Interpretable Machine Learning

Patrick Hall
Dec. 11, 2016

H$_2$O.ai

- I have said before that machine learning is uninterpretable – but I was wrong
- Nonlinear, nonpolynomial models
- ML models capture high degree interactions
- Allow a variables impact on the model predictions and interactions with other variables to change in complex ways over the variable's domain
- Interpretability is about trust and understanding – ways to increase trust and understanding

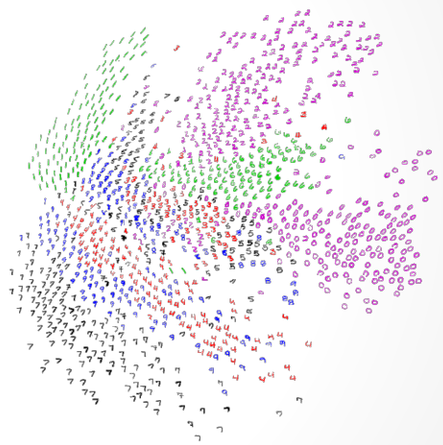
- This talk needs a white board

Contents

Part 1: Seeing all of your data
- Glyphs
- Correlation graphs
- 2-D projections

Part 2: Using machine learning in regulated industry
- OLS regression alternatives
- Build toward ML model benchmarks
- ML in traditional analytics processes
- Small, interpretable ensembles

Part 3: Understanding complex ML models
- Surrogate models
- LIME
- Maximum activation analysis
- Constrained neural networks
- Variable importance measures
- Partial dependence plots
- TreeInterpreter
- Residual analysis



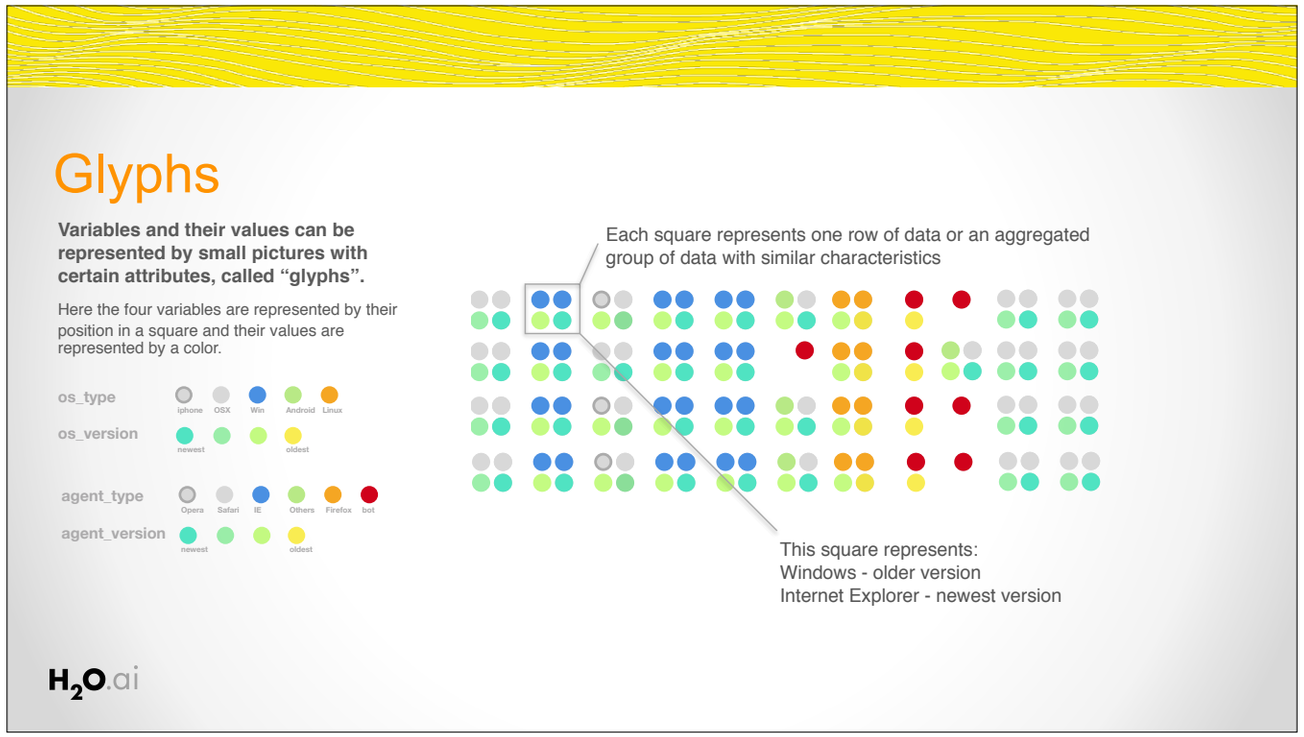
H2O.ai

Part 1: Seeing all of your data

Most real data sets are hard to see because they have many variables and many rows. Like most sighted people, I rely on my visual sense quite heavily for understanding information. For me, seeing data is basically tantamount to understanding data. Moreover, I can really only understand two or three visual dimensions, preferably two, and something called change blindness frustrates human attempts to reason analytically given information split across different pages or screens. So if a data set has more than two or three variables or more rows than can fit on a single page or screen, it's realistically going to be hard to understand what's going on in there without resulting to more advanced techniques than scrolling through rows and rows of data.

Why is seeing a data set important for creating understanding and trust in machine learning results? In machine we are attempting to model relationships in a data set. If we can see and better understand the data set and the relationships in it and we can find those relationships represented in our machine learning results, it's a basic sanity check that a model is working correctly.

Of course there are many, many ways to visualize data sets. I like the techniques highlighted below because they help illustrate all of a data set, not just univariate or bivariate slices of a data set (meaning one or two variables at a time). This is important in machine learning because most machine learning algorithms automatically model high degree interactions between variables (meaning the effect of combining many, i.e. way more than two, variables together). Of course traditional univariate and bivariate tables and plots are still important and you should use them, I just think they are slightly less helpful in understanding nonlinear models that can pick up on arbitrarily high degree interactions between independent variables.



