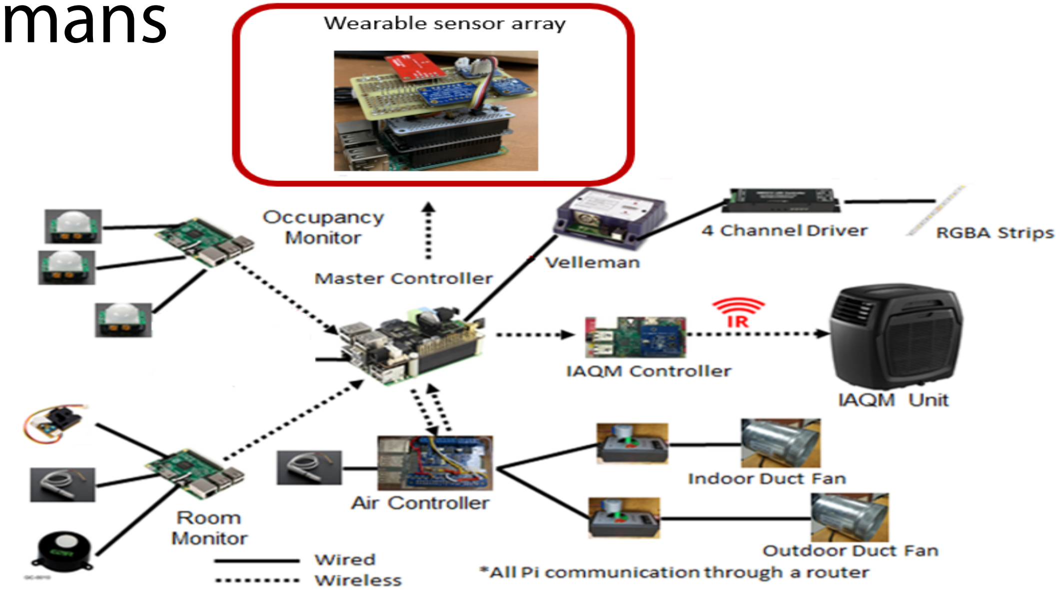
Northern Illinois University

An Open Source Software Solution for Data Acquisition, Management, and Analysis for Spectrometric Measurements

Kevin B. Martin, William J. Mills, III, Matthew Bikun
(Northern Illinois University, Dekalb, IL, USA)

Northern Illinois BEEEAM Lab

- Building Energy Efficiency, Ergonomics, and Management Lab
  - Building Management System (BMS)
  - Controlled Environment Agriculture
  - Indoor Environmental Impacts on Humans
  - Wearable Sensor Array



A Necessity for Streamlined Data Processing

- Extracting value from data acquired from multiple sensors in a timely and reliable manner requires:
  - Organization
  - Consistent formatting
  - Integrity of data throughout its lifecycle

- Finding meaning in data becomes difficult and inefficient without an optimized approach to data acquisition, management, and analysis

Lesson’s Learned from Welding Lab Monitoring

- Indoor air-quality study conducted as an undergraduate research project
  - Multiple aerosol monitoring instruments were deployed over the course of the semester
  - 35+ parameters were acquired on numerous occasions for multiple hours

A proper data organization, management, and analysis strategy was NOT utilized resulting in:

1) Inconsistent formatting
2) Overall disorganization
3) Data loss

Open Source Solution

- Data Source 1
- Data Source 2
- Data Source 3
- Data Source 4
- Data Source 5

Consistency among datasets acquired by different spectrometers is achieved via Python programming

After commonality is achieved, the re-formatted data is sent to a SQLite database

The SQLite database is queried from within an R-Studio database and subsequently analyzed

Python 3 Programming Language

- Easy readability and uncluttered simple-to-learn syntax
- Community Driven
  - Third-party packages and learning resources freely available
- Multi-platform
- Viable option for building complex multi-protocol network applications

SQLite Relational Database Management Software

- Low computational overhead
  - Ideal for on-device use or implementation in an Internet of Things (IOT) system
  - Serverless operability
  - 140 TB maximum file size
- Utilizes Structured Query Language (SQL)
  - ANSI standard
- Native Python support

