Section A: Short Answer Questions [28 marks]

Answer each of the questions in this section as briefly as possible. Expect to answer each subquestion in a couple of lines.

Question 1: Short Answer Questions [28 marks]

1. What is the primary difference between “supervised” and “unsupervised” learning? [1.5 marks]

   A: *whether the training instances are explicitly labelled or not*

2. For each of the following features, indicate which of “numeric”, “ordinal”, and “categorical” best captures its type: [4 marks]

   (a) blood pressure level, with possible values {**low**, **medium**, **high**}

   A: *ordinal*

   (b) age, with possible values **[0,120]**

   A: *numeric*

   (c) weather, with possible values {**clear**, **rain**, **snow**}

   A: *categorical*

   (d) abalone sex, with possible values {**male**, **female**, **infant**}

   A: *categorical*

3. Describe a strategy for measuring the distance between two data points comprising of “categorical” features. [1.5 marks]

   A: *Hamming distance OR cosine similarity OR jaccard OR dice*

4. What is the relationship between “accuracy” and “error rate” in evaluation? [1.5 marks]

   A: *accuracy* $= 1-$ *error rate*

5. With the aid of a diagram, describe what is meant by “maximal marginal” in the context of training a “support vector machine”. [1.5 marks]

   A: *the width of the margin (= distance from separating hyperplane and the support vectors) should be maximised*

6. What makes a feature “good”, i.e. worth keeping in a feature representation? How might we measure that “goodness”? [3 marks]

   A: *good = correlation/association with category of interest (and non-redundant)*

7. For each of the following models, state whether it is canonically applied in a “classification”, “regression” or “clustering” setting: [6 marks]

   (a) multi-layer perceptron with a softmax final layer

   A: *classification*

   (b) soft k-means

   A: *clustering*

   (c) multi-response linear regression

   A: *classification*

   (d) logistic regression

   A: *classification*

(e) model tree

A: *regression*

(f) support vector regression

A: *regression*

8. With the aid of an example, briefly describe what a "hyperparameter" is. [3 marks]

   A: *a top-level setting for a given model (which is set prior to training)*

9. With the use of an example, outline what "stacking" is. [1.5 marks]

   A: *combining the output of a number of base classifiers as input to a further supervised learning model*

10. What is the convergence criterion for the "EM algorithm"? [1.5 marks]

    A: *convergence of maximum log-likelihood to within an episilon (small) change*

11. Outline the basis of "purity" as a form of cluster evaluation. [1.5 marks]

    A: *what proportion of instances in the cluster correspond to majority class*

12. What is the underlying assumption behind active learning based on "query-by-committee"? [1.5 marks]

    A: *disagreement between base classifiers indicates that the instance is hard to classify, and thus will have high utility as a training instance*

Section B: Methodological Questions [30 marks]

In this section you are asked to demonstrate your conceptual understanding of a subset of the methods that we have studied in this subject.

Question 2: Random Forests [7 marks]

"Random forests" are based on decision trees under different dimensions of "randomisation". With reference to the following toy training dataset, provide a brief outline of two (2) such "random processes" used in training a random forest. (You should give examples as necessary; it is not necessary to draw the resulting trees, although you may do so if you wish.)

Instance ID	*Feature 1*	*Feature 2*	*Feature 3*	*Class*
A:	1	1	1	`True`
B:	0	2	0	`True`
C:	3	0	0	`False`
D:	1	1	2	`False`
E:	2	0	2	`False`

A:

1. *random sampling of training instances (similarly to bagging)*
2. *random subsampling of attributes for given decision tree*
3. *random construction of new features based on linear combinations of numeric features*

Question 3: HMMs [9 marks]

This question is on "hidden Markov models".

1. In the "forward algorithm", $\alpha_t(j)$ is used to "memoise" a particular value for each state j and observation t. Describe what each $\alpha_t(j)$ represents. [3 marks]

   A: *the probability of observing all observations up to and including t and ending up in state j*

2. In the "Viterbi algorithm", two memoisation variables are used describe for each combination of state j and observation t: (1) $\beta_t(j)$ (which plays a similar role to $\alpha_t(j)$ in the forward algorithm); and (2) $\psi_t(j)$. Describe what each $\psi_t(j)$ represents. [3 marks]

   A: *the most probable immediately preceding state for state j given observations up to and including t*

3. Why do we tend to use "log probabilities" in the Viterbi algorithm but not the forward algorithm? [3 marks]

   A: *Viterbi based on multiplication, so can convert to sum of log probabilities; forward based on sum of product of probabilities, so logging the probabilities doesn't help in the calculation*

Question 4: Model learning [14 marks]

We have discussed that the objective in much of supervised machine learning is to derive a model that explains ("fits") a set of (labelled) data. In a basic "curve fitting" scenario, we will assume that all of the needed data is available to us at the time of building the model; the objective is to fit a model to that data. In machine learning, in contrast, we have only a subset of possible data available to us when we build the model. This contrast has certain implications for how we approach machine learning, which we explore in this question.
Discuss the implications of knowing that we do not have all possible data for a given problem, in terms of the following aspects:

1. Is our primary objective in machine learning to derive a model that fits the subset of the data that we do have? Why or why not? [3 marks]

   A: *No. Want to build a model that generalises to new data.*

2. Explain how we can use our limited data, in a machine learning context, to demonstrate whether or not our objective has been met. [3 marks]

   A: *Split data into training/dev/test. Need to measure generalisation from "best" (tuned) model to unseen data.*

3. Identify and explain one important problem that can emerge with respect to this primary objective, even if we are successful in deriving a good model for the data that we have. Name one specific technique discussed in class that can be applied to mitigate this problem, and explain how it does so. [3 marks]

   A: *overfitting; model does well on training data but poorly on test data*
   technique: L1/L2/Lasso regularisation
   how: reduce complexity of model/constrain model function

4. Define "bias" and "variance", indicating how we might detect each one. Discuss how bias and variance relate to each other in the context of our primary objective. [5 marks]

   A: *bias: how well our model approximates the (training) data; approximation error (high bias=consistently poor performance)*

   variance: how well our model generalises to held-out (test) data; estimation error (high variance=sensitive to training data)

   Relationship: Increasing the complexity of our model tends to reduce bias (by fitting the data better) but increase variance (because the model will be more strongly fit to the training data); in order to obtain generalisation we seek to minimise both bias and variance in order to have a good model that generalises well (isn't overfit to training). Increase training examples and control for overfitting to lower variance.

Section C: Numeric Questions [42 marks]

In this section you are asked to demonstrate your understanding of a subset of the methods that we have studied in this subject, in being able to perform numeric calculations. Questions 5 and 6 both make use of the following training data set:

early	tie	label
N	Y	dinner
Y	N	tea
Y	N	dinner
N	N	dinner
Y	N	dinner
N	N	tea

Table 1: Training Dataset for Questions 5 and 6

Question 5: Naive Bayes [9 marks]

Given the training dataset from Table 1:

1. Using the method of "maximum likelihood estimation" (without smoothing), compute $P(\texttt{dinner})$ and $P(\texttt{dinner}|\texttt{tie} = \texttt{Y})$. [1.5 marks]

   A: $P(\texttt{dinner}) = \frac{2}{3}$
   $P(\texttt{dinner}|\texttt{tie} = \texttt{Y}) = 1$

2. Apply the method of "Naive Bayes" as it was discussed in the lectures, to predict the label of the test instance {early=N, tie=N}; show your workings. [7.5 marks]

   A: $P(\texttt{dinner}|\texttt{early} = \texttt{N}, \texttt{tie} = \texttt{N}) = \frac{4}{6} \cdot \frac{2}{4} \cdot \frac{3}{4} = \frac{1}{4}$
   $P(\texttt{tea}|\texttt{early} = \texttt{N}, \texttt{tie} = \texttt{N}) = \frac{2}{6} \cdot \frac{1}{2} \cdot \frac{2}{2} = \frac{1}{6}$

   Predicted class label is dinner

Question 6: Decision Trees [15 marks]

1. Briefly explain — in at most two sentences — the basic logic behind the "ID3" algorthmic approach toward building decision trees. This should be focussed on labelling the nodes and leaves; you do not have to explain edge-cases. [3 marks]

   A: *Recursively determine which feature has the highest information gain (i.e. does the best job of partitioning the data into pure subsets) over the subset of training instances selected by the path to that node, and add a branch for each value of that feature; continue until every leaf node is pure (and label with corresponding class)*

2. The criterion for labelling a node, as explained in the lectures, was based around the idea of "entropy" — what does entropy tell us about a node of a decision tree? [3 marks]

   A: *how skewed ("pure") the label distribution is (lower entropy → more skewed → better)*

3. A very similar alternative to entropy is the so-called GINI coefficient, defined as follows:

$$GINI = 1 - \sum_{j \in C} [p(j)]^2$$

   Calculate the GINI coefficient based on the labels in Table 1. [3 marks]

   A: $1 - [(\frac{1}{3})^2 + (\frac{2}{3})^2] = \frac{4}{9}$

4. Extend the notion of "Information Gain" to "GINI Gain", and demonstrate why `tie` would be chosen as the root of the decision tree on the given data. [3 marks]

   A: *GINI Gain = GINI − weighted sum of GINI across children*
   GINI Gain for `early` = $\frac{4}{9} - (\frac{1}{2}\{1 - [(\frac{1}{3})^2 + (\frac{2}{3})^2]\} + \frac{1}{2}\{1 - [(\frac{1}{3})^2 + (\frac{2}{3})^2]\}) = 0$
   GINI Gain for `tie` = $\frac{4}{9} - (\frac{1}{6}\{1 - [1^2]\} + \frac{5}{6}\{1 - [(\frac{2}{5})^2 + (\frac{3}{5})^2]\}) = \frac{4}{9} - \frac{5}{6} \cdot \frac{12}{25} = \frac{4}{9} - \frac{2}{5} \approx 0.0404$
   Hence, `tie` *selected as first feature*

