Name and Student ID: **Signature:**

Machine Learning BLG527E, June 2, 2018, 120mins, Final Exam SOLUTIONS

Duration: 120 minutes. *Closed books and notes. Write your answers neatly in the space provided for them. Write your name on each sheet. Good Luck!*

Q1 (30)	Q2 (20)	Q3(25)	Q4(25)	TOTAL (100)

Q1) [30pts]

You experimented with four classifiers on a dataset and obtained the following cross validation errors.

CrossVal	MLP	sqr(Xij-mj)	DecTree	sqr(Xij-mj)	KNN	sqr(Xij-mj)	DeepNN	sqr(Xij-mj)	edeepNN-ekNN
1	0.1	0.01	0.3	0	0.3	0	0.2	0	-0.1
2	0.2	0	0.3	0	0.2	0.01	0.1	0.01	-0.1
3	0.3	0.01	0.3	0	0.3	0	0.2	0	-0.1
4	0.2	0	0.2	0.01	0.3	0	0.2	0	-0.1
5	0.2	0	0.4	0.01	0.4	0.01	0.3	0.01	m=
mj	0.2		0.3		0.3		0.2		0.25
sqr(mj-m)	0.0025		0.0025		0.0025		0.0025		

Q1a) [20pts] Use ANOVA to determine if any one of the classifiers is better/worse than the others with significance of α=0.10?

L=4 SSb= 0.05 Fstat= 3.33333

K=5 SSw= 0.08 F_alpha,3,16= 2.46181

Since Fstat=3.33333 is not less than F_alpha,3,16=2.46181, we do not accept H0, that all classifier errors are equivalent. Therefore, at significance level α=0.10, **some classifier(s) are better/worse than the others.**

Q1b) [10pts] Which test would you use to determine if DeepNN is better than KNN for this problem? Give details.

We could use *5*-Fold CV Paired *t* Test.

Take the difference between errors of deepNN and kNN for each fold. (see column edeepNN-ekNN) The mean of the errors is -0.1 and the standard deviation is 0. The t-test accepts H0: errors are equal only if the t-statistics value $\frac{\sqrt{K}m}{s}$ is between $-t_{\frac{\alpha}{2},K-1}$ and $t_{\frac{\alpha}{2},K-1}$. In this case $\frac{\sqrt{K}m}{s} = -\infty$. Hence DeepNN and KNN errors are statistically significantly different and DeepNN error is less. Therefore DeepNN is better for this problem.

Hint: *Details on ANOVA:*

Null hypothesis:	$H_0 : \mu_1 = \mu_2 = \cdots = \mu_L$	
$SSw \equiv \sum_j \sum_i (X_{ij} - m_j)^2$	$SSb \equiv K \sum_j (m_j - m)^2$	$\frac{SSb/(L-1)}{SSw/(L(K-1))} \sim F_{L-1,L(K-1)}$
Accept	$H_0 : \mu_1 = \mu_2 = \cdots = \mu_L$ if $< F_{\alpha,L-1,L(K-1)}$	

F Table for α=0.10	df1=2	df1=3	df1=4	df1=5
df2=14	3.10221	2.52222	2.39469	2.30694
df2=15	3.07319	2.48979	2.36143	2.27302
df2=16	3.04811	**2.46181**	2.33274	2.24376

Name and Student ID: **Signature:**

Q2) [20pts]

You are given the following probabilities:

P(B) = 0.5, P(W) = 0.4,

P(C|W,B) = 0.8, P(C|~W,B) = 0.5, P(C|W, ~B) = 0.4, P(C|~W,~B) = 0.1.

where B:"good beans", W:"good water", C: "good coffee".

Write down the joint probability of C,W,B: **P(C,W,B)=P(B)P(W)P(C|W,B)**

Given that the coffee you drink is not good (~C), compute the probability that the water is not good quality.

P(~W|~C)=P(~W,~C)/P(~C)

P(~W,~C)

= P(~C,~W,~B)+ P(~C,~W,B)

= P(~B)P(~W)P(~C|~W, ~B) + P(B)P(~W)P(~C|~W,B)

= 0.5*0.6*0.9+0.5*0.6*0.5=0.27+0.15=0.42

P(~C) = P(~C,~W,~B)+ P(~C,~W,B)+ P(~C,W,~B)+ P(~C,W,B)

= 0.42 + P(~B)P(W)P(~C|W, ~B) + P(B)P(W)P(~C|W,B)

= 0.42 + 0.5*0.4*0.6+0.5*0.4*0.2 = 0.42+0.12+0.04=0.58

P(~W|~C)=0.42/0.58=**0.724**

Name and Student ID: **Signature:**

Q3)[25pts] Generate a decision tree using Gini index (2p(1-p)) as impurity measure.