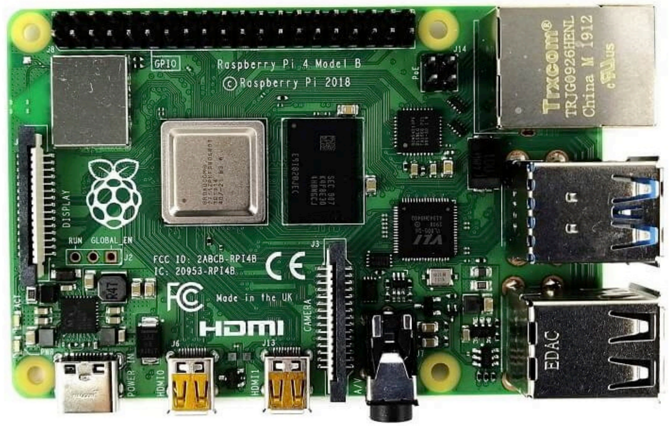
R-Statistical Programming Language

- Fully fledged statistical programming environment
- Multi-platform
- R-Studio: Integrated development environment for the R-language
  – Dynamic GUI's
  – Free for personal use* (License tiers for commercial use)
- RSQLite
  – Interfaces SQLite databases with R workspaces
- Powerful
  – Quickly analyze "big-data"

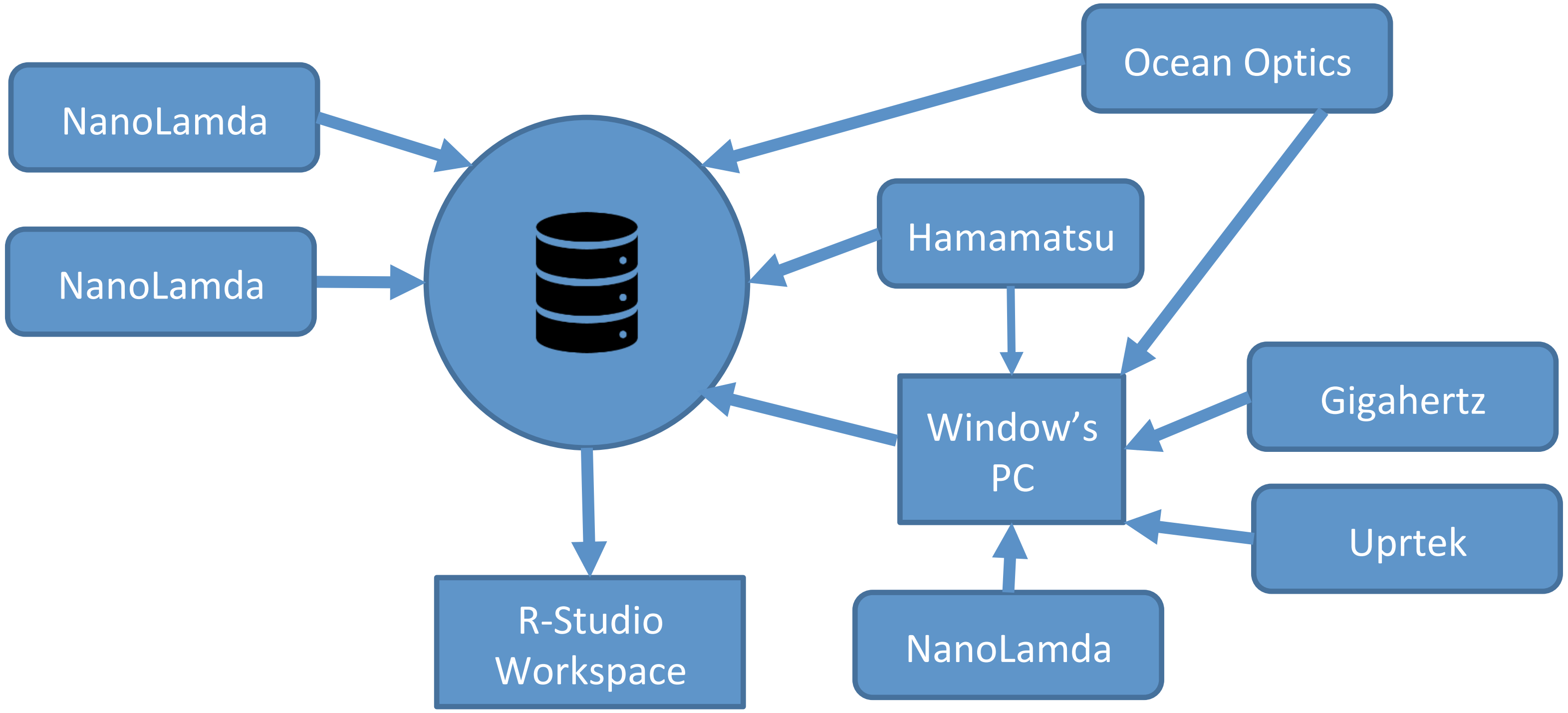
https://www.r-project.org/logo/

Low Cost Micro-Controllers & Micro-Computers

- Raspberry-Pi Computers
  - Low-cost Linux based micro-computer
  - Operates within an open-source ecosystem
  - Full computer capabilities
    - Data logging and data transmitting
  - Permits connectivity with multiple spectrometers and other sensors
  - CO_2, particulate, temperature, humidity
- Arduino
  - Open source micro-controller platform that allows simple connectivity with hardware



