Chapter 4
Loss Functions, Optimal Forecasts, and Forecast Accuracy

1 Introduction

In the last chapter, we learned aout time-series data, which we will be using extensively throughout the rest of this course. In this chapter, we will begin to shift our focus back to forecasting, and talk about loss functions, "optimal" forecasts, and how to assess forecast accuracy.

2 Loss Functions

No forecasting method will ever be perfect, but some may be more accurate then others. In order to form and assess forecasts, we need to be very clear by what we mean by "accuracy". To do so, we will use what is called a **loss function**, which, for our purposes, will transform all of our forecast errors into a single number. Mathematically, we define the forecast error at time period $T + h$ as:

$$f_h = y_{T+h} - \hat{y}_{T+h}$$

where y_{T+h} is the value that actually occured and $\hat{y}_{T+h}$ was our point forecast. Then, our loss function will be expressed as a function of all forecast errors[1] from time periods $h = 1$, the first period we forecasted, through $h = H$, the last period we forecasted:

$$L(f_1, f_2, \ldots, f_H)$$

Note that while I have defined loss functions in terms of *forecast* errors, you can also use loss functions in terms of *in-sample* errors. When expressed as a function of *in-sample* errors, loss functions can be used to compute the optimal parameter estimates in a statistical model.

[1] In general, loss functions can depend on additional parameters. In this class, we will only discuss loss functions that can be expressed as a function of the forecast errors.

For example, if your goal is to produce forecasts that minimize a particular loss function, it is common to estimate parameters of a statistical model under that same loss function. This ensures that the parameters are estimated in such a way that the statistical model will minimize the in-sample loss.

2.1 Symmetric Loss Functions

In many cases, it does not matter whether you forecast too high or too low - that is, the costs associated with under-predicting or over-predicting are identical. For example, suppose you were predicting the high temperature for tomorrow. Obviously the best thing that could happen would be a forecast error of zero (i.e. you correctly predicted the high temperature). However, if you miss by 20 degrees, it may not matter to you whether or not you missed too high or too low - all that matters is you missed by 20 degrees and you dressed incorrectly. You will be equally uncomfortable either way.

When the forecaster does not care about the sign of their forecast error, but instead only cares about the magnitute of the forecast error, the forecaster is said to have a **symmetric loss function**. That is, their loss function will punish a forecast the same if the distance between the forecast and the actual value is the same. For example, a symmetric loss function would punish a forecast error of -20 exactly as harshly as it would punish a forecast error of +20.

Symmetric loss functions are the most commonly used type of loss function used, and the ones we will focus most of our attention on in this class. Below, we introduce two commonly used symmetric loss functions: **squared error loss** and **absolute error loss**.

2.1.1 Squared Error Loss

One commonly used symmetric loss function is squared error loss. This loss function punishes large forecast errors proportionately more than it punishes small forecast errors. This is reasonable under many scenarios. For example, let's return to the weather example above, where you are interested in forecasting the high temperature tomorrow. Being off by 20 degrees is more then 20 times as bad as being off by 1 degree. Why? Well, if you are off by 1 degree, you would have dressed appropriately and you will probably not be uncomfortale at all. If you are off by 20 degrees it is highly likely that you dressed poorly and are either way too cold or way too hot.

