Interpreting R²: a Narrative Guide for the Perplexed

An accessible walkthrough of fundamental properties of this popular, yet often misunderstood metric from a predictive modeling perspective

Roberta Rocca · Follow
Published in Towards Data Science · 15 min read · 2 days ago

117 5

Photo by Josh Rakower on Unsplash

R² (R-squared), also known as the *coefficient of determination*, is widely used as a metric to evaluate the performance of regression models. It is commonly used to quantify *goodness of fit* in statistical modeling, and it is a default scoring metric for regression models both in popular statistical modeling and machine learning frameworks, from *statsmodels* to *scikit-learn*.

Despite its omnipresence, there is a surprising amount of confusion on what R² truly means, and it is not uncommon to encounter conflicting information (for example, concerning the upper or lower bounds of this metric, and its interpretation). At the root of this confusion is a "culture clash" between the explanatory and predictive modeling tradition. In fact, in predictive modeling — where evaluation is conducted out-of-sample and any modeling approach that increases performance is desirable — many properties of R² that do apply in the narrow context of explanation-oriented linear modeling no longer hold.

To help navigate this confusing landscape, this post provides an accessible narrative primer to some basic properties of R² from a predictive modeling perspective, highlighting and dispelling common confusions and misconceptions about this metric. With this, I hope to help the reader to converge on a unified intuition of what R² truly captures as a measure of fit in predictive modeling and machine learning, and to highlight some of this metric's strengths and limitations. Aiming for a broad audience which includes Stats 101 students and predictive modellers alike, I will keep the language simple and ground my arguments into concrete visualizations.

Ready? Let's get started!

What is R²?

Let's start from a working verbal definition of R². To keep things simple, let's take the first high-level definition given by Wikipedia, which is a good reflection of definitions found in many pedagogical resources on statistics, including authoritative textbooks:

> *the proportion of the variation in the dependent variable that is predictable from the independent variable(s)*

Anecdotally, this is also what the vast majority of students trained in using statistics for inferential purposes would probably say, if you asked them to define R². But, as we will see in a moment, this common way of defining R² is the source of many of the misconceptions and confusions related to R². Let’s dive deeper into it.

Calling R² a *proportion* implies that R² will be a number between 0 and 1, where 1 corresponds to a model that explains *all the variation* in the outcome variable, and 0 corresponds to a model that explains *no variation* in the outcome variable. Note: your model might also include no predictors (e.g., an intercept-only model is still a model), that’s why I am focusing on variation predicted by a model rather than by independent variables.

Let’s verify if this intuition on the range of possible values is correct. To do so, let’s recall the mathematical definition of R²:

$$R^2 = 1 - \frac{RSS}{TSS}$$

Here, RSS is the residual sum of squares, which is defined as:

$$RSS = \sum_{i=1}^{n} (y_i - \hat{y}_i)^2$$

This is simply the **sum of squared errors of the model**, that is the sum of squared differences between true values *y* and corresponding model predictions *ŷ*.

On the other hand, TSS, the total sum of squares, is defined as follows:

$$TSS = \sum_{i=1}^{n} (y_i - \bar{y})^2$$

As you might notice, this term has a similar "form" than the residual sum of squares, but this time, we are looking at the squared differences between the true values of the outcome variables *y* and *the mean of the outcome variable* ȳ. This is technically the *variance* of the outcome variable. But a more intuitive way to look at this in a predictive modeling context is the following: this term is the residual sum of squares of a model that always predicts the mean of the outcome variable. Hence, **the ratio of RSS and TSS is a ratio between the sum of squared errors of your model, and the sum of squared errors of a "reference" model predicting the mean of the outcome variable**.