Handling Multiple Spectrometers

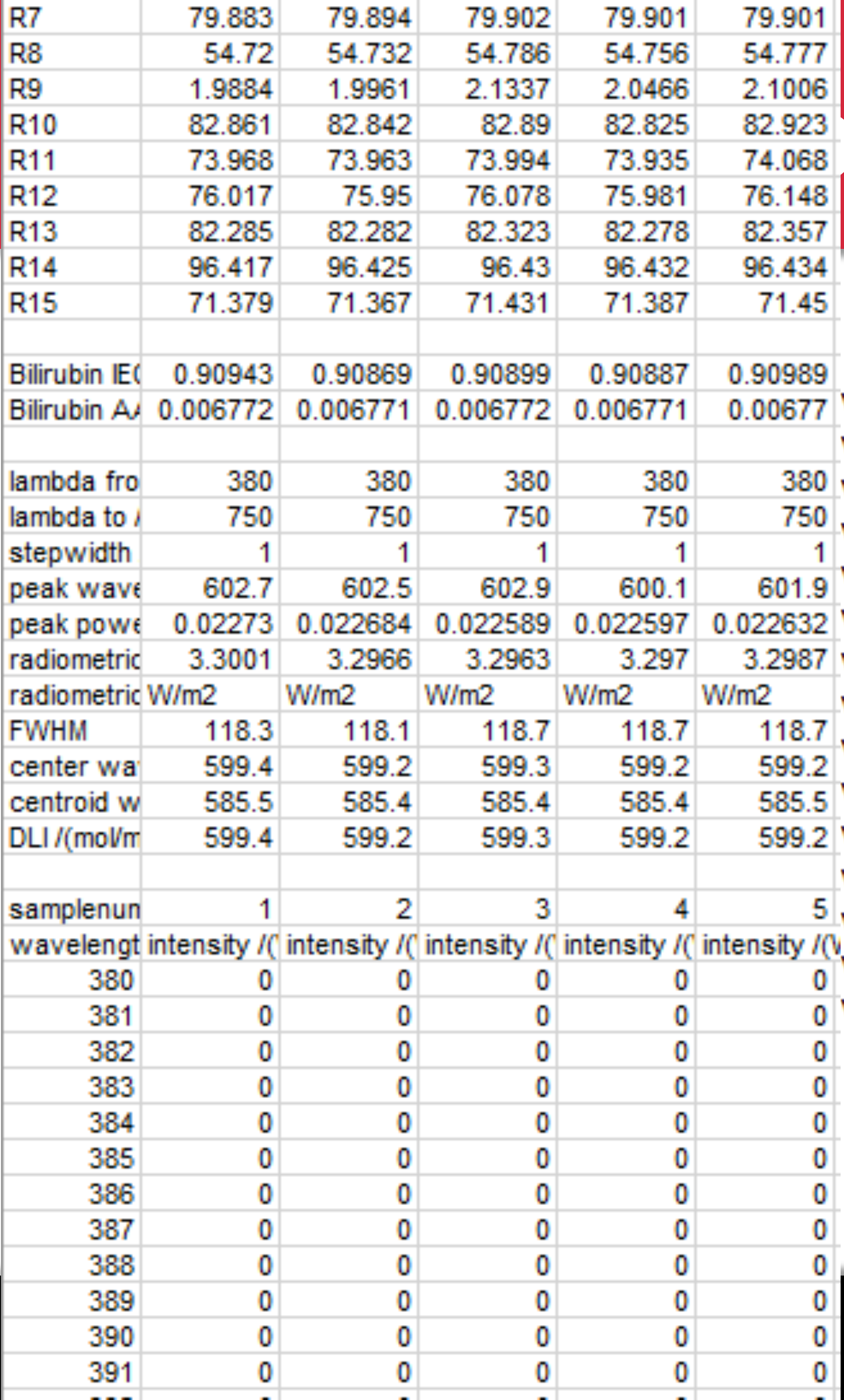


Micro-Computer Controlled Sensors

Measurements are sent directly to the database in a standardized format using Python based code

Windows Desktop Sensors

Python scripts scrub CSV files output from desktop software and the re-formatted data is input into the database