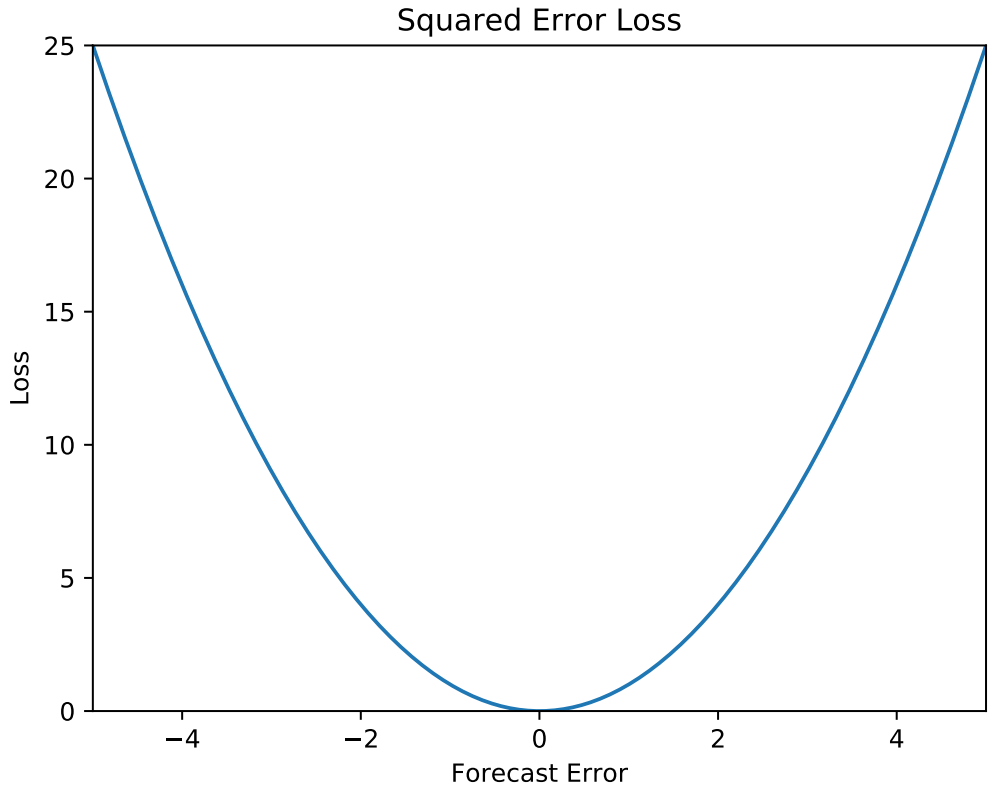
The squared error loss function takes the above logic and puts it into mathematics. It says

that under squared error loss, we square each forecast error (so as to ignore the sign), and then add the terms up:

$$L(f_1, f_2, \ldots, f_H) = f_1^2 + f_2^2 + \cdots + f_H^2$$
$$L(f_1, f_2, \ldots, f_H) = \sum_{h=1}^{H} f_h^2$$

Again, since the loss function is symmetric, we don't care about the sign of the forecast error (if we were 20 degrees high or 20 degrees low), all we care about is the magnitute of our miss. Squaring each observation makes all of the terms ≥ 0. It has the added benefit of punishing big misses proportionately more then small misses, which seems appropriate for the weather example above. Plugging in numbers, a 1 degree forecast error would count as $1^2 = 1$ towards the loss, but a 20 degree forecast error would count as $20^2 = 400$ towards the loss...400 times more! Finally, to compute the total loss, we add all of these positive terms up.

Graphically, we can see how much "loss" a forecasting error would cause, with the loss increasing *quadratically* as the forecast error increases.



2.1.2 Absolute Error Loss

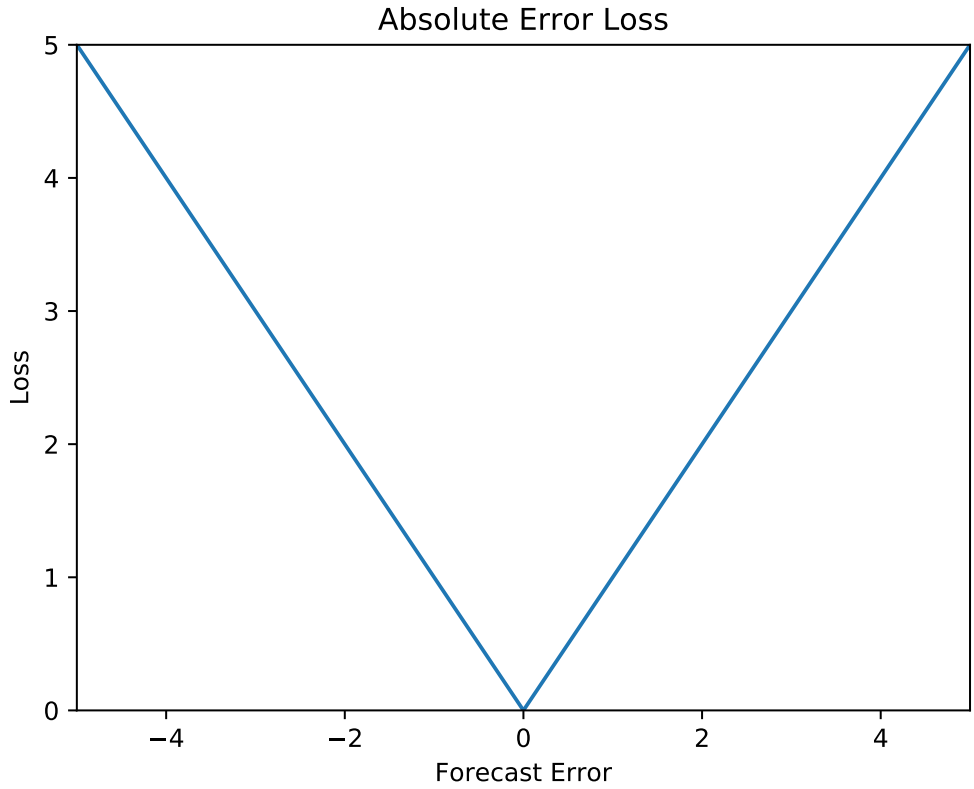
Other times, we may want a symmetrical loss function, but we are not as concerned with punishing large outliers extra harshly. Perhaps we are forecasting daily sales of a product that we already have in stock. For example, suppose we have 1,000 copies of a book in inventory, and are forecasting sales of that book tomorrow. During a typical day, 50 copies of that book are sold. In this case, since we have ample inventory, having a forecast error of 20 books rather than 1 book doesn't seem especially bad. It might make sense to treat this forecast error as exactly 20 times worse then being off by 1 book.

