CS585 Final exam

2016-12-08

Duration: 1 hour

Last Name: _____

First Name: _____

Student ID: _____

Email: _____

Question	Your score	Max score
1		2
2		2
3		2
4		6
5		4
6		4
7		3
8		3
9		2
10		2
Bonus		2
Total		32

Note - the exam is closed book/notes/web/neighbors(!), and 'open mind' :)

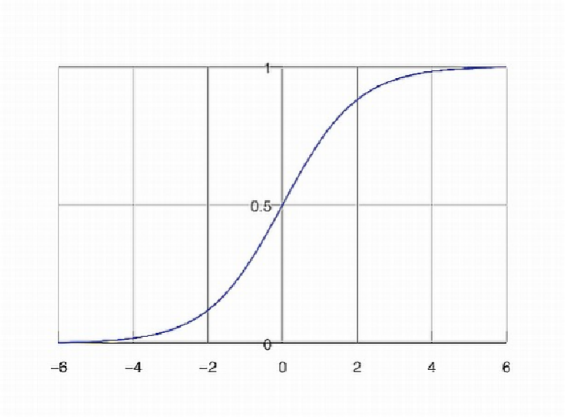
Good luck!
- Saty

Q1 (2 points). What role do minimum bounding rectangles (MBRs) play, in spatial query processing (how are they used/helpful)?

Q2 (2 points). 'Ensemble methods' are often used in machine learning – what is the single biggest benefit of using this technique?

Q3 (2 points). In WEKA, a native (custom) file format is used to read a table. Name the format, and provide a very small example.

Q4. (3+3=6 points). The 'sigmoid' function/curve shown below, is useful in at least two techniques of Machine Learning. What are the two techniques, and briefly, how is the curve used in each?



Q5 (2+2 = 4 points). Fraudulent credit card purchase detection relies on using a binary (yes/no) classifier to analyze card transactions. Name two algorithms that could be used for this purpose (we covered four), explain very briefly how each works.

Q6 (1*4=4 points). A very straightforward question - name (the) 4 types of NoSQL DBs, and provide an example (an open source or commercial implementation) of each.

Q7 (1+2 = 3 points). The MapReduce algorithm has a step between Map and Reduce - what is the step?

Briefly explain, with a diagram, how MapReduce works.

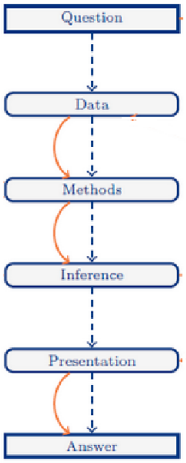
Q8 (2+1 = 3 points).

a. What are two capabilities that MapReduce v2 (YARN) provides, that v1 does not?

b. There is an architecture that can serve an alternative in some cases to MapReduce, for Big Data processing – what is it? Just naming it is adequate.

Q9 (1+1 = 2 points).

a. What does the following diagram summarize?



b. Name 4 items you would list under 'Methods' shown above.

Q10 (1+1 = 2 points). Look at the code below.

a. In what language is the code written?

b. What does it do (please be specific)? Examining the code and reading the comments should easily lead you to the answer :)

```
install.packages('neuralnet')
library("neuralnet")
#Generate 50 random numbers uniformly distributed between 0 and 100
#And store them as a dataframe
traininginput <-  as.data.frame(runif(50, min=0, max=100))
trainingoutput <- sqrt(traininginput)
#Column bind the data into one variable
trainingdata <- cbind(traininginput,trainingoutput)
colnames(trainingdata) <- c("Input","Output")
#Train the neural network
#Going to have 10 hidden layers
#Threshold is a numeric value specifying the threshold for the partial
#derivatives of the error function as stopping criteria.
net.sqrt <- neuralnet(Output~Input,trainingdata, hidden=10, threshold=0.01)
print(net.sqrt)
#Plot the neural network
plot(net.sqrt)
#Test the neural network on some training data
testdata <- as.data.frame((1:10)^2) #Generate some squared numbers
net.results <- compute(net.sqrt, testdata) #Run them through the neural network
#Lets see the results
print(net.results$net.result)
#Lets display a better version of the results
cleanoutput <- cbind(testdata,sqrt(testdata),as.data.frame(net.results$net.result))
colnames(cleanoutput) <- c("Input","Expected Output","Neural Net Output")
print(cleanoutput)
```

Bonus (2 points). Consider the four cards shown below - each has a letter on one side, and a number on the reverse. Now consider this statement: “Every card with a vowel on one side has an even number on the other side.” How many cards minimum would you need to turn over, to find out if the above statement is true or false? You need to name which card[s] you will flip, and why. Note that the answer is one of 1,2,3 or 4 :)

Question 1

Correct answer - Minimum Bounding Rectangles (MBRs) are used in the Filter and Refine step of query processing.

-1 : Partially wrong answer
-2 : Completely incorrect answer

Question 2

Correct answer - To minimize or eliminate any variances or biases between the individual learners in the ensemble
-1 : Partially wrong answer
-2 : Completely incorrect answer

Question 3
ARFF file format
Example:

```
@RELATION house

@ATTRIBUTE houseSize NUMERIC
@ATTRIBUTE lotSize NUMERIC
@ATTRIBUTE bedrooms NUMERIC
@ATTRIBUTE granite NUMERIC
@ATTRIBUTE bathroom NUMERIC
@ATTRIBUTE sellingPrice NUMERIC

@DATA
3529,9191,6,0,0,205000
3247,10061,5,1,1,224900
4032,10150,5,0,1,197900
2397,14156,4,1,0,189900
2200,9600,4,0,1,195000
3536,19994,6,1,1,325000
2983,9365,5,0,1,230000
```

ARFF- 1mark

Partially correct example- 0.5

Fully correct example -1

Description of the format - 0.25

Question 4:

Neural nets – to determine the output value of a neuron; logistic regression – to classify the incoming (unknown) data point into one of two classes, depending on the sigmoid function's value being <0.5 or >0.5.

Question 5:

- List of techniques: https://en.wikipedia.org/wiki/Binary_classification
  - Decision trees
  - Random forests
  - Bayesian networks
  - Support vector machines
  - Neural networks
  - Logistic regression
  - kNN
- Do not accept clustering techniques
- For each technique, giving name gets 1 point, giving explanation gets 1 point
- Hieu: 166 - 220
- Haoyu: 221 - 270
- Yingjun: 271 - 320
- Duc: 321- 370

Question 8(a)
YARN (Yet Another Resource Negotiator) capabilities which MR v1 does not have:

graph processing, iterative modeling
compatible with v.1.0, ie. can run MapReduce jobs
offers better scalability
better cluster utilization
create (near) real-time applications.

Marks deduction:
-1 for 1 wrong capability
-2 for 2 wrong

Question 8(b)
Bulk Synchronous Parallel (BSP) model alternative to map reduce

Marks deduction:
-1 for wrong answer

Question 9 (2points)
Expected Answer:

9a. Data Science based analysis - 1 point

9b. Data Mining, Machine Learning, Regression, Classification - 1 point

Rubrics :

9a. Must mention **Data Science based analysis/ Data Science Life Cycle** to get full 1 point

Partial points (0.5) if Data Science/ Data analysis/Data Lifecycle/ Data based analysis is mentioned.

9b. Must mention all three: **Data Mining, Machine Learning, Regression** and one other Method to get full 1 point.

Partial points(0.5) for any two methods mentioned.

Question 10
Expected Answer :
10a. R programming language - 1 point
10b. Neural Network trains on the square root of 50 random numbers and predicts the sqrt of the 10 numbers in the test data.

Or

a neural network is taught how to predict square roots

Any rephrasing of the above is acceptable. Explanation is awarded 1 point

The question clearly states **to be specific** with the answer. If you have only mentioned about neural network or the hidden layers basically rephrasing all the comments given in the question - **Only 0.5/1 marks are awarded**

Question 11

1st card and last card needs to be flipped. (A and 7)
So minimum card to be flipped to know answer is 2.
No partial marking. Student needs to mention the cards and the number of cards to be flipped

CSCI585 Final exam

2017-12-07; duration: 1 hour

Hi everyone. There are 11 questions below (10 plus a bonus), each question starting in a new page. **Please read each question carefully before answering.** There's need to elaborate on anything, so you shouldn't need extra sheets (that said, there are three blank sheets at the end).

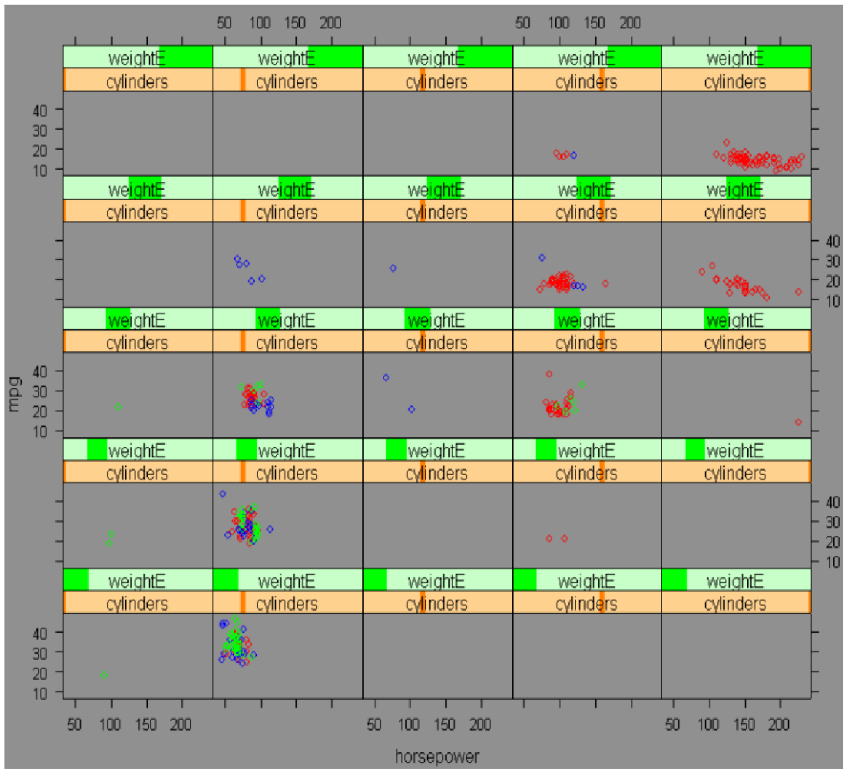
The exam is **CLOSED** book/notes/devices/neighbors(!) but 'open mind' :) If you are caught cheating in any manner, you will get a 0 on the test and also be reported to SJACS - so please don't cheat! **DO YOUR OWN WORK**.

When we announce that the time is up, you NEED to stop writing immediately, and turn in what you have; if you continue working on the exam, we will not grade it (ie. you will get a 0). So **please stick to the limit of one hour, use time wisely**!

Question	Points possible	Your score
Q1	1	
Q2	2	
Q3	4	
Q4	3	
Q5	3	
Q6	4	
Q7	4	
Q8	5	
Q9	2	
Q10	2	
BONUS	**1**	
Total	**31**	

Q1 (1 point).

Shown below is a 'trellis view' (grid view) of cars-related data. **What is a more technical term** to describe such visualization?



Trellis Display of an Auto Dataset

● American ● European ● Japanese

A. Multivariate data visualization [the word 'multivariate' does need to occur in the answer].

Q2 (1+1=2 points).

When we have numerical data (eg. home price-related), we can make predictions on a continuous scale, using linear regression (including multiple linear regression). For example, we can fit a multi linear equation for the following variables (that belong to a historical (and racially biased) dataset of home prices in Boston).

```
:Attribute Information (in order):
    - CRIM     per capita crime rate by town
    - ZN       proportion of residential land zoned for lots over 25,000 sq.ft.
    - INDUS    proportion of non-retail business acres per town
    - CHAS     Charles River dummy variable (= 1 if tract bounds river; 0 otherwise)
    - NOX      nitric oxides concentration (parts per 10 million)
    - RM       average number of rooms per dwelling
    - AGE      proportion of owner-occupied units built prior to 1940
    - DIS      weighted distances to five Boston employment centres
    - RAD      index of accessibility to radial highways
    - TAX      full-value property-tax rate per $10,000
    - PTRATIO  pupil-teacher ratio by town
    - B        1000(Bk - 0.63)^2 where Bk is the proportion of blacks by town
    - LSTAT    % lower status of the population
    - MEDV     Median value of owner-occupied homes in $1000's
```

What are two other regression-related alternatives for predicting numerical targets? Note - the alternatives can't be nearly identical to each other. Explain each, using a few sentences.

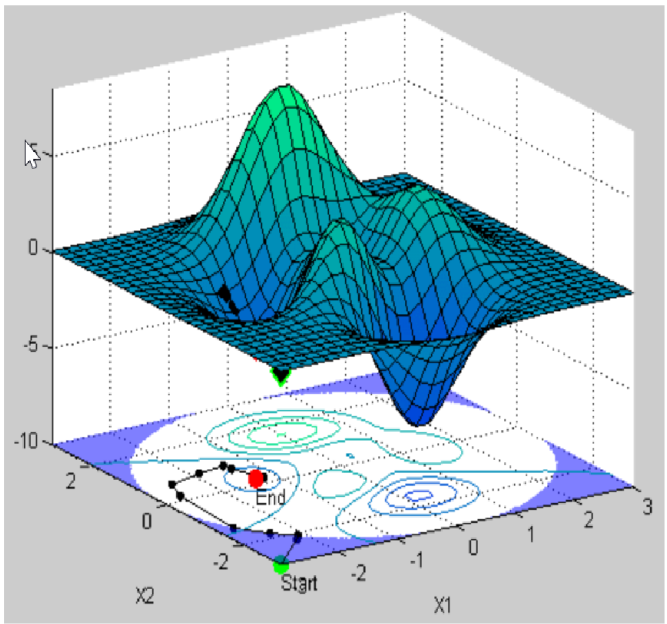
A. Non-linear (eg. polynomial, parameter-free) regression, regression tree.

Q3 (1+1+2=4 points).

a. After an 'AI winter' that lasted nearly 25 years (1985-2010), we are seeing a resurgence/explosion in machine learning, implemented using deep neural networks. **What is the technical reason why** the success rate, and flexibility (in the type of data can be learned) is astonishingly high?