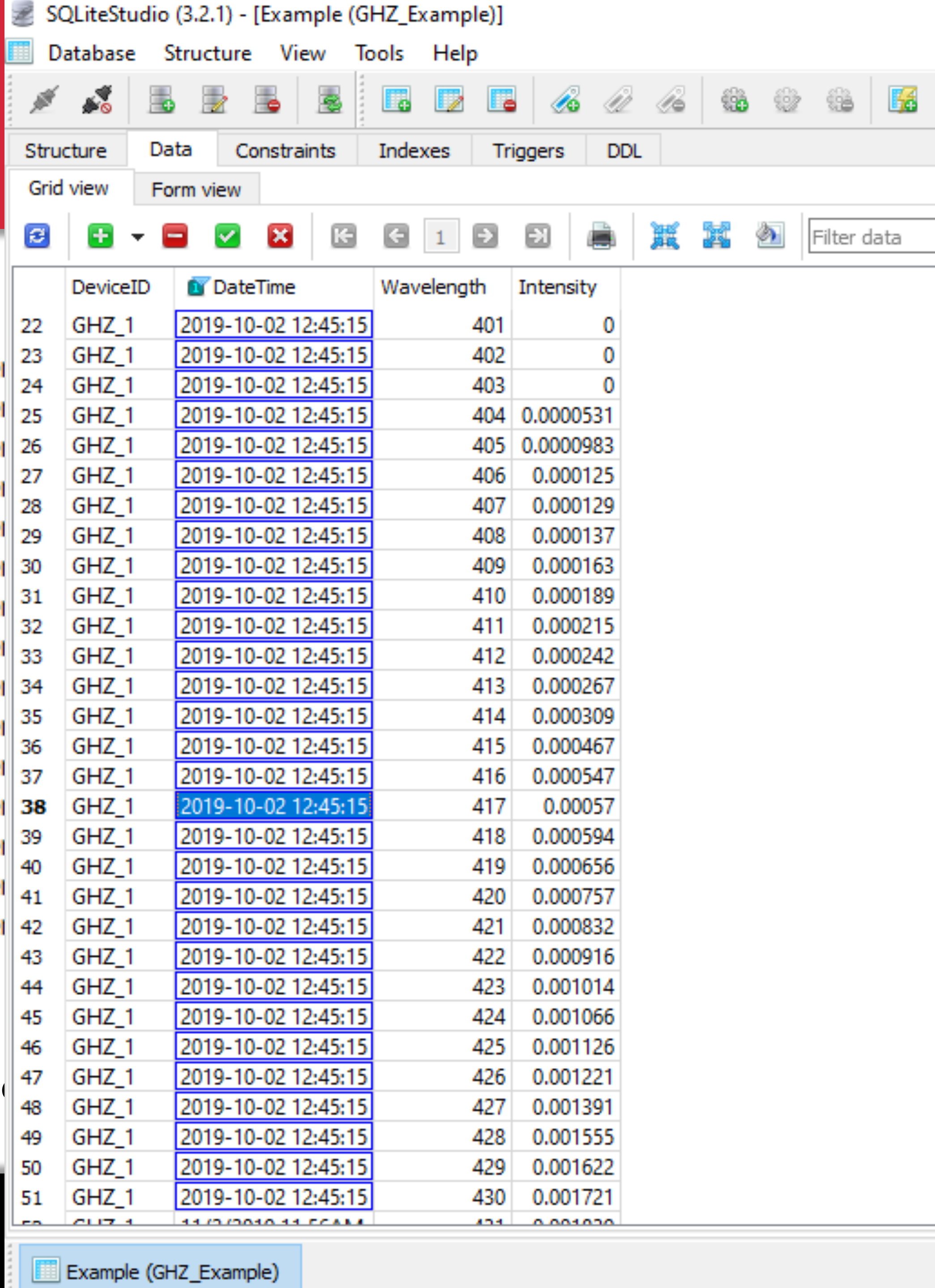


R7	79.883	79.894	79.902	79.901	79.901
R8	54.72	54.732	54.786	54.756	54.777
R9	1.9884	1.9961	2.1337	2.0466	2.1006
R10	82.861	82.842	82.89	82.825	82.923
R11	73.968	73.963	73.994	73.935	74.068
R12	76.017	75.95	76.078	75.981	76.148
R13	82.285	82.282	82.323	82.278	82.357
R14	96.417	96.425	96.43	96.432	96.434
R15	71.379	71.367	71.431	71.387	71.45
Bilirubin IE(	0.90943	0.90869	0.90899	0.90887	0.90989
Bilirubin A/	0.006772	0.006771	0.006772	0.006771	0.00677
lambda fro	380	380	380	380	380
lambda to /	750	750	750	750	750
stepwidth	1	1	1	1	1
peak wave	602.7	602.5	602.9	600.1	601.9
peak powe	0.02273	0.022684	0.022589	0.022597	0.022632
radiometric	3.3001	3.2966	3.2963	3.297	3.2987
radiometric	W/m2	W/m2	W/m2	W/m2	W/m2
FWHM	118.3	118.1	118.7	118.7	118.7
center wa	599.4	599.2	599.3	599.2	599.2
centroid w	585.5	585.4	585.4	585.4	585.5
DLI /(mol/m	599.4	599.2	599.3	599.2	599.2
samplenun	1	2	3	4	5
wavelengt	intensity /(	intensity /(	intensity /(	intensity /(	intensity /(\
380	0	0	0	0	0
381	0	0	0	0	0
382	0	0	0	0	0
383	0	0	0	0	0
384	0	0	0	0	0
385	0	0	0	0	0
386	0	0	0	0	0
387	0	0	0	0	0
388	0	0	0	0	0
389	0	0	0	0	0
390	0	0	0	0	0
391	0	0	0	0	0

Data Format

```
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
vice,Date,Time,Wavelength,measure
```

