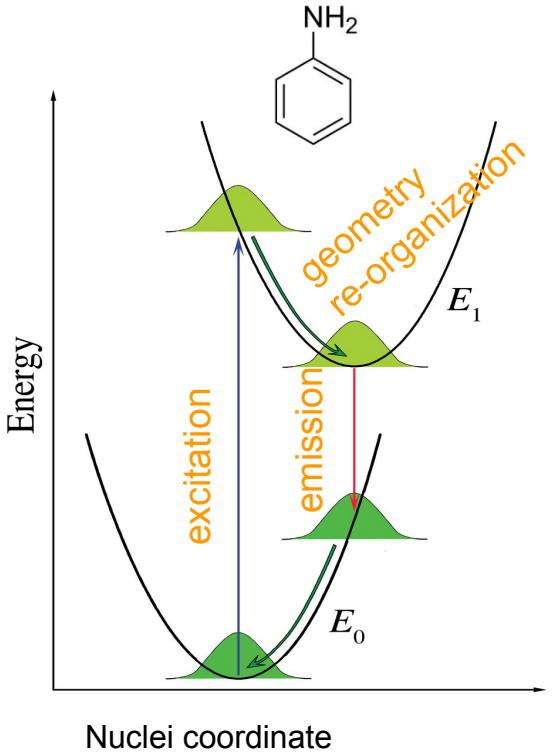
Technology Review--*SPEEDCOM*

Spectra Prediction for Excitation and Emission of Dyes and Conjugated Organic Molecules, using Machine Learning Methods

Hang, Ryan, Yang, Joe, Lixin

Background

Chromophore/Fluorophore Photophysics by *Ab Initio* Calculations



Frank-Condon Energy Diagram

Build Initial Geometry

- DFT Calculation
- DFT Gradient/Hessian for Geometry Optimization

Optimized Ground State Geometry

TD-DFT Calculation → λ_{abs}, f_{abs} , ε

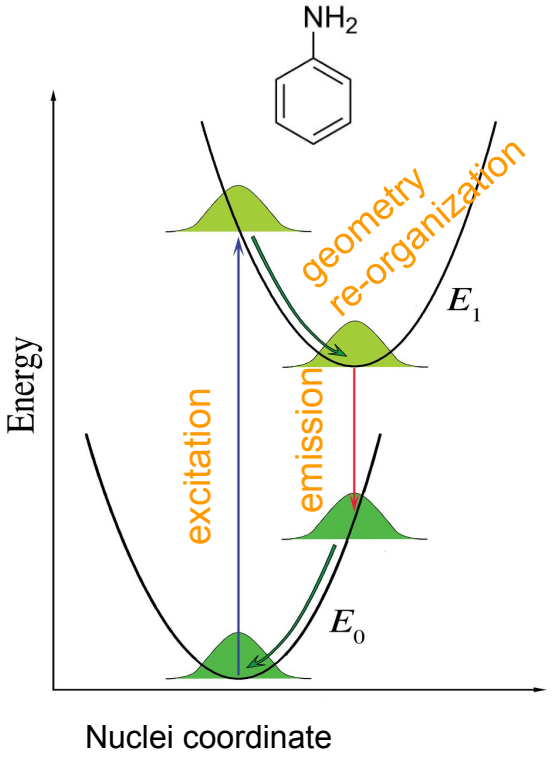
- TD-DFT Calculation
- TD-DFT Gradient/Hessian for Geometry Optimization

