ECT Lesson Plan: Working with Large Tables of Data

Lesson plan at a glance...

Core subject(s)	Science
Subject area(s)	Physics
Suggested age	11 to 18 years old
Prerequisites	Understand the definition of mean, sort, and bearing; Sorting the World's Cities with Python and Google Sheets (optional)
Time	**Preparation:** 7 to 25+ minutes **Instruction:** 90 to 150 minutes
Standards	**Core Subject:** NGSS **CS:** CSTA, UK, Australia

In this lesson plan…

Lesson Overview

In this lesson, students will work with large tables of GPS data and **sort**, manipulate, and visualize the data so it can be easily understood. Students will use **data collection, data representation, decomposition, data analysis**, and **automation** to gather information and data, choose which way to represent it, regroup and filter the information for important data, find out statistical maxima, minima, and means, and then use programming to automate the drawing of a graphical representation of the data.

Materials and Equipment

- ☐ For the teacher:
  - *Required:* Google Sheets (http://docs.google.com/spreadsheets) or other spreadsheet app
  - *Recommended*: Android phone (My Tracks app [https://market.android.com/details?id=com.google.android.maps.mytracks]), sensors (Vernier, Pasco, etc.) or online data sources (e.g. CDC [http://www.cdc.gov/datastatistics/], NOAA [http://www.ngdc.noaa.gov/ngdc.html], CIA World Factbook [https://www.cia.gov/library/publications/the-world-factbook/], UN [http://data.un.org/])
- ☐ For the student:
  - *Required:* Internet-connected computer (one (1) computer per student recommended)
  - *Required:* Software Development Environment
    - Python 2.x (https://www.python.org/) OR a web-based Integrated Development Environment (IDE) such as Trinket (https://trinket.io/)

Preparation Tasks

	Confirm that your computer is on and logged-in	1 to 3 minutes
	Install Android or iPhone app	2 minutes per phone
	Confirm that all students' computers are turned on, logged-in, and connected to the Internet	1 to 10 minutes
	Confirm that Python is installed (https://www.python.org/), or navigate to Trinket (https://trinket.io/)	3 to 10 minutes

The Lesson

Warm-up Activity: Examining GPS data	10 to 60 minutes
Activity 1: Analyzing GPS data (min, max, average)	20 minutes
Activity 2: Formatting and visualizing GPS data with Fusion Tables	20 minutes
Activity 3: Visualizing data with Python	20 minutes
Wrap-up Activity: Assessment	20 minutes

Warm-up Activity: Examining GPS Data (10 to 60 minutes)

Activity Overview: In this activity students will analyze a relatively small amount of GPS data. This activity can be adapted to any data your class is working with. The teacher will choose a method to collect data and give students an opportunity to examine the data before analyzing it. Students will use data collection to gather information to use for GPS exercises.

Notes to the Teacher:
Global Positioning Satellite (GPS) data was formerly reserved for military purposes. However, public demand and economics have driven down the cost so all phones now include the ability to use this information. GPS data is used to give directions, sync clocks, and update traffic conditions.

Activity:
While sample data is provided for following activities, you may wish to collect your own. If you have an Android phone, then **My Tracks** (https://play.google.com/store/apps/details?id=com.google.android.maps.mytracks) is a very easy way to collect and export data into a CSV file that can be uploaded and analyzed. Here is a data collection program for Android phones to make this easy: https://play.google.com/store/apps/details?id=com.brokenairplane.physicsGizmo. There are similar apps for the iPhone.

If you are using a GPS or other sensors, make sure that you are able to export the data in a readable format or that you have the sensor's proprietary software to analyze it. You can also get conversion utilities depending on the type of data.

Have your students collect data in small groups (student-student interaction).

Activity 1: Analyzing GPS data (min, max, average) (20 minutes)

Activity Overview: This activity uses kinesthetic methods as students actively collect data in the real world. Students will use data analysis to calculate statistical maxima, minima, and means for the data set.

Notes to the Teacher:
We suggest you rename the title of the spreadsheets before you share it so students can figure out where the data comes from for themselves based on the patterns and information in the data.

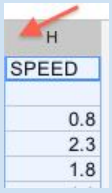
If students are using a spreadsheet program other than Google Sheets, download the data and share it with them so they can work with it as well.

Here is the list of all Google Spreadsheet functions (https://docs.google.com/support/bin/static.py?page=table.cs&topic=25273) that can be used to organize, manipulate, and analyze the data, in case you want to modify the lesson further.