Glyphs are visual symbols used to represent a data. The color, texture, or alignment of a glyph can be used to represent different values or attributes of data. In figure 1, colored circles are defined to represent different types of operating systems and web browsers. When arranged in a certain way, these glyphs can be used to represent rows of a data set.

Figure two gives an example of how glyphs can be used to represent rows of a data set. Each grouping of four glyphs can be either a row of data or an aggregated group of rows in a data set. The highlighted Windows/Internet Explorer combination is very common in the data set and so is the OS X and Safari combination. It’s quite likely these two combinations are two compact and disjoint clusters of data. We can also see that in general operating system versions tend to be older than browser versions, and that using Windows and Safari is correlated with using newer operating system and browser versions whereas Linux users and bots are correlated with older operating system and browser versions. The red dots that represent queries from bots standout visually (unless you are red-green colorblind …). Using bright colors or unique alignments for events of interest or outliers is a good method for making important or unusual data attributes readily apparent in a glyph representation.

How do glyphs enhance understanding?

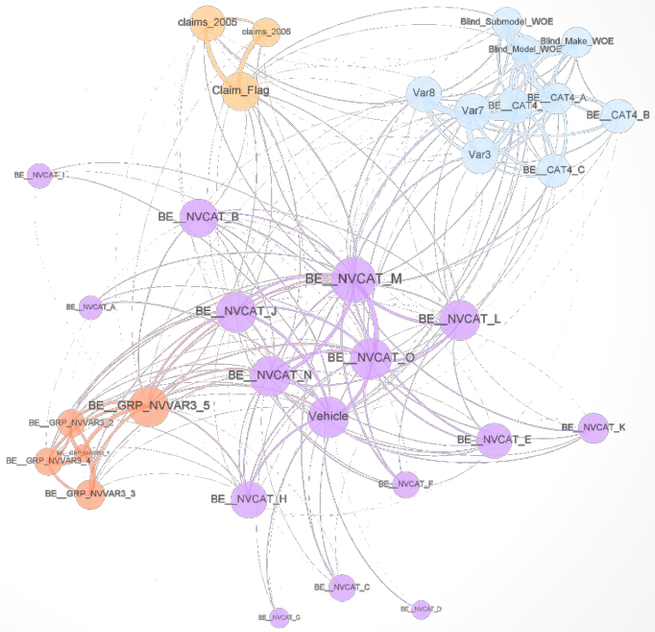
For most people, glyph representations of structures (clusters, hierarchy, sparsity, outliers) and relationships (correlation) in a data set are easier to understand than scrolling through plain rows of data and looking at variables values.

How do glyphs enhance trust?

Seeing structures and relationships in a data set usually makes those structures and relationships easier to understand. An accurate machine learning model should create answers that are representative of the structures and relationships in a data set. Understanding the structures and relationships in data set is a first step to knowing if a model’s answers are trustworthy.

Correlation Graphs

The nodes of this graph are the variables in a data set. The weights between the nodes are defined by the absolute value of their pairwise Pearson correlation.



H2O.ai

A correlation graph is a two dimensional representation of the relationships (correlation) in a data set. While many details regarding the display of a correlation graph are optional and could be improved beyond those chosen for figure 3, correlation graphs are a very powerful tool for seeing and understanding relationships (correlation) between variables in a data set. Even data sets with tens of thousands of variables can be displayed in two dimensions using this technique.

In figure 3, the nodes of the graph are the variables in an anonymized auto insurance claims data set and the edge weights (thickness) between the nodes are defined by the absolute value of their pairwise Pearson correlation. For visual simplicity, weights below a certain threshold are not displayed. The node size is determined by a node's number of connections (node degree), node color is determined by a graph communities calculation, and node position is defined by a graph force field algorithm.

The dependent variable in the data set represented by figure 3 was Claim_Flag, non-zero auto insurance claims in 2007. While many variables are correlated with one another, Claim Flag is only weakly correlated with most other variables in the data set, except for claims_2005 and claims_2006. Figure 3 tells us that a good model for Claim Flag would likely emphasize claims from the previous years and their interactions very heavily, a good model would likely give some emphasis to the BE__NVCAT family of variables and perhaps their interactions, and would likely ignore most other variables in the data set.

How do correlation graphs enhance understanding?

For most people, correlation graph representation of relationships (correlation) in a data set are easier to understand than scrolling through plain rows of data and looking at variables values, especially for data sets with many variables.

How do correlation graphs enhance trust?

Seeing relationships in a data set usually makes those relationships easier to understand. An accurate machine learning model should create answers that are representative of the relationships in a data set, and understanding the relationships in a data set is a first step to knowing if a model's answers are trustworthy.