Optimized 1st Excited State Geometry

TD-DFT Calculation → λ_{em}, f_{em}, B

Solvent Contribution:
Polarizable Continuum Model (PCM)
The QM/MM Approach

Chromophore/Fluorophore Photophysics by Machine-Learning methods



Frank-Condon Energy Diagram

Build Initial Geometry

Goals:
- Faster
- More accurate

A Machine Learning Model

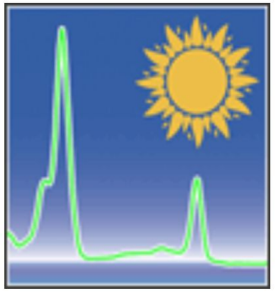
Structure-based Descriptors

λ_{abs}, f_{abs} , ε

λ_{em}, f_{em}, B

TD-DFT Calculation

The PhotochemCAD DataBase



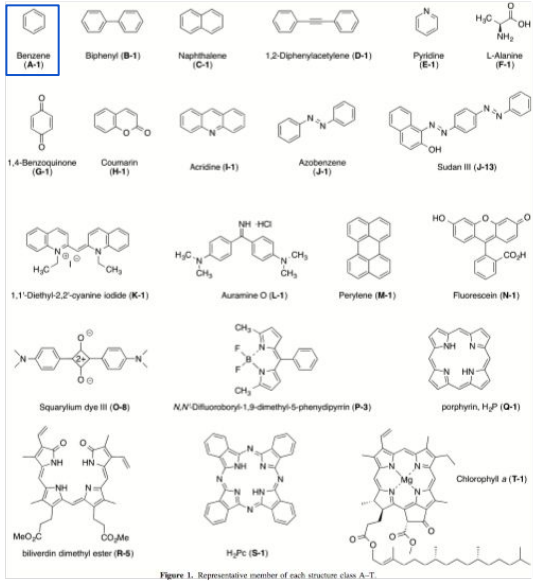
PhotochemCAD™

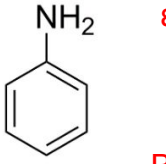
PhotochemCAD™ is a program of calculational modules and accompanying database of spectra aimed at advancing the photosciences. A streamlined version of the same database of spectra can now be viewed on our new website. The user is directed to the following journal articles for full description of the program and database:

http://www.photochemcad.com/PhotochemCAD.html

21 Categories
339 Molecules
339 Absorption Spectra
213 Fluorescence Spectra

Taniguchi, et al, Photochemistry and photobiology 94.2 (2018): 277-289.

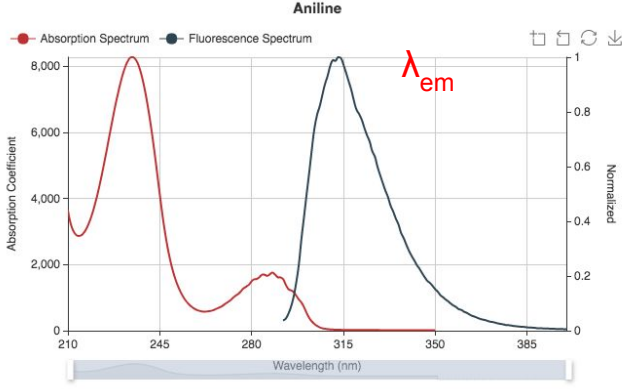




ε at λ_{abs} Absorption coefficient: 1760 at 288 nm

Φ_f Fluorescence quantum yield: 0.17

$B = \varepsilon \cdot \Phi_f$ Brightness



Use Cases

- Visualization of spectra
  - Structures in PhotochemCAD database
  - User input structures, and get predicted spectra from backend

UI

- Building customized ML models with user input data as the training set
- **Prediction with a user input structure**
  - Predict λ_{abs}, ε, λ_{em}, Φ_f, *etc.*
  - Generate their absorption/emission spectra
  - Use either the default ML model trained by PhotochemCAD data, or customized ML models

In this presentation we will mostly focus on the use of ML packages for prediction.

BackEnd

Machine Learning Methods

Existing packages

Theano

https://github.com/Theano/

28,000 commits.

Well regarded and ‘revolutionary’ learning resource amongst online communities.

The Montreal Institute of Learning Algorithms (MILA) will stop developing Theano.

Last update was 5 months ago.

Doesn’t meet our requirements; more for evaluation of mathematical expressions.