5. Why will the resulting decision tree have difficulty classifying test instances like {`early=N`, `tie=N`}? [3 marks]

   A: *because there is a tie (NJ: terrible pun?) in the training data for that feature combination ("impure" leaf)*

Question 7: Nearest Prototype and k-Nearest Neighbour [10.5 marks]

Given the following training dataset:

abv	opacity	label
4.8	−0.20	ale
5.2	−0.10	ale
5.0	−0.33	ale
4.7	−0.02	lager
5.1	−0.23	lager
4.6	−0.05	lager

1. Generate the "prototype" for each class in the training data. [3 marks]

   A: `ale` = $(5.0, -0.21)$
   `lager` = $(4.8, -0.1)$

2. For the test instance {`abv`=5.1, `opacity`=−0.23}, determine which of the prototypes is more similar. (You will need to choose an appropriate metric, and show your work; do not just use inspection.) [3 marks]

   A: *Manhattan distance:*
   MD(`ale`*,T) :* $|5.0 - 5.1| + |-0.21 - (-0.23)| = 0.12$
   MD(`lager`*,T) :* $|4.8 - 5.1| + |-0.1 - (-0.23)| = 0.43$
   `ale` *is more similar*

3. What should happen if we instead use the "1-Nearest Neighbour" method to predict the label of that test instance? (You do not need to show your work.) [1.5 marks]

   A: *Nearest neighbour is* `lager` *(exactly the same as T)*

4. Give one possible reason why each of these two methods could reasonably claim to be making a better prediction for this instance. [3 marks]

   A: *1-NN : we've already seen this beer, and we know it's a* `lager`
   NP : that `lager` *instance looks like an outlier; most beers that are dark and strong are* `ale`

Question 8: Evaluation [7.5 marks]

Assume that our development set contains 100 instances (truly) labelled as one of three classes as follows: 50 `acro` instances, 30 `base` instances, and 20 `claw` instances. We then build a classifier, apply it to this dataset, and observe the following confusion matrix:

1. Determine the "classification accuracy" of the system described above. [1.5 marks]

   A: *Accuracy: number correct divided by total:* $\frac{50}{100} = 0.5$

		Actual		
		acro	base	claw
Predicted	acro	28	5	7
	base	10	10	0
	claw	0	8	12

2. Calculate the "micro-averaged precision" and "macro-averaged precision" of this system. (Show your workings; you should simplify these to a single fraction or decimal value.) [3 marks]

A: *M-P* : $\frac{1}{3}(P(\texttt{acro}) + P(\texttt{base}) + P(\texttt{claw}))$
M-P : $\frac{1}{3}(\frac{28}{40} + \frac{10}{20} + \frac{12}{20}) = 0.6$
μ-P : sum of true positives divided by sum of (true positives and true negatives)
μ-P : $\frac{28+10+12}{28+5+7+10+10+8+12} = \frac{50}{80}$

3. In some contexts these three values are equal; here they are not. Briefly explain why. [3 marks]

A: *All three are equal when the system predicts a single label for every instance, and the classes are evenly distributed*
Here: some instances (probably 20) not classified; more `acro` instances and fewer `claw` instances (NJ: former not nec?)

Section D: Design and Application Questions [20 marks]

In this section you are asked to demonstrate that you have gained a high-level understanding of the methods and algorithms covered in this subject, and can apply that understanding. Expect to respond using about one-third of a page to one full page, for each of the three points below. These questions will require significantly more thought than those in Sections A–C and should be attempted only after having completed the earlier sections.

Question 9: [20 marks]

You are tasked with building a system which labels images as to whether they contain a given product type (e.g. `car` or `mobile phone`), based on a large set of labelled training instances. A given image may contain multiple or no product types, with the expectation that most (but not all) test instances will contain at least one product.

Each image has been transformed into an "embedding" (i.e. a dense real-valued vector representation of the image). This transformation process should be treated as a "black-box", which will be applied consistently to every image in the collection.

At the start of the project, you are provided with training instances for each of 500 product types, but as the project progresses, the set of product types is to be expanded in increments of around 100 new types, and new training instances are to be provided for each of the newly-added product types, up to a final total of around 1000 product types.

You will additionally be provided with extra training instances for a subset of the pre-existing product types, e.g. to capture newly-released models of cars or mobile phones. However, it is desirable to have a model that does not reverse positive predictions for a pre-existing category; rather, you are instructed to employ this extra training data to find further instances of the corresponding product type.

Finally, for each product type, you are provided with a "priority level" (high, medium or low) for how critical it is that your model has good coverage in correctly identifying instances of that product type.

Outline the following:

- the type of machine learning algorithm that you will use, and why; state any assumptions you are making in your answer.
- how you will deal with updates to the label set and also extra training instances for pre-existing labels, making specific mention of how you will maintain consistency in your model predictions, but still have your model find new instances of a given product type.
- how you will evaluate your model.

↭⋐⋐⋐⋐⋐⋐⋐ END OF EXAM ⋑⋑⋑⋑⋑⋑⋑↭