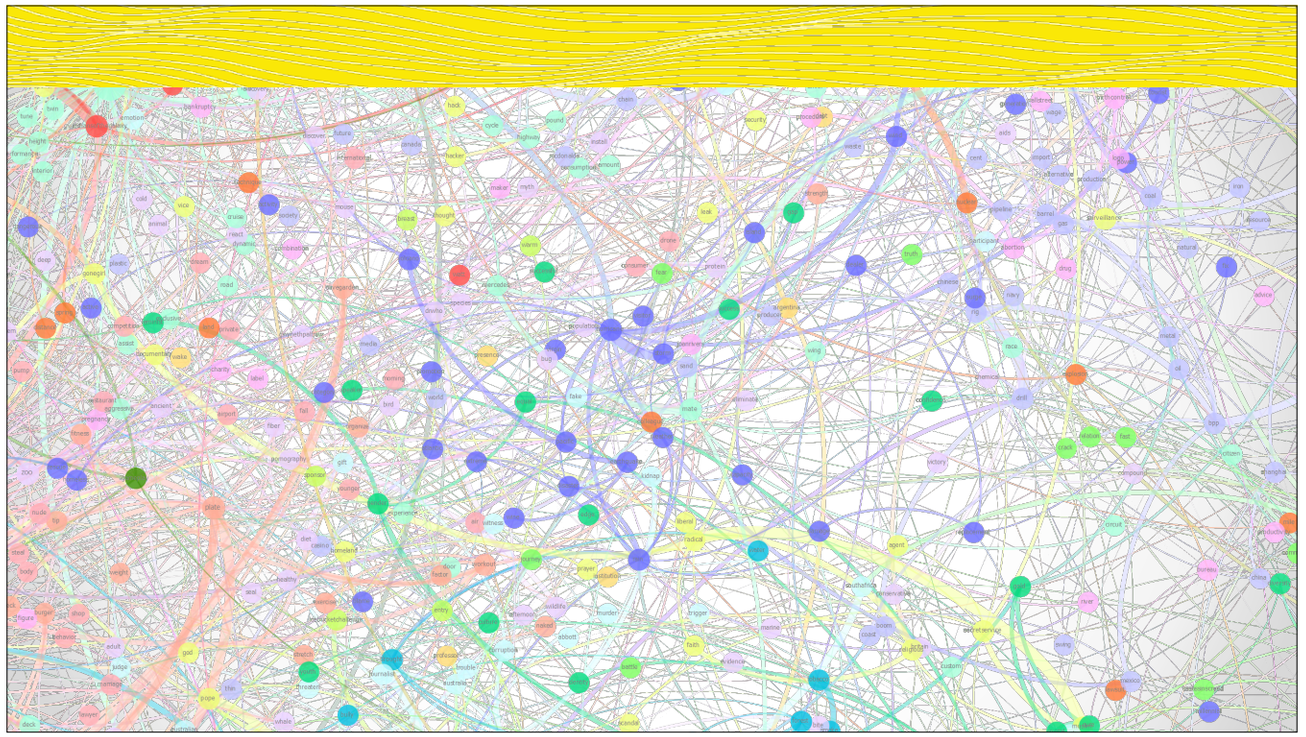
The graph in figure 3 was created with Gephi, http://www.gephi.org

Ideas on the internet - Fall 2014

Each node is a NMF feature labeled by it's top contributing term.

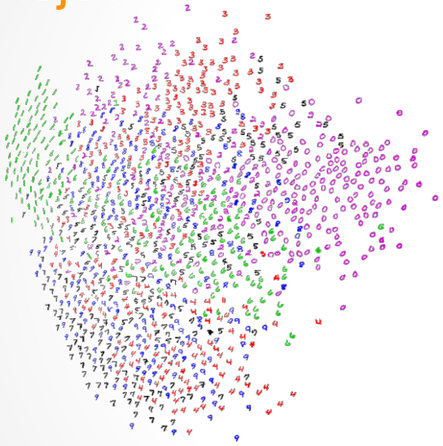
H_2O.ai

Data is courtesy of szl.it (now https://www.tanjo.net/).

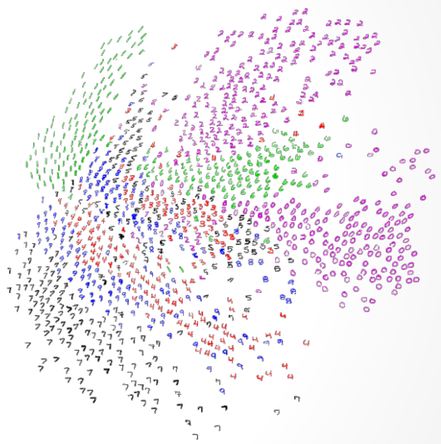


Close up for detail

2-D projections



784 dimensions to 2 dimensions
with PCA



784 dimensions to 2 dimensions
with autoencoder network

H2O.ai

Source: http://www.cs.toronto.edu/~hinton/absps/science_som.pdf

There are many techniques for projecting the rows of a data set from a usually high-dimensional original space into a more visually understandable lower-dimensional space, ideally two or three dimensions. Popular techniques include:

Principal Component Analysis (PCA)
Multidimensional Scaling (MDS)
t-SNE (t-distributed Stochastic Neighbor Embedding)
Autoencoder networks

Each of these techniques have strength and weaknesses, but the key idea they all share is to represent the rows of a data set in a meaningful low dimensional space. When a data set has more than two or three dimensions, visualizing it with a scatter plot becomes essentially impossible, but these techniques enable even high-dimensional data sets to be projected into a representative low-dimensional space and visualized using the trusty, old scatter plot. A high quality projection visualized in a scatter plot should exhibit key structural elements of a data set such as clusters, hierarchy, sparsity, and outliers.

In figure 4, the famous MNIST data set is projected from its original 784 dimensions onto two dimensions using two different techniques, PCA and autoencoder networks. The quick and dirty PCA projection is able to separate digits labeled as zero from digits labeled as one very well. These two digit classes are projected into fairly compact clusters, but the other digit classes are generally overlapping. In the more sophisticated, but also more computer-time-consuming, autencoder projection all the digit classes appear as separate clusters with visually similar digits appearing close to one another in the reduced two-dimensional space. The autoencoder projection is capturing the clustered structure of the original high-dimensional space and the relative locations of those clusters. Interestingly, both plots are able to pick up on a few outlying digits.

How do 2-D projections enhance understanding?

For most people, 2-D projections of structures (clusters, hierarchy, sparsity, outliers) in a data set are easier to understand than scrolling through plain rows of data and looking at variable’s values.

How do 2-D projections enhance trust?

Seeing structures in a data set usually makes those structures easier to understand. An accurate machine learning model should create answers that are representative of the structures in a data set. Understanding the structures in a data set is a first step to knowing if a model’s answers are trustworthy.