Attempt #2: XGBoost

~ 3,000 commits, 300 contributors

Regularly updated

Has a Scikit-Learn interface

First glance; well documented, easy to follow and implement

Not yet as well established as Theano or TensorFlow

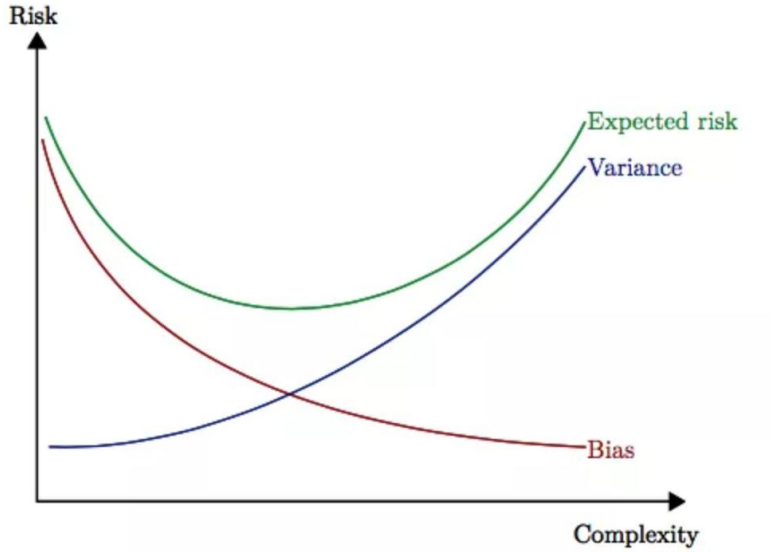
Attempt #2: XGBoost

https://github.com/dmlc/xgboost

- *“Efficient, flexible, portable.”* - good for mining classification and **regression**.
- Uses a gradient boosting framework;
  - model fits to the data by using multiple **simpler models** = ‘weak learner’, very **fast**.
  - builds and trains each tree individually but uses previous tree to correct errors.
- Highly **tunable** parametrization of the decision trees that comprise the boosting models
- May be more prone to overfitting

https://brage.bibsys.no/xmlui/bitstream/handle/11250/2433761/16128_FULLTEXT.pdf?sequence=1&isAllowed=y

Attempt #2: XGBoost



- Problem facing tree-boosting ML methods is the bias-variance trade-off when lowering ‘risk’
- Can compare XGBoost to older tree-boosting methods such as MART; uses NTB as opposed to GTB
- Seems to have a better trade-off between bias and variance as complexity of models increased.
- Growing fast as a result; well known for winning lots of ML competitions.

https://brage.bibsys.no/xmlui/bitstream/handle/11250/2433761/16128_FULLTEXT.pdf?sequence=1&isAllowed=y

Keras
https://keras.io/

- High level neural network
- Written in Python
- portable across all these backends: TensorFlow,CNTK, or Theano.
- CPU and GPU both (strong multi-GPU support and distributed training support)
- Bootstrapped common model, advanced packages available
  - Standard sequence data analysis(text): LSTM, GRU

    https://keras.io/layers/recurrent/

    Drawbacks: limited by the size of training data; hard to explain the result
  - Parse SMILES/InChI into different features: Add dense layer with no activation function on last layer (regression)

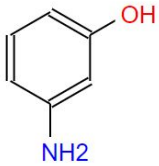
    https://keras.io/getting-started/sequential-model-guide/

    Drawbacks: hard to explore features

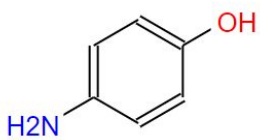
Keras
https://keras.io/

- Sequence data analysis(text): LSTM Long Short-Term Memory layer

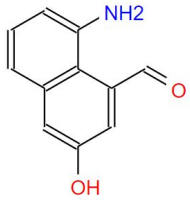
SMILES contains structure info that is represented as sequence data



Nc1cccc(O)c1



c1ccc(O)cc



Nc1cccc2cc(O)cc(C=O)c12

SMILES2vec, PNNL, 88% accuracy

In Proceedings of ACM SIGKDD Conference, London, UK, Aug 2018 (KDD 2018), 8 pages

- Features analysis

```
Number of benzene atoms: 6
Number of phenol atoms: 7
```

  - Number of aromatic atoms
  - Functional groups etc
  - Solvent
  - Might need some extra information by quick calculations

http://openbabel.org/wiki/Main_Page

https://docs.eyesopen.com/toolkits/python/oechemtk/molctordtor.html

TensorFlow

https://github.com/tensorflow/tensorflow.git :: Licenced under Apache License 2.0

What is it/ How does it work?

TensorFlow provides an easy method (python) for interacting with optimized c++ applications for training and running deep neural networks. It allows the creation of dataflow graphs through python and can work on most processing units (CPU, GPU, local computers, Google's TPUs, etc.) letting python handle the communication between the different c++ applications.