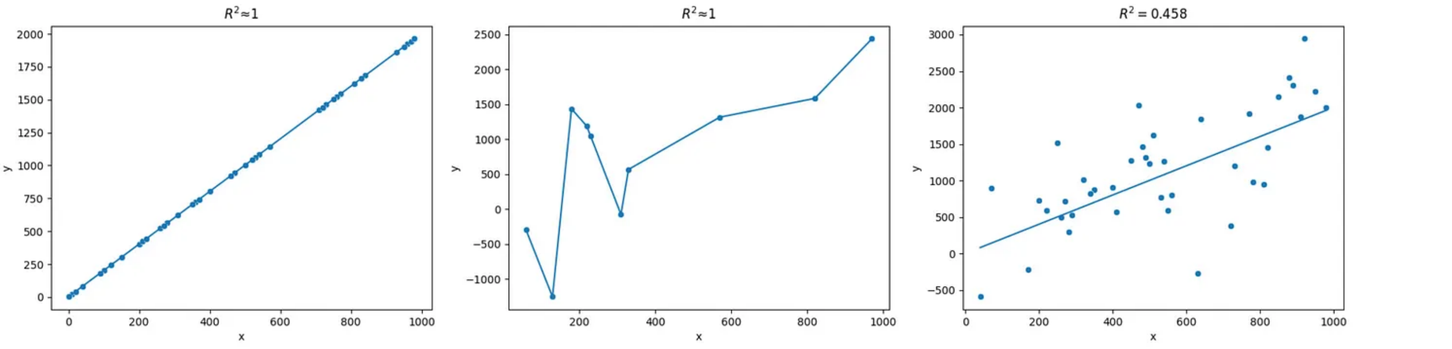
With this in mind, let's go on to analyse what the range of possible values for this metric is, and to verify our intuition that these should, indeed, range between 0 and 1.

What is the best possible R²?

As we have seen so far, R² is computed by subtracting the ratio of RSS and TSS from 1. Can this ever be higher than 1? Or, in other words, is it true that 1 is the largest possible value of R²? Let's think this through by looking back at the formula.

The only scenario in which 1 minus *something* can be higher than 1 is if that *something* is a *negative* number. But here, RSS and TSS are both sums of squared values, that is, sums of positive values. The ratio of RSS and TSS will thus *always* be positive. The largest possible R² must therefore be 1.

Now that we have established that R² cannot be higher than 1, let's try to visualize what needs to happen for our model to have the maximum possible R². For R² to be 1, RSS / TSS must be zero. This can happen if RSS = 0, that is, if the model predicts all data points *perfectly*.



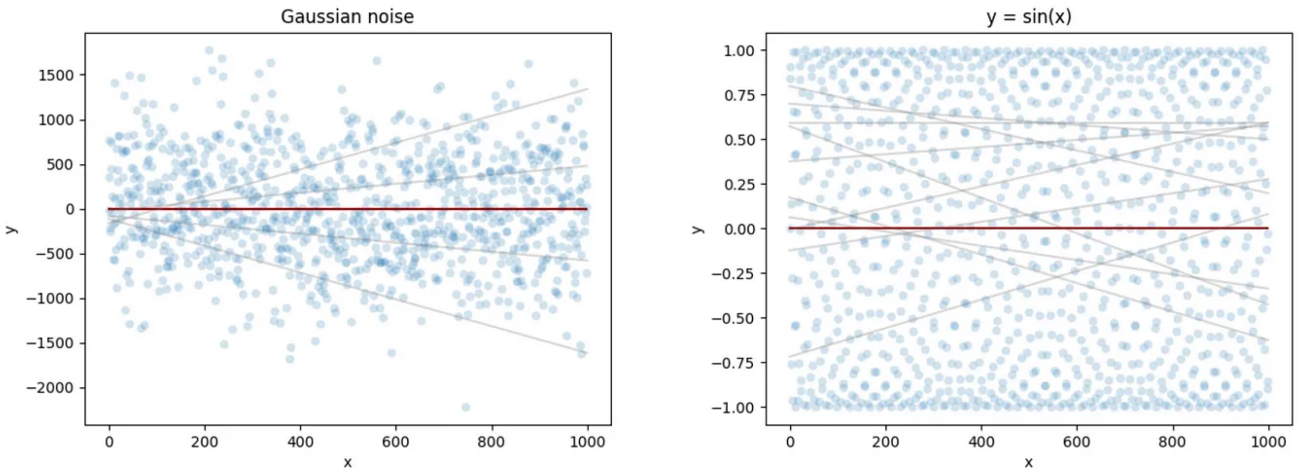
Examples illustrating hypothetical models with R² ≈ 1 using simulated data. In all cases, the true underlying model is y = 2x + 3. The first two models fit the data perfectly, in the first case because the data has no noise and a linear model can retrieve perfectly the relation between x and y (left) and in the second because the model is very flexible and overfits the data (center). These are extreme cases which are hardly found in reality. In fact, the largest possible R² will often be defined by the amount of noise if the data. This is illustrated by the third plot, where due to the presence of random noise, even the true model can only achieve R² = 0.458.

