Conditional probabilities and expectations

Conditional Probabilities and Expectations

Prediction problems can be divided into categorical and continuous outcomes. However, many of the algorithms can be applied to both due to the connection between *conditional probabilities* and *conditional expectations*.

For categorical data, for example binary outcomes, if we know the probability of Y being any of the possible outcomes k given a set of predictors $X = (X_1, \ldots, X_p)^\top$,

$$f_k(x) = \Pr(Y = k \mid X = x)$$

we can optimize our predictions. Specifically, for any x we predict the k that has the largest probability $f_k(x)$.

To simplify the exposition below, we will consider the case of binary data. You can think of the probability $\Pr(Y = 1 \mid X = x)$ as the proportion of 1s in the stratum of the population for which $X = x$. Given that the expectation is the average of all Y values, in this case the expectation is equivalent to the probability: $f(x) \equiv \mathrm{E}(Y \mid X = x) = \Pr(Y = 1 \mid X = x)$. We therefore use only the expectation in the descriptions below as it is more general.

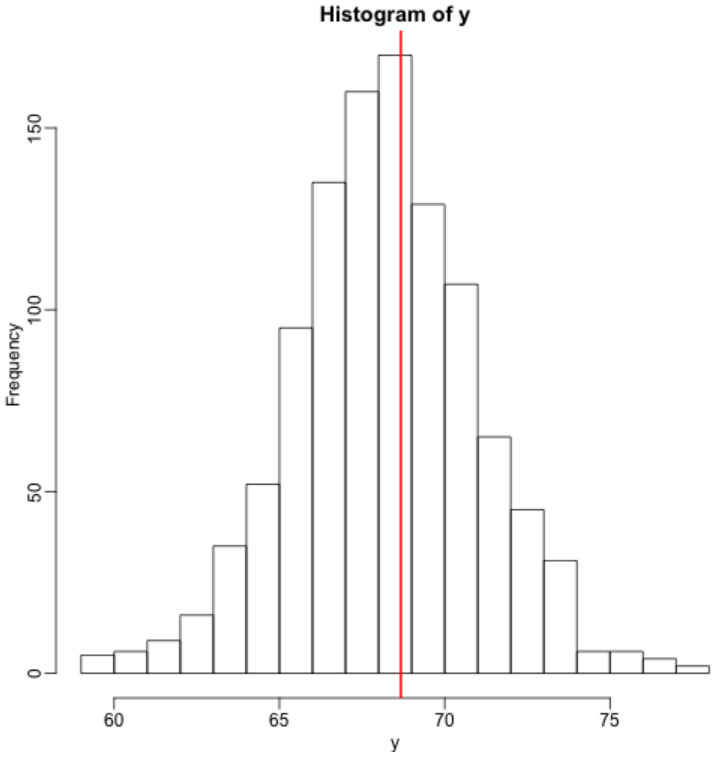
In general, the expected value has an attractive mathematical property and it is that minimized the expected distance between the predictor $\hat{Y}$ and Y:

$$\mathrm{E}\{(\hat{Y} - Y)^2 \mid X = x\}$$

Regression in the context of prediction

We use the son and father height example to illustrate how regression can be interpreted as a machine learning technique. In our example, we are trying to predict the son's height Y based on the father's X. Here we have only one predictor. Now if we were asked to predict the height of a randomly selected son, we would go with the average height:

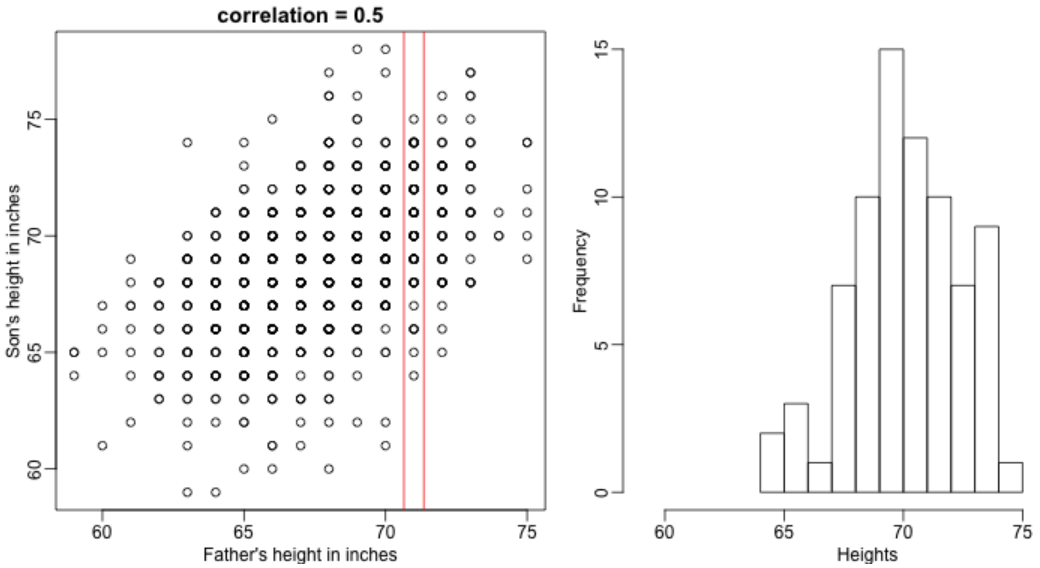
```
library(rafalib)
mypar(1,1)
library(UsingR)
data("father.son")
x=round(father.son$fheight) ##round to nearest inch
y=round(father.son$sheight)
hist(y,breaks=seq(min(y),max(y)))
abline(v=mean(y),col="red",lwd=2)
```



