For office use only
T1 _______________
T2 _______________
T3 _______________
T4 _______________

For office use only
F1 _______________
F2 _______________
F3 _______________
F4 _______________

2018

21st Annual High School Mathematical Contest in Modeling (HiMCM) Summary Sheet
(Please make this the first page of your electronic Solution Paper.)

Team Control Number: 9372
Problem Chosen: B

Please paste or type a summary of your results on this page. Please remember not to include the name of your school, advisor, or team members on this page.

In this paper, we created a smart home climate control system that saves energy and time for the customer by learning their preferences and habits through a regression model. Although there are various smart thermostats on the market right now, our system distinguishes itself through the use of a lightweight regression algorithm. As opposed to using deep machine learning models, our algorithm requires much less computing power and relies more on the collected data from various users. Our model's use of a such an algorithm leads to cheaper costs, while our modeling of thermodynamics leads to efficient usage of energy, which in turn yields a cheaper and greener solution to HVAC costs. Our model collects environmental data, such as the RealFeel temperature, and uses this data to calculate accurate predictions for home temperatures that our users would find comfortable. The user's smartphone can also connect to the system for ease of use, and the device takes advantage of location-tracking technology to anticipate when the user is arriving home, making sure that the user arrives to a comfortable home. When the system knows that the user is not home, the device shuts off cooling and heating to save energy. The home climate control system can operate with a single thermostat, or it can be extended to multiple thermostats to work in large houses or even buildings. Our thermostat currently only has a temperature sensor. There are other variables we can take into account, such as pollutants and airflow, when calculating the preferred temperature of an ideal setting. We hope to scale our product to include sensors for air quality and motion detection, as these factors play an important role in determining safe and comfortable temperatures for a room or household. Additionally, we plan on improving the machine learning model through incorporating certain novel deep learning techniques such as the inclusion of embedding matrices as these can incorporate categorical data, such as the day of the week and address, into the model. Leveraging the full power of deep learning would drastically improve our model's specificity and sensitivity.

HeatHakr: A Smart Home Climate System
Team Control Number: 9372
Problem Chosen: B

November 2018

Introduction

Importance of Smart Home Climate Systems

Smart thermostat systems save consumers time, effort, and money, and decrease household health risks. Even mobile apps that allow people to change the temperatures of their home at anytime from anywhere have their shortfalls. In order to change the temperature of their house, a person must log onto their phone, check the temperature, and then manually adjust the thermostat. However, not only is this time consuming, it is difficult to use when outside the house. What exactly does 75°at 38% humidity feel like?

Moreover, minor inaccuracies in thermostat adjustments can lead to large energy costs. Inaccurate thermostats can increase energy costs by over 8%. Moreover, this increase in energy consumption worsens climate change. Correcting these small inefficiencies can prevent the release of 100s of pounds of carbon dioxide every year.

Finally, less-than-ideal air temperatures can worsen the effects of air pollution in a household. For example, most common pollutants, which kill 200,000 Americans each year, are more dangerous with hotter temperatures. Pollen in the air can have worse effects with higher temperatures, and regulating humidity can prevent the growth of harmful and damaging mold.

Analysis of the Task

The smart home climate control system has 2 primary goals:

- Save energy when the user is away from their home
- Adjust your home to the perfect temperature when you arrive

To accomplish these goals, the device must be able to do 3 operations:

- **Learn the User's Preferences**: In a given environment, identify a temperature that the user would be comfortable in

- **Schedule**: Predict when the user will arrive home
- **Perform Heating**: Be able to heat the home to a given temperature in a given duration of time

These three operations work together in a pipeline process to produce a fully functioning climate control system:

Predict when the user is coming home → Determine a comfortable temperature → Heat the room to the comfortable temperature

The climate control system should operate within an entire household, which is made up of many zones. To accomplish this, the system is comprised of multiple smart thermostats that will be responsible for controlling each individual thermal zone.

We will first cover how an individual thermostat operates, and then explain how the thermostats work in conjunction to regulate an entire household.

Single-Thermostat System

User Preferences

Everybody's senses differ. What one person may perceive to be a comfortable temperature does not mean that another will. Our human biology makes some individuals more sensitive to cold temperatures while making others more sensitive to warmer temperatures.

