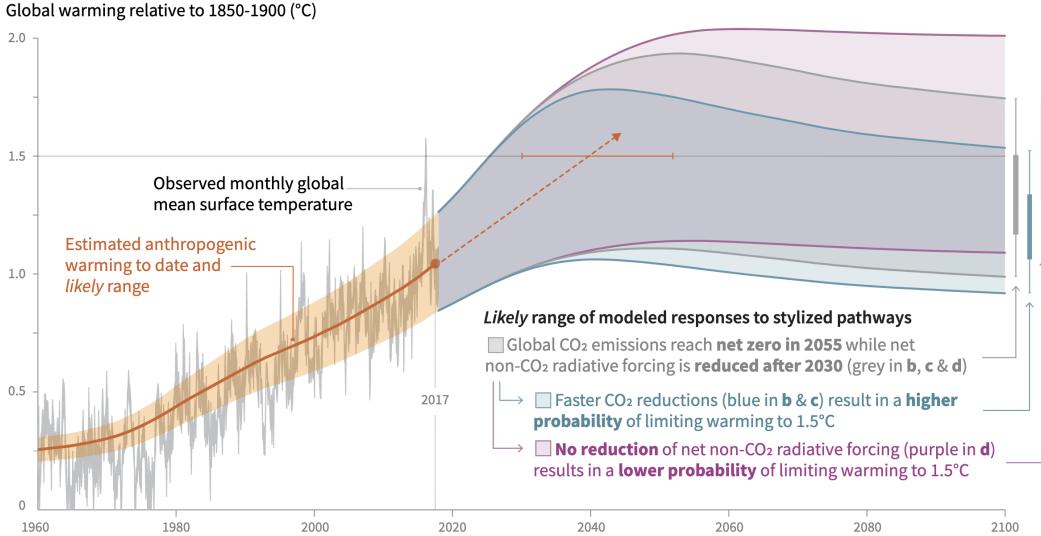


Figure 6: Atmospheric Changes with respect to Carbon Emissions and Global Warming - Observer and Projected. Source: Intergovernmental Panel on Climate Change, 2018

It is therefore imperative moving forward as a species that all steps are be taken to mitigate the emission of greenhouse gases and abate the advance of anthropomorphic climate change. The scale of the challenge is such that technology itself will prove critical in effectively combating this existential threat to civilisation. When considering energy consumption, the industrial sector (including the non-combusted use of fuels) currently consumes around half of all global energy and feedstock fuels, with residential and commercial buildings (29%) and transport (21%) accounting for the remainder [10]

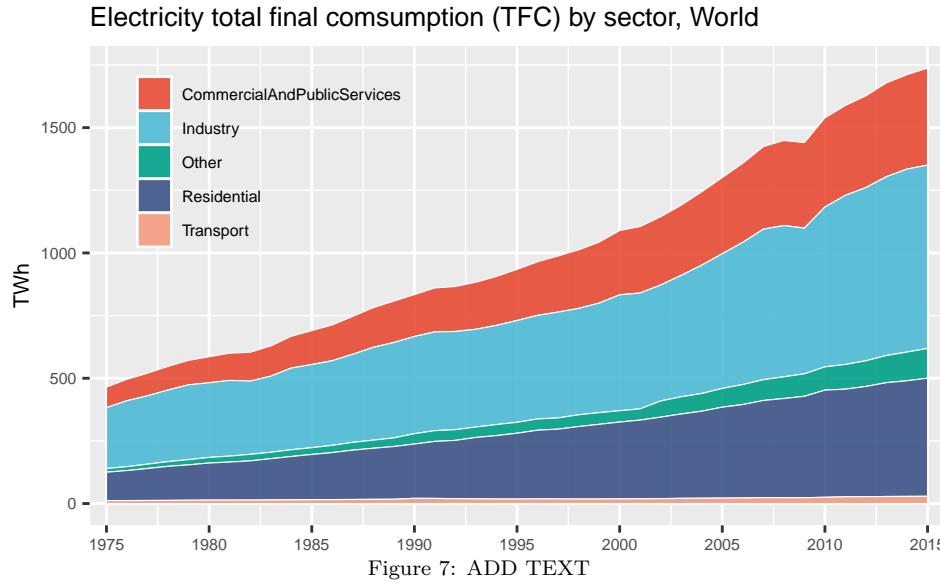


Figure 7: ADD TEXT

### 1.1.5 Optimising Global Energy Usage

Thus far, our staggering global achievements, including lifting hundreds of millions out of poverty, rapid global technological deployment and strong population growth has been inexorably linked to increased energy consumption. This in turn has led to perturbations in atmospheric chemistry, in the form of anthropogenic climate change (amongst many other environmental challenges), which fundamentally threatens our global achievements. It therefore stands that the key to continued human prosperity is to de-couple

growth in energy demand from economic growth. In this work we will explore the possibility of reducing energy consumption via the optimization of services to end users in an IoT Smart Home.

### **1.1.6 The IoT Smart Home & Service Oriented Computing**

As mentioned above, residential and commercial buildings account for 29% of energy demand globally [11]. In this work we propose that an avenue for reduced energy consumption is the optimization of existing services and utilities to end users in an IoT Smart Home. In this proposed smart home, the daily activities of the end user can be performed with the support of a personalised artificial intelligence (AI) system, such that the timing and manner of energy intensive activities is optimised to reduce overall energy consumption, whilst still considering the level of convenience afforded to the end user. This work uses sensor data analogous to what would be expected to be produced from a sensor rich IoT smart home and considers the interplay between end user convenience; predictive analytics; predictive service offering; energy consumption; demand response and smart electricity grids.

## **1.2 Aim**

### **1.2.1 Research Questions**

The aim of this work is to use sensor data from X to consider Y in the context of Z.

### **1.2.2 Research Contributions**

The aim of this work is to use

## 2 Preliminaries

Significant growth in digital interconnectivity over the last 20 years has given the internet a pivotal role as an essential element of economic growth [12]. Cloud computing and the IoT paradigm has enabled the association of previously disparate fields into a larger coherent framework. Namely, smart home appliances, demand-response, energy consumption, predictive analytics predictive service offering and end user convenience. This work proposes a framework for the association of sensor data into a model which can do X Y Z. **Mention internet in general?**

### 2.1 Related Work

#### 2.1.1 Ubiquity of the World Wide Web and Implications for Energy Consumption

The International Telecommunication Union estimated about 3.2 billion people, or almost half of the world's population, would be online by the end of the 2015 [3]. The impact of ICTs (i.e., internet usage and mobile cellular subscriptions), globalization, electricity consumption, financial development, and economic growth on environmental quality has been examined [12]. By using 1994–2014 panel data of BRICS (Brazil, Russia, India, China & South Africa) economies, empirical results demonstrate that rise in both internet usage and mobile cellular subscription (ICTs) likely mitigates CO<sub>2</sub> emissions in BRICS economies.

**BRIDGE** – Greater internet availability / connectivity facilitates the deployment of smart electricity grids

#### 2.1.2 Smart Metering

A Smart meter is an electronic device that records consumption of electric energy and communicates the information to the electricity supplier for monitoring and billing. In the United States (for example) smart meters are a significant part of the larger Smart Grid infrastructure, and as far back as 2012, had been installed in over 25 million U.S. homes [13]. Smart Meters transmit information about consumer electricity use to utility companies at vastly shorter time intervals than before and this information helps utility companies to coordinate power supply and demand, detect outages, implement time-of-use and dynamic pricing, and in other ways improve system efficiency and reliability. Additionally, these data are also becoming increasingly available, and sought-after, by end-users themselves. Indeed, the main purpose of provisioning smart metering data to end users is to encourage the use of less electricity, by better informing users of their consumption patterns (Jui-Sheng Chou, 2019). In the aggregate, these savings can significantly reduce national energy use and curb energy emissions while addressing pressing geopolitical and environmental concerns related to energy security and sustainability (Graab, 2011).

Since the widespread deployment of smart meter technology, there has been huge interest in the capability of these technologies with respect to the technical capacity of utility companies to manage demand (through demand response programs), incorporate renewable sources of electricity into the system, and increase the overall efficiency and reliability of the system (Horne et al., 2015). **It is worth noting, however, that the scenario described above relies on end user intervention.**

#### 2.1.3 Electricity Demand Response, Smart Electricity Grids and Smart Home Data

Predicting and influencing residential energy use has been the subject to extensive study (**REF**). Literature indicates that factors such as occupant behaviour and socio-economic status are important [15 from

**Cetin**. Nielsen attributed 36% of variation in energy consumption of homes to lifestyle and occupant behaviour, and 64% to socio-economic influences. This is exemplified by the work of De et al, who show that in developing nations, cooking consumes up to 90% of the overall residential energy consumption and is mainly based on non-renewable energy (International Energy Agency, 2006; De et al., 2013). Other factors such as climate zone, number of occupants, income level, age of home, and size of home have also been correlated with home energy use (K.S. Cetin, 2014). The has been much work both on using smart meter data to evaluate consumer behaviour, and the interplay between smart appliances and shifting households' electricity demand.

Kavousian et al identify the need for developing an analytical method that can leverage 15-minute or 30-minute interval energy consumption data produced via smart metering in order to improve the effectiveness of energy efficiency programs. They note that utilities spend millions of dollars annually to improve appliance energy efficiency. By way of example, in 2013 utilities in California US spent \$80 M USD on appliance and plug load efficiency programs, the highest expenditure among all utility energy efficiency programs (**CPUC, 2013 FIND REF**). A need for analytical methodologies that can process smart meter data and allow energy reduction savings to be identified is demonstrated. Using a smart meter dataset of 4231 households, containing information on electricity consumption at 30-minute intervals, with aggregated local weather data, the authors use smart meter data to rank residential appliance efficiency. Various control methodologies are embedded into the analytical process, taking into account building type, building size (e.g., apartment, detached), household type (e.g., de facto, single, dependents), respondent age and heater type (e.g., electric or gas). The data set included 120+ household variables, many of which were highly correlated, model selection techniques were used to successfully reduce dimensionality. A set of load profiles were compiled based on the results (where normalized load was plotted as a function of hour of day). It was determined that household behaviour and demographic can be used to generate positive and negative energy efficiency coefficients with respect to load profiles. This work has implications for energy grid planning – for example in high-density housing versus suburban housing.

Karjalainen considers the manner in which smart meter data is communicated to end users, with the principal purpose of the paper being to study what kind of electricity consumption feedback consumers understand and prefer (Karjalainen, 2011). It is noted that the most effective feedback tools for engaging households in reducing energy consumption are both computerized and interactive. In this work a series of options for feedback is gathered, for example consumption (kWh), power (W), cost (\$), environmental impact (e.g., kg CO<sub>2</sub>), total for household, disaggregation by appliance, real time, hour, day, and so on. Examples of feedback options are shown in Figure X, below.

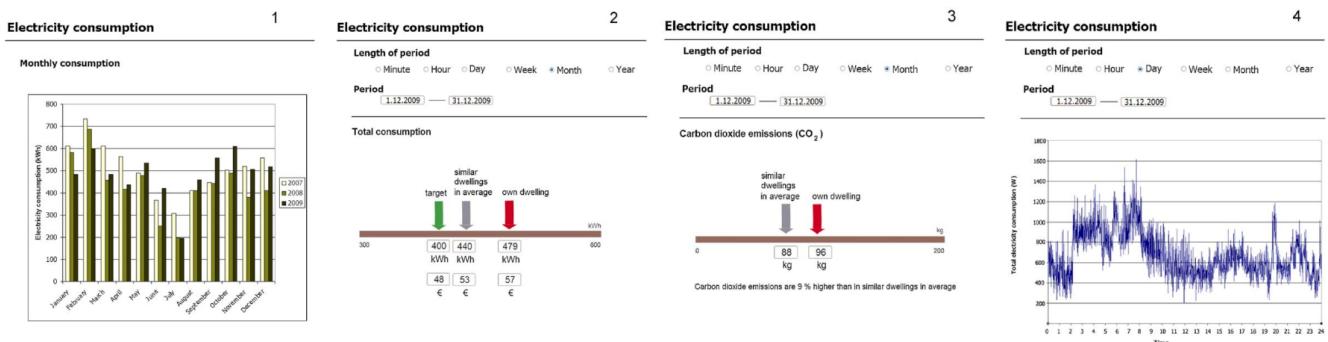


Figure 8: ADD TEXT

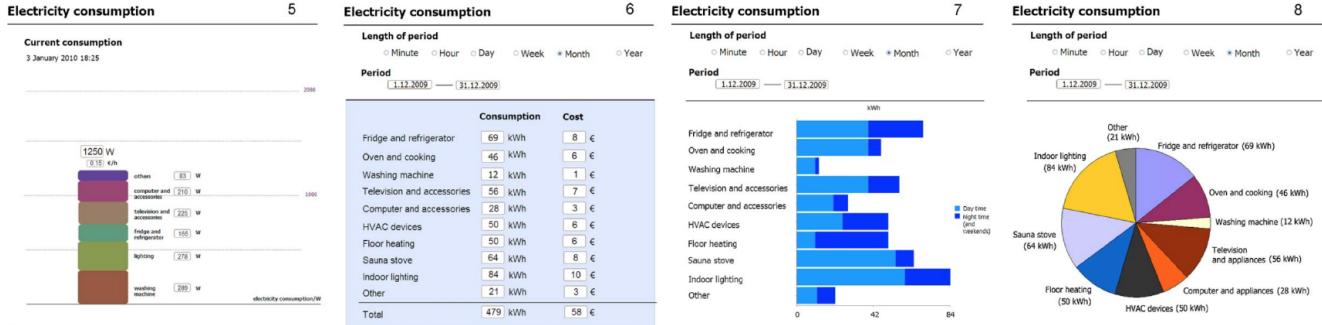


Figure 9: ADD TEXT

The results of qualitative participant interviews clearly showed that while some consumers are very interested in saving household electricity, other consumers show only a little interest (**NOTE they were asked about saving electricity, not about saving money or convenience**). When specifically asked to list what measures (if any) were typically taken at home to realize reductions in energy usage, some respondents listed numerous measures, while others just said they turn lights off in rooms that are empty. The author also found some participants were unaware of differences in stand-by versus active modes of operation for electrical appliances, resulting in practising inefficient energy saving measures in the home environment. When trialling feedback prototypes to participants, two main issues were encountered: (1) many people are not familiar with scientific units and do not understand the difference between W and kWh and (2) many people do not understand how carbon dioxide emissions are related to electricity consumption. It is perhaps surprising that the overall most popular prototype was ‘6’, below. Perhaps because unlike the other prototypes, this clearly (the most clearly) articulates cost. The concept of convenience was absent entirely from consideration.

In the domain of supply and demand economics, consumer behaviour and smart appliances Kobus et al investigated if households can shift their electricity demand to times when electricity is abundantly available (Charlotte B.A. Kobus, 2015). Using a household electrical monitoring system (EMS) coupled to a smart appliance (smart washing machine), photovoltaic cells and the electricity grid, they were able to show that households can shift 10–77% of the electricity demand of their washing machine. This longitudinal study was conducted via the participation of 50 Dutch households over a period of one year. By utilizing an EMS which shows appliance status, dynamic tariff information and current status of household electricity usage, participants are able to schedule smart washing machine use in such a way that cost is minimized.



Figure 10: ADD TEXT

Their work is in response to two major challenges to domestic electricity supply and demand. **REMOVED:** It is well observed that over electricity grids, demand shows a pattern over time, influenced by natural and social circumstances. This poses a great challenge when considering renewable energy sources like PV – people are generally not at home when local

**electricity is abundantly produced (for example the daylight hours of Monday till Friday)** The first being that the amount of distributed renewable electricity generation is increasing with time (for example, as more households install photovoltaic (PV) panels), and the second that electricity demand will continue to significantly increase moving into the foreseeable future. These developments pose great challenges to ‘traditional’ power systems, where supply follows demand entirely. Smart grids are proposed as a potential solution to facilitate the affordable introduction of cleaner electricity producing and consuming technologies.

Cetin et al consider electricity usage of appliance, as when aggregated, this accounts for approximately 30 per cent of electricity used in the residential building sector (K.S. Cetin, 2014). This, together with small appliances, home electronics and lighting, account for more than 2/3 of total residential electricity use. Cetin et al also highlight that influencing ‘time of use’ is becoming increasingly important to control the stress on today’s electrical grid infrastructure. The authors seek to determine when refrigerators, clothes washers, clothes dryers and dishwashers are predominantly used (and thus consume energy) and what causes variation in their use. Using disaggregated energy data from 40 homes over a period of 1-year, normalized load profiles for the four target appliances were generated, as a percentage of daily electricity load.