Activity:
Preface this activity by saying how working with large tables of data can be overwhelming, so we use formulas and functions to quickly let us analyze and discover patterns. Three functions used often are minimum, maximum, and average.

1. Share the data with your students or use your own.
    a. Bicycle Commute, Bus Commute

2. Have students find the minimum:
    a. Click on the header of the column in which you wish to find the minimum value.



    b. In the menu bar, click on **Data → Sort Sheet by Column A → Z**
    c. The column is **sorted** from smallest to largest. The first number is the minimum.
    d. Another way to find the minimum is as follows:
        i. Create a blank cell
        ii. Enter the formula **=MIN(H3:H1000).** Where **H3** in this case is the first cell and **H1000** is the last and don't forget the equals sign in the beginning.
        iii. After pressing **Enter**, the cell will show the minimum for column H.
3. Finding the maximum:
    a. To find the maximum, either resort the column from **Z → A**, or replace **MIN** with **MAX** in the formula from above.
4. Finding the mean:
    a. Find a blank cell and enter **=AVERAGE(H3:H1000)**
    b. If you want to enter the formula for calculating average instead of using AVERAGE, type **=SUM(H3:H1000)/ROWS(H:H)** where ROWS(H:H) returns the number of rows.

Q1: For each of these jobs, write a sentence or two explaining how maximum, minimum, and average would be useful functions to a:
- **a. business person looking at quarterly reports**
- **b. doctor reading a patient's lab report**
- **c. politician evaluating the people's opinion on an issue**
- **d. meteorologist forecasting the day's weather**

Teaching Tips:
- Remind students of how much time they just saved by using these formulas. Building on the work of others increases our ability to be productive and creative.

Assessment:
A1: Answers will vary.

Activity 2: Formatting and visualizing GPS data with Fusion Tables (20 minutes)

Activity Overview: Raw data (similar to the data we gathered or collected or was given to us) is rarely useful to researchers. Whether it comes from sensors or a survey, sometimes the data needs to be formatted, other times, **outliers** need to be removed to not skew the data. In this activity, we will format the data in order to filter out unwanted information and keep information we do want like the time. This activity involves lower levels of learning (remembering,

understanding, and applying) in order to find important values and hopefully higher levels of learning (analysis) to interpret the material. Students will use **decomposition** and **data analysis** to regroup information and filter out what is unimportant while retaining parts of the data we do want. Students will use **data representation** to visualize the data in different ways or visualize different parts of the data in chart format.

Notes to the Teacher:
The bicycle and bus data was collected from the My Tracks (https://play.google.com/store/apps/details?id=com.google.android.maps.mytracks) phone app which recorded the date and time in "Zulu" or Greenwich mean time (http://wwp.greenwichmeantime.com/info/zulu.htm). As you only need the actual time to plot the data, remove everything else.

Introduce this activity to the students by explaining to them how rarely raw data is useful to researchers. Whether it comes from sensors or a survey, sometimes the data needs to be formatted; at other times, outliers need to be removed to not skew the data.

This information is saved as a "string" of data. We only want to keep a portion of that string which has the time. To filter out the unwanted information we can use the MID function. This function needs to know what data we want to filter, where to start and how many characters (bits of data) to extract.

So for each cell, we want to skip to the **12th spot** and go up **8 characters**.

Activity:
Have your students do the following activity:

1. **Look at the size of the data provided in the link above (or use your own collected data). To do this:**
   a. **Open the Google Sheet (bicycle or bus)**
   b. ***File → Download as → comma separated values* (.csv)**
   c. **Look at the properties for the downloaded file (depending on your operating system, try right-clicking the file and select properties.**

Q1: What is the size of the file?

2. **How does the size of the data compare with the amount of data stored? If you collected 5 minutes of data how much do you think the size of the data would increase if you collected 10 minutes? What about 1 hour worth of data?**
3. **How much bicycling data would you need to collect in order to fill up a computer with 1 Tb of memory? *Note: 1 Tb = 1000000000000 kB***

4. **Insert a new *Column C* to the right of *Column B* by right-clicking on the header of *Column B* and selecting *Insert 1 Right*.**
   a. **In the new *Column C* select *Cell C3* and enter =*VALUE(MID(B3, 12,8))*. After you push *Enter*, the contents of *B3* and *C3* should look like the picture below. You may need to change the column formatting to Time (*Format → Number → Time*).**

2011-08-16T14:16:55Z	14:16:55

You may need to format the cells by selecting the column, selecting *Format → Number → Time*.

