All the problems we've looked at have included data - the time it takes an employee to do a job, amount of chemical produced, etc. We've always assumed that the data was perfect.

What if the data isn't perfect? Then we don't know if the constraints will actually be satisfied in practice.

One approach is to just ignore the noisy/statistical data, solve problems like we've been doing - then hope for the best.

An alternative approach is to account for the noisy/statistical data using "chance constraints." Given a statistical distribution of the data, require that the constraints be satisfied w/ a certain probability.

Suppose that the deterministic constraint is

$$a^T x \leq b$$

where a + b are fixed data in the problem.

If a & b are not actually known, but they are known to belong to statistical distributions, then we may want to ensure that the constraint is satisfied p percent of the time.

$$P\left[a^T x \le b\right] \ge p$$

Probability

If this constraint represents a safety related limit, we may want $p = 0.99$ or $p = 1.0$ to ensure it is almost always satisfied.

Otherwise, we could let p be anything in $[0,1]$. If we pick $p=0$, we are saying there is no constraint.

We can enforce this probabilistic, or chance, constraint by sampling a's & b's from the distribution. We'll take S samples and denote each as a_i, b_i.

Assuming that our sampling is sufficient to be representative of the distribution, we can now add S constraints to the problem - but only require p percent of them to be satisfied. To do this, we'll introduce a binary variable z_i for every sample.

$$\begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & P[a^T x \le b] \ge p \end{aligned} \quad \longrightarrow \quad \begin{aligned} \min \quad & c^T x \\ \text{s.t.} \quad & a_1^T x \le b_1 + M(1-z_1) \\ & a_2^T x \le b_2 + M(1-z_2) \\ & \vdots \\ & a_m^T x \le b_s + M(1-z_s) \\ & \frac{\sum z_i}{m} \ge p \end{aligned}$$

Big M

When $z_1 = 1$, the constraint for the first sample is active.

When $z_1 = 0$, the constraint for the first sample is not active.

And so on for each z_i.

The last constraint ensures that at least p percent of the constraints are active.