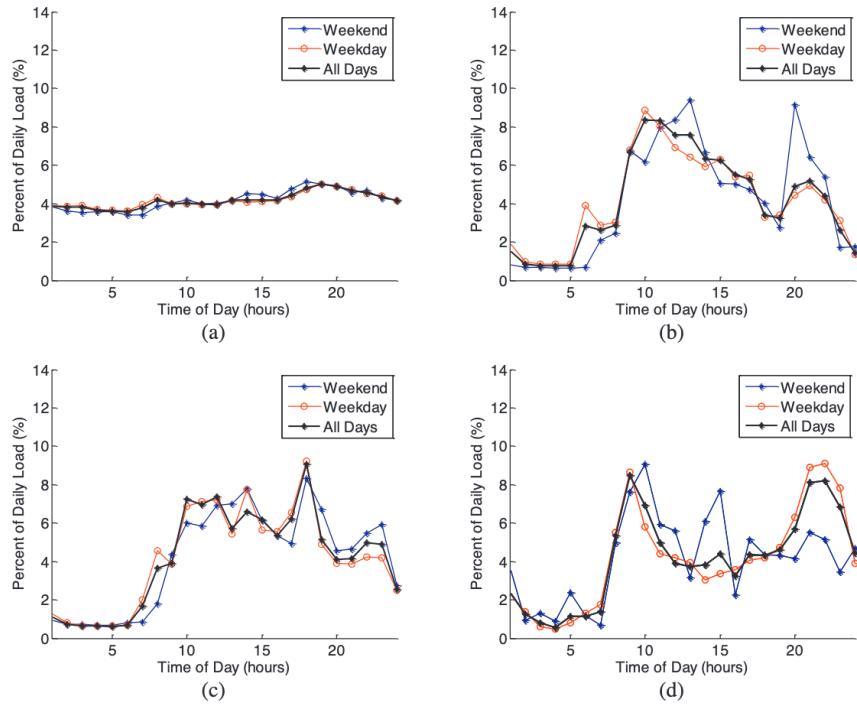


Figure 11: ADD TEXT

It was found that the refrigerator had the most consistent consumption profile across all homes surveyed. Influencing factors for the refrigerator were correlated to both indoor and outdoor temperature, however effect was found to be minimal. The clothes washer and dryer were found to have the greatest variation in normalised energy use by hour, with the greatest period of use from 9am until 2pm. The dishwasher had distinct peaks in load profile at 9am and 10pm. The authors found that user-dependent appliance use patterns vary more between homes and between days than automated appliances, weekday and weekend use patterns of appliances are similar, and that electricity use varies more between houses during peak use times of day than during low-use times. Murray et al further explore the topic of home appliance energy consumption using smart meter data, specifically pertaining to residential activities around food storage and preparation (D.M. Murray, 2018), with the aim of providing a model that can easily be applied to existing smart meter energy datasets. The authors use real-time energy consumption data

from microwaves and ovens, from which user behaviour / desired outcome can be implicitly associated (for example, oven usage on a weekday between the hours of 6pm and 8pm can be associated with the behaviour of preparing dinner). Consideration is given to the concept of the RisingEdge (when an appliance is switched on or moves to a high consuming state) and FallingEdge (when an appliance is switched off or moves to a low consuming state) of power consumption. Accurate energy consumption models for major cooking appliances are successfully constructed, further re-enforcing the value and widespread applicability of residential smart meter data.

In this work,

This supports \_\_\_\_ that smart meters represent a valuable source of data that can be used for a variety of purposes, beyond just real-time monitoring. For example the model constructed by Murray et al does not rely on difficult to obtain parameters such as food type, temperature and weight. it has been shown that ...

- Authors are able to use smart meter data to build accurate energy consumption models of major cooking appliances
- Smart =
- Using consumption data from real homes, the authors are able to create scalable mathematical models to account for energy consumption associated with food preparation
- Indeed, from an energy-efficiency perspective, targeted energy feedback can be provided on consumers appliance usage habits and activities (Stankovic et al., 2016) leading to informed energy savings programmes and retrofit strategies for replacing appliances.
- And if this variation can be influenced??

*Demand Response allows for the management of demand side resources in real-time; i.e. shifting electricity demand according to fluctuating supply. When integrated into electricity markets, Demand Response can be used for load shifting and as a replacement for both control reserve and balancing energy. These three usage scenarios are compared based on historic German data from 2011 to determine that load shifting provides the highest benefit: its annual financial savings accumulate to €3.110M for both households and the service sector. This equals to relative savings of 2.83% compared to a scenario without load shifting.*

[???] discuss electricity demand response - allowing the management of demand side resources in real-time. **Based on Darby and McKenna (REF), we define demand response and a household action (automated, manual, or both) due to which electricity use is shifted in time in response to a price signal of other stimuli.** This can result in more efficient usage of the available sustainable electricity, like self-consumption of on-site PV electricity (REF) and peak demand reductions (REF) - all from ???{kobus2015}

#### 2.1.4 Predictive Modelling and Forecasting

In the domestic predictive energy consumption space Basu et al consider home automation systems linked via a communication network to enable interaction, data collation and control of appliances remotely by end users (Kaustav Basu, 2012). The potential competing priorities of energy savings versus comfort optimization for home occupants is discussed. The objective of this work is to propose a learning system that is able to help the home automation system compute an energetic plan that is also satisfactory to user requests. Taking into account correlation between appliances, a time-series based multi-label classifier is used to predict appliance usage up to one hour into the future.

The attribute construction technique of Knowledge Extraction applied. This process aims to extract novel attributes from underlying substructures in the training instances in the form of sub-events, for

example periodicity in data. This Knowledge Extraction process is similar to the implicit associations used by Murray et al, in the analysis of energy consumption for food preparation (D.M. Murray, 2018). The substructures are then fed to a propositional learner. The proposed model is trained in an iterative manner and attempts to take into account all the possible information based on consumption data, time of the event and meteorological information. Time is specifically model time as a periodic variable, segmenting on hour of day and day of week, noting that this takes into account the periodic nature of human behaviour. Using BR1, LP, CC1, CC2 and MLk machine learning algorithms, the precision, recall and accuracy for a variety of electricity-consuming appliances is determined (where each appliance constitutes a target variable over a set of iterations). In the evaluation phase they find that user behaviour toward an appliance is highly variable and the predictability of an appliance is dependent on the regularity of usage patterns of the inhabitants. It is specifically noted that *it is therefore very difficult for now to propose a generic methodology of appliance prediction for private houses.*

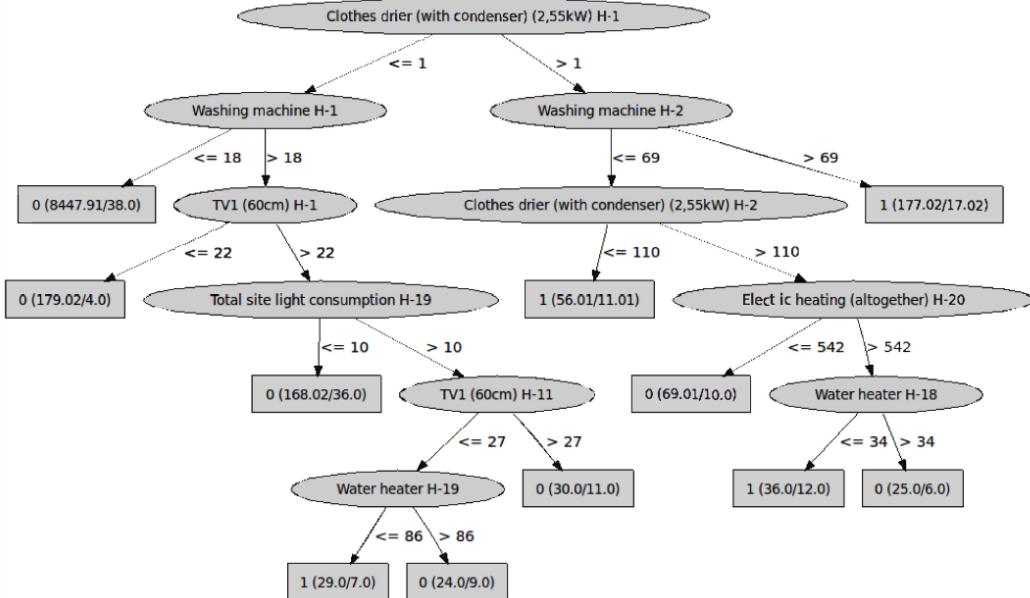


Figure 12: ADD TEXT

Chou et al consider global energy consumption in the residential housing sector in the context of domestic energy information system (EIS) and smart meter system (smart grids) technologies (Jui-Sheng Chou, 2019). They note two influential factors on overall residential energy consumption, the first being the type and number of electrical appliances and the second being the usage of these appliances by occupants. It is proposed that a major challenge for people who are willing to save energy at home is a lack of information about their energy consumption. To test this hypothesis, the authors develop a web-based energy information management system for the power consumption of home appliances that monitors the energy load of a home, analyses its energy consumption based on machine learning, and then sends information to various stakeholders. Interaction with end-users in the home is achieved via energy dashboards and emails. The authors propose that end-users of this system can use forecast information and anomalous data to enhance the efficiency of energy usage in their buildings especially during peak times by adjusting the operating schedule of their appliances and electrical equipment. **In order to achieve the desired model (???)**

During this study, an EIS with 5 main components was installed in an experimental building. The parts were as follows; (1) the internal communication network, (2) the data management infrastructure, (3) the automated prediction system, (4) the web-based system and dashboard and (5) the early warning system. Data from both the smart meter and from sensors was used to compile the model, including

timestamps (YYYY-MM-DD hh:mm:ss), outdoor temperature ( $^{\circ}\text{C}$ ), total building energy consumption (kWh), aggregated energy consumption data (e.g., second floor lighting) and individual appliances. The backend of the application was notably cloud-hosted.

- Improve consumer satisfaction by providing real-time services that enable end-users to monitor the energy consumption easily.
- Energy Information System - personalised terminal in domestic home which provides real-time or daily updating of hourly energy consumption data allows users to address building performance issues that are otherwise difficult to identify.
- The data stream from the smart meters arrives at 15 min intervals, generating 96 data points daily for each smart meter.
- The hardware of the DBMS consists of the communication system, the smart meter system, the environmental sensor system, and the server system, and the software programs that are used to process the data written in MATLAB. A DBMS stores real-time data that are retrieved from the smart grid metering infrastructure, including real-time energy consumption data, information about appliances, data from temperature and humidity sensors, and analytical results, including predicted electricity consumption, and data on electricity-saving alternatives. Moreover, the DBMS stores electrical parameters such as voltage, current, power, frequency, and power factor.
- The proposed prediction model (i.e., SARIMA-MetaFA-LSSVR) consists of stage 1, involving univariate linear technique, and stage 2, involving optimized multivariate nonlinear technique to fit the energy consumption data. SARIMA is adopted to handle the linear component, whereas MetaFA-LSSVR is used to model the nonlinear part of energy consumption data. In particular, the MetaFA is employed for automatically tuning LSSVR hyper parameters, thus improving the overall prediction performance.
- Output
- A web-based system to display energy consumption that is measured in real time with previously predicted values.
- Early-warning system integrated into web-based system to detect anomalous power consumption and provide warnings (notifications) thereof to users.
- The real-time prediction model in web-based system is designed and implemented based on our previous work [20, 33, 49], which developed a novel sliding window metaheuristic optimization-based machine learning system to analyze time-series data that are generated by a smart grid to efficiently forecast energy consumption one day in advance.

### **2.1.5 Smart Home and Energy Efficiency**

[??? NOT IN LIB] - As recently as 2009 it was posited that “The proposed model allows for introduction of power priorities for consumer’s electrical equipment by the importance of their functions. The relevance of the functions carried out by the data device is evaluated by each consumer individually. The relevance of functions and priorities assigned to power mode / groups of electrical equipment can be changed over time.”

‘Haseeb et al [??? NOT IN LIB] have (through review) shown that at a macro scale, global adoption of internet reduces energy consumption’ + At the micro scale (that is, the behaviour of individuals...)

### **2.1.6 IoT and End User Convenience**

The term Internet-of-Things (IoT) is used as an umbrella keyword for covering various aspects related to the extension of the Internet and the Web into the physical realm, by means of the widespread deployment of spatially distributed devices with embedded identification, sensing and/or actuation capabilities

(Daniele Miorandi, 2012). The IoT paradigm is fundamental to this work, representing the confluence of multiple technological advancements (Luigi Atzori, 2010) including, ubiquity of internet access, the availability of high-performance internet connectivity, inexpensive consumer electronics with embedded sensing and control systems, automation, real-time analytics, machine learning, commodity sensors and embedded systems. In the IoT paradigm, digital and physical entities can be linked, by means of appropriate information and communication technologies, to enable a whole new class of applications and services (Daniele Miorandi, 2012).

One consequence resulting from the widespread deployment of consumer electronics with IoT capability is the evolution of the Internet from interconnecting computers to interconnecting things. Figure X, below, shows XYZ.

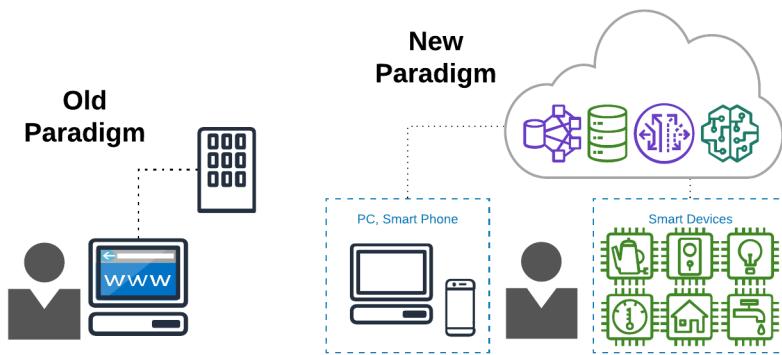


Figure 13: ADD TEXT

### 2.1.7 Heading???

Huang et al propose a novel service mining framework to personalize services in an IoT-based smart home (Bing Huang, 2018). This work considers the notion of personal ‘convenience’ as a driving force behind the provisioning of services to end users. That is, in an IoT smart home, where everything is interconnect, can services (for example, the switching on or off of a light) be automatically served to users such that their level of effort (to interact with their surroundings) will be diminished, and thus their level of convenience will be increased?

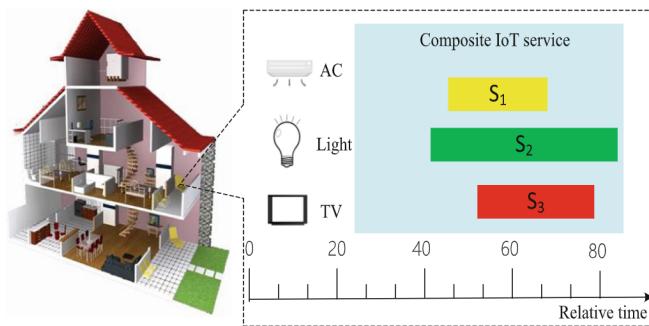


Figure 14: ADD TEXT

Using this framework, IoT services can be composed in a periodic fashion, for the convenience of the end user / consumer.

- Use intelligence to compose IoT services for the convenience of the end user / consumer

The input for this work was data from domestic IoT services, with a corresponding timestamp. The end user in this scenario performs daily activities by interacting with IoT services, these interactions are recorded as IoT service event sequences. A novel PCMiner algorithm (Periodic Composite IoT Service Miner)

Periodic composite IoT services can be loosely defined as the composite IoT services' repeating occurrence at certain locations with regular time intervals

The output for this work was an IoT service model and a composite IoT service model (based on spatio-temporal features). A periodic composite IoT service model to represent the regularity of composite IoT services occurring at a certain location at a certain time interval

IoT service event sequence			Periodic composite IoT services
Microwave	9:59:56	10:00:03	Preparing breakfast
Stereo	10:00:49	10:14:10	(6:00-7:00, in the kitchen)
Freezer	10:03:13	11:36:16	
Sink faucet	11:30:09	11:30:21	Taking a shower (22:00-23:00, in the bathroom)
.....	.....	.....	.....
Refrigerator	18:09:34	18:09:41	Going to bed
Cabinet	18:11:19	18:11:32	(23:00-0:00, in the bedroom)
Toaster	18:11:54	18:11:56	
Door	18:12:21	18:12:26	

Figure 15: ADD TEXT

Critical to note that in this work model composition is reliant on temporal, proximity and duration features.

Output: Spatio-temporal proximity model

- Filter out insignificant spatio-temporal IoT service relationships Categories (e.g., objective):
- Quantify the correlation strength of among IoT services from time and location aspects
- IoT devices are highly heterogeneous in terms of supporting infrastructure ranging from networking to programming abstraction
- Hiding (abstracting) complex and diverse supporting infrastructure
- Proximity and correlation strength

### Service Mining on the Web ???{zheng2009}

In the domain of Service Mining and Service Recognition, Zheng et al aim to provide end users with value-added services through the composition of existing services (output = composed web services). They attempt to craft a Service Mining Tool \* More generalised \* Technology = Service mining tool, web service mining \* Output = composed web services \* Aims at providing value-added services through composing existing services \* It is essential to be able to proactively discover opportunities for composing useful services, even when the goals are unspecified at the moment, or simply hard to imagine or unknown \* Much like the easy access to a glut of data has provided a fertile ground for data mining research, there is an expectation that an increase in Web services' availability will also spur both the need and opportunities to break new ground on Web Services Mining \* We define Web Service Mining as a bottom-up search process aimed at the proactive discovery of potentially interesting and useful web services from existing services.