Q2: What is the significance of *B3, 12*, and *8* in the *MID* function?

b. **Select *Cell C3* and copy. Next, highlight all of the remaining blank cells in *Column C* and paste. The spreadsheet will automatically apply the formula to the remaining cells.**
c. **Now delete the original *Time Column***

Teaching Tips:

- This is another opportunity to point out to the students how incredible it is that the computer was able to save us all of that time, so we can focus on interpreting the data.

Assessment:
A1: The provided data should be less than 100kB (kilobytes). Data collected by the students will be different.

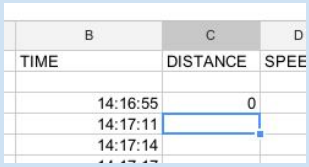
A2: Data (**D3**), start (**12**), length to keep (**8**)

Activity 3: Visualizing data with Python (20 minutes)

Activity Overview: For this activity, we learn to format our data in a different way so it can be used in Python. In this activity, students will create a visual representation or graph of their data. This activity utilizes the higher levels of learning (evaluation, creation, and analysis). Students will use automation and data representation, using a Python program to visualize the data in graphed pathway.

Activity:
Have students go through the following:

1. **Return to the original spreadsheet and add a column to the right of the formatted *TIME* column and title it *DISTANCE*.**
2. **Click on the 3rd cell in that column and enter a 0.**



	B	C	D
	TIME	DISTANCE	SPEE
	14:16:55	0	
	14:17:11		
	14:17:14		

3. **In the fourth cell enter =*SECOND(C4-C3)*D4***

Q1: Why are we subtracting C4-C3?

a. **This will find the distance traveled (*distance = speed * time)*.**
b. **Note: This will show the time in seconds. If you would like to see the time in minutes or hours use *MINUTE*() or *HOUR*() for those cells)**

4. **Copy *C4*, then select the rest of the column and paste.**
5. **Select and copy the data in *Column C* you just made (only the numbers).**
6. **Open Python and create a new blank page (Ctrl-N or Command-N), or open Trinket.**
7. **In the blank page type:**

```
dist = '''
```

   - **The three quotation marks allow you to paste in your data on multiple lines.**
8. **Position your cursor after the last quotation mark and paste in your data. At the end, add three more single quotation marks and hit enter. It should look something like this:**

```
dist = '''35     #Make sure your data starts right after the
12               #quotation marks or you will get an error.
47
...
42
31'''
```

9. **Return to the spreadsheet and highlight and copy the numbers in the *Bearing Column*.**
10. **Return to Python and type:**

```
bearing = '''
```

11. **Paste the bearings in from the spreadsheet and add three single quotation marks at the end.**
12. **Next, we need to transform our data into a format that can be used to draw the path from the bearings. We do this by creating a function that removes the "\n" (line breaks) and converts the string to numbers.**

```
def clean_up(myData):
      my_data = my_data.split('\n')
      for num in range(1,len(my_data)):
            my_data[num] = float(my_data[num])
      return my_data
```

13. **This algorithm will go through all the distances and bearings to draw the path the person took when they collected this data.**

```
for leg in range(1, len(dist) - 1):
      forward(dist[leg])
      setheading(bearing[leg])
```

14. **The final code with some required details added is below.**
15. **Open Python and press Ctrl-N or Command-N to open a new page.**
16. **Paste in the code (replacing the `dist` and `bearing` data with yours) and press *F5* to run the code.**

```
# Drawing a path based on large tables of data
from turtle import *

dist = '''35
12
...    # "..." represents all of the data, but abbreviated to save space
31'''

bearing = '''266
45
...
180'''

def clean_up(my_data):
    """Removes \n from data, splits it into a list, and converts it
    into a floating point (decimal) number"""
    my_data = my_data.split('\n')
    for num in range(0, len(my_data)):
        my_data[num] = float(my_data[num])
    return my_data

#Clean up your data (remove the \n and convert into numbers)
```

```
dist = clean_up(dist)
bearing = clean_up(bearing)

SCALE = 0.5 #Adjusts to fit path on larger/smaller screens

for leg in range(0, len(dist)):
    forward(dist[leg] * SCALE)
    setheading(bearing[leg])

#Add the following two lines to save as a pic (optional)
 picture = getscreen()
picture.getcanvas().postscript(file="track.ps")
```

Q2: What does \n mean in the code (example in line 17)?

