In supervised learning, why is it bad to have correlated features?

Asked 3 years, 5 months ago Active 1 year, 3 months ago Viewed 63k times

63

I read somewhere that if we have features that are too correlated, we have to remove one, as this may worsen the model. It is clear that correlated features means that they bring the same information, so it is logical to remove one of them. But I can not understand why this can worsen the model.

42

machine-learning correlation

Share Improve this question Follow

edited Nov 8 '17 at 23:30
Valentin Calomme
4,757 3 16 44

asked Nov 7 '17 at 14:37
Spider
959 1 9 12

4 This rule applies more strongly in some models and analysis than others. Any chance you could add some context to "I read somewhere" - e.g. was it relation to training a specific model? – Neil Slater Nov 7 '17 at 14:46

4 Correlated features will not necessarily worsen a model. Removing correlated features helps to infer meaning about the features. – Hobbes Nov 7 '17 at 14:48

7 Answers

Active | Oldest | Votes

52

Correlated features in general don't improve models (although it depends on the specifics of the problem like the number of variables and the degree of correlation), but they affect specific models in different ways and to varying extents:

1. For linear models (e.g., linear regression or logistic regression), multicolinearity can yield solutions that are wildly varying and possibly numerically unstable.
2. Random forests can be good at detecting interactions between different features, but highly correlated features can mask these interactions.

More generally, this can be viewed as a special case of Occam's razor. A simpler model is preferable, and, in some sense, a model with fewer features is simpler. The concept of minimum description length makes this more precise.

Share Improve this answer Follow

answered Nov 7 '17 at 15:21
Ami Tavory

11 Numerical stability aside, prediction given by OLS model should not be affected by multicolinearity, as overall effect of predictor variables is not hurt by presence of multicolinearity. It is interpretation of effect of individual predictor variables that are not reliable when multicolinearity is present. – Akavall Nov 7 '17 at 22:23

Adding to the point on Random Forests: if you are using say, shap values for feature importance, having highly features can give unexpected results (shap values are additive, so the total contribution may be split between the correlated features, or allocated disproportionately to one of them). Similarly, if you are determining feature importance from number of times a feature is used to split trees and you have highly correlated features, you can get misleading results. – Jonathan Rayner Oct 30 '20 at 15:01

@Akavall Prediction given by OLS model is affected by multicolinearity, due to the fact that the parameter estimates are imprecise. This results in imprecise prediction on unseen data set, i.e., overfitting. The Wiki page has also discussed about this: en.wikipedia.org/wiki/… – Sean Yun-Shiuan Chuang Feb 1 at 9:21

24

(Assuming you are talking about supervised learning)

Correlated features will not always worsen your model, but they will not always improve it either.

There are three main reasons why you would remove correlated features:

- Make the learning algorithm faster

Due to the curse of dimensionality, less features usually mean high improvement in terms of speed.

If speed is not an issue, perhaps don't remove these features right away (see next point)

- Decrease harmful bias

The keyword being harmful. If you have correlated features but they are also correlated to the target, you want to keep them. You can view features as hints to make a good guess, if you have two hints that are essentially the same, but they are good hints, it may be wise to keep them.

Some algorithms like Naive Bayes actually directly benefit from "positive" correlated features. And others like random forest may indirectly benefit from them.

Imagine having 3 features A, B, and C. A and B are highly correlated to the target and to each other, and C isn't at all. If you sample out of the 3 features, you have 2/3 chance to get a "good" feature, whereas if you remove B for instance, this chance drops to 1/2

Of course, if the features that are correlated are not super informative in the first place, the algorithm may not suffer much.

So moral of the story, removing these features might be necessary due to speed, but remember that you might make your algorithm worse in the process. Also, some algorithms like decision trees have feature selection embedded in them.

A good way to deal with this is to use a wrapper method for feature selection. It will remove

A good way to deal with this is to use a wrapper method for feature selection. It will remove redundant features only if they do not contribute directly to the performance. If they are useful like

in naive bayes, they will be kept. (Though remember that wrapper methods are expensive and may lead to overfitting)

- Interpretability of your model

If your model needs to be interpretable, you might be forced to make it simpler. Make sure to also remember Occam's razor. If your model is not "that much" worse with less features, then you should probably use less features.

Share Improve this answer Follow

answered Nov 8 '17 at 18:56

Valentin Calomme
4,757 3 16 44

11

Why is Multicollinearity a Potential Problem?

A key goal of regression analysis is to isolate the relationship between each independent variable and the dependent variable. The interpretation of a regression coefficient is that it represents the mean change in the dependent variable for each 1 unit change in an independent variable when you hold all of the other independent variables constant. That last portion is crucial for our further discussion about multicollinearity.

The idea is that you can change the value of one independent variable and not the others. However, when independent variables are correlated, it indicates that changes in one variable are associated with shifts in another variable. The stronger the correlation, the more difficult it is to change one variable without changing another. It becomes difficult for the model to estimate the relationship between each independent variable and the dependent variable independently because the independent variables tend to change in unison.

What Problems Do Multicollinearity Cause?

Multicollinearity causes the following two basic types of problems:

- The coefficient estimates can swing wildly based on which other independent variables are in the model. The coefficients become very sensitive to small changes in the model.
- Multicollinearity reduces the precision of the estimate coefficients, which weakens the statistical power of your regression model. You might not be able to trust the p-values to identify independent variables that are statistically significant.

Imagine you fit a regression model and the coefficient values, and even the signs, change dramatically depending on the specific variables that you include in the model. It’s a disconcerting feeling when slightly different models lead to very different conclusions. You don’t feel like you

know the actual effect of each variable!

