Statistics and Data Analysis in MATLAB
Kendrick Kay, kendrick.kay@wustl.edu
March 21, 2014

Lecture 5: Model accuracy

1. The basics

- Suppose we have fit a model to a set of data. We would like to know how well the model describes the data, that is, *model accuracy*. Knowing model accuracy helps us decide whether the model is useful at all and also helps us choose among multiple competing models.
- The first thing to do before getting into any details is to look at your data and your model fit. This will give you a sense of the data and model and may reveal anomalies (or bugs) in your data or model that you should fix up front.
- To quantify model accuracy, we could in principle use the same metric of squared error (or, alternatively, absolute error) that we discussed in the context of model fitting (see Lecture 4). However, this metric is difficult to interpret because the magnitude and range of the metric depends on the units of the data and the number of data points.

2. Coefficient of determination (R^2)

- A useful metric for model accuracy is the *coefficient of determination* (R^2). Before describing R^2, we first need to understand *variance*. Technically, variance is the square of the standard deviation:

$$\text{variance} = \frac{\sum_{i=1}^{n}(x_i - \bar{x})^2}{n-1}$$

where n is the number of data points, x_i is the ith data point, and $\bar{x}$ is the mean of the data points. For intuition, we can ignore the denominator and think of variance as the total squared deviation of a set of data points from their mean.
- R^2 is the percentage of variance explained by a model:

$$R^2 = 100 \times \left(1 - \frac{\text{unexplained variance}}{\text{total variance}}\right) \Rightarrow R^2 = 100 \times \left(1 - \frac{\sum_{i=1}^{n}\left(d_i - m_i\right)^2}{\sum_{i=1}^{n}\left(d_i - \bar{d}\right)^2}\right)$$

where n is the number of data points, d_i is the ith data point, m_i is the model fit for the ith data point, and $\bar{d}$ is the mean of the data points. There are two main components of the formula. The first component is in the numerator and is the sum of the squares of the residuals (which is just the usual concept of squared error). The second component is in the denominator and is the sum of the squares of the deviations of the data points from their mean. The way the formula works is to quantify the variance that *is not* explained by the model (the numerator), express this as a fraction of the total variance (the denominator), subtract the result from one so that we get the variance that *is* explained, and then multiply by 100 to obtain a percentage. (The denominator of the variance formula, $n - 1$, cancels out in the computation of the ratio.)
- The R^2 metric has an upper bound of 100% (corresponding to the case where a model matches the data exactly) and does not have a lower bound (since a model can be arbitrarily bad). An R^2 of 0% is achieved by a model that gets the mean of the data correct and nothing else.

- Note that R^2 is not the same as r^2. This is because correlation (r) implicitly fits a linear model with free parameters (an offset parameter and a gain parameter), whereas coefficient of determination (R^2) does not include these parameters. It is true that after applying an offset and gain to match a set of model predictions to a set of data, then r^2 will equal R^2. However, the correct approach is to include estimation of offset and gain as part of the model-building process and then use the R^2 metric.
- Given that R^2 involves squaring, it in some sense assumes Gaussian noise (see Lecture 4). One could use alternative metrics (such as a version of R^2 in which absolute deviations are summed), but we will stick with R^2 for the purposes of this lecture. In the end, it is important to eyeball your data and your model fit to make sure your metric is doing something reasonable.

3. Accuracy on the sample vs. accuracy on the population
- Now that we understand metric of R^2, we could calculate it for the data and the model fit that we have. This approach is fine if characterizing the set of data that we have collected is the goal of the modeling effort.
- However, the observed set of data is just one sample from the population, that is, the distribution that underlies the data-collection process. We are probably interested not necessarily in how well the fitted model describes the sample, but in how well the fitted model describes the population. The problem is that the accuracy of a model evaluated on data used to fit the model will, on average, overestimate the true accuracy of the model.
- For example, suppose we knew the true model t (which can be thought of as the underlying function that generates the observed data). Imagine that we measure the accuracy of model t by obtaining a dataset and calculating how well the model describes the data. The resulting accuracy level R^2_{true} reflects the true accuracy of model t. Now suppose we allowed the parameters of the model to vary in order to fit the dataset; this produces fitted model f. If we were to calculate how well the fitted model describes the dataset, the resulting accuracy level R^2_{fitted} would be larger than the original accuracy level: $R^2_{fitted} > R^2_{true}$. Furthermore, since the parameters of fitted model f differ from the parameters of true model t, the fitted model will not perform as well as the true model in describing new data. Thus, if we were obtain a new dataset and calculate how well the fitted model describes the data, the resulting accuracy level $R^2_{generalize}$ would be less than the accuracy level of the true model, R^2_{true}. We can summarize the various relationships as follows: $R^2_{fitted} > R^2_{true} > R^2_{generalize}$.
- In words, what this means is that after fitting a model to a set of data, the R^2 value of the fitted model on the observed sample will be larger than the R^2 value of the true model on the population, which in turn will be larger than the R^2 value of the fitted model on the population. In practice, we do not know the true model, so the bottom line is that the R^2 on the observed sample will, on average, be an overestimate of the R^2 on the population.

