Predicting with both continuous and categorical features

Asked 7 years, 10 months ago Active 10 months ago Viewed 56k times

▲
26
▼

Some predictive modeling techniques are more designed for handling continuous predictors, while others are better for handling categorical or discrete variables. Of course there exist techniques to transform one type to another (discretization, dummy variables, etc.). However, are there any predictive modeling techniques that designed to handle both types of input at

The closest thing that I know of would be that usually decision trees handle discrete data well and they handle continuous data without requiring an *up front* discretization. However, this isn't quite what I was looking for since effectively the splits on continuous features are just a form of dynamic discretization.

For reference, here are some related, non-duplicate questions:

- How should decision tree splits be implemented when predicting continuous variables?
- Can I use multiple regression when I have mixed categorical and continuous predictors?
- Does it ever make sense to treat categorical data as continuous?
- Continuous and Categorical variable data analysis

classification predictive-models categorical-data continuous-data discrete-data

edited Apr 13 '17 at 12:44

Community ♦
1

asked Apr 19 '12 at 14:56

Michael McGowan
4,203 ● 3 ● 24 ● 46

1 Can you say more about what you want to do? Certainly, you can use multiple regression with both continuous & categorical covariates to build a predictive model. This is rather elementary. Do you mean predicting multiple *response* variables instead (where some are cont & some cat, eg)? – gung - Reinstate Monica ♦ Apr 19 '12 at 15:16

@gung How do you do multiple regression involving categorical covariates *without* converting converting the categorical predictors into numbers in some sense? – Michael McGowan Apr 19 '12 at 15:33

1 The 'numbers' don't mean anything--they aren't really *numerical*. In terms of how, you use a coding scheme; reference cell coding (typically called 'dummy coding') is the most common, but there are many schemes. This is a good resource for learning about the variety of schemes possible. In R, eg, you don't actually have to do that, though, if you include a vector or character data (ie, names), it will handle everything for you. – gung - Reinstate Monica ♦ Apr 19 '12 at 19:06

2 I'm afraid I still don't follow the impetus behind the question (I'm a little slow). The categorical variables are not "transformed" or "converted" into numerical variables; they are *represented* by a 1, but that 1 isn't really numerical. Qualitative predictors aren't any more numerical in multiple regression than they are in decision trees (ie, CART), eg. From a theoretical perspective, there's just nothing there; from a practical perspective, if you're trying to save a few lines of code, eg, software (R, eg) will often do that for you. – gung - Reinstate Monica ♦ Apr 19 '12 at 19:43

1 Fair enough, can you explain what you think is there (from a theoretical perspective)? How does dummy coding turn a categorical variable into one that's *meaningfully* numerical? How is it that what CART does allows the variable to retain its 'categorical-ness', but that dummy coding does not? I'm at a loss to figure out what the answers to these questions could be from your question above, & I don't see your previous post. – gung - Reinstate Monica ♦ Apr 19 '12 at 20:03

4 Answers

7

As far as I know, and I've researched this issue deeply in the past, there are no predictive modeling techniques (beside trees, XgBoost, etc.) that are designed to handle both types of input at the same time without simply transforming the type of the features.

Note that algorithms like Random Forest and XGBoost accept an input of mixed features, but they apply some logic to handle them during split of a node. Make sure you understand the logic "under the hood" and that you're OK with whatever is happening in the black-box.

Yet, **distance/kernel based models** (e.g., K-NN, NN regression, support vector machines) can be used to handle mixed type feature space by defining a “special” distance function. Such that, for every feature, applies an appropriate distance metric (e.g., for a numeric feature we’ll calculate the Euclidean distance of 2 numbers while for a categorical feature we’ll simple

Euclidean distance of 2 numbers while for a categorical feature we'll simple calculate the overlap distance of 2 string values). So, the distance/similarity between user $u_1 u_1$ and $u_2 u_2$ in feature $f_i f_i$, as follows: [Math Processing Error] if feature $f_i f_i$ is categorical, [Math Processing Error] if feature $f_i f_i$ is numerical. and 1 if feature $f_i f_i$ is not defined in $u_1 u_1$ or $u_2 u_2$.

