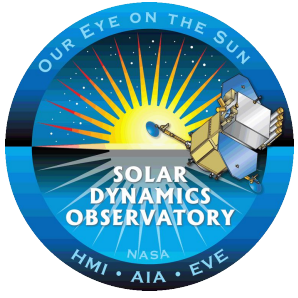
Data Science in CEDAR

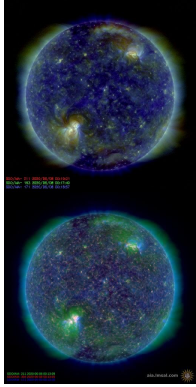
Progress, Capacity-Building, and Traversing Disciplines

Shea A. Hess Webber, PhD



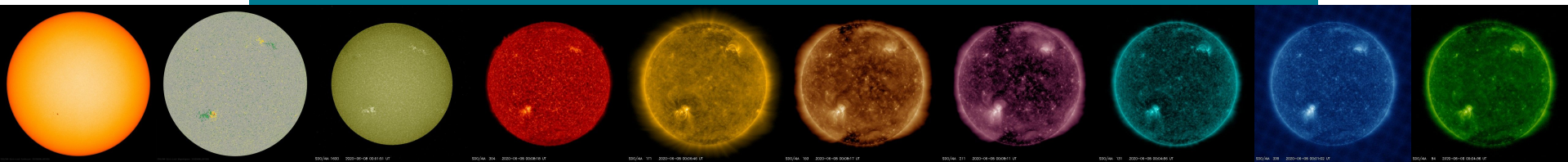
Stanford University

Big Data - Solar Dynamics Observatory



- SDO sends down about **1.5 terabytes of data per day**, equivalent to downloading half a million songs each day
- Most of the data are **4096x4096 full-disk images**
  - an image in one of 8 wavelengths every 10 seconds
- On November 5, 2019, SDO's AIA instrument took its 200-millionth image of the Sun
  - and that's just from one instrument!
- We currently have **over 350 million images** are in the SDO data archives
  - not including higher level data products, such as vector magnetic field, various helioseismology products, etc...

LARGE DATA & LOTS OF DATA



Big Data - Considerations

1. **How to retrieve it all?**
    a. Orbit
    b. Ground station(s) location
    c. Antenna specs (spacecraft and ground station)
    d. etc...
2. **How to store it all?**
    a. Raw data
    b. Calibrated data (corrections for spatial resolution, scattered light, and filter characteristics, etc)
    c. Additional higher level data products (*e.g.* dopplergrams -> 4 different helioseismology products)
    d. Backup systems
3. **How to access it all?**
    a. Administration
    b. Upkeep and maintenance
    c. Production scripts and documentation
    d. Database framework (e.g. PostgreSQL)
    e. Data management and server access (e.g. SUMS - Storage Unit Management System)
4. **How to *USE* it all?**
    a. Likely that less than half of all SDO data has been used for research
    b. Even using smaller “patches” of data for short time span, size of data cubes is prohibitive

Big Data *Usage* - Considerations

- **Supercomputers**
  - Machines with many CPU/GPU cores available for calculations
  - Various facilities/resources improve computation times for large data
- **Parallel computing**
  - Making use of local CPU cores for multiple threaded calculations simultaneously
  - Or GPU cores… ?
- **Code optimization**
  - Increase speed of calculations by making “smart” coding choices
  - *Very* basic example:

    100*100 is a faster process than 100^2

DISCLAIMER: I am not an expert in the above! I know something about them, but I’ve not had much “hands-on” experience. Like many of you, I’m just trying to get research done and papers written! Who’s got time to implement good big-data coding practices?! 😅😬 How do we collectively overcome this hurdle???