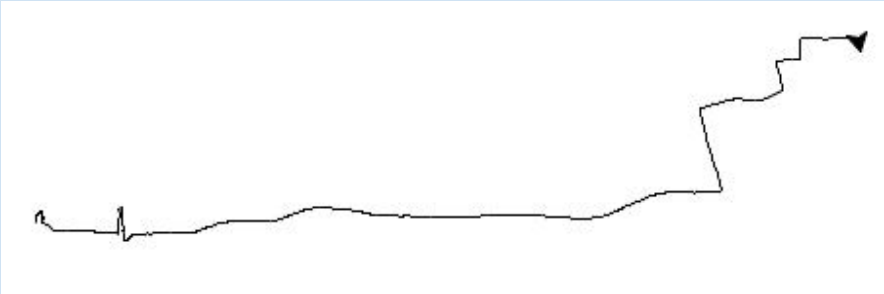
Q3: How would you adjust the code so SCALE is 5 instead of 0.5?

Q4: Your modification from question 2 does not work perfectly, why? How could you fix this?

Q5: How long does this code take to finish processing your data? Try doubling the size of your data (the fastest way may be to copy and paste the data stored in the variables `dist` and `heading`). How much more time does it take? Is it twice as much, less, more?

17. **Press F5 and save the file as *bigData.py*. The picture is saved in the program's directory.**
18. **Adjust the SCALE to fit the drawn path on one page.**

Bike Example Output:



Assessment:
A1: To find the change in time.

A2: Start a new line

A3: Instead of multiplying by **SCALE** in line 29, divide by **SCALE.**

A4: In Python, when an integer is divided by another integer, the program rounds down. To fix this, make **SCALE = 5.0**

A5: Answers may vary but the time for an algorithm to process data is not always linear. Said another way, some algorithms take twice as long to process twice as much data while others require more or less time. Finding the most efficient algorithm is often a goal for those who work with data to save time and resources. You can see a comparison of different algortihms' efficiency at sorting data on Wikipedia (http://en.wikipedia.org/wiki/Sorting_algorithm#Comparison_of_algorithms).

Wrap-up Activity: Assessment (10 minutes)

Activity Overview: In this student-focused assessment, students will assess their understanding of vocabulary, data collections, spreadsheet use, or use of fusion tables. This assessment will utilize spreadsheets, fusion tables, or a focus on vocabulary. Students may use **data collection**, **decomposition**, and **data analysis** to collect information, filter it, and use regroupings to make insights about the data. Students also may use **data representation** to visualize the data in narrative format.

Activity:
Choose from the following activities for your students:

- Vocabulary - Ask students to define and explain **mean, sort, outlier**, and **bearing.** You may want to recheck their knowledge of words like data, information, continuous, and discrete, as well.
- Assess for understanding:
    - Data Collection:
        - If you have access to sensors or smartphones, you might ask the students to collect data that create a certain pattern (sharp increase → constant/no change → slow increase, etc). Remember this lesson is a template that you can adapt to any continuous data (temperature, motion, sound intensity, etc).
        - Depending on the sensor, you might modify this question, but you might ask them if data could be collected to create specific shapes when drawn. For example could we collect data that looks like every letter in the alphabet, or a smiley face? **The goal is to show that not all shapes can be created with continuous data. For example, we could not draw a lowercase "i" with the data.**
    - Spreadsheets:
        - Provide a smaller data set to which you have added artificial outliers to skew the data. Ask students for the pros and cons of using mean to determine the average. You might challenge them to come up with a better average that minimizes the effects of outliers (e.g. **median**)
        - Challenge the students to find 3 functions for the spreadsheet software you are using (that have not previously been used in these lessons) and apply them to the data to get a new piece of information or see a pattern more clearly.
        - Have students add a column which converts the speed (m/s) into miles per hour.

Learning Objectives and Standards

Learning Objectives	**Standards**
LO1: Students will be able to define velocity as the rate of change of an object's position.	*Core Standards* NGSS MS-PS2-2: Plan an investigation to provide evidence that the change in an object's motion depends on the sum of the forces on the object and the mass of the object. [Clarification Statement: Emphasis is on balanced (Newton's First Law) and unbalanced forces in a system, qualitative comparisons of forces, mass and changes in motion (Newton's Second Law), frame of reference, and specification of units.] [Assessment Boundary: Assessment is limited to forces and changes in motion in one-dimension in an inertial reference frame and to change in one variable at a time. Assessment does not include the use of trigonometry.] NGSS HS-PS2-1: Analyze data to support the claim that Newton's second law of motion describes the mathematical relationship among the net force on a macroscopic object, its mass, and its acceleration. [Clarification Statement: Examples of data could include tables or graphs of position or velocity as a function of time for objects subject to a net unbalanced force, such as a falling object, an object rolling down a ramp, or a moving object