Projections can add an extra and specific degree of trust if they are used to confirm machine learning modeling results. For instance if known hierarchies, classes, or clusters exist in training or test data sets and these structures are visible in 2-D projections, it is possible to confirm that a machine learning model is labeling these structures correctly. A secondary check is to confirm that similar attributes of structures are projected relatively near one another and different attributes of structures are projected relative far from one another. Consider a model used to classify or cluster marketing segments, it is reasonable to expect a machine learning model to label older, richer customers differently than younger, less affluent customers, and moreover to expect that these different groups should be relative disjoint and compact in a projection, and relatively far from one another. Such results should also be stable under minor perturbations of the training or test data, and projections from perturbed vs. non-perturbed samples can be used to check for stability.

Part 2: Using ML in regulated industry

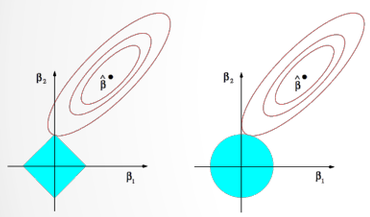
H2O.ai

For analysts and data scientists working in regulated industries, the potential boost in predictive accuracy provided by machine learning algorithms may not outweigh the current realities of internal documentation needs and external regulatory regimes. For these practitioners, traditional linear modeling techniques may be their only option for predictive modeling. However, the forces of innovation and competition don't stop because you work under a regulatory regime. Data scientists and analysts in the regulated verticals of banking, insurance, and other similar industries face a particular conundrum. They have to find ways to make more and more accurate predictions, but keep their models and modeling processes transparent and interpretable.

The techniques presented in this section are newer types of linear models or they use machine learning to augment traditional, linear modeling methods. They're meant for practitioners who just can't use machine learning algorithms to build predictive models because of interpretability concerns. They produce results similar, if not identical, to traditional linear models, but with a boost in predictive accuracy provided by machine learning algorithms.

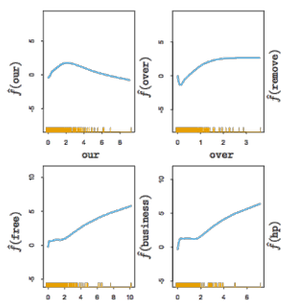
OLS regression alternatives

Penalized Regression



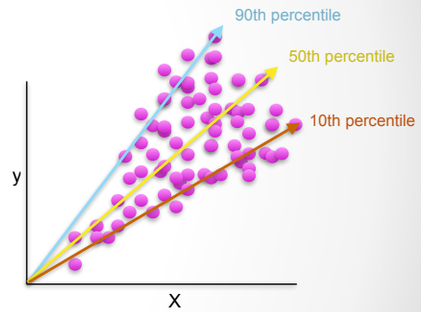
- Fewer assumptions
- Well suited for **N << p**
- No multiple comparison issues during variable selection
- Preserves interpretability by selecting a small number of variables (L1 penalty)

Generalized Additive Models



- Fit linear terms to certain variables
- Fit nonlinear splines to other variables
- Hand-tune a trade-off between interpretability and accuracy

Quantile Regression



- Fit an interpretable linear model to different percentiles of training data
- Find different sets of drivers across percentiles of an entire customer market or portfolio of accounts

H_2O.ai

Source: http://statweb.stanford.edu/~tibs/ElemStatLearn/printings/ESLII_print10.pdf

Penalized regression

Ordinary least squares (OLS) regression is about 200 years old. Maybe it's time to move on? If you're interested, penalized regression techniques can be a gentle introduction to machine learning. Contemporary penalized regression techniques usually combine L1/LASSO penalties and L2/ridge penalties in a technique known as elastic net. They also make fewer assumptions about data than OLS regression.

Figure 5: Shrunken feasible regions for L1/LASSO penalized regression parameters (left) and L2/ridge penalized regression parameters (right).

Instead of solving the classic normal equation or using statistical tests for variable selection, penalized regression minimizes constrained objective functions to find the best set of regression parameters for a given data set that also satisfy a set of constraints or penalties. You can learn all about penalized regression in Elements of Statistical Learning, but for our purposes here, its just important to know when you might want to try penalized regression. Penalized regression is great for wide data, even data sets with more columns than rows, and for data sets with lots of correlated variables. L1/LASSO penalties drive unnecessary regression parameters to zero, avoiding potential multiple comparison problems that arise in forward, backward, and stepwise variable selection, but still picking a good, small subset of regression parameters for a data set. L2/ridge penalties help preserve parameter estimate stability, even when many correlated variables exist in a wide data set or important predictor variables are correlated. It's also important to know penalized regression techniques don't usually create confidence intervals or t-test p-values for regression parameters. These types of measures are typically only available through empirical bootstrapping experiments that require a lot of extra computing time.

Generalized Additive Models (GAMs)

Generalized Additive Models (GAMs) enable you to hand-tune a tradeoff between accuracy and interpretability by fitting standard regression coefficients to certain variables and nonlinear spline functions to other variables. Also most implementations generate convenient plots of the the fitted splines. In many cases you may be able to eyeball the fitted spline and switch it out for a more interpretable polynomial, log, trigonometric or other simple function of the predictor variable. You can learn more about GAMs in Elements of Statistical Learning too.

Figure 6: Spline functions for several variables created by a generalized additive model.

Quantile regression

Quantile regression allows you to fit a traditional, interpretable, linear model to different percentiles of your training data, allowing you to find different sets of variables with different parameters for modeling different behaviors across a customer market or portfolio of accounts. It probably makes sense to model low value customers with different variables and different parameter values from those of high value customers, and quantile regression provides a statistical framework for doing so.

Figure 7: A diagrammatic representation of quantile regression in two dimensions.

How do alternative regression techniques enhance understanding and trust?