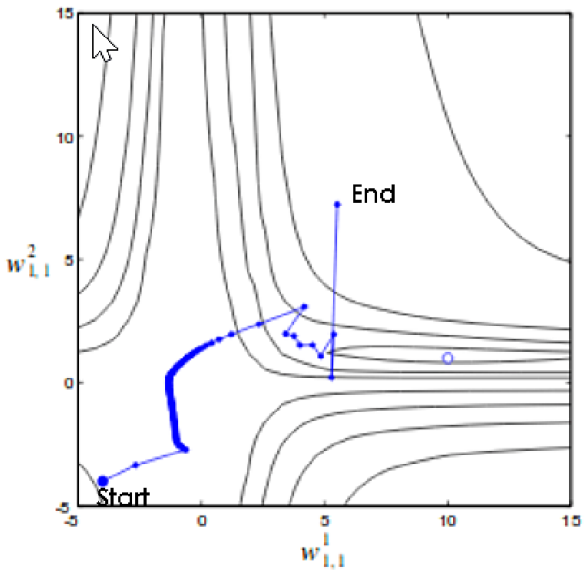
A. Layers of neurons with nonlinear (eg. sigmoid) activation functions makes it possible to approximate any smooth, continuous signal with low-enough error [they are universal function approximators]. FYI, ref: http://www.dartmouth.edu/~gvc/Cybenko_MCSS.pdf

b. For a neuron with two inputs, the error surface is shown below. Also shown below the error surface (in the 'contour plot') are the start and end values of the weights (we begin backpropagation with 'Start' weights, and stop when we get to the 'End' weights). **What do you notice**, about the training?



A. The minimum error that was reached is a local mininum, ie not the best one - there is a better (global) minimum to the right!

c. Shown below for a different neuron is its error surface's contour plot, and a training sequence of weights from 'Start' to 'End'. **What do you notice, and what is the reason you would attribute to it?**



A. The 'End' stopping point has overshot past the ideal end. This typically happens when the learning rate is set to be too high.

Q4 (3 points).

Google's TensorFlow API offers a powerful, dataflow-based approach to implementing DM/ML algorithms that process vast amounts of data (eg. realtime processing of data generated by a self-driving car). A 'tensor' is simply a multi-dimensional array datatype. Most tensorflow functions output tensors (which can be passed to other functions as inputs), some output a scalar (ie. single) value. Consider the following TensorFlow snippet: the tf.constant() calls create 1D arrays X and Y; tf.reduce_mean() finds the average (mean) of its input, and tf.reduce_sum() outputs the sum of elements. **What does the snippet calculate?** Explain, in a few sentences.

```
X = tf.constant(data[:,0], name="X")
Y = tf.constant(data[:,1], name="Y")

Xavg = tf.reduce_mean(X, name="Xavg")
Yavg = tf.reduce_mean(Y, name="Yavg")
num = (X - Xavg) * (Y - Yavg)
denom = (X - Xavg) ** 2
rednum = tf.reduce_sum(num, name="numerator")
reddenom = tf.reduce_sum(denom, name="denominator")
m = rednum / reddenom
b = Yavg - m * Xavg
```

A. The snippet does linear regression between values in arrays X and Y, fitting a line through the data. It solves for m and b, in the line equation y=mx+b, using least-squares estimates [from elementary statistics].

More FYI:

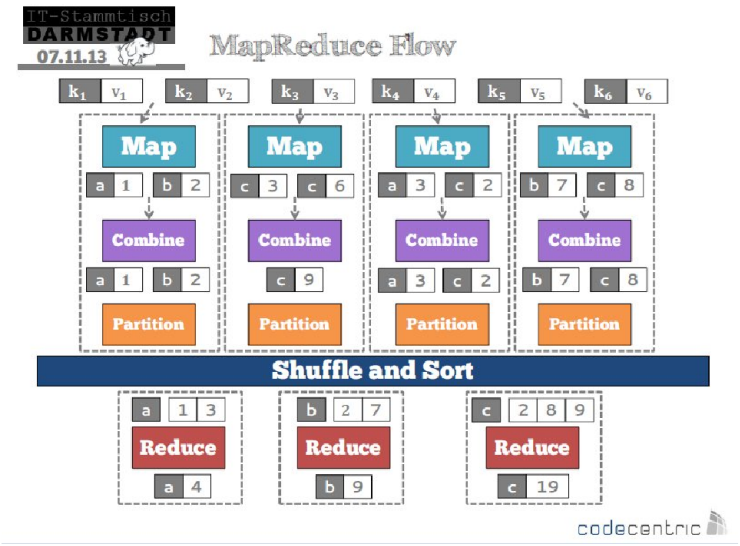
$$\hat{\beta}_1 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{\sum_{i=1}^{n}(X_i - \bar{X})^2}$$

$$\hat{\beta}_0 = \bar{Y} - \hat{\beta}_1 \bar{X}$$

Q5 (2+1=3 points).

In Map(Shuffle)Reduce, there is sometimes an optional 'Combine' step. **Illustrate and explain** this with a small example (diagram). If included, **what is its purpose** (what does it achieve)?

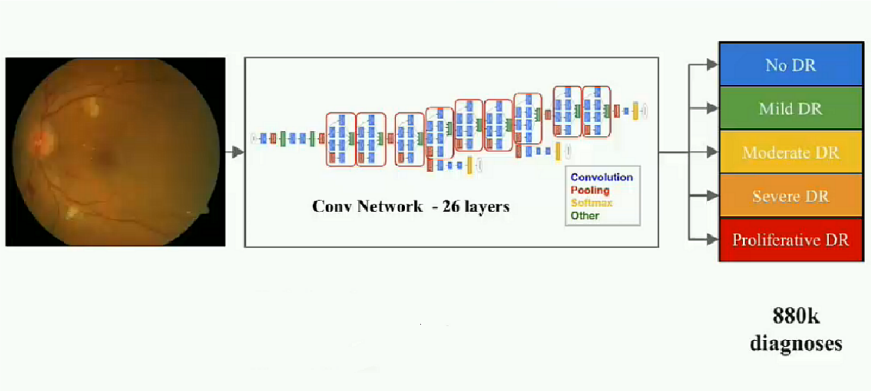
A. A combiner performs local reduction, where it collates/aggregates values of multiple identical keys output by a mapper. From the class notes:



We would do this in order to minimize the work the shuffler has to do (ie to minimize data transfer volume), and to decrease the load on the reducers.

Q6 (3+1=4 points).

Diabetic retinopathy is an eye disease (that eventually leads to blindness, if left untreated!), caused by high blood sugar levels damaging the retina's blood vessels. This can be detected via a retinal scan (which looks like the image in the left of the figure below) - an ophthalmologist makes the diagnosis from the scan. Machine learning, using a large body of existing patient data, can be used to automate such diagnosis, as summarized in the picture below (where 880,000 such diagnoses were used!). In the diagram, 'DR' stands for 'diabetic retinopathy'. **Explain, precisely, how this (automated diagnosis) would be achieved** (what is being summarized in the diagram). Also, it is likely that at first, the error (misdiagnosis) rates might be higher than those of human doctors. **Why isn't this a problem, in such cases?**



A. A deep CNN (with 26 layers, pictured above) is used to train the network, using existing patient scans and hand-labeled (derived from doctors' diagnoses) results. Of the 880K diagnoses available as 'past data', a large fraction would be used for training, after which the network is ready to classify unseen data; the rest of the prior data is treated as 'test data', to evaluate the accuracy of prediction. The network would need to be tuned (number of layers, neurons in each layer, learning rate, momentum) in order to end up with a small-enough error rate. After this, the network can be deployed, and diagnose new patients' scans into one of 5 different classes (No DR through Proliferative DR).

We aren't concerned with initial error levels because each new misclassified input (as verified by a doctor) would be used as training data (along with the prior 880K inputs) to improve classification accuracy - over time, the network will only get asymptotically better at this, never worse.

Q7 (4 points).

Many real-world data processing tasks that get parallel-processed on Hadoop/YARN require more than a single map() and reduce() step. Such 'cascades' (dataflow) of map-shuffle-reduce (M-S-R) chains can be managed efficiently in YARN, using Oozie or Mahout.

James Joyce's massive masterpiece book,'Ulysses', has 265,222 words (!) **How would you devise a cascaded M-S-R approach**, to output the distinct occurrences of words in Ulysses, sorted by their decreasing frequencies? Illustrate, using diagrams. Eg. the top 5 words might be output as:
31354 a
20045 of
18342 the
12038 and
9432 in

A.
Stage 1: mapper: output (<word>,1) k-v pairs; reducer: output (<word>,count) pairs.
Stage 2: mapper: output (count,<word>) pairs [swap key and value]; reducer: do nothing, just output (count,<word>) pairs
FYI: the shuffler (always) sorts keys, which is why the second-stage reducers above are sent sorted keys, which in our case is wordcount - so the overall output is a list of most-occurring to least-occurring words.

Q8 (10*0.5=5 points).

By definition, geospatial DBs help visualize data that have spatial extent. Given a map of the US (such as the one below), **give an example** of the type of data for each category listed below (the first one is filled out for you, as a sample answer).

a. Everyday human activity: automobile traffic during rush hour
b. Agricultural: **bushels of corn grown last year**
c. Climate/weather-related: **amount of annual rainfall for last year**
d. Consumer-related: **number of Walmart stores**
e. Environmental: **locations of 'superfund' (nuclear waste processors)**
f. Education-related: **number of universities**
g. Energy-related: **locations of coal-fired generators**
h: Financial: **median home price**
i. Health/medical: **% of obese people**
j. Public safety-related: **number of homicides last year**
k: Science/engineering/tech-related: **locations of national laboratories**

For a,b,c,d,f,h,i and j, the examples are for per-state values.

Q9 (0.5*4=2 points).

The world today is awash in massive quantities of 'Big Data', generated these days from Internet content (web pages, tweets, blogs, videos, pics...), user behavior online, sensors/instruments, etc. Prior to all this, "data" resulted from relatively limited sources and processes/collection practices. **Name 4 distinct sources/practices** that resulted in such "old school" data.

A.
Census collection
Weather data
Data warehousing
Banks: checking/savings, loans
Stock market
...

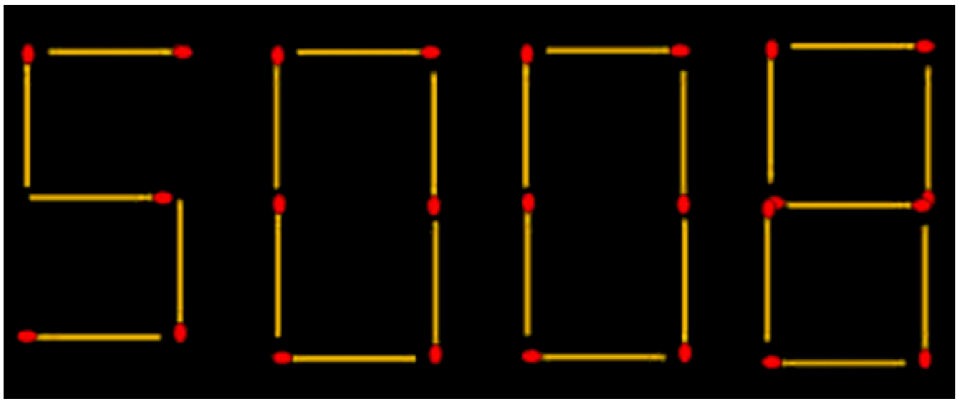
Q10 (1+1=2 points).

Recall 'polyglot persistence', when it comes to NoSQL DBs. **What two [somewhat inter-related, but distinct] reasons can you think of**, for this to be a 'bad' thing (eg. why a company wouldn't choose to encourage this in their data infrastructure)?

A. By possibly needing to 'port' code from one language to another (eg. when a decision is made to switch a part of the application from Python to R), bugs and inefficiencies could creep in; also, the lack of a standard modeling+query language such as SQL, creates lowered productivity overall (solutions can't be reused across other in-house applications, algorithms need to be coded up from scratch...).

Bonus (1 point).

Shown below is 5008, "written" using matches. By moving **exactly two matches** (no more, no fewer), what is the **largest** number you can express? Write/sketch your answer below the puzzle. **There is only one correct answer** - no point (fractional or full) for any other answer :) Read the question carefully... Be creative!



A. 11^81105 (!!!).

CSCI585 Final exam

2017-05-04

Duration: 1 hour

Last Name: _____

First Name: _____

Student ID: _____

Email: _____

Hi there! There are 9 questions below (8 plus a bonus), one question per page. Please read each question carefully before answering. There's need to elaborate on anything, so you shouldn't need extra sheets.

The exam is **CLOSED** book/notes/devices/neighbors(!) but 'open mind' :) If you are observed cheating, or later discovered to have cheated in any manner, you will get a 0 on the test and also be reported to SJACS - so please don't! **DO YOUR OWN WORK**.

When we announce that the time is up, you NEED to stop writing immediately, and turn in what you have; if you continue working on the exam, we will not grade it (ie. you will get a 0). So **please stick to the limit of one hour, use time wisely**!

Have fun, and good luck - hope you do well!

Saty

Q1 (1+1=2 points).

a. In what sense is Data Mining, an expanded version of 'statistics'?

A.
Statistics is about summarization of data: we collect and analyze numerical data in large quantities, for the purpose of inferring proportions in a whole, from those in a smaller representative sample.

With Data Mining, we don't summarize or make inferences about a larger population - we analyze all available data, and look for patterns in it.

b. How is Machine Learning related to Data Mining?

A.
Data Mining stops with the discovery of patterns in data. In Machine Learning, we 'publish' the model that we mine, and continue processing new incoming data, using our generated model.

Q2 (4 points).

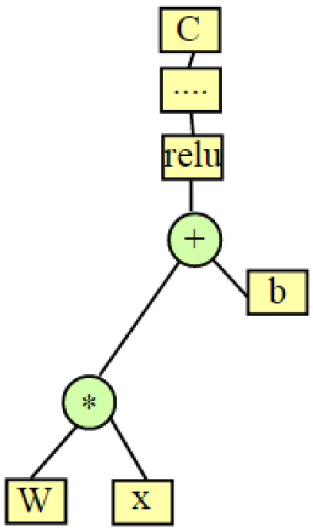
The following shows a small TensorFlow fragment:

```
import tensorflow as tf

b = tf.Variable(tf.zeros([100]))
W = tf.Variable(tf.random_uniform([784,100],-1,1))
x = tf.placeholder(name="x")
relu = tf.nn.relu(tf.matmul(W, x) + b)
C = [...]
```

Draw a graph (where the output would be C) that shows the computation above.

A.



Q3 (2+2=4 points).

'R' is at its core, a statistics programming language, which is why it has enjoyed a recent resurgence, in data mining and machine learning. What are the two most important datatypes in 'R' that are specifically meant for data processing?

A.