For this reason, the smart thermostat must be able to learn from and eventually predict the user's temperature preferences. Such an algorithm would take the outside environment as an input and output what it thinks the user's preferred home temperature is.

It is important to quantify the input and output of such a model. The environment can be described through many ways: temperature, humidity, wind speeds, type of precipitation, etc. To provide an accurate description of the environment, a few of these would need to be used. This would make the input data to the predictive algorithm a vector. However, attempts to make a scalar index that describes many aspects of the environment have already been made. One such example is AccuWeather's RealFeel® temperature, which takes into account many environmental variables to produce a single temperature of what a human would truly feel. We propose using the RealFeel® index as an initial input to the predictive algorithm.

This approach has many benefits. First, using a scalar for the input as opposed to a vector saves processing power and memory, especially when a lot of data is involved. The input to the algorithm is also independent of geography. Since any region on Earth may be described with environmental data, there is no need to account for geography. Lastly, in the event that the input is expanded upon and made into a vector, the algorithm is flexible and may easily adapt to the new input vector.

The output of the predictive algorithm is the user's preferred home temperature, as the smart thermostat device cannot change anything beyond the temperature of the home. Thus, the output is a scalar that represents the temperature of the home.

The predictive algorithm can be abstracted as follows:

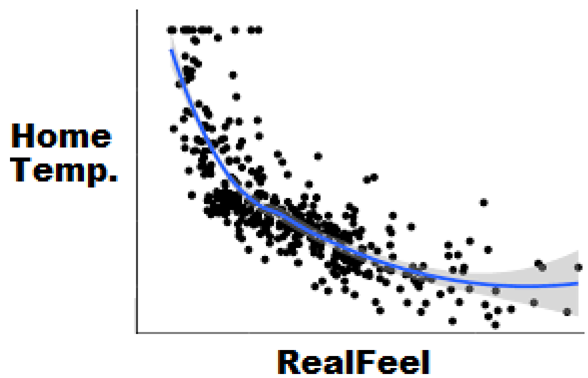
$$\underbrace{RealFeel \circledR Temperature}_{\text{Input}} \rightarrow \boxed{algorithm} \rightarrow \underbrace{HomeTemperaturePrediction}_{\text{Output}} \tag{1}$$

The general form of the algorithm used is a regression algorithm. In the case of the scalar input, the algorithm is a single variable regression algorithm. One important observation about the data is that the way in which RealFeel varies is mainly continuous. For example, when the winter season begins, the temperature is relatively high. As the winter progresses, the temperature progressively drops. This means that if we let the user set their own manual temperature when they first purchase our system, and we record the RealFeel temperature outside along with the user's actual home temperature, over some period of time we will get data points whose RealFeel temperatures are very close together. This continuous property of the graph makes polynomial regression an appropriate algorithm for this task.

The degree of the polynomial regression must be chosen carefully to prevent underfitting or overfitting. The relationship between the outside environment and the perceived human comfortable temperature is complex enough for the degree to be higher than linear, but it is also not very complex. Thus we propose a degree of 3 for the regression algorithm.

The algorithm would perform regression on a set of data points of the form (RealFeel, Home Temp.), and would compute a third degree polynomial that best represents the relationship between the RealFeel input and the Home Temp. output.

An example of the third-degree polynomial regression algorithm



Where does the data for the regression come from? To answer this question, we propose two solutions: one in the short-run, and one in the long-run.