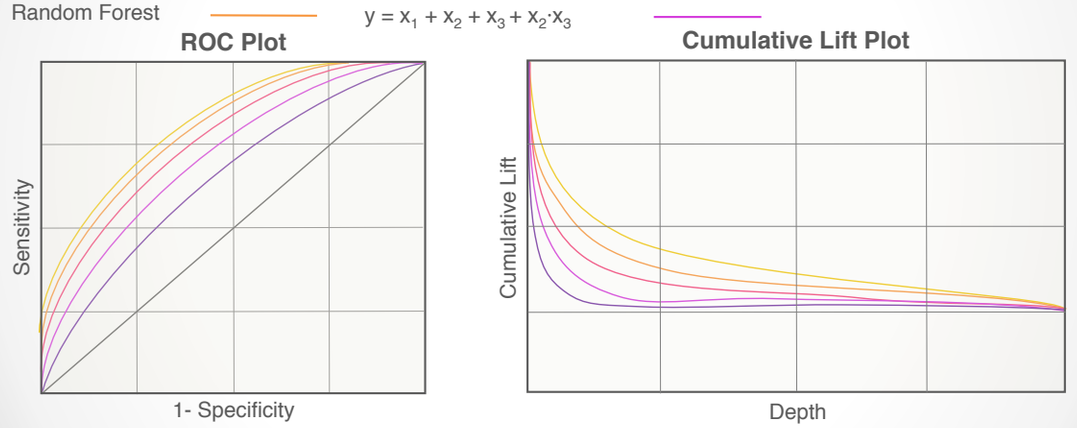
Basically these techniques are plain old understandable, trusted linear models, but used in new and different ways. It's also quite possible that the lessened assumption burden, the ability to select variables without problematic multiple statistical significance tests, the ability to incorporate important but correlated predictors, the ability to fit nonlinear phenomena, or the ability to fit different quantiles of the data's conditional distribution (and not just the mean of the conditional distribution) could lead to more accurate models and more accurate understanding of modeled phenomena.

Build toward ML model benchmarks

Gradient Boosting — $y = x_1 + x_2 + x_3 + x_1 \cdot x_3 + x_2 \cdot x_3$

Random Forest — $y = x_1 + x_2 + x_3 + x_2 \cdot x_3$

$y = x_1 + x_2 + x_3$



Machine learning models typically incorporate a large number of implicit variable interactions and easily fit nonlinear, non-polynomial patterns in data. If a traditional regression model is much less accurate than a machine learning model, the traditional regression model may be missing important interactions or a piecewise modeling approach maybe necessary.

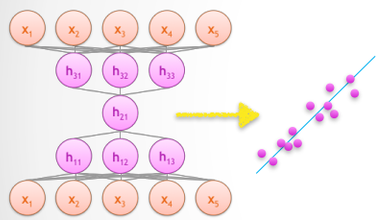
H_2O.ai

How does it increase trust and understanding?
It helps us make our understandable, trust worthy models more accurate

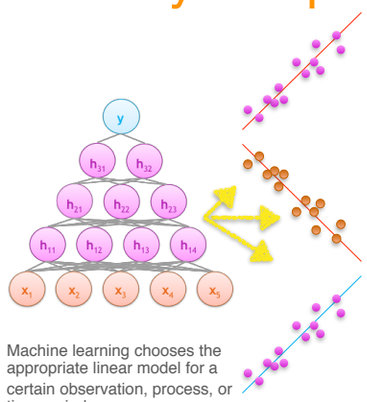
- Machine learning models often take into consideration a large number of implicit variable interactions
- If your regression model is much less accurate than your ML model, you've probably missed some important interaction(s)
- Decision trees area great way to see the potential interactions
- Important interactions may only be occurring at certain values of certain variables

ML models intrinsically allow:
- high degree interactions between input variables - include 2nd, 3rd degree interactions to approximate
- nonlinear, nonpolynomial behavior across the domain of a single input variable - use piecewise models to approximate

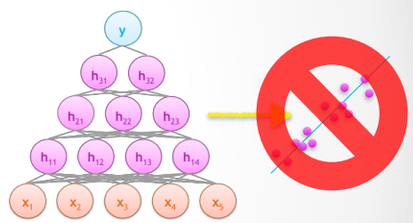
ML in conventional analytical processes



Machine learning extracts nonlinear, nonpolynomial features or high degree interactions for use as inputs to a linear model.



Machine learning chooses the appropriate linear model for a certain observation, process, or time period.



Machine predicts when a deployed linear model will become stale.

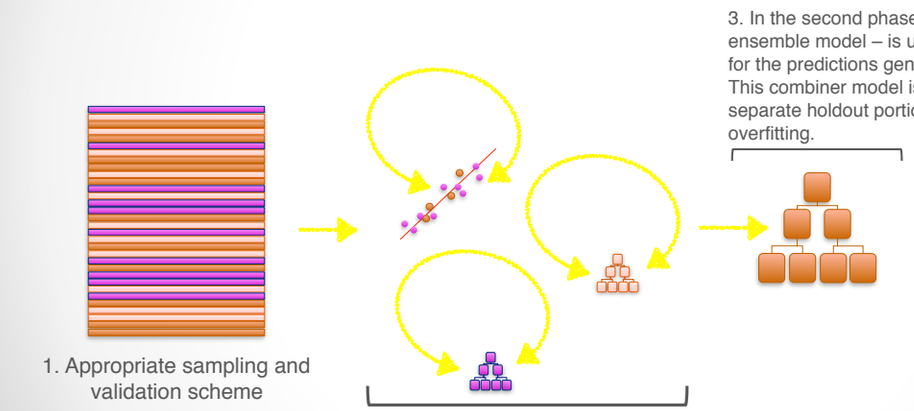
H2O.ai

How does it increase trust and understanding?
It helps us make our understandable, trust worthy models more accurate and helps us use them more efficiently

- Introduce nonlinear predictors into the model
- (Predictors that capture more complex, nonlinear, nonpolynomial relationships)

Based on past model performance data:
- Use an ML model as a gate to pick which linear model to use
- Use a machine learning model to predict when traditional deployed models need to be retrained or replaced before their predictive power lessen

