Statistical learning in Machine learning

until today, I figure out the reason why to introduce statistical learning to machine learning. In general, model equals to a function with some unknow parameter θ. What we learn from the perspective of statistics is the process of data generating, we let our models to fit and describe this process. To intuitively understand this, suppose the task at hand is image classification. Obviously, data here refers to images with n x h x w. n is the amount, h is the height, w is the width. Diffrent kinds of images have distinguishable properties and lie in the same region of probability density function, which corresponds to a probability that this image belong to this specific kind. Given a large-scale dataset, there is a chance that we can model this data generating distribution by estimating true value of θ, which is the underlying parameter that generates the given dataset. We denote $\hat{\theta}$ as the estimated parameter. Thus, to measure how well the $\hat{\theta}$ is. Bias and variance are proposed in statistics. Bias plays the role in keeping $\hat{\theta}$ close to θ while variance depicts how much we want $\hat{\theta}$ to vary from θ. While for gaussian\bernoulli\exponential distributions, we can compute the form of $\hat{\theta}$ explicitly. But in practice, distributions are hard to modeled, therefore, we can't directly solve a equation to get the desired $\hat{\theta}$, that's why we need building block3(infer hidden parameters) as illustrated in Building blocks for machine learning.