1. Supervised learning is when a programmer is attempting to determine the effects that other variables have on a single variable. That single variable is provided during the training phase, and the algorithm is tested for accuracy in predicting it in the testing phase. Unsupervised learning is when the data is aggregated and no specific variable is targeted, this is often used for clustering, etc.
2. I play a few online video games, and I think that it would be useful to determine the likelihood of a new account being one of a cheater (in Counterstrike, for example, cheaters use 'aimbots', and are frequently banned in the first two weeks of playing with the aimbots—the game is relatively inexpensive, so the cheaters often purchase a new copy and start to play again with the aimbot enabled). The data collected could be the average accuracy of a new account, as well as things like win percentage, etc. This could be used to determine the likelihood of a cheater before anyone reports them as such—thus removing them more efficiently (otherwise oversight from players is required). Obviously if someone was just very skilled it would be possible that the algorithm might flag them as a cheater as well, so the data collected would have to be able to track mouse movement speed, etc. as well as looking for machine like patterns in aiming (humans make mistakes with aim, no matter how good, an aimbot would not).
3. Another thing of interest in Counterstrike is the price and rarity of 'weapon skins'. People pay unreal amounts of money for 'rare' weapon skins, easily several hundred dollars. I think that it would be of great value to attempt to reverse engineer the probabilities of these skins being given to players (it seems arbitrary), as this could be used to collect rare skins, or determine under/overvaluation of a skin, and thus make money off the sale/purchase of it. The estimate would involve pulling the data, cleaning it, and using Bayes' to determine approximately the rarity of various items.
4. a. regression, inference
   b. classification, prediction
   c. regression, prediction
5. I think that predicting the value of a house before it goes on market is a problem largely of 'learning' as opposed to design. This is because the relevant data is easy to accumulate and aggregate (area, size of property, floors, bathrooms, etc.). While it could still benefit from design (choosing pertinent variables) most of the irrelevant ones would be dropped by an efficient algorithm.
6. I expect to use python and Azure ML. This is because I obviously want more experience with Azure ML, which is definitely a promising technology. I would also like to use python because I want to further develop my skills in python, and I think that the usage of it will only help my prospects.
7. Completed!