In practice, this will never happen, unless you are *wildly* overfitting your data with an overly complex model, or you are computing R² on a ridiculously low number of data points that your model can fit perfectly. All datasets will have *some* amount of noise that *cannot* be accounted for by the data. In practice, the largest possible R^2 will be defined by the amount of unexplainable noise in your outcome variable.

What is the worst possible R^2?

So far so good. If the largest possible value of R^2 is 1, we can still think of R^2 as the proportion of variation in the outcome variable explained by the model. But let's now move on to looking at the lowest possible value. If we buy into the definition of R^2 we presented above, then we must assume that the lowest possible R^2 is 0.

When is $R^2 = 0$? For R^2 to be null, RSS/TSS must be equal to 1. This is the case if RSS = TSS, that is, if the sum of squared errors of our model is equal to the sum of squared errors of a model predicting the mean. If you are better off just predicting the mean, then your model is really not doing a terribly good job. There are infinitely many reasons why this can happen, one of these being an issue with your choice of model — if, for example, if you are trying to model really non-linear data with a linear model. Or it can be a consequence of your data. If your outcome variable is very noisy, then a model predicting the mean might be the best you can do.



Two cases where the mean model might be the best possible (linear) models because: a) data is pure Gaussian noise (left); b) the data is highly non-linear, as it is generated using a periodic function (right).

But is $R^2 = 0$ truly the lowest possible R^2? Or, in other words, can R^2 ever be negative? Let's look back at the formula. $R^2 < 0$ is only possible if $RSS/TSS > 1$, that is, if $RSS > TSS$. Can this ever be the case?

This is where things start getting interesting, as the answer to this question depends very much on contextual information that we have not yet specified, namely which type of models we are considering, and which data we are computing R^2 on. As we will see, whether our interpretation of R^2 as the proportion of variance explained holds depends on our answer to these questions.

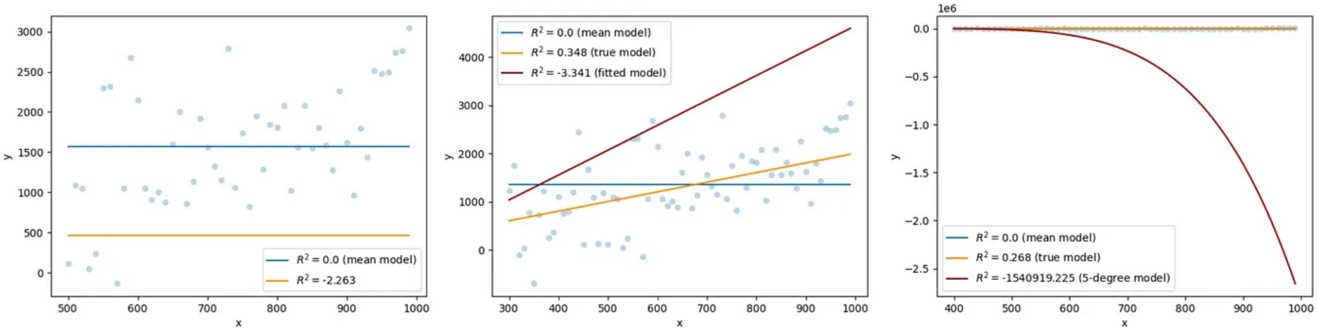
The bottomless pit of negative R^2

Let's looks at a concrete case. Let's generate some data using the following model $y = 3 + 2x$, and added Gaussian noise.

```
import numpy as np

x = np.arange(0, 1000, 10)
y = [3 + 2*i for i in x]
noise = np.random.normal(loc=0, scale=600, size=x.shape[0])
true_y = noise + y
```