Short-Run

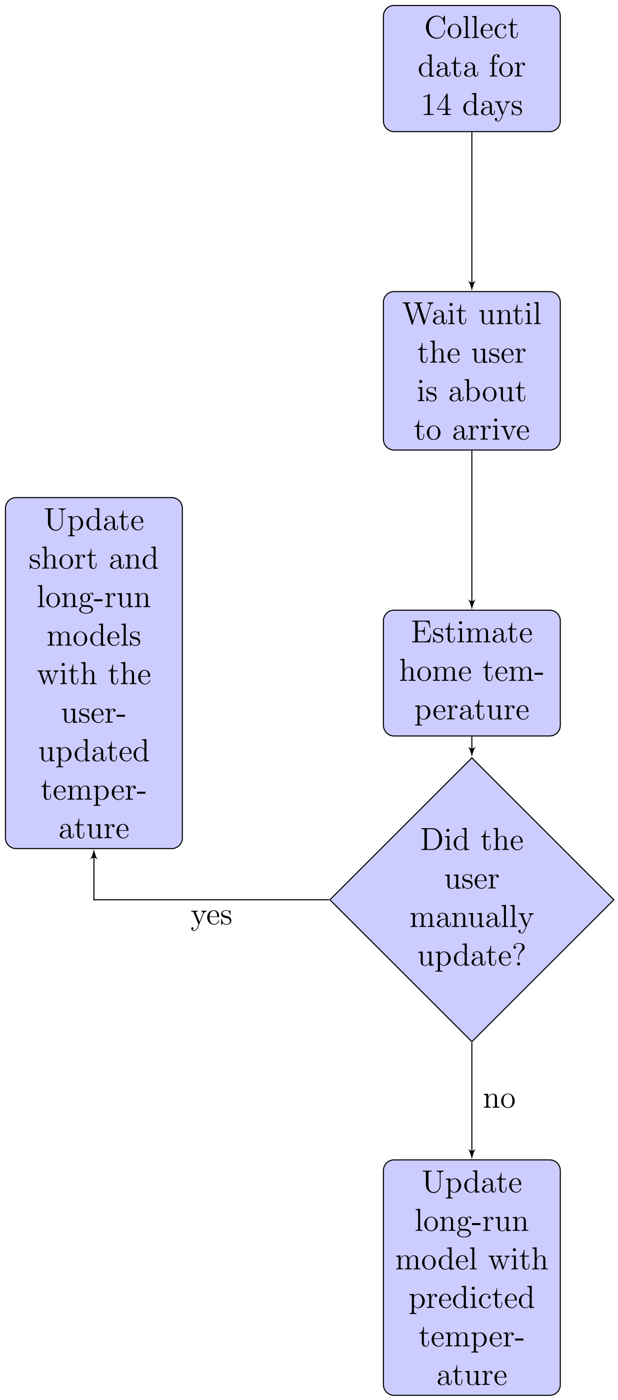
We define the short-run as one season. An observation in the short run is that the desired home temperatures fall in a unique range for each season. In other words, there is no one universal desired home temperature range among all the seasons. The reasons for this phenomenon are plenty, and come from precipitation patterns, the typical activities that are performed outside during a season, and even human psychology. For example, when it snows in the winter, people feel colder because cold snow is contacting their skin. When they come home, they desire an inflated home temperature because the snow has cooled down their body temperatures. Another example of this phenomenon takes place during the summer. People are more likely to be dehydrated in the summer, and thus desire an unusually low home temperature when they arrive home. Psychology also plays a big role: the more extreme the outside temperature or precipitation is, the more people may exaggerate their home temperatures in either direction. Someone who is sitting inside and sees snow outside might "feel" colder and have the need to turn up the temperature inside.

In the short-run, we propose collecting and storing data from the user. When there is enough data (we suggest 14 days), the system is ready to start making home temperature predictions. When the system thinks that the user is coming home (covered in the Scheduling section), the thermostat runs the polynomial regression on the data that exists so far. The thermostat then passes the current RealFeel temperature into the polynomial function as an argument. The polynomial function returns the estimated home temperature. The system then sends a signal to heat the room to the estimated home temperature (covered in the Heating section).

If the system is wrong with its prediction, and the user ends up changing the home temperature manually because they are not comfortable, the system stores the manually-updated (RealFeel, Home Temperature) pair and will use it in the future when it runs another regression cycle.

If the system is correct and the user does not change the home temperature manually because they are comfortable, the system stores the predicted (RealFeel, Home Temperature) pair and will use it in the future for the long-term model.

A Flowchart for the Short-Run Model


Collect data for 14 days
Wait until the user is about to arrive
Estimate home temperature
Did the user manually update?
yes
Update short and long-run models with the user-updated temperature
no
Update long-run model with predicted temperature

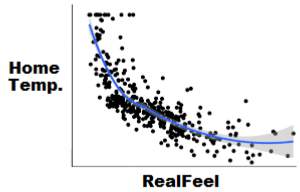
One important thing to note is that there is a range of error that is acceptable for the regression. Humans cannot detect extremely subtle changes in temperature, so our model has a higher probability of determining a home temperature that the user would be comfortable in.