**Two Main Challenges** \* Combinatorial Explosion \* Evaluation of interestingness and usefulness

### 2.1.8 Summary

In summary there is lots of variability and this informs the construction of our model – it must be robust. By stripping the time variable this greatly simplifies modelling. As mentioned in the work of Murray et al (Appliance electrical consumption modelling at scale using smart meter data) smart home data contains implicit associations, the example given being that, if a person is using the oven between the hours of 6pm and 8pm on a weekday, it can be assumed they are preparing dinner – there is no need to explicitly determine this. In a similar way, we **believe** that we can implicitly obtain spatial and temporal information about how a series of appliances (electrical and non-electrical) are interrelated.

As demonstrated above, there has been a vast swathe of work done in the areas of smart meter data and predictive analytics, X, Y & Z. What is, however, of significant note, is the absence of holistic consideration with respect to energy consumption, smart home data, predictive analytics, IoT and end user convenience. As demonstrated above, numerous studies have found that creating and maintaining behaviour (such as monitoring a smart meter and then intervening) is ... shit.

Due to  $n = 1$ , it would be a fair criticism to say that the methodology employed cannot claim to be robust... this is fair. However, the Kavousian et al thoroughly analyse the effect of household variables on overall energy efficiency. Our analysis ( $n=1$ ) has no such ... Take for example Consumer preferences for feedback on household electricity consumption (REF). The work considers, in isolation, the concept of saving electricity. For example during participant interviews, the line of questioning is framed around electricity saving NOT cost or convenience.

1. The cost of the sensors is negligible (EXCEPT for discussion later RE top 5 features)
  2. The energy requirement for the sensors is negligible
  3. In our proposed model, computation is performed in the cloud
  4. An app is available to interact with the smart home
  5. All of the electrical appliances in our smart home can be remote controlled to some extent
- What about, instead of responding to the average time, the antagonistic AI responded to price signals in the market?

## 2.2 The Data Set

The datasets were created during the thesis *Activity Recognition with End-User Sensor Installation in the Home* by Randy Joseph Rockinson, Submitted to the Program of Media Arts and Sciences, School of Architecture and Planning, in partial fulfilment of the requirement for the degree of Master of Science in Media Arts and Sciences at the Massachusetts Institute of Technology (MIT) February 2008 (REF).

The aim of Rockinson was to consider the effect of end user versus professional installation of a sensor array in the home - on the basis that, if installation of sensors is to be considered as a high initial cost, and a barrier to entry for end users wanting this technology, is there a difference if a professional versus an end user performs the installation?

Between 80-100 reed switch sensors were installed in two single-person apartments collecting data about human activity for two weeks. The sensors were installed in everyday objects such as drawers, refrigerators, containers, etc to record opening-closing events (activation deactivation events) as the subject carried out everyday activities.

### 3 Data Preprocessing & Visualisation

#### 3.1 Importing & Preprocessing the Activities Meta Data

The dataset `S1Activities.csv` was imported into the interactive development environment. These data contains a tabulated summary of Heading, Category, Subcategory and a corresponding unique code. After importation, the dataset has dimensionality of [3, 33], with `Heading`, `Category` & `Subcategory` present as non-null objects, as seen in Table 1 below. The attribute `Code` (which codifies the unique set of `Heading`, `Category` & `Subcategory`) was imported as an index value ( $n=33$ ). At this time, the activities data will be kept in it's native state and will not be subject to preprocessing.

Table 1: The S1 activities dataset

	Heading	Category	Subcategory
1	Employment related	Employment work at home	Work at home
5	Employment related	Travel employment	Going out to work
10	Personal needs	Eating	Eating
15	Personal needs	Personal hygiene	Toileting
20	Personal needs	Personal hygiene	Bathing

#### 3.2 Importing & Preprocessing the Sensor Meta Data

The dataset `S1sensors.csv` was imported into the interactive development environment. These data contains a tabulated values for Sensor ID, Room and Sensor Descriptor (e.g., light switch), with no header row present in the original dataset. After importation, the dataset has dimensionality of [3, 76], with header 0, 1 & 2 corresponding to `SensorID`, `Room` & `Sensor Descriptor`, respectively, as seen in Table 2. As all attributes are nominal, they were imported as String data types.

Table 2: The S1 sensor meta data

0	1	2
100	Bathroom	Toilet Flush
101	Bathroom	Light switch
104	Foyer	Light switch
105	Kitchen	Light switch
106	Kitchen	Burner

After importation these data were checked for duplicate values. The `Sensor ID` attribute had a 76 unique values, the `Room` attribute had only 7 unique values and the `Sensor Descriptor` attribute had 28 unique values. Examples of the degeneracy in these attributes can be seen in Table 2, e.g., Bathroom & Light Switch. `Room[1]` and `SensorDescriptor[2]` were stripped of whitespace, coerced to lowercase and concatenated using an underscore. The concatenated vector was then bound to the dataframe and the column names updated, resulting in 3, below.

Table 3: The first iteration of processed S1 sensor meta data

subActNum	room	activity	concat
100	Bathroom	Toilet Flush	bathroom_toiletflush
101	Bathroom	Light switch	bathroom_lightswitch
104	Foyer	Light switch	foyer_lightswitch
105	Kitchen	Light switch	kitchen_lightswitch
106	Kitchen	Burner	kitchen_burner
107	Living room	Light switch	livingroom_lightswitch

The number of unique values in `dsS1Sensors.subActNum` is checked once more and found to be  $n=76$ , indicating no degeneracy in this attribute. The number of unique values in `dsS1Sensors.concat` is found

to be n=41, indicating the presence of degeneracy. This was investigated by aggregating the duplicate attributes, which are summarised in Table 4, below.

Table 4: The first iteration of processed S1 sensor meta data

n	concat
3	kitchen_lightswitch
4	kitchen_burner
2	livingroom_lightswitch
7	kitchen_drawer
3	kitchen_refrigerator
15	kitchen_cabinet
2	kitchen_door
5	bedroom_drawer
2	bathroom_medicinecabinet
2	bathroom_cabinet

Refering to the work of (REF) it was determined that these duplicate values are the result of multiple sensors with extremely similar functionality. Kitchen\_burner has a value of n=4, accounted for by the presence of one sensor per burner in the original work. While this level of granularity may provide an avenue for further analysis, for the purposes of this research such values serve to significantly increase the dimensionality of the overall dataset, with a low corresponding gain in information. High dimensionality can also lead to difficulties in machine learning models and challenges with data visualisation. By way of example, kitchen\_cabinet has a value of n=15, indicating that for the various cabinets in the kitchen, a total of 15 sensors were fitted to monitor activity. It was determined that such values would aggregated and collapsed to a value of n=1 (with the first occurrence of the degenerate subactivity thus providing the subActNum). The data set was inspected and stripped of special characters, and the sub activities requiring an input of energy (e.g., electrical, gas) were flagged with a boolean value, resulting in Table 5. EXPORTED as **S1Sensors\_preprocessed.csv DUPES NOT YET DROPPED**

Table 5: Demo table

subActNum	room	activity	concat	reqEnergy	subActNumConcat
100	Bathroom	Toilet Flush	bathroom_toiletflush	False	subActNum_100
101	Bathroom	Light switch	bathroom_lightswitch	True	subActNum_101
104	Foyer	Light switch	foyer_lightswitch	True	subActNum_104
105	Kitchen	Light switch	kitchen_lightswitch	True	subActNum_105
106	Kitchen	Burner	kitchen_burner	True	subActNum_106
107	Living room	Light switch	livingroom_lightswitch	True	subActNum_107

### 3.3 Importing & Preprocessing the Activities Data

The activities data set **S1activities.csv** will be imported, evaluated and cleansed. The goal of the pre-processing will be to restructure the dataset into a ‘tidy’ format, that is, where the attributes are columns, the rows are instances, and each cell contains only one value. As the data is time-series, timestamps may be used as indexes. Figure 16 shows the data set open in a spreadsheet program. Inspection of the original dataset via an interactive development environment shows that the original dataset follows a structure such that each 5 rows contains is one discrete set of data.

- Row 1: Activity, Date, Start Time, End Time
- Row 2: Sub-activity (an action that can be executed as part of performing the activity) code-values
- Row 3: Sub-activity descriptive values
- Row 4: Sub-activity start time
- Row 5: Sub-activity end time

	A	B	C	D	E	F	G	H	I	J	K	L	M	N	O	P	Q	R	S	T
1	Bathing	4/1/03	20:41:35	21:32:50																
2	100	68	81	101	93	137	93	58	57	67	93	58	68	88	57	67	100	68	67	76
3	Toilet Flush	Sink faucet - Closet	Light switch	Shower faucet	Freezer		Shower faucet	Medicine cabinet	Medicine cabinet	Cabinet	Shower faucet	Medicine cabinet	Sink faucet -	Sink faucet -	Medicine cabinet	Cabinet	Toilet Flush	Sink faucet - Cabinet	Lamp	
4	20:51:52	20:51:58	20:53:36	20:53:49	20:53:52	20:58:22	20:58:43	21:05:23	21:05:46	21:05:47	21:18:34	21:18:55	21:19:41	21:20:04	21:20:38	21:20:39	21:21:13	21:21:16	21:21:37	21:22:08
5	21:05:20	20:52:05	20:53:43	21:21:43	20:58:42	20:58:32	21:06:09	21:05:45	21:18:55	21:05:49	21:18:35	21:20:37	21:20:05	21:20:34	21:21:41	21:20:42	23:10:23	21:21:23	21:21:38	23:11:08
6	Toileting	4/1/03	17:30:36	17:46:41																
7	100	68																		
8	Toilet Flush	Sink faucet - hot																		
9	17:39:37	17:39:46																		
10	18:10:57	17:39:52																		
11	Toileting	4/1/03	18:04:43	18:18:02																
12	68	107																		
13	Sink faucet - Light switch																			
14	18:11:02	18:12:28																		
15	18:11:13	21:21:53																		
16	Toileting	4/1/03	11:52:01	11:58:50																
17	100	137																		
18	Toilet Flush	Freezer																		
19	11:55:43	11:56:02																		
20	16:35:49	11:56:13																		

Figure 16: ADD TEXT

Table 6: The S1 activities dataset after importation into the interactive development environment

0
5 Toileting,4/1/2003,17:30:36,17:46:41
6 100,68
7 Toilet Flush,Sink faucet - hot
8 17:39:37,17:39:46
9 18:10:57,17:39:52

The activities data was imported into an indexed dataframe, containing only one column, with 1475 rows, with all values comma-separated (per row). This style of import had to be used, due to the varying number of comma-separated elements in each row (as seen in Figure 16, above).

- Between each increment of 5 (0 - 4), the sub-arrays have different lengths Table [ref], below, contains a summary of the data structure after the operation `np.array(dsS1)` is performed. In order to continue pre-processing, the array had to be flattened from a 2D structure to a 1D structure, using `flatten()`.

The function `dsS1Activities_processing.py` was run in-line to perform the required preprocessing. The dataframe was then converted to a 2D array, using `np.array()`, here, each row from the dataframe became an array within the 2D array [Step 1]. The 2D array was flattened to a 1D array using `.flatten()` [Step 2]. each group of observations was then chunked into a list of 5 [Step 3]

Mention sanity checking here

We used the Step 3 data structure to then extract the values for `activity`, `date`, `startTime` and `endTime`. These values were populated into a series of temporary arrays, which were then compiled into a dataset with the four attributes previously listed. We then used `Step3[i][j]` to access all the required elements and parse them into the arrays.

### 3.3.0.1 Algorithm 1

Text

---

### Algorithm 1: Extraction of data from S1 Activities dataset

---

```

Result: Intermediate dataframe with 'activity', 'date', 'startTime', 'endTime' as attributes
Input : S1 Activities
Output: S1 Activities Intermediate

1 begin
2     array1 = []
3     array2 = []
4     array3 = []
5     array4 = []
6     array5 = []
7     array6 = []
8     array7 = []
9     array8 = []
10    while i < length(dataframe) do
11        array1.append(dataframe[i][0])
12        array2.append([x.strip() for x in array1[i].split(',')])
13        array3.append(array2[i][0])
14        array4.append(array2[i][1])
15        array5.append(array2[i][2])
16        array6.append(array2[i][3])
17        i = i + 1
18    end
19    dfIntermediate = pandas.DataFrame(list(zip(array3, array4, array5, array6)))
20    start = (dfIntermediate.date + " " + dfIntermediate.startTime)
21    end = (dfIntermediate.date + " " + dfIntermediate.endTime)
22    while i < length(start) do
23        array7.append(datetime.strptime(start[i], mm/dd/yyyy HH:MM:SS))
24        array8.append(datetime.strptime(end[i], mm/dd/yyyy HH:MM:SS))
25        i = i + 1
26    end
27    dfFinal = pandas.DataFrame(list(zip(array3, array7, array8)))
28    return dfIntermediate
29    return dfFinal
30 end

```

---

Table 7: Demo table

activity	date	startTime	endTime
Bathing	4/1/2003	20:41:35	21:32:50
Toileting	4/1/2003	17:30:36	17:46:41
Toileting	4/1/2003	18:4:43	18:18:2
Toileting	4/1/2003	11:52:1	11:58:50
Going out to work	4/1/2003	12:11:26	12:15:12
Preparing lunch	4/1/2003	11:21:17	11:38:22

Table 8: Demo table

activity	start	end
Bathing	2003-04-01 20:41:35	2003-04-01 21:32:50
Toileting	2003-04-01 17:30:36	2003-04-01 17:46:41
Toileting	2003-04-01 18:04:43	2003-04-01 18:18:02
Toileting	2003-04-01 11:52:01	2003-04-01 11:58:50
Going out to work	2003-04-01 12:11:26	2003-04-01 12:15:12
Preparing lunch	2003-04-01 11:21:17	2003-04-01 11:38:22

## 3.4 Importing, Visualizing and Preprocessing the SubActivities Data

Importing `S1Activities_data.csv`, convert df to an array (list?), flatten to a 1D array (list?), chunk the array [5], extract subActNum, subActivity, time & date. Merge time and date into datetime elements, determine start and end time. The variable `dsS1` was used (still on stack from previous section). As previously observed (FIGURE X, TABLE Y), data was all in one column. A processing method with additional steps BLAH BLAH.

