points isn't great. Selecting random choices for grid points can help us from falling into the trap of loose grids. Figure 5-4 illustrates this fact.

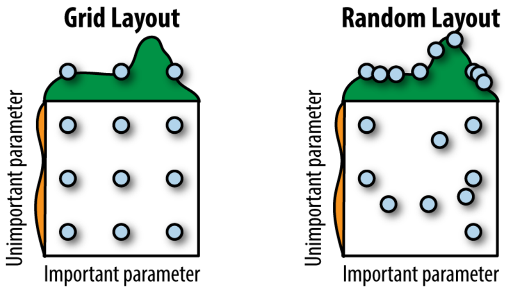

Figure 5-4. An illustration of why random hyperparameter search can be superior to grid search.

How can we implement random hyperparameter search in software? A neat software trick is to sample the random values desired up front and store them in a list. Then, random hyperparameter search simply turns into grid search over these randomly sampled lists. Here's an example. For learning rates, it's often useful to try a wide range from .1 to .000001 or so. Example 5-5 uses NumPy to sample some random learning rates.

Example 5-5. Sampling random learning rates

```
n_rates = 5
learning_rates = 10**(-np.random.uniform(low=1, high=6, size=n_rates))
```

We use a mathematical trick here. Note that $.1 = 10^{-1}$ and $.000001 = 10^{-6}$. Sampling real-valued numbers between ranges like 1 and 6 is easy with `np.random.uniform`. We can raise these sampled values to a power to recover our learning rates. Then `learning_rates` holds a list of values that we can feed into our grid search code from the previous section.

Challenge for the Reader

In this chapter, we've only covered the basics of hyperparameter tuning, but the tools covered are quite powerful. As a challenge, try tuning the fully connected deep network to achieve validation performance higher than that of the random forest. This might require a bit of work, but it's well worth the experience.

Review

In this chapter, we covered the basics of hyperparameter optimization, the process of selecting values for model parameters that can't be learned automatically on the training data. In particular, we introduced random and grid hyperparameter search and demonstrated the use of such code for optimizing models on the Tox21 dataset introduced in the last chapter.

In Chapter 6, we will return to our survey of deep architectures and introduce you to convolutional neural networks, one of the fundamental building blocks of modern deep architectures.