The figure below displays three models that make predictions for *y* based on values of *x* for different, randomly sampled subsets of this data. These models are not made-up models, as we will see in a moment, but let's ignore this right now. Let's focus simply on the sign of their R^2.



Three examples of models for data generated using the function: y = 3 + 2x, with added Gaussian noise.

Let's start from the first model, a simple model that predicts a constant, which in this case is lower than the mean of the outcome variable. Here, our RSS will be the sum of squared distances between each of the dots and the orange line, while TSS will be the sum of squared distances between each of the dots and the blue line (the mean model). It is easy to see that for most of the data points, the distance between the dots and the orange line will be higher than the distance between the dots and the blue line. Hence, our RSS will be higher than our TSS. If this is the case, we will have RSS/TSS > 1, and, therefore: 1 — RSS/TSS < 0, that is, $R^2<0$.

In fact, if we compute R^2 for this model on this data, we obtain $R^2 = -2.263$. If you want to check that it is in fact realistic, you can run the code below (due to randomness, you will likely get a similarly negative value, but not exactly the same value):

```
from sklearn.metrics import r2_score

# get a subset of the data
x_tr, x_ts, y_tr, y_ts = train_test_split(x, true_y, train_size=.5)
# compute the mean of one of the subsets
model = np.mean(y_tr)
# evaluate on the subset of data that is plotted
print(r2_score(y_ts, [model]*y_ts.shape[0]))
```

Let’s now move on to the second model. Here, too, it is easy to see that distances between the data points and the red line (our target model) will be larger than distances between data points and the blue line (the mean model). In fact, here: R^2= -3.341. Note that our target model is different from the *true* model (the orange line) because we have fitted it on a subset of the data that also includes noise. We will return to this in the next paragraph.

Finally, let’s look at the last model. Here, we fit a 5-degree polynomial model to a subset of the data generated above. The distance between data points and the fitted function, here, is *dramatically* higher than the distance between the data points and the mean model. In fact, our fitted model yields R^2 = -1540919.225.

Clearly, as this example shows, models *can* have a negative R^2. In fact, there is no limit to how low R^2 can be. Make the model bad enough, and your R^2 can approach minus infinity. This can also happen with a simple linear model: further increase the value of the slope of the linear model in the second example, and your R^2 will keep going down. So, where does this leave us with respect to our initial question, namely whether R^2 is in fact that proportion of variance in the outcome variable that can be accounted for by the model?

Well, we don’t tend to think of proportions as arbitrarily large negative values. If are really attached to the original definition, we could, with a creative leap of imagination, extend this definition to covering scenarios where arbitrarily bad models can *add* variance to your outcome variable. The inverse proportion of variance *added* by your model (e.g., as a consequence of poor model choices, or overfitting to different data) is what is reflected in arbitrarily low negative values.

But this is more of a metaphor than a definition. Literary thinking aside, the most literal and most productive way of thinking about R^2 is as a comparative metric, which says something about how much better (on a scale from 0 to 1) or worse (on a scale from 0 to infinity) your model is at predicting the data *compared to a model which always predicts the mean of the outcome variable*.

