

CS4001 Report

Fabian Miiro

Group project, in collaboration with:

Aran Sena

Sebastian Ruder

Contents

Introduction.....	4
Background.....	4
Sleep.....	4
Usage of fuzzy logic	4
TSK vs. Mamdani	4
Simulation / Model	4
Overall design structure	4
Inputs.....	5
Last meal or drink	5
Slept day before	5
Activity during the day	5
Current sleep cycle.....	6
Time to sleep.....	6
Quality.....	6
Commute time.....	6
Rules	6
Defuzzification methods	6
Outputs.....	7
Sleep Duration	7
Easiness of Falling Asleep.....	7
Volume of alarm	7
Strength of Coffee.....	7
Rules	7
Results & Analysis.....	8
Results.....	8
Evaluation	8
Conclusions.....	9
References.....	9
Internet	9
Lectures	9
Others	9
Appendix.....	10
Creation of subsystem with sleep duration as output.....	10

Surfaces	10
Easiness of falling asleep	10
Easiness of waking up.....	11
Sleep Duration	12

Introduction

As a part of the course Fuzzy Logic & Control Systems we have created a control system that uses fuzzy logic. This control system is based upon the notion of an alarm but has also been thought about to be extended to a fully featured “wakeup system”. The difference is that the wakeup system helps you all the way until you’re fully awake. This could include things as the amount of coffee you get, the length of the shower and so on.

We did it because the idea was interesting enough and something where we thought we could incorporate what we had learned about fuzzy systems.

The Fuzzy system we created outputs favorable results that we think gives us a certain degree of confidence that it was worth using fuzzy logic. It was definitely worth using fuzzy logic because of the learning process.

Background

Sleep

Sleep is something we all do, we all need to do it, but we really don’t know why we do it. It is a really fuzzy concept that is a good match for a fuzzy system.

Our sleep is affected by numerous of things. Most of these are highly individual, such as how much we need to sleep or how much movement every day is normal. While some are a little bit more universal, for example is it considered that everyone should eat at least three hours before bedtime to be able to fall asleep more easily.¹

The concept behind this system is that it should be used in a ‘connected’ (IoT) home. This wakeup system would be integrated with the rest of the house and be able to talk to for example the coffee machine, lights, and the car to make sure everything works to make the mornings as pleasant as possible, most of the data used would also have its origins

from IoT devices around the house or wearables connected to the user.

Usage of fuzzy logic

This problem is a good match for a fuzzy system because of the fuzzy nature of the problem (as described in previous chapter). There are no crisp border values that can be used on such a personal system and every day is different from a systems point of view.

TSK vs. Mamdani

We have chosen to use the Mamdani system for a couple of reasons. Firstly it’s the first system we got to play with and go through during lectures so it’s now the one we feel most comfortable with. On the other hand, the strong point for TSK and why it’s mainly used is its mathematical simplicity which makes it extremely fast.² Because our system is designed to work while you sleep we will have several hours for computation. Therefore we don’t really care about how quick the method is. TSK is excellent for real-time systems. But for us, the way we model the system and the preciseness is more important.

Simulation / Model

Overall design structure

This wakeup system is designed as three subsystems. One of these subsystems calculates how easy it is to fall asleep (for input in another system), another how long time the user should sleep (as output to set the alarm clock), and the third calculates how the user should be awoken (volume of alarm and strength in coffee). One of the reasons why we have three subsystems is to reduce the number of rules needed. Another to give each member of the group its own subset to be responsible for. We started by determining the important outputs we needed, these were identified as being “Wake up time”, “Alarm Volume” and “Coffee Strength” (even though we only had one coffee drinker in the group). After this inputs were selected and divided

¹ Joy Baur – How Food Affects Sleep – Collected: 2014-11-20 - Source: <http://www.joybauer.com/insomnia/how-food-affects-sleep.aspx>

² Lecture: Fuzzy Control: Mamdani & Takagi-Sugeno Controllers – Teacher: Khurshid Ahmad, Professor of Computer Science, Department of Computer Science, Trinity College – November 2014

upon the different systems so they would have similar number of inputs and outputs. As seen in figure 1