### 3.4.0.1 Algorithm 1

Text

---

### Algorithm 2: Extraction of data from S1 Activities dataset

---

```

Result: Intermediate dataframe with 'activity', 'date', 'startTime', 'endTime' as attributes
Input : S1 Activities
Output: S1 Activities Intermediate

1 begin
2     array1-21 = []
3     while i < length(dataframe) do
4         array1.append(dataframe[i][0])
5         array2.append([x.strip() for x in array1[i].split(',')])
6         array3.append(array2[i][0])
7         array4.append(array2[i][1])
8         array5.append(array2[i][2])
9         array6.append(array2[i][3])
10        i = i + 1
11    end
12    while i < length(dataframe) do
13        array7.append(dataframe[i][1])
14        array8.append([x.strip() for x in array7[i].split(',')])
15        array9.append(dataframe[i][2])
16        array10.append([x.strip() for x in array9[i].split(',')])
17        array11.append(dataframe[i][3])
18        array12.append([x.strip() for x in array11[i].split(',')])
19        array13.append(dataframe[i][4])
20        array14.append([x.strip() for x in array13[i].split(',')])
21        i = i + 1
22    end
23    while i < length(dataframe) do
24        for x in range(len(array8[i])) : array15.append(array4[i])
25        i = i + 1
26    end
27    for sublist in array8: for item in sublist: array16.append(item)
28    for sublist in array10: for item in sublist: array17.append(item)
29    for sublist in array12: for item in sublist: array18.append(item)
30    for sublist in array13: for item in sublist: array19.append(item)
31    dfIntermediate = pandas.DataFrame(list(zip(array16, array17, array15, array18, array19)))
32    start = (dfIntermediate.date + " " + dfIntermediate.startTime)
33    end = (dfIntermediate.date + " " + dfIntermediate.endTime)
34    while i < length(start) do
35        array20.append(datetime.strptime(start[i], mm/dd/yyyy HH:MM:SS))
36        array21.append(datetime.strptime(end[i], mm/dd/yyyy HH:MM:SS))
37        i = i + 1
38    end
39    dfFinal = pandas.DataFrame(list(zip(array16, array17, array20, array21)))
40    return dfIntermediate
41    return dfFinal
42 end

```

---

Table 9: Demo table

subActNum	subAct	date	startTime	endTime
100	Toilet Flush	4/1/2003	20:51:52	21:5:20
68	Sink faucet - hot	4/1/2003	20:51:58	20:52:5
81	Closest	4/1/2003	20:53:36	20:53:43
101	Light switch	4/1/2003	20:53:49	21:21:43
93	Shower faucet	4/1/2003	20:53:52	20:58:42
137	Freezer	4/1/2003	20:58:22	20:58:32

Table 10: Demo table

subActNum	subAct	start	end
67	Cabinet	2003-03-27 06:43:40	2003-03-27 06:43:43
100	Toilet Flush	2003-03-27 06:44:06	2003-03-27 07:12:41
101	Light switch	2003-03-27 06:44:20	2003-03-27 07:46:34
57	Medicine cabinet	2003-03-27 06:44:35	2003-03-27 06:44:48
58	Medicine cabinet	2003-03-27 06:44:36	2003-03-27 06:44:48
67	Cabinet	2003-03-27 06:44:49	2003-03-27 06:44:56

**Extracting the Activity, Time and Data** If we run `a[0][0]`, we get 'Bathing,4/1/2003,20:41:35,21:32:50'

\* A string with 4 comma-separated 'items' \* We won't check the entire DF, rather, will rely on errors thrown back to confirm validity ('validation of the method')

**Constructing the while loop** \* Create empty lists \* Extracts the relevant elements \* Adds (appends) the elements to the lists

**Sanity Checks** \* We won't check the entire DF, rather, will rely on errors thrown back to confirm validity ('validation of the method') \* Divisible by 5

`len(dsS1) = 1475 len(df) = 2772` "We have a df of 2772 subActs over time period X in the MIT Place Lab"

### 3.4.1 Sub Activity Preprocessing - Duplicate sub activities

The following duplicate sensors were identified in section XX

- From `S1Sensors_preprocessed.csv` I need to find the numbers associated with these - SUBACT- NUM
- 'Arbitrarily' choose one number to represent all of them
- Fill them all with that one number

**For discussion later** - The kitchen cabinets have 15 sensors, say one of these cabinets just contains a blender, so there will always be a one-to-one between the cabinet opening and the blender being used. This isn't particularly helpful to us. Also, more sensors could always be added... Better to do analysis with less (explain?) Note for discussion, `S1Sensors_preprocessed` itself should no longer be modified, EXAMPLE, going back upstream to remove the 'dups', would perhaps cause an lack of traceability downstream later, Importing `S1Sensors_preprocessed` now has been done to inform, how to remove 'dups' from the current ds

### 3.4.2 SubActivity Visualisation and Outlier Inspection

```
p <- ggplot(total_counts, aes(x = subAct, y = count, group = DAY, color = DAY))
p <- p + geom_line(size=0.5) + geom_point(size = 0.5) + theme_grey()
p <- p + coord_flip() + labs(title = "Plot of SubActivity Count versus SubActivity",
                               x = "SubActivity",
                               y = "Aggregated Count") +
  theme(text = element_text(size=8),
        axis.title.x=element_blank(),
        legend.title = element_blank(),
        legend.position = c(0.9,0.84),
        legend.background = element_rect(fill=alpha('white', 0.5)),
        legend.key.size = unit(5, "mm"))

ggsave('/Users/alistaирgj/Documents/GitHub/IoT_ResearchProject/IoT_November/images/lineChart.png'
       plot = last_plot(), device = png(),
       scale = 1.5, width = 8, height = 10, units = c("cm"), dpi = 300)
```

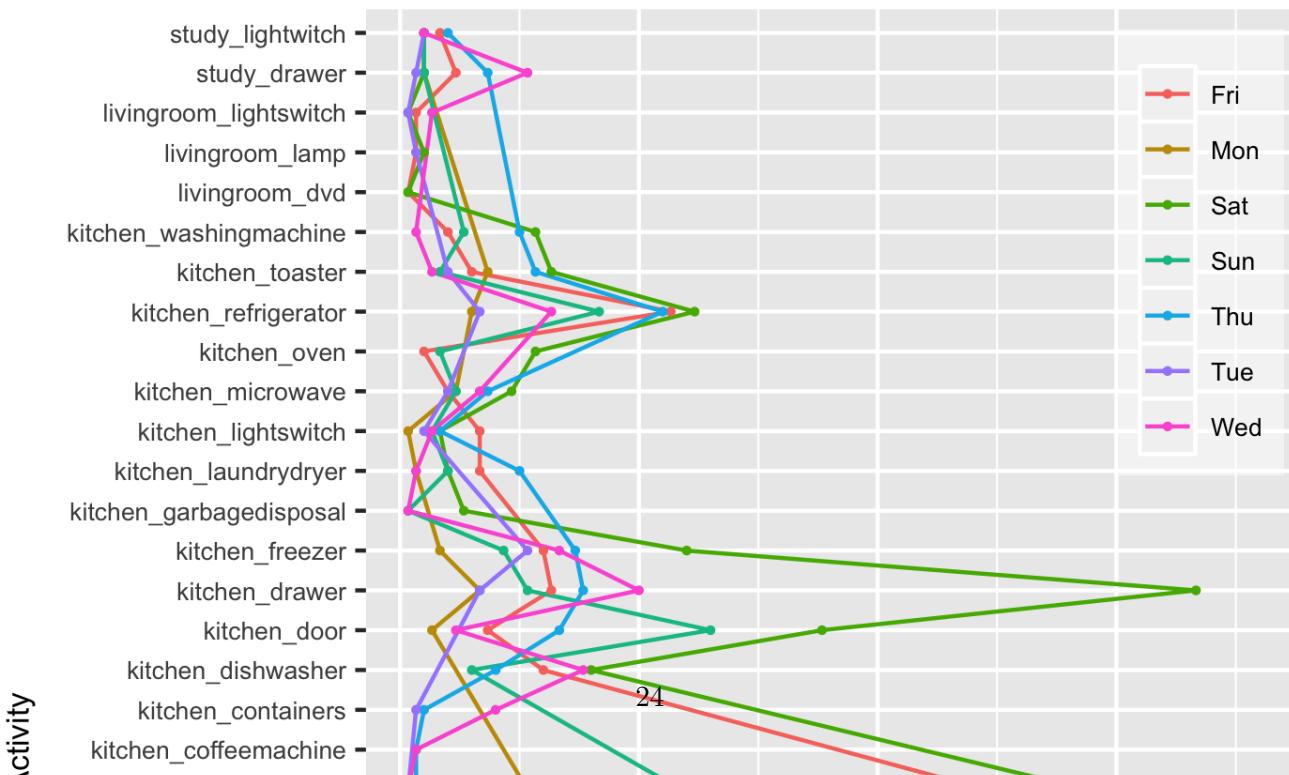
#### 3.4.2.1 Aggregated Line Chart

The following figure shows XYZ

Table 11: Demo table

subActNum	subAct	start	end	dayNumeric	DAY	WDWE	HOUR	durationSec
67	bathroom_cabinet	2003-03-27 06:43:40	2003-03-27 06:43:43	3	Thu	WD	6	4
100	bathroom_toiletfush	2003-03-27 06:44:06	2003-03-27 07:12:41	3	Thu	WD	6	1716
101	bathroom_lightswitch	2003-03-27 06:44:20	2003-03-27 07:46:34	3	Thu	WD	6	3735
57	bathroom_medicinecabinet	2003-03-27 06:44:35	2003-03-27 06:44:48	3	Thu	WD	6	14
82	study_drawer	2003-03-27 06:45:45	2003-03-27 06:45:48	3	Thu	WD	6	4
146	bedroom_drawer	2003-03-27 06:46:12	2003-03-27 06:46:20	3	Thu	WD	6	9
132	kitchen_cabinet	2003-03-27 06:51:43	2003-03-27 06:51:46	3	Thu	WD	6	4
143	kitchen_microwave	2003-03-27 06:54:09	2003-03-27 13:07:43	3	Thu	WD	6	22415
141	kitchen_door	2003-03-27 06:57:05	2003-03-27 06:57:08	3	Thu	WD	6	4
93	bathroom_showerfaucet	2003-03-27 07:05:22	2003-03-27 07:05:24	3	Thu	WD	7	3
125	kitchen_drawer	2003-03-27 07:38:48	2003-03-27 07:38:51	3	Thu	WD	7	4
70	kitchen_dishwasher	2003-03-27 07:40:32	2003-03-27 07:40:34	3	Thu	WD	7	3
126	kitchen_refrigerator	2003-03-27 07:41:08	2003-03-27 07:41:16	3	Thu	WD	7	9
88	bathroom_sinkfaucet-cold	2003-03-27 07:43:22	2003-03-27 07:43:48	3	Thu	WD	7	27
68	bathroom_sinkfaucet-hot	2003-03-27 07:43:23	2003-03-27 07:43:48	3	Thu	WD	7	26
140	foyer_door	2003-03-27 07:48:49	2003-03-27 07:48:53	3	Thu	WD	7	5
137	kitchen_freezer	2003-03-27 11:36:15	2003-03-27 11:36:21	3	Thu	WD	11	7
106	kitchen_burner	2003-03-27 11:37:13	2003-03-27 11:42:24	3	Thu	WD	11	312
105	kitchen_lightswitch	2003-03-27 11:38:54	2003-03-27 12:31:05	3	Thu	WD	11	3132
92	study_lightswitch	2003-03-27 16:28:23	2003-03-27 18:44:14	3	Thu	WD	16	8152
130	bathroom_door	2003-03-27 17:43:10	2003-03-27 17:43:12	3	Thu	WD	17	3
104	foyer_lightswitch	2003-03-27 19:36:32	2003-03-27 19:50:07	3	Thu	WD	19	816
131	kitchen_toaster	2003-03-28 12:32:00	2003-03-28 15:10:16	4	Fri	WD	12	9497
96	bathroom_exhaustfan	2003-03-28 17:43:17	2003-03-28 18:25:19	4	Fri	WD	17	2523
108	bedroom_lightswitch	2003-03-28 18:24:48	2003-03-28 19:55:44	4	Fri	WD	18	5457
129	kitchen_oven	2003-03-28 19:09:26	2003-03-28 19:09:30	4	Fri	WD	19	5
56	livingroom_dvd	2003-03-28 19:55:24	2003-03-28 19:55:29	4	Fri	WD	19	6
76	livingroom_lamp	2003-03-29 13:30:51	2003-03-29 18:50:13	5	Sat	WE	13	19163
139	bedroom_jewelrybox	2003-03-29 15:12:44	2003-03-29 15:12:45	5	Sat	WE	15	2
142	kitchen_washingmachine	2003-03-29 15:44:24	2003-03-29 15:44:26	5	Sat	WE	15	3
90	kitchen_laundrydryer	2003-03-29 15:48:18	2003-03-29 15:48:54	5	Sat	WE	15	37
98	kitchen_garbagedisposal	2003-03-29 15:54:50	2003-03-29 15:54:51	5	Sat	WE	15	2
107	livingroom_lightswitch	2003-03-29 18:50:43	2003-03-29 18:51:09	5	Sat	WE	18	27
81	foyer_closet	2003-04-01 06:13:26	2003-04-01 06:13:34	1	Tue	WD	6	9
145	kitchen_cereal	2003-04-01 06:37:18	2003-04-01 06:37:59	1	Tue	WD	6	42
60	kitchen_containers	2003-04-01 06:38:14	2003-04-01 06:38:48	1	Tue	WD	6	35
119	kitchen_coffeemachine	2003-04-09 08:46:07	2003-04-09 08:46:35	2	Wed	WD	8	29
64	bedroom_lamp	2003-04-09 20:16:36	2003-04-09 21:13:19	2	Wed	WD	20	3404

Plot of SubActivity Count versus SubActivity



### 3.4.2.2 Aggregated Box Plot

The following figure shows XYZ

- We can see many potential outliers.
- Of particular concern Kitchen\_Toaster with a duration of almost 30,000 seconds, Bathroom\_toiletflush distributed up to the range of 20,000 seconds.

In order to investigate this we will work through each subAct

```
p <- ggplot(df, aes(subAct, durationSec))
p <- p + geom_boxplot(fill= "plum", outlier.alpha = 0.8) + # aes(colour = WDWE) #, outlier.size =
coord_flip() + theme_grey() +
labs(title = "Box Plots",
x = "SubActivity",
y = "Duration (sec)") + theme(text = element_text(size=8))
ggsave("images/boxPlots.png", plot = last_plot(), device = png(),
scale = 1.5, width = 18, height = 12, units = c("cm"), dpi = 300)
```

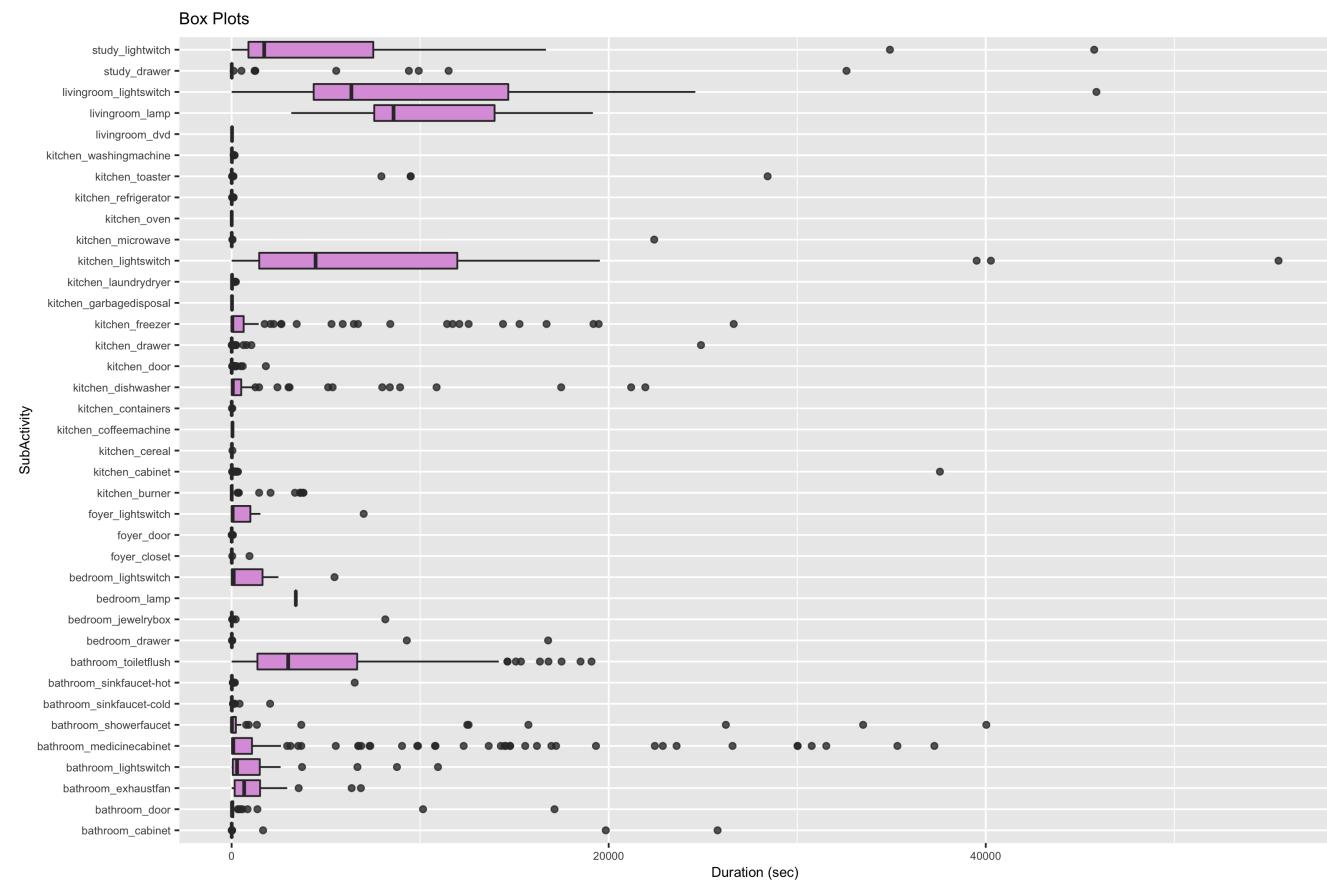


Figure 18: ADD TEXT

### 3.4.2.3 Strip Plots

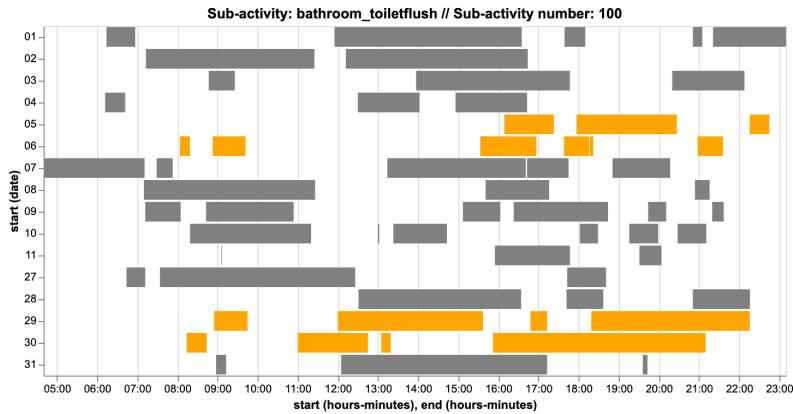


Figure 19: ADD TEXT

### 3.4.3 SubActivity Cleansing - Outlier Removal

Based on the investigation of the strip plots + XYZ

```
# valueDrop.py
ds = ds[ds.subAct != 'bedroom_jewelrybox']
ds = ds[ds.subAct != 'foyer_closet']
ds = ds[ds.subAct != 'kitchen_cereal']
ds = ds[ds.subAct != 'kitchen_containers']
ds = ds[ds.subAct != 'bedroom_lamp']
ds = ds[ds.subAct != 'livingroom_dvd']
```

## 4 Jewelry + foyer closet + cereal + containers + lamp

%run -i valueDrop.py

Outliers as defined by how much from the mean / median / mode???

Table 12: Demo table

SubAct	Count	Median	Mean	Std
bathroom_cabinet	104	4.0	460.22115	3176.1633
bathroom_medicinecabinet	194	88.0	3188.97423	7354.1424
study_drawer	45	6.0	1634.48889	5432.4063
bedroom_drawer	99	10.0	273.47475	1918.7262
kitchen_cabinet	406	7.0	112.82020	1864.0122
kitchen_microwave	61	6.0	377.40984	2868.6679
kitchen_door	134	4.0	46.11194	174.1857
bathroom_showerfaucet	88	15.0	1754.35227	6540.7295
kitchen_drawer	208	4.0	145.06250	1727.5767
bathroom_sinkfaucet-hot	169	11.0	57.90533	501.2729
kitchen_freezer	130	39.0	1738.10769	4512.5738
bathroom_door	73	17.0	461.61644	2310.8116
kitchen_toaster	71	5.0	790.36620	3793.6744
kitchen_lightswitch	32	4454.0	9318.71875	13081.3404
study_lightswitch	26	1728.5	6430.73077	11002.3025
kitchen_dishwasher	86	63.5	1518.40698	4146.6375
livingroom_lightswitch	8	6352.5	12545.12500	15556.6628

S1SubActivities\_temporalFeaturesCLEANSED.csv = 'Final' dataset

#### 4.0.0.1 Bathroom Cabinet

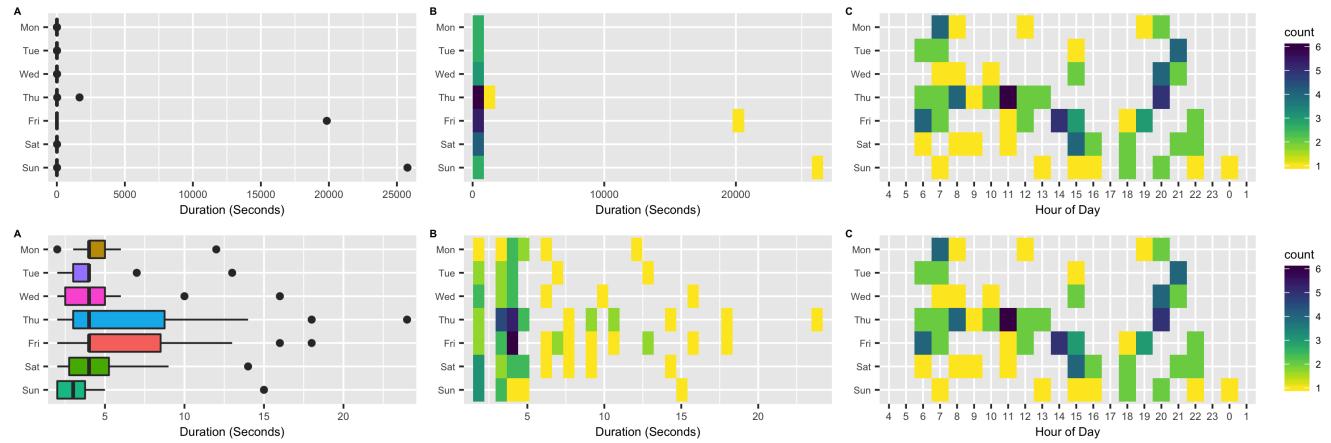


Figure 20: A caption

#### 4.0.0.2 Bathroom Toiletflush

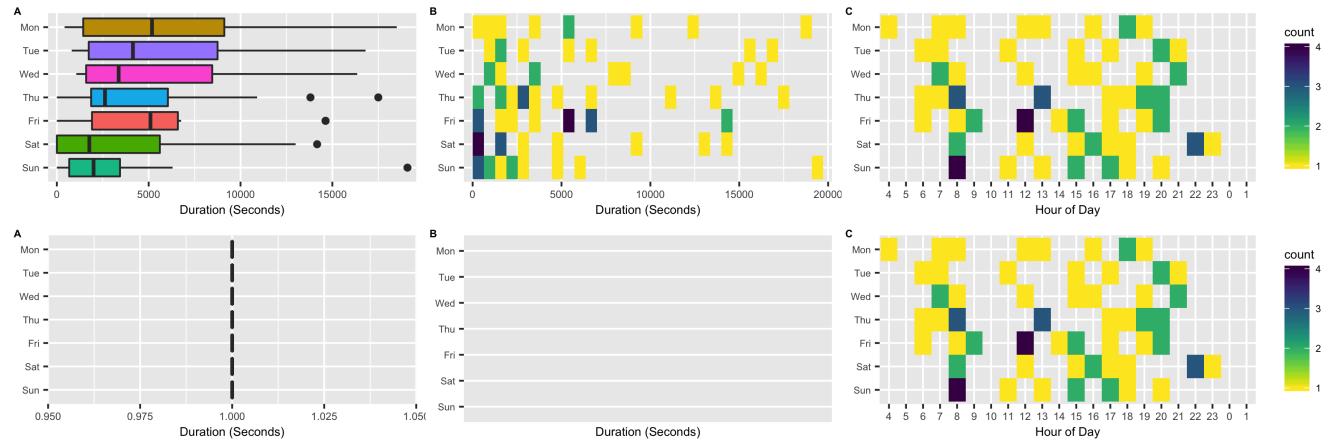


Figure 21: A caption

#### 4.0.0.3 Bathroom Lightswitch

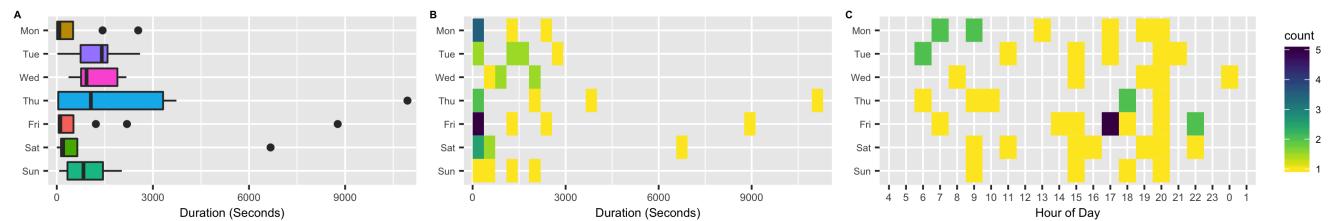


Figure 22: A caption

#### 4.0.0.4 Bathroom Medicine Cabinet

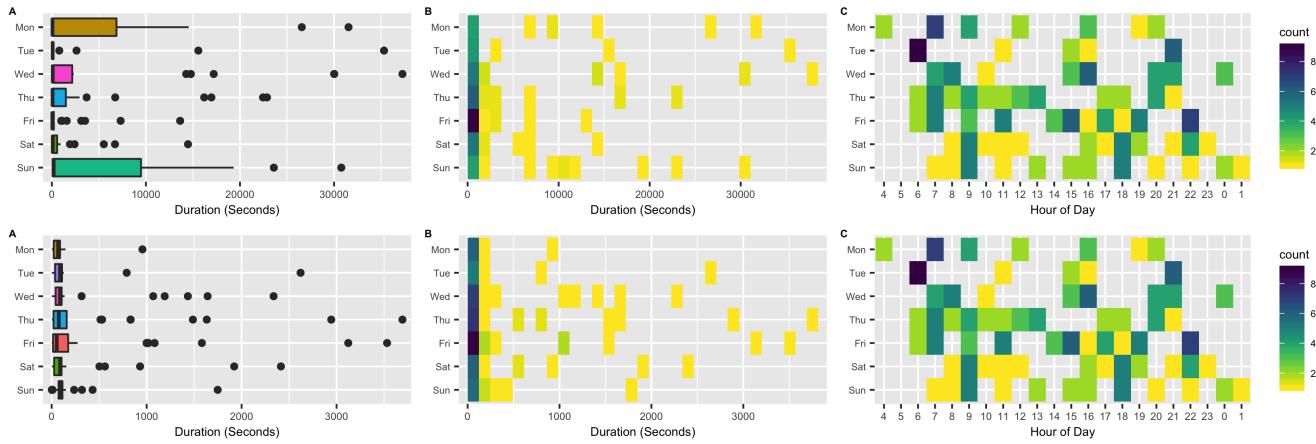


Figure 23: A caption

#### 4.0.0.5 Study Drawer

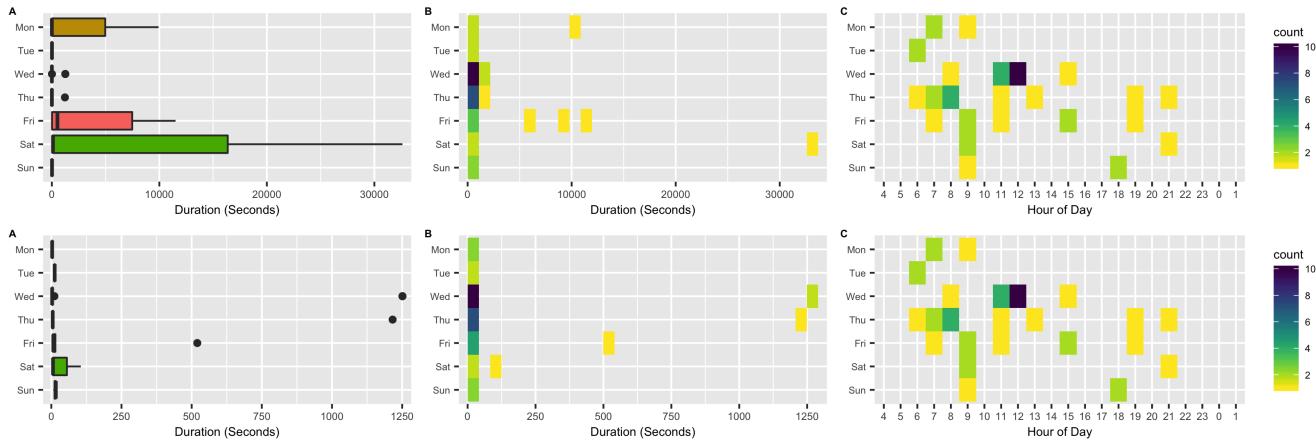


Figure 24: A caption

#### 4.0.0.6 Bedroom Drawer

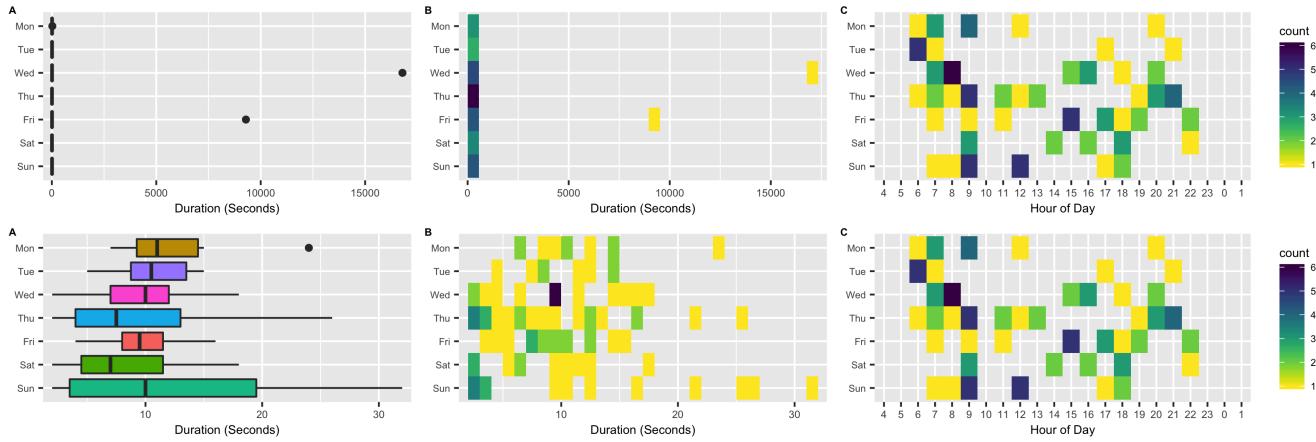


Figure 25: A caption

#### 4.0.0.7 Kitchen Cabinet

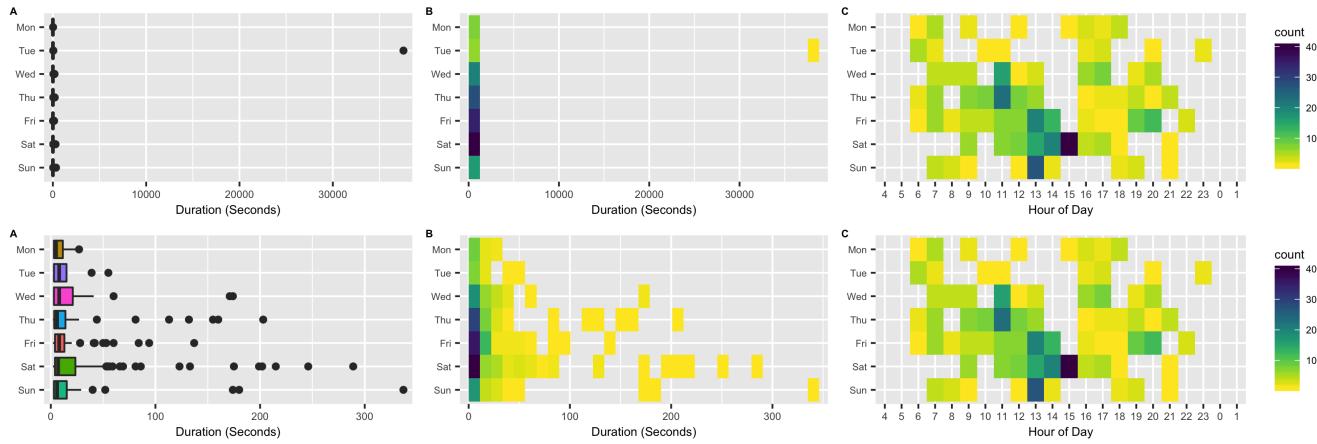


Figure 26: A caption

#### 4.0.0.8 Kitchen Microwave

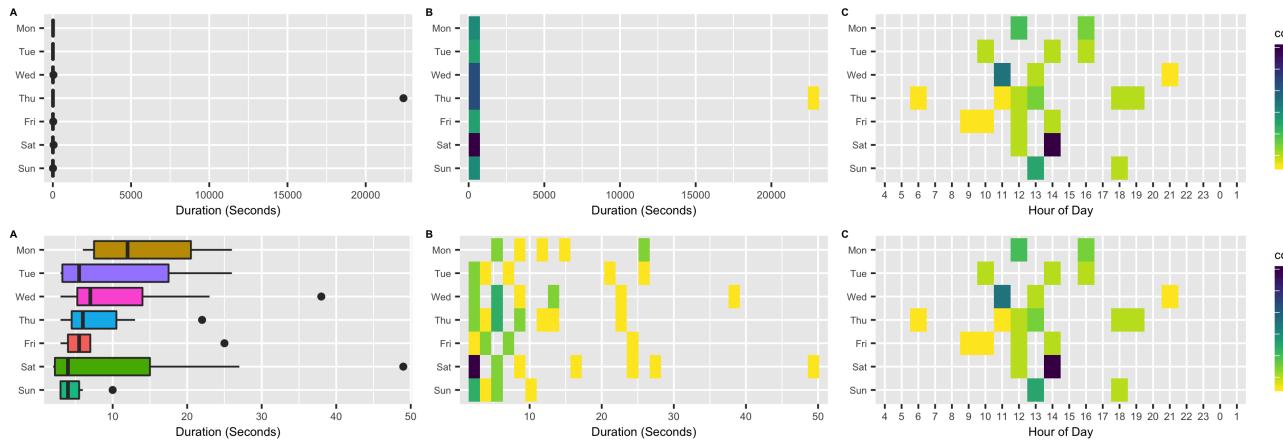


Figure 27: A caption

#### 4.0.0.9 Kitchen Door

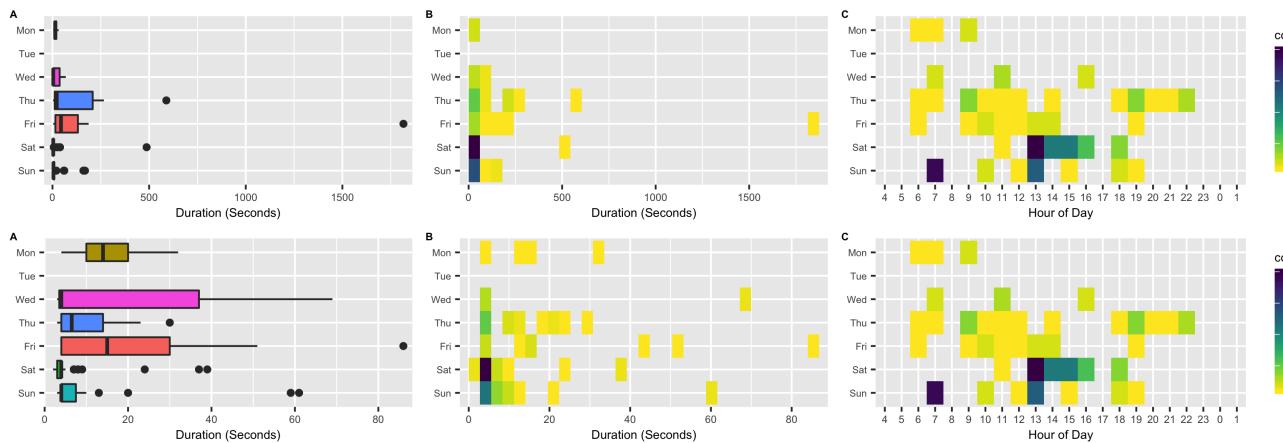


Figure 28: A caption

#### 4.0.0.10 Bathroom Shower Faucet

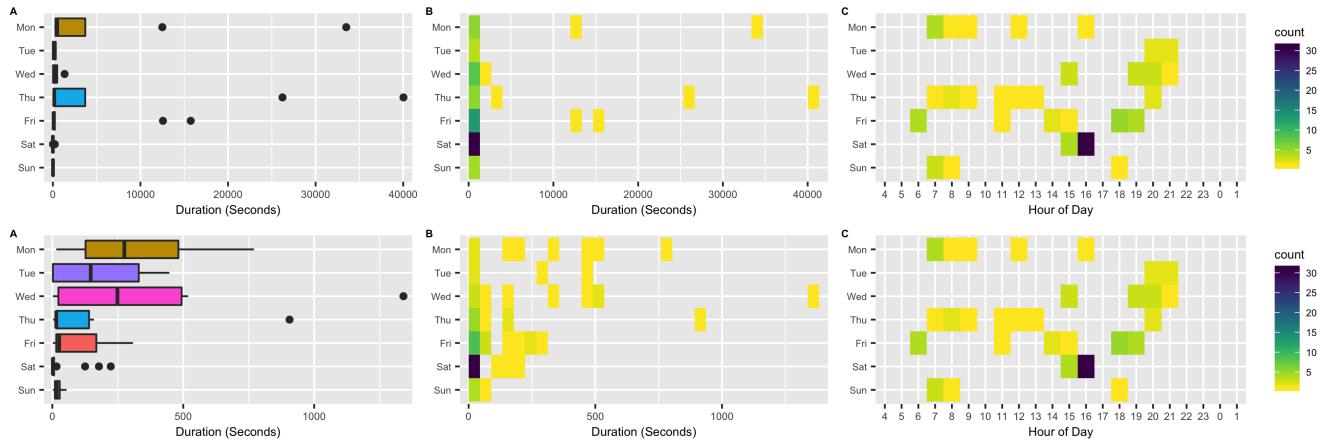


Figure 29: A caption

#### 4.0.0.11 Kitchen Drawer

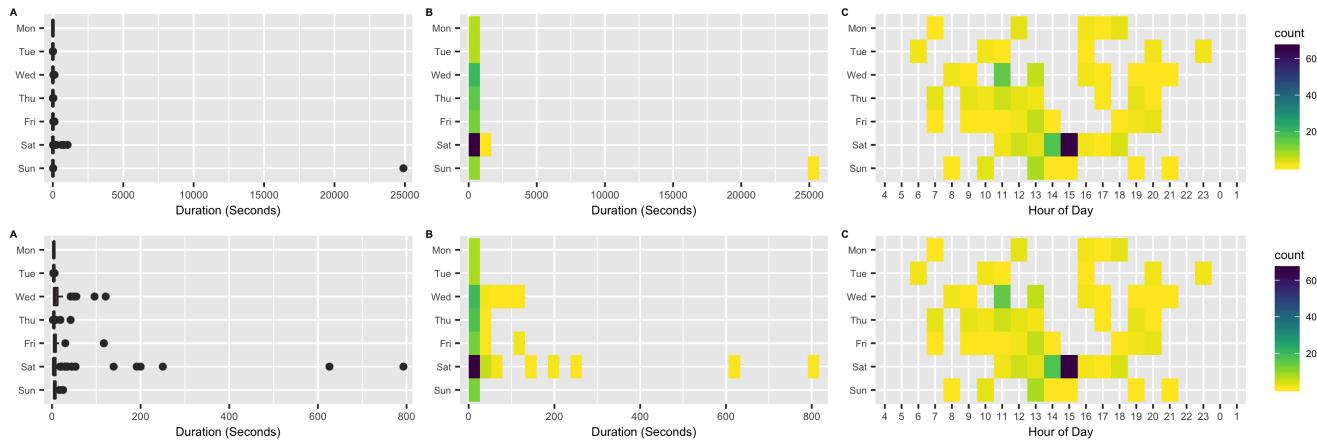


Figure 30: A caption

#### 4.0.0.12 Kitchen Dishwasher

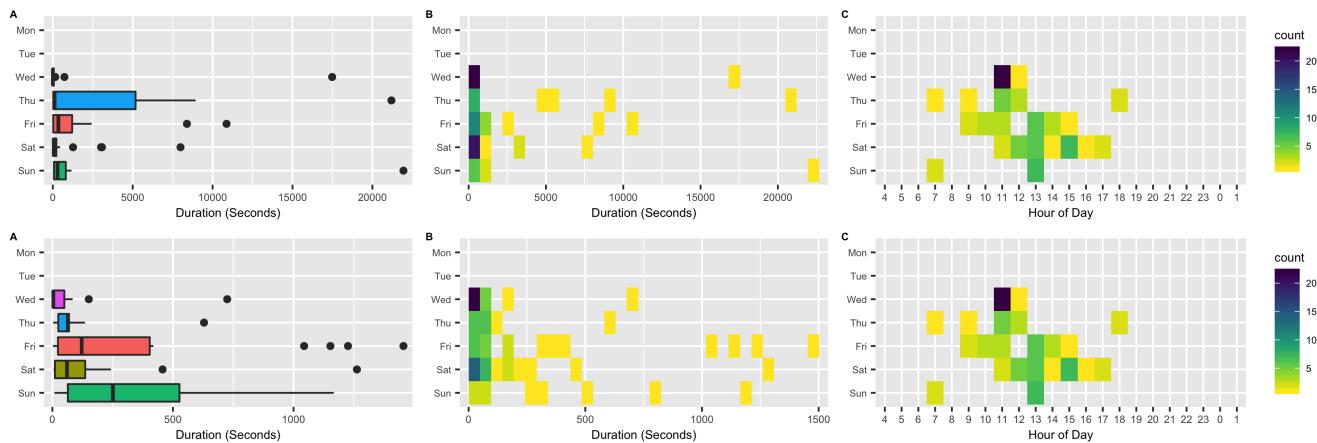


Figure 31: A caption

#### 4.0.0.13 Kitchen Refrigerator

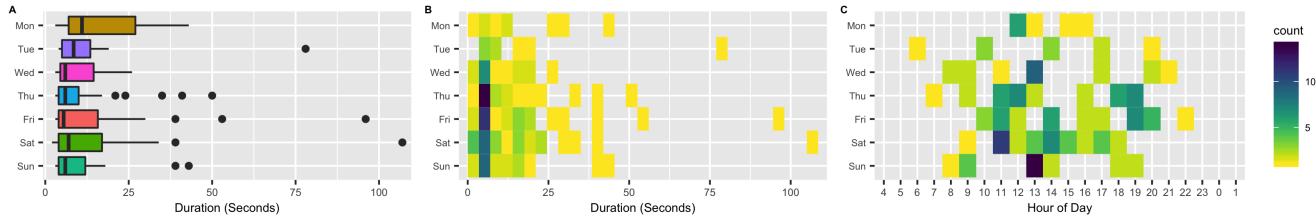


Figure 32: A caption

#### 4.0.0.14 Bathroom Sink Faucet - Cold

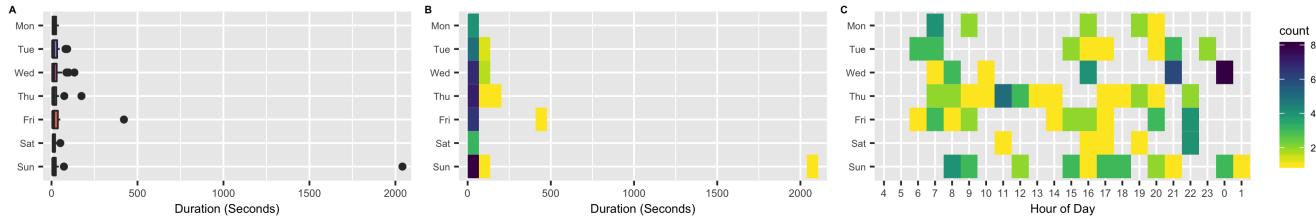


Figure 33: A caption

#### 4.0.0.15 Bathroom Sink Faucet - Hot

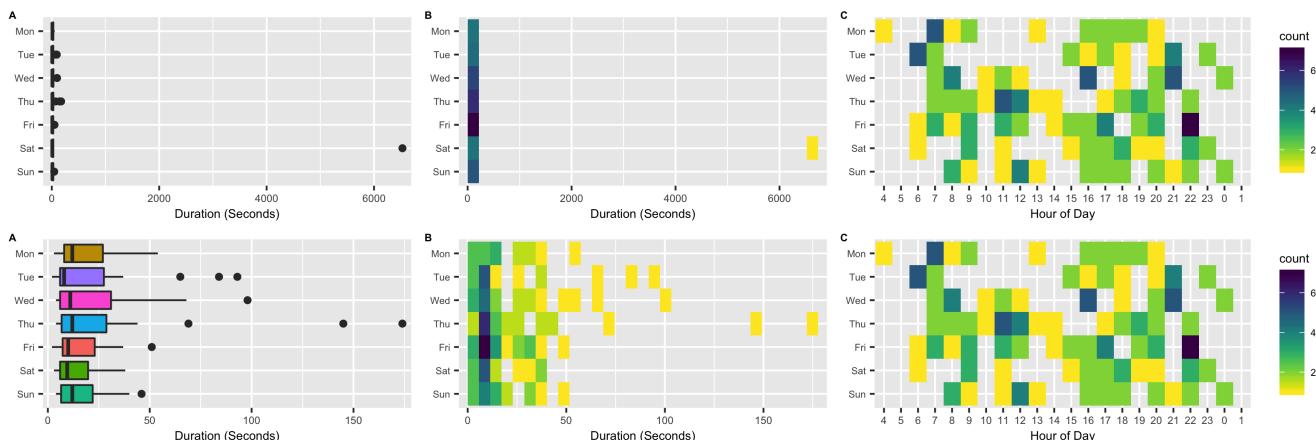


Figure 34: A caption

#### 4.0.0.16 Foyer Door (SubActNum 140)

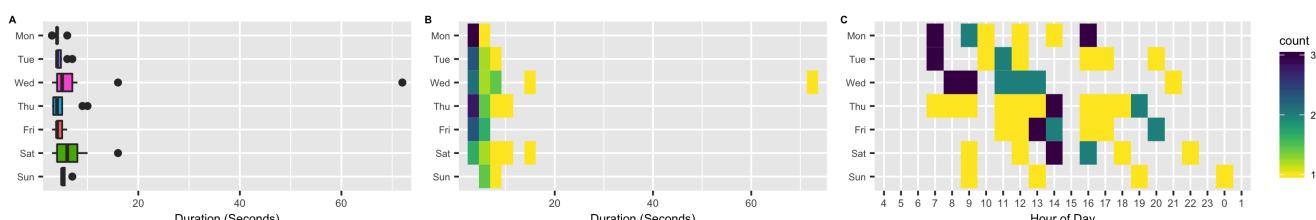


Figure 35: A caption

#### 4.0.0.17 Kitchen Freezer

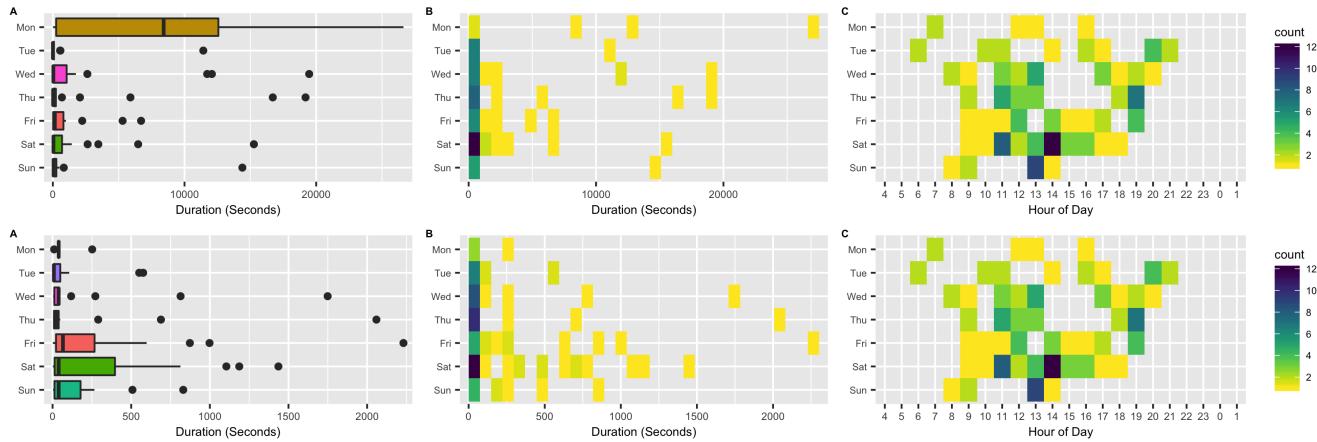


Figure 36: A caption

#### 4.0.0.18 Kitchen Burner

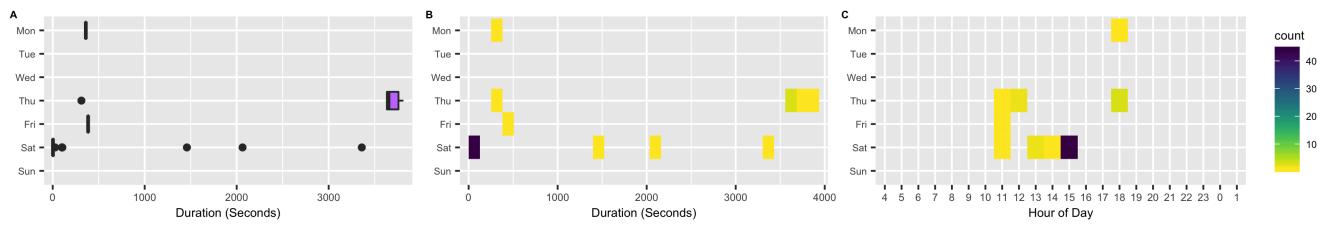


Figure 37: A caption

#### 4.0.0.19 Kitchen Lightswitch

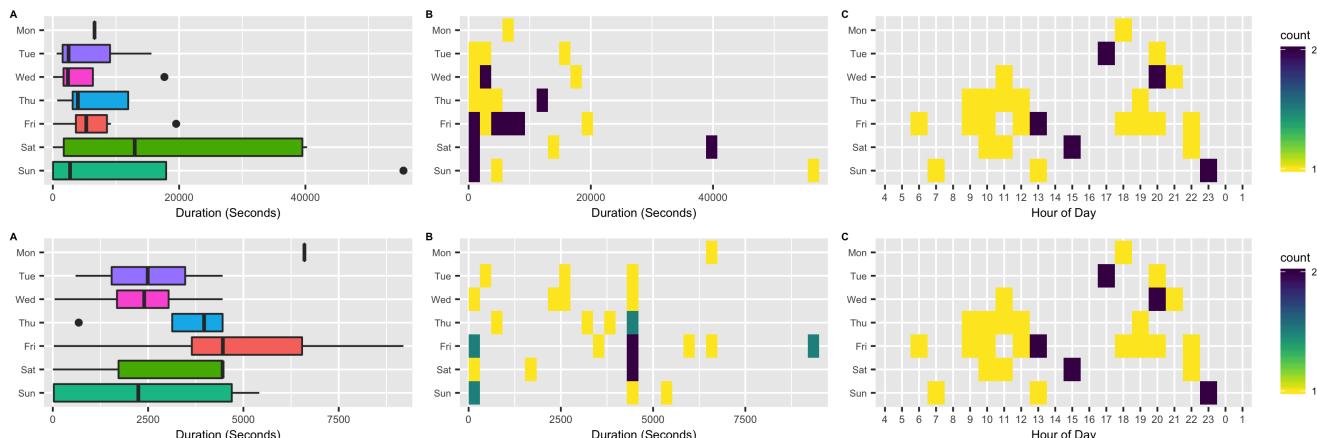


Figure 38: A caption

#### 4.0.0.20 Study Lightswitch (TYPO)

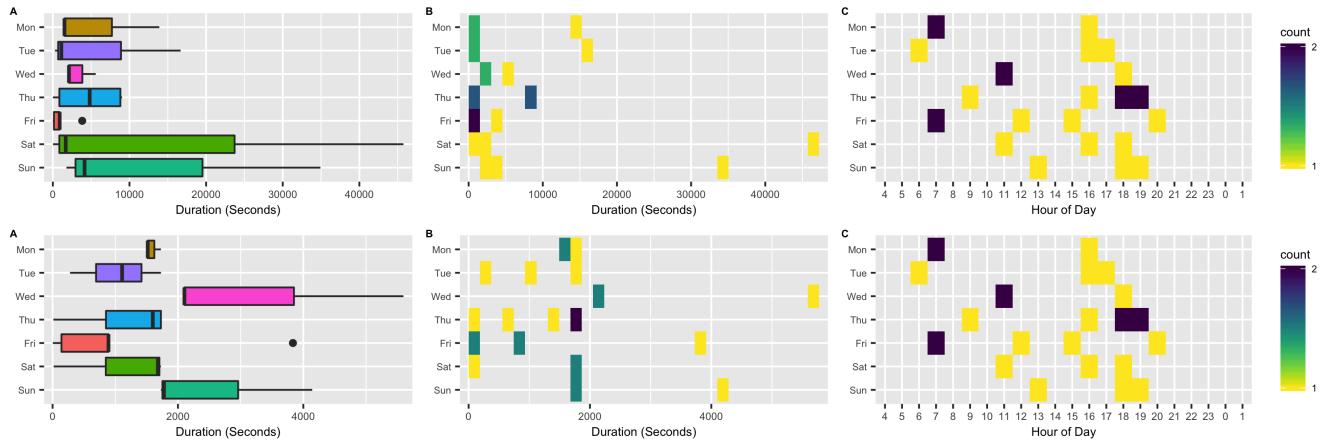


Figure 39: A caption

#### 4.0.0.21 Bathroom Door

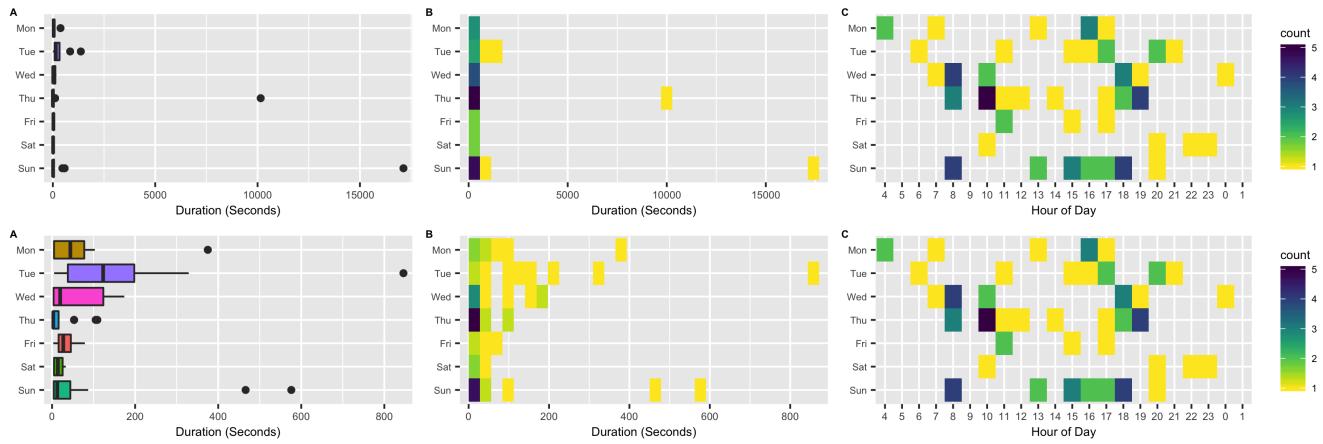


Figure 40: A caption

#### 4.0.0.22 Foyer Lightswitch

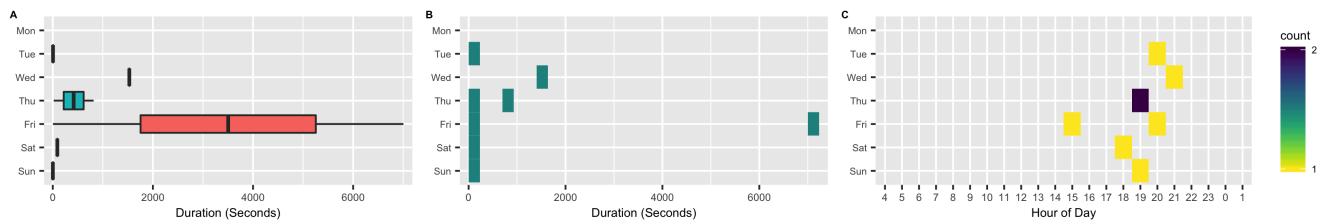


Figure 41: A caption

#### 4.0.0.23 Kitchen Toaster

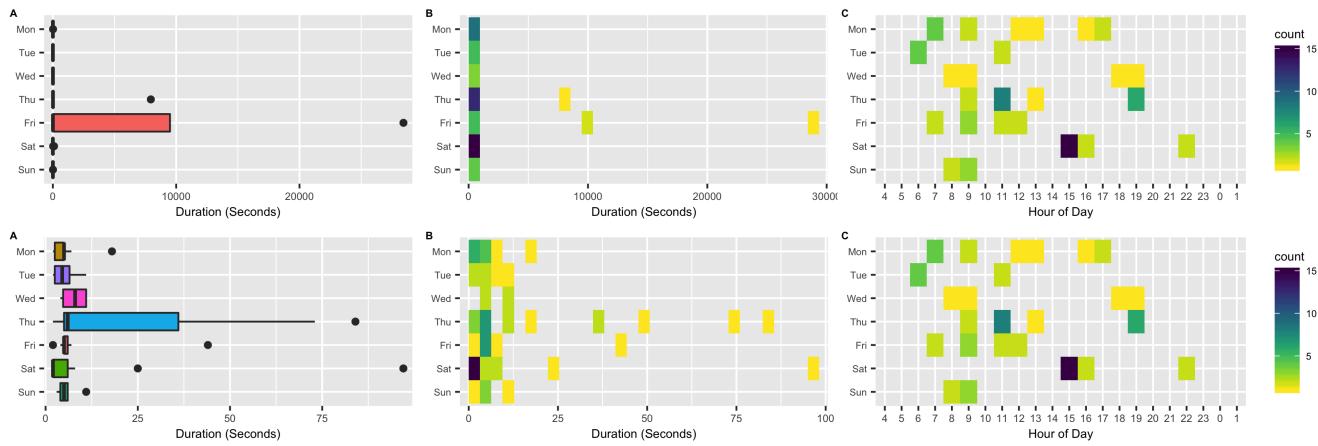


Figure 42: A caption

#### 4.0.0.24 Bathroom Exhaust Fan

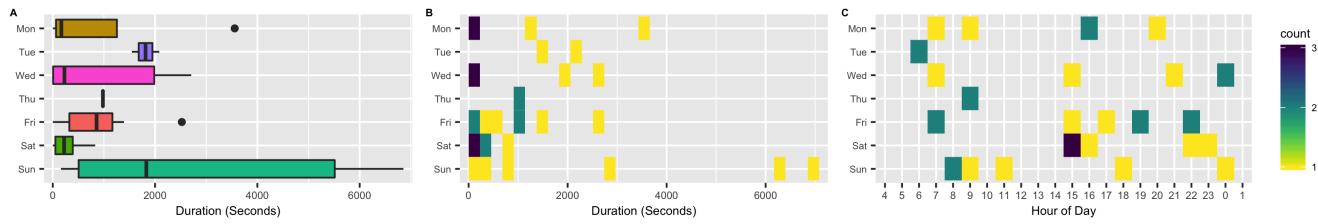


Figure 43: A caption

#### 4.0.0.25 Bedroom Lightswitch

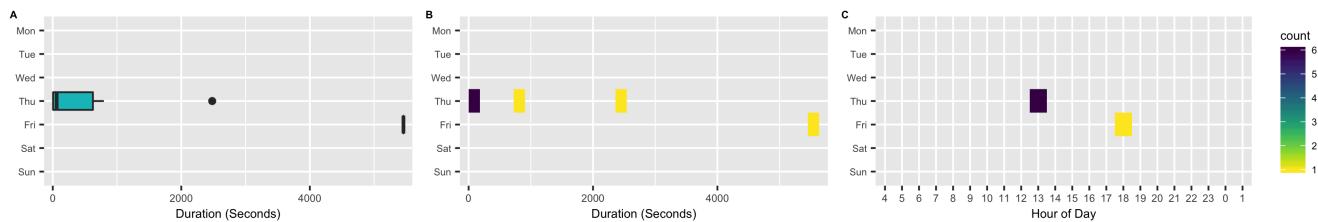


Figure 44: A caption

#### 4.0.0.26 Kitchen Oven

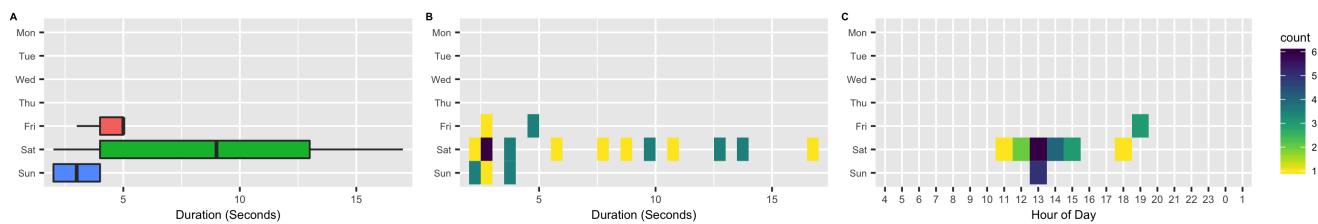


Figure 45: A caption

#### 4.0.0.27 Livingroom DVD

DROPPED - One value only \* 2

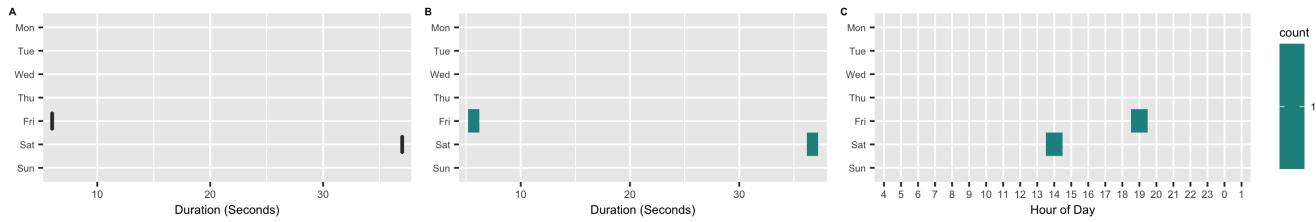


Figure 46: A caption

#### 4.0.0.28 Livingroom Lamp

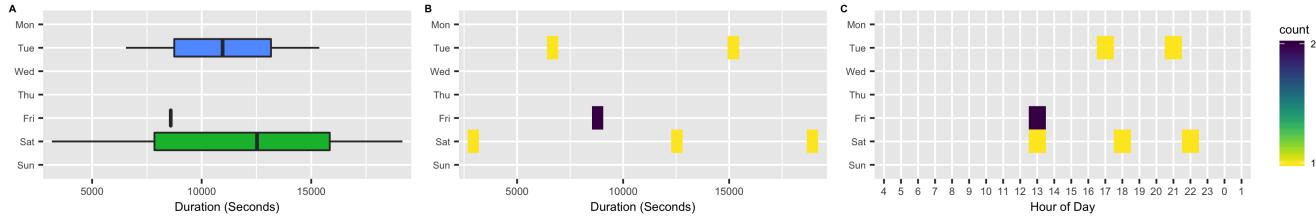


Figure 47: A caption

#### 4.0.0.29 Bedroom Jewelrybox

DROPPED

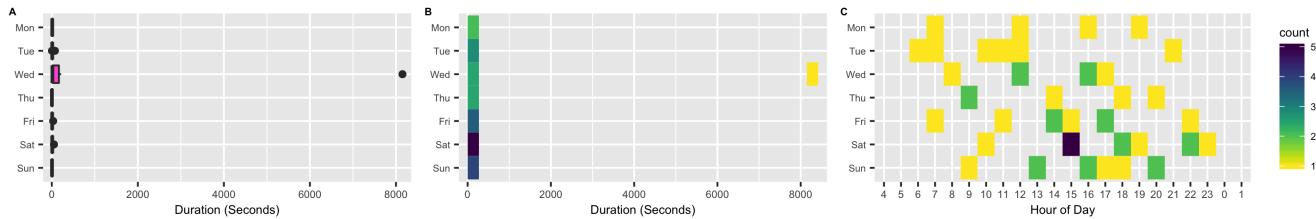


Figure 48: A caption

#### 4.0.0.30 Kitchen Washingmachine

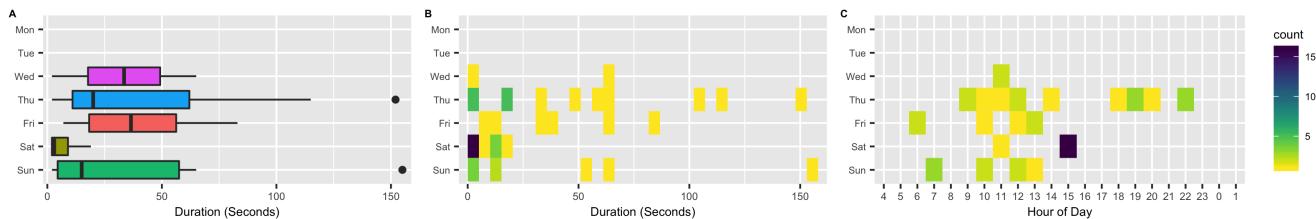


Figure 49: A caption

#### 4.0.0.31 Kitchen Laundry Dryer

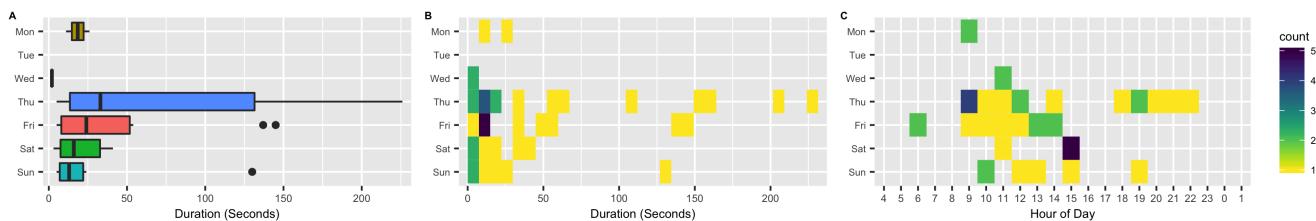


Figure 50: A caption

#### 4.0.0.32 Kitchen Garbage Disposal

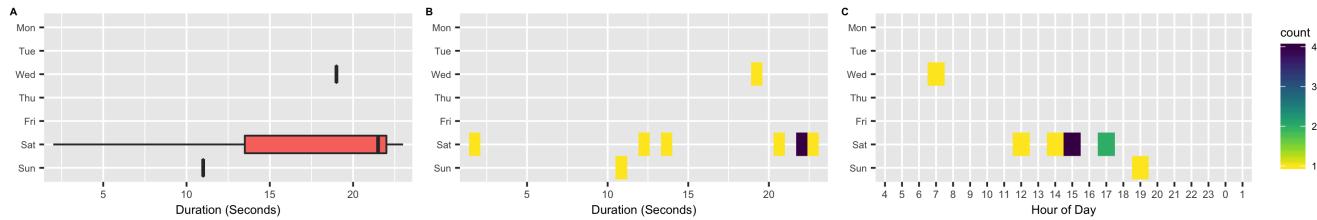


Figure 51: A caption

#### 4.0.0.33 Livingroom Lightswitch

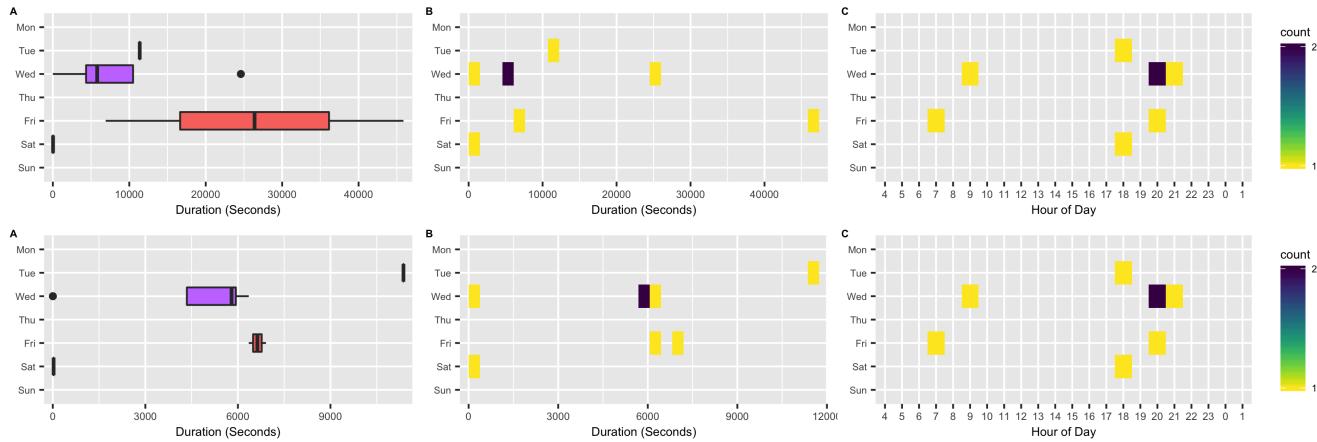


Figure 52: A caption

#### 4.0.0.34 Foyer Closet (SubActNum 81)

DROPPED

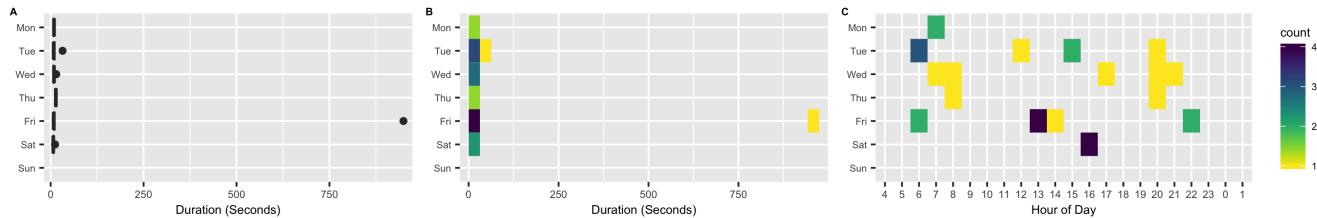


Figure 53: A caption

#### 4.0.0.35 Kitchen Cereal (SubActNum 145)

DROPPED

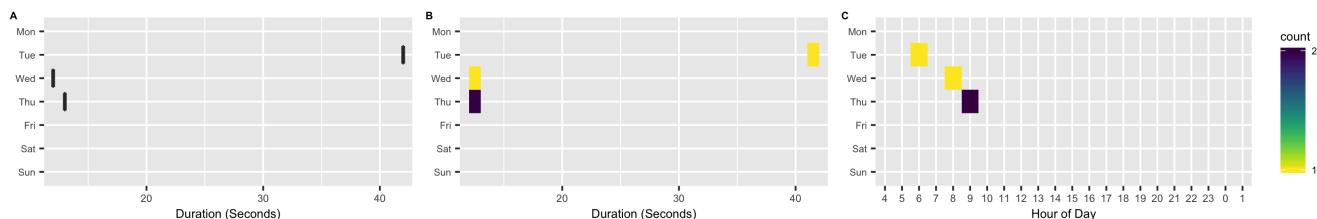


Figure 54: A caption

#### 4.0.0.36 Kitchen Containers (SubActNum 60)

DROPPED

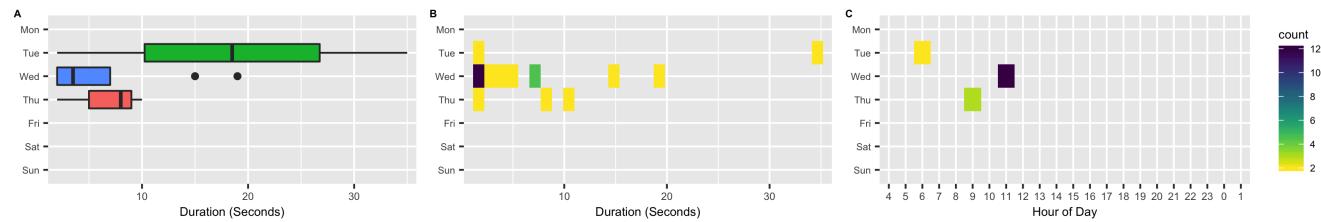


Figure 55: A caption

#### 4.0.0.37 Kitchen Coffee Machine

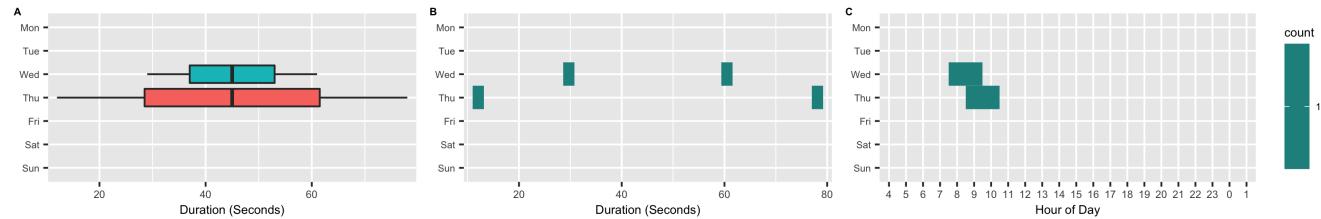