Long-Run

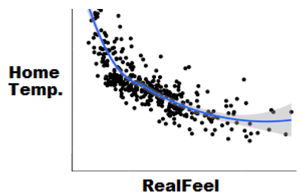
We define the long-run as the collection of all seasons. Only using the short-run model has a few drawbacks. As discussed previously, the desired home temperatures for a given season fall in a unique range. This means that when one season transitions into another, the model for the first season becomes useless for predicting the second season's home temperatures. So, a new seasonal short-term model must be constructed. But this is inconvenient for the user: every time a season changes, the user must manually update the thermostat for 14 days before the model can make predictions, and then he will have to manually update every time the inexperienced new model makes an incorrect prediction

To fix the inconvenience of building a new seasonal model, we must look at how the population reacts to seasonal changes. Although individuals may prefer different temperatures at a given moment, the overall change of temperature preferences across the seasons remains constant among all people. Suppose that there are two individuals, A and B. Individual A prefers slightly colder home temperatures in the winter compared to individual B. When the seasons change from winter to spring, and from the spring to the summer, individuals A and B will both prefer colder home temperatures in the summer, and the change between each individual's winter and summer home temperature will be similar.

The relationship above can also be described in terms of functions. The rate of change of both individual A and individual B's home temperature functions are very similar. But each individual's graph is a vertical translation of the other. Two graphs showing this relationship can be shown below:



Individual A



Individual B

The long-term model takes advantage of this relationship between different users of the device. If data about winter temperature preferences already exists from a collection of users (and regression is performed upon that data to obtain a polynomial function) we may estimate what the winter temperature preference is for a user whose device has not yet encountered the winter:

First, find an input RealFeel value that exists in the datasets of both the new user and the old users:

$$R_{common} = R_{new} = R_{old}$$

Next, find the difference between the output of the old user's polynomial function whose argument is the common RealFeel value and the output of the new user's polynomial function whose argument is the common RealFeel value:

$$\Delta T = P_{old}(R_{common}) - P_{new}(R_{common})$$

Note that many difference values can be calculated (if multiple intersecting RealFeel values exist). To obtain a final difference value, we propose taking the average of all difference values that exist. With this difference, we may now estimate a preferred temperature value for a RealFeel value that the new user's device has never encountered:

$$P_{new}(R_{unknown}) \approx P_{old}(R_{unknown}) - \Delta T$$

In the real world, instead of comparing data with some existing users, data can be compared with the entire population. As outlined in the flowchart for the short-term model, each device constantly sends data to the long-term model. The data from each device can be gathered in a database, and the database can then frequently perform polynomial regression to this aggregate data. Then, the procedure outlined above may be executed with this aggregate data and function to predict a preferred temperature for a user who is entering a new season.

Using Both Models

Both the long-term model and the short-term models have their advantages. The short-term model works well during the season, but performs poorly when the season ends. The long-term model performs better than the short-term model to predict initial seasonal preferences, but lacks performance during the duration of the season. To maximize functionality, build the short-term model for the season, and when the season ends, use the long-term model to estimate the beginning of the next season. After 14 days, begin constructing a new short-term model for the new season. Repeat this process for the year.

Once the user has used the device for an entire year, a combination of the long-term models and the previous year's short term models may be used to maximize accuracy.

If the input to the algorithm changed from a scalar to a vector, much of what has been covered will remain the same. Only the implementation of the regression algorithm will change: from a single-variable polynomial regression to a multivariate polynomial regression.

Scheduling

The device only heats (or cools) the home when the user is about to come home, or is at home. When the user leaves the home, the device shuts off the heating and cooling systems to save energy. The problem of knowing if the user is home or not can be solved in a few ways. We propose two main solutions: using location systems, and manual programming.

Location Systems
The schedule of the user will always, at some point in time, be irregular and break any pattern. The very nature of schedule data makes it very difficult to accurately predict when the user will arrive home based on a past history of time-stamped arrivals.

