Machine Learning for Material Science

Jianzhi Yan, Harsh Asnani, SUNY POLYTECHNIC INSTITUTE

Abstract— In material science, machine learning is critical in areas such as new materials discovery and material property prediction. A machine learning method can be used to model the relationships between conditional factors and decision attributes based on a given sample. We discuss how machine learning applies to material science and navigate by example by taking into view drug discovery.

Index Terms— Autoencoders, Generative adversarial networks, MACCS fingerprints, Molecular activity prediction, Merck activity dataset.

INTRODUCTION: Using material properties data and advanced computer models, engineers can realistically simulate the behavior of new materials in specific applications and avoid lengthy cycles of build and test. These simulations cover a wide range of operating environments and length and time scales. The new field, called computational materials science, is one of the fastest growing areas within the field of chemistry and materials science. Machine learning automates analytical model building, using algorithms that iteratively learn from data, allowing computers to find hidden insights without being explicitly programmed.

I. Machine Learning for Predicting Properties

Regardless of the problem under study, a criterion for machine learning is existence of past data, either clean, curated and reliable data corresponding to the problem under study should already be available, or an effort is in place for the creation of the data.

In material science, a machine learning framework for predicting material properties, includes a dataset with attributes relating to a variety of materials that fall within a chemical class of interest, and a relevant measured or computed property of those materials i.e., the material, is referred to as "input", and the property of interest, is referred to as the "target" or "output." Thus, the learning problem is then defined as follows: Given a {materials → property} dataset, what is the best estimate of the property for a new material not in the original dataset?

An approach to understanding this problem, is to first represent numerically the various input cases (or materials) in the dataset. Each input case would have been reduced to a string of numbers, it is important to emphasize this step, because this is where one requires significant expertise and knowledge of the materials class and the application ('domain expertise").

The second step establishes a mapping between the features and the target property, and is entirely numerical in nature, largely devoid of the need for domain knowledge. Both the fingerprinting and mapping/learning steps are schematically. Several algorithms,

ranging from simple (e.g., linear regression) to highly sophisticated (kernel ridge 4 regression, decision trees, deep neural networks), are available to establish this mapping and the creation of surrogate prediction models. While some algorithms provide actual functional forms that relate input to output (e.g., regression), others do not (e.g., decision trees).

Basic Steps Of Machine Learning In Material Science:
Sample construction: Raw data is collected from computational simulations and experimental measurements. Mostly, data is incomplete, noisy and inconsistent, hence, data cleaning should be performed when constructing a sample from raw data.
Model building: Input data is linked to output data using a set of nonlinear or linear functions. In materials science, complex relationships usually exist between the conditional factors and the target attributes, which traditional methods have difficulty handling.
Model evaluations: A data-driven model should achieve good performance not only on existing data but also on unseen data. Generally, we can evaluate the generalization errors of models by means of calculation-based tests and use the results to select the best one.

II. Drug Discovery using Machine Learning:

Getting a new drug to the market is a long and tedious process; it can take many years or even decades. There are all sorts of experiments, clinical studies, and clinical trials that you have to go through. And about 90% of all clinical trials in humans fail even after the molecules have been successfully tested in animals.
The process is as follows:
the doctors study medical literature, in particular associations between drugs, diseases, and proteins published in other papers and clinical studies, and find out what the target for the drug should be, i.e., which protein it should bind with;
after that, they can formulate what kind of properties they want from the drug: how soluble it should be, which specific structures it should have to bind with this protein, should it treat this or that kind of cancer;
then they sit down and think about which molecules might have these properties; there is *a lot* to choose from on this stage: e.g., one standard database lists 72 million molecules, complete with their formulas, some properties and everything; unfortunately, it doesn't always say whether a given molecule cures cancer, this we have to find out for ourselves;
then their ideas, called *lead molecules*, or *leads*, are actually sent to the lab for experimental validation;
if the lab says that the substance works, the whole clinical trial procedure can be initiated; it is still very long and tedious, and only a small percentage of drugs actually go all the way through the funnel and reach the market, but at least there is hope.
We can use machine learning models to try and choose the molecules

.

that are most likely to have desired properties. But when we have 72 million of something, “choosing” ceases to look like classification and gets more into the “generation” part of the spectrum.

The idea is to use Adversarial Autoencoders- a type of Generative Adversarial Network.