	being pulled by a constant force.] [Assessment Boundary: Assessment is limited to one-dimensional motion and to macroscopic objects moving at non-relativistic speeds.] *Computer Science* CSTA L2.CT.15: Provide examples of interdisciplinary applications of computational thinking.
LO2: Students will be able to solve problems that involve constant and average speed. (This lesson can be applied to other subjects as well. Physics is only chosen as an example.)	*Core Standards* NGSS MS-PS2-2: Plan an investigation to provide evidence that the change in an object's motion depends on the sum of the forces on the object and the mass of the object. [Clarification Statement: Emphasis is on balanced (Newton's First Law) and unbalanced forces in a system, qualitative comparisons of forces, mass and changes in motion (Newton's Second Law), frame of reference, and specification of units.] [Assessment Boundary: Assessment is limited to forces and changes in motion in one-dimension in an inertial reference frame and to change in one variable at a time. Assessment does not include the use of trigonometry.] NGSS HS-PS2-1: Analyze data to support the claim that Newton's second law of motion describes the mathematical relationship among the net force on a macroscopic object, its mass, and its acceleration. [Clarification Statement: Examples of data could include tables or graphs of position or velocity as a function of time for objects subject to a net unbalanced force, such as a falling object, an object rolling down a ramp, or a moving object being pulled by a constant force.] [Assessment Boundary: Assessment is limited to one-dimensional motion and to macroscopic objects moving at non-relativistic speeds.] *Computer Science* AUSTRALIA 8.4 (Collecting, managing and analyzing data): Analyse and visualise data using a range of software to create information; and use structured data to model objects or events. CSTA L3A.CT.8: Use modeling and simulation to represent and understand natural phenomena. UK 4.2: Develop and apply their analytic, problem-solving, design, and computational thinking skills.

Additional Information and Resources

Lesson Vocabulary

Term	Definition	For Additional Information
Mean	The central tendency or balance of a set of data. For example, if the ages in a family are 35, 33, 12, 7, and 2, then the mean is the sum of the ages divided by the number of people = 17.8. One could conclude that this family has slightly more younger people than older.	http://en.wikipedia.org/wiki/Mean
Sort	To reorder data according to a condition (minimum to maximum, alphabetical, etc).	http://en.wikipedia.org/wiki/Sorting

Bearing/ Heading	Bearing is the angular direction in which one is moving, relative to North.	http://en.wikipedia.org/wiki/Bearing_(navigation)
Outlier	A point of data that is extremely different from the rest. This could be a significant finding, or an experiment error, but the statistician must decide what to do with it as it could skew the data.	http://en.wikipedia.org/wiki/Outlier
Raw Data	Data collected which has not been cleaned up or formatted for a specific purpose.	http://en.wikipedia.org/wiki/Raw_data
Kilobyte	The **kilobyte** is a multiple of the unit byte for digital information. In the International System of Units (SI) the prefix *kilo-* means 1000 (10^3); therefore one kilobyte is 1000 bytes in this system. The unit symbol is **kB**.	https://en.wikipedia.org/wiki/Kilobyte

Computational Thinking Concepts

Concept	Definition
Data Collection	Gathering information
Data Analysis	Making sense of data by finding patterns or developing insights
Data Representation	Depicting and organizing data in appropriate graphs, charts, words or images
Decomposition	Breaking down data, processes or problems into smaller, manageable parts
Automation	Having computers or machines do repetitive tasks

Extension Activities for Student Enrichment

- Google has so far discovered trillions of web addresses. If you were to print 100 websites out onto paper, how would you organize those pieces of paper to make it easiest to search and find again?
- Data, when placed in a context, tells a story. It can be as simple as a pot of water coming to a boil, or the story of a person being late for work. Using Fusion Tables, Python Turtle or another method for visualizing data, have the students collect and then visualize data that is significant to them (e.g. daily commute, intensity of the sunlight that reaches their garden; encourage creativity).
    - It is very important to make sure that the data does not reveal private information (e.g. street address). In the data I shared, I was careful to remove the latitude and longitude, so that data could not reveal anything private. The amount of data we have access to is powerful, so we must be equally mindful of its use.

Administrative Details

Contact info For more info about Exploring Computational Thinking (ECT), visit the ECT website (g.co/exploringCT)

Credits Developed by the Exploring Computational Thinking team at Google and reviewed by K-12 educators from around the world.

Last updated on 07/02/2015

Copyright info Except as otherwise noted, the content of this document is licensed under the Creative Commons Attribution 4.0 International License, and code samples are licensed under the Apache 2.0 License.