Problem 2. Write up your Project Proposal with the following sections. The result should be 2-3 pages long.

2a. *Goal*: What is your primary project goal? What you would like to learn?

I want to study the application of ϵ-Machines to model-based reinforcement learning. That is, beyond understanding a system, an agent must *intervene* on the system to achieve outcomes they consider desirable. This consists of the observer understanding the physical system they are confronted with, determining how their behavior influences the system, and selecting a policy to maximize some goal.

Overall, the system under study consists of three parts:

1. A physical system: Though there are many possible systems an agent might want to control, I want to look at cases where the system is a hidden Markov model.
2. An agent-system interaction: This determines how the agent's possible actions perturb the physical system.
3. An agent: The agent is specified by a set of possible actions, a utility function (preferences over observed sequences), and an algorithm which chooses actions based on observations. The agent first observes the physical system and develops a model of it. Then, the agent performs different actions and observes their influence on the physical system, creating a new model of it's interactions with the system. Finally, it selects a policy based on its understanding of the system in order to obtain desired outcomes.

2b. *System*: Describe how the dynamical system is nonlinear and time-dependent.

What's the state space?

The state space is determined by the possible states of the physical system, the set of possible actions the agent can take, and the interaction between the agent and the system.

What's the dynamic?

The agent-system pair begins with the agent simply observing the dynamics of the system itself. Then, the agent begins to implement a procedure for choosing actions based on observations. This induces entirely new dynamics. For example, imagine the agent has a fixed policy in response to observed sequences; in this situation, the agent-system combination will *itself* be well described by an ϵ-Machine. However, the agent can observe how well its current policy achieves its goals and update the policy. Because of this, the agent-system dynamic will evolve into an equilibrium where the agent can no longer improve upon it's current policy.

Why is the system behavior interesting?

Reinforcement learning is a highly general problem and is considered the ultimate goal of artificial intelligence research. Studying the application of ϵ-Machines to reinforcement learning can provide better information-theoretic foundations of RL and potentially help create new RL algorithms with better performance. Additionally, the system-agent model extends the study of ϵ-Machines to situations where the observer strongly influences the system under study, which may produce new concepts applicable to the study of quantum systems. Finally, having a foundational understanding of how agents respond to their environment can help us understand interactions between agents and the evolutionary dynamics which arise when many agents compete.

2c. *Dynamical properties*: What dynamical properties are you going to investigate?

This question is not particularly applicable to my project since I anticipate the agent will converge to a fixed policy rather than constantly update. However, the dynamics of the converged agent-system pair may be of interest.

2d. *Intrinsic computation properties*: What information processing properties are you going to investigate?

There are two important properties of the system-agent interaction. First, I want to analyze how much information is required for an agent to find a good policy and how this relates to the system's statistical complexity (along with other information measures). Second, I want to determine how sophisticated the agent's policy must be in order to adequately control the system with an eye towards how this sophistication grows with the statistical complexity of the system.

2e. *Methods*: What methods will you use? Why are they appropriate?

I plan on modeling the system itself as a hidden Markov model, which is appropriate because many systems can be approximated as HMM's (and it becomes easy to compute information measures). I plan on modelling the agent as a utility maximizer, which, based on the VNM utlity theorem, appropriately describes any rational agent. The specific utility model will assign utilities to words of a specified depth, so that the agent will seek to maximize the utility of the regulated system's word distribution. Thus far, I have not determined the appropriate RL algorithm to use, though it must involve the agent experimenting with different possible actions, observing their effects, and updating it's policy to maximize expected utility.

2f. *Hypothesis*: What is your current guess as to what you will find?

The amount of information the agent is required to collect will likely depend on the number of possible actions the agent has available as well as the complexity of the system-agent interaction. For example, if the effect of an action only depends on the state of the system and the current action, the agent only needs to examine how each individual action influences the system in different states. However, if the system behaves differently based on long histories of previous actions, the agent will have to test many sequences of actions in order to find a good policy.

Based on the Good Regulator Theorem, I anticipate that the statistical complexity of the policy required for an agent to have good performance will be roughly proportional to the statistical complexity of the system itself. Essentially, the agent will have to develop an internal model of the system in order to control it.

2g *Steps*: List the appropriate steps for your investigation; for example, read literature, write simulator, do mathematical analysis, estimate properties from simulation, write up report, and so on.

1. Review the literature on model-based RL, with a focus on the control of hidden Markov models (0.5 weeks)
2. Choose a simple system to control, specify a utility function, write a simple RL algorithm, and model the system-algorithm interaction (1 week)
3. Update model in the previous step based on lessons learned and apply to a more sophisticated system (1 week)
4. Compute information measures for the agent's best policy compared to the statistical complexity of the system itself as a function of system complexity (0.5 weeks)
5. Write report and do additional work as needed (1 week)

2h *Time*: Estimate how long each step will take. Can you complete the project within one month?

See times above. I believe I can complete this in 1 month.

```
In [0]:
```