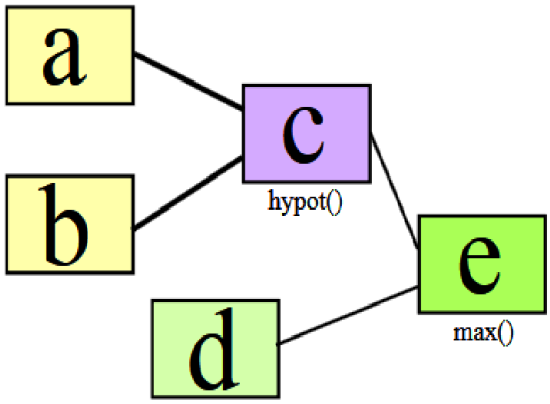
Vector - used to create an array of values that can be processed as a single entity (together, without an explicit loop).

Data Frame - used to create tabular (table-like) objects, with column names that are assigned column data as vectors. In other words, a data frame is composed of named columns, where each column is a vector type.

Q4 (4 points).

As you know, Apache Pig is a framework for expressing MapReduce calculations in a simple (compared to writing mappers and reducers in Java, Python etc) way; TensorFlow is a Python framework for expressing computation (Google uses it for DNNs). What is common to these two systems? In other words, what manner of computation do they help us carry out? Explain in 3 or 4 sentences, using diagrams.

A. Both Pig and TF help carry out dataflow computation, where data processing nodes are connected in the form of an acyclic graph, with data flowing through the nodes. The systems (Pig, TF) track the dependencies between the nodes, and schedule parallel node execution whereever possible. Here is a sample dataflow graph:



Q5 (2+1+1=4 points).

a. What is Apache Spark?
A. Spark makes Big Data real-time and interactive - it is an in-memory data processing engine (so it is FAST), specifically meant for iterative processing of data.

b. Name two Spark addons (libraries), and mention what they are used for.
A.
Spark Streaming [for handling streaming data]
Spark SQL [for executing SQL queries]
Spark MLib [for machine learning applications]
Spark GraphX [for creating graph DBs]

Q6 (1+3=4 points).

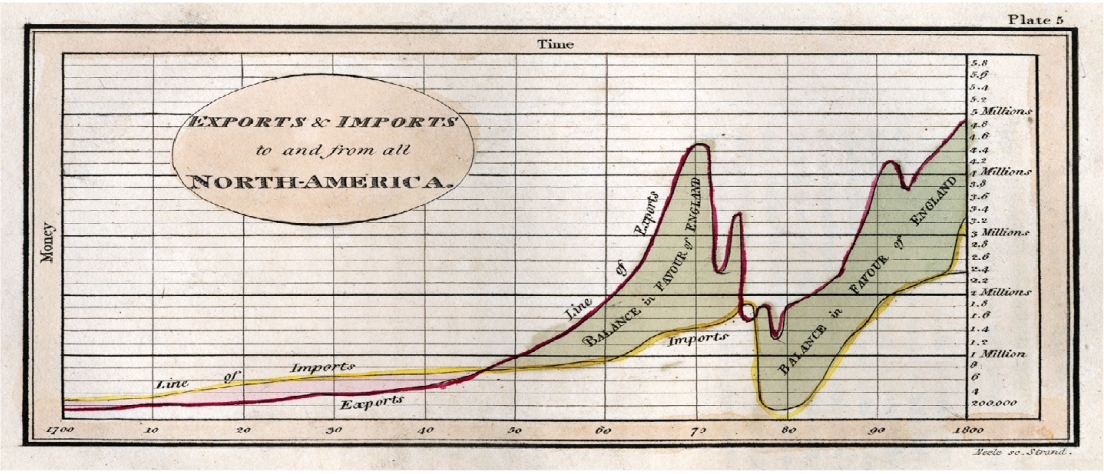
Usually in Big Data processing, we would employ horizontal fragmentation (split up a relation into groups of rows) to speed up processing. But we also use a strategy where we do vertical fragmentation (where we split columns).

a. What is this type of NoSQL database called?
A. Column family database.

b. How does this strategy help process queries faster (what is the advantage of doing this)?
A. Column families (groups of columns) that are accessed more frequently can be kept in a separate file, and non-essential columns in another file. The essential columns file can then be stored in high speed memory (eg. SSD) and accessed faster.

Q7 (2+2=4 points).

Following is an old chart that shows, from England's point of view, exports and imports to/from the US, between 1700 and 1800:



If the UK (England, Scotland, Ireland, Wales) wants to create a similar plot of export/import with the US, for 2020 to 2030, it would presumably want to collect more fine-grained data (categorized) for those 11 years.

a. What data categories (dimensions) can you think of?
A. Countries (England, Scotland..), cities within countries, months between 2020 and 2030, product categories (eg. food, clothing, toys..)..

b. Once all the data is collected, plotting all the categories in a single graph like the above would make it cluttered. So what would you do, to help viewers understand the data best?
A. Create an interactive plot, where the user can turn on/off the various categories such as countries and product categories; have a time slider that helps display data at a chosen time (eg at a specific month in a given year).

Q8 (4 points).

Here is a small table of phone numbers:

ID	Name	Number	Type
1	John	06472643	Work
1	John	01164322	Home
2	Jane	01726443	Work
2	Jane	06243344	Mobile
3	Jack	01167343	Home

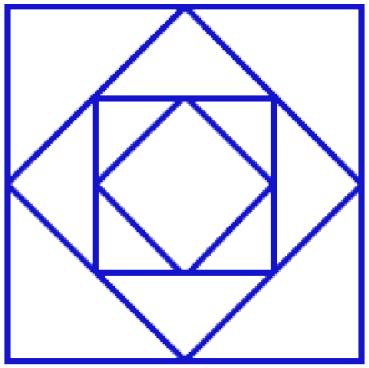
Represent the above data as (valid!) JSON, using 'id', 'name' and 'phoneNumber' as keys. You can use 'data' as the key for the entire data above.

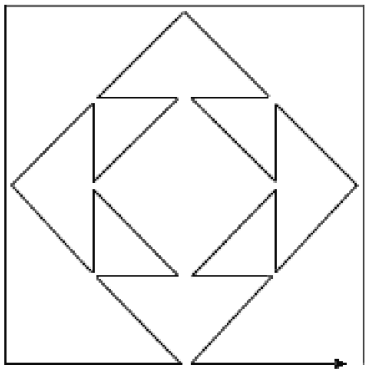
A.

```
{
  data: [
    {
      'id': 1,
      'name': 'John',
      'phoneNumber':[
          {'Work': 06472643},
          {'Home': 01164322}
       ]
    },
    ....objects for Jane and Jack
  ]
}
```

Bonus question (1 point).

How would you draw the following figure using a single, unbroken line: you can't lift the pen while drawing, and you can't draw over even a part of an existing line.





CSCI585 Final Exam

June 27th, 2017

Last Name: _____

First Name: _____

Student ID: _____

Email: _____

Signature: _____

Duration: 2 hours

CLOSED book and notes. No electronic devices.

DO YOUR OWN WORK.

If you are discovered to have cheated in any manner, you will get a 0 and be reported to SJACS.

If you continue working on the exam after time is up you will get a 0.

Problem Set	Number of Points	Your Score
Q1	2+2=**4**	
Q2	1+2=**3**	
Q3	1+1+1=**3**	
Q4	1+1+1=**3**	
Q5	1+1+1+1=**4**	
Q6	4+1=**5**	
Q7	2+1=**3**	
Total	**25**	

Q1. (4 points total) SPATIAL DATABASES

a. (2 points) Name two (or more) data structures used to create spatial indices.

Answer: R trees, R+ trees, R* trees, K-d trees, K-d-b trees, Quadtrees, Octrees.

Any one data structure will get 1 point.

b. (2 points) Google Earth uses a KML format to encode spatial data. Write down how KML encodes a geological point (longitude, latitude).

Answer: <Point><coordinates>longitude, latitude</coordinates></Point>

(The order of longitude and latitude doesn't matter.)

Each tag (<point> and <coordinates>) earns 1 point.

If you didn't write down either tag but include <placemark> tag, I will give 1 point. Other somewhat reasonable explanations would get 1 point.

Q2. (3 points total) BUSINESS INTELLIGENCE (BI)

a. (1 point) What's the key factor that impacts the effectiveness of BI?

Answer: The quality of operational data / data gathered at the operational level.

I also accept the answer "operational data". Otherwise, you cannot get the credit.

b. (2 points) Which data schema is widely used in data warehouses for BI? What are the key characteristics of this schema?

Answer: Star schema. / Snowflake (1 point)

It maps multidimensional decision support data into a relational database. / Many-to-one relationship between table and each dimension table (1 point)

0-point answer: Components of star schema: Facts, Dimension, Attributes, and Attribute hierarchy. / Only mention support multidimension facts.

Q3. (3 points total) NoSQL DATABASES

a. (1 point) What are the categories (types) of NoSQL databases?

b. (1 point) Give at least one example (name of the vendor or product) for each type of NoSQL database.

Rubric:

a) and b)
Answer:
1) Key-value store: DynamoDB (Amazon), ProjectVoldemort, Redis, Tokyo Cabinet/Tyrant, memcached, Riak
2) Column-family store: Cassandra, HBase, Google BigTable, HyperTable
3) Document store: MongoDB, CouchDB, MarkLogic, ArangoDB
4) Graph store: Neo4j, HyperGraphDB, Seasame, Graphbase, FlockDB, ArangoDB, Giraph

Rubrics:
- 0.25 per one correct type of NoSQL database.
- 0.25 per one correct example of NoSQL database.
- At least one wrong example for a type of NoSQL database --> No point for that type (0.0/0.25)

c. (1 point) Why some applications prefer NoSQL databases over SQL databases?

Answer:

- Easier to distribute [big] data.

- Performance

- Other data models that fit with specific applications rather than relational model.
- Easier to partition data and store them distributedly (Scale well).
- Performance improvement (because of partition and replication)
- Prevent the expensive "join" operation.
- Schema-free, dealing with unstructure, semi-structured data.

Rubrics:
- Got full point if mentioning all of them.
- If the answer has "big data", got 0.5 point. You need to elaborate more why SQL fails to handle big data to get full point. Similar with mentioning 3 Vs.

Q4. (3 points total) MAP-REDUCE

a. (1 point) Please re-arrange steps below into the correct flow for Map-Reduce.

A. Related key/value pairs from all mappers are forwarded to a shuffler (there are multiple shufflers); each shuffler consolidates its values into a list.
B. Mapper task is run in parallel on all the segments (ie. in each cluster, therefore on each segment); each mapper produces output in the form of (key,value) pairs.
C. Shufflers forward their keys and lists, to reducer tasks; each reducer processes its list of values and emits a single value (for its key).
D. Big data is split into segments, held in a computer cluster.

Answer: D, B, A, C

b. (1 point) Could a commodity machine be used for both map and reduce tasks?

Answer: Yes.

Rubrics:
- Got full point if simply answer "Yes".
- Yes but with a wrong explanation, deduct 0.5 point.
- 0 point with "No" answer.

c. (1 point) MapReduce involves two tasks: Map and Reduce. In each task, cluster of machines are used to perform computations in parallel. Could the Reduce task start before some machines running Map task complete?

Answer: No. Map task must be completed before Reduce task starts.

An example is counting the occurrence of words in a big text file. The steps are:
- Partition the text file into segments (smaller text files).

- Map task: each segment is processed by a node that would generate <word: num_of_occurrences> key-value pairs. Note that a word "the" may be presented in multiple segments, so the output key-value pairs may have multiples ("the", 1).
- Reduce task: aggregate the values for each word. The final result should include ("the", N) where N is the number of occurences of the word "the" in the text file.
With this example (and in general), the reduce task must wait until all machines running map task to finish. If one maching running Map task hasn't finished, the reduce task may have the counter wrong for the word "the" because the segment processed by that machine may have some "the" words in it.

Q5. (4 points total) BIG DATA / DATA SCIENCE INTRO

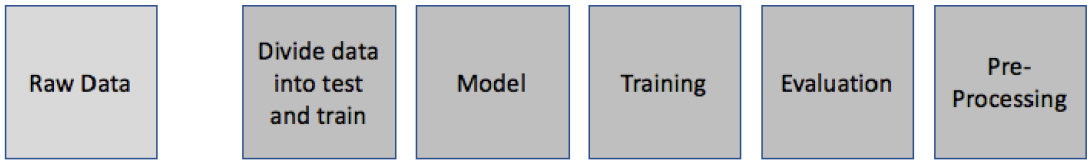
Assume you have been hired by a chocolate shop chain as a data scientist. The marketing group is suggesting that they can boost up the sales by sending the right type of sample advertisement chocolate pieces to online customers. They have already asked hundreds of customers to fill a survey which collected biographical/preference questions (raw data). Of course, the last question was "What type of chocolate do you like the most?" You are asked to come up with a machine learning approach to accomplish this task.

a. (1 point) Is this a supervised or unsupervised problem? Briefly explain.

Answer: This is supervised learning, as labels are already provided.

The term Supervised AND a correct brief explanation is required, any of these being missed means no credit, as we don't have partial credits for this question.

b. (1 point) Use the blocks below and connect them in a meaningful way to create the outline of your machine learning system.



Answer:

The diagram can be:

- Raw data > Pre processing > Divide > Training > Model > Evaluation
- Raw data > Divide > Pre processing > Training > Model > Evaluation

I also accepted this, as (somehow) theoretically it can be accepted:

- Raw data > Divide > Pre processing > Training > Evaluation > Model

Any explanation has been read and if valid has been given credit.

c. (1 point) Briefly explain the purpose of the evaluation step?

Answer:

Evaluates how system performs on test data. It is accepted if explained about a measurement of how accurate we are, even if the term “test set” was not mentioned.

d. (1 point) In another project, we are trying to classify different types of chocolate (A, B, and C). What technique could be useful for this task?

Answer:

Any of the classification techniques (or simply visualization).

They should name at least one technique
Regression (logistic/linear) is not accepted, unless it has been explained how we can deal with a multi-class classification problem using a regression model.
The term "a classification technique" is not accepted.

Q6. (5 points total) MACHINE LEARNING
a. (4 points) There are different ways to categorize the algorithms used in data mining; in class we discussed one of them with four different categories. Based on the description, write down the name of the algorithm:

[-----] involves labeling the data.

Answer: classification

- If a student answered supervised and unsupervised, instead of classification and clustering, +1 credit (instead of +2) has been given.

[-----] involves grouping data not based on pre-defined labels, but based on their similarity.

Answer: clustering

- If a student answered supervised and unsupervised, instead of classification and clustering, +1 credit (instead of +2) has been given.

[-----] in these algorithms, we try to couple the data with a continuous result values. We can use these models for predicting continuous parameters, such as amount of rain in California next week.