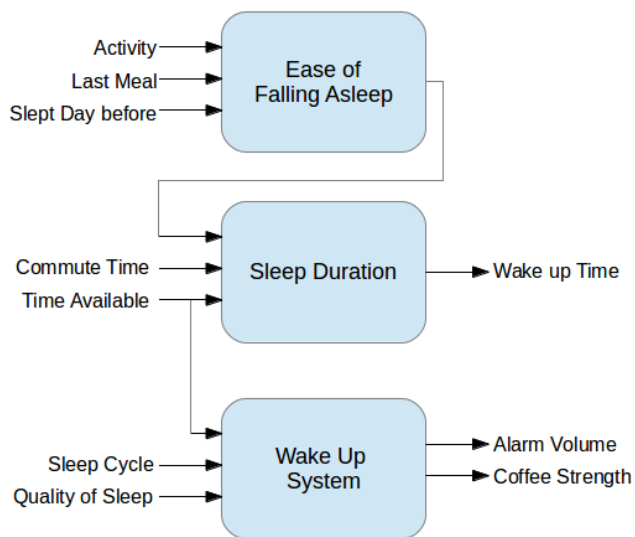


Figure 1

Inputs

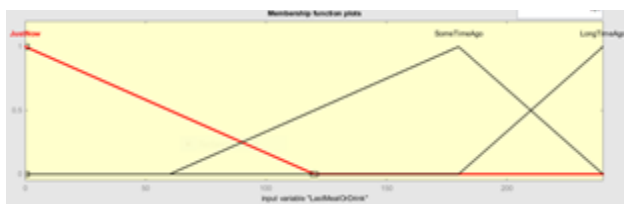
Last meal or drink

Linguistic values: Just now, some time ago, long time ago

Scale: 0-360

Description: The input value is the number of minutes since the last meal or drink was consumed. The membership function is based on data from Joy Bauer's article "How Food Affects Your Sleep"³

MF:



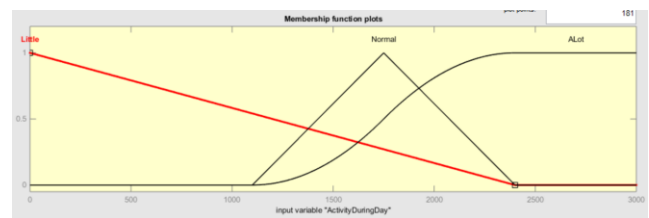
Slept day before

Linguistic values: Little, Normal, A lot

Scale: 0 – 720

Description: The number of minutes that the 'user' slept the previous day. This data could possibly be from the systems logs of previous days, user input or some activity tracker. This membership function is based upon the group members own views of how much sleep is necessary for the different levels.

MF



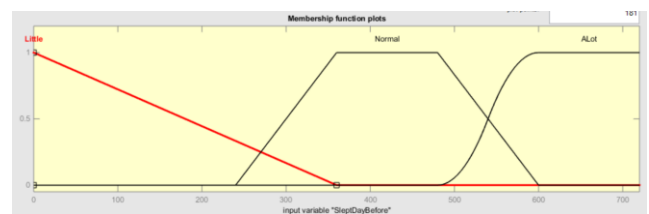
Activity during the day

Linguistic values: Little, Nromal, A Lot

Scale: 0 – 3000

Description: How many calories were burnt during the day. (data from wearables or phone). The membership function here is based upon an article by "Michèle Turcotte"⁴, which outlines how much energy we should consume during a day. And to maintain the body we have, calories burnt should be the same as calories consumed.⁵

MF



³ Joy Baur – How Food Affects Sleep – Collected: 2014-11-20 - Source: <http://www.joybauer.com/insomnia/how-food-affects-sleep.aspx>

⁴ Michèle Tucotte – Calories: What's an ideal daily intake? - Collected: 2014-11-24- Source: <http://www.thedietchannel.com/AskTheExpert/dieting-weightloss-obesity/Calories-Whats-an-ideal-daily-intake.htm>

⁵ A caolirie counter – "Daily Calorie Intake – Why Calories In vs Calories Out Is The Key To Your Diet" – Collected: 2015-01-08 – Source: <http://www.acaloriecounter.com/diet/daily-calorie-intake-calories-in-vs-calories-out/>

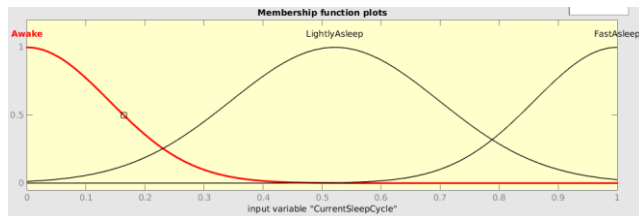
Current sleep cycle

Linguistic values: Awake, Lightly asleep, Fast asleep

Scale: 0 – 1

Description: In which sleep cycle is the user currently?
This data can be obtained using a number of methods. For once it could be extracted using some kind of wearable that could track heartrate and movement. But since sleep cycles are often very similar it could be calculated based upon when the user fell asleep.

MF



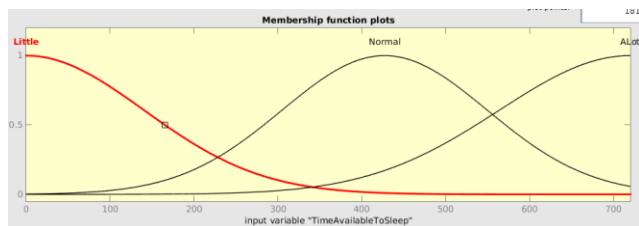
Time to sleep

Linguistic values: Very little, Little, Avarage, More, Lots

Scale: 0 – 600

Description: In minutes, how long time there is available to sleep. This should be the number of minutes from that the user goes to bed to when the first meeting is about to start. This value is used as an absolute maximum in the system. A user can NEVER sleep more than the available time, what the system does is reduce this theoretical maximum so that the user will have time to eat and commute while still getting the proper amount of sleep.

MF



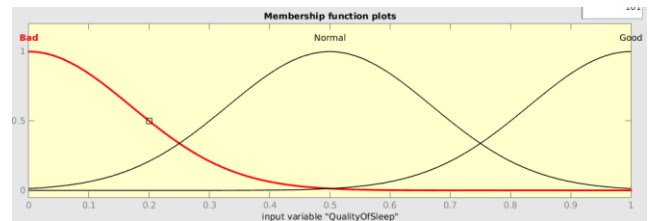
Quality

Linguistic values: Bad, Normal, Good

Scale: 0 – 1

Description: What is the quality of the sleep? (Calculated by an external system). This is also a very fuzzy thing and should in real life be calculated by another fuzzy system, but since this isn't real life we just assume a value for how good the sleep is.

MF



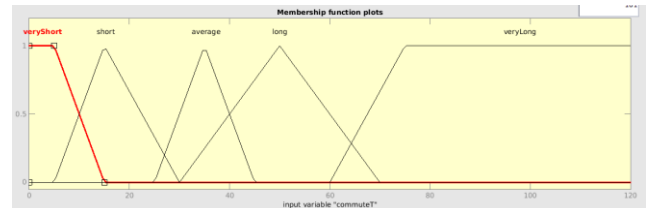
Commute time

Linguistic values: Very short, short, average, long, very long

Scale: 0 – 120

Description: This is the commute time in minutes, aka how long it takes to get from door to door.

MF



Rules

See appendix for full rule list. But here are some interesting rules:

TODO!

Defuzzification methods

We used the Centroid method to defuzzify the output from the different systems. No aggregation was required since

all outputs where either used as input to a new system or as the final output.

Outputs

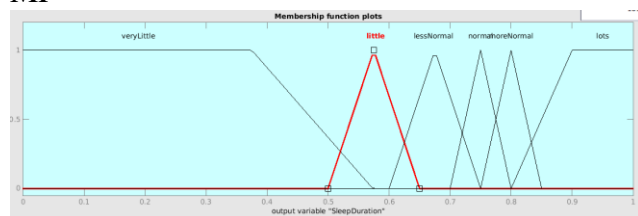
Sleep Duration

Linguistic values: Very little, little, less normal, normal, more normal, lots

Scale: 0 – 1

Description: This output is a multiplier that should get multiplied to the time to sleep to get the total minutes that the user should sleep to still be able to have time to commute, get dressed and eat breakfast.

MF



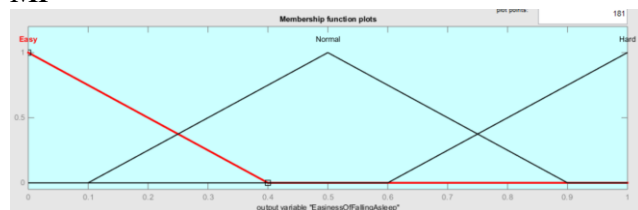
Easiness of Falling Asleep

Linguistic values: Easy, Normal, Hard

Scale: 0 – 1

Description: This output tries to explain how easy it was to fall asleep. This will affect how soon the user fell asleep and therefore also how much time the user will be able to sleep until it's time to wake up.

MF



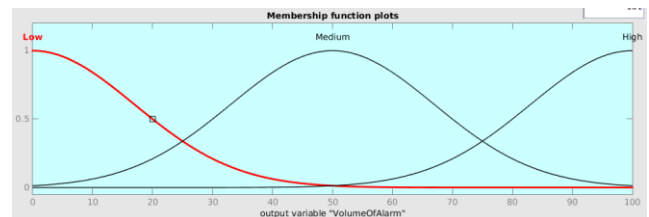
Volume of alarm

Linguistic values: Low, Medium, High

Scale: 0 – 100

Description: This output is basically the volume knob on the alarm clock. 0 is the same as mute, 100 is the same as "ouch".

MF



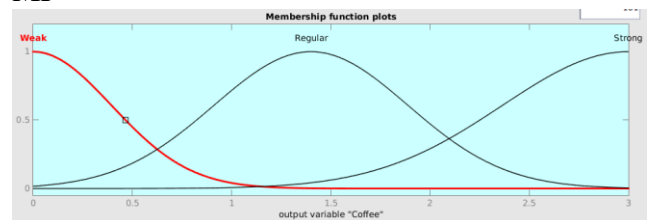
Strength of Coffee

Linguistic values: Weak, Regular, Strong

Scale: 0 – 3

Description: This output describes how many shots of coffee the user should be given. Is used to make sure that we give the user as little coffee as possible but still let them have that feeling of alertness and energy that coffee can help with.

MF



Rules

Some key rules for every subsystem is listen in the appendix.

Some of our systems are a bit different to each other, for example the system with output value "Sleep Duration" is designed and built a bit differently, simply because it was designed by a team member that wanted to try and design and create it in a different way. This different way is that instead of using all inputs for a single output the inputs are used two and two, this creates a subsystem with the same amounts of input, but a different number of rules (54 compared to 27 in the others). For the creator of this system it

was apparently easier to work with and understand, but that is seen by me as a preference depending on the person designing it.

Results & Analysis

Results

Testing involved the creation of some imaginary test users, each with a particular type of persona.

The 3 users lifestyles could be described as:

An average student (AS), A “work-hard-play-hard” (WHPH) business person and finally a sedate commuter (SC). The full results table is shown in the appendix, but let us discuss the Work-Hard-Play-Hard lifestyle case for an example of the results. This case attempts to simulate the type of user who could be described as a workaholic, centering their lives around their job by living close to the office and working late into the night before an early start the next day.

For subsystem one, our inputs are:

Last meal: 15 minutes- a late meal after work
Slept day before: 4 hours
Activity during day: 3000 calories – a lot of walking around

This results in an output “easiness of falling asleep” of 0.166 (the lower, the easier). This intuitively makes sense, it is easy to imagine how working at this pace could result in falling asleep very quickly when you eventually made it to bed.

For subsystem two, our inputs are:

Commute time: 20 minutes – living close to work
Ease of Sleep: This input is the output of subsystem 1
Time Available to Sleep: Assuming a 7AM start, and a bed time of 2AM, this person has a theoretical maximum of 5 hours (300 minutes) available to sleep (ignoring other activities such as commuting, getting dressed, etc.)

This results in an output “Sleep Duration Modifier” of 0.867, meaning they can only sleep for about 87% of the theoretical maximum time available. This indicates they will get roughly 4 hours and 20 minutes of sleep, leaving

them with 20 minutes to get ready before their 20 minute commute. Intuitively this result appears to make sense

For subsystem three, our inputs are:

Current sleep cycle: 0.9 – Due to exhaustion, it is assumed the body will be in a deep sleep

Time available to sleep: As before, we take the theoretical maximum of 300 minutes

Quality of sleep: 0.4 – Somewhat normal due to exhaustion, however not great due to lifestyle.

This results in an output alarm volume of 83 (Medium-High), and a coffee strength of 2.34 shots of espresso. Again, this intuitively makes sense given their deep sleep state, and the short low quality rest they are getting.

One issue encountered was that in an attempt to limit the number of rules implemented, we limited the number of fuzzy subsets implemented for each linguistic primary term, which in some cases resulted in un-smooth responses, or overcompensation for changes in the input.

Evaluation

The results are actually surprisingly good. Provided input data and examining the output will in most cases provide a reasonable wakeup time. However we’ve had to jump through some loops to make it all work, distort some amplifiers to make it work more as the real world. Something that would have been a better solution would have been to instead of having the commute time as input just subtract the commute time from the output time. Since the commute time isn’t generally fuzzy but a crisp value based on how long it takes for the bus/car to get to the meeting. (We could argue it’s fuzzy because a ‘long’ commute can be different things for different people but still be different time, but generally for me I can count on having a crisp value as my commute time, and in this case subtracting a time is much more precise than trying to create a multiplier that should be used on another time)

Since our output “easiness of falling asleep” is an amplifier being used as a percentage value we have a very clear and consistent output membership curve for that one.

We decided to use the value “time to sleep” instead of “time of alarm” after several discussions about the advantages with the two different kinds of values. We settled with time to sleep because it’s more easy work with and

doesn't require any extra steps to figure out if that's a long nights sleep or not.

Conclusions

Was a fuzzy solution justifiable?

Yes, a fuzzy solution was in this case definitely justifiable because of the fuzzy nature of the problem. A crisp system would have problems with different people and would therefore most probably be designed for a "normal" person. Since our system is fuzzy it will still be able to handle people that are outside the normal bounds in let's say sleep pattern or calories burnt per day.

However there are some areas that could have changed or been improved upon by making it into a hybrid system. Where most was done using fuzzy systems but some in-between calculations were carried out using normal calculations.

But overall, I think it was a successful experiment trying out fuzzy stuff. It has shown to me the usability of fuzziness and before I create my next programmatically solution, I will definitely consider creating a fuzzy solution.

References

Internet

[1] Joy Baur – How Food Affects Sleep – Collected: 2014-11-20 - Source: <http://www.joybauer.com/insomnia/how-food-affects-sleep.aspx>

[3] Joy Baur – How Food Affects Sleep – Collected: 2014-11-20 - Source: <http://www.joybauer.com/insomnia/how-food-affects-sleep.aspx>

[4] Michéle Tucotte – Calories: What's an ideal daily intake? - Collect-ed: 2014-11-24- Source: <http://www.thedietchannel.com/AskTheExpert/dieting-weightloss-obesity/Calories-Whats-an-ideal-daily-intake.htm>

[5] A caolirie counter – “Daily Calorie Intake – Why Calories In vs Calories Out Is The Key To Your Diet” – Collected: 2015-01-08 – Source: <http://www.acaloriecounter.com/diet/daily-calorie-intake-calories-in-vs-calories-out/>

Lectures

[2] Lecture: Fuzzy Control: Mamdani & Takagi-Sugeno Controllers – Teacher: Khurshid Ahmad, Professor of Computer Science, De-partment of Computer Science, Trinity College – November 2014

Others

Words of inspiration from Khurshid Ahmad.

Appendix

Creation of subsystem with sleep duration as output

		Time Available to Sleep				
		veryLittle	little	average	more	lots
Commute Time	very short	lessNormal	normal	moreNormal	lots	lots
	short	little	normal	moreNormal	lots	lots
	average	veryLittle	lessNormal	normal	moreNormal	lots
	long	veryLittle	little	lessNormal	normal	morenormal
	very long	veryLittle	veryLittle	little	lessNormal	normal

		Time available to Sleep				
		veryLittle	little	average	more	lots
Ease of Falling Asleep	hard	veryLittle	veryLittle	little	lessNormal	Normal
	normal	little	lessNormal	Normal	moreNormal	lots
	easy	lessNormal	Normal	MoreNormal	lots	lots

		Commute Time				
		very short	short	average	long	very long
Ease of Falling Asleep	hard	moreNormal	normal	lessNormal	little	veryLittle
	normal	lots	moreNormal	normal	lessNormal	little
	easy	lots	lots	moreNormal	normal	lessNormal