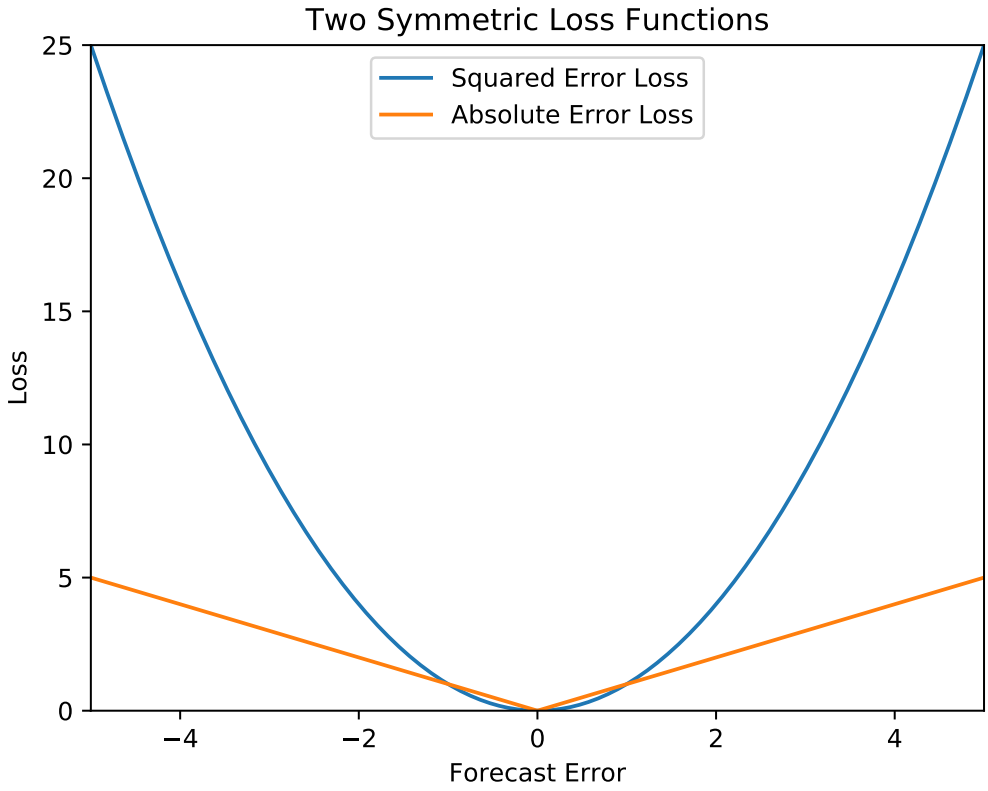
The absolute error loss function provides a symmetrical loss function that punishes forecast errors proportionately. In other words, loss increases linearly in the forecast error. Mathematically,

$$L(f_1, f_2, \ldots, f_H) = |f_1| + |f_2| + \cdots + |f_H|$$
$$L(f_1, f_2, \ldots, f_H) = \sum_{h=1}^{H} |f_h|$$

Since the loss function is symmetric, we take the absolute value so the sign of the forecast error is irrelevant (it doesn't matter if we forecast too low or too high), and only the magnitude of the forecast error matters. Since we are taking the absolute value, forecasts errors twice as large cause exactly twice as much loss. For example, having a forecast error of 2 books leads to an increase of loss of $|2| = 2$, while having a forecast error of -1 books leads to an increase of loss of $|-1| = 1$.