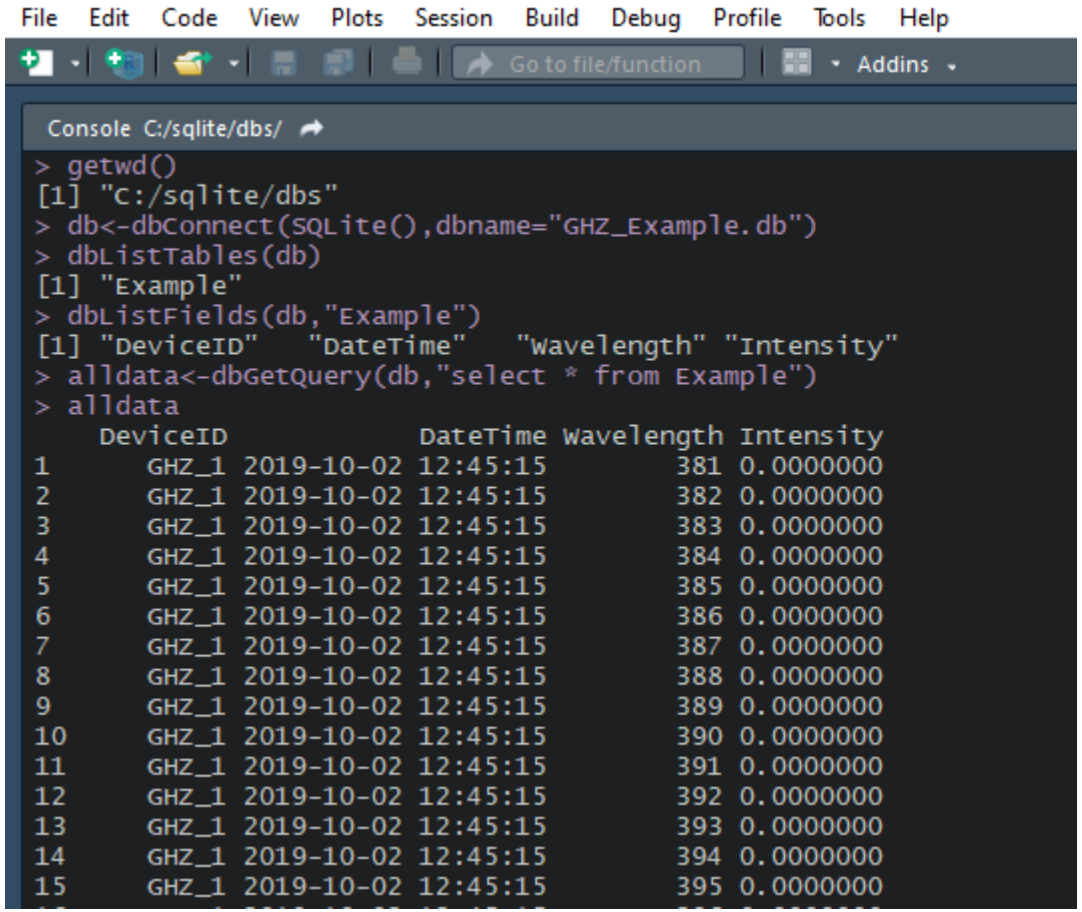
Standardized comma separat format



	DeviceID	DateTime	Wavelength	Intensity
22	GHZ_1	2019-10-02 12:45:15	401	0
23	GHZ_1	2019-10-02 12:45:15	402	0
24	GHZ_1	2019-10-02 12:45:15	403	0
25	GHZ_1	2019-10-02 12:45:15	404	0.0000531
26	GHZ_1	2019-10-02 12:45:15	405	0.0000983
27	GHZ_1	2019-10-02 12:45:15	406	0.000125
28	GHZ_1	2019-10-02 12:45:15	407	0.000129
29	GHZ_1	2019-10-02 12:45:15	408	0.000137
30	GHZ_1	2019-10-02 12:45:15	409	0.000163
31	GHZ_1	2019-10-02 12:45:15	410	0.000189
32	GHZ_1	2019-10-02 12:45:15	411	0.000215
33	GHZ_1	2019-10-02 12:45:15	412	0.000242
34	GHZ_1	2019-10-02 12:45:15	413	0.000267
35	GHZ_1	2019-10-02 12:45:15	414	0.000309
36	GHZ_1	2019-10-02 12:45:15	415	0.000467
37	GHZ_1	2019-10-02 12:45:15	416	0.000547
38	GHZ_1	2019-10-02 12:45:15	417	0.00057
39	GHZ_1	2019-10-02 12:45:15	418	0.000594
40	GHZ_1	2019-10-02 12:45:15	419	0.000656
41	GHZ_1	2019-10-02 12:45:15	420	0.000757
42	GHZ_1	2019-10-02 12:45:15	421	0.000832
43	GHZ_1	2019-10-02 12:45:15	422	0.000916
44	GHZ_1	2019-10-02 12:45:15	423	0.001014
45	GHZ_1	2019-10-02 12:45:15	424	0.001066
46	GHZ_1	2019-10-02 12:45:15	425	0.001126
47	GHZ_1	2019-10-02 12:45:15	426	0.001221
48	GHZ_1	2019-10-02 12:45:15	427	0.001391
49	GHZ_1	2019-10-02 12:45:15	428	0.001555
50	GHZ_1	2019-10-02 12:45:15	429	0.001622
51	GHZ_1	2019-10-02 12:45:15	430	0.001721

Simple DB Connection

- RSQLite
  - Freely available and frequently maintained R Library package
  - Allows interfacing of SQLite databases from within R
  - Allows database editing and creation from within R
- Once in R, data can be analyzed using other community created packages
  - PCA