Answer: regression

[------] In these approaches, we try to find informative relationships within our data. For example, understanding what parameters will certainly lead to specific disease for patients.

Answer: rule extraction (association rules)

b. (1 point) What's the major problem that any of the models created by these algorithms might suffer from?

Answer: overfitting

- Any valid answer is considered correct. Low accuracy, complicated process, small number of sample inputs, etc. are not accepted as a valid answer.

Q7. (3 points total) DATA VISUALIZATION

a. (2 points) A height field is a regular array of 2D points h = f(x, y), where h is an altitude above or below a point (x, y). Height fields are often used to represent terrain data or depth underground depending on whether the 'h' are all above a point (x,y) or below it. How would you choose to visualize the height field? Describe what properties of the data it would express. HINT: Think creatively! :)

Some Possible Answers:

1. Giving different colors to heights and depths.
2. Divide the data into two maps for better visualization for ease of use.
3. The dimensions of the data, e.g. how high it is, if the height is changing over years and therefore has a time factor related to it, and whether mountains are entirely different from plains in the representation, are some of the things I expect students to write.
4. Visualization aspects of all the dimensions that the student has mentioned.

b. (1 point) Interactivity is a keyword when dealing with visualization. Mention different types of interaction that would be useful to explore the height field mentioned above.

Some Possible Answers:

1. Allow to visualize sections of the height.
2. Check and uncheck boxes for heights and depths.
3. Keep sea depths and mountain heights in different visualizations or not, tradeoffs.
4. If people can click an area and see the history of that terrain data.
5. Putting a timeline would be nice, to see how the terrain changed over time.

CS585 Final

Fall term, 12/12/18
Duration: 1 hour

Instructions/notes

- the exam is closed books/notes/devices/neighbors, and open mind :)
- there are 8 questions, and a 'non-data-related' bonus
- there are no 'trick' questions, or ones with long calculations or formulae
- **please do NOT cheat; you get a 0 if you are found to have cheated**
- **when time is up, stop your work**; you get a 0 if you continue

Q	Your score	Max possible score
1		5
2		5
3		4
4		5
5		4
6		5
7		4
8		3
Bonus		1
Total		**36**

Q1 (3+2=5 points).

a. What is the most straightforward way to transfer (ie. use in a different app, or server or device etc) the results of training a neural network on a large set of training data?

A: Use of a weights-only file, or a config file with architecture+weights - eg. the .m5 weights file is what we used in the ML homework, to transfer the training results to the classification part.

'Transfer learning' is not the right answer - the question mentions 'a different app...', not a different learning domain.

b. Name two practical applications where you might do such transferring (train, use elsewhere).

A: Self-driving car [where the weights are transfered to hardware], a smartphone app to identify birds/mushrooms/flowers/clouds....

Q2 (2+3=5 points). Machine Learning, ie. "ML", has enjoyed runaway success within the last decade, eg. in the form of Alexa, self-driving cars, etc. This is on account of the availability of big datasets, large computing power, adequate memory, and good algorithms/APIs. The modern version of ML is DL, ie. "Deep Learning".

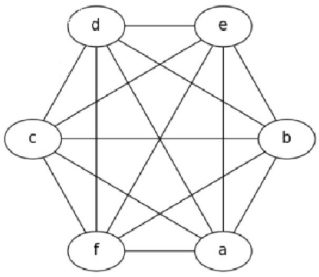
a. What makes DL "deep"?

A: The number of intermediate/'hidden' layers.

b. Even with DL, there is a serious, fundamental, show-stopper flaw in the entire approach to AI. What is it? In other words, what is ML's/DL's limitation, one that cannot be solved by faster processing, more memory, more training data, etc?

A: The limitation is that there is no genuine UNDERSTANDING of what the ML/DL is able to learn/classify! For example, the HW5 NN could tell apart cats and dogs, but it does not know that cats and dogs are the most common type of pets [doesn't even know what a pet is, etc], and, has no way of being 'told'. Also, ML/DL has no way to tell apart, correlation in data, from causation (where a part of the data ('output columns'), RESULT from factors that the input columns describe).

Q3 (4 points). Consider the following graph:



As you know, graph data can be represented via JSON, to make it be universally readable via a simple parser. Following are two representations; what is a third? You need to show your representation clearly, using valid and complete JSON like below.

```
{
 "graphData": {
  "vertices": ["a","b","c","d","e","f"],
  "edges":[
     [a,b], [b,c], [c,d], [d,e], [e,f], [a,f], [a,c], [a,d], [a,e], [b,d], [b,e], [b,f], [c,e],
     [c,f], [d,f]
   ]
 }
}
```

```
{
 "graphData": {
  "edges":[
     [a,b], [b,c], [c,d], [d,e], [e,f], [a,f], [a,c], [a,d], [a,e], [b,d], [b,e], [b,f], [c,e],
     [c,f], [d,f]
   ]
 }
}
```

A:

```
{
 "graphData": {
  "neighbors":[{"a":["f","c","d","e","b"]}, {..}, {..}, {..}, {..}, {..}]
 }
}
```

A variation of the above, also acceptable, would be the elimination of the 'neighbors' key, and simply make the value of 'graphData' be an array of objects like the one shown above.

Another more creative variation (which only works for a fully connected graph!) would be to list each loop, ie. make the value of 'graphData' be [["a","b","f"],["a","b","e"]...]. Note that there are 3-element loops, 4-element and 5-element ones, and a 6-element one.

Q4 (1+4=5 points). MapReduce is a great architecture, for executing mappers in parallel, then aggregating their outputs via a reducer step; cascading these provides enough flexibility to handle a variety of data-processing tasks.

There is another architecture [not YARN], a "MapReduce++", if you will, which extends the MapReduce paradigm.

a. What is it called?

A: Flink.

b. What are a couple of enhancements that it provides (just name them)?

A: additional transformations (beyond map(), reduce()) such as Join, Filter; additional datatypes (based on Java and Scala).

OK if the answer lists Join, Filter etc. as the 'couple' of enhancements.

Q5 (4 points). Geo-spatial data is inherently 2D, being composed of (lat,long) [or (long,lat)] pairs. What is the fundamental difference in how we set up the DB engine to query such spatial data, compared to standard querying (of non-spatial data)? Illustrate with a diagram.

A: the use of two-level processing - at the first level (filter step), MBRs are used to discard entities outside the query region; at the second stage (refine step), candidates from the first stage are queried exactly, to output the final results.

Q6 (5 points). As you know, there is a variety of algorithms used for data mining. If our data needs to be binary-classified (A or B, yes or no, low or high...), what are our choices, in other words, what algorithms will help us do this? Name/discuss briefly, 5 of them.

A [just names are here - see notes for descriptions]:

a. decision tree
b. clustering
c. regression
d. neural network
e. SVM
f. sigmoid (logistic regression)
...

Q7 (4 points). Augmented Reality (AR) is where we superpose computer graphical (CG) rendering over live (video) imagery, and modify the graphics to sync with changes in viewpoint (camera motion) - this makes it possible to 'pin' the CG renders on to arbitrary real-world surfaces.

How would you use AR, for data visualization and interaction? Be imaginative - this is an open-ended question.

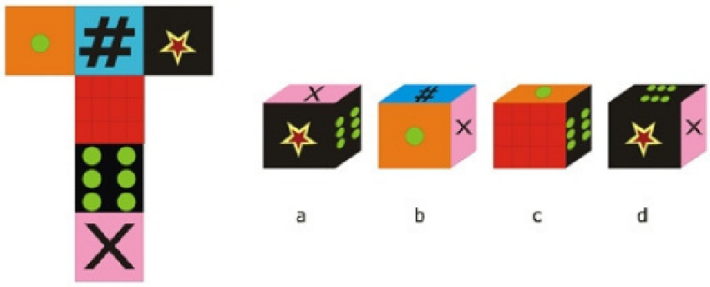
A: a flat surface on a wall, eg. a blank wall, or a blank whiteboard on it, or a poster... can be used to display 2D visualizations; or, a tabletop or coffeetable etc. can be used to display 3D viz, eg. a multi-linear (two inputs) regression plane, 3D stacked bar charts, SVM plane, 3D clusters...

Q8 (3 points). The use of JSON for representing semi-structured data provides us flexibility, compared to relational tables, when it comes to handling missing data (eg. a customer in a bank does not provide an email address while signing up for an account by walking into a bank, because "the Government will track me because of it"). What are some options for handling missing data in a JSON representation [eg. the value for an 'Email' key] of the customer's account? Name/list 3 valid ways [be imaginative] - they all don't need to be equally practical/efficient.

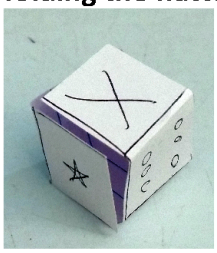
A:
a. just leave the missing key out!
b. "email":""
c. "email":"null"

The first is the best option.

Bonus (1 point). Look at the flattened cube below on the left. Which of the four shown cubes would produce the flattening?



A: 'a' [look at the photo below, that's one way to solve - create a paper cube by folding the flattened 'T' pattern:)]



CS585 Final

Spring 2018: 5/3/18
Duration: 1 hour

Instructions/notes

- the exam is closed books/notes/devices/neighbors, and open mind :)
- there are 10 questions, plus a bonus
- there are no 'trick' questions
- please do NOT cheat; you get a 0 if you are found to have cheated
- when time is up, stop your work; you get a 0 if you continue
- good luck, hope you do well!

Q	Your score	Max possible score
1		2
2		4
3		4
4		4
5		2
6		3
7		2
8		4
9		5
10		5
Bonus		1
Total		**36**

Q1 (1+1=2 points). The usual 'MapReduce' steps are mapping, shuffling and reducing. But sometimes, an extra 'combining' step is inserted.

a. where does this occur (in the sequence of steps)?

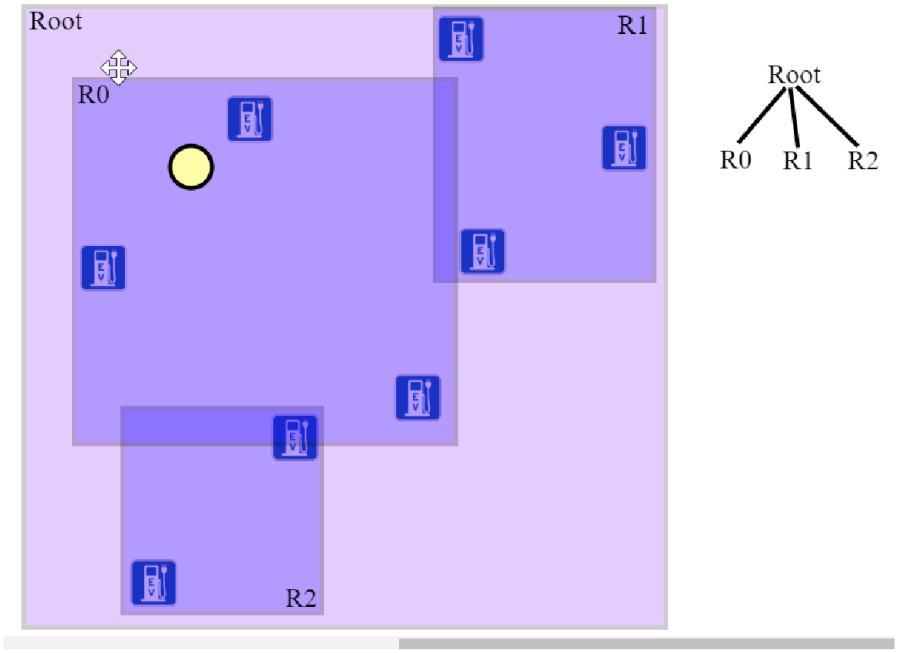
A. The combining step occurs at the end of a mapping stage.

b. what is its purpose (why include it)?

A. Combining helps aggregate values that belong to identical keys; this enhances efficiency by reducing network traffic during shuffling, since fewer keys need to get forwarded to reducers.

Q2 (2+2=4 points). You are running low on battery, driving around in an unfamiliar neighborhood - you need to recharge soon. You ask your navigation software for the nearest electric charging station, it instantly obliges - an R-tree helped, behind the scenes. Draw a simple map that shows your location, a scattering of 8 charging stations in the surrounding area, and an R-tree to contain the 8+1=9 locations; explain how the R-tree helps in quick retrieval of the result.

A. Our map and R-tree could look as follows:



Our location is indicated by the yellow circle. Starting at the root of the R-tree, we identify which child node our location is in - for us this will be R0. Then we retrieve the three charging stations within R0, compute the closest one of the three. We don't need to consider the five stations that are in R1 and R2 - that is how we speed up the lookup (by processing only 3 locations, instead of 8).

Q3 (4 points). In standard 'basket analysis', we search for rules of the type A->B, where we specify 'support' threshold for A and a confidence for the ->. In other words, what are the (A,B) pairs such that given that A occurs, B occurs as well.

In 'differential basket analysis', we can take this a big step further - we can compare the occurrence of (A,B) pairs, given other (unrelated to (A,B)) factors. This can help make better use of the (A,B) associations. Eg. does (A,B) only occur (or NOT occur) in a certain store or groups of them?

What other factors (than store locations) can you think of? You can assume (bet!) that vast amount of customer data is available. List two factors related to customers, and two factors not related to customers.

1. Time of the year

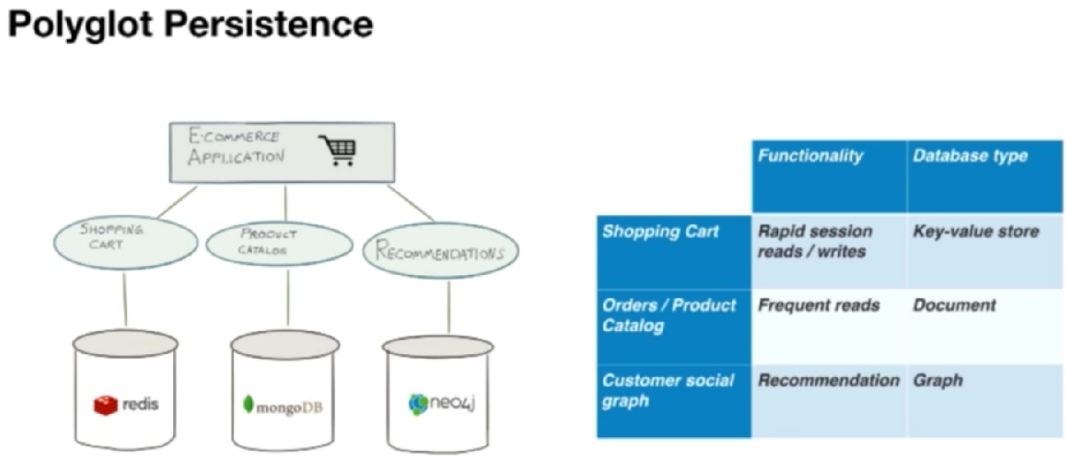
2. Time of the day

3. Customer's gender

4. Cusomer's income

Q4 (4 points). Before NoSQL, almost all databases were based on the relational model: tables (entities), and PK/FK relationships between them. The NoSQL paradigm offers us alternatives. There is one specific 'freedom' that NoSQL offers, from an architectural standpoint (ie. in the design, and redesign, of a large, complex application). What is it? Be specific, and describe it in a paragraph or two, with an illustration.