4. Cross-validation
- To quantify the accuracy of a fitted model, we can use the technique of *cross-validation*. The idea is simple: First, use a set of *training data* to fit the parameters of a model. Then, use an independent set of *testing data* to evaluate the accuracy of the fitted model.
- There are various flavors of cross-validation. In *leave-one-out cross-validation*, a single data point is omitted from the fitting process and the fitted model is used to predict that data point. The process is repeated for each of the remaining data points. Finally, the model predictions of the data points are aggregated and then compared to the data using some metric (such as R^2). In

k-fold cross-validation, the dataset is randomly divided into *k* parts, and then the process proceeds as usual (omit one part from the fitting process, use the fitted model to predict that part, omit the next part, etc.). Finally, there can be simple forms of cross-validation, such as collecting two sets of data and using one for training and the other for testing.
- Which cross-validation scheme to use depends on balancing computational time against model performance: (1) the larger the number of cross-validation iterations, the more computational time will be needed, (2) the larger the amount of data used to fit the model, the better the estimates of the model parameters and the higher the model accuracy, and (3) the larger the amount of data used in the accuracy calculation, the more reliable the accuracy estimate. Leave-one-out cross-validation is at one extreme of the spectrum: it involves many cross-validation iterations (one for each data point); it uses the maximum amount of data for model fitting; and it uses all of the data points to calculate model accuracy.
- When performing cross-validation with more than one iteration, a tricky issue is that a different fitted model is obtained on each iteration. The accuracy level that is obtained can be interpreted as the expected accuracy of the model when fitted with a particular amount of data. For example, suppose we perform 5-fold cross-validation. The R^2 between the model predictions and the data can be interpreted as an estimate of the accuracy of the model when fitted on a dataset whose size is equal to 80% of the number of data points in the observed dataset.
- When dividing a data set into different parts for training and testing, it is important to ensure that the division is as strict as possible. It is all too easy to slip and allow some dependencies between the training and testing data, which will invalidate the idea that the performance on the testing data is an unbiased estimate of model accuracy.
- There are other techniques for quantifying model accuracy. One common method is using an *F*-test to assess whether a more complex model produces a significantly better fit than a simpler model. However, this approach is applicable only for nested models (where one model is a simpler version of a more complex model). Other methods include AIC and BIC which, like cross-validation, also attempt to assess the out-of-sample error of a model. Cross-validation seems preferable, as it makes fewer assumptions than these approaches and is conceptually simpler. However, a drawback of cross-validation is that it is computationally intensive.

5. Model selection

- With cross-validation, we can use model accuracy as an objective criterion for selecting one model over another. The idea is that the best model is the one that, after parameter estimation, provides the most accurate description of the mapping between input and output. (Without cross-validation, comparing the accuracies of different models is not very informative: the model with the most flexibility will always be able to provide the best fit to the observed data.)
- Note that there are other criteria that might be important in model selection. Such criteria include simplicity, ease of interpretation, and computational tractability.

6. Overfitting

- The concept of *overfitting* is useful for thinking about why cross-validation is important. For any given set of data, some of the data is signal and some of the data is noise. When we fit a model to data, we should be careful to not fit too much of the data, i.e. overfit the data. This is because if we were to fit all of the data, we will have fit not only the signal but also the noise in the data. Fitting the noise in the data is undesirable since the resulting model will deviate from the true model. In practice, we of course do not know which part of the data is signal and which part is noise. But what we can do is to use cross-validation to help us determine when overfitting is occurring.

- For example, suppose the true model underlying a set of data is quadratic, and suppose we are fitting a model consisting of polynomials of increasing degree (a constant regressor, a linear regressor, a quadratic regressor, a cubic regressor, etc.). As we increase the maximum degree of the polynomials in the model, we will invariably fit the data better and better. However, beyond a certain point, the improvements in fit will mostly reflect fitting the noise in the data instead of the signal. To determine when this overfitting occurs, we can examine the cross-validation performance of the model.

7. Simple models vs. complex models

- It is useful to think of model complexity as a dimension along which models vary. Complex, flexible models have the potential to describe many different types of functions. The advantage of such models is that the true model (i.e. the model that most accurately describes the population from which the data are sampled) may in fact be contained in the set of models that can be described. The disadvantage of such models is that they have many free parameters and it may be difficult to obtain good parameter estimates with limited or noisy data. On the other hand, simple, less flexible models describe fewer types of functions compared to complex models. The advantage of simple models is that they have fewer free parameters and so it becomes feasible to obtain good parameter estimates with limited or noisy data. The disadvantage of simple models is that the types of functions that can be described may be poor approximations to the true underlying function.
- Suppose that for some reason you really want to fit a complex model to some data, but you find that the model is overfit. There really isn't any magical solution: you have to either collect more data, reduce the noise level, change how the data are sampled, or some combination of these approaches.
- *A priori*, it is impossible to say whether a simple or complex model will yield the most accurate fitted model for a given dataset, since this depends on the amount of data available, the nature of the underlying effect, etc. We just have to try different models and see which one works the best.