Small interpretable ensembles



H2O.ai

How does it increase trust and understanding?
It allows us to boost the accuracy of traditional trustable models without sacrificing too much interpretability
It increases trust if models compliment each other in expected ways, e.g.
A logistic regression model that is good at rare events slightly increases a good decision tree model that is not good at rare events in the presence of rare events

- Probably the most important recent breakthrough in machine learning aside from deep learning
- Combining predictions between a handful of good, but different, models often results in better predictions
- Ex: train an interpretable regression model and an interpretable decision tree and average their predictions
- Different over sampling in each model very useful for rare events
- Hill climbing
- Stacking:
    - A linear model is often used to optimally weight the predictions of several different models that are then assembled together
    - Cross validation

Van der Laan 2007 reference: http://biostats.bepress.com/ucbbiostat/paper222/

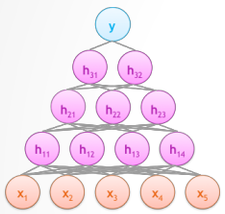
Part 3: Understanding complex ML models

H2O.ai

For those who can use any type of ML model, this is how can they explain their behavior.

Probably a combination of these techniques works best.

Surrogate models

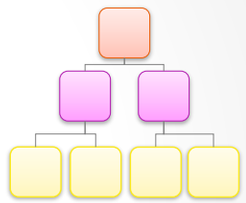


y	x_1	x_2	x_3
1	2.54	1.65	0.02
0	1.14	0.70	0.82
1	0.99	0.51	2.11
⋮	⋮	⋮	⋮

Target vector Input vectors

First fit a complex machine learning model to your training data.

Then train a single decision tree on the original training data, but instead of using the actual target in the training data, use the predictions of the more complex algorithm as the target for this single decision tree.*



y	$\hat{y}$	x_1	x_2	x_3
1	0.98	2.54	1.65	0.02
0	0.32	1.14	0.70	0.82
1	0.85	0.99	0.51	2.11
⋮	⋮	⋮	⋮	⋮

Predicted target vector

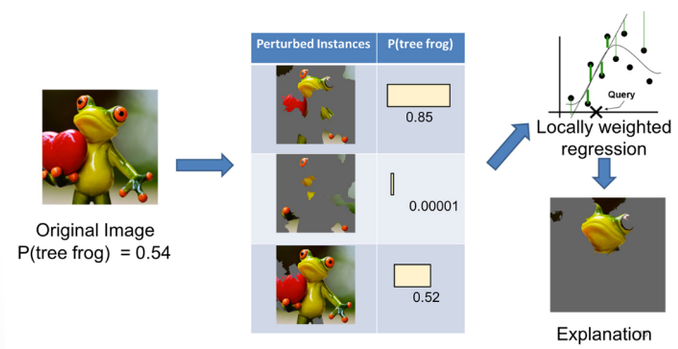
H_2O.ai

How does it increase trust and understanding?
It helps us understand the inner workings of a complex system
It increases trust if we can see the logic in the surrogate model matches our domain experience or expectation
It increases trust if the logic is stable under mild perturbations of the data

- Interpretable models used as a proxy to explain complex models
- For example:
    - Fit a complex machine learning model to your training data.
    - Then train a single decision tree on the original training data, but use the predictions of the more complex algorithm as the target for this single decision tree
    - This single decision tree will likely* be a more interpretable proxy you can use to explain the more complex machine learning model

* Few (possibly no?) theoretical guarantees that the surrogate model is highly representative of the more complex model

Local Interpretable Model-Agnostic Explanations (LIME)



H2O.ai

Source: https://www.oreilly.com/learning/introduction-to-local-interpretable-model-agnostic-explanations-lime

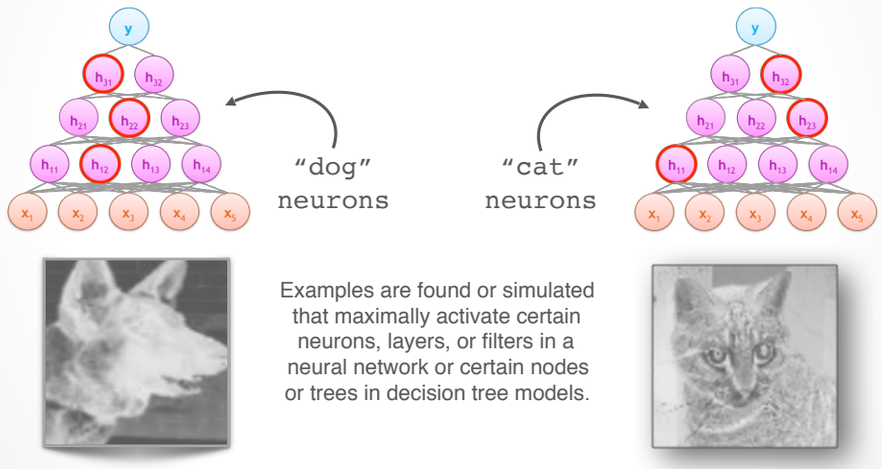
How does it increase trust and understanding?
It helps us understand the predictions made for key observations
It helps us understand the behavior of the model at local, important places no matter how complex the global model is
It increases trust because we can see how the model makes decisions for key observations
It increases trust if we see decisions being made about similar observations being made in similar ways

- Pick or simulate ‘marker’ records/examples
- Score them for probability or value of target with complex ML model
- Choose a ‘query’ record/example with a prediction to be explained
- Weight ‘marker’ records/examples closest to the query record/example
- Train an L1 regularized linear model on the data set of ‘marker’ records/examples
- The parameters of the linear model will help explain the prediction for the ‘query’ record/example

- Local surrogate model + more structured type of activation analysis
- You can include ‘marker’ records/examples in your training data
- For traditional analytics data, explanatory data samples could potentially be simulated – e.g. customers with highest, lowest, and median credit scores

https://www.oreilly.com/learning/introduction-to-local-interpretable-model-agnostic-explanations-lime
https://arxiv.org/pdf/1606.05386.pdf

Maximum activation analysis



Examples are found or simulated that maximally activate certain neurons, layers, or filters in a neural network or certain nodes or trees in decision tree models.