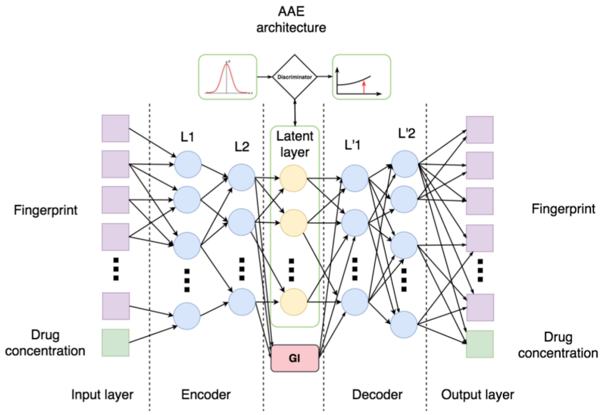

Image source: (Kadurin et al., 2017)

In AAE, the idea is to learn to generate objects from their latent representations. Idea is that in the middle of the architecture, the input must go through a middle layer that learns a latent representation, i.e., a set of features that succinctly encode the input in such a way that afterwards subsequent layers can decode the object back:
Either the middle layer is simply smaller (has lower dimension) than input and output, or the autoencoder uses special regularization techniques, but in any case, it’s impossible to simply copy the input through all layers, and the autoencoder has to extract the really important stuff.

Basically, the problem becomes to “translate” the condition, i.e., desired properties of a molecule, into more “low-level” properties of the molecular structure encoded into their MACCS fingerprints. Then a simple screening of the database can find molecules with the fingerprints most similar to generated ones.

III. Predicting Molecular Activity with Merck Dataset

Merck has datasets of these fingerprints and activity of a lot of molecules against the targets listed as number. We use one of the dataset which enlists some molecules and their activity against a specific target to train a neural network with supervised learning and predict activity. This way we can select potential candidates for drug testing in the lab. Merck provides descriptors of a molecule as numeric data in feature columns.

We use a particular dataset and tryout various techniques as listed in detail below. As recommended by Merck, we test the efficiency of our model by R^2 metric against the average R^2 scores reported by Merck.

We conduct two rounds. In particular

Round 1: We use two hidden layers to separate our data. This round tests various combinations of neurons in those layers. We refer to [4] as a guideline for trying 4 combinations.
[neurons in first layer, neurons in second layer]
[5,2], [10,5], [50,25], [100,50].
We conduct 10 trials for each combination and get R^2 values.

Looking at the results, [50,25] looks like the only somewhat stable choice.

Round 2: We try three kind of features transformation namely; binary, logarithmic and none. We use a neural network with best results from round 1. The transformations are compared by R^2 again. The plot is below.

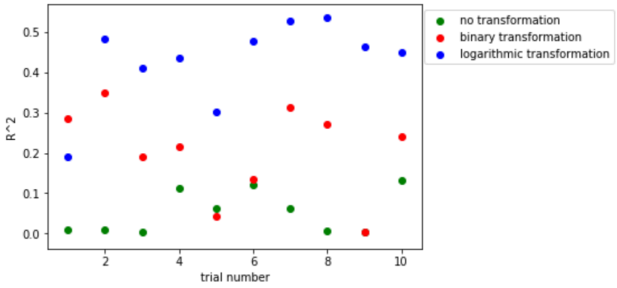

fig: trained with [50,25] neurons

The figure suggests logarithmic transformation works best.

Round 3: We choose the neurons from first round’s result and feature transformation from second round’s result and present our network the R^2 for which is: 0.43 from average of 10 trials and peaks at 0.54.

A dropout of 25% is used in training to avoid overfitting.
Both hidden layers and output layer use RELU as activation function.
Learning rate of 0.001 is used by researching discussion boards on Merck.

IV. Conclusion

We trained a model with 43% accuracy for a single target molecule. Current average on Merck’s leader board over all datasets which include 15 target molecules in 15 different datasets is 48%.
The steps described can be used over a dataset combining all molecules or a specific model per dataset can also be developed. Due to limited computational power, we were only able to model for one target molecule.
Please refer to notebook attached.
Round 1 is in NN hidden neurons.ipynb
Round 2 is in Transformations.ipynb
Final model is in FinalModelLog[50,25].ipynb

References

[1] Jean-leah Njoroge, "How is Machine Learning Applicable in Material Science", *http://www.jeannjoroge.com/How-is-Machine-Learning-Applicable-in-Material-Science/*,2017

[2] Sergey Nikolenko, Creating Molecules from Scratch I: Drug Discovery with Generative Adversarial Networks, *Neuromation,* 2018

[3] Mariya Popova, Olexandr Isayev, Alexander Tropsha. *Deep Reinforcement Learning for de-novo Drug Design*. Science Advances, Vol. 4, no. 7, eaap7885. DOI: 10.1126/sciadv.aap7885, 2018

[4] Predicting Molecular Activity Using Deep Learning in TensorFlow, https://towardsdatascience.com/predicting-molecular-activity-using-deep-learning-in-tensorflow-f55b6f8457f9