Machine Learning, Spring 2018: Final Exam

Your name:

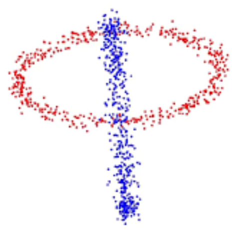
How this works. *On the back side you find a list of 30 claims which are either true or false. To the left of each claim you find two boxes, one of which has a tiny* f *and the other a tiny* t *shown in it. If you think the claim is true, tick the box with the tiny* t*, and if you think the claim is false, tick the* f *box. In this way you can get a maximum score of 10 correct answers.*

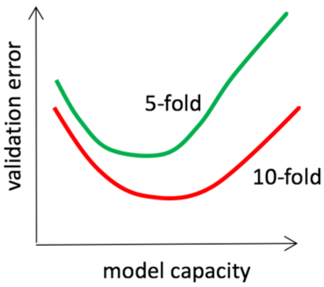
How this is scored. *If you get everything right, you earn a 100% score. If you get 15 out of 30 right (which means random guessing), the score is 35%. (grade 5.0). Anything else is obtained by linear interpolation, and never less than 0%. This spells out to what you find in the following table:*

correct n	score
0 – 6	0
7	0,33
8	4,67
9	9,00
10	13,33
11	17,67
12	22,00
13	26,33
14	30,67
15	35,00
16	39,33
17	43,67
18	48,00
19	52,33
20	56,67
21	61,00
22	65,33
23	69,67
24	74,00
25	78,33
26	82,67
27	87,00
28	91,33
29	95,67
30	100,00

Notation used in this exam: let $\mathbf{P} \subseteq \mathbb{R}^n$ always denote a pattern space of patterns given as vectors, $x \in \mathbf{P}$ patterns, f: $\mathbf{P} \to \mathbb{R}$ features, $\mathbf{f}$: $\mathbf{P} \to \mathbb{R}^k$ feature vector functions.

Answer box		**Claims** (claims marked with a ✌ are a little more challenging than the rest)
A. Elementary probability and basic math notation		
F		1. The expectation of a numerical random variable is always at least as large as its variance.
F		2. If p: $\mathbb{R} \to \mathbb{R}$ is the pdf of the distribution of a RV X, and $[a, b]$ is an interval of $\mathbb{R}$, then p cannot be exactly equal to zero everywhere in this interval.
T		3. If p: $\mathbb{R} \to \mathbb{R}$ is the pdf of the distribution of a RV X, then q: $\mathbb{R} \to \mathbb{R}$, defined by $q(x) = p(x + E[X])$, is the pdf of the centered version of X.
T		4. If X is a RV that takes values {male, female} and Y is a RV that takes values in {tall, short}, the probability $P(X = \text{male})$ can be computed from the two joint probabilities $P(X = \text{female}, Y = \text{tall})$, $P(X = \text{female}, Y = \text{short})$.
T		5. If $a \neq b$, then always $P(X = a \mid Y = c) + P(X = b \mid Y = c) \leq 1$.
F		6. $\underset{x \in [-1,1]}{\operatorname{argmax}} \cos(x) = 1$.
B. **Features and dimension reduction, PCA.**		
T		7. The grayscale value of the topmost leftmost pixel of a picture is a feature.
T		8. ✌ Consider a real-life, high-dimensional pattern space of n-dimensional patterns (like image spaces), and a training sample size N which is large (millions of examples -- as in deep learning it usually is). The training sample has been obtained by i.i.d. sampling. You pick a random pattern x from that training sample. Then, the probability that there is another pattern x' in that sample, which is very similar to x (in the metric distance sense that $\|x - x'\| < 0.1$), is so exceedingly small that it would lead to numerical underflow if represented by floating-point precision on a digital computer.
F		9. ✌ If one has two feature vector functions $\mathbf{f}_1$, $\mathbf{f}_2$: $\mathbf{P} \to \mathbb{R}^k$, then there always exists a feature transformation T: $\mathbb{R}^k \to \mathbb{R}^k$, such that for all patterns x, $T(\mathbf{f}_1(x)) = \mathbf{f}_2(x)$.
T		10. The codebook vectors c_i obtained from K-means clustering of patterns can be used to construct features defined by $f_i(x) = \|x - c_i\|$.

F		11. If $\mathbf{L}_1, \mathbf{L}_2 \subseteq \mathbb{R}^n$ are two l-dimensional manifolds, then $\mathbf{L}_1 \cup \mathbf{L}_2$ is a $2l$-dimensional manifold.
T		12. Carrying out a full PCA on the world's image data in our TICS image pattern space (which has dimension n = 1,440,000) would yield 1,440,000 principal component vectors u_i, each of dimension 1,440,000.
F		13. Given a *centered* datset $(x_i)_{i=1,\ldots,N}$, the leading principal component vector u_1 is the data mean μ.
		14. (question dismissed – was not stated precisely enough)
C. Classification problems, loss and decision functions		
F		15. A binary decision tree learnt for a classification task has exactly as many leaves as there are classes.
F		16. A loss function assigns a real number to a set $(x_i, y_i)_{i=1,\ldots,N}$ of labelled data.
F		17. ✌ Consider a feature representation with feature vectors $\mathbf{f}_i$ and linear classifiers $d(\mathbf{f}_i) = W\,\mathbf{f}_i$ where the class decision is given by the index of the maximal value in $d(\mathbf{f}_i)$. Let z_i denote the binary indicator vector of the correct class for pattern i. Claim: if $d_1(\mathbf{f}_i) = W_1\mathbf{f}_i$ achieves the minimal expected misclassification rate that is possible for such a linear classifier, and if $d_2(\mathbf{f}_i) = W_2\,\mathbf{f}_i$ achieves the minimal possible expected quadratic loss $\|d_2(\mathbf{f}_i) - z_i\|^2$, then $d_1 = d_2$.
F		18. ✌ In a two-class classification problem with 3-dimensional patterns distributed in pattern space as indicated in the figure to the right, two features f_1, f_2: $\mathbb{R}^3 \rightarrow \mathbb{R}$ are minimally needed such that (almost) perfect classification becomes possible when only those features are used as input to the decision procedure.  [image taken from lovingscience.com/category/data-science/]
T		19. A k-class classification problem with one-dimensional patterns $x \in \mathbb{R}$ can be solved by a simple linear classifier of the kind $d(x_i) = w\,x_i$ only if $k \leq 2$.
D. Bias-variance dilemma, cross-validation, and friends		
F		20. A simple linear classifier of the kind $d(\mathbf{f}_i) = W\,\mathbf{f}_i$ can never be overfitting.
T		21. In cross-validation, the validation error serves as an estimate for the risk.

T		22. The bias-variance problem becomes less of an issue if one has more training data available.
F		23. Let D and D' be two decision functions that achieve exactly zero training error. Then their risks are equal, that is $R(D) = R(D')$.
F		24. When using ridge regression to solve a linear regression task (formula below), then the optimal α found by cross-validation will move closer to zero when one has smaller training data sets. $w'_{\text{opt}} = (\frac{1}{N}\, X\, X' + \alpha^2\, I_{n\times n})^{-1}\, \frac{1}{N}\, X\, Y$
T		25. ✌ An optimal model capacity has been determined by m-fold cross-validation in two separate learning experiments. The experiments differed only in a single aspect: the choice of the number m of folds. In the first experiment, m =5 was used, and in the second, m = 10. It was observed that the validation error plots (green and red line in figure) came out differently. Claim: the green and red lines have been labelled correctly, that is, the green line indeed shows the results of 5-fold and the red line that of the 10-fold setting. 
F		26. The "variance" mentioned in the phrase "bias-variance dilemma" derives from the fact that that when a model with high capacity is used for training, its estimated parameters $\hat{\theta}$ will vary more strongly across different learning trials (using freshly sampled data each time) than when a model of low capacity is used.
E.	**MLPs and gradient descent**	
T		27. After training, an MLP with n input units and k output units represents a function from $\mathbb{R}^n$ to $\mathbb{R}^k$.
F		28. Let $\mathcal{M}_1$ be an MLP with a single input unit, a single output unit, and 100 hidden units in a single hidden layer. Let $\mathcal{M}_2$ be an MLP with a single input unit, a single output unit, and 10 hidden units in each of 10 hidden layers. Both MLPs are without bias units. Then $\mathcal{M}_1$ has more trainable parameters than $\mathcal{M}_2$.
F		29. When one uses early stopping as a method to prevent overfitting, one must split the available data into *one* training and *one* validation set; that is, one cannot use k-fold cross-validation with $k > 2$.

F		30. When applying the backpropagation algorithm for training an MLP, the MLP parameters θ will remain frozen at their initial zero values if the model is initialized with all parameters set to zero.