Surfaces

Easiness of falling asleep

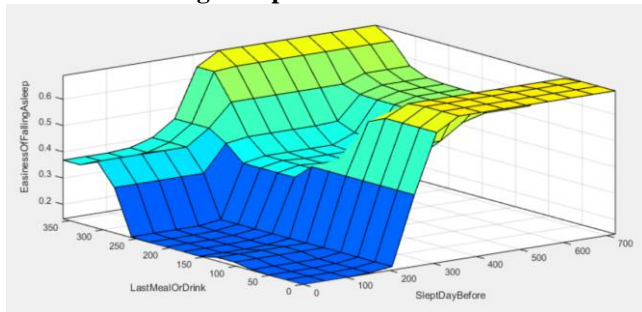


Figure 2 Last meal or drink compared to slept day before

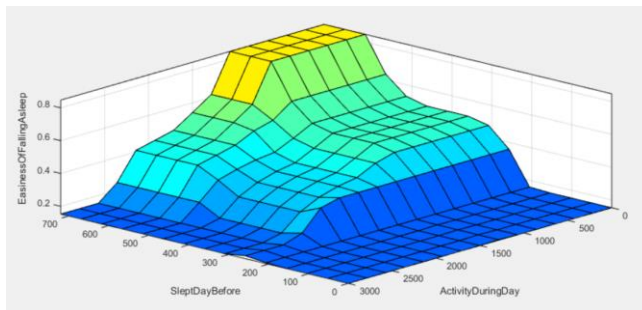


Figure 3 Slept day before compared to activity during day

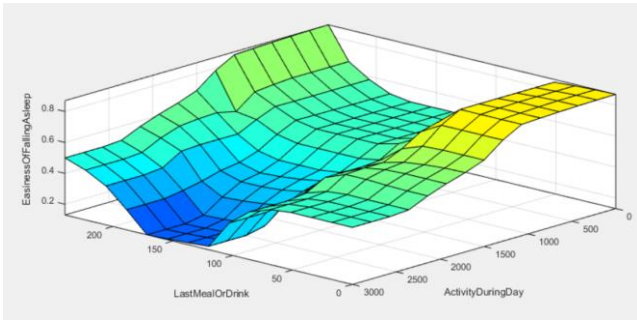


Figure 4 Last meal or drink compared to activity during day

Easiness of waking up

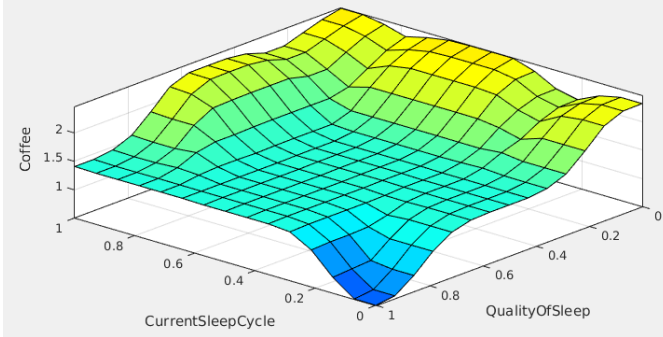


Figure 5 Shows how coffee strength is connected to current sleep cycle and quality of sleep

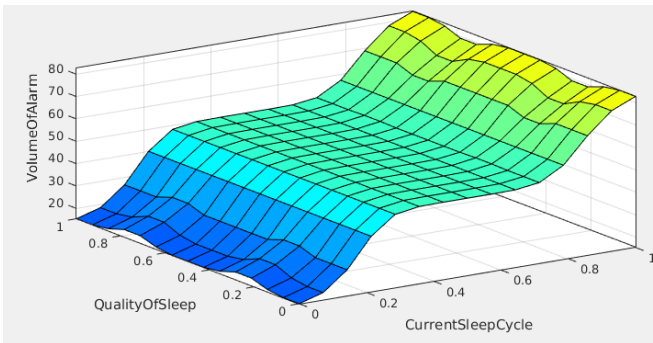


Figure 6 Shows how volume of alarm is affected by quality of sleep and current sleep cycle