A. We have the freedom to implement a mixed-model architecture for our data storage, ie. we can have 'polyglot persistence'. With polyglot persistence, we don't need to decide on a single type of NoSQL database for the entire application; instead, based on the data to store, we can pick the appropriate type (k-v, column, document or graph).



[example from neo4j.com]

Q5 (0.5*4=2 points). A highly effective, powerful and conceptually simple way to analyze data, is to employ 'dataflow'. Name two uses of dataflow that you learned from the course (we covered three).

a. Pig Latin, for MapReduce tasks

b. TensorFlow, for neural network architectures

Data can also be processed in multiple stages (pipelining). Name two architectures we covered, that permit such multistage data analysis.

a. YARN, ie. MapReduce v2

b. BSP

Q6 (3 points). What is the most flexible way to model ('any') data? And, what structure can be used to do so?

A. A graph DB offers the most flexible way, via nodes and edges to model data and relationships.

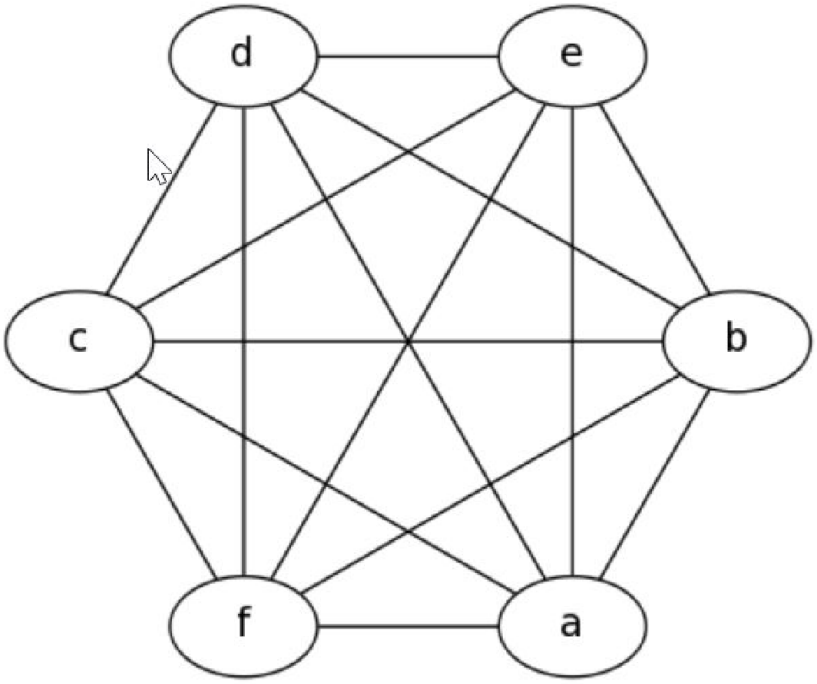
Specifically, a 'triple store' ie. (subject,predicate,object), can be used.

Q7 (2 points). Functional programming is an expressive, compact way of specifying data processing operations. You have seen two examples in the course - what are they?

a. MapReduce operations, via map(), reduce()

b. Graph processing, via Tinkerpop/Gremlin

Q8 (2+2=4 points). Represent the following graph in (valid!) JSON notation [suitable for storing in a text file, and reading it back to construct the graph] - do it two different ways!



One way:

```
{
  "graphData": {
    "vertices": ["a","b","c","d","e","f"],
    "edges":[
        [a,b], [b,c], [c,d], [d,e], [e,f], [a,f], [a,c], [a,d], [a,e], [b,d], [b,e], [b,f], [c,e],
        [c,f], [d,f]
      ]
  }
}
```

Another way (no need for the verts list!):

```
{
  “graphData”: {
    “edges”:[
        [a,b], [b,c], [c,d], [d,e], [e,f], [a,f], [a,c], [a,d], [a,e], [b,d], [b,e], [b,f], [c,e],
        [c,f], [d,f]
      ]
  }
}
```

Q9 (0.5*10=5 points). An autonomous car is deployed on the streets, having its neural networks trained (using tens of thousands of hours of hand-labeled traffic videos). 'Autonomous warfare' is a scenario that is increased being discussed by analysts, where in a (distant?!) future, self-guided soldier 'bots' would conduct war operations against 'live' enemies (who presumably don't have the means to deploy automated soldiers). Our auto-guided soldier bots could operate on land (with wheels or limbs), in the water, and be airborne as well ('killer drones').

What would the required training data be? Think broadly (including multi-sensory modes!), and name 10 different classes (detection targets, ie. things to learn to recognize) that would be useful. In other words, what can we teach our bots' neural nets?

1. Look of the enemy (eg. head dress)
2. Landmarks on the enemy terrain (eg. ridges, rivers, bridges)
3. Look of enemy ships
4. Look of enemy aircraft
5. Look of enemy fuel depots
6. Look of enemy missile launchers
7. Faces of high-value targets
8. 'Smell' (chemical signature) of explosives
9. Land features indicating IEDs and landmines
10. Look etc. of 'friendlies'

Q10 (5 points). Many events/phenomena in our lives have a spatial dependence - eg. if you live under the flight path of airplanes (eg. in Westchester, near LAX), your car would need to get cleaned quite often because of black, oily jet fuel getting deposited on it by landing planes! To prove such a causal link (landing airplanes cause oily buildups), you'd need to plot on a map: flight paths, and amount of soot (for example) per week found on cars near and away from the flight path. Cars closer to, and on the flight path, would show higher levels. Given the following list of causes and effects in no particular order, pick out 5 pairs of (cause,effect) relations that can be investigated for possible occurrence (or merely shown/documented), by plotting data on a map. Eg. the above example would be listed as (flight path, oily deposit). Note - your pairings need to be plausible, not frivolous.

Here is the list: flight path, freeways, particulate matter, power lines, noise, cancer, known gang locations, oily deposit, cellphone towers, wealthy neighborhoods, graffiti, BMW dealerships, low-income neighborhoods, crime, fast-food restaurants, vibrations (rattling), banks, pawn shops.

Among other things, such visualizations can be used to effect public policy, document social disparities, etc.

a. (freeways, crime)

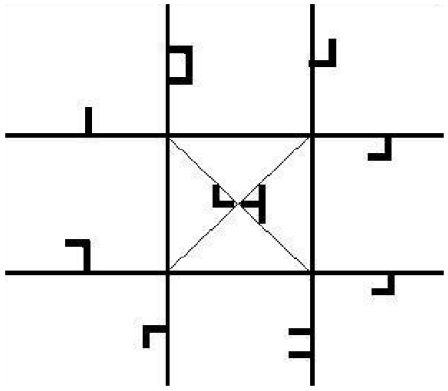
b. (low-income neighborhoods, fast-food restaurants)

c. (wealthy neighborhoods, banks)

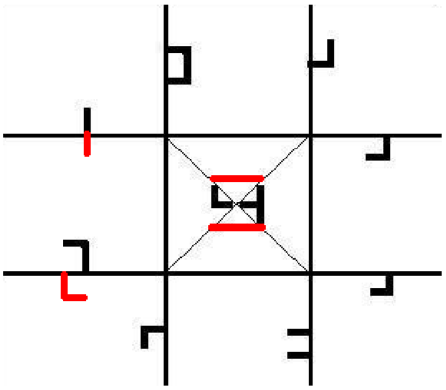
d. (wealthy neighborhoods, BMW dealerships)

e. (power lines, cancer)

Bonus (1 point). What is being indicated below?



A. Starting from the top-left and going counter-clockwise, they indicate 1,2,3,4,5,6,7,8,9 [with 9 at the center], as expressed using 7-segment LED displays oriented appropriately, with each letter's lower-half missing!



CS585 Final
Solution & Rubrics

Fall term, 12/11/19 Duration: 1 hour 15 min

Q1 (1*2+2 = 4 points). The following diagram shows how a function f(x), possibly involving data obtained from a scientific measurement setup, could be computed [using vars to store partial results]:
In the above, given x, we're computing f. A standard practice would be to code up a function, eg. 'fn', that encompasses the calculations [using local variables to store results, as shown in the figure], and then call it, eg. fn(4), fn(3.1415), etc.

a. What would be an alternate way to compute f? Why is it a better approach?

Answer: the alternate way is to set this up as a graph, ie. use **DATAFLOW.** It's better because computations can occur in **parallel** (sqrt and cos).

+1 for mentioning graph or Dataflow
+1 for mentioning valid reason

or
+0.25 for mentioning any other different (and somewhat valid) answer

b. Explain how you could further speed up the computation of f, when we have arather large array of data (eg. 20 million long) of x values to process.

Answer: to futher speed this up, the array could be split horizontally (eg. 20 splits, each with a million values), and these could be **distributed in a cluster and processed in parallel** (using MapReduce for example).

+2 for mentioning either distribution or parallel or MapReduce
-1 if the answer seems partially correct

Q2 (3*1 = 3 points).

Historically, data viz has been carried out on print media (newspapers, books, magazines, journals) - these provide zero interaction, allowing for just passive consumption (and possibly leading to attendant lack of interest). Today, what else we have are interactivity, and animation (more engaging). What three other modes of data presentation can lead even more engagement and utility?

Answer: we can use **VR**, to be visually surrounded by the data (which can even be superposed over 3D scenes that led to the data creation); we can use **AR**, to have the data be visualized over existing real-world surfaces (eg. tabletops, walls), collaboratively (eg by analysts sitting around a coffee table). We can create **holograms** out of the data (eg. to show polar ice caps melting). Or we could **3D-print** the data (including in color). Or we can use **projection mapping** to project data on to surfaces.

+1 for each valid data presentation mode (not necessarily those mentioned above), totally 3

Extra FYI: https://www.intechopen.com/books/holographic-materials-and-optical-systems/3d-capture-and-3d-contents-generation-for-holographic-imaging

Q3 (3+3=6 points).

Consider a key-value (k-v), in-memory store architecture below (shown at the center of the diagram):

a. What are 3 typical reasons why you would set up such a data store for clients?

Answer: **faster access, less load on the backend, higher throughput (more clients can be served).**

+1 for any valid reason (not necessarily those mentioned above), totally 3 reasons

b. Typically, such a setup as shown above, would reside in your own IT infrastructure (connected to your organization's web server). **What 3 additional advantages would you get, by switching to a cloud-based service that offers a clustered version of the above (in-memory DB instances running in multiple nodes that are connected together)?**

Answer: **unlimited horizontal scaleout**, even **higher throughput** (could even use clients' location to determine nearest nodes to serve data), **no maintenance** (from our, ie data holder's, perspective).

+1 for any valid advantage (not necessarily those mentioned above), totally 3 advantages

Q4 (4 points). Association mining (eg. using the A-priori algorithm we studied), applied to 'shopping baskets' (large groups of transactions) produces rules of the type A->B, given 'support' threshold for A and a confidence for the association - in other words, it outputs items purchased together.

A refinement of the above, can produce even more specific associations - we can consider the occurrence of (A,B) pairs, in the broader context of other (unrelated to (A,B)) factors that might nevertheless influence A->B. This can help us make better use of the (A,B) associations.

Eg. does (A,B) only occur (or NOT occur) in a certain store or groups of them? Name 4 such additional factors that can be used to analyze our data (mined associations).

Answer:

1. **Time of the day**
2. **Time of the year (seasonal variation)**
3. **Ethnicity of customers (yes, such data *is* available)**
4. **Customer's annual income** 5 ... [eg. customer's age]

FYI: look up differential market basket analysis ["differential MBA" :)]

+1 for each valid factors unrelated to A->B (Can be different from the ones above)

There should be 4 points atleast

Q5 (2+4 = 6 points). Consider the following diagram (from an existing source), related to nearest neighbor (NN) queries (like from your HW3 on spatial data):
The R-tree shown is in three levels, with the leaves (a..i) being items of interest, ie. what we hope to get from our NN search. There are several different algorithms for traversal, which might result in different items (any of a through i) being returned by the query. **a. Assuming we do a DEPTH-FIRST search, what (single) item will be returned by a closest-point query?**

Answer: **h.**

+2 for correct answer

b. What paths would you consider, to arrive at your answer above (returned item)? Note - the numbers shown in the R-tree, indicate the closest-distances from the query point, to the bounding boxes, and to the actual leaf items [which, for simplicity, are located ON the bounding box edges and corners, but this doesn't affect the results].

Answer. We consider E1 -> E4 -> (a,b,c) - of these, we'd pick 'a' [smallest of a..c]. Next we'd consider E1->E5 (because E5 is also sqrt(5), like E1) -> (d,e,f), but reject all the leaves because they are bigger than sqrt(5). **We skip going down E1->E6**.

Skip E2->E7. Do E2->E8->(h,g,i), then pick 'h' as the better value than 'a' (sqrt(2) < sqrt(5)). **We'd skip going down E2->E9**.

E1 -> E4 -> (a,b,c) -> pick 'a' (sqrt(5)) (+1)

E1 -> E5 -> (d,e,f) -> reject all leaves (+0.5)

E1 -> E6 -> reject, sqrt(9) > sqrt(5) (+0.5)

E2 -> E7 -> reject, sqrt(13) > sqrt(5) (+0.5)

E2 -> E8 -> (h,g,i) -> pick 'h' (sqrt(2)) (+1)

E2 -> E9 -> reject, sqrt(17) > sqrt(2) (+0.5)

If only the paths to a and h is written but reason is given why rest of the nodes not considered, +4 is awarded.

If only the paths to a and h is written, +2 is awarded.

Q6 (5+1 =6 points).

a. In the context of neural networks (which are a way to carry out supervised learning, using pre-labeled data), explain (using just two or three sentences), the following terms.

i. Weights

Weights determine the strength of the connection of the neurons. It shows how a specific input attribute is linked to the output.

Explaining with example or with definition to be given +1 points.

ii. Backprop

For a deep network, given an error function backprop calculates the gradient of error function wrt neural network's weight. Since weights are randomly assigned for the neural network at the beginning, it is through back prop that the set of weights that generalise the data well are found.
+1 for explanation

iii. Loss

It is the quantitative measure of deviation or difference between the model's predicted output and the actual ground truth.

