

CHAPTER OVERVIEW

8: A Few Things to Try Next

HERE are a few suggested exercises to help you learn more about regression modeling using R.

1. Show how you would clean the data set for one of the selected benchmark results (Int1992, Int1995, etc.). For example, for every column in the data frame, you could:
 - Compute the average, variance, minimum, and maximum.
 - Sort the column data to look for outliers or unusual patterns.
 - Determine the fraction of NA values for each column.

How else could you verify that the data looks reasonable?

2. Plot the processor performance versus the clock frequency for each of the benchmark results, similar to Figure 3.1.
3. Develop a one-factor linear regression model for all the benchmark results. What input factor should you use as the predictor?
4. Superimpose your one-factor models on the corresponding scatter plots of the data (see Figure 3.2).
5. Evaluate the quality of the one-factor models by discussing the residuals, the p-values of the coefficients, the residual standard errors, the R^2 values, the F-statistic, and by performing appropriate residual analysis.
6. Generate a pair-wise comparison plot for each of the benchmark results, similar to Figure 4.1.
7. Develop a multi-factor linear regression model for each of the benchmark results. Which predictors are the same and which are different across these models? What other similarities and differences do you see across these models?
8. Evaluate the multi-factor models' quality by discussing the residuals, the p-values of the coefficients, the residual standard errors, the R^2 values, the F-statistic, and by performing appropriate residual analysis.
9. Use the regression models you've developed to complete the following tables, showing how well the models from each row predict the benchmark results in each column. Specifically, fill in the x and y values so that x is the mean of the `delta` values for the predictions and y is the width of the corresponding 95 percent confidence interval. You need only predict forwards in time. For example, it is reasonable to use the model developed with Int1992 data to predict Int2006 results, but it does not make sense to use a model developed with Int2006 data to predict Int1992 results.

	Int1992	Int1995	Int2000	Int2006
Int1992				
Int1995	x(±y)	x(±y)	x(±y)	x(±y)
Int2000		x(±y)	x(±y)	x(±y)
Int2006			x(±y)	x(±y)
Fp1992				
Fp1995	x(±y)	x(±y)	x(±y)	x(±y)
Fp2000		x(±y)	x(±y)	x(±y)
Fp2006			x(±y)	x(±y)

Fp1992	Fp1995	Fp2000	Fp2006

Int1992			$x(\pm y)$	$x(\pm y)$
Int1995	$x(\pm y)$	$x(\pm y)$	$x(\pm y)$	$x(\pm y)$
Int2000		$x(\pm y)$	$x(\pm y)$	$x(\pm y)$
Int2006			$x(\pm y)$	$x(\pm y)$
Fp1992				$x(\pm y)$
Fp1995	$x(\pm y)$	$x(\pm y)$	$x(\pm y)$	$x(\pm y)$
Fp2000		$x(\pm y)$	$x(\pm y)$	$x(\pm y)$
Fp2006			$x(\pm y)$	$x(\pm y)$

10. What can you say about these models' predictive abilities, based on the results from the previous problem? For example, how well does a model developed for the integer benchmarks predict the same-year performance of the floating-point benchmarks? What about predictions across benchmark generations?
11. In the discussion of data splitting, we defined the value f as the fraction of the complete data set used in the training set. For the Fp2000 data set, plot a 95 percent confidence interval for the mean of `delta` for $f = [0.1, 0.2, \dots, 0.9]$. What value of f gives the best result (i.e., the smallest confidence interval)? Repeat this test $n = 5$ times to see how the best value of f changes.
12. Repeat the previous problem, varying f for all the other data sets.

This page titled [8: A Few Things to Try Next](#) is shared under a [CC BY-NC 4.0](#) license and was authored, remixed, and/or curated by [David Lilja \(University of Minnesota Libraries Publishing\)](#) via [source content](#) that was edited to the style and standards of the LibreTexts platform.