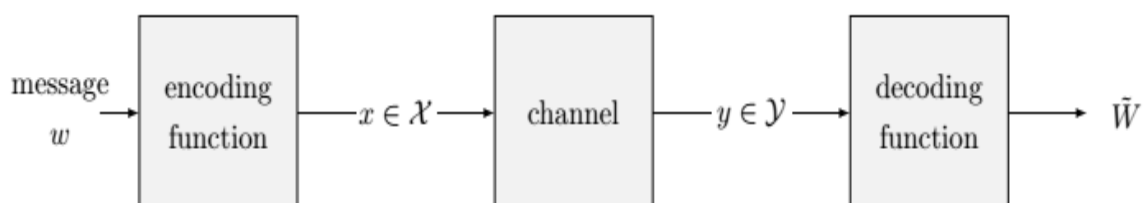


Introduction: Noise

In Module 3, we saw how to encode information from a source such that it can be stored or sent over some channel, and decoded at a later point in time. We assumed that the channel used to send the encoded information was perfect, meaning that no information got altered or lost while being sent over the channel. In this chapter, we consider a different setting, where the channel possibly contains some **noise**, that may convert the channel input x to some potentially different value y :



The goal is to design encoding and decoding functions that can resist this noise, so that the recovered message \tilde{W} is as close as possible to the original message w . The question is how short (efficient) such codes can be while still providing resistance to noise.