Figure 56: A caption

#### 4.0.0.38 Bedroom Lamp (SubActNum 64)

DROPPED

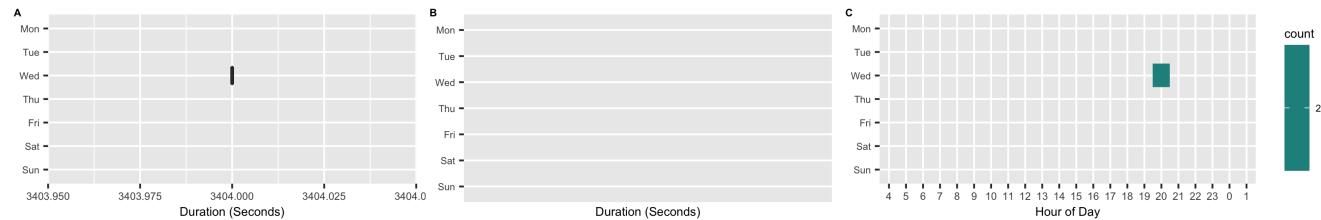


Figure 57: A caption

```
%run -i reqEnergy_containSpecialCharClean.py
```

Test for in 59 we see XYZ

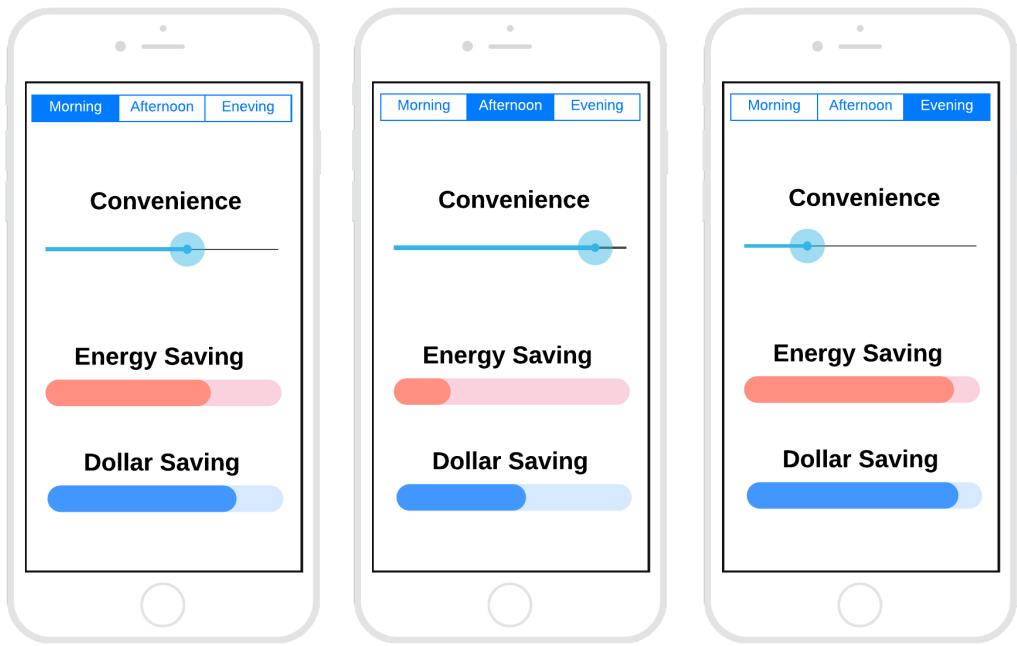


Figure 58: ADD TEXT

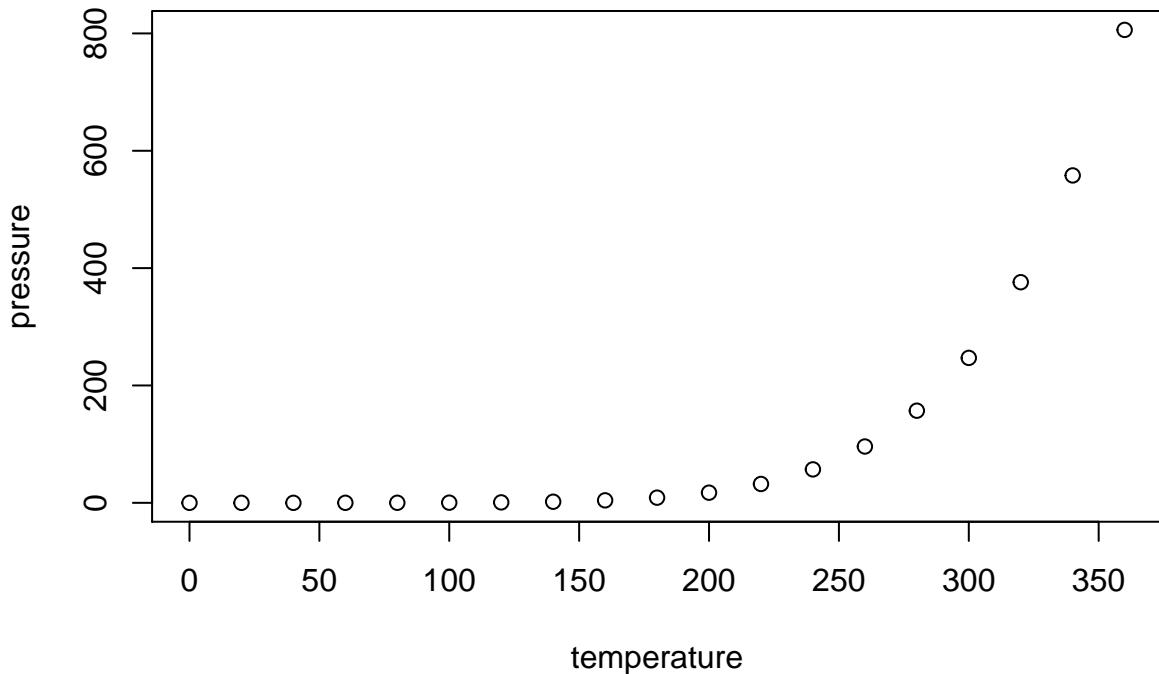


Figure 59: This is a caption

Reduce image border (not working???) <https://holtzy.github.io/Pimp-my-rmd/>

```
object <- [some kind of calculation]
Note that this; is not valid R code
```

#### 4.0.0.39 Algorithm 1

Text

---

**Algorithm 3:** While loop with If/Else condition

---

**Result:** Write here the result  
**Input :** Write here the input  
**Output:** Write here the output

```
1 while While condition do
2     instructions
3     if condition then
4         instructions1
5         instructions2
6     else
7         instructions3
8 end
9
```

---

## References

1. Press L (1993) Before the Altair: The history of personal computing. *Commun ACM* 36:27–33. <https://doi.org/10.1145/162685.162697>
2. Dutta S, Geiger T, Lanvin B, et al (2015) The global information technology report 2015: ICTs for inclusive growth
3. Information and communication technologies (ICTs) | Poverty Eradication. <https://www.un.org/development/desa/socialperspectiveondevelopment/issues/information-and-communication-technologies-icts.html>. Accessed 5 Nov 2019
4. Data Never Sleeps 5.0 | Domo. [https://www.domo.com/learn/data-never-sleeps-5?aid=ogsm072517\\_1&sf100871281=1](https://www.domo.com/learn/data-never-sleeps-5?aid=ogsm072517_1&sf100871281=1). Accessed 5 Nov 2019
5. Cloud computing | Definition of Cloud computing at Dictionary.Com. <https://www.dictionary.com/browse/cloud-computing>. Accessed 14 Nov 2019
6. Daniele Miorandi FDP Sabrina Sicari (2012) Internet of things: Vision, applications and research challenges. *Ad Hoc Networks* 10:1497–1516. <https://doi.org/10.1016/j.adhoc.2012.02.016>
7. Bing Huang AGN Athman Bouguettaya (2018) Service-Oriented Computing, 16th International Conference, ICSOC 2018, Hangzhou, China, November 12–15, 2018, Proceedings. 660–678. [https://doi.org/10.1007/978-3-030-03596-9\\_48](https://doi.org/10.1007/978-3-030-03596-9_48)
8. Luigi Atzori GM Antonio Iera (2010) The Internet of Things: A survey. *Computer Networks* 54:2787–2805. <https://doi.org/10.1016/j.comnet.2010.05.010>