Graphically, we can see how much "loss" a forecasting error would cause, with the loss increasing *linearly* as the forecast error increases.


Absolute Error Loss
Loss
Forecast Error
0
1
2
3
4
5
−4
−2
0
2
4


Two Symmetric Loss Functions
Squared Error Loss
Absolute Error Loss
Loss
Forecast Error
0
5
10
15
20
25
−4
−2
0
2
4

2.2 Asymmetric Loss Functions

While we will primarily work with symmetric loss functions in this course (more specifically, squared error loss), in many forecasting applications an asymmetric loss function may be more appropriate. For example, suppose you are in charge of buying coffee beans for a coffee shop, and that new orders of beans can only be filled on Mondays (for simplicity, suppose you can't drive to the grocery store to buy more beans). You will need to forecast weekly coffee sales so you have an idea of how many beans to buy. In this scenario, you would probably prefer to overforecast on average, so that even during unexpectedly very busy weeks you have enough beans to make drinks for all of your customers. On average, this will mean that you will have increased costs due to carrying extra inventory and increased waste due to having to discard old beans. However, it will ensure that you almost always have enough beans on hand to meet the demand of your customers.[2]

Under scenarios like the one described above, it makes sense to use an **asymmetric loss function**, so that you take into account both the magnitude and sign of forecasting errors. Like symmetric loss functions, there are many ways to do this. One common asymmetric loss function is the piecewise linear or "lin-lin" loss function, which gernalizes the absolute error loss function:

$$L(f_h) = \begin{cases} (1-\alpha)|f_h| & \text{if } f_h \leq 0 \\ \alpha|f_h| & \text{if } f_h > 0 \end{cases}$$

where $0 < \alpha < 1$ represents the relative weight on positive forecast errors. As α increases in size, positive error become punished more harshly. If $\alpha > 0.5$, positive forecast errors are punished more harshly then negative forecast errors.

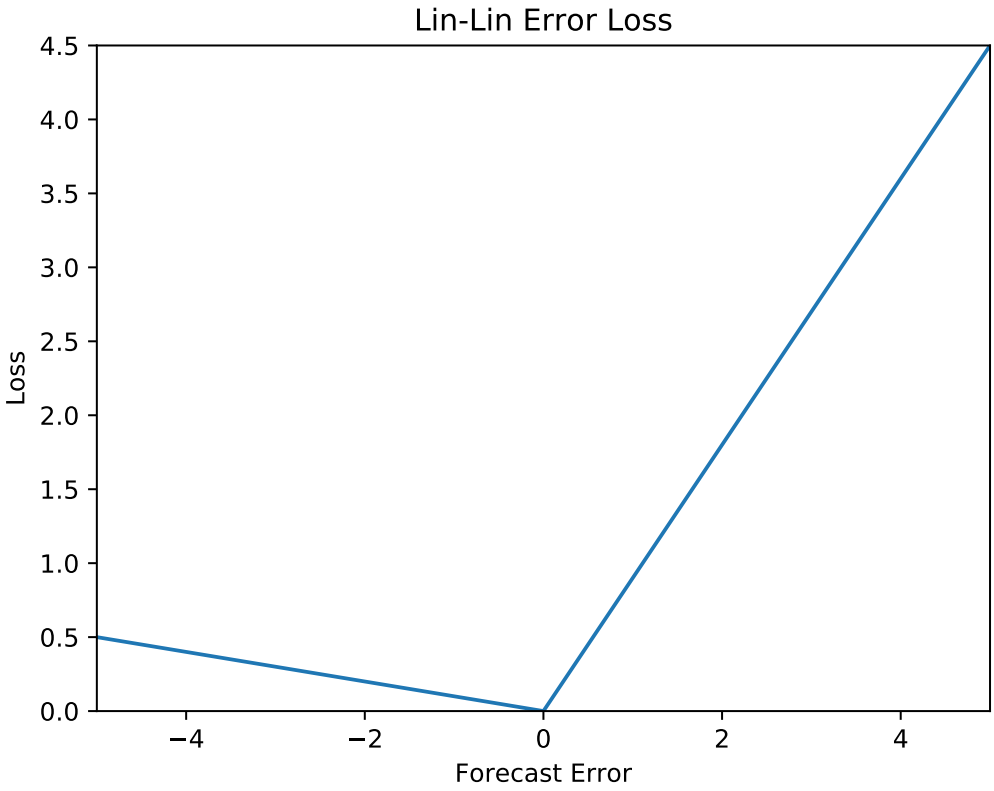
Recall that the forecast error is computed as:

$$f_h = y_{T+h} - \hat{y}_{T+h}$$

Therefore, the forecast error will be positive if we under-forecast and negative if we over-forecast. Continuing with the coffee shop example above, since you do not want to place too small of an order, you want to avoid positive forecast errors (i.e. you don't want actual demand, y_{T+h} to

[2]Note that in this example, if you used the symmetric absolute error loss function you would be expected to run out of beans in 50% of all weeks...Not good!

end up being bigger than forecasted demand, $\hat{y}_{T+h|T}$). Therefore your decision process could be modeled as a lin-lin loss function with $\alpha > 0.5$. For example, with $\alpha = 0.9$, the loss function would look like:



3 Optimal Forecasts

Once we choose a loss function that is appropriate for the foreasting task at hand, we can form **optimal forecasts** - forecasts that, given a particular statistical model and loss function, minimize expected loss. Forming forecasts this way is not guaranteed to produce the smallest amount of loss in all samples, but asymptotically (i.e. as we form more and more forecasts and collect more and more data) these "optimal" forecasts should lead to the smallest amount of loss.

Note that we can only call a forecast "optimal" if we first decide on a statistical model and a loss function - without one (or either) of these, we have no way of computing an optimal forecast. Additionally, if we hold the statistical model constant, changing the loss function can result in a different optimal forecast. Similarly, if we hold the loss function constant, changing the statistical model can result in a different optimal forecast. In other words, an "optimal

forecast" is only optimal under a specific pair of (1) statisitcal model and (2) loss function. We will consider some examples below.

3.1 Optimal Estimate in an Intercept Model

One of the simplest statistical models we will use is the intercepet model. This model says that our random variable, Y, has an intercept and an error term:

$$y_t = b_0 + \varepsilon_t$$
$$\varepsilon_t \sim \text{ iid, mean=0}$$

In other words, this series is always equal to a constant, b_0, plus some random noise, ε_t, which is zero on average. Suppose that we had stationary data, and we thought that the data generating process for our observed random variable, Y, could be approximated by this simple intercept model. Further suppose that we were interested in forming an optimal forecast of y_{T+h}, the value of the data series during any arbitrary period out-of-sample.

3.1.1 Optimal Forecast in Intercept Model under Squared Error Loss

Under squared error loss, our one-period out-of-sample loss function is:

$$L(f_h) = f_h^2$$

Recall that $f_h = y_{T+h} - \hat{y}_{T+h|T}$, where $\hat{y}_{T+h|T}$ is the point forecast and y_{T+h} is the actual value that ends up happening. Substitute this into the loss function:

$$L(f_h) = (y_{T+h} - \hat{y}_{T+h|T})^2$$

We want to choose the optimal forecast that will minimize expected loss:

$$\min_{\hat{y}_{T+h|T}} E(L(f_h)) = E((y_{T+h|T} - \hat{y}_{T+h|T})^2)$$

Using calculus, we can show that the global minimum exists and occurs at:

$$\begin{aligned} 0 &= -2E(y_{T+h|T} - \hat{y}_{T+h|T}) \\ 0 &= -2E(y_{T+h|T}) + 2E(\hat{y}_{T+h|T}) \\ \hat{y}_{T+h|T} &= E(y_{T+h|T}) \end{aligned}$$

Note that $E(y_{T+h|T})$ is the mean of the observed series, Y up to time period T (i.e. the mean through the last in-sample observation). Therefore, conditional on the intercept model, *under squared error loss, the optimal forecast is the* **mean**. In more complicated models, the optimal forecast is the "conditional" mean, a property that will be very useful in more complex models.