In this case, we can also approximate the distribution of Y as normal, which implies the mean maximizes the probability density.

Let's imagine that we are given more information. We are told that the father of this randomly selected son has a height of 71 inches (1.25 SDs taller than the average). What is our prediction now?

```
mypar(1,2)
plot(x,y,xlab="Father's height in inches",ylab="Son's height in inches",main=paste("correlation =",signif(cor(x,y),2)))
abline(v=c(-0.35,0.35)+71,col="red")
hist(y[x==71],xlab="Heights",nc=8,main="",xlim=range(y))
```



The best guess is still the expectation, but our strata has changed from all the data, to only the Y with $X = 71$. So we can stratify and take the average, which is the conditional expectation. Our prediction for any x is therefore:

$$f(x) = E(Y \mid X = x)$$

It turns out that because this data is approximated by a bivariate normal distribution, using calculus, we can show that:

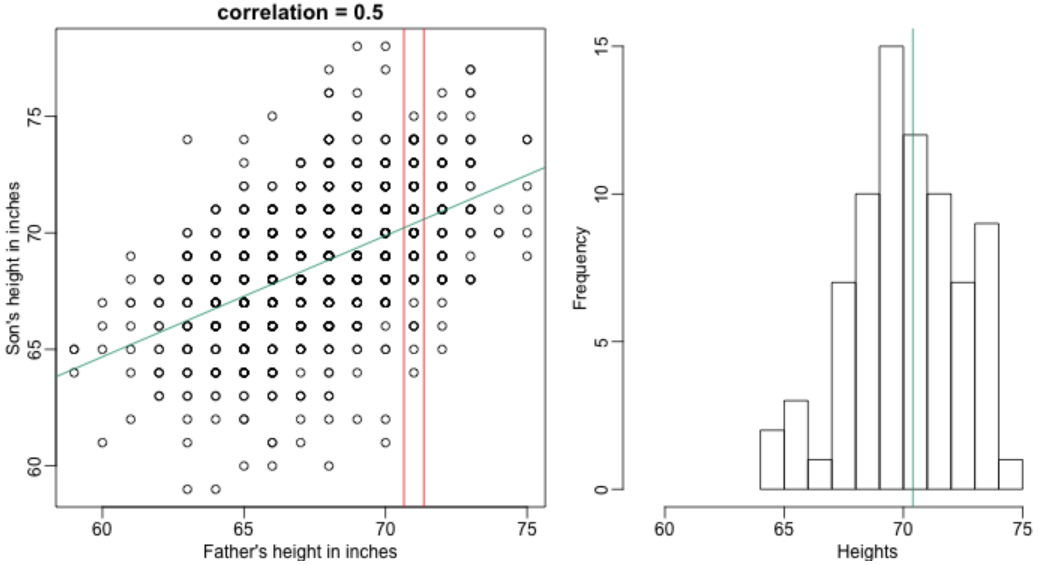
$$f(x) = \mu_Y + \rho\frac{\sigma_Y}{\sigma_X}(X - \mu_X)$$

and if we estimate these five parameters from the sample, we get the regression line:

```
mypar(1,2)
plot(x,y,xlab="Father's height in inches",ylab="Son's height in inches",main=paste("correlation =",signif(cor(x,y),2)))
abline(v=c(-0.35,0.35)+71,col="red")

fit <- lm(y~x)
abline(fit,col=1)

hist(y[x==71],xlab="Heights",nc=8,main="",xlim=range(y))
abline(v = fit$coef[1] + fit$coef[2]*71, col=1)
```



In this particular case, the regression line provides an optimal prediction function for Y. But this is not generally true because, in the typical machine learning problems, the optimal $f(x)$ is rarely a simple line.