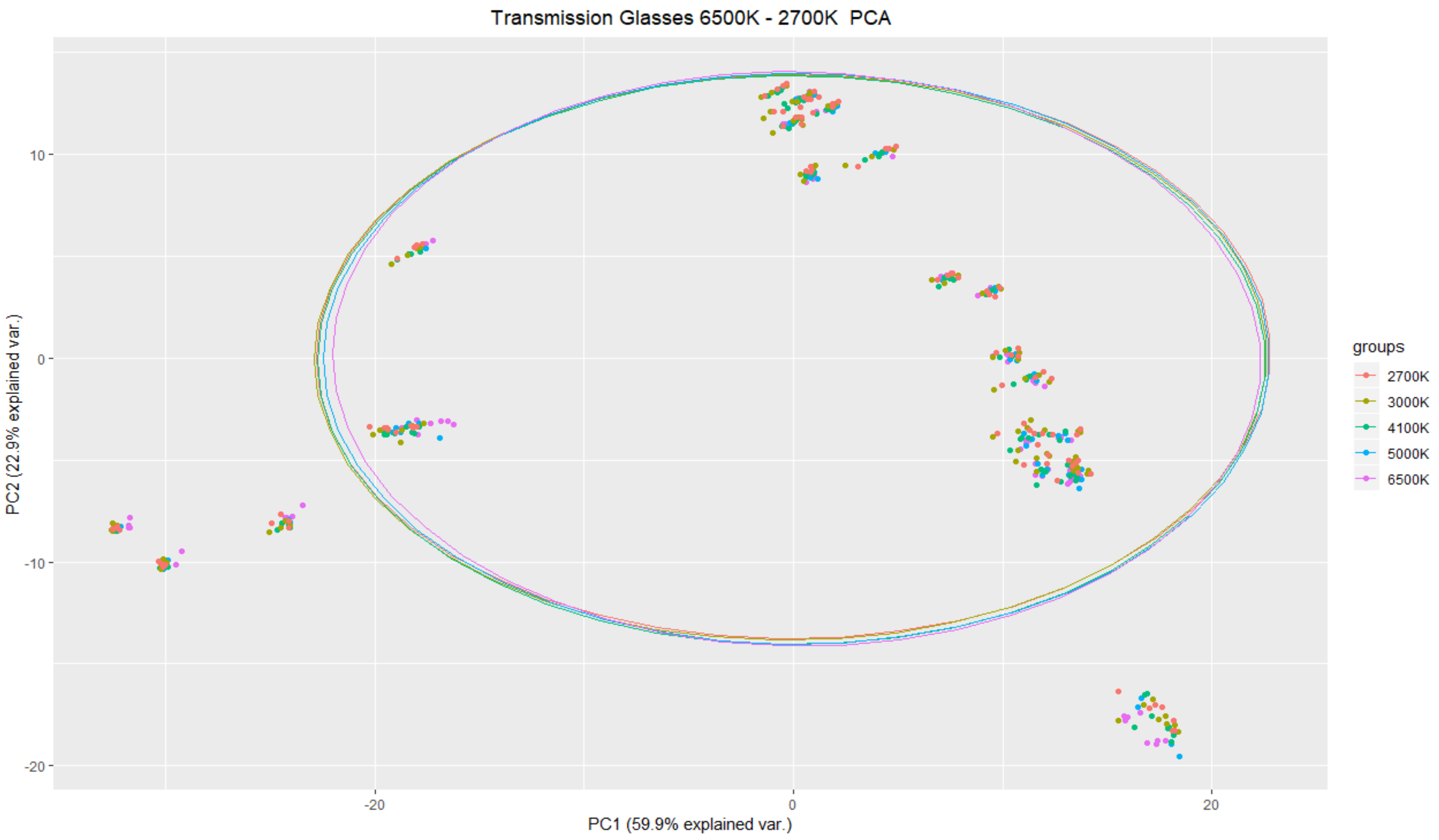
R-Based Lighting Calculations

- R based programs can be tailored to user needs
- Ideal for spectrometric data given the intensity values assigned to each wavelength

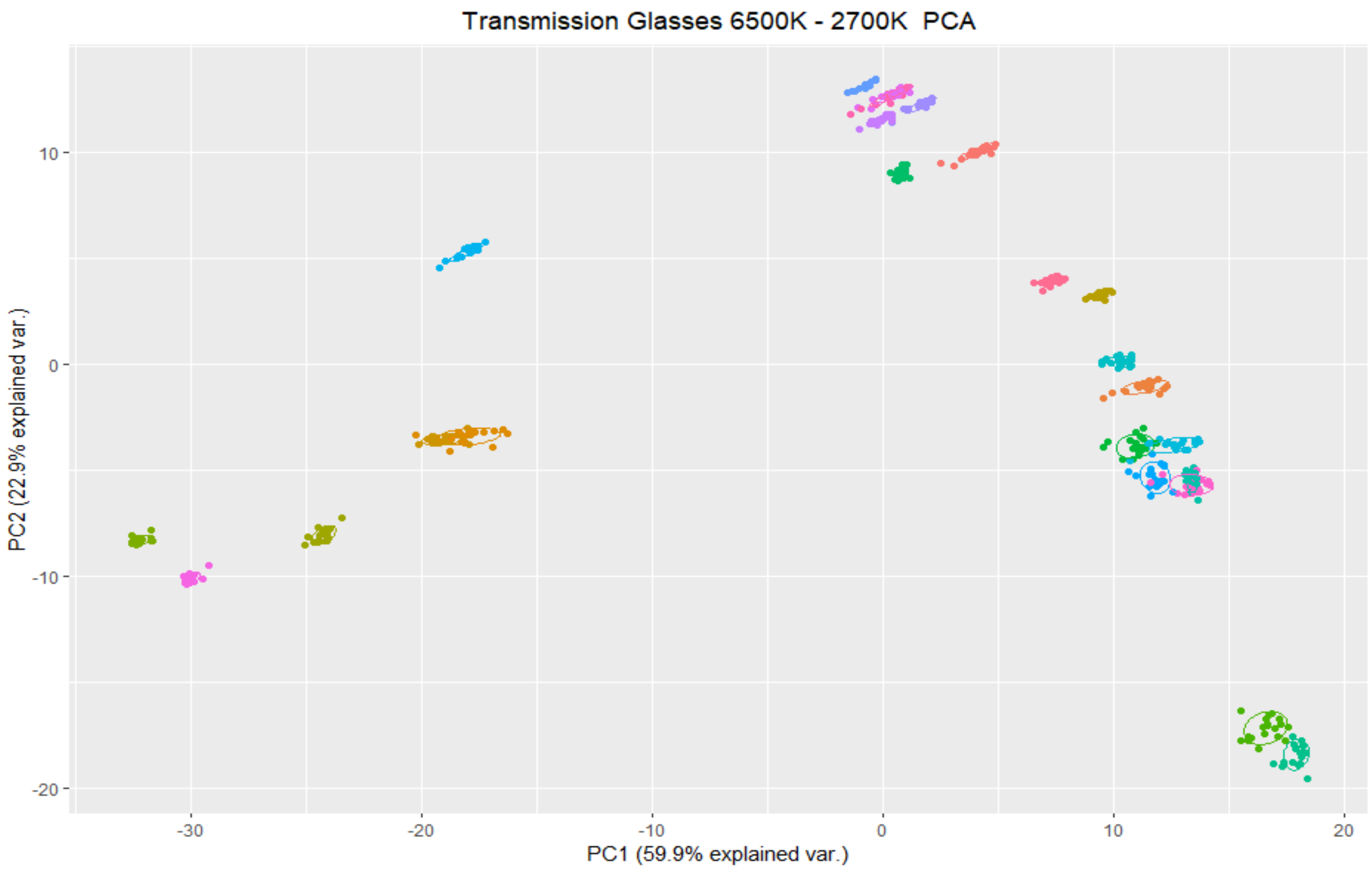
	x
Photopic_Area	1.5782667
Photopic_Area_Rel	69.7114267
Scotopic_Area	0.8269033
Scotopic_Area_Rel	36.5239961
Photopic_lux	1077.9593133
Scotopic_lux	1405.7355624
SP_Ratio	1.3040711
CCT	2861.6189645
X	1.7278334
Y	1.5785038
Z	0.5463422
CCT_x	0.4484758
CCT_y	0.4097158
skewness	0.5506112
kurtosis	2.1174859
Melanopic_lx	537.9356395
CLA	986.7349954
CS	0.5284402