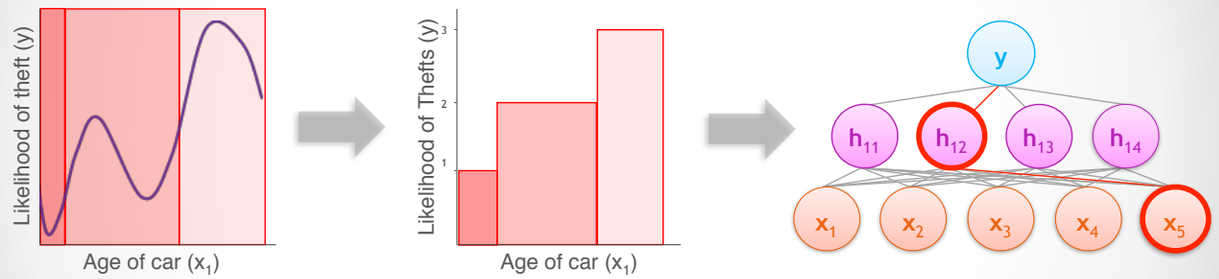
H2O.ai

How does it increase trust and understanding?
It increases understanding because it elucidates the structure of the model
(If we have dogs and cats in our data we would expect certain neurons to maximally learn certain visually features, i.e. dog nose neuron is activated for all dog picks, but not in cat pictures)
It increases understanding because see interactions when input units activate the same hidden unit consistently
It increases trust if we see stability in what units are activated for similar inputs
It increases trust if similar data points proceed through the model in the same way
It increases trust if interactions and structure match

- Which data creates the maximum output from certain neurons
- Which neurons create the maximum output for some archetypal data example
- You can include 'marker' records/examples in your training data

http://yosinski.com/deepvis
http://yosinski.com/media/papers/Yosinski__2015__ICML_DL__Understanding_Neural_Networks_Through_Deep_Visualization__.pdf

Constrained neural networks



Age of car is a nonnegative quantity, but is not monotonic wrt to car theft, the target in this example.

Through a binning scheme, age of car can be transformed to be monotonically increasing with the target.

When all inputs are nonnegative, monotonic wrt to the target, and model weights are constrained to be nonnegative it's easier to parse extra information from machine learning models.

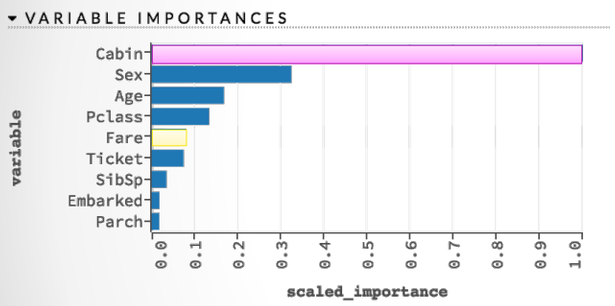
H2O.ai

How does it increase trust and understanding?
It increases understanding because it we can learn interactions, important variables, and the direction in which a input effects the predicted outcome
It increases trust if these are parsimonious with domain expertise or expectations
It increases trust if similar data points proceed through the model in the same way
It increases trust if interactions, important features, and direction of an input are consistent for similar data sets

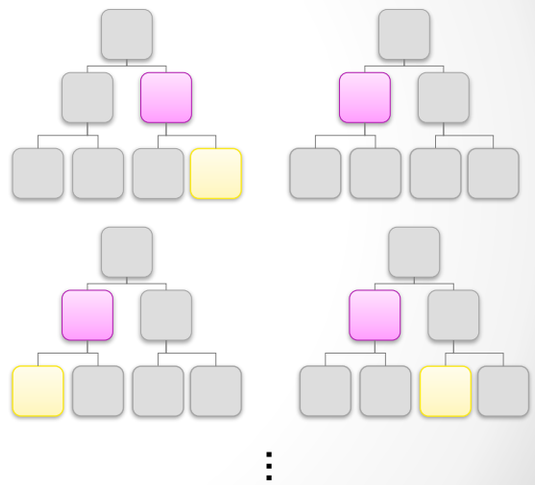
- Scale inputs to be non-negative
- Transform inputs such that their relationship with the target is monotonically increasing or decreasing
- Enables the human user to parse extra information from machine learning models:
    - In a neural network with only positive weights
    - For a binary classification task where the target value 1 indicates an event and the target value 0 indicates a non-event
    - All predictor variables are non-negative and monotonically increasing with respect to the target
    - Higher values of that predictor lead to increased occurrences of the target event
    - By following the maximum activation of neurons through the network it may even be possible to determine high-order interactions

- Binning does reduce the resolution of the information presented to the model during training
- But it can lead to better generalization (intricate patterns in the training data can be noise)
- Allows for elegant handling of outliers

Variable importance measures



Variable importance in tree-based models is often based on the amount splitting on a given variable increases predictive accuracy or node purity.