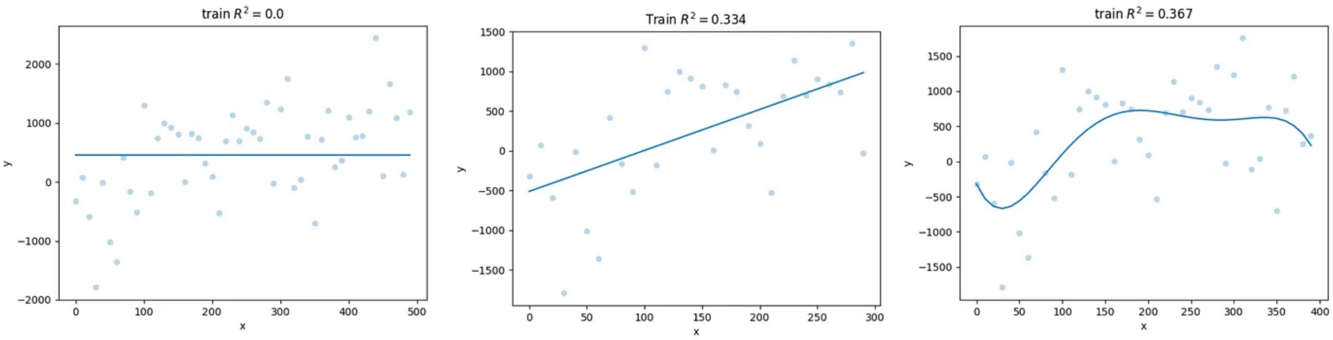
Importantly, what this suggests, is that while R² can be a tempting way to evaluate your model in a scale-independent fashion, and while it might makes sense to use it as a comparative metric, it is a far from transparent metric. The value of R² will not provide explicit information of how wrong your model is in absolute terms; the best possible value will always be dependent on the amount of noise present in the data; and good or bad R² can come about from a wide variety of reasons that can be hard to disambiguate without the aid of additional metrics.

Alright, R² can be negative. But does this ever happen, in practice?

A very legitimate objection, here, is whether any of the scenarios displayed above is actually plausible. I mean, which modeller in their right mind would actually fit such *poor* models to such simple data? These might just look like *ad hoc* models, made up for the purpose of this example and not actually fit to any data.

This is an excellent point, and one that brings us to another crucial point related to R² and its interpretation. As we highlighted above, all these models *have*, in fact, been fit to data which are generated from the same true underlying function as the data in the figures. This corresponds to the practice, foundational to predictive modeling, of splitting data intro a *training set* and a *test set*, where the former is used to estimate the model, and the latter for evaluation on unseen data — which is a "fairer" proxy for how well the model generally performs in its prediction task.

In fact, if we display the models introduced in the previous section against the data used to estimate them, we see that they are not *unreasonable* models in relation to their training data. In fact, R² values for the training set are, at least, non-negative (and, in the case of the linear model, very close to the R² of the true model on the test data).



Same functions displayed in the previous figure, this time displayed against the data they were fit on, which were generated with the same true function y = 3 + 2x. For the first model, which predicts a constant, model "fitting" simply consists of calculating the mean of the training set.

Why, then, is there such a big difference between the previous data and this data? What we are observing are cases of *overfitting*. The model is mistaking sample-specific noise in the training data for signal and modeling that — which is not at all an uncommon scenario. As a result, models' predictions on new data samples will be poor.

Avoiding overfitting is perhaps the biggest challenge in predictive modeling. Thus, it is not at all uncommon to observe negative R² values when (as one should always do to ensure that the model is generalizable and robust) R² is computed *out-of-sample*, that is, on data that differ "randomly" from those on which the model was estimated.

Thus, the answer to the question posed in the title of this section is, in fact, a resounding *yes*: negative R² do happen in common modeling scenarios, even when models have been properly estimated. In fact, they happen all the time.

So, is everyone just wrong?

If R² is *not* a proportion, and its interpretation as variance explained clashes with some basic facts about its behavior, do we have to conclude that our initial definition is wrong? Are Wikipedia and all those textbooks presenting a similar definition wrong? Was my Stats 101 teacher wrong? Well. Yes, and no. It depends hugely on the context in which R^2 is presented, and on the modeling tradition we are embracing.