A much more reliable system is to use location-tracking software to anticipate the user's arrival. Since the user's smart phone is already connected to the smart device, and the smart phone will likely be with the user when he is traveling away from home, the thermostat may ping the smart phone with a request to gather location data. If the distance between the home and the phone is gradually decreasing, then the system will know that the user is coming home, or is near their home. Additional processing may be done to calculate exactly how long it will take the user to arrive home. The system can then begin the process of estimating a comfortable temperature (discussed in the User Preferences section), and can begin heating the home to the temperature.

This approach will surely require encryption to ensure that the location data of the user is secure and is only being communicated to the thermostat device.

Manual Programming
Since our device can be customized by the user at any time, we also offer the option of manual programming. The user may program the device to let it know when they will be gone for long periods of time (on a vacation, perhaps). The user may also program some-what rigid schedules. For example, if the user knows that they are not sensitive to extreme temperatures during their sleep, the user may want to program this information into the device in an attempt to save energy when they sleep.

Heating

One of the ways our thermostat optimizes user experience and energy use is through determining how many kilowatts per hour a radiator should use. To determine how much energy a radiator should use to heat a room of volume

$$V$$

to a certain temperature

$$t$$

we used the laws of thermodynamics.

The amount of heat energy,

$$Q$$

to heat a mass is:

$$Q = m * c * \Delta T$$

where

$$m$$

is the mass,

$$c$$

is the specific heat and

$$\Delta T$$

is the change in temperature. In a room of volume

$$V$$

the mass is equal to

$$\rho * V$$

where

$$\rho$$

is the density of air, 1.225 kg/m.

$$c$$

is equal to 1005 J/kgK. Thus,

$$Q = 1.225 * V * 1005 * \Delta T$$

which equals

$$2.791 * 10^{-4} * V * \Delta T$$

kilowatts/hour.

The amount of time it takes to heat up a room is

$$\frac{Q}{P}$$

where

$$P$$

is power in kilowatts/hour.

When the system knows that the user is coming home in some amount of time, the system can calculate how much power the system needs to output in the heating or cooling interface to reach the estimated comfortable temperature by the time the user comes home. The user would have to program in the volume of their room when they first purchase the device to aid in this process.

Advantages, Disadvantages, and Future Developments

Since our thermostat works using a multivariate polynomial regression algorithm, we are able to handle any environmental factors. This includes factors such as temperature of the outside environment, which plays a part pertaining to the user's preference, to even air pollution. Even though the output is a single value, the temperature of the internal environment, the factors that play into it have a large significance. For example, the difference between RealFeel temperature and actual temperature may play into account through the user's preference, as RealFeel temperature is much closer to what the user will feel as though the temperature is. The air pollution levels, another major factor, play into account because of the correlation between air temperature and the chemical reactions that many pollutants experience.

One disadvantage with our thermostat is the limited hardware. Although our algorithm is easily able to handle various external variables like air pollution and air flow, the hardware within our thermostat is currently only able to handle internal temperature. Our temperature sensors are currently the only sensors we have integrated within the device, which means our algorithm has only one type of data to compute off of. This can easily be improved with the addition of airflow sensors and air pollutant measuring devices which can be connected to the thermostat.

Another disadvantage our device faces is data collection. As of now, our device records data at all time of everybody using the device. This can easily become a problem of data storage, but an easy fix to this is to instill a cycle of data collection. One proposed cycle could be collecting data from each person twice a day, every other day. By doing so, we can prevent a surplus of data and collect only what we need.

Our algorithm works as an advantage when compared to other smart thermostats, as it runs on a lightweight regression algorithm. While some thermostats may use large data sets and deep learning models, our algorithm is much smaller in comparison and uses less computational power. Even though we have an astronomically smaller model, and less number crunching, our algorithms still learns from the user's preferences and variables of the external environment to accurately reproduce and handle various situations.

Additionally, we plan on improving the machine learning model through incorporating certain novel deep learning techniques such as the inclusion of embedding matrices, as these can incorporate categorical data, such as the day of the week, into the model. Leveraging the full power of deep learning would drastically improve our model's specificity and sensitivity.