Some known distance function for categorical features:

- Levenshtien distance (or any form of "edit distance")
- Longest common subsequence metric
- Gower distance
- And more metrics here

edited Apr 22 '19 at 12:22

answered Dec 21 '15 at 16:09

Serendipity

402 ● 4 ● 13

5

I know it's been a while since this question was posted, but if you're still looking at this problem (or similar ones) you may want to consider using generalized additive models (GAM's). I'm no expert, but these models allow you to combine different models to create a single prediction. The process used to find coefficients for the models you put in solves for all of them at once, so you can send a generalized additive model your favorite model for categorical predictors and your favorite model for continuous predictors and get a single model that minimizes RSS or whatever other error criterion you want to use.

Off the top of my head, the only software package which I know has an implementation of GAM's is the language R, but I'm sure there are others.

edited Mar 28 '15 at 21:36

gung - Reinstate Monica ♦

117k ● 39 ● 306 ● 591

answered Mar 28 '15 at 21:14

Forrest Cinelli

51 ● 1 ● 1

SAS has procedure called Proc Gam. – Alph Mar 28 '15 at 21:24

1 Most major statistical packages (eg, Stata) can probably implement GAMs. More to the point however, GAMs will use dummy codes to represent categorical variables as predictors. It isn't clear what the OP wants in looking for a model that uses categorical predictors as categorical but w/o representing them by dummy codes, but this isn't likely to be it. – gung - Reinstate Monica ♦ Mar 28 '15 at 21:34

Welcome to CV. Note that your username, identicon, & a link to your user page are automatically added to every post you make, so there is no need to sign your posts. In fact, we prefer you don't. – gung - Reinstate Monica ♦ Mar 28 '15 at 21:37

4

While discretization transforms continous data to discrete data it can hardly be said that dummy variables transform categorical data to continous data. Indeed, since algorithms can be run on computers there can hardly be a classificator algorithm which does NOT transform categorical data into dummy variables.

In the same sense a classificator ultimately transforms it predictors into a discrete variable indicating class belonging (even if it outputs a class probability, you ultimately choose a cutoff). De facto many classificators like logistic regression, random forest, decision trees and SVM all work fine with both types of data.

I suspect it would be hard to find an algorithm which works with continous data but cannot handle categorical data at all. Usually I tend to find it makes more difference on what type of data you have on the left side of your model.

answered Apr 19 '12 at 15:14

Erik

6,381 ● 15 ● 44

2 No, my point is that logistic regression et al do not "work" in the sense I'm describing with both types of data. They require you to, at least in some sense, treat all predictors as numbers or none of them as numbers. I know, for

instance, that one can often get great results with a logistic regression by coding something like "gender" as 1 for "male" and 0 for "female." However, I'm wondering whether this type of process can be avoided with any known modeling paradigm. – Michael McGowan Apr 19 '12 at 15:26

1

This is a deep philosophical question which is commonly addressed from the statistical as well as machine learning end. Some say, categorizing is better for discrete to categorical indicator, so that the packages can easily digest the model inputs. Others say, that binning can cause information loss, but however categorical variables can/must be converted to {1,0} indicator variables leaving out the last class for the model residuals.

The book - Applied linear regression (Kutner et al.) mentions about the logic of introducing indicator variables in the model in the first few chapters. There may be other similar text too.

My take on this maybe a bit too far-fetched: If we imagine the categorical variables like blocks in an experimental design, the indicator variable is a natural extension to non-experiment based data analysis. With respect to data mining algorithms (decision tree families), categorization is inevitable (either manually or automated-binning) which has to be fed to the model.

Hence, there may not be a model that is specialized for numerical as well as categorical variables in the same way (without binning-numerical or using indicators-categorical).

answered Sep 7 '15 at 0:46

KarthikS

746 🟨 1 ⬜ 6 🟧 15