If we simply analyse the definition of R^2 and try to describe its general behavior, *regardless* of which type of model we are using to make predictions, and assuming we will want to compute this metrics out-of-sample, then yes, they are all wrong. Interpreting R^2 as the proportion of variance explained is misleading, and it conflicts with basic facts on the behavior of this metric.

Yet, the answer changes slightly if we constrain ourselves to a narrower set of scenarios, namely *linear models*, and especially linear models *estimated with least squares methods*. Here, R^2 *will* behave as a proportion. In fact, it can be shown that, due to properties of least squares estimation, a linear model can *never* do worse than a model predicting the mean of the outcome variable. Which means, that a linear model can never have a negative R^2 — or at least, it cannot have a negative R^2 on the same data on which it was estimated (a debatable practice if you are interested in a generalizable model). For a *linear regression* scenario with in-sample evaluation, the definition discussed can therefore be considered correct. Additional fun fact: this is also the only scenario where R^2 is equivalent to the squared correlation between model predictions and the true outcomes.

The reason why many misconceptions about R^2 arise is that this metric is often first introduced in the context of linear regression and with a focus on *inference* rather than prediction. But in predictive modeling, where *in-sample* evaluation is a no-go and linear models are just one of many possible models, interpreting R^2 as the proportion of variation explained by the model is at best unproductive, and at worst deeply misleading.

Should I still use R^2?

We have touched upon quite a few points, so let's sum them up. We have observed that:

- R^2 cannot be interpreted as a proportion, as its values can range from $-\infty$ to 1
- Its interpretation as "variance explained" is also misleading (you can imagine models that *add* variance to your data, or that combined explained existing variance and variance "hallucinated" by a model)
- In general, R^2 is a "relative" metric, which compares the errors of your model with those of a simple model always predicting the mean
- It is, however, accurate to describe R^2 as the proportion of variance explained *in the context of linear modeling with least squares estimation* and *when the R^2 of a least-squares linear model is computed in-sample*.

Given all these caveats, should we still use R^2? Or should we give up?

Here, we enter the territory of more subjective observations. In general, if you are doing predictive modeling and you want to get a concrete sense for *how wrong* your predictions are in absolute terms, R^2 is *not* a useful metric. Metrics like MAE or RMSE will definitely do a better job in providing information on the magnitude of errors your model makes. This is useful in absolute terms but also in a model comparison context, where you might want to know by how much, concretely, the precision of your predictions differs across models. If knowing something about precision matters (it hardly ever does not), you might at least want to complement R^2 with metrics that says something meaningful about how wrong each of your individual predictions is likely to be.

More generally, as we have highlighted, there are a number of caveats to keep in mind if you decide to use R^2. Some of these concern the "practical" upper bounds for R² (your noise ceiling), and its literal interpretation as a *relative*, rather than absolute measure of fit compared to the mean model. Furthermore, good or bad R² values, as we have observed, can be driven by many factors, from overfitting to the amount of noise in your data.

On the other hand, while there are very few predictive modeling contexts where I have found R² particularly informative in isolation, having a measure of fit relative to a “dummy” model (the mean model) can be a productive way to think critically about your model. Unrealistically high R² on your training set, or a negative R² on your test set might, respectively, help you entertain the possibility that you might be going for an overly complex model or for an inappropriate modeling approach (e.g., a linear model for non-linear data), or that your outcome variable might contain, mostly, noise. This is, again, more of a “pragmatic” personal take here, but while I would resist fully discarding R² (there aren’t many good global and scale-independent measures of fit), in a predictive modeling context I would consider it most useful as a complement to scale-dependent metrics such as RMSE/MAE, or as a “diagnostic” tool, rather than a target itself.

Concluding remarks

R² is everywhere. Yet, especially in fields that are biased towards explanatory, rather than predictive modelling traditions, many misconceptions about its interpretation as a model evaluation tool flourish and persist.