9. Intergovernmental Panel on Climate Change (2018) Global warming of 1.5°C
10. Energy demand by sector | Energy economics | Home. In: BP global. <https://www.bp.com/en/global/corporate/energy-economics/energy-outlook/demand-by-sector.html>. Accessed 14 Nov 2019
11. Consumption. <https://www.iea.org/statistics/kwes/consumption/>. Accessed 14 Nov 2019
12. Haseeb A, Xia E, Saud S, et al (2019) Does information and communication technologies improve environmental quality in the era of globalization? An empirical analysis. *Environ Sci Pollut Res* 26:8594–8608. <https://doi.org/10.1007/s11356-019-04296-x>
13. Horne C, Darras B, Bean E, et al (2015) Privacy, technology, and norms: The case of Smart Meters. *Social Science Research* 51:64–76. <https://doi.org/10.1016/j.ssresearch.2014.12.003>
14. Arel-Bundock V, Enevoldsen N, Yetman C (2018) Countrycode: An r package to convert country names and country codes. *Journal of Open Source Software* 3:848
15. Wilke CO (2018) Cowplot: Streamlined plot theme and plot annotations for 'ggplot2'

16. Francois R (2017) Bibtex: Bibtex parser
17. Wickham H, François R, Henry L, Müller K (2019) Dplyr: A grammar of data manipulation
18. Attali D, Baker C (2018) GgExtra: Add marginal histograms to 'ggplot2', and more 'ggplot2' enhancements
19. Wickham H (2017) Tidyverse: Easily install and load the 'tidyverse'
20. Wickham H (2016) Ggplot2: Elegant graphics for data analysis. Springer-Verlag New York
21. Wilke CO (2018) Ggridges: Ridgeline plots in 'ggplot2'
22. Zhu H (2018) KableExtra: Construct complex table with 'kable' and pipe syntax
23. Xie Y (2019) Knitr: A general-purpose package for dynamic report generation in r
24. Grolemund G, Wickham H (2011) Dates and times made easy with lubridate. Journal of Statistical Software 40:1–25
25. Wickham H, Hester J, Francois R (2017) Readr: Read rectangular text data
26. Wickham H, Henry L (2018) Tidyr: Easily tidy data with 'spread()' and 'gather()' functions
27. Garnier S (2018) Viridis: Default color maps from 'matplotlib'
28. Ushey K, Allaire J, Tang Y (2019) Reticulate: Interface to 'python'
29. Xiao N (2017) Ggsci: Scientific journal and sci-fi themed color palettes for 'ggplot2'
30. Wickham H (2018) Scales: Scale functions for visualization