Now, throw in the fact that you can't necessarily trust the p-values to select the independent variables to include in the model. This problem makes it difficult both to specify the correct model and to justify the model if many of your p-values are not statistically significant.

As the severity of the multicollinearity increases so do these problematic effects. **However, these issues affect only those independent variables that are correlated. You can have a model with severe multicollinearity and yet some variables in the model can be completely unaffected.**

Do I Have to Fix Multicollinearity?

Multicollinearity makes it hard to interpret your coefficients, and it reduces the power of your model to identify independent variables that are statistically significant. These are definitely serious problems. However, the good news is that you don't always have to find a way to fix multicollinearity!

The need to reduce multicollinearity depends on its severity and your primary goal for your regression model. Keep the following three points in mind:

1. The severity of the problems increases with the degree of the multicollinearity. Therefore, if you have only moderate multicollinearity, you may not need to resolve it.
2. Multicollinearity affects only the specific independent variables that are correlated. Therefore, if multicollinearity is not present for the independent variables that you are particularly interested in, you may not need to resolve it. Suppose your model contains the experimental variables of interest and some control variables. If high multicollinearity exists for the control variables but not the experimental variables, then you can interpret the experimental variables without problems.
3. **Multicollinearity affects the coefficients and p-values, but it does not influence the predictions, precision of the predictions, and the goodness-of-fit statistics**. If your primary goal is to make predictions, and you don't need to understand the role of each independent variable, you don't need to reduce severe multicollinearity. (Reference: "The fact that some or all predictor variables are correlated among themselves does not, in general, inhibit our ability to obtain a good fit nor does it tend to affect inferences about mean responses or predictions of new observations." — Applied Linear Statistical Models, p289, 4th Edition.)

Source: Multicollinearity in Regression Analysis: Problems, Detection, and Solutions by Jim Frost

edited Jan 8 '20 at 9:19

answered Jan 5 '20 at 20:57

Pratik Nabriya
111 1 6

7

In perspective of storing data in databases, storing correlated features is somehow similar to storing redundant information which it may cause wasting of storage and also it may cause inconsistent data after updating or editing tuples.

If we add so much correlated features to the model we may cause the model to consider unnecessary features and we may have curse of high dimensionality problem, *I guess this is the reason for worsening the constructed model.*

In the context of machine learning we usually use `PCA` to reduce the dimension of input patterns. This approach considers removing correlated features by someway (using `SVD`) and is an unsupervised approach. This is done to achieve the following purposes:

- Compression
- Speeding up learning algorithms
- Visualizing data
- Dealing with curse of high dimensionality

Although this may not seem okay but I have seen people that use removing correlated features in order to avoid overfitting but I don't think it is a good practice. For more information I highly recommend you to see here.

Another reason is that in deep learning models, like `MLPs` if you add correlated features, you just add unnecessary information which adds more calculations and parameters to the model.

Share Improve this answer Follow

edited Aug 1 '18 at 17:08

answered Nov 7 '17 at 16:01

Media
12.3k 9 44 85

6

Sometimes correlated features -- and the duplication of information that provides -- does not hurt a predictive system. Consider an ensemble of decision trees, each of which considers a sample of rows and a sample of columns. If two columns are highly correlated, there's a chance that one of them won't be selected in a particular tree's column sample, and that tree will depend on the remaining column. Correlated features mean you can reduce overfitting (through column sampling) without giving up too much predictive quality.

Share Improve this answer Follow

answered Nov 7 '17 at 20:14

Dan Jarratt
305 1 5

3

Making a decision should be done on the minimum necessary variables to do so. This is, as mentioned above, the formalization of Occam's razor with minimum description length above. I like that one.

I would tend to characterize this phenomena in something like a HDDT to mean the most efficient tree that makes no spurious decision based on available data, and avoiding all instances of

decisions that may otherwise have been made on multiple data points without understanding that they were correlated.

Share Improve this answer Follow

answered Aug 1 '18 at 16:28

tjborromeo
31 2

Regarding datascience.stackexchange.com/users/38887/valentin-calomme comment: "Correlated features will not always worsen your model, but they will not always improve it either." I don't see or can't think of where having high correlation between variables doesn't make your model worse. At least in the sense that, given the choice: I'd rather train a network with less correlated features. Anything other than that is functionally and provably worse. Are there instances when this isn't true? – tjborromeo Aug 7 '18 at 10:07

2

The answer to this question depends greatly upon the purpose of the model. In inference, highly correlated features are a well-known problem. For example, two features highly correlated with each other and with y, might both come out as insignificant in an inference model, potentially missing an important explanatory signal. Therefore, in inference it is generally recommended to thin them out.

If your supervised learning is for prediction, the answer - counter to conventional wisdom - is usually the opposite. The only reason to remove highly correlated features is storage and speed concerns. Other than that, what matters about features is whether they contribute to prediction, and whether their data quality is sufficient.

Noise-dominated features will tend to be less correlated with other features, than features correlated with y. Hence, as mentioned above in the example by Valentin, thinning out the latter will increase the proportion of the former.

In particular, methods like random forests and KNN treat all features equally, so thinning out correlated features directly reduces their signal-to-noise ratio.

Methods that auto-select features like single trees, "pure" lasso, or neural networks, might be less affected. But even then, other than longer computing time, there is rarely anything to lose prediction-wise from keeping correlated features in the mix.

Share Improve this answer Follow

answered May 12 '19 at 17:34

Assaf
121 1