+1 for specifying the difference between actual output and predicted output.

iv. Architecture

It refers to the arrangement of neurons into layers and the connection pattern between layers, activation functions and the loss functions. It determines how the neural network transforms the input to output.

+1 for mentioning layers of neurons.

v. pre-trained model

It is a model that was trained on a large benchmark dataset to solve a problem similar to the one that we want to solve.

+1 for explaining the purpose . 0.5 for not specifying the purpose of using pretrained models.

An answer that indirectly mentions all the above (using rather incorrect language) gets partial credit.

Answer: these are all straight from the lecture and discussion. You don't need to use wording from the slides**, your own descriptions are fine as long as they are correct.**

b. Even with a large training dataset, it seems rather easy to 'fool' a standard *convolutional neural network* (CNN) into misclassification [or in some cases, no need to explicitly attempt to fool it - it simply seems incapable of correct classification (eg. might classify an upright scooter as a parachute)]. **What is the underlying cause of such drastic failures?**

Answer: the neural net has **no additional data** beyond training data such as images, audio [knowledge of features of objects, or hierarchies (assemblies), groupings, context (eg. what objects are found where, and why), etc] - **they only learn from pixels/audio/text...** that have been previously labeled (by humans) and input to them.

+1 for specifying that neural network relies solely on training data and output to carry out the classification tasks.

Q7 (1+2 = 3 points). Machine learning ('ML'), especially deep learning (which uses massive amounts of training data that passes through deep layers of neurons), is a "revolution" that has rapidly (starting in 2012) taken over every industry and field.

But, for all its successes, there are many glaring issues, one of which is 'bias' - this has resulted in sentences unjustly imposed, medical insurance unreasonably denied, people misidentified as criminals, etc.

a. What is the source of bias, in ML?

Answer: **simply, the chief source of bias is in the dataset.** FYI - an additional source of bias could be in the NN algorithm, ie in the calculation of loss.

1. **1 point if answer talks about data**
2. **-0.5 if the answer only talks about bias error (What is bias in model? High bias, low bias) and not about bias in data**
3. **-1 If answer is neither about data nor bias error**

b. How would we fix the issue?

Answer: by analyzing ("auditing") the data for **fairness** (data cannot be incorrect/inaccurate) and **completeness/balance** (eg. cannot predominantly contain data about select labels).

+ 2 Any two solutions can be accepted if it used to cure the bias problem(balancing labels, correctness of estimates, distributed random initialization for missing values)

1. **-1 If only one solution provided with insufficient explanation**
2. **-0.5 if only one solution provided with example and explanation**
3. **-1 If answers are not about data, but about bias error and rectification with parameters in training**

Q8 (1*3 = 3 points). Here's a blue-sky ("be imaginative!) question. As voluminous as the data we process today seems, the future is sure to involve even more of it (eg. via higher resolution scientific instruments, more sensor-generated 'IoT' data, etc.). **Where do you see the following headed (in other words, what's the trend, what's coming up (even if it is in research or prototype stages)?** Think BROADLY, in terms of new technologies (including phenomena, materials, devices, designs...)! In other words, how do YOU plan to deal with these?

a. storage (to hold data)

Answer: **DNA, holographic storage, alternate materials.**

+1 for mentioning correct answer (even just one item is ok)

b. processing (computing)

Answer: **quantum computing, DNA computing, optical computing.**

+1 for mentioning correct answer (even just one item is ok)

c. infrastructure (how the above two are accessed and utilized)

Answer: **edge computing** (at or near the source of the data), **custom SoCs** (eg. intelligent cameras that output labels+bounding boxes in addition to raw video), **custom 'AI' processors, newer chip architectures** (eg. NN in hardware)...

+1 for mentioning correct answer (even just one item is ok)

Bonus (1 point). This bonus point will count, ie. be added to your total for Q1-Q8, if the total is < 35 (if you already have a 35, it will be skipped). In other words, the max you can get for the entire exam is 35, not 36.

The figure below, represents something specific - what is it? The answer is not open ended (eg. you can't say 'an abstract stained-glass window pattern'!). A very big hint it's something we COMMONLY use!
Answer: **it shows A-Z (uppercase!). It also shows 0..9.**

+1 if the answer is correct (one of the above mentioned answers)

CS585 Final

Spring term, 2019-05-02
Duration: 1 hour

Instructions/notes
the exam is **closed** books/notes/devices/neighbors, and **open** mind :)

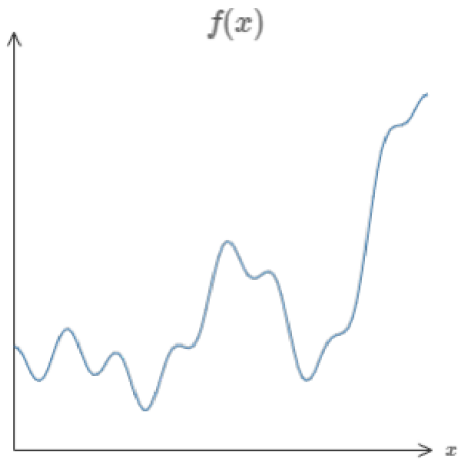
- there are 10 questions, together they touch upon every lecture topic [look for the word 'data' in each!], worth a total of 35 points
- the questions will make you think [not tax your memory]
- there are no 'trick' questions, or ones with long calculations or formulae
- there is a single blank 'scratch') page at the end, you can use it if you like
- **please do NOT cheat; you get a 0 if you are found to have cheated**
- **when time is up, stop your work**; you get a 0 if you continue

Good luck, have fun, hope you do well.

Q1 (3 points). 'Big Data Mining' involves making sense of ever-growing volumes of **data**. Also growing is the proportion of non-technical users (managers, domain experts, laypeople...) who want to analyze/mine all this Big Data. How would you, as a data expert (DBA + data scientist + data engineer...) ENABLE this, ie. what would you build (or buy) for their use? Note that you'd be placing yourself out of a job when you do this... :) Please be rather specific/technical in your description, instead of being vague.

A. The non-technical user would be best served by a 'data science platform' (appliance/turnkey solution), where they simply need to feed it input (eg. connect to a database, or streaming data source, etc), and do analysis via point-and-click, including creating dashboards for results. The idea is to not require an expert be available always, needing them only for occasional tuning, upgrading, training, etc. Examples of such platforms include Qubole, Splunk, Azure Databricks, Netezza, Dataiku.

Q2 (4 points). Given any function (that represents collected, labeled **data**, in any number of dimensions), including the 'wiggly' 2D one shown below, a neural network (NN) can always be constructed to approximate it (ie. to 'learn' the pattern in the labeled data). In this sense, an NN is a 'universal function approximator', which is the reason for its runaway success.



What, in math, is this (function approximation) roughly analogous to? Describe, using diagrams or simple equations.

A. This is loosely analagous to Fourier synthesis/summation/decomposition, where a periodic signal can be reconstructed using sines and cosines, eg:

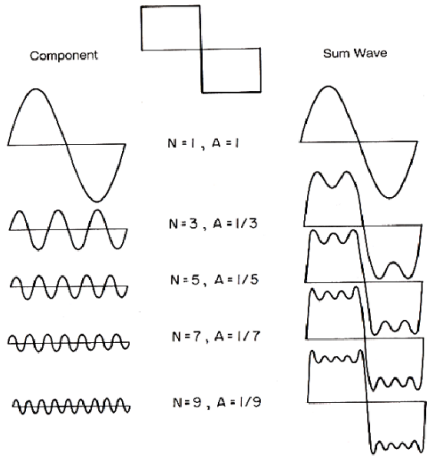
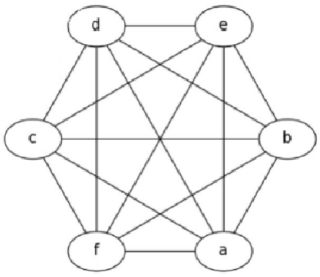


Figure 4–5 Fourier synthesis of a square wave. At the left are the successive harmonics; at the right are the sum waves including each successive harmonic. The graph at the top is the wave being synthesized.

From Berg and Stork

Q3 (2+2=4 points). Consider the following graph:



a. Represent the above, using a non-JSON, non-XML **data** format.

A.

Here is one way (verts, followed by edges):

a
b
c
d
e
f
a-b
b-e
e-d
d-c
c-f
f-a
a-e
e-c
c-a
f-d
d-b
b-f
a-d
e-f
b-c

b. What is the problem with using such formats (that aren't JSON or XML) for data description?

A. Such formats are difficult to parse, or at the least, would need a custom parser [prone to error, might lack features, might need resources to be developed, might need to be maintained ongoing...]. Further, adding new fields (eg. edge weights and labels, in the above example) would necessitate rewriting the parser, and might make the older data files unusable.

Q4 (1*3=3 points). 'BI' of yesteryear, looked a lot like this (PowerPoint slides, of a company's yearly sales **data**, presented at an annual meeting, for example):

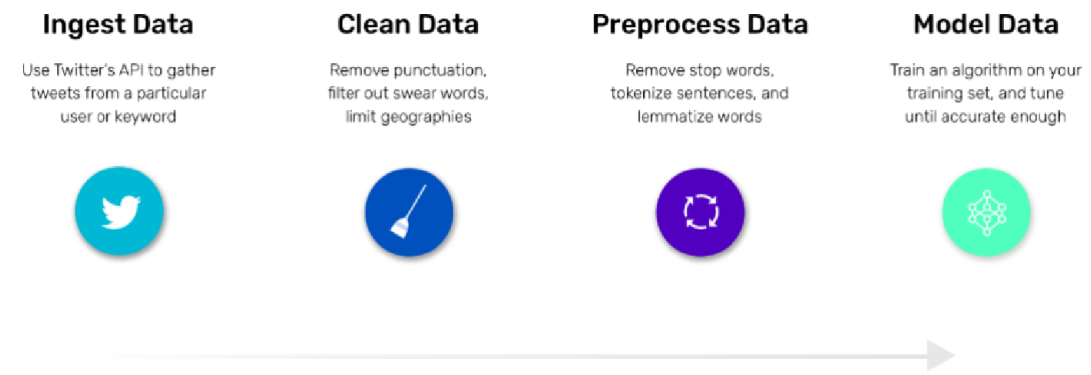
Todays BI is a quantum leap compared to the above - name/describe briefly, three ways in which it is much better today.

A.

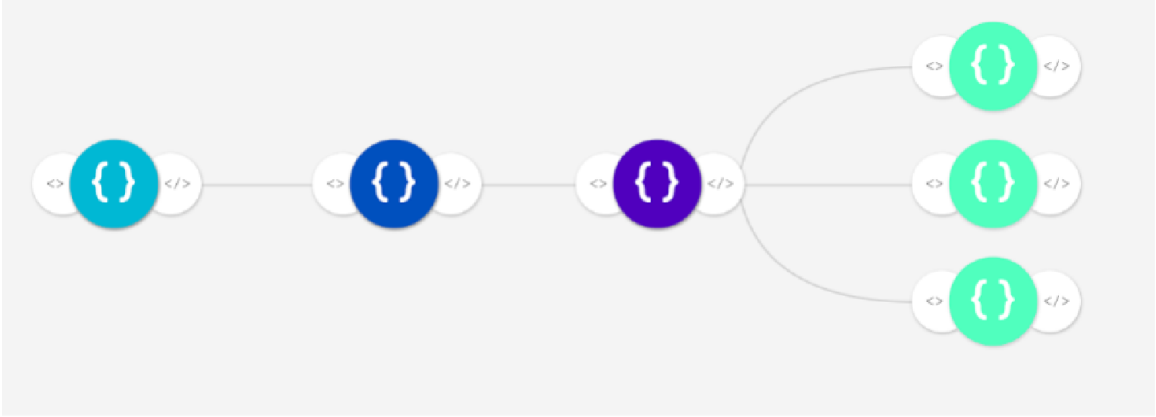
1. Interactivity, animation
2. Real-time (or near-real-time) processing, ie continuous analytics
3. Cloud processing
4. Mobile dashboards

...

Q5 (2+2=4 points). Here is a standard way to analyze Twitter feed **data** (eg. you could create it via a Jupyter notebook, using Python calls to an API or APIs):



Here is a better way:



In the 'better' way above, the processing pipeline is expressed as a graph, where each {} represents a node, inside which the functionality (ingesting data, cleaning...) could be expressed via a function or microservice call. What is such an analysis scheme called, and why is it a better alternative?

A. This would be 'dataflow', which is better because:
* an individual step [ie a node's contents/calls] could be swapped out, possibly to improve performance, reduce operating costs...
* changes made to one node will only require recomputing downstream nodes (as opposed to having to re-execute the entire pipeline)
* nodes could be run in parallel where possible
* nodes could be distributed across devices, cloud...

Mentioning two of the four items above, would be a sufficient answer.

Q6 (1*3=3 points). In a small company, a classic **data** science usage scenario would look like this: a team of data scientists comes up with, or selects, a data mining algorithm for use, working with domain experts to learn about the data to analyze; a team of data engineers would then build a pipeline around this algorithm (including coding the algorithm from scratch, or using API calls for it from a library), test, and deploy it on the company's servers, along with the data, for mining/learning and ongoing usage.

What are three current/emerging trends that are alternatives to the above?

A.

1. Using the cloud.
2. Using VMs or containers [VirtualBox, Docker, Kubernetes...]
3. Using a platform.

...

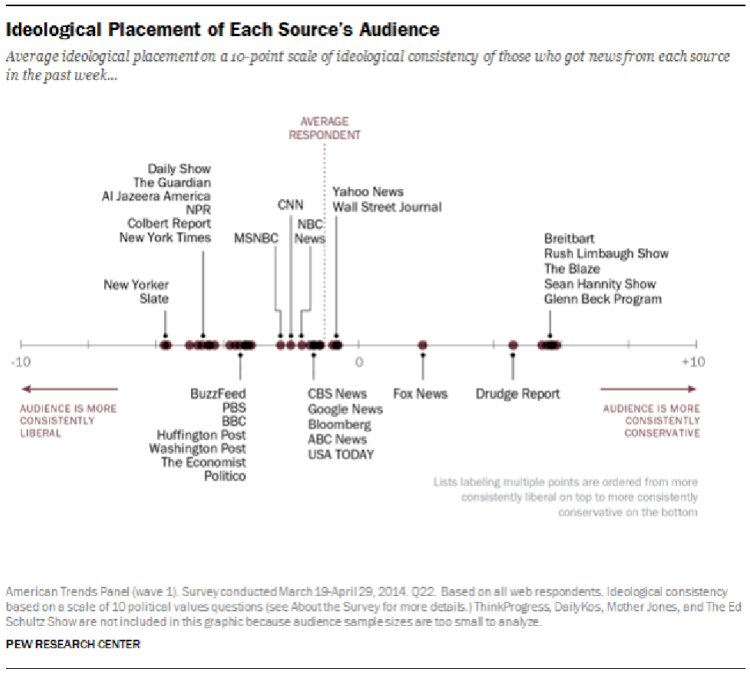
Q7 (1+3=4 points). As you know, very large graph datasets (eg. from social media, web page linking, etc) are commonplace. 'Connected components' (clusters of inter-connected vertices) is an algorithm that is run on such graph **data** [eg. to divide a set of web pages into groups (clusters), where in a group, pages link to each other, but not to any page outside the group]. MapReduce can be used for this computation, after partitioning the data into horizontal fragments.

However, use of MapReduce would require looping through several map-reduce iterations, to grow the components/clusters incrementally. What is a better alternative, and why?

A. BSP is a better algorithm for this, because nodes can process data (expand a vertex's neighborhood incrementally) till they need to exchange data with other nodes (eg. pairs of nodes that contain an edge that is split up because of partitioning) in order to complete the partitioning.

Note that 'YARN' is not the best answer, because it is simply an augmentation of the original (v1) MapReduce algorithm [MapReduce is already mentioned in the question].

Q8 (1.5+1.5=3 points). The following 'political **data**' visualization shows where, in the left/right political "spectrum", various audiences that consume content from news outlets lie (we went over this in class) - left is liberal, right is conservative:



What are a couple of ways by which you could make the above depiction more informative (ie. how would you improve it), if you could gather more data?

A.

1. We could create a 'bubble' chart, with readership/viewership info - bigger bubbles mean a bigger audience.

2. We could indicate, using a bar chart for example, how many news items each outlet publishes, each day/month/year...

...

Q9 (2+2=4 points). People increasingly have 'always on' microphones in their homes, in the form of Alexa/Echo, Google Assistant, etc. "Alexa" at times responds when "she" is not spoken to (when this happens, your speech not intended for Alexa does get transmitted to Amazon's servers). Also, it has been reported that a ('small') team of Amazon ML engineers are authorized to listen in on anonymized conversations at random, with the goal of making interactions better – but, they do have access to at-home devices' location coordinates (which means they might be to figure out an 'anonymous' user's location).

What can go wrong, from a privacy and security POV, when (a) voice **data** not meant for Alexa gets sent to Amazon's servers, and (b) a malicious employee might be able to locate a user who is supposed to have been anonymous? Be as specific as possible.

A. If Alexa (eg. an Echo device at home) transmits a user's speech incorrectly (ie. data not meant to be consumed by Alexa) to Amazon's servers, that could lead to the user be profiled (info can be learned about them), and consequently blackmailed; a government subpoena of the user's Alexa queries would lead to all this extra data be sent to the government, with all sorts of repurcussions [extra surveillance, imprisonment etc, depending on the content of those extra conversations]. If a malicious employee is able to locate a user's address, that employee can publish it online, sell the info, notify local police about supposedly 'bad' intent on the user's part, etc. The bottom line is that a typical user is unaware that their private thoughts and location might be available to people whom they will never know/meet, and has no idea how, or when, or for what purpose, such info might be misused.

Q10 (3 points). Unlike the dominating AI algorithms today (esp. neural networks, including DL, CNN etc), 'AGI' (artificial general intelligence, ie. "next gen" AI) is not expected to be data-hungry at all. Why would an AGI system not need to rely heavily on training **data** (ie. how could it be constructed)? You can explain in your own words, based on what was discussed in class, or based on your own ideas.

A. An AGI system would be expected to learn **concepts/features** related to the world (everything from physical phenomena (eg. rain), bodily and mental feelings, human behavior, etc) by experiencing the world directly, and reason/act using them. Such concepts become categorized, and generalized, and interlinked with other concepts, forming a giant knowledge graph of sorts. For all this, we don't need large amounts of training data [in the real world, humans and animals, don't]. In contrast, a standard DL network, lacking any features, solely relies on input, to learn to categorize - that is the reason it needs excessive amounts of data.

CSCI 585, Final Exam, Fall '20

Please read the following carefully, before starting the test.

1. The exam is open books/notes/devices - feel free to look up **whatever** you want!

2. Not counting 'Q0', there are 10 questions, each containing the 'd' word (**data**), numbered Q1..Q10, worth 5 points each. You can pick any 7, for a total of 35. ADDITIONALLY (if you want to, have time) you can answer one, two, or three more - **this means there are 15 bonus points!** We will cap your score at 35, if it exceeds that.

3. There are no 'trick' questions, or ones with long calculations or formulae, nothing that requires you to needlessly write a lot. There aren't questions whose answers are a Google search away [or available directly from the lecture notes] either! The questions do make you think, imagine, and, apply what you learned.

4. **Please do NOT cheat** - this means NOT communicating with anyone via any device/medium/channel - you will get a 0, and be reported to SJACS, if you are found to have cheated; ANY attempt to get help from others in any form is a VIOLATION, as per https://policy.usc.edu/scampus-part-b/ (https://policy.usc.edu/scampus-part-b/), sections 11.11 through 11.14 [read it, if you are not familiar with it].

5. When the time is up (90 minutes), stop your work, then spend the rest of time (30 minutes) on submission.

6. **Good luck**! Hope you do well, and enjoy coming up with the answers. Try to stay calm, take a deep breath, start!

Q0 [0 points]. You **MUST** turn this in - DO NOT omit doing so - there is a **penalty of 2 points** if you omit this.

Please write the following line, and sign it - it is your acknowledgment of having read USC's policies on academic misconduct (https://policy.usc.edu/scampus-part-b/ (https://policy.usc.edu/scampus-part-b/), 11.11-11.14) and agreement to honor them: **I have read USC's standards on academic integrity, and agree to abide by them.**

Q1. The ML revolution, as you know, is fueled by **DATA.**

Q. [1+1= 2 points] The 'supervised' in 'supervised ML' refers to the use of labeled data. Where does the labeling come from? And, how might we automate a part of the labeling task?

A. The labeling is manually done [via software] - by humans. Semi-supervised learning helps automate a part of it, by labeling only a part of the data manually, using that to train a network, then using that network to automatically label the rest.

Q. [1+1+1= 3 points] What features/columns/measurements would you need (ie. data you'd collect), in order to train an ML, on the following:

a. helping someone exercise better (eg. Kemtai (https://app.kemtai.com/setup))

A. Head posture, body posture, movements that are specific to the exercise that would be taught by the AI...

b. helping someone speak better on stage (eg. imagine a person giving a TED talk (https://www.ted.com/talks/david_mccandless_the_beauty_of_data_visualization?language=en) for 20 minutes - people have a variety of things they can be coached on, to improve themselves)

A. Gaze direction (where the speaker is looking), speech mannerisms, hand gestures, pacing the stage...

c. an AI replacement for a news reader (https://says.com/my/tech/south-korea-mbn-now-has-an-ai-news-anchor-kim-ju-ha-capable-of-working-24-7) :)

A. Pronounciation, emotional tone, facial expressions, body posture...

Q2. [1*5 = 5 points] Provide an example (using words) for each type of **data** viz indicated below - the example should not be from the viz (https://bytes.usc.edu/cs585/f20_db0DS/lectures/Viz/slides.html) lecture slides. In other words, what data would you visualize?

a. bubble plot

A. Popularity of the various social media platforms.

b. choropleth

A. Number of hours per day spent online, in every state/county in the US.

c. donut chart

A. Relatively popularity of the various types of game consoles (Switch, PlayStation, Xbox, Wii...)

d. network (ie. graph)

A. COVID-19 contact tracing.

e. histogram

A. Popularity of various coding languages in 2020

Q3. [2.5*2 = 5 points] Even though relational DBs are not being used as much anymore, compared to the past, eg. 80s and 90s [when they were the ONLY type of DB used!], their query language, SQL, still lives on, by being used for analyzing non-relational **data**. Name, and discuss, two examples of such 'living on' that we looked at.

A. Hive/HQL: a way to query large amounts of data (equivalent to a data warehouse) held in a Hadoop cluster.

CQL - Cassandra Query Language - for use with Cassandra, a column family DB.

SPARQL: SQL-like language to query RDF triples

Q4. Every major hospital is a huge repository of **data** - about illness, recovery, deaths, medications, surgical procedures, patient comfort, pain management.... For the questions below, assume you have data available from multiple (eg dozens) of hospitals across the country.

Q. []*5 = 5 points] Pick 5 different DM algorithms, and indicate what you would use each for (ie what type of data would you feed it, what would you predict/learn).

A.

Regression tree: using a terminally ill patient's vital signs, predict how long they will live.

Clustering: based on post-discharge patient survey data, group hospitals (eg into good, average, bad)

Logistic regression: to classify a patient as at-risk/not-at-risk, for a potential surgical procedure

SVM: to assess if a new drug would be safe/unsafe, on a patient

Neural network: use chest x-rays to train an NN, on COVID-19 detection

....

Q5. [2+1+2 = 5 points] BI is all about extracting/deriving value from massive amounts of existing (transactional) **data**, loaded into a warehouse.

Q. In such a context, what does 'rollup' mean, ie. how does it help in data analysis?

A. Rolling up involves breaking down, or its opposite - aggregating - data, along a dimension that involves a hierarchy (eg. time, space (location), product category, etc). We can get the 'big picture' by rolling up to the highest level in the hierarchy, or 'drill down' to the lowest level to see the breakdown.

Q. How is rollup analogous to viewing spatial data?

A. We can zoom in and out in a map, which is similar to rolling up or drilling down.

Q. Explain, with a simple example, how the SQL extension called ROLLUP help perform rollup.

A. Here is an example that uses a two-level hierarchy warehouses->products:

The answer doesn't need to contain SQL - a diagram/table illustrating subtotals and totals (similar to the output of the query shown above) is acceptable.

Q6. The big benefit of MapReduce is the dramatic acceleration of the processing of large volumes of **data.**

Q. [1 point] Sometimes, a local reduction is performed, at the mapping stage. Why?

A. To reduce network traffic, identical keys' values can be coalesced into a list that is then output.

Q. [1*4 = 4 points] List, and say a few words about, four alternative ways to specify data processing tasks.

A.

Java, for specifying mappers and reducers: a familiar language.

Python, for specifying mappers and reducers: also a familiar language, less verbose than Java.

Pig, for specifying data flow graphs.

Hive, for specifying data processing using the familiar SQL syntax.

Q7. [*5 = 5 points] As ML/DM matures, we are seeing a whole suite of tools/APIs/hardware... that help with the core task, of handling **data**.

Q. What tool helps visualize a TensorFlow graph (since the graph is specified via coding)?

A. TensorBoard

Q. What is a specification for 'packaging' neural network architectures?

A. CoreML/CreateML, Turi, ONNX...

Q.What are a couple of higher-level APIs that simplify neural network creation?

A. Keras, Pytorch (and mxnet, scikit-learn...)

Q. What are a couple of tools that offer code-free data processing?

A. WEKA, KNIME, RapidMiner, Orange...

Q. What are a couple of hardware-accelerated solutions for processing (NN) data [in addition to GPUs and TPUs - do not list these!].

A. Coral, Jetson Nano, Movidius NCS (and Pixy II, etc)

Q8. [1*5 = 5 points] With regards to COVID-19, comment on governance, security, ethics, privacy, compliance, all from a **data** perspective [in other words - how to do things right, what can be a problem...]. Think broadly: disease spread, tracking, hospitalizations, drugs, virus structure, vaccines...

A. This is a 'freeform' question, with a wide variety of correct answers - but, each answer does need to address an item mentioned above.

Governance: Use of a shared/common language for communicating data (eg about hospitalization rates, testing, etc). Others examples might include ensuring lineage/provenance of data (eg virus genome sequences), ensuring reliable data collection, etc.

Security: Guarding sensitive data such as virus genome sequences, drug trial results, drug discovery details...

Privacy: Safeguarding patient medical records.

Ethics: Ensuring validity of data analyses to include minority populations, choices related to opening up data which might speed up developing vaccines...

Compliance: Legally ensuring that hospitals report accurate, timely data; legally ensuring that testing and tracing protocols are enforced...

Q9. Good use of **data** implies good practices for storing (eg table design) as well as accessing it.

Q. [1 point] What is the single biggest reason, to performance tune an RDBMS' execution?

A. Speedy processing of queries (or just, 'speed').

Q. [1*4 = 4 points] Briefly discuss 2 SQL-based query tuning ways, and 2 non-SQL-based ones.

A. From the notes... in a WHERE condition, use simple operands for column names; with multiple conditions connected via AND, list the one most likely to fail, first [with OR, last]; rule-based optimizing; cost-based optimizing; even, 'creation and use of indices'.

Q10. [2.5*2 = 5 points] We index spatial **data**, for the same reason we index non-spatial data - for speedy retrieval. Name, and briefly discuss, two different ways of indexing spatial data.

A. From the notes: R tree, k-d tree, k-d-b tree, quadtree, even recursive curve schemes such as the z curve one.

CSCI 585 final exam
6/29/21; duration: 1 hour (plus 30 minutes for submission)

Please read the following carefully, before starting the test:

• the exam is **open** books/notes/devices - feel free to look up whatever you want!

• **you NEED to do Q0** (worth 0 points, but still!); from Q1 through Q10 (worth 5 points each), you can choose **any 7**, for a total of 35; if you want you can do **8, 9, or even, all 10** - we will ADD up ALL your scores from all the questions you answer, then CAP it at 35, meaning, a total > 35 will be set to 35 - amazing :) This is all exactly like how it was, in the midterm.

• there are no 'trick' questions, or ones with long calculations or formulae, and there's certainly nothing to memorize [it's all OPEN, duh :)] It doesn't mean the questions are trivial! Please do answer carefully: in other words, **answer WHAT IS ASKED**, otherwise you won't get points (eg if a question is ABOUT TM, don't DESCRIBE/DEFINE TM!)

• **please do NOT cheat** - this means NOT communicating with anyone via any device/medium/channel - you will get a 0, and be reported to SJACS, if you are found to have cheated; ANY attempt to get help from others in any form is a VIOLATION, as per https://policy.usc.edu/scampus-part-b/ (https://policy.usc.edu/scampus-part-b/), sections 11.11 through 11.14 [read it, if you are not familiar with it]

• **how much to write?** Use your judgment! In general, less is more, ie. don't feel the need to 'pad' your answers; that said, DO answer what's asked, including using a diagram if necessary

• when the time is up (60 minutes), stop your work, then spend the rest of time (30 minutes) on submission [students with DSP accommodations - your exam duration will be as per DSP determination] - **submitting past the deadline comes with a penalty**, because it is not fair to others if you go over when they don't

Good luck! Hope you enjoy answering the questions :) And, hope you do continue your involvement with 'data', long past this course!

Q0 [0 points]. DO turn this in - DO NOT omit doing so [you will LOSE 5 points if you skip this].

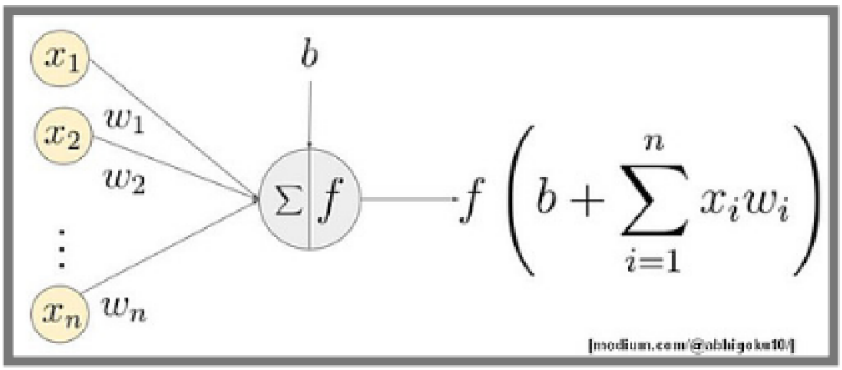
Please write the following line, and sign it - it is your acknowledgment of having read USC's policies on academic misconduct (https://policy.usc.edu/scampus-part-b/ (https://policy.usc.edu/scampus-part-b/), 11.11-11.14) and agreement to honor them.

I have read USC's standards on academic integrity, and agree to abide by them.

Q1 [2+2+1 = 5 points].

a. Supervised ML comes down to **'multi multilinear' curve-fitting, through existing data**. Explain, with a diagram.

A. Supervised ML is simply, a neural network (NN) - multiple neurons connected to each other, where each neuron, like a function, accepts multiple inputs, and returns a single output. Each neuron does the equivalent of multilinear curve-fitting, when it uses training data to 'learn' its weights (which are analagous to slopes) and bias (analogous to the 'y intercept'). Overall, this is therefore eqvt to **'multi'** (because the NN contains multiple neurons) **multilinear curve fitting** (performed by each neuron).



+1 for correct explanation of NN (+0.5 for partially correct answer)
+1 for connecting the NN to multilinear curve fitting (+0.5 for partially correct answer)

b. For all its amazing versatility, and seemingly magical applications, (supervised) ML relies on APPROXIMATION (not an exact 'solution') - why is that a desirable property? Explain using a couple of sentences.

A. Because we want to learn the OVERALL PATTERN in the input data, as opposed to learning JUST all the inputs and nothing else (which is called 'overfitting', which we want to avoid). This is similar to 2D line/curve fitting through data - we don't want the line/curve to pass through every input (x,y) data point, we only want the best approximation line/curve through all (x,y).

+2 for correct answer as above (+1 for partially correct answer)

c. What is an(other) example of a popular, existing system/technology/application/app where a somewhat similar technique as in question a. is employed [hint: curve, existing]? Explain in a couple of lines.

A. 'Fourier analysis/synthesis/decomposition', for music/audio generation - the input signal is decomposed into discrete sines and cosines, each with its own amplitude (and frequency). The input signal can be reconstructed from the sine and cosine components, similar to how, training data can be reconstructed from the neurons' learned weights and biases [along with sigmoids].

+1 for correct answer as above

Q2 [4+1 = 5 points].

a. When we tune a relational DBMS, we employ two techniques that similar to what we do in AI. What are the two techniques - explain each in a line or two, making comparisons with the corresponding AI technique.

A. The two techniques are 'rule-based query optimization' and 'statistically-based query optimization'. In rule-based, which is similar to an expert system in AI, we 'mine' a DB expert's knowledge (about data access from tables) to formulate rules. In statistics-based, which is similar to a neural network (trained using labeled data) in AI, we let the DB engine gather statistics related to tables and data access, and use those to formulate access plans.

[+1 for mentioning correct technique] * 2
[+1 for appropriate explanation (+0.5 for partially correct explanation)] * 2

b. Where possible, we use indices to speed up data access. WHAT exactly gives indices, their power (ie. how are they able to provide rapid access, compared to a raw table scan)?

A. Indices provide "pre-sorting", that makes it possible to rapidly access row/raw data - eg. by using a b-tree to access data in log(n) steps (eqvt to binary search), or in the case of spatial data and R-trees, using pre-sorted (spatially) bounding boxes to narrow down location searches by descending a hierarchy of regions [or with quadtrees, choosing one of four paths at each level of the quadtree hierarchy, or with a kd tree, using binary search to eliminate half the search area each time, etc].

+1 for correct answer as above (+0.5 for partially correct answer)

Q3 [3+1*2 = 5 points].

Ethical use of data and deployment of data-related applications is crucial. So is safeguarding privacy, ensuring security.

a. Briefly discuss three examples of how data might be unethically used - choose from medical data, insurance (home, health...), government, disease (eg COVID-19), law enforcement, education.

A. MANY answers possible! Eg. medical data might be sold by hospitals, insurance data might be gathered/sampled inadequately in order to bias results against certain populations, disease data might be faked or incorrectly interpreted on purpose, etc.

[+1 for 1 example of unethical usage of data] * 3

b. What might be consequences to individuals (eg you and me), if privacy is violated, security is breached? Given an example of each (violation, breach).

A. Medical privacy being violated might lead to insurance companies trying to deny coverage or employers finding a way to not hire someone; security breach can lead to financial loss, being impersonated (identity theft), being targeted/harassed, etc.

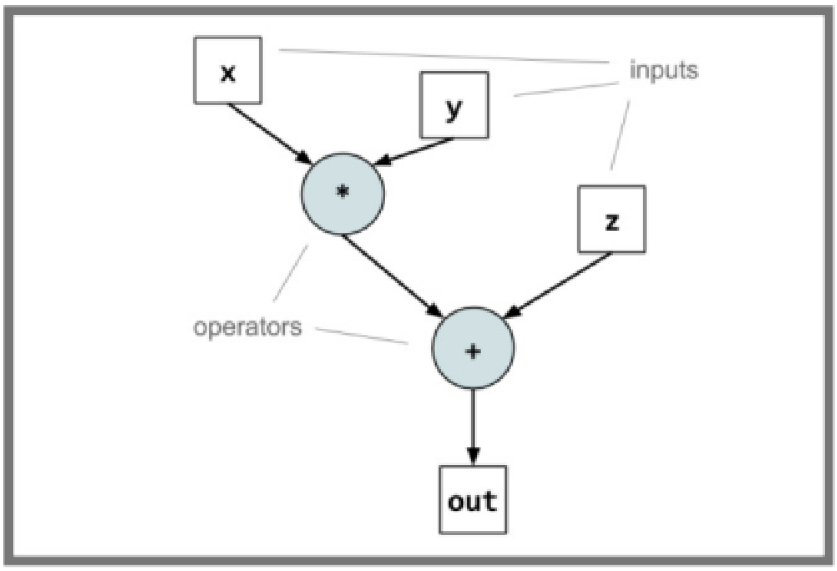
+1 for correct answer for violation (+0.5 partially correct)
+1 for correct answer for security breach (+0.5 partially correct)

Q4 [2+2+1 = 5 points].

Dataflow is an excellent data-processing technique, as you know.

a. Explain the principle (idea) behind dataflow, in a few lines, using a diagram.

A. Dataflow involves a graph of operators/nodes, and data flowing through (in input, output fashion) through the nodes. The key idea is this: only nodes that are 'affected'/'dirtied' on account of their inputs having been modified, will need to re-execute (which can cause their "downstream" nodes to be re-executed); the rest of the graph does NOT need to be re-executed, resulting in potentially vast computation savings. This is in contrast to a script where data is 'chained' using function calls and return values, where ALL "nodes" (functions) get executed, whenever there is any (even trivial) change in any part of the script, eg. even in the last stages.



+2 for correct answer as above (+1 for partially correct answer)

b. What makes dataflow advantageous, over conventional process (eg using standard function calls in an application)?

A. The fact that only affected/damaged/'dirtied' sections of the dataflow graph need to be re-executed - this can result in enormous execution savings.

+2 for correct answer as above (+1 for partially correct answer)

c. To whom is 'visual dataflow' (eg KNIME, RapidMiner) useful?

A. To non-programmers, eg. business analysts - they can simply wire together "boxes" (operators/nodes) to construct a data analysis (or any other type of) pipeline, without writing code; the graph becomes "self documenting" as well, where it is easy to understand what is happening to the data - as opposed to having to comprehend 'raw' code.

+1 for correct answer as above (+0.5 for partially correct answer)

Q5 [2+3 = 5 points].

a. Queen Padme is right! What does modern BI do instead, and why the need to change?

A. Modern BI does ELT as opposed to ETL - the change is on account of streaming, and semi-structured, data that's prevalent in today's data-processing environments (rather than tabular, relational data).

+1 for correct answer of what BI does (+0.5 for partially correct answer)
+1 for correct answer why the need to change (+0.5 for partially correct answer)

b. When can a star schema not be able to be transformed to a snowflake schema? Explain, in a few lines, with a small example.

A. Where is there no hierarchy that can be constructed from fact columns! Eg. a student table with [name, GPA, major, advisor] is not suitable to construct a snowflake schema from [except possibly for 'major' - it can have a 'school' type at a higher level]. This does lead to a conceptual question/argument - can ANY column/feature/attribute be generalized recursively, to lead to a snowflake-like hierarchy? Often yes, but not always [like in our simple 'student' example].

+1.5 for correct answer as above (+1 for partially correct answer)
+1.5 for correct example

Q6 [2+2+1 = 5 points].

a. Explain this: 'The popular, powerful, MapReduce paradigm, has lower level components as well as higher level ones".

A. The lower level components comprise of a mapper and reducer; higher level components are abstractions, such as dataflow graph (eg. Pig), relational-like processing (eg. Hive).

+1 for correct answer of lower level components
+1 for correct answer of higher level components

b. When is each (higher level, lower level) component useful (ie. which to use when)?

A. Higher level components are adequate for straightforward tasks where the abstraction (eg dataflow) is adequate to express the data processing. But if custom functionality is required, they can always be implemented at the lower level, ie via a mapper and reducer function. To put it differently, 'routine' cases benefit from higher-level simplicity; conversely, 'special' cases are possible, via lower-level coding.

+1 for correct answer of higher level component usefulness
+1 for correct answer of lower level component usefulness

c. What would have been the consequence, had GFS/HDFS not been invented to be part of MapReduce/Hadoop?

A. It would not be possible to ignore file system details related to data (fragment) locations, making the construction and execution of mappers and reducers, much more complex; also, higher level abstractions such as Pig might not have been feasible at all.

+1 for correct answer as above (+0.5 for partially correct answer)

Q7 [2+2+1 = 5 points].

a. Why is spatial data handled distinctly from other data types (eg. number, string), ie. what makes it unique?

A. Spatial data is SCALE dependent, ie we can zoom in/out to reveal/hide detail.

+2 for correct answer as above (+1 for partially correct answer)

b. What makes spatial data indexing different, compared to indexing other data types?

A. The indexing needs to involve (take into account) 2D or 3D (or 4D) data, so it can't involve simple hashing, or b-tree construction, etc.

+2 for correct answer as above (+1 for partially correct answer)

c. Why do we care so much about **geo**spatial data?

A. We humans are visual creatures, with an inborn sense of "location" - so it is intuitive for us to quickly grasp data overlaid on to maps, ie geospatial data viz.

+1 for correct answer as above (+0.5 for partially correct answer)

Q8 [1*3 + 1*2 = 5 points].

a. Charles Minard, Florence Nightingale, and John Snow (who created the 1854 London cholera map) created amazing data visualizations without any modern tools, especially computers. Given today's tech, were they alive today, what would each person do, to make their visualizations better (ie how would they improve their creations)? Mention at least one feature/improvement for each.

A. Minard would have provided checkboxes for each variable to be independently turned on/off, eg temperature, troop count, etc. Florence would have created an animating or draggable timeline to indicate the change of data (proportions of mortality) over time. John would have made the London map zoomable and pannable, and possibly augmented with mortality data (eg age of each dead person).

+1 for correct answer for Minard
+1 for correct answer for Florence
+1 for correct answer for John

b. What is an example where you'd use AR (augmented reality) for data viz? What about VR (virtual reality) - what would you visualize? Explain each, in a line or two.

A. AR could be used to overlay occupancy data over buildings, nutrition data over food containers (in a supermarket for ex) or even LinkedIn or social media profile data over a person, performance data over a machine (eg turbine), medical data over a patient, etc. VR could be used to visualize spatial data, or any table data, by being 'in' the data (ie having the data spatially laid out all around the viewer, for example).

+1 for appropriate answer for AR
+1 for appropriate answer for VR

Q9 [2+3 = 5 points].

"Graph DB processing, via partitioning, is not as simple as horizontal fragmentation of a collection of documents".

a. Why would we horizontally fragment a document collection in the first place?

A. In order to speed up processing (in parallel, of the fragments).

+2 for correct answer as above (+1 for partially correct answer)

b. Why is graph data partitioning and analyzing, not (as) simple?

A. Because the partitions would not be independent in general (only exception - when the graph is already partitioned into multiple subgraphs) - there would be lost edges across partitions [ie. they would become semi-edges] - so, there needs to be extra processing, eg. via BSP supersteps.

+3 for correct answer as above (+1.5 for partially correct answer)

Q10 [5 points].

There's always going to be 'Big Data', and 'Data Science' - but things do change... Briefly (using a line each!) answer the following.

a. What are a couple of future sources of 'big' data?

A. More powerful telescopes and particle accelerators, increased amount of medical data etc.

+1 for appropriate answer as above

b. How might we store increasingly large quantities of data?

A. Using DNA (or holographic storage, or even as electron spin, etc.).

+1 for correct answer as above

c. Broadly speaking, what are the (one word) names given to local, and remote, processing of (big) data?

A. Edge, cloud.

+1 for correct answer as above

d. What overall trend do you see, in the wide variety of data-processing-related tools we discussed?

A. Increased abstraction, visually-oriented, self-service tools oriented towards casual users (eg managers, analysts, business owners etc).

+1 for correct answer as above

e. What are you going to do about it [the trend, that is]?

A. "If you can't beat them, join them" :) Become good at using the available tools, acquire domain/business knowledge to function more as an analyst than a developer, or seek employment at companies that create such tools...

+1 for correct answer as above