N0 N1
1
1
1
2
2
3
3
3
2
2

Weekend (x_1)	Rain (x_2)	Daytime (x_3)	Take Taxi C
Yes	No	Morning	Yes
Yes	Yes	Morning	Yes
Yes	Yes	Morning	Yes
No	Yes	Evening	Yes
No	Yes	Evening	Yes
No	No	Noon	No
Yes	No	Noon	No
Yes	Yes	Noon	No
No	No	Evening	No
No	No	Evening	No

p1yesyes=p1yy
Weekend (x1):
p1yy=3/5 p1yn=2/5 p1ny=2/5 p1nn=3/5
12/25 *1/2 + 12/25*1/2 =12/25=24/50

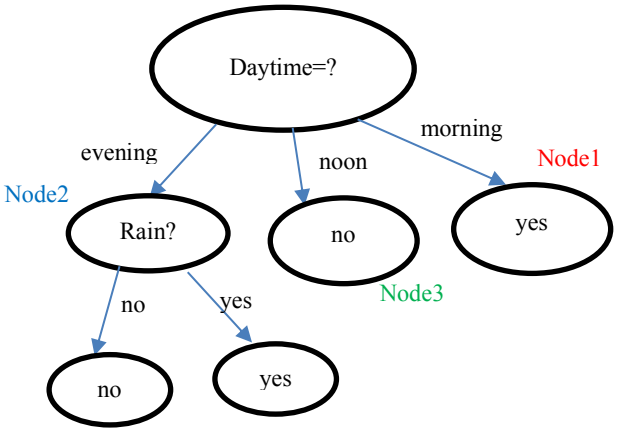
Rain(x2)
p2yy=4/5 p2yn=1/5 p2ny=1/5 p2nn=4/5
8/25*1/2 + 8/25*1/2 =8/25=16/50

Daytime(x3)
p3my=3/3 p3mn=0 p3ey=2/4 p3en=2/4 p3ny=0 p3nn=3/3
0*3/10 + 1/2*4/10+ 0*3/10=2/10=**10/50**

Minimum impurity happens when we divide acc to x3

x3=m (node1), pure, label=yes, 3 instances
x3=n (node3), pure, label=no, 3 instances

x3=e (node2)
nye: y, nye: y, nne:n, nne: n
Rain(x2)
p2yy=2/2 p2yn=0/2 p2ny=0/2 p2nn=2/2
Impurity = 0
Minimum impurity happens when we divide acc to x2



Name and Student ID: **Signature:**

Q4) [25pts]

Q4a)[7pts] Given a dataset with N=200 instances and a neural network to classify it,
-state two neural network hyperparameters that you could choose?
Number of Hidden Layers: NumHiddenLayers,
Number of Hidden Units at layer h: NumHiddenUnits[h]
-what would their initial values be? NumHiddenLayers:1, NumHiddenUnits[1]:2
I would choose a simple nn so that it doesn't overfit.
-which cross validation methods would you use to determine the optimal values of those parameters?
1 fold CV since the number of instances available are quite small.

Q4b,c,d)[6pts each] Describe and state **one** difference and **one** similarity between……

Q4b)	Describe	1 Difference	1 Similarity
Bagging	Ensemble classification method where all classifiers are trained on random instance subsets and then combined with equal weight.	Each instance has equal probability of being included in any selected subset.	Classifiers trained on instance subsets and then they are combined.
Adaboost	Ensemble classification method where the kth classifier is trained on a random subset of instances selected such that the misclassified instances selected with higher probability. When combining, classifiers with more error have less weight.	Misclassified instances are more likely to appear in the classifiers trained later.	

Q4c)	Describe	1 Difference	1 Similarity
HMM	Hidden Markov Model:	HMMs are used for modeling dependencies between observations that are dependent in time.	They both use EM (Expectation Maximization) algorithm to determine the optimal model parameters.
GMM	Gaussian Mixture Model	GMMs are used for clustering instances. The instances in GMMs are independent from each other.	

Q4d)	Describe	1 Difference	1 Similarity
Regularization	Forcing a machine learning model to become simpler so that the model does not overfit the training data and generalizes better.	Regularization forces the machine learning model to just be simpler, this could be at the expense of more training error.	They can both help with the overfitting problem.
Learning from Hints (additional information)	Forcing a machine learning model to fit not only the training data but also some additional information known about the function that generated the training data.	Learning from hints can help models learn better than plain regularization, because more model specific information, such as invariances, monotonicity etc. are taught to the machine learning model.	