Values derived from example data

Principal Component Analysis

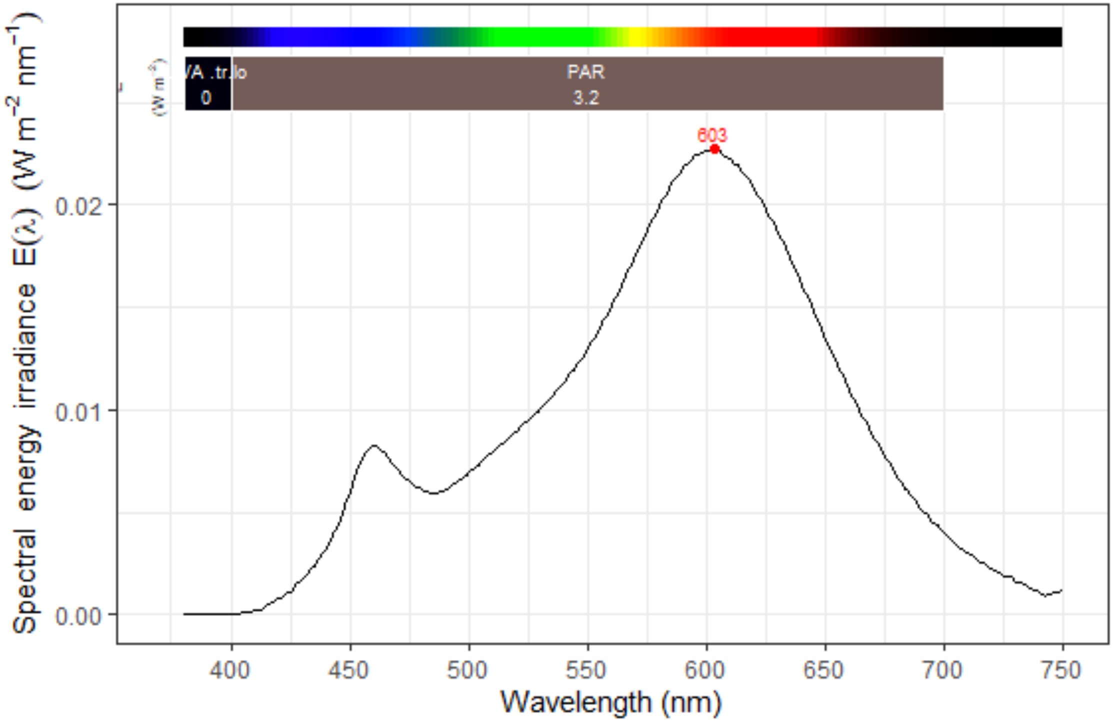


Principal Component Analysis



Other R Realted Benefits

- Plotting libraries (ggplot & ggspectra)
- Photobiology (agricultural lighting parameter calculator)



Conclusions

- Combination of SQLite and R in conjunction with Python provides a flexible, highly capable pathway for data set management and analysis
- Conceptually simple
  - Allows for template like approach
- Implementation
  - Sensor integration still can be problematic
  - Python code not always supplied by manufacturer