TensorFlow (Benefits/Drawbacks)

49637 commits with 1839 contributors (github)

Closing issues currently (2-20-19 10:15 pm)

Open-Source

Linux build supported - Current Windows/ Mac builds failing

Detailed install /build instructions

Able to interface with other high-level NN packages (e.g. Keras)

Training/ applying the model may not be as fast as packages such as CNTK

TensorFlow (Benefits/Drawbacks)

Mostly written in c++ with a python interface

Support/ guides for adding new features to the program (https://www.tensorflow.org/guide/)

3489 questions on Stack Overflow

Supports GPU/TPU acceleration

It can be difficult to obtain a fully deterministic model (the model can vary depending on the system it was trained on, though seems to more heavily affect GPU methods).

There is work going into addressing this issue and there are workarounds

Machine Learning Methods

Our package choice

TensorFlow?

- Well supported package
- Optimised code
- Run over other processing unit types
- Can work with other high-level NN packages (required as trying to map all the descriptor variables to the emission/absorption)

Visualization Tools

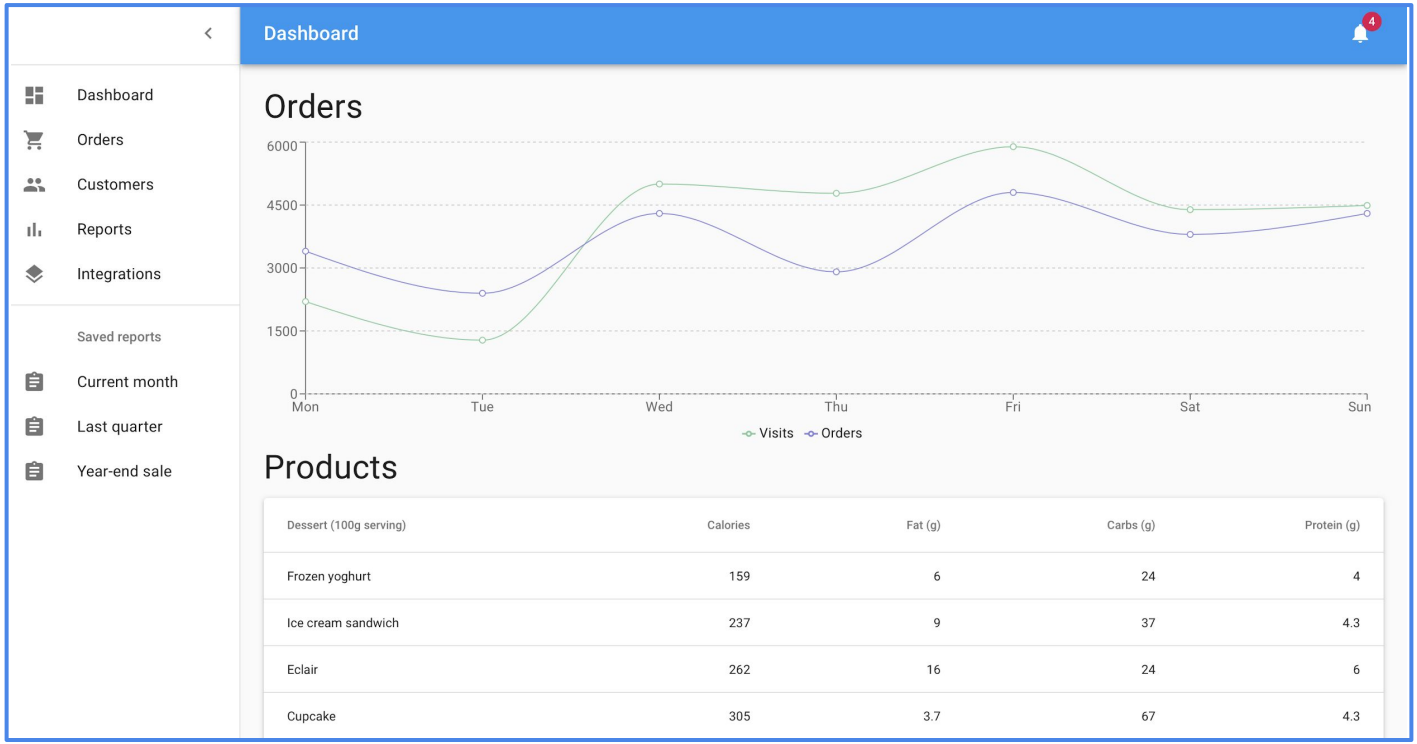
User Interface Libraries

Material-UI

Open-source project;