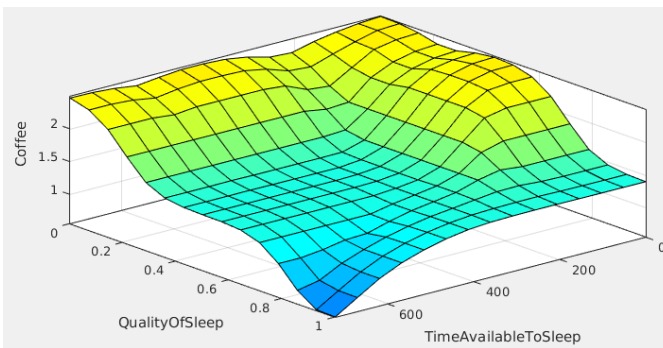


Figure 7 Shows how coffee strength is affected by Quality of sleep and Time available to sleep

Sleep Duration

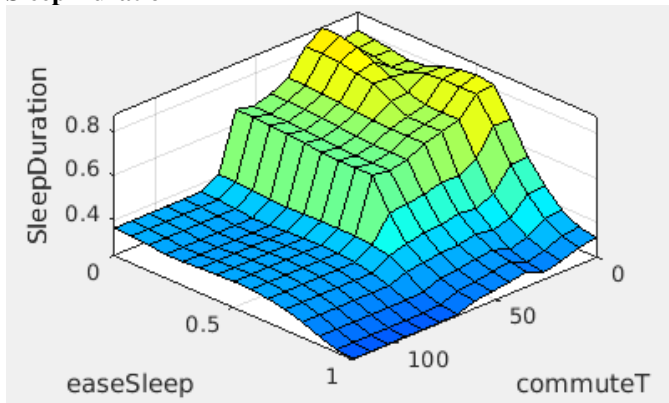


Figure 8 The affect between ease of sleep and commute time on sleep duration

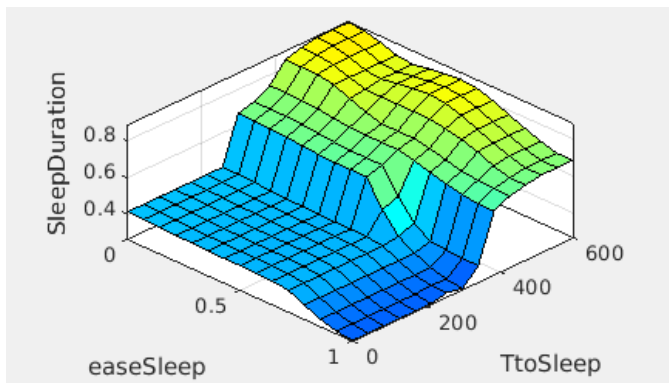


Figure 9 The effects on sleep duration with regards to easiness to fall asleep and time to sleep

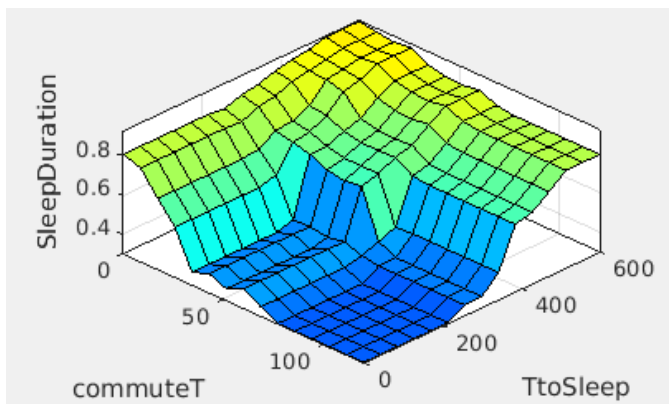


Figure 10 The effects on sleep duration with regards to commute time and time to sleep

Important rules

Important rules for sleep duration subsystem

---rules that heavily influence more sleep---

We can sleep longer if we know the commute will be short and we have a decent amount of time to sleep:

if (commuteT is veryShort) and (TtoSleep is average) then (SleepDuration is lots)

if (commuteT is short) and (TtoSleep is more/lots) then (SleepDuration is lots)

if (commuteT is average) and (TtoSleep is lots) then (SleepDuration is lots)

If we can get to sleep reasonably well, and we have a decent amount of time to sleep, we sleep more:

if (easeSleep is normal) and (TtoSleep is lots) then (SleepDuration is lots)

if (easeSleep is easy) and (TtoSleep is more/lots) then (SleepDuration is lots)