3.1.2 Optimal Estimate in an Intercept Model under Absolute Error Loss

Under absolute error loss, our one-period out-of-sample loss function is:

$$L(f_h) = \sum_{t=1}^{T} |f_h|$$

Using the same substitution as the previous section, we want to choose the forecast that will minimize our expected loss:

$$\min_{\hat{y}_{T+h|T}} E(L(f_h)) = E(|y_{T+h|T} - \hat{y}_{T+h|T}|)$$

The calculus is trickier in this case, since there is a discontinuity at $y_{T+h|T} = \hat{y}_{T+h|T}$ (ie. at $f_h = 0$). However, it can be shown that given this interpcept model, *the optimal forecast under absolute error loss is the* **median** of the observed series, Y. Similarly to the squared error loss function, in more complicated models the optimal forecast is the conditional median. While the median of the observed data is fairly simple to compute, the conditional median in a more complex model is harder to compute. Therefore, in practice, absolute error loss is not as commonly used as squared error loss.

3.1.3 Optimal Estimate in an Intercept Model under Lin-Lin Error Loss

Similarly to the absolute error loss above, we can use calculus to find the optimal point forecast under lin-lin loss. Under this intercept model, *the optimal forecast under lin-lin loss is the* α*-quantile* of the observed data, Y. For example, if $\alpha = 0.9$, the optimal foreast is the 0.90^{th}

quantile (i.e the 90^{th} percentile) of the observed data. Thinking back to the coffee shop example, this means that we should forecast relatively high sales so that we do not under-forecast. By setting $\alpha = 0.9$, we should have enough coffee beans on hand to meet demand in 90% of all weeks.

Like absolute error loss, lin-lin loss can be used relatively easily in simple statistical models. However, it is harder to find a specific quantile in more complex statistical models, so we will not use this loss function often in this course.

4 Forecast Quality and Accuracy

It is important to derive optimal forecasts so that we can determine the best possible forecasts we could make under a given statistical model (such as the intercept model). However, in practice we do not know what statistical model will best approximate the data and produce the best forecsats. In the future, we will consider several different types of statistical models, any of which might work best in practice. To help us determine which model is most appropriate, we will use measures of forecast quality and forecast accuracy.

Like our loss functions, the measures of forecast accuracy that we use will be functions only of our forecast errors. That is, given a series of point-forecasts, we compute a series of forecast errors: $f_1, f_2, \cdots, f_H$. Then, we use some numerical measures to determine how accurate our forecasts were, and if our forecasts can be improved.

4.1 Forecast Quality

First, under a symmetric loss function, we would like our forecasts to be unbiased and serially uncorrelated (i.e. NOT autocorrelated). If our forecasts are unbiased, then the average forecast error should be roughly zero. In other words, our forecasts should not consistently miss too high or too low. Mathematically, the mean forecast error (MFE) should be equal to zero:

$$\text{MFE} = \frac{1}{H}\sum_{h=1}^{H} f_h = 0$$

After forming forecasts and then observing the true values, we can perform a statistical hypothesis test to test whether our mean forecast error is equal to zero. If our forecasts are biased they could be improved by correcting the bias. For example, suppose that the average forecast error is -2.0, meaning that on average, we over-forecasted by 2.0. We could simply subtract 2.0

from all of our forecasts to get an unbiased forecast.[3]

We also want our forecast errors to be serially uncorrelated. That is, we want the autocorrelation at all lags to be roughly equal to zero. If the forecast errors are serially correlated, then it means that our forecasting method did not take advantage of all the information we had - our forecasts should be able to be imporved somehow, although it is not always clear exactly how. However, the reverse is not true - just because a model produces unbiased and serially uncorrelated errors does not necessarily mean that it is the most accurate model.

Summarizing:

1. If we are using a symmetric loss function and the forecast errors from a statistical model are biased or serially correlated, there must exist a statistical model that forecasts more accurately.

2. If the forecast errors from a statistical model are unbiased and NOT serially correlated, it does not necessarily mean that there are no statistical models that will produce more accurate forecasts.

4.2 Forecast Accuracy Measures

We will consider two measures of forecast accuracy, each of which is designed to give us a rough idea of how far away our forecasts are from the actual values, on average. The first measure of forecast accuracy we will consider is called **root mean squared forecast error (RMSFE)**, and makes the most sense as a measure of accuracy if we are using squared error loss. The next measure of forecast accuracy is called **mean absolute forecast error (MAFE)**, which makes the most sense if we are using absolute error loss.

The basic idea behind both of these measures is that we want forecasts that minimize loss. However, the total loss is depends in large part on how many periods we are forecasting for, and does not have an easy interpretation. However, by rescaling the loss to put it in the same units as one observation, these forecast measures are easy to interpret.

Mean absolute forecast error (MAFE) starts with the total loss under absolute error loss, $\sum_{h=1}^{H} |f_h|$. Then, take the mean of this total loss, to get the mean loss per observation, or the

[3] Under symmetric loss, this bias correction will decrease the loss associated with our forecasts. In other words, our forecasts will be made more accurate under symmetric loss by adjusting for the bias.

MAFE. Mathematically:

$$\text{MAFE} = \frac{1}{H}\sum_{h=1}^{H}|f_h|$$

where H is the total number of forecast errors being considered. MAFE has a nice interpretation - it tells us how far each forecast has been from the true value, on average. In other words, if you selected a forecast at random, you should expect that it was MAFE units from the true value. Since MAFE is a measure of the average magnitude of error, models that produce forecasts with smaller MAFE are preferred to models that produce forecasts with larger MAFE. for example, it is better to miss by an average of 1 unit then to miss by an average of 10 units.

On the other hand, root mean squared forecast error (RMSFE) starts with the total loss under squared error loss, $\sum_{h=1}^{H} f_h^2$. Then, take the mean of the total loss to get loss per observation, or mean squared forecasting error (MSFE):

$$\text{MSFE} = \frac{1}{H}\sum_{h=1}^{H} f_h^2$$

This gives us the average squared forecasting error for each observation. However, since we have squared the forecasting errors, this measure is in units^2, whereas our observation is in units, so the number produced is still difficult to interpret. To make interpretation easier, we take the square root:

$$\text{RMSFE} = \sqrt{\text{MSFE}}$$

$$\text{RMSFE} = \sqrt{\frac{1}{H}\sum_{h=1}^{H} f_h^2}$$

Now, RMSFE has a nice interpretation - after building in an increased punishment for outliers, the RMSFE tells us roughly how far each forecast has been from the true value. Like MAFE, smaller values of RMSFE are preferred.

In most circumstances, MAFE and RMSFE will provide similar answers - if one statistical model produces a lower MAFE it typically produces a lower RMSFE as well. However, this may not be the case if the model that has a smaller MAFE had a couple of large misses. Since outliers are punished more harshly under RMSFE, it is possible for one model to have the lowest MAFE and another model to have the lowest RMSFE.

I recommend that you think about these accuracy measures in the context of loss functions. If you believe that squared error loss is more appropriate, you should rely on RMSFE. If you believe that absolute error loss is more appropriate, you should rely on MAFE.

4.3 Summary

Summarizing, we consider measures of forecast quality and forecast accuracy when comparing forecasts from different sources. In general, we would like a person or a model to produce forecasts whose errors are both unbiased and serially uncorrelated. If either of these properties is violated, it is possible to find a better forecasting method. Under squared error loss, we prefer models that produce forecasts with smaller RMSFE. Under absolute error loss, we prefer models that produce forecasts with smaller MAFE.

4.4 Exercises

1. Consider the following lin-lin loss function:

$$L(f_h) = \begin{cases} 0.8|f_h| & \text{if } f_h \leq 0 \\ 0.2|f_h| & \text{if } f_h > 0 \end{cases}$$

   (a) Under this loss function, is your loss greater if you under-forecast or if you over-forecast? Explain.

   (b) Under this loss function and the intercept model, what value would be the optimal forecast?

2. Suppose you collected the following data:

Time (h)	Data (Y)	Forecast ($\hat{Y}_f$)	Forecast Error
1	0.97	1.0	
2	0.71	1.0	
3	1.14	1.0	
4	0.25	1.0	
5	1.08	1.0	

   Fill in the last column of the table. Then compute the total loss under the asymmetric loss function given above.

   (a) Compare and contrast squared error loss with absolute error loss.

(b) Using the table from 1(c), compute the MAFE and RMSFE. Are either of these forecast accuracy measures appropriate if our loss function is asymmetric? Explain.

(c) Suppose you were hired as a consultant to make forecasts for a company. Given the problem at hand, you thought that squared error loss was most appropriate. When evaluating the forecasts that you made, which measure of forecast accuracy would you use? Explain.