H2O.ai

How does it increase trust and understanding?
It increases understanding because we can learn important variables and their relative rank
It increases trust if these rankings match domain expertise or expectations
It increases trust if these ranks are repeatable in similar data

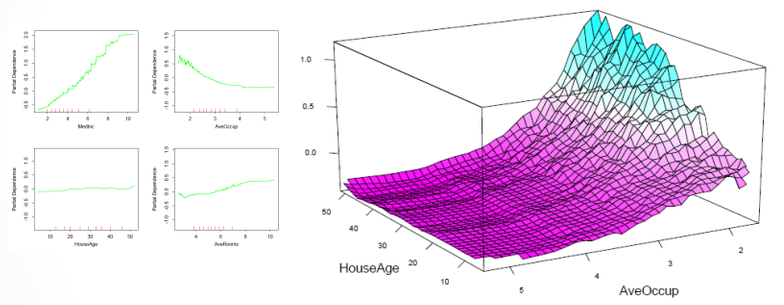
In Tree:
Split criterion change caused by an input for each node

In RF:
Split criterion change caused by an input for each node
Difference in OOB predictive accuracy when the predictor of interest is shuffled
(shuffling is seen as ‘zeroing out’ the effect of the variable in the trained model, because other variables are not shuffled)

In GBM:
Split criterion change caused by an input for each node

Simplistic variable importance measures can be biased toward larger scale variables or variables with a large number of categories

Partial dependence plots



HomeValue ~ MedInc + AveOccup + HouseAge + AveRooms

H2O.ai

Source: https://web.stanford.edu/~hastie/local.ftp/Springer/OLD/ESLII_print4.pdf

How does it increase trust and understanding?
It increases understanding because we can see the behavior of individual inputs and their 2 way interactions
It increases trust if the displayed behavior is consistent with domain expertise and expectations
It increases trust if displayed behavior is repeatable

Images: Elements of Statistical Learning, https://web.stanford.edu/~hastie/local.ftp/Springer/OLD/ESLII_print4.pdf, pg. 374

“Partial dependence tells us how the value of a variable influences the model predictions after we have averaged out the influence of all *other* variables. (For linear regression models, the resulting plots are simply straight lines whose slopes are equal to the model parameters.)”
- https://cran.r-project.org/web/packages/datarobot/vignettes/PartialDependence.html

Can be calculated efficiently for tree-based models, because of tree structure

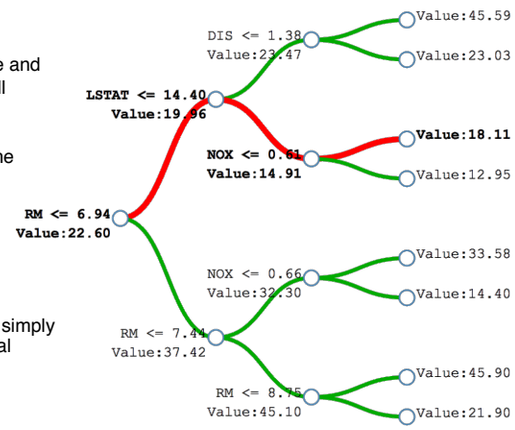
Interesting variant: https://github.com/numeristical/introspective

TreeInterpreter

Tree interpreter decomposes decision tree and random forest predictions into bias (overall average) and component terms.

This slide portrays the decomposition of the decision path into bias and individual contributions for a simple decision tree.

For a random forest model, treeinterpreter simply prints a ranked list of the bias and individual contributions for a given prediction.



Prediction: 18.11 ≈ 22.60 (trainset mean) - 2.64(loss from RM) - 5.04(loss from LSTAT) + 3.20(gain from NOX)

Source: http://blog.datadive.net/interpreting-random-forests/

H2O.ai

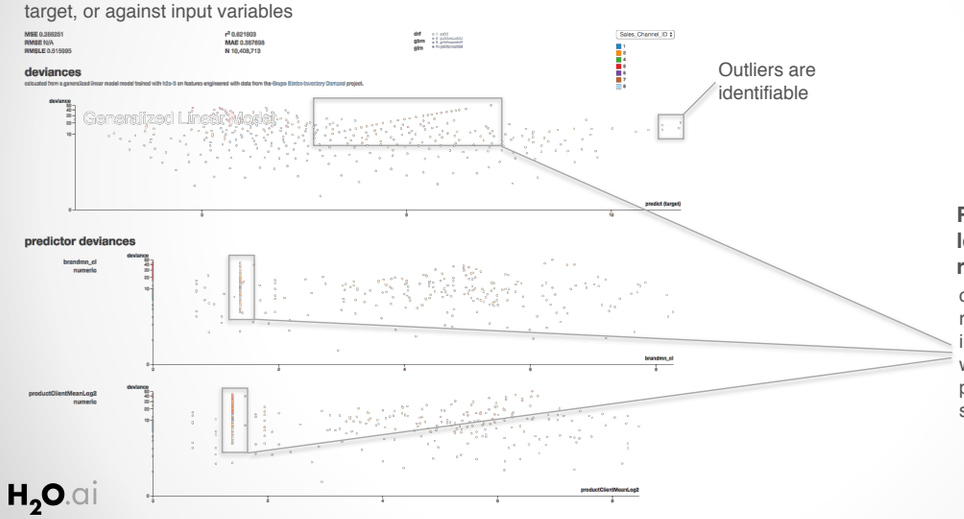
Currently only for sklearn decision tree and forest models.

How does it increase understanding? It allows for easy explanations of the internal mechanics of model
How does it increase trust? It increases trust if …
- internal mechanics represent known or expected phenomenon in the training data
- different decision paths lead to different results
- similar decision paths lead to similar results
- if model remains stable over time or over minor perturbations of training data

https://github.com/andosa/treeinterpreter

Residual analysis

Residuals can be plotted against the target, the predicted target, or against input variables



Outliers are identifiable

Residuals from a machine learning model should be randomly distributed

obvious patterns in residuals can indicate problems with data preparation or model specification

How does it increase understanding? Patterns in residuals can help elucidate patterns in the data that would otherwise be obscured by the curse of dimensionality, i.e. outliers, clusters, hierarchies, sparsity, etc.
How does it increase trust? If overall residuals are randomly distributed, this is a good indication that the ML model is fitting the data well. Obvious patterns in residuals could point to problems in model specification or data preparation that can be iteratively corrected by preprocessing data, building a model, and analyzing residuals

http://residuals.h2o.ai:8080/

Questions?