If our commute is short, and we get to sleep very easily, we can sleep more of the time available:

if (commuteT is veryShort) and (easeSleep is normal/easy) then (SleepDuration is lots)

if (commuteT is short) and (easeSleep is easy) then (SleepDuration is lots)

---rules that heavily influence less sleep---

If we know we have long to travel, and little time to sleep, we can't use as much of the available time:

if (commuteT is average/long/verylong) and (TtoSleep is veryLittle) then (SleepDuration is veryLittle)

if (commuteT is verylong) and (TtoSleep is Little) then (SleepDuration is veryLittle)

If it is hard to sleep, and we have little time to sleep, then we can't sleep for as long:

if (easeSleep is hard) and (TtoSleep is little/veryLittle) then (SleepDuration is veryLittle)

If the commute is long, and it's hard to fall asleep, then we can't sleep for as long:

if (commuteT is verylong) and (easeSleep is hard) then (SleepDuration is veryLittle)

Rules for easiness of falling asleep system

---rules that heavily influence easiness of falling asleep---

We will more easily fall asleep if we have slept little the day before (all rules except one that include slept little day before have easy as output).

If (LastMealOrDrink is SomeTimeAgo) and (SleptDayBefore is Little) and (ActivityDuringDay is ALot) then (EasinessOfFallingAsleep is Easy)

If (LastMealOrDrink is SomeTimeAgo) and (SleptDayBefore is Little) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Easy)

If (LastMealOrDrink is SomeTimeAgo) and (SleptDayBefore is Little) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Easy)

If (LastMealOrDrink is JustNow) and (SleptDayBefore is Little) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Easy)

If (LastMealOrDrink is JustNow) and (SleptDayBefore is Little) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Easy)

If (LastMealOrDrink is JustNow) and (SleptDayBefore is Little) and (ActivityDuringDay is ALot) then (EasinessOfFallingAsleep is Easy)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is Little) and (ActivityDuringDay is ALot) then (EasinessOfFallingAsleep is Easy)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is Little) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Easy)

Normality also leads to easily falling asleep

If (LastMealOrDrink is SomeTimeAgo) and (SleptDayBefore is Normal) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Easy)

---rules that shows it's hard to fall asleep ---

Almost half of the rules (4/9) that results in it being hard to fall asleep includes that we have eaten a long time ago

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is ALot) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is ALot) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is Normal) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is Normal) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Hard)

Off all the 'hard to fall asleep rules' 77% of them got atleast two rules at thier peakes (Little | ALot | LongTimeAgo | JustNow).

If (LastMealOrDrink is SomeTimeAgo) and (SleptDayBefore is ALot) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is JustNow) and (SleptDayBefore is ALot) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is JustNow) and (SleptDayBefore is ALot) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is JustNow) and (SleptDayBefore is Normal) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is Normal) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is ALot) and (ActivityDuringDay is Normal) then (EasinessOfFallingAsleep is Hard)

If (LastMealOrDrink is LongTimeAgo) and (SleptDayBefore is ALot) and (ActivityDuringDay is Little) then (EasinessOfFallingAsleep is Hard)

Rules for subsystem that outputs volume of alarm and coffee strength

—Example of Rules which heavily influence stronger Coffee—

Assumptions: Current sleep cycle is awake; thus volume of alarm can be low. As the person hasn't slept a lot and the sleep he had was only of normal quality, coffee should be strong to make the person feel more awake and alert.

If Current sleep cycle is Awake and Time available to sleep is Little and Quality of sleep is normal, then Volume of alarm is Low and Coffee is Strong.

Assumptions: Lightly asleep, so alarm should be medium. User has slept alot, but quality of sleep was bad indicates user isn't fully awake on waking up, means coffee should be strong to make user feel more awake.

If Current sleep cycle is Lightly Asleep and Time available to sleep is Alot and Quality of sleep is Bad, then Volume of alarm is medium and Coffee is Strong.

—Example of Rules which heavily influence louder Volume—

Assumptions: Volume of alarm is high, since user needs to be woken from deep sleep. The user has slept a lot and slept well; but because he was awoken from deep-sleep, he might not be fully awake yet, so coffee should be regular instead of weak.

If Current sleep cycle is Fast Asleep and Time available to sleep is Alot and Quality of sleep is Good, then Volume of alarm is High and Coffee is Regular.