https://github.com/mui-org/material-ui

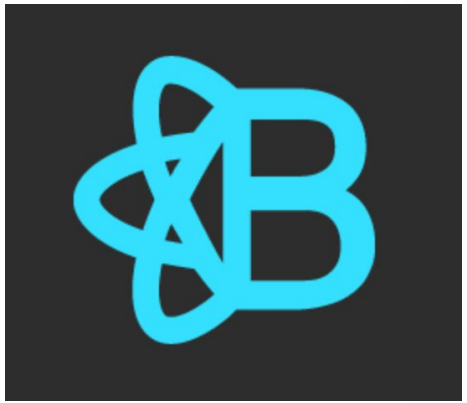
React component with Google's Material Design;



React Bootstrap

https://react-bootstrap.github.io/getting-started

Complete re-implementation of the Bootstrap components using React;



More examples for our package...

Filter

Toggle Filter

Sex

M | F | Both

Length of Dog's name

Short Names | Long Names | All

Minimal number of letters

0 | 3 | 5 | 10 | 20

Maximal number of letters

3 | 5 | 10 | 20 | Infinity

Here's a wrapped component

Using redux data

So, we are currently looking at 3549 dogs (after filtering)...

Let's add an interactive svg here, without d3:

50.85% of all dogs (filtering)

Sending events from text...

Names starting with L (mouseover here)... bla bla ... or short names (mouseover here)

Or send redux actions from text

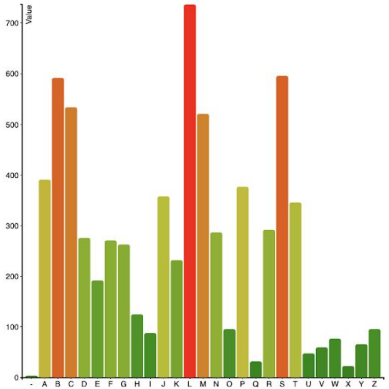
So female dogs (filter on mouseover!)... blah blah ... male dogs (filter on mouseover!)... blah blah ... both sexes (filter on mouseover!)

Using eventData from d3-react-squared

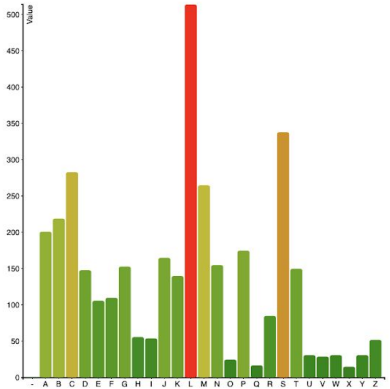
The last event was called mouseout on eventGroup "perLetter" and happened on Thu Feb 21 2019 06:54:04 GMT-0800 (PST)

Dog names by Letter

All Dogs

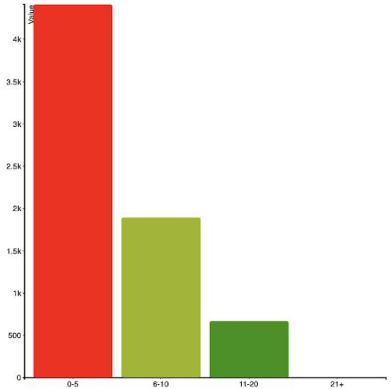


Filtered Data

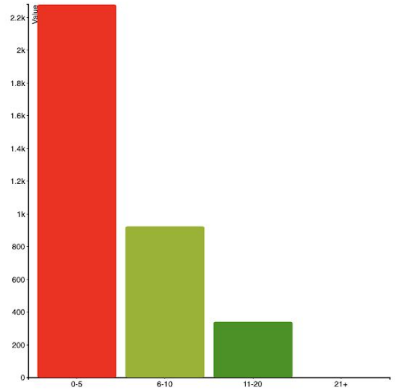


Dog names by Length

All Dogs



Filtered Data



Questions?