Alternatives

Compared to other smart thermostats, our product is different in a couple of ways. Some of the most popular smart thermostats in the market are the Nest Learning Thermostat (3rd generation) and the Ecobee4 Alexa-Enabled Thermostat. The Nest Learning Thermostat has its own merits in that it is able to rapidly collect massive amounts of data of a user's preferences.

However, this can only be accomplished with the use of Nest's full suite of devices such as the carbon monoxide monitor. On the other hand, our thermostat is standalone and has a very lightweight regression algorithm that makes the product much cheaper and user-friendly. In fact, in the advent of the Internet of Things, sensors such as carbon monoxide and temperature are becoming much smaller and cheaper. This would enable our thermostat to incorporate more sensors and possibly be on par with Nest's connected-device data collection method while still keeping costs low.

On the other hand, the Ecobee4 Alexa-Enabled Thermostat is unique because of its voice-based interaction. This vastly improves user experience. However, in today's world, researchers are constantly working on improving machine learning algorithms, making tasks such as voice-based interaction much more accessible to smaller companies such as our smart-thermostat development group.

Multiple-Thermostat System

Extending the single-thermostat system to a multiple-thermostat system in a house can be done by treating each thermostat as its own unit, which covers one room (or zone).

If a room in a home has a main occupant (for example, the room of a child), the thermostat may be linked with the child's smart phone and operate identically to a single-thermostat system. If the room is mainly shared, such as a living room or kitchen, then there are a few solutions. First, the room can be linked with the smart phone of the person that the household chooses. The other option for the user is to take advantage of the manual scheduling that the system offers, and manually program when to heat the shared room.

Another creative solution is to use motion sensors to determine traffic into a room, which is then used to decide whether or not to heat the room based on how often it is used.

<u>HeatHakr Press Release</u>

Come buy your all new Heathakr today, for just $12.95. Using next-generation machine learning and artificial intelligence technology, this thermostat will automatically adjust the temperature of you home to suit your needs! Bring your SmartHome to the next level! It's easy to use. Just manually adjust its settings for a few days, and it will learn from your habits. You can even adjust it from your phone. Based upon your preferences, your Heathakr will correct the conditions of your home based upon the temperature, humidity, present air pollution, and more! It will never set the temperature too high or too low, and will thus save hundreds of dollars in energy costs over its lifetime. You will also never feel too hot or too cold again! Get yours today, for just $12.95, and start saving money.

<u>Call 1-800-Heathakr today, or order yours online with free shipping at www.heathakr.com/order.</u>

Your Annual Energy Cost	Your Potential Biannual Savings	Savings - Cost of Heathakr
$500	$80	$67.05
$1000	$160	$147.05
$1500	$240	$227.05
$2500	$400	$387.05

Wow, that's a lot of money! Order your Heathakr today!

Works Used

Bailes, Allison, 2-17-2012, "If You Think Thermostat Setbacks Don't Save Energy, You're Wrong!," Energy Vanguard, https://www.energyvanguard.com/blog/50152/If-You-Think-Thermostat-Setbacks-Don-t-Save-Energy-You-re-Wrong

Chu, Jennifer, 8-29-2013, "Study: Air pollution causes 200,000 early deaths each year in the U.S.," MIT News, http://news.mit.edu/2013/study-air-pollution-causes-200000-early-deaths-each-year-in-the-us-0829

D'Aprile, Jason. "Best Smart Thermostat: Reviews and Buying Advice." TechHive. October 11, 2018. Accessed November 20, 2018. https://www.techhive.com/article/3206 smart-thermostat.html.

ENGINEERING.com. "Research Report: How the Shrinking Cost of Sensors Is Fueling the Internet of Things." Biomedical Engineering Jobs — ENGINEERING.com. Accessed November 20, 2018. https://www.engineering.com/ResourceMain.

"The AccuWeather RealFeel Temperature." AccuWeather. September 28, 2011. Accessed November 20, 2018. https://www.accuweather.com/en/outdoor-articles/outdoor-living/the-accuweather-realfeel-tempe/55627?fbclid=IwAR1cJOCC27oJLeoFn9hi zYOJY6E.

Us Epa, Office Of Air and Radiation, Office Of Atmospheric Programs, Climate Change Division, "Carbon Footprint Calculator," accessed 2018, https://www3.epa.gov/carbon-footprint-calculator/