Answer the questions in the spaces provided on the question sheets. If you run out of room for an answer, continue on the back of the page.

1. [17 points]

   (a) Below each scenario (i, ii, iii, iv) indicate which learning paradigm does it belong to:

      a) supervised learning

      b) unsupervised learning

      c) reinforcement learning.

   In addition, provide a brief justification.

      i. (2 points) You are given high-dimensional data that describes the position of molecules in a complex system. You want to better understand this data by visualizing it in a lower-dimensional space (e.g., in 2 dimensions).

      ii. (2 points) You are tasked with developing a learning-based English-to-French translation system. You are given a set of English text along with their French translations.

iii. (2 points) You must program a system to play a video game involving a princess saving a prince. To do so the princess must sequentially go through different levels. At each level she must find a sequence of actions that will allow her to move to the next level.

iv. (2 points) Consider the previous example (princess saving a prince). Does your answer change if instead your task is solely to recognize the objects in the video game's screen throughout the game?

(b) (2 points) Describe a learning scenario (e.g., a task) in which a combination of supervised and unsupervised could be helpful?

(c) (4 points) Describe what is a *regularizer* and give an example of a regularizer.

(d) (3 points) Define the i.i.d. assumption and describe why it is relied upon.

2. [16 points]

Unsupervised Learning

(a) (4 points) List two advantages of clustering with a Gaussian mixture model (GMM) compared to K-Means?

(b) (6 points) K-Means is meant to cluster real-valued data. How would you modify k-means clustering to cluster other types of data (e.g., integers)? The K-Means algorithm is reproduced below for convenience. You can refer to line numbers when describing your changes.

```
1  Algorithm: K-Means clustering
2  Inputs: Number of clusters $K$ and data $x_1, x_2, \ldots, x_n$;
3  Initialize cluster centers $\mu_1, \mu_2, \ldots, \mu_K$;
4  Initialize cluster responsabilities $r_1, r_2, \ldots, r_n$;
5  while Not converged do
6      1) Update cluster responsabilities:
7        $r_i = \arg\min_{r_i} \sum_k r_i \parallel x_i - \mu_k \parallel^2 \;\; \forall i$;
8      2) Update cluster centers:
9        $\mu_k = \arg\min_{\mu_k} \sum_i r_i \parallel x_i - \mu_k \parallel^2 \;\; \forall k$;
10 end
```

(c) (6 points) Could K-Means be distributed using Apache Spark? If so, please describe how to do it and how much of a speed up could you expect (for example, in terms of computation time). If not, please justify your answer.

3. [22 points]
   Deep Learning

   (a) Neural network size.

   i. (3 points) What is the minimum number of parameters a fully-connected feed-forward neural network requires to model data with 100 features and 2 outputs?

   ii. (3 points) For the same task as above, how many parameters does a neural network with 2 hidden layers of size 20 each has?

(b) Neural networks for recommender systems.

i. (8 points) Describe a way to model ratings data using a neural network. In particular, you can imagine that you have access to past ratings data. Each datum is a combination of a user ID, an item ID, and a rating. Each rating is a real number between 0 and 1.

$$X = [(1,5),(10,2),(2,10),\dots,(5,5)]$$
$$Y = [0.1,0.5,0.2,\dots,0.9]$$

This data shows that user 1 gave a rating of 0.1 to item 5.
Please be as specific as possible. You can answer with a sketch (a drawing) of your proposed neural network.

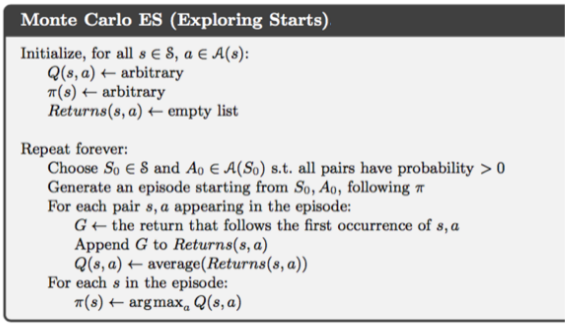
ii. (8 points) Imagine a second scenario where instead of modelling ratings you only wish to model a user's sequence of consumed items. Further assume that the last consumed items tend to be predictive of the next items that will be consumed.
Suggest a second neural network to model this situation. You are (again) encouraged to answer using the sketch of your proposed neural network. Please clearly specify: 1) the inputs, 2) the outputs of your proposed model, and 3) how will it be used to perform recommendations (i.e., at test time).

4. [29 points]

Reinforcement Learning

(a) (2 points) Describe the usefulness of the discount factor (γ) in Markov decision processes (MDPs).

(b) Consider the Monte Carlo with exploring starts method (reproduced from the slides below).



Monte Carlo ES (Exploring Starts)

```
Initialize, for all $s \in \mathcal{S}$, $a \in \mathcal{A}(s)$:
    $Q(s,a) \leftarrow$ arbitrary
    $\pi(s) \leftarrow$ arbitrary
    $Returns(s,a) \leftarrow$ empty list

Repeat forever:
    Choose $S_0 \in \mathcal{S}$ and $A_0 \in \mathcal{A}(S_0)$ s.t. all pairs have probability $> 0$
    Generate an episode starting from $S_0, A_0$, following $\pi$
    For each pair $s, a$ appearing in the episode:
        $G \leftarrow$ the return that follows the first occurrence of $s, a$
        Append $G$ to $Returns(s,a)$
        $Q(s,a) \leftarrow$ average($Returns(s,a)$)
    For each $s$ in the episode:
        $\pi(s) \leftarrow \text{argmax}_a Q(s,a)$
```

i. (5 points) Describe the goal of this algorithm.

ii. (3 points) What do we mean with by "exploring starts"? In environments where this is not practical, what is another technique that is used instead of exploring starts?

iii. (10 points) Consider a new environment with 4 states (A, B, C, D) and 2 actions (L, R). For all states and actions write down the $Q(s, a)$ and $\pi(s)$ calculated by the Monte Carlo ES method after considering the following two episodes.
Consider that each element of $Q(s, a)$ was initialized to 0.0, $\pi(s)$ was initialized uniformly at random, and a $\gamma = 1$.

Episode 1: $(B, L, 5) \rightarrow (A, L, 5) \rightarrow (C, L, 2) \rightarrow (A, L, 5) \rightarrow (C, R, 0)$;
Episode 2: $(C, R, 2) \rightarrow (A, L, 5) \rightarrow (C, L, 2)$;

Each tuple represents a state, action, reward triplet. E.g., (A,L,5) means that the agent started in state A, executed action L and received a reward of 5.

(c) You are tasked with creating a reinforcement learning agent for driving a car (i.e., for autonomous driving). Describe exactly how you would setup the reinforcement learning problem in terms of actions, states, and rewards.

i. (2 points) Actions:

ii. (4 points) States:

iii. (3 points) Rewards: