CIS 7000-005: Debugging Data & Models Fall 2022

Explainabilty — September 29

Prof. Eric Wong

To find problems in our models, we need ways to inspect and debug them. How can we do this for complex models often seen as opaque? This problem is called explainability in machine learning.

- What is an ideal explainability method?
- What do explainability methods currently provide?

1 Desiderate for explainabilty

What do we want out of an explanation? There are tons of different criteria one could hope for. Here's a few:

- Faithfulness - explanations should reflect an actual change in the model
- Usability - explanation should be meaningful to the end user
- Necessity - model needs the identified explanation to do a prediction
- Sufficient - explanation covers the entire model's prediction

1.1 Interpretabile by design

Some models are "interpretable by design".

Linear models

$$y = \beta^T x + \beta_0$$

What is the claim of interpretability here? it depends on the feature and the R^2.

$$R^2 = 1 - SSE/SST$$

where $SSE = \sum_i (y^i - \hat{y}^i)^2$ and $SST = \sum_i (y^i - \bar{y})^2$. If the R^2 value is high, then the the weights of the linear model reflect the variation in the data. Otherwise, if R^2 is low, then little variance is explained by the linear. In this case, the weights of the linear model can still technically be interpreted but the interpretation is virtually meaningless.

Logistic regression Linear models are typically used for regression tasks. In the case of classification, we usually use a logistic regression model instead. Is this as interpretable as a linear regression model?

$$P(y=1) = \frac{1}{1+\exp(-(\beta^T x))}$$

How to interpret the weights now? Take a log transform of the odds:

$$\ln \frac{P(y=1)}{1-P(y=1)} = w^T x \tag{1}$$

Lets call this the log odds. What happens if we increase the unit of β_i by one?

$$\frac{Odds_{x_j+1}}{Odds_{x_j}} = \exp(\beta_j) \tag{2}$$

So one unit of change corresponds to an increase in the "log odds ratio" by β_j.

1.2 LIME

Suppose we are happy with linear models. We can try to fit a local surrogate linear model to explain local decision boundaries. For a specific example x, model f, and linear model class G, we can solve

$$\min_g \ell(f, g; x) + \Omega(g)$$

- Select x to explain
- Perturb x locally to get a dataset of $(x_i, f(x_i))$
- Weight each x_i according to its distance to x with w_i
- Train a g on the weighted dataset
- Interpret g

For example, we can use LASSO:

$$\min_w \|w^T X - y\| + \|w\|_1$$

The first term here fits a linear model to the dataset, while the second ℓ_1 regularizer encourages a simple solution via sparsity.

Assumptions of LIME. At a first glance, interpreting with LIME appears to be a simple task—simply interpret the linear model. What assuptions are we making here?

1. We are assuming that the linear model is a good fit for the perturbed data, i.e. tha the R^2 is high and that the linear model achieves good accuracy. In other words, that the network is locally linear in the region of perturbed inputs. This encourages faithfulness.

2. We assume a specific sparsity threshold to interpret, i.e. the number of features with non-zero weights in the linear model. This is hopefully sparse enough to be usable.

But how many features do we interpret, or how sparse should we make the linear model?

2 References

Notes largely sourced from `https://christophm.github.io/interpretable-ml-book/`